Efficient and Global Optimization-Based Smoothing Methods for Mixed-Volume Meshes

Dimitris Vartziotis[1,2,⋆] and Benjamin Himpel[1]

[1] TWT GmbH Science & Innovation, Department for Mathematical Research & Services, Bernhäuser Straße 40–42, 73765 Neuhausen, Germany
dimitris.vartziotis@nikitec.gr

[2] NIKI Ltd. Digital Engineering, Research Center, 205 Ethnikis Antistasis Street, 45500 Katsika, Ioannina, Greece

Summary. Some methods based on simple regularizing geometric element transformations have heuristically been shown to give runtime efficient and quality effective smoothing algorithms for meshes. We describe the mathematical framework and a systematic approach to global optimization-based versions of such methods for mixed volume meshes, which generalizes to arbitrary dimensions. We also identify some algorithms based on simple regularizing geometric element transformation optimizing certain global algebraic mesh quality measures.

Keywords: smoothing, quality metric, quality measure, finite element method, global optimization, optimization-based method, GETMe.

1 Introduction

In the context of the finite element method mesh quality affects numerical stability as well as solution accuracy of this method [28]. The class of geometric element transformation methods (GETMe) consists of mesh smoothing methods based on simple geometric element transformations. In [34] such a smoothing algorithm is introduced for tetrahedral meshes based on shifting vertices of a tetrahedron by the opposing face normals, normalized to be scaling-invariant. Since it has been tested numerically and extended to other volume types in a series of papers [34, 31, 32, 30, 33], our focus lies on a rigorous theoretical understanding of certain GETMe algorithms and we will postpone numerical experiments to future publications.

In [29] we show that the mean volume can be viewed as a quality measure for tetrahedron, pyramid, prism and hexahedron, and that the discretization

[⋆] Corresponding author.

J. Sarrate & M. Staten (eds.), *Proceedings of the 22nd International Meshing Roundtable*,
DOI: 10.1007/978-3-319-02335-9_17, © Springer International Publishing Switzerland 2013

of its gradient flow is a natural generalization of the aforementioned tetrahedral GETMe algorithm. It enables us to prove that it regularizes certain polyhedron types and therefore is a local optimization-based smoothing and untangling method. For tetrahedra, this quality measure is related to the mean ratio quality measure [16]. Note that our generalization to other polyhedron types is inherently very different from previous extensions via dual polyhedra to hexahedra, prisms and pyramids [31, 32] as well as the mean ratio criterium for other polyhedra [16].

Several numerical tests have shown, that the GETMe approach combines the advantages of Laplacian smoothing variants in terms of runtime efficiency with the mesh quality effectiveness of a global optimization-based smoothing [33]. While we have shown in [29] that our generalizations of the geometric element transformations in [34] regularize certain polyhedron types, we have not touched upon the subject of global optimization for meshes, which is of particular interest to the meshing community. The main goal of our work is to describe the mathematical framework, which allows us to systematically construct global optimization-based GETMe smoothing methods reminiscent of the transformation in [34]. As a result we identify some simple regularizing geometric element transformations, which optimize certain algebraic mesh quality measures. Clearly, if the global quality measure does not accurately describe the desired quality of a mesh, the smoothing methods are not useful, but all of them are runtime efficient. This approach allows for rigorous analysis, natural generalizations and problem-specific extensions. We hope, that it will also serve as a basis for new runtime efficient and quality effective smoothing algorithms.

Mesh smoothing methods can be classified [20, 21, 35] as geometry-based [10, 34], optimization-based [4, 12, 22, 25, 18, 23, 5], phyics-based [24] and combinations thereof [6, 11, 7]. In order to be more effective, these methods should be combined with topological modifications [3, 13, 15]. Optimization-based methods often lend themselves to untangling algorithms [17, 19, 14, 1] and can be further differentiated as local or global optimization-based. Geometry-based methods like the Laplacian [10] and GETMe [34] smoothings have the advantage of being fast, but have always been heuristic until we proved in [29], that a minor variation of the GETMe method presented in [34] generalizes to a local optimization-based mesh smoothing and untangling method for mixed-volume meshes. In the present work we present GETMe methods that optimize global objective functions containing barriers, which—at least in the case of triangular and tetrahedral meshes—prevent non-inverted meshes from becoming inverted. In short, we bridge the gap between geometry-based and global optimization-based smoothing methods.

In Sect. 2 we introduce the notions of mesh quality and show, how we can turn the mean volume function into a useful volume element quality measure. Sect. 3 describes how the GETMe algorithm in [34] fits into our mathematical framework. In Sect. 4 we see, how the mean volume function gives the simple geometric element transformation underlying the tetrahedral GETMe algorithm, and construct other simple geometric element transformations based on quality measures. Most importantly, Theorem 2 shows, when homogeneous degree d vector fields give global optimization-based scaling-invariant smoothing methods. In particular, this gives a way to construct simple geometric element transformations optimizing certain algebraic mesh quality measures. In Sect. 5 we generalize our results in various ways. We give concrete formulas generalizing the tetrahedral geometric element transformation in [34] to the pyramid, the prism and the hexahedron, and briefly discuss the generalization to meshes in arbitrary dimensions, the isoperimetric quotient as a volume element quality measure, whose discretized gradient flow gives yet another GETMe smoothing method, and shape preservation.

2 The Mathematical Framework

It is not apparent why a simple geometric transformation for tetrahedra given by shifting vertices by the opposing face normals should yield such a reliable and efficient smoothing algorithm for meshes. An elegant mathematical explanation provides a way to systematically develop and analyze some of these promising GETMe algorithms further. This is the motivation for creating a suitable mathematical framework, which was used in the proof of the regularizing behavior for the transformation in [34].

A global optimization-based smoothing method increases in each iteration step the value of a (global) quality function for meshes by repositioning the vertices of a mesh. A word of caution up front for mathematicians: local and global optimization refers to whether a mesh is optimized locally in a neighborhood of a vertex or globally for the entire mesh. In particular, global optimization generally refers to the search for a local optimum of a global quality function. Global optimization-based smoothing methods are known to yield meshes with superior element quality [8, 5]. Our goal is to find simple geometric element transformations, which give global optimization-based GETMe algorithms, in order to combine the mathematical advantages of global optimization with the speed of GETMe.

