1-15-2014

Multiscale Geometric Modeling of Macromolecules I: Cartesian Representation

Kelin Xia
Michigan State University

Xin Feng
Michigan State University

Zhan Chen
Georgia Southern University, zchen@georgiasouthern.edu

Yiying Tong
Michigan State University

Guo-Wei Wei
Michigan State University

Published in final edited form as:
J Comput Phys. 2014 January ; 257(Pt A): . doi:10.1016/j.jcp.2013.09.034.

Multiscale geometric modeling of macromolecules I: Cartesian representation

Kelin Xia[1], **Xin Feng**[2], **Zhan Chen**[1], **Yiying Tong**[2,*], and **Guo Wei Wei**[1,3,†]

[1]Department of Mathematics, Michigan State University, MI 48824, USA

[2]Department of Computer Science and Engineering, Michigan State University, MI 48824, USA

[3]Department of Biochemistry and Molecular Biology, Michigan State University, MI 48824, USA

Abstract

This paper focuses on the geometric modeling and computational algorithm development of biomolecular structures from two data sources: Protein Data Bank (PDB) and Electron Microscopy Data Bank (EMDB) in the Eulerian (or Cartesian) representation. Molecular surface (MS) contains non-smooth geometric singularities, such as cusps, tips and self-intersecting facets, which often lead to computational instabilities in molecular simulations, and violate the physical principle of surface free energy minimization. Variational multiscale surface definitions are proposed based on geometric flows and solvation analysis of biomolecular systems. Our approach leads to geometric and potential driven Laplace-Beltrami flows for biomolecular surface evolution and formation. The resulting surfaces are free of geometric singularities and minimize the total free energy of the biomolecular system. High order partial differential equation (PDE)-based nonlinear filters are employed for EMDB data processing. We show the efficacy of this approach in feature-preserving noise reduction. After the construction of protein multiresolution surfaces, we explore the analysis and characterization of surface morphology by using a variety of curvature definitions. Apart from the classical Gaussian curvature and mean curvature, maximum curvature, minimum curvature, shape index, and curvedness are also applied to macromolecular surface analysis for the first time. Our curvature analysis is uniquely coupled to the analysis of electrostatic surface potential, which is a by-product of our variational multiscale solvation models. As an expository investigation, we particularly emphasize the numerical algorithms and computational protocols for practical applications of the above multiscale geometric models. Such information may otherwise be scattered over the vast literature on this topic. Based on the curvature and electrostatic analysis from our multiresolution surfaces, we introduce a new concept, the polarized curvature, for the prediction of protein binding sites.

Keywords

Protein characterization; Variational multiscale surfaces; Curvature analysis; High order geometric PDEs; Free energy functional; EMDataBank; Protein data bank

1 Introduction

Structural biology is an essential part of modern biological sciences. A basic role of structural biology is to provide structural information of biological macromolecules, especially proteins and nucleic acids, and the interpretation of macromolecular structures, namely, structure-function correlations. Typically, macromolecular structures are

*Corresponding author. ytong@msu.edu. †Corresponding author. wei@math.msu.edu.

determined via macromolecular crystallography, NMR, EPR, etc. Macromolecular structural data are deposited at the Protein Data Bank (PDB), which is a source for much biophysical modeling, simulation and analysis. One most important new trend in structural biology is the study of large protein complexes and subcellular organelles, which plays an essential role in many key biological processes, including genome replication, transcription, translation, protein-folding, signal transduction, and viral infection. The structural information of large protein complexes and subcellular organelles is crucial for exploring the molecular mechanisms behind complex biological processes. Unfortunately, most conventional experimental means and imaging modalities well-suited for relatively small proteins do not work well for multiprotein complexes and subcellular organelles. Recently, electron tomography, especially cryo-electron microscopy (cryo-EM), has become a powerful tool for revealing three-dimensional (3D) structures of macromolecular complexes in different functional or biological states. The feasible resolution of cryo-EMs ranges from 80 to 2Å, capable of bridging the gap between live-cell imaging and atomic resolution structures. Structures determined by cryo-EMs are deposited at the EMDataBank (EMDB), a significant resource for global deposition and retrieval of cryo-EM data. Unlike PDB, which usually contains information about structures of proteins, nucleic acids, and complex assemblies obtained from X-ray crystallography or NMR spectroscopy at the atomic level resolution, EMDB typically provides information about multiproteins, organelles, cell and tissue from cryo-EMs at the molecular level resolution.

It remains a great challenge to quantitatively model and predict the structure, function, dynamics and transport of complex self-organizing biological systems. Geometric modeling not only bridges the gap between biomolecular data and biological conceptualization and interpretation, but also provides a basis for mathematical modeling, analysis and computation [78, 22]. In 1953, Corey and Pauling proposed the atom and bond model of molecules [18], which has since become a cornerstone in physical science. Numerous other models, including the van der Waals surface, the solvent-excluded surface (SES) (also known as molecular surface (MS)), and the solvent-accessible surface have been proposed [36, 46]. The combination of these biomolecular surfaces with the calculated electrostatic potentials on and around them, has become an important procedure in the analysis of biomolecular structure, function, and interaction, such as ligand-receptor binding, protein specification, drug design, macromolecular assembly, protein-nucleic acid and protein-protein interactions, and enzymatic mechanism [47]. A variety of physical and graphical models are developed during the past few decades.

The widely applied biomolecular surfaces, especially SESs, have well-known drawbacks in their definitions. One of these problems is the admission of geometric singularities, i.e., tips, cusps and self-intersecting facets, which lead to computational instabilities and induce numerical errors [17, 20, 32, 50]. Another defect is that these surfaces are simply *ad hoc* divisions of a biomolecule from its surroundings, without the consideration of the physical laws of surface energy minimization and surface evolution under the interaction with the aqueous environment. At the fundamental level, there is no sharp division between solvent and solute because their electron densities overlap with each other. In the past few years, many theoretical models have been proposed to address these problems [67, 5, 6, 7, 4, 82].

