

Valence Optimization and Angle Improvement for Molecular Surface Remeshing

Dawar Khan¹✉ · Alexander Plopski² · Yuichiro Fujimoto¹ · Masayuki Kanbara¹ · Zhanglin Cheng³ · Hirokazu Kato¹

Received: date / Accepted: date

Abstract Molecular surface mesh generation plays a vital role in molecular modeling and visualization. However, meshes extracted directly from PDB (Protein Data Bank) files have several issues such as small and large triangles, redundant elements, self-intersections, and irregular vertices. The state-of-the-art mesh improvement methods often fail to deal with these issues. In this paper, we present a novel method for valence optimization and angle improvement. For valence optimization, we remove the bad valence vertices with its neighbor triangle making regular holes in the mesh. The holes are filled in a careful manner to improve their valences as well as angle quality. We also use a segmentation-based surface remeshing which segments the mesh into random segments and then each segment is independently remeshed. In addition, a point insertion scheme is applied to minimize the ratio of obtuse triangles. Experimental results show that our method not only improves the maximal and minimal angles to an angle bound of $[30^\circ 120^\circ]$ but also improves the vertices' regularity, reduces the ratio of obtuse triangles, preserves the area and volume and always succeed with downstream applications.

Keywords Molecular Surface Remeshing · Mesh Quality · Valence Optimization · Molecular Modelling

✉ Correspondence: Dawar Khan
dawar@is.naist.jp

¹ Interactive Media Design Lab, Nara Institute of Science and Technology, Nara 630-0192, Japan.

² Department of Information Science, University of Otago, New Zealand.

³ Shenzhen VisuCA Key Lab, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China.

1 Introduction

Molecular surface meshing and molecular modelling is an interesting research direction used in many fields including computer graphics, mathematics, molecular biology, biophysics and chemistry. It plays a vital role in various phenomena such as protein folding, docking, implicit-solvent modeling, structure prediction, interaction of molecules, and measuring their areas/volumes [1, 2]. However, meshes generated directly from molecular data have several defects including self-intersections, small and large angles, and redundant and irregular vertices. There are several methods to improve the mesh quality prior to its use in the downstream applications. Some of these defects such as self-intersections have been addressed in SMOPT [3]. However, SMOPT fails in maximal and minimal angle improvements. Similarly, small and large angles have been improved up to an acceptable level by a recently proposed cut and fill (CAF) method [1]. However, CAF fails to achieve a considerable improvement in valence optimization.

Fig. 1 shows the main steps in molecular surface remeshing. Protein Data Bank (PDB) files are downloadable from their online repository (<https://www.rcsb.org/>). PDB2PQR [4] is a tool used to convert these files from PDB format to PQR format. The next step is mesh generation, where the mesh generating tools such as TMSmesh [5] are used to generate surface mesh from a PQR file. Unfortunately, these meshes are generated in raw form, thereby containing low-quality elements including zero degree angles, large angles, and irregular vertices [1]. Such raw meshes with low-quality elements are difficult to be directly used in downstream applications such as TetGen [6] or visualization (Fig. 2). Therefore, remeshing for quality enhancement is desired

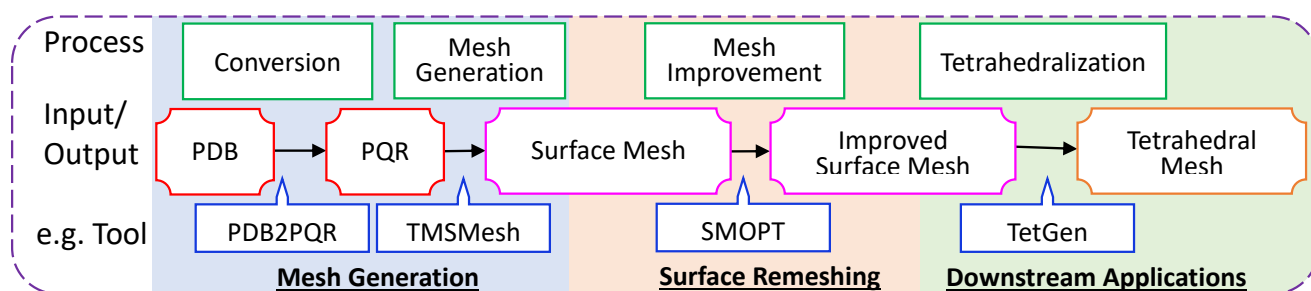


Fig. 1 The main pipeline of the molecular surface mesh generation and surface remeshing. The red color boxes represent the protein files (PDB and PQR), the magenta color boxes are surface meshes, the brown box represents tetrahedral mesh, the green boxes are processes and the blue boxes are examples of available tools for each particular process.

at this stage to refine the raw mesh. However, previous methods have no significant improvement in mesh quality. Fig. 2 shows an example of the molecular surface mesh in the downstream applications i.e. the electrostatic potentials calculated with AFMPB [7] mapped on the molecular surface.

Typically, a triangle with an angle smaller than 30° or obtuse angle (i.e. $\geq 90^\circ$) is called a bad triangle [8]. Similarly, it has been shown in the previous research that valence optimization can speed up the convergence. A vertex with valence (no. of adjacent edges) equal to four for a boundary vertex and valence six for an interior vertex is called optimal or regular vertex while the others are non-optimal. Generally, valence 5, valence 6, and valence 7 vertices are treated as optimal valences, and are found easy for surface remeshing [1, 8, 9].

In generic surface remeshing, the state-of-the-art methods have a significant improvement in the mesh quality. However, these methods from generic remeshing fail in molecular surface meshes due to additional issues of these meshes. Like generic meshes, it is challenging to preserve the area and volume as well as features and topology during molecular surface remeshing. Unlike the generic meshes, the meshes generated from molecular data have additional challenges including:

- The existence of very small and large angles (the small angles are equal to or nearly to 0°).
- High ratio of the defective elements such as isolated vertices, self-intersections, and redundant elements.
- Complexity of the molecular surface meshes in their shapes and number of vertices.

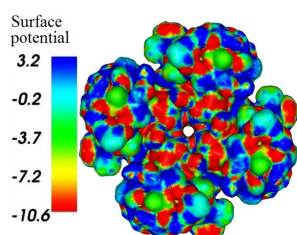


Fig. 2 Electrostatic potential on molecular surface, calculated with AFMPB [7].

In this paper, we present a simple method for valence optimization and angle improvement of molecular surface remeshing. We start with centroidal Voronoi tessellation (CVT) [10] initialization. The bad valence vertices are removed with their adjacent triangles in a careful way to create holes in regular structures. The holes are refilled with insertion of new vertices in the calculated feasible position which improves the vertices regularity and avoids the creation of small and large triangles. Then, we apply mesh segmentation to divide the mesh into random independent segments. Each segment is independently remeshed using realtime adaptive remeshing (RAR) [11]. After, the segment-wise remeshing, we remove small-angle triangles ($\theta < 30^\circ$). Then we apply a point insertion scheme that inserts a new point near each obtuse triangle to eliminate the obtuse triangle. In this manner, the ratio of obtuse triangles is significantly reduced. Finally, we apply Laplacian smooth and RAR method to the bad triangles and their neighbor triangles locally, with constraints to avoid the creation of new small triangles. We compared our results with four existing methods. Experimental results show that our method performs better than existing molecular surface remeshing methods in terms of angle improvements and valence optimization. In addition, there is no failure of the downstream applications for our output mesh. In summary, we have the following contributions in this paper.