In order to keep the mathematical overhead to a minimum, we adapt the notation from [8] and refer to [29] for a complete treatment offered to the interested reader.

2.1 Element and Mesh Quality

A mesh consists of an ordered set of n vertices V and $|E|$ volume elements E. Let us write V and E as a tuples $(v_1, \ldots, v_n)$ and $(e_1, \ldots, e_{|E|})$ respectively. Let $x_v \in \mathbf{R}^3$ denote the coordinates for the vertex $v \in V$, so that $x = (x_{v_1}, \ldots, x_{v_n}) \in \mathbf{R}^{3 \times n}$ is the tuple of all vertex coordinates. Each volume element $e \in E$ consists of an ordered set V_e of n_e vertices of V and the edges between these vertices. If the edges are fixed, then we will simply identify

$e = V_e$. Note that the order of vertices determines a preferred orientation for e. Let $x_e \in \mathbf{R}^{3\times n_e}$ be the collection of coordinates for e. Associated with each element e is an element quality measure $q_e\colon\ \mathbf{R}^{3\times n_e} \to \mathbf{R}$ as a continuous function of the vertex positions, where a larger value of q_e indicates a higher quality element. The overall quality of a mesh is measured by a continuous function $Q\colon\ \mathbf{R}^{|E|} \to \mathbf{R}$, taking as input the vector of volume element qualities and combining them in a certain way, for example by taking the arithmetic mean, but even combinations depending on the size, location, direction, and other properties are conceivable.

In order to simplify the notation, we call the function

$$\begin{aligned} q\colon\ \mathbf{R}^{3\times n} &\to \mathbf{R}\ , \\ x &\mapsto Q\left(q_{e_1}(x_{e_1}), \ldots, q_{e_{|E|}}(x_{e_{|E|}}))\right) \end{aligned} \tag{1}$$

the mesh quality measure. In general, useful quality measures possess other properties in addition to continuity, like invariance under translation, scaling, rotation and reflection [16]. Therefore, let us only consider mesh quality measures, which are at least invariant under translation and scaling. Alternatively, we can consider the coordinates $x \in \mathbf{R}^{3\times n}$ up to scaling and translation. By [29, Sect. 2] the resulting space is simply a $(3n-4)$–dimensional sphere $S^{3n-4} \subset \mathbf{R}^{3n-3}$, if in addition we disregard the one-point meshes, where all vertex coordinates are equal. We can describe this space concisely as a quotient of a subspace of $\mathbf{R}^{3\times n}$ by the equivalence relation given by translation and scaling. This observation allows us to view a translation- and scaling-invariant quality measure $q\colon\ \mathbf{R}^{3\times n} \to \mathbf{R}$ as a function on this $(3n-4)$–dimensional sphere embedded in $\mathbf{R}^{3\times n}$. Since spheres are compact manifolds without boundary and a global optimization-based method is essentially the gradient flow of a function on this sphere, we can now unleash the power of geometric analysis to study this method.

By the above argument, we have not only gained a different angle from which we can view and study quality measures, but an entirely new approach to constructing optimization-based quality measures. In fact, any function on a $(3n-4)$–sphere embedded in $\mathbf{R}^{3\times n}$ representing all meshes up to translation and scaling gives rise to a translation- and scaling-invariant quality measure, whose gradient flow will give a global optimization-based smoothing method through the method of steepest descent.

2.2 The Mean Volume Function

The signed volume of a tetrahedron with vertex coordinates $x = (x_1, \ldots, x_4) \in \mathbf{R}^{3\times 4}$ is given by

$$\mathrm{vol}(x) = \frac{1}{6}((x_2 - x_1) \times (x_3 - x_1)) \cdot (x_4 - x_1)\ . \tag{2}$$

The orientation of the tetrahedron and therefore the sign of the volume function is determined by the order of vertices. Notice, that it is a well-defined function for all meshes with the structure of a tetrahedron. This function naturally extends to convex polyhedra by first triangulating them, in particular to hexahedra, prisms and pyramids. By averaging over all possible triangulations, we can extend the volume function to a mean volume function on all meshes with the edge structure of a convex polyhedron and again denote it by vol.

There are alternative definitions for this mean volume function, but this is the most convenient one for our purpose. For example, in the case of a pyramid mesh $e = (v_1, \dots, v_5)$, where v_5 is the apex, we only have the two different triangulations depicted in Fig. 1. For a detailed discussion of triangulations and the mean volume function see [29, Sect. 5].

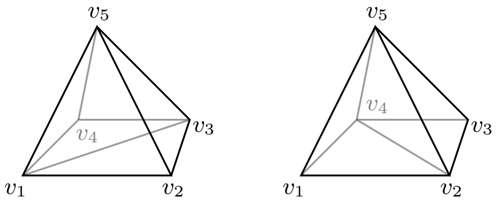


Fig. 1 The two different triangulations of a pyramid

Clearly, the mean volume function is translation-invariant, but not scaling-invariant. However, if we consider its restriction to a $(3n-4)$–sphere representing all meshes with the structure of a fixed convex polyhedron with n vertices up to translation and scaling, this yields a scaling- and translation-invariant quality measure q_e for polyhedra, which is essentially maximized by regular polyhedra [29]. To be more concrete, the quotient map by translation and scaling viewed as a projection to the submanifold

$$N := \{x \in M \mid \|x\| = 1 \text{ and } x_* := \sum_{i=1}^{n} x_i = \mathbf{0}\} \subset M \tag{3}$$

is given by

$$\begin{aligned} \pi\colon\ \mathbf{R}^{3\times n}\backslash\{(x_0, \dots, x_0) \mid x_0 \in \mathbf{R}^3\} &\to N\ , \\ (x_1, \dots, x_n) &\mapsto \frac{(x_1 - x_*, \dots, x_n - x_*)}{\|(x_1 - x_*, \dots, x_n - x_*)\|}\ , \end{aligned} \tag{4}$$

and the diffeomorphism from N to $S^{3n-4} \subset \mathbf{R}^{3(n-1)} \cong \mathbf{R}^{3\times(n-1)}$ is given by

$$(x_1, \dots, x_n) \mapsto \frac{1}{\|(x_1, \dots, x_{n-1})\|}(x_1, \dots, x_{n-1})\ . \tag{5}$$

Similarly, let $N_e \subset \mathbf{R}^{n_e}$ be the sphere corresponding to e.

