Appl. Math. J. Chinese Univ.
2022, 37(3): 453-469

Enhanced optimal delaunay triangulation methods with connectivity regularization

HAI Yong-qing[2,†] GUO Yu-fei[1,†,*] DONG Mo[2]
ZHAO Rong-li[1] SUN Ke-wu[1] SHANG Fei-fei[3]

Abstract. In this paper, we study the underlying properties of optimal Delaunay triangulations (ODT) and propose enhanced ODT methods combined with connectivity regularization. Based on optimizing node positions and Delaunay triangulation iteratively, ODT methods are very effective in mesh improvement. This paper demonstrates that the energy function minimized by ODT is nonconvex and unsmooth, thus, ODT methods suffer the problem of falling into a local minimum inevitably. Unlike general ways that minimize the ODT energy function in terms of mathematics directly, we take an outflanking strategy combining ODT methods with connectivity regularization for this issue. Connectivity regularization reduces the number of irregular nodes by basic topological operations, which can be regarded as a perturbation to help ODT methods jump out of a poor local minimum. Although the enhanced ODT methods cannot guarantee to obtain a global minimum, it starts a new viewpoint of minimizing ODT energy which uses topological operations but mathematical methods. And in terms of practical effect, several experimental results illustrate the enhanced ODT methods are capable of improving the mesh furtherly compared to general ODT methods.

Nowadays, with the fast development of information technology, computer simulation casts an increasingly important role in many fields, such as finite element analysis, game physics engine, geographic information system, and so on. And mesh is the foundation for most of the computing tasks. There are many works for generating meshes and optimizing meshes [1], [2], [3], [4], [5], [6], [7], [8], [9]. Among these, the variational meshing methods based on energy function minimization have attracted more and more attention recently [10], [11], [12], [13], [14], due to the excellent performances in both mesh generation and mesh improvement. The concept of optimal Delaunay triangulation (ODT) is introduced by Jin and Chen [13] first and they prove that ODT methods obtain an optimal mesh by minimizing the linear interpolation error [14]which is called the ODT energy. Though ODT methods are very effective in improving the

Received: 2021-10-15. Revised: 2021-12-13.
MR Subject Classification: 62F10, 62H12.
Keywords: mesh optimization, connectivity regularization, ODT methods, triangular mesh.
Digital Object Identifier(DOI): https://doi.org/10.1007/s11766-022-4588-1.
Supported by the National Natural Science Foundation of China(11802064).
†These authors contributed equally to this work.
*Corresponding author.

quality of bad elements [15], it is not easy to obtain its optimal solution of ODT energy function. Driven by this problem, we propose an outflanking strategy, which combines ODT methods with connectivity regularization and makes connectivity regularization serve as a perturbation to help ODT methods jump out of a poor local minimum.

The ODT energy function is usually a very complicated form and related to the node position and mesh topology. There are no ODT methods that can optimize the node position and mesh connectivity at the same time. Jin and Chen [13] prove that the ODT energy function attains its minimum when the mesh is the Delaunay triangulation for a fixed number of nodes. Hence, they present a typical solution to minimize the ODT energy function by optimizing the mesh topology and node positions iteratively. Since there are many algorithms and open sources to obtain a Delaunay triangulation [7], [16], it is easy to optimize the mesh topology. However, minimizing the ODT energy is difficult. Chen [14] present some local relaxation methods that move only one node at a time. These methods are easy to fall into a local optimum. Alliez *et al*. [15] move all the nodes simultaneously, but this method cannot guarantee to decrease the energy. To jump out of local optima, Chen *et al*. [17] combine edge flip with ODT. Chen and Holst [11] present a global method using Newton's method. They prove that the method converges faster and can obtain a better mesh, but this method is not suitable for largescale problems and still suffers the problem of falling into poor local optima. Chen *et al*. [18] apply a quasi-Newton method. Nevertheless, since the ODT energy function is nonconvex, any local search method inevitably falls into a local optimum. There are some works [18], [19] combining a perturbation scheme to avoid this problem. But if the magnitude of random perturbation is too small, the method will roll back to the same local minimum, if the magnitude of random perturbation is too large, the method would amount to restarting optimization with a random initialization.

Our work demonstrates that even if the optimal node positions can be obtained under any fixed mesh topological connection, we cannot know whether the node positions correspond to the global minimum. Consequently, ODT methods still suffer the problem of falling into a local minimum by optimizing the mesh topology and node positions iteratively. ODT methods tend to equally distribute the edge lengths, thereby, generate regular meshes [20], [21]. Obviously, a triangular mesh where all elements are regular triangles and valences of all nodes within boundaries are 6 is the typical high-quality mesh. Based on this observation, we start another viewpoint that focuses on the combination of connectivity regularization with ODT rather than the iterative scheme to minimize the ODT energy function itself through mathematical methods. As a further development of ODT methods, the proposed methods edit the node connectivity directly, which is very different from the previous works.

The rest of the paper is organized as follows. In Section 2, we briefly review the optimal Delaunay triangulations and the class of ODT methods. And then we demonstrate that the ODT energy function is nonconvex and unsmooth with a simple case, and even if the optimal node position can be obtained under any fixed mesh topological connection, it still suffers the problem of falling into a local minimum by optimizing the mesh topology and node positions iteratively. In Section 3, we specify the connectivity regularization to reduce the irregular nodes by basic topological operations. And under an outflanking strategy, the enhanced ODT methods are proposed by introducing the connectivity regularization into ODT methods. In

Section 4, some examples including uniform meshes and graded meshes are tested with the proposed enhanced ODT methods to show its effectiveness. Section 5 is the conclusive part of this paper.

§1 Revisiting Optimal Delaunay Triangulations

ODT methods are very effective in mesh generation and mesh improvement, but they may suffer the problem of getting stuck in the local optimal solution.

