

Research Article

Open Access

Tiantian Liu, Shiyang Bai, Bin Tu, Minxin Chen, and Benzhuo Lu*

Membrane-Channel Protein System Mesh Construction for Finite Element Simulations

DOI 10.1515/mlbmb-2015-0008

Received August 1, 2015; accepted October 20, 2015

Abstract: We present a method of constructing the volume meshes of the membrane-channel protein system for finite element simulation of ion channels. The membrane channel system consists of the solvent region and the membrane-protein region. Our method focuses on labeling the tetrahedra in the solvent and membrane-protein regions and collecting the interface triangles between different regions. It contains two stages. Firstly, a volume mesh conforming the surface of the channel protein is generated by the surface and volume mesh generation tools: TMSmesh and TetGen. Then a walk-and-detect algorithm is used to identify the pore region to embed the membrane correctly. This method is shown to be robust because of its independence of the pore structure of the ion channels. In addition, we can also get the information of whether the ion channel is open or closed by the walk-and-detect algorithm. An on-line meshing procedure will be available at our website www.continuummodel.org.

Keywords: membrane; ion channel; tetrahedral mesh construction; walk-and-detect

1 Introduction

Ion channels are pore-forming membrane proteins whose function is to control the diffusion of ions across biological membranes. Membranes are electrical insulators and build a hydrophobic and low dielectric barrier to hydrophilic and charged molecules. Ion channels build a high conducting, hydrophilic pathway across the hydrophobic interior of the membrane for ions. Therefore, the membrane plays an important role in ion conductance simulations of an ion channel.

Various theoretical and computational approaches have been developed to help understand the biological mechanism of ion channels. Molecular dynamics simulations [1–3], Brownian dynamics simulations [4–7] and continuum modeling [8–11] are the most commonly used techniques in the field. A widely used continuum model for simulating ionic transport is based on the Poisson-Nernst-Planck (PNP) equation [9, 12]. A number of numerical algorithms including finite difference [13, 14], finite element [15–19], spectral element [20], and finite volume methods [21] have been utilized in the past two decades to solve the PNP equation. The finite element method has the advantage of handling complex geometries and curved boundaries. However, due to the complexity of ion channel structures, it is not a trivial task to build a high-quality sur-

***Corresponding Author: Benzhuo Lu:** State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, National Center for Mathematics and Interdisciplinary Sciences, Chinese Academy of Sciences, Beijing 100190, China, E-mail: bzlu@lsec.cc.ac.cn. Tel: +86 1082541904

Tiantian Liu: State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, National Center for Mathematics and Interdisciplinary Sciences, Chinese Academy of Sciences, Beijing 100190, China

Shiyang Bai: State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, National Center for Mathematics and Interdisciplinary Sciences, Chinese Academy of Sciences, Beijing 100190, China and Beijing DoOnCloud Technology Co, LTD.

Bin Tu: National Center for NanoScience and Technology, Chinese Academy of Sciences, Beijing 100190, China

Minxin Chen: Center for System Biology, Department of Mathematics, Soochow University, Suzhou 215006, China

 © 2015 Tiantian Liu et al., licensee De Gruyter Open.

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 3.0 License.

face and volume mesh for the simulation. In this paper, we present a method to generate qualified surface and tetrahedra meshes for the finite element simulation.

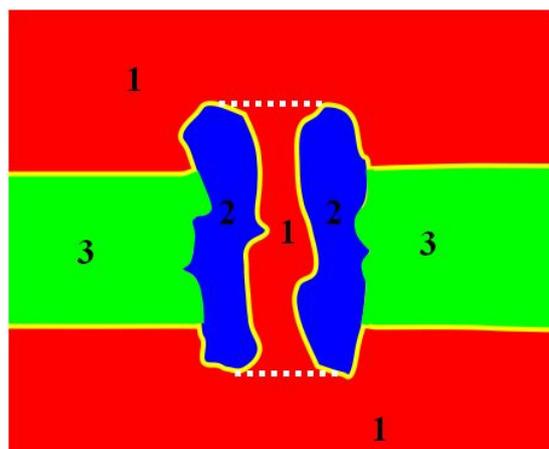


Figure 1: 2D schematic picture for the cross section of an ion channel system. The solvent region is labeled by 1, the channel protein is labeled by 2 and the membrane is labeled by 3. The solvent part between the white dotted lines is the pore region.

Figure 1 shows a cross-section of the membrane channel system. The system consists of the channel protein, the membrane and the solvent region. The channel protein region and the membrane region constitute the solute region, also called the membrane-protein region. The membrane-protein is located in a solvent box. The box is separated into two parts. The up and down parts are connected by the open channel pore. The part of the solvent region between the two white dotted lines is the pore region. The yellow curve in Figure 1 shows the interface of the solute region and the solvent region (the surface of the membrane-protein region). During the process of the finite element simulation of the continuum modeling of the ion channel, usually different boundary conditions need to be enforced on different parts of the boundaries/interfaces. Therefore, the triangular mesh on the interface should be collected and marked.

We have developed a method to generate high-quality surface and volume meshes for the membrane channel system [18, 19]. The method essentially has four components: surface meshing, quality improvement, volume mesh generation and membrane mesh construction. The triangulated molecular surface can be generated by some surface mesh generator, such as TSMesh [22, 23], MSMS [24], NanoShaper [25], Molsurf [26, 27], GAMer [28], EDTsurf [29] and so on. These softwares generate triangulated mesh based on different definitions of the molecular surface, including van der Waals surface, the solvent accessible surface [30], the solvent excluded surface [31], Gaussian surface [32] and the molecular skin surface [33]. If necessary, the surface mesh quality can be improved by such as ISO2mesh and TransforMesh [34]. Once a surface mesh is generated, the tetrahedral volume mesh of the system, which consists of the molecule and the solvent box can be generated by the program TetGen [35]. The membrane mesh in our former approach was obtained by three steps. In the first step, two planes $z = z_1$ and $z = z_2$ are used to mark the range of the membrane region, and the tetrahedra with all the vertices located between the two planes are marked as belonging to the membrane region. In the second step, tetrahedra which intersect with the two planes are first marked as the ‘interface tetrahedra’ between the membrane region and the solvent region, then the faces of these ‘interface tetrahedra’ are picked up and connected to form the membrane boundary. Finally, a careful topology check is performed on the membrane boundary to ensure its continuity, closeness, etc.