2.3 The Mean Ratio Metric

Algebraic mesh quality measures had been introduced by [16], and have been influential for smoothing, optimization and edge swapping techniques. These quality measures also fit nicely in our framework. Let us consider the mean ratio quality measure. It is given by

$$q(x_e) := \frac{3\det(S)^{2/3}}{\|S\|_F^2}, \tag{6}$$

with $\|S\|_F := \sqrt{\operatorname{tr}(S^{\mathrm{t}}S)}$ denoting the Frobenius norm of the matrix $S := DW^{-1}$. Here D represents the difference matrix given by

$$D := (x_2 - x_1, x_3 - x_1, x_4 - x_1) \quad \text{for } \det(x_e) > 0 \tag{7}$$

and W denotes the difference matrix of a reference tetrahedron. It holds that $q(x_e) \in [0,1]$, where very small values indicate nearly degenerated elements and larger values elements of good quality. In particular it holds that $q(x_e) = 1$, if x_e is regular.

When the desired shape is a regular tetrahedron, the Frobenius norm is simply a way to make the volume scaling-invariant while preserving the translation- and rotational invariance. The main reason for this choice is that this matrix norm is easily implemented and efficient, and has therefore been used in quality measures. Now, $\det(S) = \det(D)\det(W^{-1}) = \operatorname{vol}(x_e)/w$, where $w = \det(W)$ is constant. Furthermore $\det(D) = 6\operatorname{vol}(x_e)$, when e is a tetrahedron. We can rewrite

$$C\,(q(x_e))^{3/2} = \operatorname{vol}\left(\frac{1}{\|S\|_F}x_e\right) \quad \text{for some constant } C = C(W) > 0. \tag{8}$$

The volume function restricted to the submanifold of M diffeomorphic to a $(3n-4)$–sphere given by

$$\{p \in M \mid x_* = \mathbf{0} \text{ and } x_e = \|S\|_F\} \subset M \tag{9}$$

is therefore an equivalent way of describing the mean ratio quality measure.

With the results from [29] in hand, we see, that [34] starts with the same volume function, but normalizes its gradient rather than the quality measure, so that it is scaling-invariant. Furthermore, we have proven, that this scaling-invariant gradient optimizes the quality measure given by a certain normalized volume. In effect, the aim of both methods is the optimization with respect to the volume function. This is a heuristic explanation, why Mesquite [5] and GETMe [34] give meshes of similar quality.

3 GETMe

The original GETMe algorithm for tetrahedral meshes [34] is based on shifting vertices by opposing face normals. Let us give a few details in order to describe how this algorithm fits into the mathematical framework.

3.1 Scaling-Invariance

For a tetrahedral volume element $e = (v_1, \ldots, v_4)$ consider the vector

$$X_e = \begin{bmatrix} X_{e,1} \\ X_{e,2} \\ X_{e,3} \\ X_{e,4} \end{bmatrix} \coloneqq \begin{bmatrix} (x_4 - x_3) \times (x_3 - x_2) \\ (x_4 - x_1) \times (x_1 - x_3) \\ (x_4 - x_2) \times (x_2 - x_1) \\ (x_1 - x_2) \times (x_2 - x_3) \end{bmatrix} \in \mathbf{R}^{3 \cdot 4} \cong \mathbf{R}^{3 \times 4} \;, \tag{10}$$

where we identify the column vector of (column) vector entries with the row vector of the same vector entries. The GETMe algorithm without the imposed scaling-invariance applied only to this one volume element e transforms its coordinates via

$$\begin{aligned} \mathbf{R}^{3\times 4} &\to \mathbf{R}^{3\times 4} \;, \\ x_e &\mapsto x'_e = x_e + \sigma X_e \end{aligned} \tag{11}$$

for some parameter $\sigma > 0$ independent of e. The tetrahedral GETMe smoothing introduced in [34] applied to an entire mesh then shifts x_i for $i = 1, \ldots, n$ by averaging all (vector) coordinates of X_e over all e containing v_i. More precisely, if $V = (v_1, \ldots, v_n)$ are the vertices of a mesh with coordinates x and $e = (w_1, \ldots, w_4)$ is a tetrahedral volume element within the mesh, then let us introduce the notation

$$\begin{aligned} \phi_e\colon \quad & \mathbf{R}^{3\times 4} \hookrightarrow \mathbf{R}^{3\times n} \;, \\ & (y_1, \ldots, y_4) \mapsto (x_1, \ldots, x_n) \text{ where } \begin{cases} x_i = y_j & \text{if } v_i = w_j \;, \\ x_i = \mathbf{0} & \text{else} \end{cases} \end{aligned} \tag{12}$$

where $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, 4\}$, and let λ_i be the number of volume elements containing v_i. If we set

$$X \coloneqq \operatorname{diag}(\lambda_1^{-1}, \ldots, \lambda_n^{-1}) \sum_{e \in E} \phi_e(X_e) \;, \tag{13}$$

then the original GETMe transformation is given by

$$\begin{aligned} \mathbf{R}^{3\times n} &\to \mathbf{R}^{3\times n} \;, \\ x &\mapsto x' = x + \sigma X \;. \end{aligned} \tag{14}$$