1.1 The concept of ODT

ODT is an optimization-based approach in terms of linear interpolation error for a given function $f(x)=\|x\|^2$. Let X be a set of nodes in a convex domain Ω and T be a triangulation of X.

Then we can obtain another set of nodes $X_1(x, \|x\|^2)$ by lifting up the X to the function $f(x)$. Let $f_1(x)$ be the piecewise linear nodal interpolant of X_1 based on T, as shown in Figure 1. And the ODT energy function is defined as the L^1 norm of the interpolation error of $f(x)$ by $f_1(x)$, as the one bellow:

$$E_{ODT}(X,T) = \int_{\Omega} \|f_1(x) - f(x)\| dx. \tag{1.1}$$

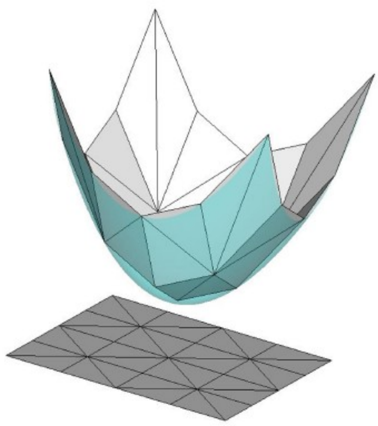

Figure 1. Illustration of the ODT energy.

The ODT energy function is related to the node position and mesh topological connection, and it is very difficult to optimize the node position and mesh topology connection at the same time. Chen [13] prove that the ODT energy function attains its minimum when the mesh is the Delaunay triangulation for a given number of fixed nodes. Hence, he decomposes it into two sub-problems: solving $E_{ODT}(X,T)$ with the location of nodes fixed, and solving $E_{ODT}(X,T)$ with the connectivity of vertices fixed. Constructing a Delaunay triangulation for a given set of nodes have been well studied [6], [7], [8] hence the focus of most ODT methods is on optimizing the node positions.

More general, as for graded meshes, the function of interpolation error can be defined with incorporating a density function $\rho(x)$,

$$E_{ODT}(X,T) = \int_{\Omega} \|f_1(x) - f(x)\|\rho(x)dx. \tag{1.2}$$

1.2 Methodology of ODT

For the application to mesh improvement, ODT methods can be classified into local and global classes. The local methods are easy to implement. The global methods are more effective but have heavier time cost. The differences among ODT methods are the way to solve $E_{ODT}(X,T)$.

Local methods. Considering moving only one node x_i at one time, we calculate the derivative of $E_{ODT}(X,T)$ and make it equal to 0. Chen and Holst [20] give the following formula to update the node position,

$$x_i^* = \frac{1}{2\|\Omega_i\|} \sum_{T_j \in \Omega_i} (\sum_{x_k \in T_i, x_k \neq x_i} \|x_k\|^2 \nabla_{x_i} \|T_j(x)\|), \tag{1.3}$$

where T_j represents a triangle and $\| * \|$ represents the area operation. And Ω_i is the first-ring neighbours of the vertex x_i.Alliez *et al.*[15] simplify the Eq. 3 and presented a more commonly used equation, as bellowing:

$$x_i^* = \frac{1}{\|\Omega_i\|} \sum_{T_j \in \Omega_i} (c_j \|T_j(x)\|), \tag{1.4}$$

where c_j is the circumcentre of T_j. Namely for an ODT, each interior vertex x_i is a weighted centroid of circumcentres of simplices in the star shape of x_i. There are also some other modifications [21], nevertheless, they all move a node by considering the information of its first-ring neighbours.

Global methods. Global methods move all nodes simultaneously to minimize the $E_{ODT}(X,T)$. As an optimization problem, there are many practical ways to solve it [22]. Chen and Holst [11] adopt Newton's method, and showe that global methods can obtain a better mesh than local methods. Chen *et al.* [18] propose to apply a quasi-Newton method. However, since $E_{ODT}(X,T)$ is nonconvex, any local search method inevitably falls into a local optimum. We will show the nonconvex of $E_{ODT}(X,T)$ in the next section and demonstrate that even if the optimal node position can be obtained under any fixed mesh topological connection, it still suffers the problem of falling into a local minimum by optimizing the mesh topology and vertex positions iteratively.

As for surface meshes, ODT methods are implemented through reprojecting nodes [15] or suboptimal methods [23] , and Feng *et al.* [24] make ODT methods appropriate for surface curved triangulations which are different from the linear discretization we talk about. For 3D cases, Gao *et al.* extend ODT to tetrahedral meshes [25] .

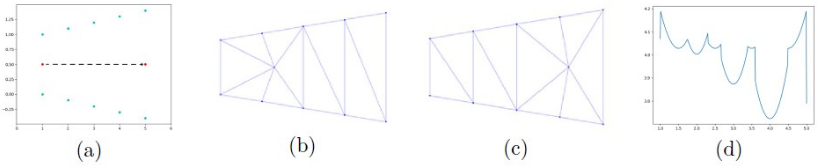

Figure 2. The example of 11 nodes and its corresponding ODT energy.(a) The example of 11 nodes; (b) and (c) Delaunay triangulations of the 11 nodes; (d) The corresponding energy.