The key point in the process of membrane construction is marking the tetrahedra belonging to the membrane region correctly and getting the triangles on the surface of the membrane-protein region. The difficulty lies in distinguishing the tetrahedra in the membrane region from those in the pore region, since they all locate in the range between $z = z_1$ and $z = z_2$ and these two regions may be connected by some reasonable holes or unreasonable crevices through the channel protein. In our former work [18, 19], we use one or more

spheres or cylinders to separate the membrane and pore regions. However, we need to locate the spheres or cylinders manually depending on the structure of a given ion channel. It is impossible to find a universal sphere suitable for different ion channels. A reasonable method is to detect the inner surface of the pore to distinguish the membrane region and pore region. There have been some methods to detect the surface of the pore of an ion channel for visualization based on the information of the atoms. HOLE [36] is a commonly used program to detect the inner surface of the pore. However, most of these methods are based on the molecular structure instead of a mesh, and provide no mesh information, which is not appropriate in our FE simulations. In this paper we propose an algorithm to detect the inner triangulated surface of the pore to separate the membrane region and the pore region in tetrahedral mesh. Based on the algorithm, we develop a new method to add a membrane for ion channel simulation. In the next section, the method will be described in detail. Two example ion channel systems, gramicidin A and connexin 26 will be shown in the Results section.

2 Method

Before identifying the membrane region, we rotate the protein coordinate system to make the orientation of the channel pore parallel to the z -axis first. This function has been embedded into our visualization tool, VCMM [37]. The corresponding algorithm is described in the following first subsection.

Two planes $z = z_1$ and $z = z_2$ are considered as the lower and upper bounds of the membrane respectively. The membrane construction contains three steps.

- First, the tetrahedra in the pore region and between the two planes are labeled as immutable.
- Second, the tetrahedra with all the vertices located between the two planes $z = z_1$ and $z = z_2$ are marked as the membrane region except the tetrahedra in the protein region and the ones labeled in the first step.
- Third, the surface triangles of the membrane-protein region are collected and marked.

During the process of membrane construction, locating the tetrahedron containing a specified point is a frequently used procedure. Therefore, we first describe the algorithm of finding the tetrahedron containing a specified point. We then detail the method of membrane construction step by step. To facilitate the usage of this method for membrane-protein system construction, we are making these complicate meshing steps available online at www.continuummodel.org.

2.1 Rotate the Channel Pore Parallel to the Z-Axis

The key point in rotating the channel pore to the z -axis is to get the direction of the pore. The following algorithm shows how to compute the direction and rotate the orientation of the channel pore to the z -axis. The algorithm is suitable for the channel protein which is symmetric around the direction of the channel pore.

Firstly, we compute the covariance matrix based on the coordinates of all the atoms of the channel protein. The covariance matrix is shown in Eq. (1).

$$M = E[(X - E(X))(X - E(X))^T], \quad (1)$$

where M is a 3×3 matrix, X is a $3 \times N$ matrix and N is the number of atoms. The i th column of X represents the x, y, z coordinates of the i th atom. $E(X)$ is the expectation of X .

Then the eigenvalues and eigenvectors of M are calculated. Due to the symmetry, M has two similar eigenvalues. The plane spanned by the two eigenvectors corresponding to the two similar eigenvalues is perpendicular to the direction of the channel pore. The remaining eigenvector represents the direction of the channel pore. Finally, a rigid-body rotation is performed on the molecule to align the direction of the channel pore to z -axis.

2.2 Determining the Tetrahedron Containing a Given Query Point

There exist different strategies to find the path from the tetrahedron containing the query point p to the tetrahedron containing a source point q , such as straight walk, orthogonal walk, visibility walk, stochastic walk and so on. All these methods have been described detailedly in [38]. The straight walk consists in traversing all the tetrahedra that are intersected by the line segment qp . As we need the tetrahedra along qp , we choose the straight walk to locate the query point.

The basic predicate in the straight walk is the orientation predicate, which is defined in Eq. (2).

$$\text{orientation}(\alpha, \beta, \gamma, \delta) = \text{sign} \begin{pmatrix} \beta_x - \alpha_x & \gamma_x - \alpha_x & \delta_x - \alpha_x \\ \beta_y - \alpha_y & \gamma_y - \alpha_y & \delta_y - \alpha_y \\ \beta_z - \alpha_z & \gamma_z - \alpha_z & \delta_z - \alpha_z \end{pmatrix}, \quad (2)$$

where $\alpha, \beta, \gamma, \delta$ are four points. $\text{Orientation}(\alpha, \beta, \gamma, \delta)$ determines which side δ lies on or whether δ lies in the plane which the triangle $\alpha\beta\gamma$ belongs to.

The straight walk algorithm contains two steps. First, we find the facet e which qp across through the tetrahedron that q belongs to. Second, we traverse all the tetrahedra that intersect with qp . By testing which side the intersection facet e lies p using orientation test, we determine whether a tetrahedron contains p . We use the algorithm given in [38] and make a few changes in the first step. The source point q which we choose is in the tetrahedron instead of being a vertex of the tetrahedron.

More precisely, the algorithm first performs initialization as follows. Given a source point q and the tetrahedron t where $q \in t$, for each face e in t , we calculate the intersection between qp and e . If there is no intersection between qp and the four facets of tetrahedron t , it indicates that p is also in t . If qp intersects e , then qp goes out t through the face e .