However, in order to make it scaling-invariant, we need to modify X. We have arrived at a subtle issue, where our solution in [29] and the one in [34] differ, and which we will encounter again in Sect. 4.2. This can only be appreciated when considering GETMe from a mathematical rather than a heuristic point of view. In practice it does not make much of a difference, how we make the algorithm scaling-invariant, but there is a preferred way within our mathematical framework. In [34] each face normal in $X_{e,j}$ was divided by the square root of its norm $\|X_{e,j}\|$, which seemed to work very well for all practical purposes. In the language of our mathematical framework on the other hand, the vector

$$X \in \mathbf{R}^{3\times n} \tag{15}$$

determines the dynamical behavior on the $(3n-4)$–sphere rather than the vectors $X_{e,j} \in \mathbf{R}^3$ individually, which is why we would prefer not to change the direction of X. Therefore, we divide X by the square root of its norm $\|X\|$, which results in a scaling of X_e depending on the size of the volume element e. In summary, if we define the normalization function

$$\begin{aligned} \Psi\colon\ \mathbf{R}^{3\times n} &\to \mathbf{R}^{3\times n}\ , \\ X &\to \begin{cases} \frac{1}{\sqrt{\|X\|}}X & \text{if } X \neq \mathbf{0} \\ 0 & \text{if } X = \mathbf{0}\ , \end{cases} \end{aligned} \tag{16}$$

then we consider the transformation T_σ where

$$\begin{aligned} T_\sigma\colon\ \mathbf{R}^{3\times n} &\to \mathbf{R}^{3\times n}\ , \\ x &\mapsto x' = x + \sigma\,\Psi(X) \end{aligned} \tag{17}$$

as our variation of the original GETMe smoothing algorithm [34].

If we take a closer look, we choose a scaling different from [34] in two ways:

1. in the normalization for each individual volume element, and
2. in the averaging procedure over all volume elements for the mesh.

From a global point of view we are closer to the initially conceived GETMe algorithm for tetrahedral meshes without the imposed scaling-invariance than we have been in [34], because the vector $\Psi(X)$ points in the same direction as X. As we will see in Sect. 4.1, this entirely rigorous approach to the algorithm presented [34] is not very useful, because the latter includes other scaling methods and additional control mechanisms in order to exhibit such favorable behavior.

3.2 Face Normals

As we have seen, face normals play an essential role in the tetrahedral GETMe approach. By the face normal of an oriented triangle we simply mean the cross product of two edge vector compatible with the orientation. In order to simplify notation, let us define face normals for arbitrary polygonal curves in $\mathbf{R}^3$ by

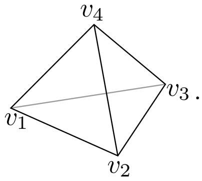
$$\begin{aligned}\nu(1,\dots,k) :=&(x_2-x_1)\times(x_3-x_1)+(x_3-x_1)\times(x_4-x_1)+\dots\\ &+(x_{k-2}-x_1)\times(x_{k-1}-x_1)+(x_{k-1}-x_1)\times(x_k-x_1) \quad (18)\\ =&x_1\times x_2+x_2\times x_3+\dots+x_{k-1}\times x_k+x_k\times x_1\ .\end{aligned}$$

Clearly, if (x_1,x_2,x_3) are the coordinates of a triangle in $\mathbf{R}^3$, then

$$\nu(1,2,3)=(x_1-x_2)\times(x_2-x_3) \qquad (19)$$

is the usual face normal considered in the previous section. In particular, we can write

$$X_e=\begin{bmatrix}\nu(4,3,2)\\ \nu(4,1,3)\\ \nu(4,2,1)\\ \nu(1,2,3)\end{bmatrix} \text{ for a tetrahedral element } e=$$



. (20)

For polygonal curves in a hyperplane of $\mathbf{R}^3$, ν is a natural generalization of the face normal from triangles to the planar surface enclosed by the curve. The direction of ν can be determined by the right-hand (grip) rule. For arbitrary polygonal curves in $\mathbf{R}^3$, ν corresponds to the sum of the face normals of an arbitrary triangulated surface, whose boundary is the curve. For more details on face normals, please consult [29, Sect. 3].

4 An Efficient and Optimization-Based Approach

Often, quality effective smoothing methods are developed using global optimization. In fact, global optimization-based methods yield meshes of superior quality with respect to the corresponding global quality measure. We show how to find they way back from GETMe to the basics of global optimization via reverse engineering, because the resulting algorithms will combine the best of both worlds, being not only very efficient, but also effective with respect to the choice of global quality function. We will also discuss how previously tested algorithms fit into this framework.

4.1 Suitable Mesh Quality Functions

As we have mentioned in Sect. 2.1, any function on N can be viewed and used as a global mesh quality function, whose gradient flow will find local maximums. There are lots of different ways of combining element quality functions in order to get a global one for meshes. Our immediate goal is to find potentially useful measures, whose gradients preserve the GETMe characteristic.

The most naive option is to add all the element quality functions up. A straight-forward computation as in [29, Sect. 3] shows, that the mean volume for meshes defined in Sect. 2.2 satisfies

$$6\nabla \operatorname{vol} = X \ , \tag{21}$$

where ∇ vol is the gradient vector field of vol. The method of steepest descent applied to six times the negative mean volume clearly yields the transformation

$$x' = x + \sigma X \quad \text{for } \sigma \text{ sufficiently small.} \tag{22}$$

Up to the averaging process at each node, this is the original GETMe algorithm [34] without the imposed scaling-invariance, relaxation, weighted averages and any further control mechanisms. This might provide some confidence, that this would yield an efficient and effective GETMe algorithm. Instead, this uncovers a big drawback of the core GETMe transformation: inner vertices are fixed when shifting via X, because the mean volume is independent of the location of the inner vertices. More concretely, consider a regular tetrahedra with a single inner vertex and four smaller tetrahedra whose edges connect the outer vertices $v_1, \ldots, v_4$ with the inner vertex v_5 as in Fig. 2.

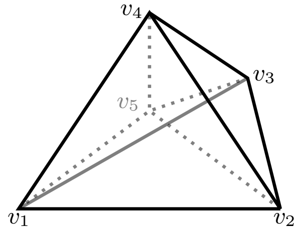


Fig. 2 The triangulation of a regular tetrahedra with one inner vertex

The inner point is fixed by the flow of X and and stays fixed after averaging the corresponding face normals, no matter where it is located, because the opposite face normals add up to the trivial vector. One could argue, that by rescaling the regular tetrahedra, the inner point moves towards the center. However, if we have more than one inner point, all of them would move towards the center. Therefore, the correct way to rescale the resulting mesh is by projecting the boundary vertices onto the original boundary surface, even though this means, that the inner vertices are fixed. It is therefore surprising, that the GETMe algorithm in [34] works as well as it does. In any case, a systematic approach might lead to a more stable and mathematically predictable algorithm without the need of control mechanisms.