1.3 The property of the ODT energy and limitation of the ODT methods

E_{ODT} is related to the mesh topology and node positions. Although it is difficult to fully study the underlying mathematical property of this function rigorously, we compute the E_{ODT} functions of plenty of triangulations to seek the rules. Herein, we use one example partly with a simple 2D triangulation of eleven nodes for demonstration. In this example, we move a node along a straight line parameterized by t as shown in Figure 2a. In Figure 2a, the red node is inserted within the envelope of the ten blue nodes. A Delaunay triangulation is constructed using the ten fixed nodes and the one moving node, and Figures 2b and 2c show two middle times of the Delaunay mesh during the movement. We plot the ODT energy function of the mesh with respect to t in Figure 2d. It can be seen that E_{ODT} is nonconvex and unsmooth. Furthermore, changing the connectivity of a mesh will cause abrupt changes in E_{ODT}. Hence, even if the optimal node positions can be obtained under any fixed mesh topological connection at some point, we cannot know whether the E_ODT corresponds to the global minimum and ODT methods still suffer the problem of falling into a local minimum by optimizing the mesh topology and vertex positions iteratively.

Clearly, any large-scale mesh can be seen as a series of connected simple triangulations, and the connected simple triangulations are mutually independent during ODT optimization. Hence, the total E_{ODT} of a large-scale mesh can be seen as a sum of E_{ODT} of its piecewise parts. The simple triangulation depicted in Figure 2 only has one parameter t, the x coordinate of the red nodes, which is why we are capable of plotting the curve of E_{ODT} in detail. Since the E_{ODT} of such a simple triangulation is nonconvex and unsmooth, one can imagine the complexity of the E_{ODT} of general cases. In fact, the E_{ODT} of large-scale meshes is too complex to grasp, and the global minimum is no doubt hard to obtain.

As mentioned above, Chen *et al.* [18] apply random node perturbation to help these methods jump out from the ditch. However, an appropriate magnitude of random perturbation is not easy to be obtained. And if it is too small, the methods will roll back to the same local minimum; if it is too large, the methods will restart the optimization with a random initialization.

§2 Enhanced ODT methods

In this section, we introduce an outflanking strategy based on the connectivity regularization. Although connectivity regularization serves as a perturbation, it works in most cases and casts off the problem of perturbation amplitude, which is more qualified to handle the inherent problems of ODT methods. It is worth mentioning that Chen *et al*. [17] also think the topological operation can be regarded as a perturbation to help ODT jump out of poor local optima. But our method has a more organized strategy of topological operations instead of single edge swap. ODT methods tend to generate regular meshes [20], [21] and the regularity of mesh is influenced by the connectivity also known as topology to a great extent. Therefore, it is very meaningful to study how to optimize the connectivity of the mesh by edge flip, edge split and edge collapse while keeping the number of nodes in the mesh unchanged. The introduction of connectivity regularization is not based on positive-going mathematical formula derivation but experience and observation. The experience tells us that although exploiting mathematical methods to minimize ODT energy directly is most people's choice and well developed, it hits a bottleneck to some extent and its biggest variable is the selection of optimization algorithm. The observation shows us that each connectivity change is likely to cause a sudden change in ODT energy, as depicted in Figure 2. Hence, we are inpired to combine topological optimization with ODT methods. In spite of the lack of rigorous mathematical proof, this can lead to further improvements for ODT methods instead of pure mathematical methods. And testing result prove our idea, which will discussed in the next section.

2.1 Connectivity regularization

Generally, uniform ODT methods tend to equally distribute the edge lengths, thereby generates regular meshes [20], [21]. As for mesh connectivity, a regular mesh is a mesh where all node valances are 6 for planar triangular meshes. As the differences between node valences and 6 are smaller, the level of regularity and the quality of the mesh are higher.

Alliez *et al*. [26] propose a method by randomly picking an edge and performing an edge flipping only if it favors valence 6 for interior nodes, and valence 4 on boundary nodes. Surazhsky and Gotsman [27] propose a method that can move the irregular node pairs to further improve meshes. Li *et al*. [28] propose a theoretical analysis and an interactive editing framework combining edge flip, edge split, and edge collapse to reduce the irregular nodes. Aghdaii *et al*. [29] and Vidal *et al*. [30] propose the concept of 5-6-7 mesh, referring to a closed surface triangle mesh where each vertex has valence 5, 6, or 7.

For the mesh with a certain number of nodes, the connectivity can be arbitrarily modified by the simple edge flip [26]. The edge split and edge collapse can also improve the mesh connectivity by adding a node and deleting a node. The number of nodes should not be changed in ODT methods usually. Clearly, performing the same number of edge splitting and edge collapsing will not change the node number. Basic operations including edge flip, edge collapse, and edge split are illustrated in Figure 3.

Essentially, the optimal valence of a node is determined by its local Gaussian curvature [31]. If the Gaussian curvature is greater than 0, the local shape is an elliptical surface and the optimal valence can be less than 6. And if the Gaussian curvature is less than 0, the local shape is hyperbolical surface and the optimal valence can be greater than 6. As for plane and paraboloid, the Gaussian curvature is always 0. Inspired by the formulas in [32], the optimal valence of a node v can be defined as,

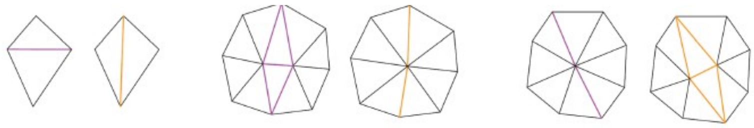

Figure 3. Basic topological operations. Edge flip (right), edge collapse (middile) and edge split (right).

$$val_{opt}(v) = round(\sum_{j\in N}\theta_j/60), \tag{2.5}$$

where N is the number of neighboring elements, θ_j is the angle of its neighboring element at the angular vertex v . The $val_{opt(v)}$ of an interior node on a plane is 6, and a boundary node is related to its boundary angle. We denote a node with the valence of n as vn, e.g. for a regular interior node on a plane, it is $v6$.