- We propose a novel mechanism for valence optimization of molecular surface remeshing, that improves vertex regularity without losing the mesh quality.
- A divide-and-conquer strategy is used for segment-based surface remeshing which improves the mesh quality without introducing self-intersections or other defects. It preserves the input area and volume.
- Our method not only improves vertex regularity, but also achieves an angle bound of $[30^\circ \ 120^\circ]$, and improves other mesh quality parameters. It reduces the ratio of obtuse triangles significantly.

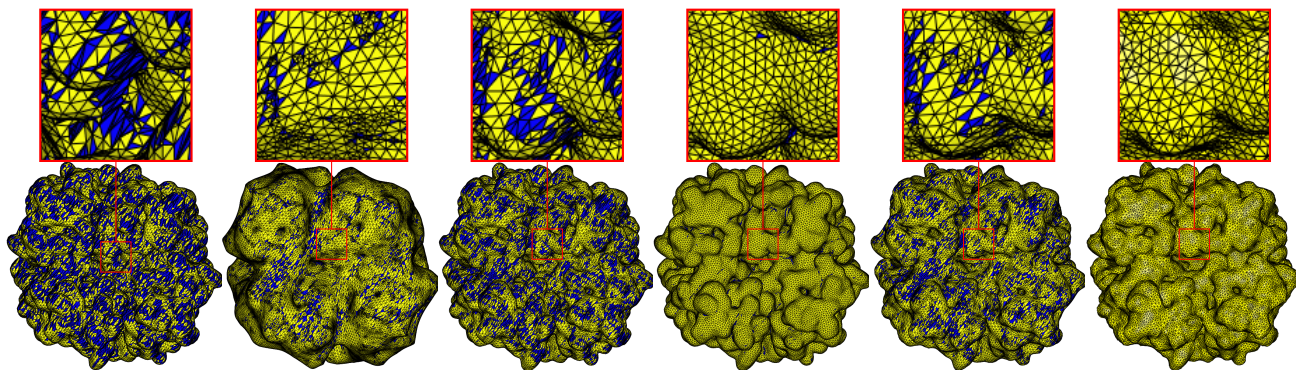


Fig. 3 Results from the state-of-the-art methods in molecular surface remeshing. Blue color indicates triangles with angle(s) smaller than 30° . From left to right, the input mesh (i.e. the mesh generated by TMSmesh 2.0 [5]), ISO2mesh [32], the Taubin method [35], CVT [10], the SMOPT method [3], and CAF method [1]. (PDB ID/ Molecular name *NaR1R4*).

2 Related Work

In generic surface remeshing, numerous methods have been proposed. These methods can be classified as mesh simplification-based methods [12], Delaunay insertion methods [13], advancing-front based method [14], field-based approaches [15], and local operators based mesh optimization [11]. Recent methods have a significant improvement in minimal and maximal angles and meshing quality for graphical models. For example, Yan and Wonka [16] used additional operators with CVT to avoid short Voronoi edges and remove obtuse and small triangles ($< 30^\circ$). However, their method does not work for noisy meshes.

The use of edge-based operators (edge split, edge collapse, vertex translation) [17] also has a significant achievement in meshing quality. These methods can improve the mesh quality for common graphical models, such as CAD models and human-made meshes. However, as molecular remeshing as additional challenges, these methods are often failed. The meshes generated from protein files have complex structures with many defects. Therefore, the generic meshing methods do not work to improve these meshes.

Researches reveal that small and large triangles badly affect the simulation results, and even a single bad triangle can affect the results [1,18,8]. To the best of our knowledge, a recent method of non-obtuse remeshing [8] in computer graphics better performs for angle improvement achieving an angle bound $[30^\circ 90^\circ]$. However, for molecular surface remeshing, there is no significant achievement in angle improvement with consideration of their area and volume preservations, valence optimization, and features preservations. Irregular vertices are also problematic in remeshing algorithms [9, 19]. In this regard, Aghdaii *et al.* [9] introduced regular meshes called 567 meshes, which was recently improved

with a low budget method [19], achieving comparatively higher geometric fidelity.

Molecular surfaces have different structures such as the van der Waals (VDWs) surface [20], the solvent accessible surface (SAS) [21], the solvent excluded surface (SES) [22], the minimal molecular surface [23], the molecular skin surface [24] and the Gaussian surface [25]. TMSmesh [2,5] is an efficient tool used for generating the triangular mesh of the Gaussian surface of biomolecules. Similarly, SESs are efficiently generated using alpha-shapes [26], marching tetrahedra [27], and analytical expressions [28].

In the context of molecular surface remeshing, several methods have been proposed. For example, SMOPT [3] is a tool specially designed for molecular surface mesh refinement which eliminates all redundant elements and self-intersections from the raw mesh. However, mesh refined with SMOPT still contains small angles. Recently, Khan *et al.* [1] proposed a method that starts with initialization using real-time adaptive remeshing (RAR) [11] followed by aspect ratio enhancement and a cut-and-fill module to eliminate invalid regions and small triangles. These selected regions are further improved with edge splitting, edge collapsing, and vertex translation. The results show that this method [1] better performs than SMPOT and other previous methods and that this method is able to generate a good quality mesh with an angle bound of $[30^\circ, 120^\circ]$. However, it fails to achieve a significant improvement in vertices regularities and it has a high ratio of obtuse triangles. Valence optimization can speed up the convergence. Vertices regularity is also a quality parameter used in surface remeshing [29].

Decherchi and Rocchia [30] proposed a ray-casting method for the triangulation of complex manifold surfaces in the nano bioscience field. They summarized various applications of molecular surfaces in implicit solvent model-

ing and simulations using the boundary element method (BEM) and the finite element method (FEM). Ray-casting methods [31] can also be used for fixing facet orientations. ISO2mesh [32] is a free available Matlab/octave-based toolbox used for mesh generation and processing. It is used to create tetrahedral meshes from surface meshes, and 3D binary and gray-scale volumetric images such as segmented MRI/CT scans. However, it fails to handle self-intersecting triangle pairs and small-angle triangles.

Wang and Yu [33] proposed a mesh smoothing scheme based on surface-fitting used for discrete, general-purpose mesh models. Initially, this smoothing scheme starts with the projection of each vertex onto the fitted surface. For a detailed study of molecular mesh generation and molecular visualization please refer to the recent survey paper [34]. The curvature is used to label all the vertices into four categories. Finally, post-processing modules are used to improve mesh quality. The experimental results reveal that this method [33] can handle different types of meshes, including molecular meshes, imaging data, and industrial models with mesh quality improvement and removal of small angles and short edges. However, there is no specific limit for small or large angles that this method can achieve.

Similarly, Cheng and Shi [36] used the restricted union of balls to improve the quality of molecular surface meshes. The method is able to improve the quality of large molecular meshes; however, it has a comparatively lower efficiency than other alternative methods. Similarly, Quan and Stamm used the advancing front approach and proposed a patch-wise molecular surface remeshing method [37] which is based on another previous method [38]. In addition to quality improvement and handling self-intersections, their method [37] fills the holes in molecular surface meshes. However, it fails in complex molecular surfaces.

In summary, there is a significant work on angle improvement and obtuse triangle removal in generic graphical models. However, there is no such algorithm for molecular surface remeshing. To the best of our knowledge, CAF method is the latest method that is specifically focused on angle improvement (see Fig. 3). However, CAF also has a high ratio of obtuse triangles. Furthermore, it has no significant improvement in vertices regularity.

3 Our Method

The main steps are given in Algorithm 1. Our method starts with CVT [10] initialization followed by valence optimization. Then, we segment the mesh into random

patches and apply RAR [11] method to each patch independently. After this, we eliminate small angles ($\theta < 30^\circ$). Then we use a point insertion strategy to minimize the number of obtuse angles. Finally, the segmentation is merged back to a single mesh and local refinements are applied using RAR and Laplacian method. The sub-modules are described in more detail as follows.