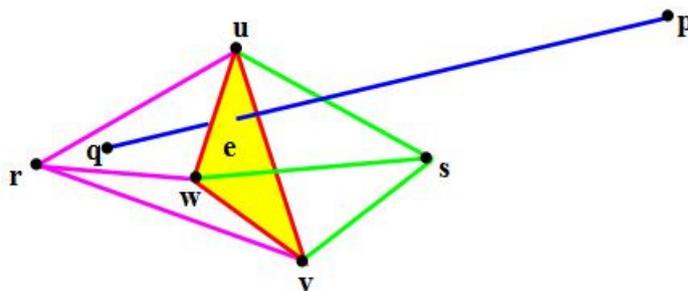


Figure 2: Straight walk in main loop [38].

After the initialization, we start the main steps of the straight walk. At a given step of straight walk, we know that the ray qp goes out of a tetrahedron t by a face e (see Figure 2). By testing on which side of e lies p , we decide whether t contains p . If t contains p , the walk reaches the end. Otherwise, the walk continues in the neighbor of t through e , and qp goes out of that neighbor by a face e' . e' is determined by two orientation tests involving q, p , the new vertex s and a vertex of e . Algorithm 1 gives a pseudo-code for a detailed description of the straight walk method [38].

2.3 Label the Tetrahedra in the Pore Region

The strategy to label the tetrahedra in the pore region and between the two planes $z = z_1$ and $z = z_2$ is walk-and-detect, which is shown in algorithm 2. Before starting the algorithm 2, suppose we already have the volume mesh of the system of an ion-channel protein in a solvent box, and the tetrahedra in solvent region

Algorithm 1: Locate the tetrahedra containing a query point.

Input: source point q , tetrahedra t where $q \in t$ and query point p .

Output: the tetrahedron where p is located in and the list of the tetrahedra, t_{qp} , where qp across through

$m = 0, flag = 0$

while $m < 4$ **do**

v_i, v_j, v_k are the three vertices of the m th facet, e_m

if e_m intersects qp **then**

$flag = 1$

if orientation(v_i, v_j, v_k, p) > 0 **then**

$u = v_i, w = v_j, v = v_k$

else

$u = v_i, v = v_j, w = v_k$

end if

break

end if

$m++$

end while

if $flag = 0$ **then**

p is in t , append t to t_{qp} .

return

end if

// qp intersects triangle uvw ; orientation(w, v, q, p), orientation(v, u, q, p) and

//orientation(u, w, q, p) are positive.

while orientation(u, w, v, p) > 0 **do**

$t = \text{neighbor}(t \text{ through } uvw)$

append t to t_{pq}

$s = \text{vertex of } t, s \neq u, s \neq v, s \neq w$

if orientation(u, s, q, p) > 0 **then**

// qp does not intersect triangle usw

if orientation(v, s, q, p) > 0 **then**

// qp intersects triangle vsw

$u = s$

else

// qp intersects triangle usv

$w = s$

end if

else

// qp does not intersect triangle usv

if orientation(w, s, q, p) > 0 **then**

// qp intersects triangle usw

$v = s$

else

// qp intersects triangle vsw

$u = s$

end if

end if

end while

are marked as 1 and the ones in protein are marked as 2. Here the solvent region includes the region marked 1 and the region marked 3 in Figure 1. In the first step, the boundaries in x, y, z direction of the ion channel are calculated, denoted by $x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}$ and z_{\max} . These boundaries are useful in the case that the pore region has crevices connected the membrane region.

Algorithm 2: Label the tetrahedra in the pore region through a walk-and-detect procedure.

Input: PQR file, volume mesh of an ion-channel protein in a solvent box with different marked regions, walk step size h , membrane location z_1, z_2 .

Output: Tetrahedra in pore region.

- 1: read the PQR file and get coordinates ranges $x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}$ and z_{\max} .
 - 2: find a point p_0 in the pore region and the tetrahedron t_0 where $p \in t_0$, append (p_0, t_0) to the point-tetrahedron pair list, L .
 - 3: get a point p in L and the tetrahedron t_p where $p \in t_p$, compute the six neighbouring points: $(p.x - h, p.y, p.z), (p.x + h, p.y, p.z), (p.x, p.y - h, p.z), (p.x, p.y + h, p.z), (p.x, p.y, p.z - h), (p.x, p.y, p.z + h)$.
 - 4: for each point $q_i, i = 1, \dots, 6$ in step 3:
 - If q_i has been visited:
 - go to the next point q_{i+1} .
 - find the tetrahedron t_{q_i} where $q_i \in t_{q_i}$, and get the tetrahedra list t_{pq_i} where $\overrightarrow{pq_i}$ pass through by straight walk algorithm.
 - 5: If $t_{q_i}.\text{mark} = 1$:
 - If $x_{\min} \leq q_i.x \leq x_{\max}$ and $y_{\min} \leq q_i.y \leq y_{\max}$ and $z_1 \leq q_i.z \leq z_2$ and $q_i \notin L$:
 - append (q_i, t_{q_i}) to L , label the tetrahedra in t_{pq_i} as immutable.
 - else if $t_{q_i}.\text{mark} = 2$:
 - label the tetrahedra in t_{pq_i} as immutable except t_{q_i} .
 - 6: If the six points in step 3 have been tested, mark p to indicate that p has been visited. Go to step 3 until all the points in L have been visited.
-

The walk-and-detect method begins from an initial point p_0 located in the pore region. The tetrahedron where p_0 is located is found by the straight walking method described in section 2.2. The source point for locating p_0 is obtained by the following two stages. First, we construct a small box whose boundary is $[p_0.x - 2h, p_0.x + 2h] \times [p_0.y - 2h, p_0.y + 2h] \times [p_0.z - 2h, p_0.z + 2h]$, where h is the walk step size. Here $p_0.x, p_0.y$ and $p_0.z$ denote the x, y and z coordinate of p_0 , respectively. Second, we find a tetrahedron who has a vertex in the box and then we choose the inner point of the tetrahedron as the source point. The list L in step 2 contains the test points and the tetrahedra containing the test points.