Practically, as for a node valence, what we care is not the absolute valence but the relative difference to the optimal valence. We take a residual form to measure the level of connectivity regularity as the following function,

$$res(v) = \sum_{v\in M}(val(v) - val_{opt}(v))^2, \tag{2.6}$$

where $val(v)$ is valence of node v, and $val_{opt}(v)$ is its optimal valence. Connectivity regularization is to minimize the value of Equation.6. The viewpoint of the residual form enables us to extent connectivity regularization to surface meshes. Herein, we take the case of planar meshes as an example for illustration, and the case for surface meshes can be derived naturally. Usually, most optimal node valences of a surface mesh are still equal to 6.

We conduct connectivity regularization using valence editing, which is composed of a series of basic operations. Furthermore, improvable configurations are the direct subjects we conduct connectivity regularization on. Figure 4 illustrates the simple improvable configurations formed by $v3, v4, v8, v9, v10$, and they can be improved with basic operations. Through iterative operations, all of the nodes can be reduced to $v5, v6, v7$, which have a residual value no more than 1. Some connected $v5, v7$ can form complex improvable configurations and they can be improved through basic operations, as demonstrated in Figure 5. The time cost of diminishing each improvable configuration is $O(1)$. And similarly, connectivity regularization can be done for surface meshes on the condition that each node gets its own optimal valence and the difference to its optimal valence.

Although there are still existing potential improvable cases besides the involving 13 configurations [33], numerical experiments show that chasing the perfection of mesh connectivity blindly will reduce the effectiveness rate. Further valence improvements can lead to more time cost and large-scale disturbance for meshes. On the other hand, the purpose of connectivity regularization is to help the ODT methods jump out of the local minimum iteratively but not to obtain the optimal mesh valence at once. Some extremely complex improvable cases in the potential can naturally convert into easy ones or even do not exist anymore after times of iterations. Hence, in view of a trade-off between effect and complexity, these mentioned operations

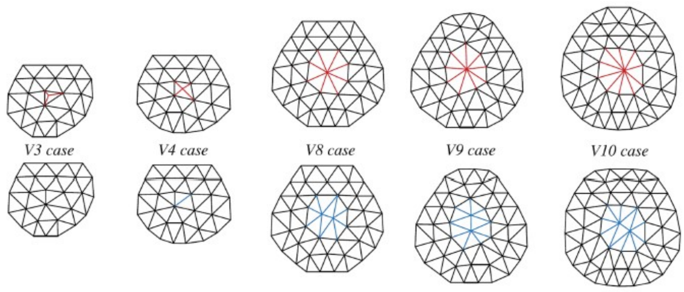

Figure 4. Simple improvable configurations for connectivity regularization.

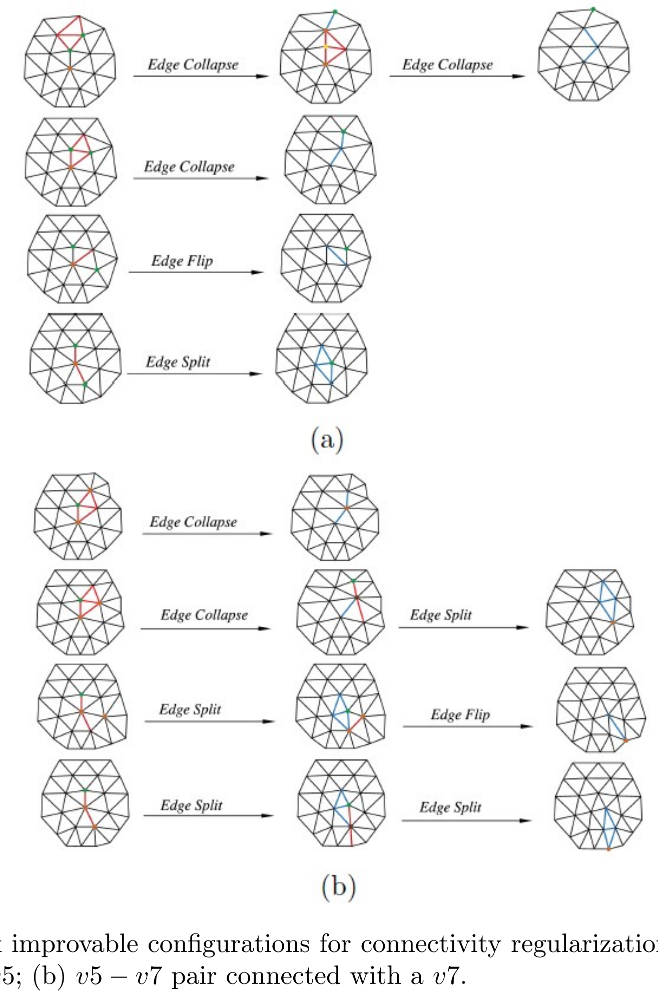

Figure 5. Complex improvable configurations for connectivity regularization. (a) $v5 - v7$ pair connected with a $v5$; (b) $v5 - v7$ pair connected with a $v7$.

have already pushed ODT methods to satisfactory results.

2.2 Connectivity regularization with the number of nodes unchanged

The concept of ODT is usually not allowed to change the number of nodes. Although connectivity regularization without limitations will boost the ODT methods greatly, there is still a necessity to develop the enhanced ODT methods with the number of nodes unchanged in some cases.