Algorithm 1 Molecular Surface Remeshing

- 1: CVT [10] Initialization.
 - 2: Valence Optimization. // Algorithm 2.
 - 3: Segment and apply RAR to each segment
 - 4: Remove small angles
 - 5: Queue all obtuse angles in pentagonal structures
 - 6: Make holes in pentagonal structures
 - 7: Fill holes
 - 8: Merge segments
 - 9: Repeat steps 4 to 7
 - 10: Apply RAR Locally
 - 11: Local Smooth
 - 12: END
-

3.1 Initialization

The mesh generated from the PQR file is typically found with different issues including zero degree angles, redundant vertices, and self-intersection, which makes the failure of different remeshing operators. Therefore, we apply CVT [10] as initialization (42 iterations as a default value). We select CVT for initialization since it provides a smooth distribution of the vertices in an efficient manner avoiding the major defects. Various methods including [1, 39, 40, 41] have used CVT either for initializations or to improve it.

3.2 Valence Optimization

The valence optimization method is presented in algorithm 2. For valence optimization, first, we apply two operators including edge flip and edge collapse operators (Fig. 4) with condition that it does not create an angle smaller than 30° .

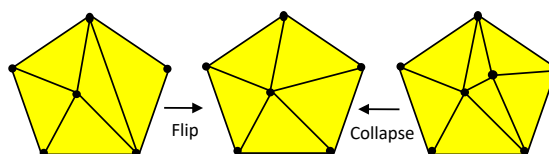


Fig. 4 Valence optimization with edge flip and collapse.

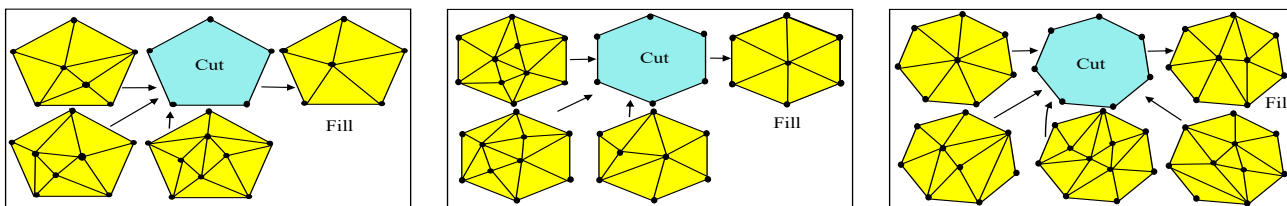


Fig. 5 Vertices removal and filling strategy. From left to right: Holes with 5, 6, and 7 boundary vertices.

Algorithm 2 Valence Optimization

```

1: Valence Flip
2: Edge Collapse
3: Delete vertices with bad valences (make holes)
4: for each hole do
5:   if (boundary vertices are 5 or 6) then
6:     Calculate feasible position and insert a single vertex
7:   else
8:     Insert two vertices
9:     Repeat step (1) and (2) (if required)
10:    Apply RAR [11] on newly filled hole
11:   end if
12: end for
13: END

```

In addition, we apply a vertex removal and refilling strategy for further valence optimization. We remove all vertices having non-optimal valences (i.e. not v_5, v_6, v_7). The holes created with vertices removal are filled in a careful manner to improve the regularity of the vertices (see Fig. 5). The vertices are removed in two phases (see Fig. 6). In the first phase, we remove the bad valence vertices with their adjacent edges. In the second phase, we remove vertices directly connected to the holes created in phase 1 and satisfying two conditions. (1) Vertex is a bad valence vertex (not V_5, V_6, V_7). (2) The total area triangles removed (in both phases) is lesser than $9 \times \bar{A}$, where \bar{A} is the average area of a triangle. These holes are filled carefully as shown in Fig. 5. For the holes with five or six boundary vertices, we insert one vertex whereas for holes with seven or more boundary vertices, we insert two vertices. In case of the two vertices insertion, if the valence of the vertices at the hole boundary is not optimal, it is optimized with edge collapse and edge flip operators.

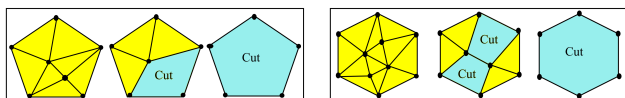


Fig. 6 Two phase vertices removal. First bad valence vertices (not V_5, V_6, V_7) are removed, then bad vertices directly connected to holes are removed. Making a hole with 5 (left) and 6 (right) boundary vertices.

The new points are inserted carefully to maintain the angle quality i.e. to avoid the creation of small or large angles. In this regard, for a single point insertion (i.e. v_5 and v_6 vertices) we draw circles/sphere around each boundary edge of the hole. The region outside all these circles/spheres is called feasible region for vertex insertion. Fig. 7 shows a v_5 vertex insertion in a calculated feasible region (the green color). Vertex insertion at this feasible position improves the regularity as well as the angles. For two vertices insertions, we also try to follow the same rules with an additional constraint that the edge between the two vertices must long enough that the opposite angles on both sides are 30° or above. In case, if these conditions are not satisfied, we filled the holes and apply the RAR method to the newly filled regions for angle improvements.

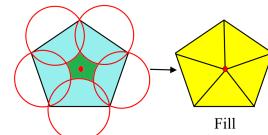


Fig. 7 Hole filling with angle improvement.

3.3 Mesh Segmentation

The molecular meshes are typically generated with high complexity, where the common operations sometime either fail or create new issues such as self-intersections. Similarly, deformation and shrinkage is another issue which causes decreasing the area/volume of the input model (see results of the ISO2mesh [32] in Fig. 3). To solve these issues, we follow a divide-and-conquer approach [42] of segment-based surface remeshing. We divide the mesh into random patches and apply the remeshing operations (RAR method [11]) to each patch independently. The lines drawn for segmentations are locked during this patch-wise remeshing. We used RAR method [11] for the segment-based remeshing because this method is optimal for local modification. It is a simple method with easy implementation. More importantly, it is computationally efficient. RAR uses an adaptive module $L(x)$ which calculates edge length L . Edges having a length longer $\frac{4}{3}L$ is split, whereas edges shorter than $\frac{4}{5}L$ are collapsed.

Fig. 8 shows the mesh segmentation used in our method. Segment-based surface remeshing on one side address-

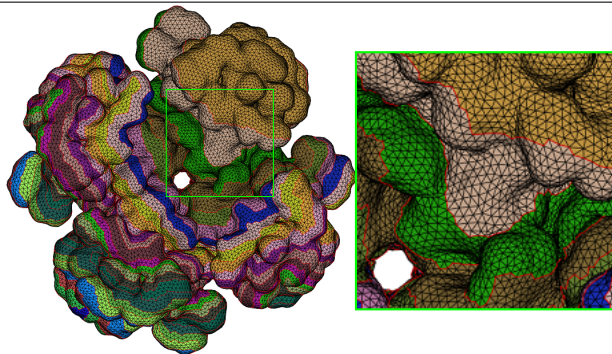


Fig. 8 Mesh segmentation for patch-wise remeshing. The red lines are locked and each segment is independently remeshed.

es the issue of mesh complexity by dividing the input mesh into segments. It preserves the area and volume of the input model, by locking the segmentation boundaries and thereby avoiding the possible deformation and shrinkage during surface remeshing. Furthermore, it helps in features and topology preservations. After remeshing each segment with RAR method, and angle improvements the segmentation boundaries are unlocked and the mesh is further refined with local smoothing (Section 3.6).