Instead of simply using the mean volume as a global mesh quality function, the sum of the scaling-invariant element quality functions q_e

$$q(x) = \sum_{e \in E} q_e(x_e) \tag{23}$$

given by first projecting x_e to N_e would be a more promising albeit more complicated approach, which we will not pursue at this point. However, there are other ways of combining the mean volumes to give potentially useful mesh quality functions. Let us consider the restriction of

$$q_1(x) = \prod_{e \in E} \mathrm{vol}(x_e) \quad \text{and} \quad q_2(x) = -\sum_{e \in E} \frac{1}{\mathrm{vol}(x_e)} \tag{24}$$

to N. We compute

$$\nabla q_1 = k_1(x) \sum_{e \in E} \frac{1}{\mathrm{vol}(x_e)} X_e \quad \text{and} \quad \nabla q_2 = k_2(x) \sum_{e \in E} \frac{1}{\mathrm{vol}(x_e)^2} X_e \ , \tag{25}$$

where

$$k_1(x) = q_1(x) \quad \text{and} \quad k_2(x) = 1 \ . \tag{26}$$

Obviously, vanishing (and negative) mean volume is problematic. For convenience, let us assume that all of our volume elements have positive mean volume. In particular, all valid volume elements should satisfy this mild assumption. For tetrahedra, positive mean volume is equivalent to them being valid. In particular, both quality measures ensure in the case of tetraehedra, that they stay valid under sufficiently small transformations using the gradient.

We observe, that ∇q_1 and ∇q_2 look similar, and therefore we expect them to behave similarly. Furthermore, in the above concrete example of the regular tetrahedron triangulated by 4 tetrahedra, the inner point is not fixed by the gradient flow. Instead, it moves away from the smaller (and more irregular) tetrahedra towards the center. In general, the resulting transformation seems to prefer volume element distributions, which are similar in size and at the same time as regular as possible. As Fig. 4 for the two-dimensional analogue of q_1 shows, this does not create good meshes for the finite element method. There are a lot of other conceivable mesh quality functions and combinations of them. In this paper we emphasize the mathematical framework rather than specific algorithms.

4.2 Scaling-Invariance

Whatever the gradient flow does, it would be nice to have a scaling-invariant flow of some related vector field. This can be achieved just like we have done in [29] by introducing an appropriate factor. The proof of the following result is entirely analogous to the proofs of [29, Lemma 3.3 and Theorem 4.2], but it is more general.

Theorem 1. *Let ∇_π denote the gradient with respect to the induced submanifold metric $\langle \cdot, \cdot \rangle_\pi$ on N. Let X be a homogeneous vector field of degree d and q a quality measure satisfying*

$$d(\pi)(X_x) = c_x \nabla_\pi q|_x \quad \textit{and} \quad c_x > 0 \quad \textit{for } x \in N. \tag{27}$$

Consider $\Psi\colon \mathbf{R}^{3\times n} \to \mathbf{R}^{3\times n}$ *given by*

$$\Psi(X) = \|X\|^{(1-d)/d} X\ . \tag{28}$$

Then the vector fields on N *given by*

$$Y := D(\pi)(X) \quad \textit{and} \quad \tilde{Y} := D(\pi)(\Psi(X)) \tag{29}$$

start and end at singularities of Y*, and the vector field* $\tilde{Y}$ *on* N *corresponds to* $\Psi(\nabla q)$ *on the quotient manifold* $\mathbf{R}^{3\times n}\backslash\{(x_0,\ldots,x_0) \mid x_0 \in \mathbf{R}^3\}/\sim$.

In particular, the above theorem applies to q_i defined in Sect. 4.1, where $c_x = (k_i(x))^{-1}$ and $X_i = c_x \nabla q_i|_x$ is homogeneous of degree $-(3i+1)$.

In [29, Theorem 4.2] we have been content with results about the dynamical behavior for volume elements. If we look more closely, our proof implicitly shows something else: on the mesh given by the single volume element, the GETMe algorithm with our normalization is optimization-based. The same is true above. The vector fields Y and $\tilde{Y}$ yield optimization-based GETMe transformations.

4.3 Global Optimization-Based Algorithm

Since there are many quality functions which might yield global optimization-based algorithms, let us consider an arbitrary homogeneous degree d vector field X and a quality measure q satisfying

$$\mathrm{d}(\pi)(X_x) = c_x \nabla_\pi q|_x \quad \text{and} \quad c_x > 0 \quad \text{for } x \in N\ . \tag{30}$$

In particular the following theorem applies to q_i in Sect. 4.1 and similar algebraic quality measures.

Theorem 2. *The GETMe operator given by*

$$\begin{aligned} T_\sigma\colon\ & N \to N\ , \\ & x \mapsto x' = \pi(x + \sigma\,\Psi(X)) \quad \textit{for } X = \sum_{e\in E} X_e\ . \end{aligned} \tag{31}$$

is a global optimization-based smoothing method for mixed-volume meshes as long as $\sigma > 0$ *sufficiently small, which is invariant under translation and scaling. More precisely, for the mesh quality function* q *and for any* $x \in N$ *we have*

$$q(T_\sigma(x)) > q(x) \quad \textit{for } \sigma > 0 \textit{ sufficiently small.} \tag{32}$$

Proof. Fix $x \in N$, and let $x_\sigma = x + \sigma\,\Psi(X) \in \mathbf{R}^{3\times n}$. Since $\pi(x) = x$ and

$$\frac{d}{d\sigma}(x_\sigma) = \Psi(X) = \|X\|^{(d-1)/d} X \ , \tag{33}$$

the chain rule gives

$$\frac{d}{d\sigma} q(T_\sigma(x))\bigg|_{\sigma=0} = \frac{d}{d\sigma}(q(\pi(x_\sigma)))\bigg|_{\sigma=0} = \|X\|^{(d-1)/d} \mathrm{d}q_x \circ \mathrm{d}\pi_x(X_x) \ . \tag{34}$$