Obviously, any simple improvable configuration or complex improvable configuration has a certain number of edge flips and edge splits or edge collapses, resulting in the certain number change of nodes. Once we recognize all of the improvable configurations, we want to diminish improvable configurations with the number of nodes unchanged as more as possible. This situation is analogous to the crammed knapsack problem with the capacity of the knapsack being 0. Supposing that there are N items and i is the item index, each improvable configuration can be seen as an item to fill in the knapsack with a capacity of c_i and a value of v_i. The capacity c_i is equal to the number change of nodes including -2, -1, 0, 1, 2, and the value v_i can be uniform or distinguished between simple configurations and complex configurations. Dynamic programming can maximize the value of the knapsack with the knapsack crammed. And the state transition formula is defined as,

$$dp(j) = max\{dp(j - c_i) + v_i, dp(j)\}, 1 <= i <= N, 0 <= j <= 4N, \tag{2.7}$$

where j is the current capacity of knapsack. Noticing there are negative capacities, we set a pivot equal to $2N$ to handle this problem, which is equivalent to translate the capacity range. The index j ranging from 0 to $4N$ stands for the actual capacity ranging from $-2N$ to $2N$. Naturally, $dp(2N)$, the pivot is initialized as 0 and other elements in dp are initialized as negative infinity. The index i loops from 1 to N as c_i is tried to push in the knapsack. If $c_i > 0$, we update $dp(j)$ with j ranging from 4N to c_i. If $c_i < 0$, we update $dp(j)$ with j ranging from 0 to $4N + c_i$.

Herein, we denote the connectivity regularization based on a dynamic program as $ValImp$. The time complexity of searching all the improvable configurations is linear, thus, the time complexity of algorithm $ValImp$ is $O(N^2)$, determined by the dynamic programming scheme. Note that pure combining connectivity regularization with ODT methods is also accepted if there is a relaxation for the number of nodes. The purpose of the algorithm is to help the ODT methods jump out from the local minimum not to obtain the optimal mesh valence, hence it does not need to handle all cases that can be improved at once, and we iterate the mesh only once for every stage in the method.

2.3 Extension to surface meshes

Enhanced-ODT methods mainly consist of three kinds of operations: Delaunay triangulation, optimizing node positions, and connectivity regularization. Realizing the enhanced-ODT methods for surface meshes only requests some modifications to the original version. Surface Delaunay triangulation is well studied in [34] [35] [36] [37] [38] and we adopt the metric proposed in [35] [36] to maintain the Delaunay properties. Optimizing node positions on surface meshes can be conducted through reprojection [39]. And connectivity regularization for surface meshes can be also naturally derived with considerations on feature curves on surface meshes.

2.4 Algorithms for enhanced ODT methods

The commonly-used ODT methods usually have two stages: first optimize node positions with local manners or global manners; then optimize mesh topology. Combined with connectivity regularization based on dynamic programming scheme, the enhanced ODT methods are as Algorithm 1.

Algorithm 1 Enhanced ODT methods

Input: Initial mesh T; Iteration times M and N.
Output: Final optimized mesh T.

```
Iteration variables m = 0, n = 0
while m < M do
   call the ValImp;
   while n < N/M do
      ODT optimization;
      n=n+1;
   end while
   m=m+1;
   n=0.
end while
```

Note that we use constant iterations to stop the optimization in this paper ($N = 10$ and $M = 2$), an alternative approach to stop the optimization is dependent on E_{ODT} change or E_{ODT} upper bound.

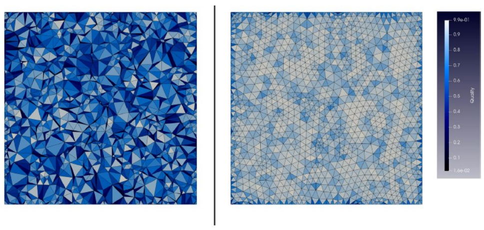

Figure 6. Mesh models of Example 1 before (left) and after (right) LMVI.

§3 Numerical experiments and analysis

The proposed algorithm is implemented aided by Cinolib [40]. Some of our testing examples including uniform meshes and graded meshes are chosen to illustrate the effectiveness of our proposed methods in this section. The qualities of the original meshes are at a low level, and their boundary nodes and feature nodes are fixed during the optimization process. We use element angles and the following quality metric to measure the meshes,

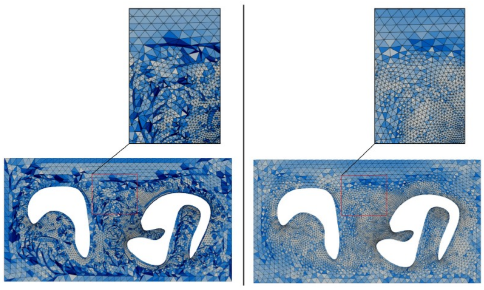

Figure 7. Mesh models of Example 2 before (left) and after (right) LMVI.

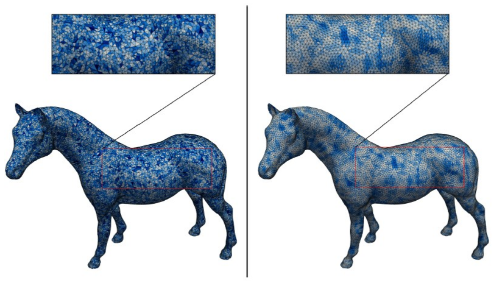

Figure 8. Mesh models of Example 3 before (left) and after (right) GMVI.

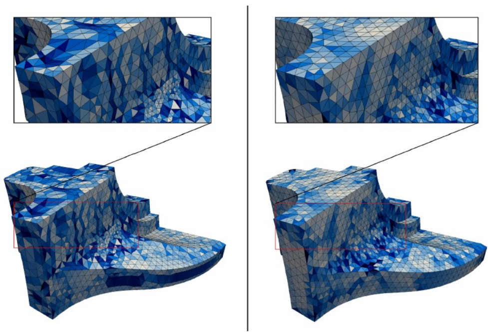

Figure 9. Mesh models of Example 4 before (left) and after (right) GMVI.

Figure 10. Mesh models of Example 5 before (left) and after (right) GMVI.