3.4 Small Angle Improvement

Our method eliminates all angles smaller than 30° . For small angle removal, we used edge-based operators including edge-collapse, edge-flip, and vertex translation. Initially, short edges having an opposite angle smaller than 20° are collapsed. Then we locally apply the Laplace smoothing to the vertices having small angles ($< 30^\circ$). Finally, we delete all vertices with small angles and refill the holes by directly connecting the boundary vertices of the holes. Then we apply RAR and Laplace smooth to newly filled holes. Similarly, we apply vertex translation to each vertex of each small triangle. For vertex translation, we calculate the new position via Laplace operator. In case if it improves the small angle the translation is applied, otherwise the new position is calculated near the old position of the vertex. The translation is only applied if it improves the small angle and does not introduce a new small angle. This process is repeated until all small angles ($< 30^\circ$) are removed.

3.5 Maximal Angle Improvement

One of our research aims is to improve the maximal angle and minimize the ratio of obtuse triangles. The

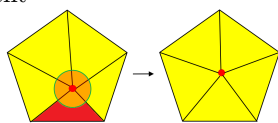


Fig. 9 Vertex translation.

angles have been generally improved via RAR (Section 3.3) and further refined via small angle improvement (Section 3.4). With small angle improvement, the maximal angle is also improved, and all the triangles with angle $\geq 120^\circ$ are removed. To further improve the maximal angle, we label the angles greater than the maximal angle threshold as bad angles. Similarly, the triangles with a bad angle are labeled as bad triangles. Then we apply special operators (vertex translation and point insertion) to each bad triangle. The maximal angle threshold is changed dynamically. First, we set the maximal angle threshold as 110° , and translate one vertex (i.e. the central vertex of the bad angle) of each bad triangle randomly around its old position to reach a feasible position where the triangle has no angle $\geq 110^\circ$. The same process is repeated for maximal angle threshold of 100° and 90° , respectively. Fig. 9 shows the vertex translation, where the circle indicates the feasible region for vertex translation and red triangle is the bad triangle. For each bad triangle, the vertex is translated if it improves the maximal angles and does not introduce a small angle ($< 30^\circ$), otherwise the translation is skipped. For each of these thresholds, the process is iterated 20 times.

In addition to vertex translation, we also use hole-filling strategy in pentagonal structures to minimize the ratio of obtuse triangles. Inspired by a recent 2D non-obtuse remeshing

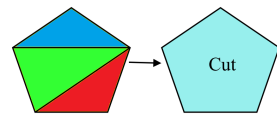


Fig. 10 Making a hole (to be refilled) for obtuse triangle (red color) removal.

method [40], we design pentagonal structures around each obtuse triangle (see Fig. 10 (left)). The pentagon contains three triangles including an obtuse triangle, its adjacent triangle on the longest edge, and the next adjacent on the longest edge. The two inside edges of the pentagonal structure are removed, which makes a hole (see Fig. 10 (Right)). Like hole-filling for valence optimization (see Fig. 7), the hole is filled with a new vertex insertion in a calculated feasible region. We draw circles/spheres around each boundary edge of the pentagon and calculate the feasible region where a new vertex is inserted. Fig. 11 shows an example of pentagonal structure design and point insertions for obtuse removal. Unlike 2D meshing [40], we use surface normal to ensure the three triangles are on the same plane. Our method easily eliminates all angles greater than or equal to 120° , and also significantly reduces the ratio of obtuse triangles. However, due to the complexity of molecular meshes and several other challenges, we are unable to eliminate all obtuse triangles. For example, if the obtuse triangle has no adjacent triangle on the same plane to make a pentagonal structure for point

insertion. Similarly, if the pentagonal structure is not regular enough, the feasible region (see the green region in Fig.7) for new vertex insertion might be empty.

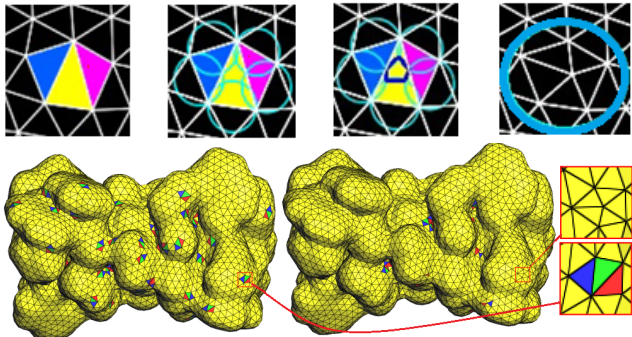


Fig. 11 Pentagonal structures and obtuse triangle removal. Top: An example from a 2D non-obtuse meshing method [40]. Bottom: An example with a molecular surface mesh. Bottom left: Before point insertion. Bottom right: After point insertion.

3.6 Local Smoothing

We flag the obtuse angles and small angles (if any) and apply local smooth. The local smooth is applied to the obtuse triangle and the triangle directly adjacent to it. The vertices are translated with a Laplace operator if it improves the maximal/minimal angles and does not create small angles. We also apply the RAR with the same constraints of small angles.

4 Experiments and Results

We implemented our algorithm using Graphite [43]. We performed the experiments using Intel Core i7 3.60 GHz with 32 GB RAM on a 64-bit Windows 10 operating system. We compared our results with relevant methods including SMOPT [3], CAF [1], RAR [11] and CVT [10].

4.1 Meshing quality parameters

For results analysis we used different statistical parameters to measure the mesh quality. These parameters include the minimal and average quality of triangle(s) denoted by Q_{min} and $Q_{avg.}$, respectively. The quality of a triangle t , is calculated as: $Q(t) = \frac{6}{\sqrt{3}} \frac{A_t}{p_t h_t}$, where A_t is the triangle area, p_t is half-perimeter and h_t is length of the longest edge of triangle [44]. The percentage ratio of the regular vertices (vertices with valence 5, 6 or 7) is also calculated which is denoted by $V567$. Similarly, minimal angles (θ_{min}), the average value of

minimal angles $\bar{\theta}_{min}$, maximal angle (θ_{max}) and the aspect ratio (AR) of triangles (maximum and average) are noted. AR represents the ration of the circum-radius to the twice of the in-radius of a triangle. It is calculated as:

$$AR = \frac{abc}{8(S-a)(S-b)(S-c)};$$

where a , b and c are the lengths of three sides of a triangle and $S = \frac{a+b+c}{2}$. An equilateral triangle has AR equal to 1. A high AR suggests low triangle quality. The smaller value indicates good quality unless $AR = 1$ [42].

In addition, the percentage ratios of the obtuse triangles ($\geq 90^\circ$) and the triangles with small angle ($< 30^\circ$) are also calculated. Furthermore, the number of atoms for each molecule, the area, volume, genus, self-intersections, and applicability in the downstream application (TetGen and AFMPB) are also recorded. In summary, our results are in three main categories including vertices regularity (valence optimization), mesh quality ($Q(t)$, θ_{min} , θ_{max} , etc.) and applicability in the downstream applications.

4.2 Valence optimization results

The state-of-the-art methods often fail for molecular surface remeshing. To the best of our knowledge, a recent method called CAF [1] gives comparative good results achieving an output mesh with an angle bound of $[30^\circ 120^\circ]$. However, this method has no significant improvement in valence optimization. Furthermore, the ratio of obtuse angles is also high. Our method attempts to handle these two issues with consideration of area and volume preservation and avoiding self-intersections. Fig. 12 shows the valence optimization results where the bad valence vertices are highlighted in red. The results show that our method can achieve better results than CAF [1] with a considerable improvement in valence optimization. Similarly, Fig. 13 shows a chart of comparison of our method with existing methods in terms of the regular vertices (V567). The ratio of regular vertices is found higher than CVT. The quantitative results are shown in Table 1.