Now consider the flow line γ of $\mathrm{d}\pi(X)$ on N starting at x. The path $\gamma(t) \in N$ therefore satisfies the initial value problem

$$\gamma(0) = x \quad \text{and} \quad \dot{\gamma}(t) = \mathrm{d}\pi(X_{\gamma(t)}) \ . \tag{35}$$

Then

$$\begin{aligned} \frac{d}{d\sigma} q(T_\sigma(x))\bigg|_{\sigma=0} &= \|X_x\|^{(d-1)/d} \mathrm{d}q(\dot{\gamma}(0)) \\ &= \|X_x\|^{(d-1)/d} \frac{d}{dt}(q \circ \gamma(t))|_{t=0} \ . \end{aligned} \tag{36}$$

Fig. 3 shows the flow line γ, the linear path in the direction of the vector field $\Psi(X)$ and its projection to the sphere N via π. Since by assumption we have $\mathrm{d}\pi(X_x) = c_x \nabla_\pi (q|_N)_x$, we get

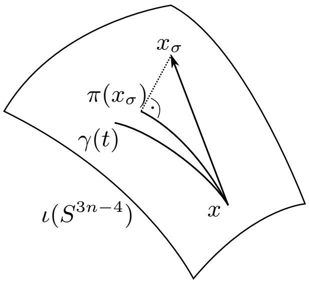


Fig. 3 The different players in the proof

$$\begin{aligned} \|X_x\|^{(1-d)/d} \cdot \frac{d}{d\sigma} q(T_\sigma(x))\bigg|_{\sigma=0} &= \frac{d}{dt}(q \circ \gamma(t))\bigg|_{t=0} = \langle \nabla_\pi (q|_N)_x, \dot{\gamma}(0)\rangle_\iota \\ &= \langle c_x^{-1} \mathrm{d}\pi(X_{\gamma(0)}), \dot{\gamma}(0)\rangle_\iota = c_x^{-1} \|\dot{\gamma}(0)\|_\iota^2 > 0 \ . \end{aligned} \tag{37}$$

Given $x \in N$, we therefore have that

$$q(T_\sigma(x)) > q(x) \quad \text{for } \sigma > 0 \text{ sufficiently small.}$$ □

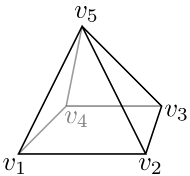
5 Generalizations

As we have discussed, the GETMe algorithm for volume meshes was originally conceived for tetrahedral meshes [34] and later generalized to mixed volume meshes [31, 32] via dual polyhedra. Furthermore, it is standard in applications to modify smoothing algorithms, so that they preserve the shape or features like corners and edges of a model. Let us outline, how such generalizations fit into our mathematical framework. Lastly, we have focused only on volume meshes, but with the same approach we can generalize the smoothing methods to arbitrary dimensions, in particular surface meshes. This allows us to construct a GETMe algorithm optimizing the isoperimetric quotient as a quality metric.

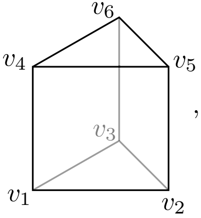
5.1 Mixed Volume Meshes

An important feature of our systematic approach is the natural compatibility of the GETMe algorithm for different volume elements by way of the mean volume. Let us give an explicit description of a method globally optimizing this mixed mesh quality. As we have already seen in (20), the vector X_e for tetrahedra e has an elegant description in terms of face normals. Not surprisingly, the same is true for hexahedra, prisms and pyramids. We compute the following vectors

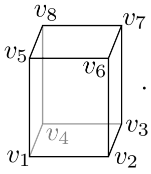
$$X_e = \frac{1}{2}\begin{bmatrix} \nu(5,4,2) + \nu(5,4,3,2) \\ \nu(5,1,3) + \nu(5,1,4,3) \\ \nu(5,2,4) + \nu(5,2,1,4) \\ \nu(5,3,1) + \nu(5,3,2,1) \\ 2 \cdot \nu(1,2,3,4) \end{bmatrix} \quad \text{for } e = \qquad , \tag{38}$$



$$X_e = \frac{1}{2}\begin{bmatrix} \nu(3,2,4) + \nu(2,5,4,6,3) \\ \nu(1,3,5) + \nu(3,6,5,4,1) \\ \nu(2,1,6) + \nu(1,4,6,5,2) \\ \nu(5,6,1) + \nu(6,3,1,2,5) \\ \nu(6,4,2) + \nu(4,1,2,3,6) \\ \nu(4,5,3) + \nu(5,2,3,1,4) \end{bmatrix} \quad \text{for } e = \qquad , \tag{39}$$



$$X_e = \frac{1}{2}\begin{bmatrix} \nu(2,5,4) + \nu(6,5,8,4,3,2) \\ \nu(3,6,1) + \nu(7,6,5,1,4,3) \\ \nu(4,7,2) + \nu(8,7,6,2,1,4) \\ \nu(1,8,3) + \nu(5,8,7,3,2,1) \\ \nu(1,6,8) + \nu(6,7,8,4,1,2) \\ \nu(2,7,5) + \nu(7,8,5,1,2,3) \\ \nu(3,8,6) + \nu(8,5,6,2,3,4) \\ \nu(4,5,7) + \nu(5,6,7,3,4,1) \end{bmatrix} \quad \text{for } e = \qquad . \tag{40}$$



For details we refer to [29, Sects. 5 and 6].

5.2 Isoperimetric Quotient

If we want to construct a quality measure, which measures how round a given polyhedron is, we can try the isoperimetric quotient given by 36π times the volume squared divided by the surface area of the boundary cubed, where the constant normalizes the measure to be 1 for the unit sphere. Steiner [26] conjectured that platonic solids maximize the isoperimetric quotient, which has been confirmed for all polyhedra but the icosahedron [27, 9, 2]. Instead of the isoperimetric quotient we will consider its square root, which allows us to naturally introduce a sign using the signed volume (2). This generalizes to a function on the space of mixed volume meshes

$$\mathrm{iq}(x_e) = 6\sqrt{\pi}\frac{\mathrm{vol}(x_e)}{\mathrm{area}(x_e)^{3/2}}, \tag{41}$$

where area is the mean surface area of the boundary of the polyhedron mesh p defined analogously to the mean volume vol. The function iq extends to all polyhedral meshes by summing or multiplying the element qualities. A straight-forward computation of the gradient of iq gives us yet another GETMe global optimization-based smoothing and untangling algorithm. We expect this algorithm to be powerful, and by the very definition of the quality measure, it yields a scaling-invariant transformation.