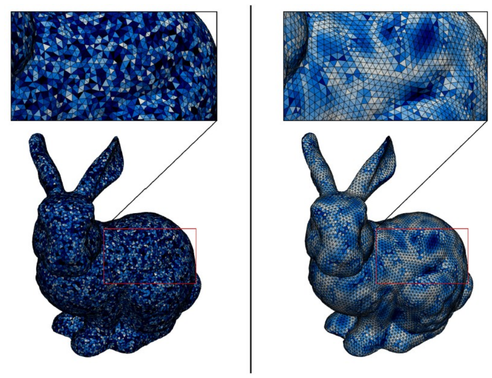

Figure 11. Mesh models of Example 6 before (left) and after (right) LMVI.

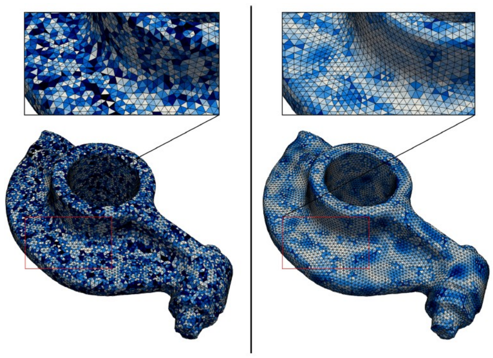

Figure 12. Mesh models of Example 7 before (left) and after (right) GMVI.

Table 1. Statistics of numerical experiments.

Model	Method*	Avg. q	Min. q	Min. $\theta/(^\circ)$	Avg. Min. $\theta/(^\circ)$	E_{ODT}	Times(s)
Example 1	Original	0.692	0.024	0.801	30.495	12.355	-
	Laplacian	0.955	0.298	15.984	48.375	4.076	-
	GETme	0.956	0.300	15.972	49.004	4.014	-
	LM	0.965	0.378	24.025	51.565	3.270	0.56
	LMVI	0.967	0.475	24.192	51.843	3.218	1.28
	GM	0.966	0.378	24.263	51.793	3.241	0.78
	GMVI	0.969	0.529	24.506	52.133	3.154	1.51
Example 2	Original	0.865	0.034	1.487	42.511	54.739	-
	Laplacian	0.941	0.301	14.347	48.685	25.483	-
	GETme	0.943	0.317	19.042	48.855	24.425	-
	LM	0.949	0.351	26.810	49.077	22.992	0.86
	LMVI	0.954	0.401	26.580	49.077	22.992	0.86
	GM	0.961	0.384	26.300	50.527	21.311	1.02
	GMVI	0.965	0.395	26.592	51.041	20.929	2.35
Example 3	Original	0.888	0.107	4.812	42.711	95.824	-
	Laplacian	0.945	0.172	5.889	48.435	44.365	-
	GETme	0.949	0.175	5.904	48.987	43.015	-
	LM	0.953	0.176	5.943	49.138	39.763	11.71
	LMVI	0.961	0.198	6.684	50.342	37.193	28.14
	GM	0.953	0.176	5.943	49.143	38.068	18.54
	GMVI	0.963	0.198	6.684	49.989	36.517	37.59
Example 4	Original	0.926	0.334	12.099	47.033	304.029	-
	Laplacian	0.950	0.494	20.028	50.002	168.236	-
	GETme	0.952	0.496	20.011	50.115	162.436	-
	LM	0.957	0.500	20.479	50.372	153.587	3.01
	LMVI	0.962	0.534	20.499	50.550	151.002	8.25
	GM	0.959	0.534	20.499	50.250	152.164	5.85
	GMVI	0.963	0.536	20.499	50.809	149.408	10.32
Example 5	Original	0.702	0.032	5.369	35.265	68.254	-
	Laplacian	0.852	0.044	6.021	46.325	27.635	-
	GETme	0.856	0.045	6.021	46.628	26.747	-
	LM	0.864	0.047	6.135	48.036	25.365	17.22
	LMVI	0.884	0.052	6.124	48.336	23.602	42.64
	GM	0.878	0.047	6.115	47.921	24.856	23.36
	GMVI	0.886	0.052	6.120	48.546	23.021	54.63
Example 6	Original	0.804	0.156	10.426	34.652	57.624	-
	Laplacian	0.945	0.213	12.414	48.464	25.460	-
	GETme	0.947	0.212	12.436	48.502	25.421	-
	LM	0.958	0.439	18.321	49.979	23.875	9.243
	LMVI	0.970	0.446	18.458	50.002	23.504	20.483
	GM	0.967	0.440	18.324	49.978	23.648	9.962
	GMVI	0.974	0.446	18.543	50.013	23.462	25.602
Example 7	Original	0.835	0.081	8.535	33.221	104.476	-
	Laplacian	0.951	0.140	9.215	47.346	49.242	-
	GETme	0.952	0.141	9.438	47.402	49.122	-
	LM	0.962	0.328	21.986	49.125	43.021	8.710
	LMVI	0.972	0.335	22.462	49.502	42.456	20.012
	GM	0.968	0.332	22.214	49.342	42.782	10.875
	GMVI	0.976	0.336	22.664	49.674	42.368	24.487

* *LM: local methods; LMVI: local methods with ValImp; GM: global methods: GMVI: global methods with ValImp.*