In the past two decades, geometric flows, particularly the mean curvature flows, have received much attention. Geometric flows have had much impact in image processing and data analysis, particularly for feature-preserving noise reduction [40, 51, 55]. Historically, Witkin pioneered diffusion equation based image denoising in 1983 [71]. In 1990, Perona and Malik proposed an anisotropic diffusion equation [43], in which the diffusion coefficient is controlled by image gradients. The Perona and Malik equation can remove the noise without blurring the image edges. Osher and Sethian invented the level set method, which

was applied far beyond the scope of image processing, to computer graphics, computational geometry, optimization, and computational fluid dynamics [51, 41]. Other related mathematical techniques include Mumford-Shah variational functional [39], total variation models designed by Rudin, Osher, and Fatemi [49], and Willmore flow formulation [69, 53, 9, 19]. Because high order partial differential equation (PDE) can more efficiently suppress the noisy component, Wei introduced the first family of arbitrarily high order nonlinear PDEs for noisy image restoration in 1999 [63]. Most geometric PDEs are designed as low-pass filters. The first nonlinear PDE based high-pass filter was proposed in 2002 [66]. Recently, PDE transform has been introduced for functional mode decomposition based on the iterative applications of Wei's high order nonlinear PDE filters [62, 61].

To overcome geometric singularity in classical macromolecular surfaces, Wei and his co-workers introduced one of the first geometric flow approaches for molecular surface generation in 2005 [67]. Bates, Wei and Zhao incorporated the energy minimization principle into macromolecular surface generation, and proposed one of the first variational frameworks for biomolecular surfaces [5, 6]. Basically, a free energy functional of the biomolecular surface model is defined. Through the Euler-Lagrange equation of surface free energy minimization, a generalized mean curvature flow equation is attained. The resulting molecular surface, called the minimal molecular surface (MMS), is then constructed by the mean curvature flow [7]. PDE algorithms for biomolecular surface generation have become a popular topic in theoretical biology [75, 79, 4]. Both aforementioned arbitrarily high-order geometric PDEs and PDE transform have been applied to biomolecular surface construction [4, 82]. Similar approaches were employed by Cheng et al. [15] to extract biomolecular surfaces from a variational solvation model.

In a physiological environment, up to 65%–90% of human cellular mass is water. Consequently, almost all the biological processes in cell, such as signal transduction, transcription, translation, protein folding, protein ligand binding, and charge and mass transport, occur in aqueous surroundings. Therefore, the understanding of the solvation is of fundamental importance for quantitative modeling and analysis of all the above-mentioned processes. Explicit solvent models and implicit solvent models are two major approaches for solvation analysis [48, 52, 54]. For explicit solvent models, both the solvent and the solute are described in atomic detail and extensive sampling is required. Implicit solvent models are designed to reduce the number of degrees of freedom by using a dielectric continuum to describe the solvent while admitting a microscopic atomic description for the biomolecules [74, 3, 8]. Due to their fewer degree of freedom, implicit solvent models, such as the Poisson-Boltzmann (PB) model or the Poisson equation (PE) model when there is no salt in the solvent, are widely used [1, 2, 45]. The coupling of the PB or PE with the generalized Laplace-Beltrami flow has the potential of describing the formation of molecular surface in realistic solvation environments. Conceptually, a solvation free energy can be divided into two major parts: a nonpolar part associated with inserting an uncharged solute into the solvent [14] and a polar part associated with charging the solute in vacuum and solvent [38, 11]. The nonpolar free energy and polar free energy can be represented by a total free energy functional [63]. By using the variational principle, a new geometric flow equation is generated that controls the biomolecular surface formation and evolution via curvature and potential driven [4, 11, 12, 13]. This model takes into consideration of the surface energy minimization and also the solvent-solute interaction, and gives a multiresolution representation of biomolecular surfaces in their native environment. Additionally, the external potential term can be used to incorporate different kinds of effects, such as chemical reaction, fluid flow, and elastic description of marcomolecules [68, 65].

The objective of the present paper is to provide an expository investigation and summary of tools, algorithms and methodologies for geometric modeling of biomolecules. We

particularly focus on tools, algorithms and methodologies required for biophysical models in the Eulerian representation. Although Eulerian formation [11] and Lagrangian formation [12] of biomolecular surfaces can be formally equivalent, they depend on different tools, algorithms and methodologies. The starting points of our discussions are experimental data from either the PDB or the EMDB. For the latter, the high order PDEs are introduced to perform noise reduction. Geometric features, such as Gaussian curvatures, mean curvatures, and shape index, are employed to describe the geometric properties of biomolecular multiresolution surfaces generated by generalized geometric flows and from the EMDB for the first time.

The rest of this paper is organized as follows. Section 2 is devoted to computational algorithms. We discuss in great detail data sources, related softwares, and computational techniques for surface construction, quality improvement, and geometric characterization. We provide advanced interface methods for the evaluation of surface area and surface enclosed volume in the Cartesian representation. Efficient algorithms for calculating various curvature properties, such as Gaussian curvature, mean curvature, maximum and minimum principal curvatures, shape index, and curvedness are developed. The performance of these algorithms is compared. Mathematical models are presented in Appendix A. Two variational models for macromolecular surface generation from PDB data are introduced. We make use of the differential geometry theory of surfaces and geometric measure theory to formulate protein surfaces in conjugation with electrostatic analysis. Additionally, geometric flow based methods are utilized to render high quality macromolecular surfaces from EMDB data. Finally, methods for characterizing geometric features of macromolecules are also proposed. This paper ends with concluding remarks.

2 Computational algorithms

2.1 PDB data processing and surface generation

The PDB is a repository for the 3D structural data of macromolecules, usually obtained by X-ray crystallography or NMR spectroscopy. Most data downloaded from the PDB need to be processed for preparing structures used in theoretical analysis and modeling [10]. Visualization is of great importance to our understanding and conceptualization of the biomolecular systems. Many softwares can be employed to generate triangular surface meshes for biomolecules. An example is the MSMS package. However, the MSMS surface cannot be directly used in Cartesian domain modeling and computation as discussed below.