Step 3, 4, 5 and 6 show the process of walk-and-detect. We take a point p in L as example. We walk along six directions of coordinate axes from p with step size h . In each direction, a new test point is obtained and denoted by q_i . Each q_i should be judged whether it has been visited, which means that q_i has been a detection point and the tetrahedra crossed by the line segment pq_i has been labeled. If q_i has been visited, there is no need for p to walk to q_i . Otherwise, the straight walk is used to locate the tetrahedron t_{q_i} which contains q_i and to get the tetrahedra list, t_{pq_i} , along the line segment pq_i . If t_{q_i} is in the solvent region and q_i satisfies the condition in step 5 (see q_1, q_3, q_4 in Figure 3), then q_i should be append to the list of test points. Meanwhile, the tetrahedra in t_{pq_i} are labeled as immutable. Otherwise, if t_{q_i} locates in the protein region (see q_2 in Figure 3), it demonstrates that the test point q_i touches the inner surface of the pore. The face in t_{q_i} which connect the protein region and solvent region is a piece of the inner pore surface. In this situation, we mark the tetrahedra in t_{pq_i} as immutable except t_{q_i} . After detecting the six new points along the six directions, p should be marked to show that it has been visited. Step 3 to step 6 are repeated until the list of test points is empty.

If the step size h is improper, some tetrahedra in the pore region may be missed. The following process is necessary to make sure that all of the tetrahedra in the pore region between $z = z_1$ and $z = z_2$ are marked as

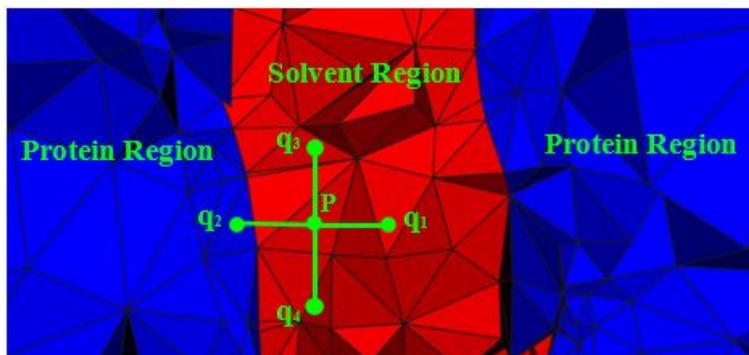


Figure 3: An example of the walk-and-detect algorithm.

immutable. We scan all the tetrahedra in the region $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \times [z_1, z_2]$, and if a tetrahedron satisfies both of the following two conditions, the tetrahedron should be labeled as immutable.

1. The tetrahedron is in the solvent region and has not been labeled.
2. One of the neighboring tetrahedra has been labeled as immutable.

Besides labeling the immutable tetrahedra, this method can also be used to judge whether the ion channel is open or close. We replace $[z_1, z_2]$ in step 5 by $[z_{\min}, z_{\max}]$. If there are two test points that touch $z = z_{\min}$ and $z = z_{\max}$ respectively, then the ion channel is open. Otherwise, the ion channel is close.

2.4 Mark the Membrane Region and Collect Surface Triangles of Membrane-Protein Region

The tetrahedra with all the vertices located between the two planes $z = z_1$ and $z = z_2$ are marked as the membrane region except the ones in the protein region and the ones labeled as immutable.

Algorithm 3: Collect surface triangles of membrane-protein region.

```

for element  $t$  in mesh do
  if  $t$ .mark = solute then
    for the  $i$ th face,  $f_i$  in  $t$  do
       $s = \text{neighbor}(t \text{ through } f_i)$ 
      if  $s$ .mark = solvent then
         $f_i$  is on the membrane-protein surface
      end if
    end for
  end if
end for

```

The strategy to collect the triangular surface mesh of the membrane-protein region is shown in algorithm 3. If a triangle whose two adjacent tetrahedra are in the solvent region and solute region respectively, then it is picked up as one of the membrane-protein surface triangles.

3 Results

In this section, we present the numerical results for two ion channels including gramicidin A (gA) and connexin 26 (Cx26).