$$q = (4\sqrt{3} \times S)/(l_1^2 + l_2^2 + l_3^2), \tag{3.8}$$

where l_i represents edge lengths and S represents area of a triangular element. The value of this metric ranges from 0 to 1 and the optimal value of the metric is 1. Results of numerical experiments using general ODT methods and enhanced ODT methods are listed in Table 1. In table 1, global methods and local methods are implemented as Section 2.2 states. LM means local methods for ODT, GM means global methods for ODT, LMVI means local methods with valence improvement for ODT, GMVI means global methods with valence improvement for ODT. And LMVI and GMVI belong to the enhanced ODT methods. Besides, we also use the well-known Laplacian smoothing and the state-of-art GETme [41] for comparison. Example 1 and Example 2 are two planar triangular meshes, and the rest three examples are surface meshes. Example 1 is a uniform triangular mesh with 2298 elements, and Example 2 is a grading mesh with 10085 elements. The two original examples before and after LMVI method are shown in Figure 6 and Figure 7. Example 3 is a horse mesh model consisting of 39698 elements. Figure 8 is the comparison of the original mesh and optimized mesh by using GMVI. Example 4 is a fandisk mesh model consisting of 4874 elements. It is an industrial model with feature curves. We treat feature nodes fixed similarly to boundary nodes. Comparison between the original Example 4 and the example optimized by GMVI is shown in Figure 9. Example 5 is a surface mesh model of the human heart consisting of 70000 elements. Comparison between the original Example 4 and the example optimized by GMVI is shown in Figure 10. Example 6 is a bunny mesh model consisting of 23124 elements. Comparison between the original Example 6 and the example optimized by LMVI is shown in Figure 11. Example 7 is a rocker mesh model consisting 21700 elements, and comparison between the original Example 7 and the example optimized by GMVI is shown in Figure 12.

Through observing the table and the figures, it can be seen that ODT methods have advantages over pure geometric methods on the whole and the enhanced methods with ValImp have advantages over classical ODT methods due to connectivity regularization. Obviously, results of LMVI seems superior to results of LM and GMVI seems superior to results of GM. This indicates connectivity optimization can furtherly improve those meshes which cannot be improved by the conventional method any further, making the combination of connectivity optimization much meaningful. On the other hand, although some improvements are not significant, result differences between LM and LMVI turn to be larger than the differences between LM and GM, which implies the connectivity optimization does work and its impacts are even larger than the impacts on optimization algorithm.

For another, the precision of FEA is strongly related to the worst element. Geometric s-moothing does not change the connectivity and cannot eliminate nodes of extremely irregular valences like $v3, v10$, which tend to result in elements at poor quality. But ODT methods can get rid of them, which is why ODT methods have significant advantages on the metrics of $Min.q$, $Min.\theta$ and $Avg.Min.\theta$. And because of the combination with connectivity regularization, the enhanced ODT methods are endowed with the extra capablity of making "*connectivity*" distribute more evnenly, which have better performances on these metrics compared with general ODT methods. Meanwhile, in terms of the metric E_{ODT} standing for the ODT energy, the enhanced ODT methods have the lowest values. As stated above, although we cannot guarantee to obtain the global minimum, we do reduce E_{ODT} through the function $ValImp$. The results have verified the fact that connectivity regularization works well as a perturbation to help ODT methods jump out of a poor local minimum based on an outflanking strategy instead of pure mathematical methods.

§4 Conclusions

In this paper, we demonstrate that the ODT energy function is nonconvex and unsmooth, and propose enhanced ODT methods combined with connectivity regularization. For ODT methods even if the optimal node positions can be obtained under any fixed mesh topological connection, they still suffer the problem of falling into a local minimum by optimizing the mesh topology and node positions iteratively. We take an outflanking strategy combining ODT methods with connectivity regularization for this problem through reducing the irregular nodes by edge flip, edge split, and edge collapse while keeping the number of nodes in the mesh unchanged. Connectivity regularization can be regarded as a perturbation to help ODT methods jump out of a poor local minimum. Although the enhanced ODT methods cannot guarantee to obtain a global minimum, we start a new viewpoint of minimizing ODT energy which uses topological operations but mathematical methods. Practically, numerical experiments demonstrate that our enhanced ODT methods can generate a higher quality mesh than general ODT methods. Our future work will try to extend the enhanced ODT methods to polygonal meshes.

References

[1] W E L E Cline. *Marching cubes: a high resolution 3d surface construction algorithm,* Computer Graphics, 1987, 163-169.

[2] K Shimada, D C Gossard. *Bubble mesh: Automated triangular meshing of non-manifold geometry by sphere packing,* SMA '95: Proceedings of the third ACM symposium on Solid modeling and applications, 1995, 409-419.

[3] J Liu, S Li, Y Chen. *A fast and practical method to pack spheres for mesh generation,* Acta Mechanica Sinica, 2008, 24: 439-447.

[4] R Löner, P Parikh. *Generation of three-dimensional unstructured grids by the advancing-front method,* International Journal for Numerical Methods in Fluids, 1988, 8: 1135-1149.

[5] H McMorris, Y Kallinderis. *Octree-advancing front method for generation of unstructured surface and volume meshes,* AIAA Journal, 1997, 35 (6): 976-984.

[6] J R Shewchuk. *Delaunay refinement algorithms for triangular mesh generation,* Computational Geometry, 2002, 22 (1-3): 21-74.

[7] H Si. *Tetgen, a delaunay-based quality tetrahedral mesh generator,* ACM Transactions on Mathematical Software, 2015, 41(2): 1-36.

[8] N P Weatherill, O Hassan. *Efficient 3dimensional delaunay triangulation with automatic point creation and imposed boundary constraints,* International Journal for Numerical Methods in Engineering, 1994, 37 (12): 2005-2039.

[9] Y Guo, Y Hai. *Adaptive surface mesh remeshing based on a sphere packing method and a node insertion/deletion method,* Applied Mathematical Modelling, 2021, 98 (1): 1-13.

[10] B P Acharya, M Acharya. *Mesh optimization based on the centroid voronoi tessellation,* International Journal of Computer Mathematics, 2005, 82 (1): 125-129.

[11] L Chen, M Holst. *Efficient mesh optimization schemes based on optimal delaunay triangulations,* Computer Methods in Applied Mechanics and Engineering, 2011, 200 (9-12): 967-984.

[12] Y Huang, H Qin, D Wang. *Centroidal voronoi tessellation-based finite element superconvergence,* International Journal for Numerical Methods in Engineering, 2008, 76 (12): 1819-1839.