2.1.1 Lagrangian to Eulerian transformation—The molecular surface generated from the MSMS software is in the Lagrangian representation, i.e., triangle meshes are used to describe the surface. In order to generate the Cartesian representation for finite difference type of methods, one needs to carry out the transformation from Lagrangian to Eulerian representation, i.e., to immerse the 2D surface obtained from the Lagrangian representation into a bounded 3D domain with the Cartesian grid. In this process, one needs to extract interface information from the triangle mesh representation, including the coordinates of intersecting points between the surface and Cartesian mesh lines, and surface normal directions at these intersecting points.

For example, if we have a surface mesh in .vert and .face files, usually, the .vert file stores the point coordinates in the form of $\mathbf{v} = (\upsilon_x, \upsilon_y, \upsilon_z)$, and the .face file contains the connectivity information with each triangle represented by three vertex indices. The bounded box to encompass the protein can be constructed by expanding the tightest axis-aligned bounding box, i.e., by decreasing (increasing) the minimal (maximal) values of surface coordinates, by a certain value denoted as d_c. The new Cartesian mesh domain is thus $[x_l, x_r] \times [y_l, y_r] \times [z_l, z_r]$, and can be obtained from,

$$x_l = \min_{m=1,\ldots,N_t}(\upsilon_{m,x}) - d_c, \quad (1)$$

$$y_l = \min_{m=1,\ldots,N_t}(\upsilon_{m,y}) - d_c, \quad (2)$$

$$z_l = \min_{m=1,\ldots,N_t}(\upsilon_{m,z}) - d_c, \quad (3)$$

$$x_r = \max_{m=1,\ldots,N_t}(\upsilon_{m,x}) + d_c, \quad (4)$$

$$y_r = \max_{m=1,\ldots,N_t}(\upsilon_{m,y}) + d_c, \quad (5)$$

$$z_r = \max_{m=1,\ldots,N_t}(\upsilon_{m,z}) + d_c, \quad (6)$$

where N_t is the total number of the node points in the Lagrangian representation of the protein surface. One can specify the mesh spacing, i.e., the size of each grid, as h, and coordinates of Cartesian mesh nodes can be calculated and represented as $\{(x_i, y_j, z_k)|i = 1, \ldots, N_x; j = 1, \ldots, N_y; k = 1, \ldots, N_z\}$, with N_x, N_y and N_z standing for the total node numbers in each dimension. It can be seen that $x_l = x_1$ and $x_r = x_{N_x}$. Similar relations exist for y and z coordinates.

As the goal is to find the intersection points of each triangle with grid lines, we first find the plane equation. For each mesh triangle, one has the coordinates for its three vertices ($\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$). The 2D plane that the triangle belongs to is

$$ax + by + cz + d = \begin{vmatrix} x - \upsilon_{1,x} & y - \upsilon_{1,y} & z - \upsilon_{1,z} \\ \upsilon_{2,x} - \upsilon_{1,x} & \upsilon_{2,y} - \upsilon_{1,y} & \upsilon_{2,z} - \upsilon_{1,z} \\ \upsilon_{3,x} - \upsilon_{1,x} & \upsilon_{3,y} - \upsilon_{1,y} & \upsilon_{3,z} - \upsilon_{1,z} \end{vmatrix} = 0. \quad (7)$$

The norm for the triangle is the same for the plane, represented as

$$\mathbf{n} = \left(\frac{a}{\sqrt{a^2+b^2+c^2}}, \frac{b}{\sqrt{a^2+b^2+c^2}}, \frac{c}{\sqrt{a^2+b^2+c^2}} \right). \quad (8)$$

We find the intersection points by testing grid edges within the bounding box of the triangle. It is easy to find the coordinate ranges for all the relevant grid edges, e.g. in x coordinate,

$$x_s = \min(\upsilon_{1,x}, \upsilon_{2,x}, \upsilon_{3,x}), \quad (9)$$

$$x_b = \max(\upsilon_{1,x}, \upsilon_{2,x}, \upsilon_{3,x}). \quad (10)$$

For all the points within the triangle, the values of x coordinate should fall in the range $[x_s, x_b]$. For any Cartesian grid line with x-coordinate x_i in $\{x_i|x_s \le x_i \le x_b\}$, the index i satisfies the restriction $i_0 \le i \le i_1$, where $i_0 = [x_s/h]$ and $i_1 = [x_b/h]$ with h being the grid spacing. Similarly, one can find similar lower and upper limit integers for the other two coordinates, $j_0 \le j \le j_1$, $k_0 \le k \le k_1$. Thus, to find an intersection between a surface triangle and a grid line

along x direction, one can choose two arbitrary index (j, k) within the corresponded ranges, with their associated coordinates defined as (y_j, z_k), and calculate the related point in the triangle plane. The related coordinates are denoted as (x_o, y_j, z_k), and evaluated from

$$ax_o+by_j+cz_k+d=0. \quad (11)$$

The intersecting points form a set, which is the collection of only three possible types of points:

$$\{(x_o, y_j, z_k)|j_0 \le j \le j_1; k_0 \le i \le k_1; ax_o+by_j+cz_k+d=0\}, \quad (12)$$

$$\{(x_i, y_o, z_k)|i_0 \le i \le i_1; k_0 \le k \le k_1; ax_i+by_o+cz_k+d=0\}, \quad (13)$$

$$\{(x_i, y_j, z_o)|i_0 \le i \le i_1; j_0 \le j \le j_1; ax_i+by_j+cz_o+d=0\}. \quad (14)$$

The only task left to do is to determine whether the planar point we calculate falls within the triangle. The points located outside the triangle are discarded. If the point located on the boundary edges or the interior of the triangle, it is indeed a point where the interface intersects with the Cartesian grid lines. The normal vector for this interface intersecting point is defined to be the same as that of the triangle, or for efficiency, it can be computed as the linear interpolations of vertex normals. The normals and coordinates are then stored in the sequence of their related Cartesian nodes.

We test our method on a sphere with radius $r = 2$. Using the MSMS software, we generate the Lagrangian representation of the surface with 100 vertices for each 1×1 area. The Cartesian representation is set with a mesh spacing h and the interface-mesh intersecting points are calculated and the average error is evaluated by

$$\text{Error}=\sum \frac{|r - \sqrt{x_o^2+y_o^2+z_o^2}|}{N_o}, \quad (15)$$

where (x_o, y_o, z_o) are the calculated interface-mesh intersecting points and N_o are the total number of such intersecting points. The errors of Lagrangian to Eulerian transformation are illustrated in Table 1. It can be seen from the table that second order accuracy is attained.