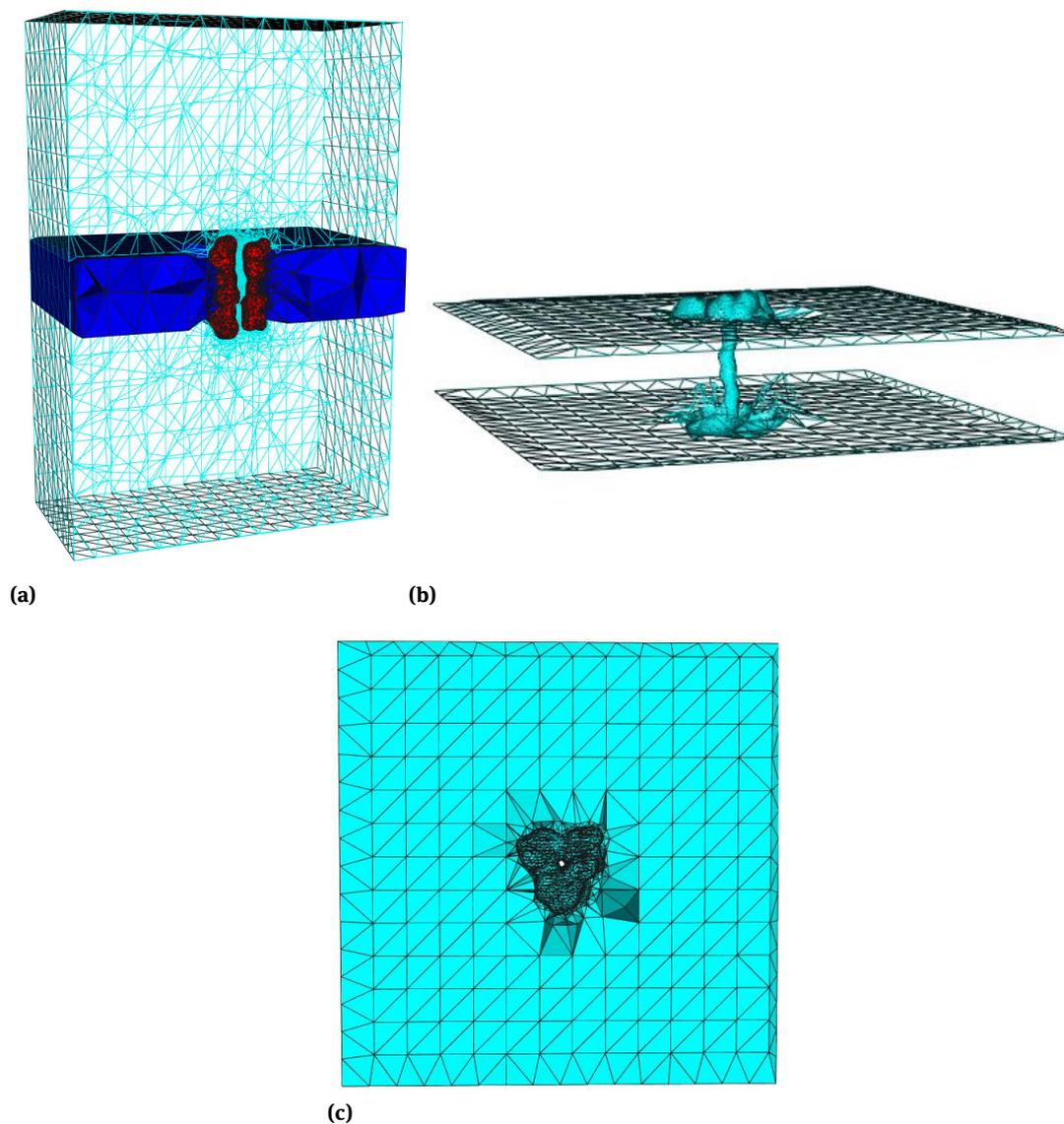


Figure 4: Volume mesh of gA: (a) Wire-frame of volume mesh conforming to the boundary of a channel protein and membrane system, (b) the surface mesh of the membrane-protein region, (c) the upper boundary surface of the membrane-protein region, in which the membrane is represented as a slab.

Gramicidin A (PDB code: 1MAG) is one of the most widely studied ion channels. It forms aqueous pores in lipid bilayers that selectively pass monovalent cations [39, 40]. Gramicidin A is a small 15-amino-acid β helical peptide with a narrow pore. The partial charges and atomic radii for each atom in the protein are the same as we used in our previous simulation [18]. The pore region of gA is along the z direction. The whole domain of the gA channel consists of the membrane-protein region and the solvent region. The triangular surface mesh for gA is generated by TMSmesh and the tetrahedral mesh is generated by TetGen. Figure 4a shows the wire-frame of the volume mesh. The tetrahedra between the two surfaces of the slab except the ones in pore

region forms the membrane-protein region, which is shown in figure 4b. Figure 4c shows the upper boundary surface of the membrane-protein region. To make the slab smoother, we add some additional points on the two planes $z = z_1$ and $z = z_2$ when generating tetrahedral meshes. The slab is almost flat except the part near the ion channel and the part close to the outer box. If the points we add on the two planes are close to the ion channel or the box, it will be hard for TetGen to generate the tetrahedral mesh. Therefore, we do not add points in the region close to the ion channel and the region near the outer box. In our experience, a small bumpiness of the membrane has no influence to the simulation.

Another example is Cx26, which is a member of the connexin family. It forms a typical four-helix bundle in which any pair of adjacent helices is antiparallel. The pore has an inner diameter of 35 Å at the cytoplasmic entrance, and the smallest diameter of the pore is 14 Å. The initial coordinates for Cx26 hemichannel are obtained from the PDB (code GJB2) [41]. The partial charges and atomic radii are obtained by PDB2PQR software [42]. The surface mesh is also generated by TMSmesh and the tetrahedral mesh is generated by TetGen. Figure 5 describes the volume mesh especially the membrane-protein region of Cx26. Our method correctly identifies the membrane-protein surface which validates the effectiveness of the method.

The quality of the volume mesh generated by our method is analyzed. Two criteria to measure the mesh quality have been considered, including dihedral angle and radius ratio [43]. The dihedral angle θ between two faces A and B in a tetrahedron is defined as follows:

$$\theta = \pi - \arccos\left(\frac{\mathbf{n}_A \cdot \mathbf{n}_B}{|\mathbf{n}_A| \cdot |\mathbf{n}_B|}\right) \quad (3)$$

where \mathbf{n}_A and \mathbf{n}_B are homodromous normal vectors of the two faces. The dihedral angles of a regular tetrahedron are around 70° . The radius ratio, denoted by ρ , is defined to be the ratio of the inscribed sphere radius over the circumscribed sphere radius, scaled by 3. $0 < \rho \leq 1$ holds for any tetrahedron and $\rho = 1$ if and only if the tetrahedron is regular. Figure 6 shows the quality of the volume meshes generated by our method. The dihedral angles are clustered around 40° to 140° and the radius ratio are clustered around 0.75, which shows that most tetrahedra in the meshes have good quality. The quality of the surface mesh generated by TMSmesh has been analyzed detailedly in Refs. [22, 23, 44]. The quality of the triangular mesh generated by TMSmesh is comparable to that generated by the commonly used software, MSMS. However, the meshes generated by both software still contain some triangles with tiny angles or short edges. It is worth noting that most of the poor quality tetrahedra of the volume mesh are caused by the irregular surface and surface mesh with aforementioned poor quality triangles. Despite the existence of a small portion of poor quality elements, those meshes generated by our method are applicable to numerical simulation as shown in our previous finite element/boundary element modeling of biomolecules [18, 19, 22].