[13] C Jin, L Chen. *Optimal delaunay triangulations,* Journal of Computational Mathematics, 2004, 22 (2): 299-308.

[14] L Chen. *Mesh smoothing schemes based on optimal delaunay triangulations,* Proceedings of the 13th International Meshing Roundtable, Sandia National Laboratories, 2004: 109-120.

[15] P Alliez, D Cohen-Steiner, M Yvinec, M Desbrun. *Variational tetrahedral meshing,* ACM Transactions on Graphics 2005, 24 (3): 617-625.

[16] A Fabri, G J Giezeman, L Kettne, S Schirra, S Schonherr. *On the design of CGAL a computational geometry algorithms library,* Software: Practice and Experience, 2000, 30 (11): 1167-1202.

[17] Z Chen, J Cao, C Yang. *Topology improvement for constructing optimal delaunay triangulation,* Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics, 2011, 23 (12): 1967-1974.

[18] Z Chen, W Wang, B Lévy, L Liu, F Sun. *Revisiting optimal delaunay triangulation for 3d graded mesh generation,* SIAM Journal on Scientific Computing, 2014, 36 (3): A930-A954.

[19] J Pintér, J Szabo. *Global optimization algorithms: Theory and some applications,* Modelling and Optimization, 1986, 84: 704-713.

[20] L Chen, M Hoist. *Efficient mesh optimization schemes based on optimal delaunay triangulations,* Computer Methods in Applied Mechanics and Engineering, 2011, 200 (9-12): 967-984.

[21] H Erten, A Üngör, C Zhao. *Mesh smoothing algorithms for complex geometric domains,*Proceedings of International Meshing Roundtable, 2009, 175-193.

[22] D K Smith. *Numerical optimization,* Journal of the Operational Research Society, 2017, 52 (2): 245-245.

[23] Z Gao, Z Yu, M Holst. *Feature-preserving surface mesh smoothing via suboptimal delaunay triangulation,* Graph Models, 2013, 75 (1): 23-38.

[24] L Feng, P Alliez, L Busé, H Delingette, M Desbrun. *Curved optimal delaunay triangulation,* ACM Transactions on Graphics, 2018, 37 (4): 1-16.

[25] Z Gao, Z Yu, M Holst. *Quality tetrahedral mesh smoothing via boundary optimized delaunay triangulation,* Comput Aided Geom Des, 2012, 29 (9): 707-721.

[26] P Alliez, M Meyer, M Desbrun. *Interactive geometry remeshing,* ACM Transactions on Graphics, 2002, 21 (3): 347-354.

[27] V Surazhsky, C Gotsman. *Explicit Surface Remeshing,* Conference: First Eurographics Symposium on Geometry Processing, Germany, 2003, 25 (3): 527-536.

[28] Y Li, R Zhang, R Kobayashi, R Wonka. *Editing operations for irregular vertices in triangle meshes,* ACM Transactions on Graphics, 2021, 29 (6): 153.

[29] N Aghdaii, H Younesy, H Zhang. *567 meshes: Remeshing and analysis,* Computers and Graphics, 2021, 36 (8): 1072-1083.

[30] V Vidal, G Lavou, F Dupont. *Low budget and high fidelity relaxed 567-remeshing,* Computers and Graphics, 2015, 47: 16-23.

[31] V Y Rovenski. *Differential Geometry of Curves and Surfaces,* Springer Netherlands, 2006.

[32] N Meyer, M Desbrun, P Schroder, A H Barr. *Discrete differential-geometry operators for triangulated 2-manifolds,* Visualization and Mathematics, 2003, 3 (8-9): 35-57.

[33] F Shang, Z Zeng, J Liu, T Chen. *An automatic method for generating carbon nanostructure atomistic models using hexagonal meshes with properly distributed defects,* International Journal for Numerical Methods in Engineering , 2017, 110 (5): 403-419.

[34] T Jurczyk, B Gut. *Metric 3d surface mesh generation using delaunay criteria,* International Conference on Computational Science, 2006: 302-309.

[35] R Dyer, Z Hao, T Möller. *Voronoi-delaunay duality and delaunay meshes,* Acm Symposium on Solid and Physical Modeling, 2007.

[36] R Dyer, Z Hao, T Möller. *Delaunay mesh construction,* Eurographics Symposium on Geometry Processing, 2007.

[37] A I Bobenko, B A Springborn. *A discrete Laplace-Beltrami operator for simplicial surfaces,* Discrete and Computational Geometry, 2007, 38 (4): 740-756.

[38] D Engwirda, D Ivers. *Off-centre steiner points for delaunay-refinement on curved surfaces,* Computer-Aided Design, 2016, 72: 157-171.

[39] Y Hai, Y Guo, S Cheng, Y Hai. *Regular position-oriented method for mesh smoothing,* Acta Mechanica Solida Sinica, 2020, 34 (3): 437-448.

[40] M Livesu. *Cinolib: a generic programming header only c++ library for processing polygonal and polyhedral meshes,* Lecture Notes in Computer Science, 2019.

[41] D Vartziotis, T Athanasiadis, I Goudas, J Wipper. *Mesh smoothing using the geometric element transformation method,* Computer Methods in Applied Mechanics and Engineering, 2008, 197 (45-48): 3760-3767.

[1]Intelligent Science and Technology Academy of CASIC, Beijing 100854, China.
Email: yfguo@pku.edu.cn, xlab_zrl@163.com, sun_kewu@126.com

[2]Department of Mechanics and Engineering Science, Peking University, Beijing 100871, China.
Email: yqhye@pku.edu.cn, dongmo@pku.edu.cn

[3]Institute of Industry Technology, Guangzhou & Chinese Academy of Sciences, Guangzhou 511458, China.
Email: shangfeifei@gziit.ac.cn