2.1.2 Surface generation in the Cartesian representation—The basic idea for surface generation is to embed an enlarged van der Waals surface in a 3D domain and evolve this hypersurface under a geometric and potential driven flow under certain biological constraint. Note that directly evolving the geometric flow equation in the Lagrangian representation for a protein may be unstable due to the possible topological changes during the surface evolution. In the Cartesian setting, some basic information of the protein is needed, including atom positions $\mathbf{x}_i$, $i = 1, \cdots, n$, atom radii r_i, $i = 1, \cdots, n$ and also the atomic charges information for electrostatic analysis when a full solvation model is used; see description in Appendix A. Here n is the total number of the atoms in the protein molecule. To set up the initial conditions, two domains are defined, one is D_χ representing the domain enclosed by the van der Waals surface; the other is an enlarged domain D:

$$D_\chi=\cup_{i=1}^{n}\{\mathbf{x}:|\mathbf{x}-\mathbf{x}_i|<r_i\}; \quad (16)$$

$$D=\cup_{i=1}^{n}\{\mathbf{x}:|\mathbf{x}-\mathbf{x}_i|<1.3r_i\}. \quad (17)$$

Here we choose a factor of 1.3 to guarantee the formation of properly connected surfaces. In fact, this special parameter can be adjusted to give different scales of the molecular surface, which may lead to dramatically different geometric features [22]. We denote S as the Cartesian representation of the hypersurface. For the initial value of S, we consider two functions,

$$S(x,y,z,0)=\begin{cases}1,(x,y,z)\in D\\0,\text{otherwise.}\end{cases} \quad (18)$$

$$\chi(x,y,z)=\begin{cases}1,(x,y,z)\in D_\chi\\0,\text{otherwise.}\end{cases} \quad (19)$$

The characteristic function $\chi(x, y, z)$ is used to protect the van der Waals surface during the surface evolution. The Dirichlet boundary condition is used in the computation, the boundary value is $S = 0$. The formation of surface is driven only by the generalized Laplace-Beltrami equation (83) in Appendix A. We spell out all the terms involved,

$$\frac{\partial S}{\partial t}=\gamma\left[\frac{(S_x^2+S_y^2)S_{zz}+(S_x^2+S_z^2)S_{yy}+(S_y^2+S_z^2)S_{xx}}{S_x^2+S_y^2+S_z^2}-\frac{2S_xS_yS_{xy}+2S_xS_zS_{xz}+2S_zS_yS_{yz}}{S_x^2+S_y^2+S_z^2}+\frac{\sqrt{S_x^2+S_y^2+S_x^2}}{\gamma}V_1\right], \quad (20)$$

where γ is the surface tension and V_1 is a general potential driven term due to other effects. We treat protein surface tension as a fitting parameter in the free energy calculation of a set of molecules [11, 12]. To take into consideration the biological constraints, we modify the evolution equation and incorporate the characteristic function $\chi(x, y, z)$,

$$\frac{\partial S}{\partial t}=(1-\chi(x,y,z))\gamma\left[\frac{(S_x^2+S_y^2)S_{zz}+(S_x^2+S_y^2)S_{yy}+(S_y^2+S_z^2)S_{xx}}{S_x^2+S_y^2+S_z^2}-\frac{2S_xS_yS_{xy}+2S_xS_zS_{xz}+2S_zS_yS_{yz}}{S_x^2+S_y^2+S_z^2}+\frac{\sqrt{S_x^2+S_y^2+S_x^2}}{\gamma}V_1\right]. \quad (21)$$

The approximated steady state solution of $S(x, y, z, t)$ is obtained after certain large iteration time, $t = T_0$. It is a smooth function with relatively rapid changes near the protected atomic boundaries of D_χ. However, the hypersurface $S(x, y, z, T_0)$ gives rise to a family of isosurfaces. It is easy to extract an isosurface by setting $S(x, y, z, T_0) = C$, where C is a value between 0 and 1. The value of C can be adjusted to achieve the effect of multiresolution surfaces. However, by choosing $C = 0.5$, one can attain a better accuracy in the calculations of the surface area and surface enclosed volume. For all the surfaces demonstrated in this paper, we choose $C = 0.5$. Figure 1 gives an example of the comparison between the molecular surface and the surface generated from the Laplace-Beltrami flow. It can be seen from the figure that the surface evolved from the Laplace-Beltrami is free from singularities.

When there are external potentials ($V_1 \neq 0$), the surface generation is usually coupled with the calculation of other physical variables governed by other equations. These coupled equations should be solved iteratively. For example, to take into consideration the electrostatic effect, the PB model is commonly employed. Due to the vastly different quantities of the dielectric constants in solute and solvent domains, the elliptic interface problems are frequently updated in the geometric and potential driven models.

2.2 EMDB data processing and surface generation

2.2.1 EMDB Data—As the data from the cryo-EM accumulated, EMDataBank.org was established to create a global deposition and retrieval network for cryo-EM maps and

associated metadata. It also serves as a portal of software tools for standardized map format conversion, segmentation, model assessment, visualization, and data integration. A list of EM software packages can be found in the website http://emdatabank.org/emsoftware.html. MRC (Medical Research Council) is the file format used in cryo-EM data, in which the data are stored on a 3D grid of voxels (volumetric cells) with values corresponding to the density of electrons. It was developed by the MRC Laboratory of Molecular Biology, and is supported by almost every molecular graphics software package that supports volumetric data, such as, visual molecular dynamics (VMD), PyMOL, and UCSF Chimera. A detailed specification of the MRC file format can be found at the website http://ami.scripps.edu/software/mrctools/mrc_specification.php. The Matlab code for extracting the voxel value information is also mentioned in the above webpage. Here we modify the code and incorporate a simple procedure to store the values in the ".dat" format for further use. For example, if we want to extract the information of "emd_1048.map", the related Matlab code is given in Appendix D.