4 Conclusion

We have presented a new method to embed a slab membrane in a tetrahedra mesh of ion channels. Firstly, we walked through the pore region and labeled the tetrahedra in the pore region as immutable. The inner surface of the pore region can also be detected during the walking process. By the walk-and-detect algorithm, we can also decide whether the ion channel is open or close. Then we marked the tetrahedra between the two planes $z = z_1$ and $z = z_2$ as the membrane region, except the ones in the solute region and the ones labeled in the first step. Finally, the triangles were picked up to form the surface of the membrane-protein region. The method is applied to two ion channels, gA and Cx26. The membranes embedded are reasonable and almost flat and the pore regions are correctly located. In addition, the new method is more general since it is robust and independent to the structure of the ion channel. A convenient on-line version for the meshing approach will be available at www.continuummodel.org.

Acknowledgement: Tiantian Liu, Shiyang Bai and Benzhuo Lu are supported by the China NSF (91230106) and the CAS Program for Cross & Cooperative Team of the Science & Technology Innovation. Minxin Chen is supported by China NSF (NSFC11301368) and the NSF of Jiangsu Province (BK20130278).

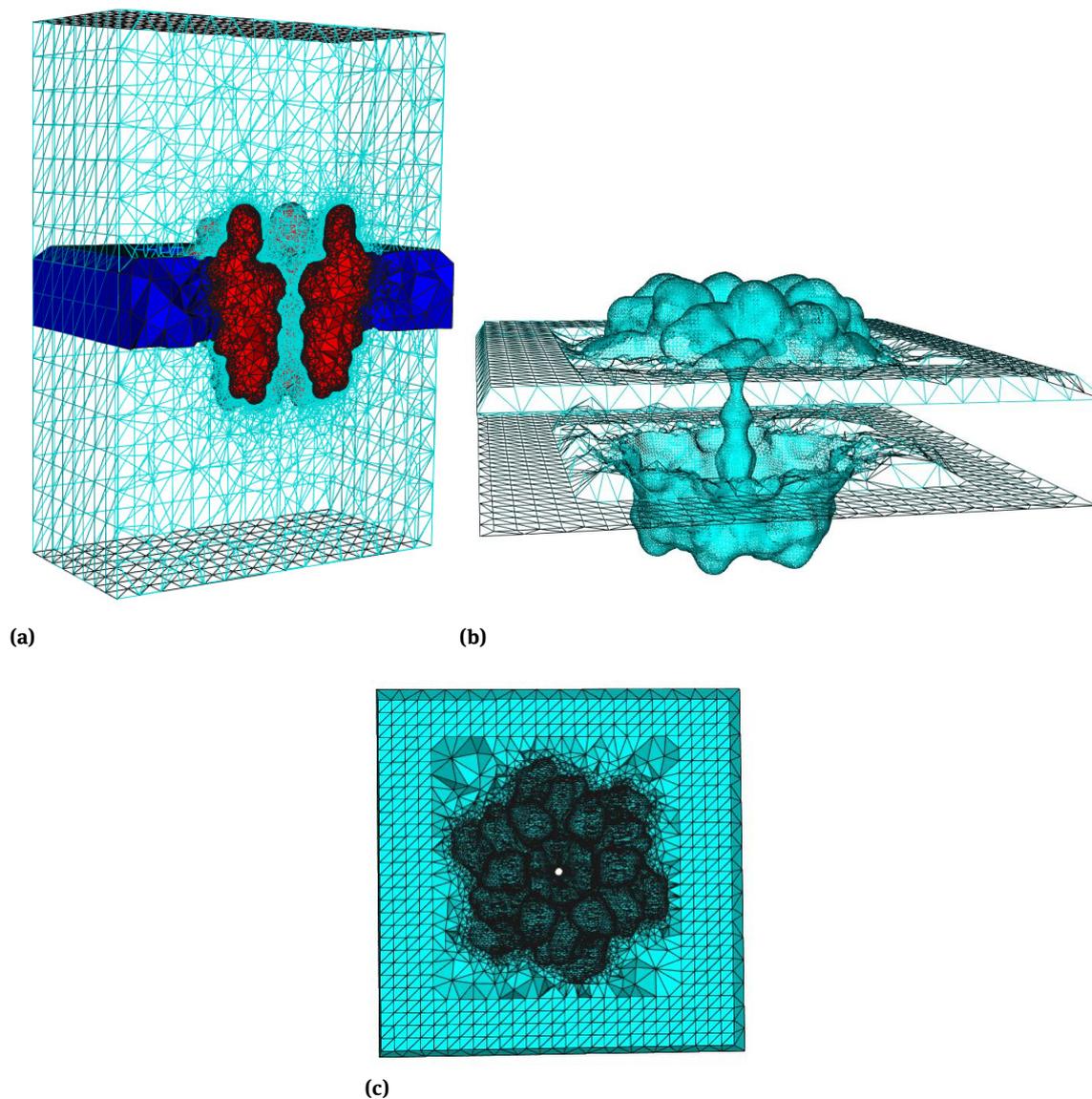


Figure 5: Volume mesh of Cx26: (a) Wire-frame of volume mesh conforming to the boundary of a channel protein and membrane system, (b) the surface mesh of the membrane-protein region, (c) the upper boundary surface of the membrane-protein region, in which the membrane is represented as a slab.

Conflict of interest: Author state no conflict of interest.

References

- [1] D. Marx and J. Hutter. *Modern Methods and Algorithms of Quantum Chemistry*. John von Neumann Institute for Computing, Jülich, 2000.
- [2] J. Ostmeyer, S. Chakrapani, A. C. Pan, E. Perozo, and B. Roux. Recovery from slow inactivation in K^+ channels is controlled by water molecules. *Nature*, 501(7465):121–124, 2013.
- [3] M. Jensen, V. Jogini, D. W. Borhani, A. E. Leffler, R. O. Dror, and D. E. Shaw. Mechanism of voltage gating in potassium channels. *Science*, 336(6078):229–233, 2012.
- [4] S. Li, M. Hoyles, S. Kuyucak, and S. Chung. Brownian dynamics study of ion transport in the vestibule of membrane channels. *Biophys. J.*, 74:37–47, 1998.