As a side note, all tested meshes with an icosahedron structure converge to the regular icosahedron. A proof of this heuristic observation will not be easy and would imply Steiner's conjecture. Note that the isoperimetric volume element quality measure can be constructed using the mean volume function restricted to all polyhedra with a fixed boundary surface area, which exemplifies that different embeddings of the sphere of polyhedra induce different quality measures from the same volume function.

5.3 Arbitrary Dimensions

The gradient of the volume has a particular nice and computationally efficient expression. Nevertheless, we get similar transformations in arbitrary dimensions. In particular, the area enclosed by a polygon is the two-dimensional analogue of the volume for polyhedra, and we also have an isoperimetric quotient for polygons. Therefore we can construct a GETMe transformation for triangulations of the plane and even of surface meshes. These will yield global optimization based smoothing methods for surface meshes. We have performed preliminary tests for planar triangle meshes (see Fig. 4).

5.4 Shape Preservation

In order to preserve shape or features of a model, there exist standard projection techniques. If we consider the flow of X rather than the discrete transformation method, we can constrain feature vertices to the submanifolds of N given by the boundary surface, edges or corners. These submanifolds might not be smooth. Nevertheless, this approach fits nicely into the mathematical framework by considering the restriction of the volume function to these submanifolds of N.

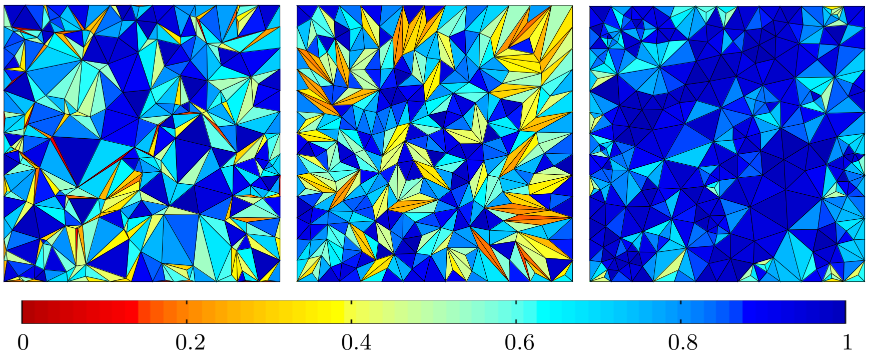


Fig. 4 From left to right: The initial mesh and results from optimizing $\prod \mathrm{area}(x_e)$ and $\prod \mathrm{iq}(x_e)$ with each element colored according to its mean ratio quality number

6 Conclusions

We have described the mathematical framework and a systematic approach to global optimization-based GETMe smoothing methods, which enables us to systematically prove, generalize and analyze properties of GETMe. These methods have the potential of being both runtime efficient and provably quality effective. Together with the local optimization-based GETMe smoothing and untangling method discussed in [29] as well as standard topology modification techniques, this provides a powerful framework for preparing meshes in the context of the finite element method. We have described some algorithms optimizing global objective functions, which will be analyzed further in future publications. See Fig. 4 for results from GETMe methods optimizing the product of the triangle areas and the product of the isoperimetric quotients, which show that the product of the areas does not yield a good global measure for the finite element method, while the isoperimetric quotient does.

References

1. Amenta, N., Bern, M., Eppstein, D.: Optimal Point Placement for Mesh Smoothing. Journal of Algorithms 30, 302–322 (1999)
2. Bezdek, D.: A proof of an extension of the icosahedral conjecture of Steiner for generalized deltahedra. Contrib. Discrete Math. 2(1), 86–92 (2007)

3. Bossen, F.J., Heckbert, P.S.: A Pliant Method for Anisotropic Mesh Generation. In: Proceedings of the 5th International Meshing Roundtable (1996)
4. Branets, L.V.: A variational grid optimization method based on a local cell quality metric. Ph.D. thesis, Austin, TX, USA, AAI3187661 (2005)
5. Brewer, M., Diachin, L.A.F., Knupp, P.M., Leurent, T., Melander, D.: The Mesquite Mesh Quality Improvement Toolkit. In: Proceedings of the 12th International Meshing Roundtable, pp. 239–250 (2003)
6. Canann, S.A., Tristano, J.R., Staten, M.L.: An Approach to Combined Laplacian and Optimization-Based Smoothing for Triangular, Quadrilateral, and Quad-Dominant Meshes. In: Proceedings of the 7th International Meshing Roundtable, pp. 479–494 (1998), `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.34.7221`
7. Chen, Z., Tristano, J.R., Kwok, W.: Combined Laplacian and Optimization-based Smoothing for Quadratic Mixed Surface Meshes. In: Proceedings of the 12th International Meshing Roundtable (2003), `http://www.andrew.cmu.edu/user/sowen/imr12.html`
8. Diachin, L.A.F., Knupp, P.M., Munson, T., Shontz, S.M.: A comparison of two optimization methods for mesh quality improvement. Engineering with Computers 22(2), 61–74 (2006), `http://dx.doi.org/10.1007/s00366-006-0015-0`, doi:10.1007/s00366-006-0015-0
9. Fejes Tóth, L.: Ein Beweisansatz für die isoperimetrische Eigenschaft des Ikosaeders. Acta Math. Acad. Sci. Hungar. 3, 155–163 (1952)
10. Field, D.A.: Laplacian smoothing and Delaunay triangulations. Communications in Applied Numerical Methods 4(6), 709–712 (1988)
11. Freitag, L.A.: On combining Laplacian and optimization-based mesh smoothing techniques. In: Trends in Unstructured Mesh Generation, pp. 37–43 (1997)
12. Freitag, L.A., Jones, M., Plassmann, P.: An Efficient Parallel Algorithm for Mesh Smoothing. In: Proceedings of the 4th International Meshing Roundtable, pp. 47–58 (1995), `http://www.andrew.cmu.edu/user/sowen/Roundtable.agenda.html`
13. Freitag, L.A., Ollivier-Gooch, C.: Tetrahedral Mesh Improvement Using Swapping and Smoothing. International Journal for Numerical Methods in Engineering 40(21), 3979–4002 (1997), doi:10.1002/(SICI)1097-0207(19971115)40:21(3979::AID-NME251)3.0.CO;2-9
14. Freitag, L.A., Plassmann, P.: Local optimization-based simplicial mesh untangling and improvement. International Journal of Numerical Methods in Engineering 49(1-2), 109–125 (2000), doi:10.1002/1097-0207(20000910/20)49:1/2(109:AID-NME925)3.0.CO;2-U
15. Klingner, B.M., Shewchuk, J.R.: Aggressive Tetrahedral Mesh Improvement. In: Proceedings of the 16th International Meshing Roundtable, pp. 3–23 (2007)
16. Knupp, P.M.: Algebraic mesh quality metrics. SIAM Journal on Scientific Computing 23(1), 193–218 (2001)
17. Knupp, P.M.: Hexahedral and Tetrahedral Mesh Untangling. Engineering with Computers 17(3), 261–268 (2001), doi:10.1007/s003660170006
18. Leng, J., Xu, G., Zhang, Y., Qian, J.: A Novel Geometric Flow-Driven Approach for Quality Improvement of Segmented Tetrahedral Meshes. In: Quadros, W.R. (ed.) Proceedings of the 20th International Meshing Roundtable, pp. 347–364. Springer Publishing Company, Incorporated (2012)