To avoid any confusion, here "emd_1048.map" contains the data directly downloaded from EMDataBank's website http://www.ebi.ac.uk/pdbe-srv/emsearch/form and is stored in the standard MRC format. With VMD, one can visualize the data directly. In the VMD, when loading the data, one needs to select the "CCP4,MRC density map" option for "Determine file type". In the "Graphical representation", the "Drawing method" is chosen to be "Isosurface". For the "isovalue" option, one needs to key in the recommended iso-values found in the webpage of the map data. One can select the "ColorID" in the "Coloring method" and adjust the value to render the surface with a specific color.

2.2.2 Noise reduction of EMD—It is seen from the left chart in Fig. 2 that electron tomography sometimes produces extremely noisy and low contrast 3D density maps. The poor signal-to-noise ratio (SNR) hinders visualization and interpretation. Therefore, noise filtering techniques are indispensable when treating EM data. There are a number of effective methods and schemes for this task, like wavelet transform techniques, nonlinear anisotropic diffusions, Beltrami flow, bilateral filter, and iterative median filtering [58, 25, 24, 59, 33, 42, 60].

To further improve the noise reduction efficiency, high order geometric PDEs are employed, see Appendix C. We can control the process by three parameters, the integration time t, the PDE order q, and the external term $P(u, |\nabla_u|)$. For the protein surface generation from the PDB, a proper combination of the PDE order and integration time is required to deliver high-quality surface [81, 22, 23]. If the solvation process is considered, the potential driven term should incorporate both the nonpolar and polar effects. In our noise reduction procedure for EMDB data, the external term is omitted. The integration time is adjusted to give different levels of noise amplitude and image construction. Figure 2 demonstrates an example of noise removal of EMD5119. To be specific, left chart in Fig. 2 is generated with the suggested contour level value 0.25 from the original noisy data. The right chart is produced with same contour level value but from the processed data. The noise is reduced efficiently, while the salient edges are preserved very well.

2.3 Surface electrostatic analysis

One of the most important problems in biological sciences is the understanding of electrostatic interactions in biomolecules. Electrostatic interactions are ubiquitous in any system of charged or polar molecules, such as proteins, nucleic acids, lipid bilayers, and sugars. The importance of electrostatics in biomolecular systems is due to the fact that electrostatic interactions frequently dominate other forces and determine the structure, function, stability, dynamics and transport of macromolecular systems. As shown in Section

A.2, electrostatic analysis is readily coupled with surface analysis. Electrostatic potential can be obtained by solving the Poisson-Boltzmann equation. The surface electrostatic potential, obtained by the projection of electrostatic potential on a surface, is important for the understanding of protein-protein interactions, ligand binding, solvation, and drug design.

From the mathematical point of view, solvent-solute boundary can be treated as an interface. If we use the Poisson equation or the PB equation to describe the electrostatic potential with the different dielectric constants in the solvent and solute domains, an elliptic interface problem is constructed. The well-posedness of this equation relies on the interface information, which usually involves the jump conditions of the function values and the derivatives with respect to normal directions on the interface.

$$[u]=u^+ - u^-=\Psi_1, \forall \mathbf{x}\text{on}\Gamma \quad (22)$$

$$[\beta u_{\mathbf{n}}]=\beta^+ u_{\mathbf{n}}^+ - \beta^- u_{\mathbf{n}}^-=\Psi_2, \forall \mathbf{x}\text{on}\Gamma \quad (23)$$

where Γ denotes the interface, and vector **n** is the normal direction. Here u^+, $u_{\mathbf{n}}^+$ and β^+ denote the limiting values of from one subdomain Ω^+, and u^-, $u_{\mathbf{n}}^-$ and β^- from the other Ω^-. The total computational domain $\Omega = \Omega^+ \cup \Omega^-$, and interface $\Gamma = \Omega^+ \cup \Omega^-$. In the PB model, the variable u is replaced by the electrostatic potential Φ. Due to the continuity of electrostatic potential and its flux density, the right terms in the jump condition equals zero, that is, $\Psi_1 = \Psi_2 = 0$.

2.3.1 Extraction of interface information from volumetric data—Interface information is required for both electrostatic analysis and geometric analysis. To extract interface information from volumetric data, one needs to know the isovalues (or level set values) at the Cartesian grid nodes. The volumetric data can be treated as a surface function on a grid with one value assigned to each grid node. When a new Cartesian mesh is employed in computation, the isovalues on the new grid nodes need to be evaluated for further applications in elliptic interface schemes and curvature analysis. For instance, if one has volumetric data $\{S_\upsilon\}_{320\times320\times320}$, the Cartesian mesh size is set to $N_x \times N_y \times N_z$ (21 × 21 × 21 in the example), and $\{S_\upsilon\}$ should be sampled on the grid to produce $\{S\}_{N_x \times N_y \times N_z}$. Here $\{S\}_{N_x \times N_y \times N_z}$ can be seen as the discrete representation of the trilinearly interpolated surface function $S(x, y, z)$. We provide details for the trilinear interpolation below. First, we assume the domain for given volumetric data as $\Omega_\upsilon = [1, 320] \times [1, 320] \times [1, 320]$, and denote the coordinates of node (i, j, k) on target Cartesian grid by (x_i, y_j, z_k), expressed as

$$(x_i, y_i, z_k)=\left(320\frac{i-1}{21}+1, 320\frac{j-1}{21}+1, 320\frac{k-1}{21}+1\right). \quad (24)$$