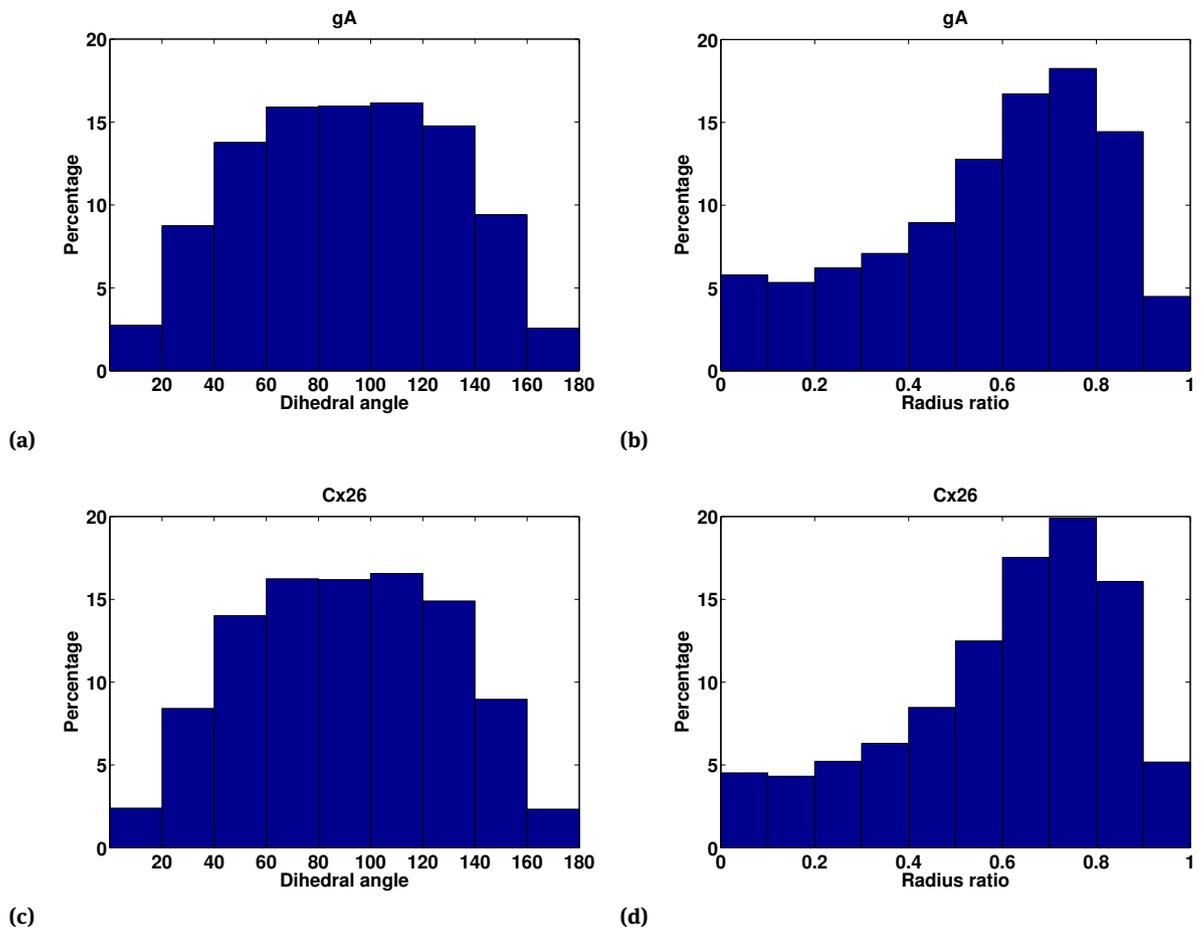


Figure 6: The quality of the volume mesh generated by our method. The normalized histograms of dihedral angles and radius ratios of the meshes for (a-b) gA and (c-d) Cx26.

- [5] B. Corry, S. Kuyucak, and S. H. Chung. Tests of continuum theories as models of ion channels. II. Poisson-Nernst-Planck theory versus brownian dynamics. *Biophys. J.*, 78:2364–2381, 2000.
- [6] S. Kuyucak, O. S. Andersen, and S. H. Chung. Models of permeation in ion channels. *Rep. Prog. Phys.*, 64:1427–1472, 2001.
- [7] G. A. Huber and J. A. McCammon. BrownDye: A software package for brownian dynamics. *Comp. Phys. Comm.*, 181:1896–1905, 2010.
- [8] B. Z. Lu and Y. C. Zhou. Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes II: Size effects on ionic distributions and diffusion-reaction rates. *Biophys. J.*, 100(10):2475 – 2485, 2011.
- [9] B. Z. Lu. Poisson-Nernst-Planck equations. *Encyclopedia of Applied and Computational Mathematics*; Engquist, B., Ed.; Springer-Verlag: Berlin Heidelberg, 2013.
- [10] B. Roux, T. Allen, S. Berneche, and W. Im. Theoretical and computational models of biological ionchannels. *Q. Rev. Biophys.*, 7(1):1–103, 2004.
- [11] R. S. Eisenberg. Ionic channels in biological membranes: natural nanotubes. *Acc. Chem. Res.*, 31:117–123, 1998.
- [12] D. Chen, J. Lear, and R. S. Eisenberg. Permeation through an open channel: Poisson-Nernst-Planck theory of a synthetic ionic channel. *Biophys. J.*, 72:97–116, 1997.
- [13] M. G. Kurnikova, R. D. Coalson, P. Graf, and A. Nitzan. A lattice relaxation algorithm for three-dimensional Poisson-Nernst-Planck theory with application to ion transport through the gramicidin a channel. *Biophys. J.*, 76:642–656, 1999.
- [14] Q. Zheng, D. Chen, and G. W. Wei. Second-order Poisson-Nernst-Planck solver for ion transport. *J. Comput. Phys.*, 230(13):5239–5262, 2011.
- [15] B. Z. Lu, M. J. Holst, J. A. McCammon, and Y. C. Zhou. Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes I: Finite element solutions. *J. Comput. Phys.*, 229(19):6979–6994, 2010.
- [16] B. Z. Lu and J. A. McCammon. Molecular surface-free continuum model for electrodiffusion processes. *Chem. Phys. Lett.*, 451(4-6):282–286, 2008.