4.3 Mesh quality results

From the state-of-the-art, we found that some methods have improvements in angle quality while others have significant improvements in vertex regularity. Some methods fail to remove self-intersections. Our method

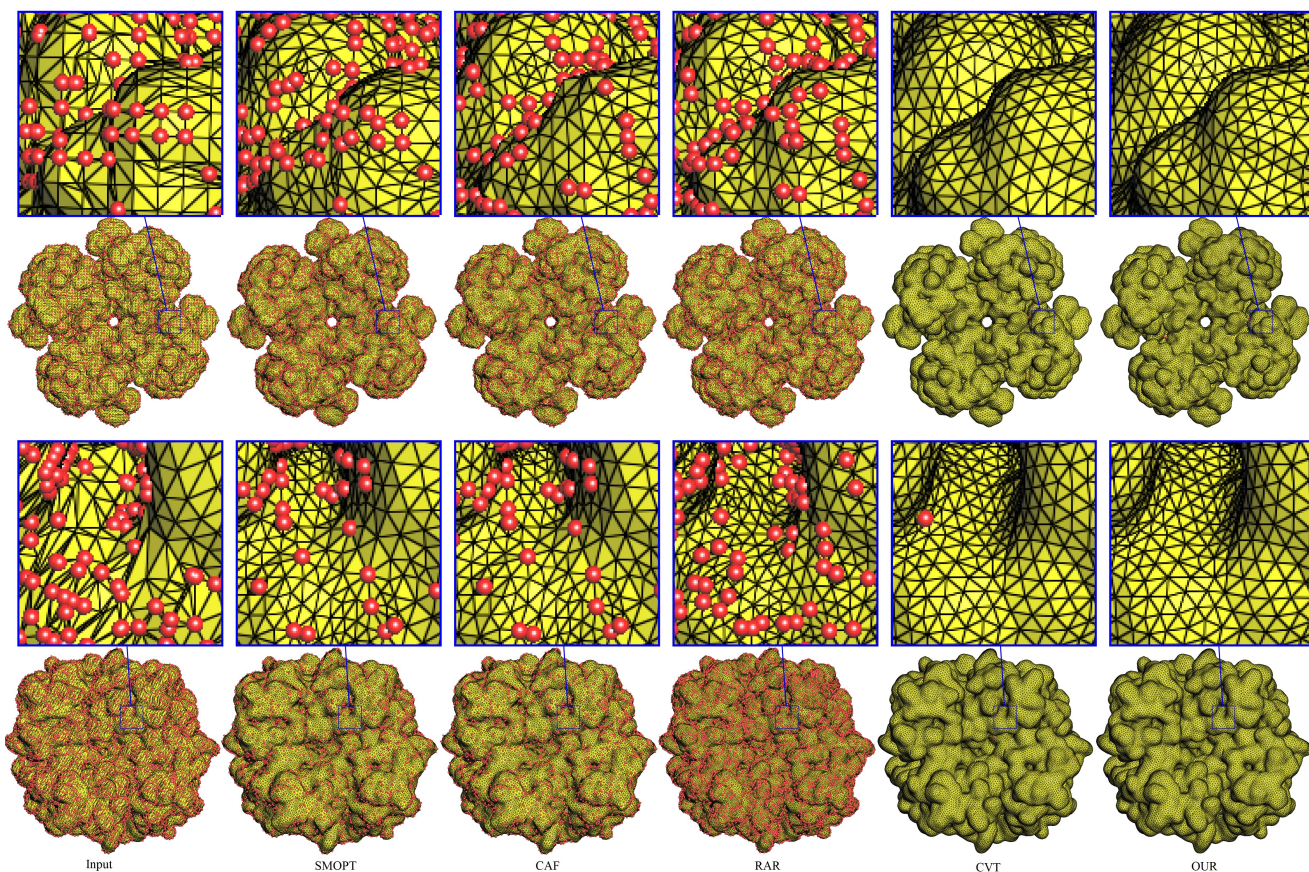


Fig. 12 Valence optimization results. The red color represents irregular vertices (i.e. not V567). From left to right: Top (1bl8): Input (V567: 75.84%), SMOPT (V567: 76.19%) [3], CAF (V567: 79.27%) [1], RAR (V567: 72.88%) [11], CVT (V567: 99.57%) [10], and Our (V567: 99.81%). Bottom (NaR1R4): Input (63.51%), SMOPT (64.96%), CAF (83.95%), RAR (76.52%), CVT (99.92%), and Our (99.96%).

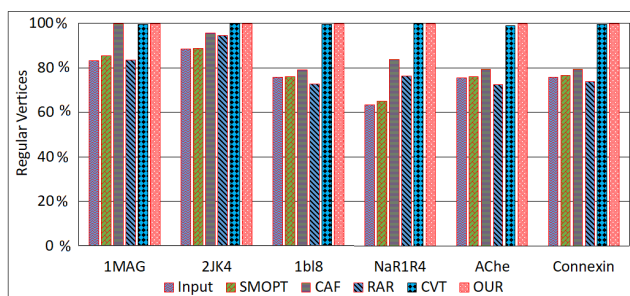


Fig. 13 The % value of regular vertices (V567), and comparison with the state-of-the-art methods using different molecular surface meshes.

attempts to give an optimal output mesh having a significant improvement in all these parameters. Table 1 shows the quantitative results of our method and few existing methods including SMOPT, CAF, RAR, and CVT. Fig. 14 shows the visual results in the form of different molecular meshes. The obtuse triangles are highlighted in red color. Our method not only maintains the angle bound of $[30^\circ 120^\circ]$ like CAF [1] but also has

a significant improvement in obtuse angle removal and provides higher regularity than CVT.

4.4 Applicability in the downstream applications

Table 2 shows results concerned with area and volume preservation during remeshing, and the success/failure of the downstream applications on the output meshes. AFMPB (solvation energy calculation) and TetGen (volumetric meshing) are used as downstream applications. Furthermore, we noted the Genus and counted the number of self-intersecting triangles. We found our method has no self-intersection and always succeeds in the downstream applications. Furthermore, it has minor changes in the area and volume of the input mesh. However, RAR method preserves the input genus better than our method.