Then, we denote the integer part of (x_i, y_j, z_k) as (i_t, j_t, k_t), and the fractional part as (x_d, y_d, z_d). The Cartesian mesh node (i, j, k) can be viewed as a point in Ω_υ, and encompassed by the cube formed by eight original grid nodes, with coordinates $(\mathbf{v}_m)_{m=1,\ldots,8}$. It is seen that the coordinates for diagonal two nodes are $\mathbf{v}_1 = (i_t, j_t, k_t)$ and $\mathbf{v}_8 = (i_t + 1, j_t + 1, k_t + 1)$. If the isovalues on these grid nodes are denoted as $S_\upsilon(\mathbf{v}_m)_{m=1,\ldots,8}$, to calculate the isovalues $S_{i,j,k}$, eight weights $W(m)_{m=1,\ldots,8}$ are needed in the interpolation form,

$$S_{i,j,k}=\sum_{m=1}^{8} S_\upsilon(\mathbf{v}_m)W(m). \quad (25)$$

One can certainly choose more than 8 points to carry out the evaluation and also the weights are by no means unique. Here we just make use of the Lagrangian shape functions on cubes,

and map the original cube to a logical cube with the coordinate of the diagonal two nodes as (−1, −1, −1) and (1, 1, 1). The mesh point (i, j, k) is then projected to a new node with coordinate ($\xi\ \eta\ \zeta$)

$$(\xi,\eta,\zeta)=(2\cdot x_d-1,2\cdot y_d-1,2\cdot z_d-1). \quad (26)$$

The weight functions corresponded to cubic nodes can be represented as

$$W(m)=\frac{1}{8}(1+\xi\xi_m)(1+\eta\eta_m)(1+\zeta\zeta_m), m=1,2,\ldots,8, \quad (27)$$

where (ξ_m, η_m, ζ_m) are the nodal coordinates of the logical cube.

For volumetric data, a recommended isovalue is given to define the interface, denoted as Γ. Therefore, the region with isovalues bigger than the recommended one is specified as the biomolecular subdomain, which we usually denote as Ω^+ and with the opposite Ω^-. Each mesh node is assigned to a region. For a given node, when its surrounding six nodes are in the same subdomain, this node is defined as a regular node. Otherwise if any of its six surrounding nodes is located in the other subdomain, the node is an irregular node. Irregular nodes usually occur in pairs.

The real physical domain for the voxel data can be also found from the EMDB data description. The unit is usually in angstrom or nanometer. It is easy to interpolate the physical coordinates into Cartesian mesh nodes. To avoid heavy notation, we still use (x_i, y_j, z_k) to represent the coordinate of node (i, j, k). However, it is now assumed to be in the real physical domain.

In the matched interface boundary (MIB) algorithm, in order to implement the jump conditions, we need to know the interface information between the pair of irregular nodes. For example, if nodes (i, j, k) and (i+1, j, k) are located in two different subdomains, the coordinate of the interface intersecting with the mesh is specified as $\mathbf{v}_o = (x_o, y_o, z_o)$,

$$\mathbf{v}_o=\left(\frac{S_0-S_{i,j,k}}{S_{i+1,j,k}-S_{i,j,k}}(x_{i+1}-x_i)+x_i,y_i,z_k\right), \quad (28)$$

where S_0 stands for the recommended isovalue of the interface, and $S_{i,j,k}$ represents the isovalue at node (i, j, k). The normal direction is interpolated from the expression,

$$\mathbf{n}_o=\frac{S_0-S_{i,j,k}}{S_{i+1,j,k}-S_{i,j,k}}(\mathbf{n}_{i+1,j,k}-\mathbf{n}_{i,j,k})+\mathbf{n}_{i,j,k}, \quad (29)$$

where $\mathbf{n}_o$ is the normal direction at interface intersecting with mesh line point. The value of $\mathbf{n}_{i+1,j,k}$ can be evaluated from

$$\mathbf{n}_{i+1,j,k}=\left(\frac{S_{i+2,j,k}-S_{i,j,k}}{x_{i+2}-x_i},\frac{S_{i+1,j+1,k}-S_{i+1,j-1,k}}{y_{j+1}-y_{j-1}},\frac{S_{i+1,j,k+1}-S_{i+1,j,k-1}}{z_{k+1}-z_{k-1}}\right). \quad (30)$$

The value for $\mathbf{n}_{i,j,k}$ can be calculated in the same manner.

In the MIB algorithm, the Cartesian grid is employed and jump conditions are only implemented at points where the interface intersects with grid lines. However, in the MIB Galerkin method, to guarantee the reversibility of the matrix, for the special geometric singularity situations, one needs to enforce jump conditions not only at points where

interface intersects with grid lines but also on the interface off grid lines. Therefore, new schemes are needed. For example, two irregular points can be located in the different subdomains and near the node where the fictitious value is evaluated. The coordinates for these two nodes are $\mathbf{v}_1$ and $\mathbf{v}_2$, and the distance between these two nodes is

$$d_{12}=\sqrt{(\upsilon_{1,x}-\upsilon_{2,x})^2+(\upsilon_{1,y}-\upsilon_{2,y})^2+(\upsilon_{1,z}-\upsilon_{2,z})^2}. \quad (31)$$

The interface point between these two nodes can be interpolated as

$$x_o=\frac{S_0-S(\mathbf{v}_1)}{S(\mathbf{v}_2)-S(\mathbf{v}_1)}\frac{(\upsilon_{2,x}-\upsilon_{1,x})^2}{d_{12}}+\upsilon_{1,x}, \quad (32)$$

$$y_o=\frac{S_0-S(\mathbf{v}_1)}{S(\mathbf{v}_2)-S(\mathbf{v}_1)}\frac{(\upsilon_{2,y}-\upsilon_{1,y})^2}{d_{12}}+\upsilon_{1,y}, \quad (33)$$

$$z_o=\frac{S_0-S(\mathbf{v}_1)}{S(\mathbf{v}_2)-S(\mathbf{v}_1)}\frac{(\upsilon_{2,z}-\upsilon_{1,z})^2}{d_{12}}+\upsilon_{1,z}, \quad (34)$$

and the normal is

$$\mathbf{n}_o=\frac{S_0-S(\mathbf{v}_1)}{S(\mathbf{v}_2)-S(\mathbf{v}_1)}(\mathbf{n}(\mathbf{v}_2)-\mathbf{n}(\mathbf{v}_1))+\mathbf{n}(\mathbf{v}_1). \quad (35)$$

In the MIB Galerkin method [73, 72], the mesh is based on the Cartesian grid, while the definition of irregular points is slightly different from that of the traditional definition in the MIB collocation method [83, 84, 76, 77, 29, 80]. Basically, the irregular points do not necessarily come in pairs. It can be seen that the MIB Galerkin method is a Galerkin implementation of the MIB ideas.