19. Li, T., Wong, S., Hon, Y., Armstrong, C., McKeag, R.: Smoothing by optimisation for a quadrilateral mesh with invalid elements. Finite Elements in Analysis and Design 34(1), 37–60 (2000), http://www.sciencedirect.com/science/article/pii/S0168874X99000268, doi: http://dx.doi.org/10.1016/S0168-874X(99)00026-8
20. Mei, G., Tipper, J.C., Xu, N.: The Modified Direct Method: An Iterative Approach for Smoothing Planar Meshes. In: ICCS, pp. 2436–2439 (2013)
21. Owen, S.J.: A Survey of Unstructured Mesh Generation Technology. In: Proceedings of the 7th International Meshing Roundtable, pp. 239–267 (1998)
22. Parthasarathy, V., Kodiyalam, S.: A constrained optimization approach to finite element mesh smoothing. Finite Elements in Analysis and Design **9**(4), 309 – 320 (1991), http://www.sciencedirect.com/science/article/pii/0168874X(91)90004I, doi: http://dx.doi.org/10.1016/0168-874X9190004-I
23. Sastry, S.P., Shontz, S.M.: Performance characterization of nonlinear optimization methods for mesh quality improvement. Eng. with Comput. 28(3), 269–286 (2012), http://dx.doi.org/10.1007/s00366-011-0227-9, doi:10.1007/s00366-011-0227-9
24. Shimada, K., Yamada, A., Itoh, T.: Anisotropic Triangulation of Parametric Surfaces via Close Packing of Ellipsoids. Internat. J. Comput. Geom. Appl. 10(4), 417–440 (2000), http://www.worldscientific.com/doi/abs/10.1142/S0218195900000243, doi:10.1142/S0218195900000243; Selected papers from the Sixth International Meshing Roundtable, Part II (Park City, UT) (1997)
25. Shivanna, K., Grosland, N., Magnotta, V.: An Analytical Framework for Quadrilateral Surface Mesh Improvement with an Underlying Triangulated Surface Definition. In: Shontz, S. (ed.) Proceedings of the 19th International Meshing Roundtable, pp. 85–102. Springer, Heidelberg (2010), http://dx.doi.org/10.1007/978-3-642-15414-0_6, doi:10.1007/978-3-642-15414-0_6
26. Steiner, J.: Über Maximum und Minimum bei den Figuren in der Ebene, auf der Kugelfläche und im Raume überhaupt. C. R. Acad. Sci. Paris 12, 177–308 (1841)
27. Steinitz, E.: Über isoperimetrische Probleme bei konvexen Polyedern. J. Reine Angew. Math. 158, 129–153 (1927)
28. Strang, G., Fix, G.: An analysis of the finite element method, 2nd edn. Wellesley-Cambridge Press, Wellesley (2008)
29. Vartziotis, D., Himpel, B.: The mean volume as a quality measure for polyhedra and meshes, arXiv:1302.6066 [math.GT] (2013), http://arxiv.org/abs/1302.6066
30. Vartziotis, D., Papadrakakis, M.: Improved GETMe by adaptive mesh smoothing. Computer Assisted Methods in Engineering and Science 20, 55–71 (2013)
31. Vartziotis, D., Wipper, J.: A dual element based geometric element transformation method for all-hexahedral mesh smoothing. Comput. Methods Appl. Mech. Engrg. 200(9-12), 1186–1203 (2011), http://dx.doi.org/10.1016/j.cma.2010.09.012, doi:10.1016/j.cma.2010.09.012

32. Vartziotis, D., Wipper, J.: Fast smoothing of mixed volume meshes based on the effective geometric element transformation method. Comput. Methods Appl. Mech. Engrg. 201/204, 65–81 (2012), `http://dx.doi.org/10.1016/j.cma.2011.09.008`, doi:10.1016/j.cma.2011.09.008
33. Vartziotis, D., Wipper, J., Papadrakakis, M.: Improving mesh quality and finite element solution accuracy by GETMe smoothing in solving the poisson equation. Finite Elem. Anal. Des. 66, 36–52 (2013)
34. Vartziotis, D., Wipper, J., Schwald, B.: The geometric element transformation method for tetrahedral mesh smoothing. Comput. Methods Appl. Mech. Engrg. 199(1-4), 169–182 (2009), `http://dx.doi.org/10.1016/j.cma.2009.09.027`, doi:10.1016/j.cma.2009.09.027
35. Wilson, T.J.: Simultaneous Untangling and Smoothing of Hexahedral Meshes. Master's thesis, Universitat Politècnica de Catalunya, Spain (2011)