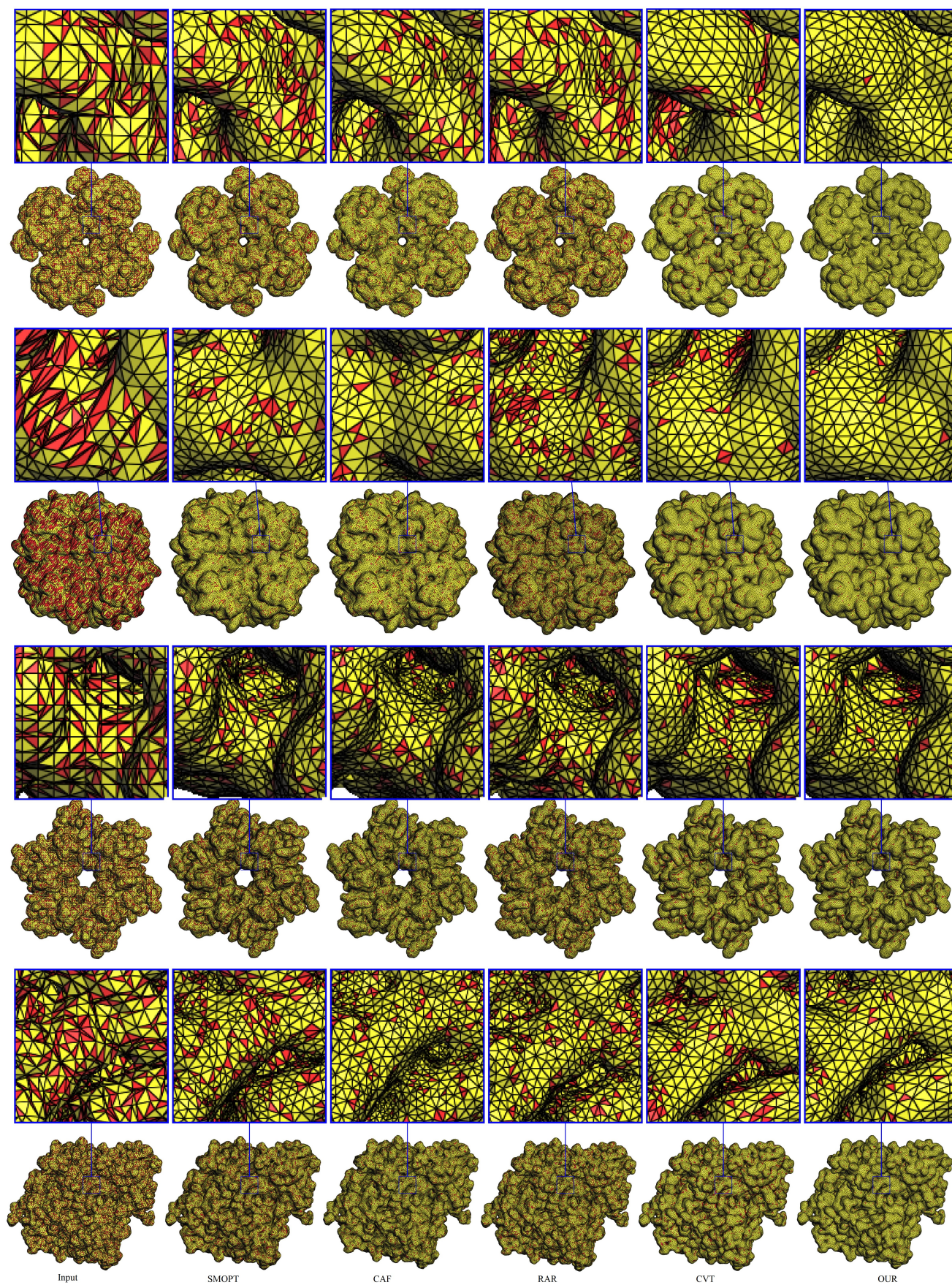


Fig. 14 Surface Remeshing results. The red color represents obtuse triangles. From top to bottom the protein ID or molecular names are: 1bl8, NaR1R4, Connexin and AChE.

Table 1 Comparative surface remeshing results. Our method shows a significant improvement in the mesh quality. The angles are measured in degree. The model names are the PDB IDs/molecular names.

Model	Method	#v	Q_{min}	Q_{avg}	θ_{min}	$\bar{\theta}_{min}$	θ_{max}	$\theta < 30^\circ$	$\theta \geq 90^\circ$	Reg. v's	AR_{max}	AR_{avg}
1MAG	Input	4824	0.0002	0.6701	0.0069	36.41	176.09	27.67%	26.25%	83.31 %	undef.	undef.
	CVT	4811	0.4316	0.9024	25.32	52.15	123.42	0.10%	1.32%	99.58%	2.41	1.04
	SMOPT	4718	0.0808	0.7958	3.37	44.19	167.94	8.42 %	17.28%	85.59%	51.30	1.21
	CAF	4719	0.5523	0.9054	30.52	52.30	109.043	0.0%	1.13%	99.84%	1.67	1.03
	RAR	4888	0.2758	0.8244	15.3	52.48	142.58	2.29%	11.19%	83.67%	5.17	1.12
	OUR	4892	0.5598	0.9043	30.4	54.01	89.58	0.0%	0.0%	99.93%	1.55	1.03
2JK4	Input	23312	0.1137	0.75867	5.20	41.51	161.27	12.97%	20.49%	88.41%	23.71	1.23
	CVT	23106	0.2880	0.8899	11.53	50.84	117.83	0.16%	2.02%	99.97%	3.40	1.05
	SMOPT	23139	0.2700	0.8402	12.09	47.22	142.27	1.53%	8.11%	88.86%	5.24	1.10
	CAF	23134	0.4660	0.8762	30.09	49.97	119.77	0.0%	3.02%	95.73%	2.14	1.05
	RAR	23505	0.3782	0.8735	19.89	53.13	129.37	0.12%	3.45%	94.73%	2.96	1.06
	OUR	23435	0.4157	0.8843	30.05	51.10	119.86	0.0%	0.93%	99.73%	3.10	1.07
1bl8	Input	85904	0.0	0.5191	0.0	26.30	179.92	56.46%	41.0%	75.84%	undef.	undef.
	CVT	85466	0.0020	0.7262	0.07	39.66	179.32	23.71%	20.32%	99.57%	undef.	undef.
	SMOPT	85601	0.0046	0.726	0.22	39.20	179.87	19.04%	26.59%	76.19%	undef.	undef.
	CAF	74623	0.4679	0.8141	30.0	45.31	119.53	0.0%	14.61%	79.27%	2.13	1.12
	RAR	83274	0.0070	0.7785	0.46	49.44	179.08	6.95%	19.77%	72.88%	7699.65	1.76
	OUR	78735	0.3956	0.8961	30.3	51.10	109.14	0.0%	0.5%	99.81%	2.15	1.04
NaR1R4	Input	63310	0.0	0.5135	0.0	26.319	179.43	58.21%	36.11%	63.51%	undef.	undef.
	CVT	61247	0.03452	0.8151	1.62	45.56	174.9	9.24%	11.96%	99.92%	undef.	undef.
	SMOPT	61129	0.0437	0.7063	1.88	38.35	173.66	23.41 %	29.82%	64.96%	179.55	1.47
	CAF	57024	0.4682	0.8233	30.0	45.90	119.49	0.0%	11.83%	83.95%	2.12	1.11
	RAR	88921	0.0241	0.7714	1.49	49.29	176.79	8.5%	20.17%	76.52%	641.51	1.22
	OUR	59781	0.4615	0.8362	30.05	49.36	119.92	0.0%	5.72%	99.96%	2.10	1.11
AChE	Input	158088	0.0	0.5162	0.0	25.95	179.99	57.59%	41.53%	75.58%	undef.	undef.
	CVT	150606	0.0008	0.6708	0.04	35.97	179.86	32.27%	26.74%	99.07%	undef.	undef.
	SMOPT	155565	0.0140	0.7250	0.59	39.10	177.19	19.62%	26.57%	76.16%	undef.	undef.
	CAF	124326	0.4656	0.8142	30.0	45.32	119.81	0.0%	13.26%	79.45 %	2.14	1.12
	RAR	153193	0.0	0.7788	0.0	49.58	179.49	5.81%	19.80%	72.54 %	25428	1.80
	OUR	159543	0.4643	0.8311	30.02	49.90	119.92	0.0%	9.90%	99.75 %	2.23	1.10
Connexin	Input	107500	0.0	0.5248	0.0	26.53	179.99	56.21%	40.97%	75.87%	undef.	undef.
	CVT	104255	0.0040	0.7407	0.16	40.41	179.35	20.0%	20.0%	99.64%	undef.	undef.
	SMOPT	105645	0.0236	0.7364	1.08	39.85	176.58	16.98%	25.37%	76.78%	614.1	1.46
	CAF	101574	0.4658	0.8158	30.0	45.44	119.79	0.0%	20.96%	79.46%	2.14	1.12
	RAR	110255	0.0109	0.7848	0.70	49.88	178.55	4.8%	18.73%	73.81%	3134	1.29
	OUR	110255	0.4172	0.8329	30.0	50.41	119.87	0.0%	13.73%	99.86%	2.08	1.07

4.5 Discussion

The literature review reveals that molecular surface remeshing is an active research area with special challenges. For general graphical models, there are very good methods to improve mesh quality. However, these methods often fail for molecular surface remeshing. There are a number of methods specifically used for molecular surface remeshing. However, the molecular surface meshing has no significant improvements. For example, SMOPT is a recent molecular surface remeshing method that can remove all intersecting elements, however, it fails to remove small and large angles. Similarly, CAF is another recent method for molecular surface remeshing that improves the minimal and maximal angles, however, the valence optimization is not satisfactory.