2.3.2 Solution of the Poisson-Boltzmann equation—In the MIB method, the Cartesian grid is employed. In its numerical schemes, the interface jump conditions are employed only at the intersecting points between the interface and the mesh lines. If the interface is analytically given, for instance, a sphere with certain radius, the coordinates of intersecting points can be easily determined when the mesh size is specified. However, for volumetric data from EMDataBank, an interpolation procedure is required. A detailed description is given below.

The MIB method has been developed to solve the elliptic interface problems with geometric singularities [83, 84, 76, 77, 29, 80]. It delivers the second order accuracy in solving the PB equation with complex protein interfaces, possible geometric singularities and charge singularities. The essential ideas of the MIB method are the following: The standard finite difference schemes are used on the simple Cartesian grids; the fictitious values are employed near the interface as a smooth extension of the non-smooth functions; interface jump conditions are incorporated into the calculation of the fictitious values; and to construct high-order schemes, the lowest order jump conditions are used repeatedly. For the PB equation, one more challenge is the charge singularities, which represent the partial charges of protein atoms assigned by the CHARMM or AMBER force field. In the PB equation, partial charges are represented by Dirac delta functions in the source term. Through the use of the Green's function formulation, the charge singularities are transformed into interface flux jump conditions, which are integrated into the geometric singularities framework [29].

When the Laplace-Beltrami equation is coupled with the PB equation, they should be solved iteratively until self-consistency is reached [11, 12]. Two approaches have been employed. One approach is a simple relaxation algorithm: the characteristic function S and electrostatic potential φ is updated by a linear combination of the previous ones and the new ones. Basically, we start with the initial condition of hypersurface function S_0. This value is used in the PB equation to calculate a temporary electrostatic potential φ. The calculated φ value is then used in the Laplace-Beltrami equation to evaluate the new S. Instead of using this new S as the new input in the PB equation, we use a weighted average as described below,

$$S_{n+1,\text{new}}=\alpha S_{n+1,\text{old}}+(1-\alpha)S_n, 0\leq\alpha\leq 1, \quad (36)$$

where $S_{n+1,\text{new}}$ is the one applied to the evaluation of the electrostatic potential $\varphi_{n+1,\text{old}}$. Once we have $\varphi_{n+1,\text{old}}$, the electrostatic potential is updated with as,

$$\phi_{n+1,\text{new}}=\alpha'\phi_{n+1,\text{old}}+(1-\alpha')\phi_n, 0\leq\alpha'\leq 1. \quad (37)$$

Relaxation coefficients are denoted as α and α'. Again, the updated $\varphi_{n+1,\text{new}}$ is used in Laplace-Beltrami equation to update the hypersurface function to $S_{n+2,\text{old}}$.

The other approach is to refresh the electrostatic potential at a lower frequency than that for updating the surface function. Basically, after a number of iterations (in our tests, 10 to 100 steps) of the generalized Laplace Beltrami equation, the electrostatic potential is then updated. By adaptively changing the number of iterations, one can increase the computational efficiency especially when the change in the surface function during each iteration step is very small.

The coupled system of Laplace-Beltrami equation and PB equation is highly nonlinear. To our best knowledge, there is no rigorous mathematical proof of the existence and uniqueness of the solution. In order to validate our model and algorithm, we evaluate the solvation free energy and compare it with the experimental results [11, 12]. We also check the volume and area of the protein structure calculated from the model during the iteration, and ensure that the convergence to the steady state is observed [12, 68]. In our algorithm, the solvation free energy is often used as an indication of the steady state. Its value can be obtained from minimizing the nonpolar part, polar part and homogeneous energy part as follows

$$\Delta G=G_{\text{nonpolar}}+G_{\text{polar}}-G_{\text{homogeneous}}=\int_\Omega\left\{\gamma|\nabla S|+pS+(1-S)\sum_\alpha\rho_\alpha U_\alpha\right\}d\mathbf{r}+\frac{1}{2}\sum_{i=1}^{N}Q(\mathrm{x_i})(\phi(\mathrm{x_i})\phi_0(\mathrm{x_i})),$$

where the nonpolar part is from Eq. (76), and $\varphi_0(\mathrm{r_i})$ is the electrostatic potential for the homogeneous environment condition at ith atomic position. Figure 3 gives a comparison of electrostatic surface potentials on the molecular surface and the surface obtained from a generalized Laplace-Beltrami flow.

2.4 Computational aspects of geometric features

We have introduced the procedure building the characteristic function representing the surface based on PDB data. For EMDB data, the surface is extracted from the volumetric data after noise reduction. Therefore, the protein surfaces from these two data banks are all in the Cartesian representation. To evaluate the surface properties, Cartesian representation based algorithms are needed. In this section, the computational methods for the calculating basic geometric features are presented and their potential applications are discussed.

2.4.1 Surface area and volume calculation—Based on PDB data and volumetric data from EMDB, the biomolecular surface can be represented by using characteristic functions in two ways: one is the sharp interface by extracting certain iso-values; and the other is the smeared interface that varies in a certain iso-value range. The smeared interface (smooth surface function) is more physical as the radius of the atom is indeed obtained by the probability measure of the electron cloud around the atomic nucleus. Mathematically, the sharp interface is simpler and straightforward.

In the Cartesian representation, the area of a sharp surface can be evaluated as [28, 12]

$$\text{Area}=\sum_{o\in R}\left(\frac{|n_{o,x}|}{h}+\frac{|n_{o,y}|}{h}+\frac{|n_{o,z}|}{h}\right)h^3, \quad (38)$$

where R is the set of intersection points located inside the protein domain. Here $\mathrm{n}_o = (n_{o,x}, n_{o,y}, n_{o,z})$ are the normal for the intersection point. As the surface information is in the Cartesian representation, interpolation is used to evaluate the interface coordinates and norms, see Section 2.3.1 for details.