- [17] B. Z. Lu, Y. C. Zhou, Gary A. Huber, S. D. Bond, Michael J. Holst, and J. A. McCammon. Electrodiffusion: A continuum modeling framework for biomolecular systems with realistic spatiotemporal resolution. *J. Chem. Phys.*, 127(13):135102, 2007.
- [18] B. Tu, M. X. Chen, Y. Xie, L. B. Zhang, B. Eisenberg, and B.Z. Lu. A parallel finite element simulator for ion transport through three-dimensional ion channel systems. *J. Comput. Chem.*, 34:2065–2078, 2013.
- [19] B. Tu, S. Y. Bai, M. X. Chen, Y. Xie, L. B. Zhang, and B. Z. Lu. A software platform for continuum modeling of ion channels based on unstructured mesh. *Computational Science & Discovery*, 7:014002, 2014.
- [20] U. Hollerbach, D. P. Chen, and R. S. Eisenberg. Two- and three-dimensional Poisson-Nernst-Planck simulations of current flow through gramicidin a. *J. Sci. Comput.*, 16(4):373–409, 2002.
- [21] S. R. Mathur and J. Y. Murthy. A multigrid method for the Poisson-Nernst-Planck equations. *SIAM J. Appl. Math.*, 52(17–18):4031–4039, 2009.
- [22] M. X. Chen and B. Z. Lu. TMSmesh: A robust method for molecular surface mesh generation using a trace technique. *J. Chem. Theory Comput.*, 7:203–212, 2011.
- [23] M. X. Chen, B. Tu, and B. Z. Lu. Triangulated manifold meshing method preserving molecular surface topology. *Journal of Molecular Graphics and Modelling*, 38:411–418, 2012.
- [24] M. Sanner, A. Olson, and J. Spehner. Reduced surface: An efficient way to compute molecular surface properties. *Biopolymers*, 38:305–320, 1996.
- [25] S. Decherchi and W. Rocchia. A general and robust ray casting based algorithm for triangulating surfaces at the nanoscale. *PLoS One*, 8(4):e59744, 2013.
- [26] C. Bajaj, G. Xu, and Q. Zhang. Bio-molecule surfaces construction via a higher-order level-set method. *Journal of Computer Science and Technology*, 23(6):1026–1036, 2008.
- [27] W. Zhao, G. Xu, and C. Bajaj. An algebraic spline model of molecular surfaces for energetic computations. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(6):1458–1467, 2011.
- [28] Z.Y. Yu, M. J. Holst, and J. McCammon. High-fidelity geometric modeling for biomedical applications. *Finite Elem. Anal. Des.*, 44(11):715–723, 2008.
- [29] D. Xu and Y. Zhang. Generating triangulated macromolecular surfaces by Euclidean Distance Transform. *PLoS ONE*, 4(12):e8140, 2009.
- [30] B. Lee and F.M. Richards. The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol.*, 55(3):379–400, 1971.
- [31] F. M. Richards. Areas, volumes, packing and protein structure. *Annual Review in Biophysics and Bioengineering*, 6:151–176, 1977.
- [32] Y.J. Zhang, G.L. Xu, and C. Bajaj. Quality meshing of implicit solvation models of biomolecular structures. *Comput. Aided Geom. Des.*, 23(6):510–530, 2006.
- [33] H. Edelsbrunner. Deformable smooth surface design. *Discrete and Computational Geometry*, 21:87–115, 1999.
- [34] A. Zaharescu, E. Boyer, and R. P. Horaud. Transformesh: a topology-adaptive mesh-based approach to surface evolution. *In Proceedings of the Eighth Asian Conference on Computer Vision*, 11:166–175, November 2007.
- [35] H. Si. Tetgen, a Delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software*, 41(2).
- [36] Oliver S. Smart, Joseph G. Neduelil, Xiaonan Wang, B.A. Wallace, and Mark S.P. Sansom. Hole: A program for the analysis of the pore dimensions of ion channel structural models. *Journal of Molecular Graphics*, 14(6):354 – 360, 1996.
- [37] S.Y. Bai and B.Z. Lu. VCMM: A visual tool for continuum molecular modeling. *Journal of Molecular Graphics and Modelling*, 55:44–49, 2014.
- [38] O. Devillers, S. Pion, and M. Teillaud. Walking in a triangulation. *International Journal of Foundations of Computer Science*, 13(02):181–199, 2002.
- [39] O. S. Andersen. Gramicidin channels. *Annu. Rev. Physiol.*, 46:531–548, 1984.
- [40] R. E. Koeppe and O. S. Andersen. Engineering the gramicidin channel. *Annu. Rev. Biophys. Biomol. Struct.*, 25:231–258, 1996.
- [41] S. Maeda, S. Nakagawa, M. Suga, E. Yamashita, A. Oshima, Y. Fujiyoshi, and T. Tsukihara. Structure of the connexin 26 gap junction channel at 3.5 Å resolution. *Nature*, 458(7238):579–602, 2009.
- [42] T. J. Dolinsky, J. E. Nielsen, J. A. McCammon, and N. A. Baker. PDB2PQR: an automated pipeline for the setup, execution, and analysis of poisson-boltzmann electrostatics calculations. *Nucleic Acids Res.*, 32:W665–W667, 2004.
- [43] A. Liu and B. Joe. Relationship between tetrahedron shape measures. *BIT Numerical Mathematics*, 34(2):268–287, 1994.
- [44] 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), 2015.