The results reveal that our method improves the angles

to an angle bound of $[30^\circ 120^\circ]$. The ratio of the obtuse triangles is also significantly reduced. Furthermore, our method outperforms in terms of vertex regularities than existing methods including CVT. There was no defect found that causes the failure in the downstream applications.

Our method improves mesh quality in an optimal way and provides better mesh quality than CAF and other molecular surface remeshing methods. Furthermore, it gives a higher valence regularity than CVT and removes all defects that may cause the failure of the downstream applications. Similarly, the area and volume preservation are also significant. However, unlike RAR, the input genus is not preserved with our method. Furthermore, there are two main limitations of our current work. First, our method significantly reduces the number of obtuse triangles, however, it is unable to remove all obtuse triangles. Second, our method is strongly de-

Table 2 Results concerned with area (unit: A^2) and volume (unit: A^3) preservation and applicability in the downstream applications (AFMPB (unit: $kcal/mol$) and TetGen). The model names are the PDB IDs/ molecular names.

Model	Natoms	Method	Area	Volume	Genus	Self-intersect.	TetGen	AFMPB (solv. energy)
1MAG	552	Input	2807.82	4531.35	-1	0	✓	4.07610e+01
		CVT	2625.30	4499.45	3	0	✓	3.22092e+01
		SMOPT	2195.70	4174.41	3	0	✓	2.63009e+01
		CAF	2624.00	4499.73	2	0	✓	3.22525e+01
		RAR	2771.85	4530.85	-1	0	✓	3.44965e+01
OUR	2705.92	4529.30	-1	0	✓	3.42362e+01		
2JK4	4393	Input	14787.54	37962.87	4	0	✓	-1.59041e+03
		CVT	14307.57	37908.52	5	0	✓	-1.68641e+03
		SMOPT	12770.33	37284.66	6	0	✓	-1.63068e+03
		CAF	14286.88	37911.41	5	0	✓	-1.65811e+03
		RAR	14721.21	37964.70	4	0	✓	-1.61630e+03
OUR	14623.46	37920.88	5	0	✓	-1.60238e+03		
1bl8	5892	Input	22094.98	50919.54	103	0	✓	Failed
		CVT	21166.78	50688.84	97	0	✓	-1.34629e+03
		SMOPT	19673.38	49935.09	103	0	✓	-1.50455e+03
		CAF	19629.99	50901.04	58	0	✓	-1.39705e+03
		RAR	21612.95	50918.6	103	50	×	Failed
OUR	21312.46	50823.17	32	0	✓	-1.34286e+03		
NaR1R4	7443	Input	21301.15	76894.66	-1	0	✓	-1.54776e+03
		CVT	20480.95	76807.18	14	0	✓	-1.59706e+03
		SMOPT	18574.04	75558.13	14	0	✓	-1.85589e+03
		CAF	18723.79	76881.15	8	0	✓	-1.74595e+03
		RAR	20924.04	76970.50	-1	2	×	-1.56296e+03
OUR	20511.28	76811.54	0	0	✓	-1.59712e+03		
AChE	8280	Input	37653.08	69058.97	225	0	✓	Failed
		CVT	35760.42	68699.76	341	0	✓	-2.07574e+03
		SMOPT	33222.25	67940.30	345	0	✓	-2.48270e+03
		CAF	30099.40	70371.76	93	0	✓	-2.37904e+03
		RAR	36734.38	69081.50	225	385	×	Failed
OUR	35978.74	69011.32	93	0	✓	-2.14293e+03		
Connexin	19883	Input	55604.89	209415.59	31	0	✓	Failed
		CVT	53272.10	208731.02	94	0	✓	-1.04865e+04
		SMOPT	49695.69	206505.02	96	0	✓	-1.08436e+04
		CAF	50806.90	209318.82	68	0	✓	-1.06104e+04
		RAR	54600.48	209396.56	31	36	×	Failed
OUR	53938.17	209232.45	74	0	✓	-1.04332e+04		

pendent on the previous methods including CVT (for initialization) and RAR.

5 Conclusion

We proposed a novel method for valence optimization and angle improvement of the molecular surface remeshing. Our method searches for bad valence vertices and removes it with its neighbor vertices making holes in the mesh. The holes are filled based on the boundary vertices of each hole. In addition, we segment the mesh into random patches and improve the minimal angle and maximal angles in a divide-and-conquer fashion. To minimize the ratio of the obtuse triangles we extended a 2D non-obtuse meshing algorithm [40]. We found our method with better performance in terms of mesh quality, vertices regularity, area/volume preservation as well as applicable for the downstream applications.

In the future, we are planning for non-obtuse molecular surface remeshing to remove all obtuse triangles. We are also planning to develop a mesh extraction algorithm

that extracts surface mesh from PQR files without any major issues.

Acknowledgements This work is partially funded by the Japan Society for the Promotion of Science (JSPS) KAKENHI (19K24346), the National Natural Science Foundation of China (61972388), and Shenzhen Basic Research Program (JCYJ20180507182222355). We are thankful to the anonymous reviewers for their valuable comments.

Conflict of interest

The authors declare that they have no conflict of interest.

References

- Dawar Khan, Dong-Ming Yan, Sheng Gui, Benzhuo Lu, and Xiaopeng Zhang. Molecular surface remeshing with local region refinement. *International Journal of Molecular Sciences*, 19(5):1383:1–1383:20, 2018.
- Minxin Chen, Bin Tu, and Benzhuo Lu. Triangulated manifold meshing method preserving molecular surface topology. *Journal of Molecular Graphics and Modelling*, 38:411–418, 2012.
- Tiantian Liu, Minxin Chen, Yu Song, Hongliang Li, and Benzhuo Lu. Quality improvement of surface triangular mesh using a modified laplacian smoothing approach avoiding intersection. *PLOS ONE*, 12(9):1–16, 09 2017.
- Todd J. Dolinsky, Jens E. Nielsen, J. Andrew McCammon, and Nathan A. Baker. PDB2PQR: An automated pipeline for the setup of Poisson-Boltzmann electrostatics calculations. *Nucleic Acids Research*, 32(suppl-2):W665–W667, 2004.
- Tiantian Liu, Minxin Chen, and Benzhuo Lu. Efficient and qualified mesh generation for Gaussian molecular surface using adaptive partition and piecewise polynomial approximation. *SIAM Journal on Scientific Computing*, 40(2):B507–B527, 2018.
- Hang Si. Tetgen, a Delaunay-based quality tetrahedral mesh generator. *ACM Trans. Math. Softw.*, 41(2):11:1–11:36, February 2015.
- Huang J Pitsianis NP Sun X Lu-B Zhang B, Peng B. Parallel afmpb solver with automatic surface meshing for calculations of molecular solvation free energy. *Computer Physics Communications*, (190):173–181, 2015.
- Y. Wang, D. Yan, X. Liu, C. Tang, J. Guo, X. Zhang, and P. Wonka. Isotropic surface remeshing without large and small angles. *IEEE Transactions on Visualization and Computer Graphics*, 25(7):2430–2442, July 2019.
- Nima Aghdaii, Hamid Younesy, and Hao Zhang. 5-6-7 Meshes: Remeshing and analysis. *Computer and Graphics (Extended version from GI’12)*, 36(8):1072–1083, 2012.
- Yang Liu, Wenping Wang, Bruno Lévy, Feng Sun, Dong-Ming Yan, Lin Lu, and Chenglei Yang. On centroidal Voronoi tessellation - energy smoothness and fast computation. *ACM Trans. on Graphics*, 28(4):101:1–101:11, 2009.
- Marion Donyach, David Vanderhaeghe, Loïc Barthe, and Mario Botsch. Adaptive remeshing for real-time mesh deformation. In *Eurographics Short Papers Proceedings*, pages 29–32, 2013.

12. Yong-Jin Liu, Chunxu Xu, Dian Fan, and Ying He. Efficient construction and simplification of Delaunay meshes. *ACM Trans. on Graphics*, 34(6):174:1–174:13, 2015.
13. Siu-Wing Cheng, Tamal K. Dey, and Jonathan R. Shewchuk. *Delaunay Mesh Generation*. CRC Press, 2012.
14. John Schreiner, Carlos E. Scheidegger, Shachar Fleishman, and Cláudio T. Silva. Direct (re)meshing for efficient surface processing. *Computer Graphics Forum (Proc. EUROGRAPHICS)*, 25(3):527–536, 2006.
15. Wenzel Jakob, Marco Tarini, Daniele Panozzo, and Olga Sorkine-Hornung. Instant field-aligned meshes. *ACM Trans. on Graphics (Proc. SIGGRAPH Asia)*, 34(6):189:1–189:15, 2015.
16. Dong-Ming Yan and Peter Wonka. Non-obtuse remeshing with centroidal Voronoi tessellation. *IEEE Trans. on Vis. and Comp. Graphics*, 22(9):2136–2144, 2016.
17. K. Hu, D. M. Yan, D. Bommers, P. Alliez, and B. Benes. Error-bounded and feature preserving surface remeshing with minimal angle improvement. *IEEE Transactions on Visualization and Computer Graphics*, 23(12):2560–2573, 2017.
18. Guanghui Ma, Juntao Ye, Jituo Li, and Xiaopeng Zhang. Anisotropic strain limiting for quadrilateral and triangular cloth meshes. *Comput. Graph. Forum*, 35:89–99, 2016.
19. Vincent Vidal, Guillaume Lavou, and Florent Dupont. Low budget and high fidelity relaxed 567-remeshing. *Computers & Graphics*, 47:16 – 23, 2015.
20. M. Gerstein, F. M. Richards, M. S. Chapman, and M. L. Connolly. Protein surfaces and volumes: measurement and use. *International Tables for Crystallography*, f:531–545, 2000.
21. B Lee and F. M. Richards. The interpretation of protein structures: estimation of static accessibility. *Journal of Molecular Biology*, 55(3):379–400, 1971.
22. F M Richards. Areas, volumes, packing, and protein structure. *Annual Review of Biophysics Bioengineering*, 6(6):151, 1977.
23. P. W. Bates, G. W. Wei, and S. Zhao. Minimal molecular surfaces and their applications. *Journal of Computational Chemistry*, 29(3):380, 2008.
24. H. Edelsbrunner. Deformable smooth surface design. *Discrete & Computational Geometry*, 21(1):87–115, Jan 1999.
25. Tiantian Liu, Minxin Chen, and Benzhuo Lu. Parameterization for molecular gaussian surface and a comparison study of surface mesh generation. *Journal of Molecular Modeling*, 21(5):1–14, 2015.
26. Wenyu Chen, Jianmin Zheng, and Yiyu Cai. Kernel modeling for molecular surfaces using a uniform solution. *Computer-Aided Design*, 42(4):267 – 278, 2010.
27. Shek Ling Chan and Enrico O. Purisima. Molecular surface generation using marching tetrahedra. *Journal of Computational Chemistry*, 19(11):1268–1277, 1998.
28. Michel F. Sanner, Arthur J. Olson, and Jean-Claude Spehner. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305–320, 1996.
29. Yifan Men, Zeyu Shen, Dawar Khan, and Dong-Ming Yan. Improving regularity of the centroidal Voronoi tessellation. In *SIGGRAPH Posters*, pages 66:1–66:2. ACM, 2018.
30. Sergio Decherchi and Walter Rocchia. A general and robust ray-casting-based algorithm for triangulating surfaces at the nanoscale. *PLOS ONE*, 8(4):1–15, 04 2013.
31. Kenshi Takayama, Alec Jacobson, Ladislav Kavan, and Olga Sorkine-Hornung. A simple method for correcting facet orientations in polygon meshes based on ray casting. *Journal of Computer Graphics Techniques (JCGT)*, 3(4):53–63, 2014.
32. Qianqian Fang and D. A. Boas. Tetrahedral mesh generation from volumetric binary and grayscale images. In *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1142–1145, June 2009.
33. Jun Wang and Zeyun Yu. A novel method for surface mesh smoothing: Applications in biomedical modeling. In Brett W. Clark, editor, *Proceedings of the 18th International Meshing Roundtable*, pages 195–210, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
34. Sheng Gui, Dawar Khan, Qin Wang, Dong-Ming Yan, and Ben-Zhuo Lu. Frontiers in biomolecular mesh generation and molecular visualization systems. *Visual Computing for Industry, Biomedicine, and Art*, 1(1):7:1–7:13, Sep 2018.
35. Gabriel Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '95*, pages 351–358, New York, NY, USA, 1995. ACM.
36. Ho-Lun Cheng and Xinwei Shi. Quality mesh generation for molecular skin surfaces using restricted union of balls. *Computational Geometry*, 42(3):196 – 206, 2009.
37. Chaoyu Quan and Benjamin Stamm. Meshing molecular surfaces based on analytical implicit representation. *Journal of molecular graphics and modelling*, 71:200 – 210, 2017.
38. Chaoyu Quan and Benjamin Stamm. Mathematical analysis and calculation of molecular surfaces. *Journal of Computational Physics*, 322:760 – 782, October 2016.
39. Long Chen and Michael Holst. Efficient mesh optimization schemes based on optimal Delaunay triangulations. *Computer Methods in Applied Mechanics and Engineering*, 200(9):967 – 984, 2011.
40. Dawar Khan, Dong-Ming Yan, Yiqun Wang, Kaimo Hu, Juntao Ye, and Xiaopeng Zhang. High-quality 2D mesh generation without obtuse and small angles. *Computers & Mathematics with Applications*, 75(2):582 – 595, 2018.
41. Dong-Ming Yan and Peter Wonka. Gap processing for adaptive maximal Poisson-disk sampling. *ACM Trans. on Graphics*, 32(5):148:1–148:15, 2013.
42. Dawar Khan, Dong-Ming Yan, Fan Ding, Yixin Zhuang, and Xiaopeng Zhang. Surface remeshing with robust user-guided segmentation. *Computational Visual Media*, 4(2):113–122, Jun 2018.
43. Graphite. <http://alice.loria.fr/index.php/software/3-platform/22-graphite.html>. [Online; accessed Feb. 21, 2020].
44. P. Frey and H. Borouchaki. Surface mesh evaluation. In *6th Intl. Meshing Roundtable*, pages 363–374, 1997.