In a smeared surface representation, the mean surface area and the related coarea formula are defined in Eq. (69)

$$\text{Area}=\int_0^1\int_{S^{-1}(c)\cap\Omega} d\sigma dc=\int_\Omega |\nabla S(\mathbf{r})|d\mathbf{r}, \mathbf{r}\in\mathbb{R}^3, \quad (39)$$

where the surface integration is converted to a volume integration, which is easier to evaluate in the Cartesian representation.

We can obtain a similar expression for the volume calculation. When the protein surface is defined as a sharp interface, a simple summation is used [12],

$$\text{Vol}=\sum_{(i,j,k)\in\Omega}\tilde{\chi}(i,j,k)h^3, \quad (40)$$

where $\tilde{\chi}(i, j, k)$ is a characteristic function with value 1 inside the protein domain and value 0 for the other. For a smooth surface function, the volume is computed as

$$\text{Vol}=\int_\Omega S(\mathbf{r})dr=\sum_{(i,j,k)\in\Omega}S(x_i,y_j,z_k)h^3, \quad (41)$$

where $S\in[0, 1]$ is the surface characteristic function.

The scheme for computing sharp surface area in the Cartesian representation is tested with an analytical example. We use a sphere with the analytical expression of $x^2 + y^2 + z^2 = 4$ in the domain $[-5, 5] \times [-5, 5] \times [-5, 5]$. The second order central finite difference scheme is used in our computation. The error is defined as the absolute value of the difference between the analytical surface area and the calculated surface area. The result is presented in Table 2. It is seen that second order accuracy is attained.

2.4.2 Curvature evaluation—The evaluation of curvature properties from iso-surface embedded volumetric data has been thoroughly studied in geometric modeling. There are a variety of elegant methods in the literature. Essentially, the first and second fundamental forms in the differential geometry are involved in the definition and evaluation of the curvatures. We give a brief introduction of the mathematical background [56, 7].

The surface of interest can be extracted from a level set with iso-value S_0, i.e., $S(x, y, z) = S_0$. We assume S to be non-degenerate, i.e., the norm of its gradient is non-zero when it is equal to S_0. Without loss of generality, we further assume that its projection onto z is non-zero. According to the implicit function theorem, locally, there exists a function $z = f(x, y)$, which parameterize the surface as $\mathbf{S}(x, y) = (x, y, f(x, y))$. One has the relation $S(x, y, f(x, y)) = S_0$. The differentiation with respect to x and y produces two more equations

$$S_x(x,y,f(x,y))+S_z(x,y,f(x,y))f_x(x,y)=0, \quad (42)$$

$$S_y(x,y,f(x,y))+S_z(x,y,f(x,y))f_y(x,y)=0. \quad (43)$$

Thus $f_x(x, y)$ and $f_y(x, y)$ can be expressed as:

$$f_x(x,y)=-\frac{S_x(x,y,z)}{S_z(x,y,z)}, \quad (44)$$

$$f_y(x,y)=-\frac{S_y(x,y,z)}{S_z(x,y,z)}. \quad (45)$$

We define $E(x, y, z)$, $F(x, y, z)$, $G(x, y, z)$, $L(x, y, z)$, $M(x, y, z)$ and $N(x, y, z)$ below to be the coefficients in the first and second fundamental forms. For simplicity, we omit parameter parts. Their values for surface $\mathbf{S} = (x, y, f)$ can be given as

$$E=\langle \mathbf{S}_x, \mathbf{S}_x\rangle=1+f_x^2=1+\frac{S_x^2}{S_z^2}; \quad (46)$$

$$F=\langle \mathbf{S}_x, \mathbf{S}_y\rangle=f_xf_y=\frac{S_xS_y}{S_z^2}; \quad (47)$$

$$G=\langle \mathbf{S}_y, \mathbf{S}_y\rangle=1+f_y^2=1+\frac{S_y^2}{S_z^2}; \quad (48)$$

$$L=\langle \mathbf{S}_{xx}, \mathbf{n}\rangle=\frac{2S_xS_zS_{xz}-S_x^2S_{zz}-S_z^2S_{xx}}{g^{\frac{1}{2}}S_z^2}; \quad (49)$$

$$M=\langle \mathbf{S}_{xy}, \mathbf{n}\rangle=\frac{S_xS_zS_{yz}+S_yS_zS_{xz}-S_xS_yS_{zz}-S_z^2S_{xy}}{g^{\frac{1}{2}}S_z^2}; \quad (50)$$

$$N=\langle \mathbf{S}_{yy}, \mathbf{n}\rangle=\frac{2S_yS_zS_{yz}-S_y^2S_{zz}-S_z^2S_{yy}}{g^{\frac{1}{2}}S_z^2}, \quad (51)$$

where $g=S_x^2+S_y^2+S_z^2$ and the normal direction $\mathbf{n}=\frac{(S_x,S_y,S_z)}{g^{\frac{1}{2}}}$. As the Gaussian curvature can be represented as the ratio of the determinants of the second and first fundamental forms, it can be given by

$$K=\frac{2S_xS_yS_{xz}S_{yz}+2S_xS_zS_{xy}S_{yz}+2S_yS_zS_{xy}S_{xz}}{g^2} - \frac{2S_xS_zS_{xz}S_{yy}+2S_yS_zS_{xx}S_{yz}+2S_xS_yS_{xy}S_{zz}}{g^2} + \frac{S_z^2S_{xx}S_{yy}+S_x^2S_{yy}S_{zz}+S_yS_{xx}S_{zz}}{g^2} - \frac{S_x^2S_{yz}^2+S_y^2S_{xz}^2+S_z^2S_{xy}^2}{g^2}. \quad (52)$$

The mean curvature is the average second derivative with respect to the normal direction,

$$H=\frac{2S_xS_yS_{xz}+2S_xS_zS_{xz}+2S_yS_zS_{yz}-(S_y^2+S_z^2)S_{xx}-(S_x^2+S_z^2)S_{yy}-(S_x^2+S_y^2)S_{zz}}{2g^{\frac{3}{2}}}. \quad (53)$$

An alternative algorithm for the curvature extraction from volumetric data is the Hessian matrix method [34]. For volumetric data $S(x, y, z)$, we define the surface gradient $\mathbf{g}$ and surface norm $\mathbf{n}$.

$$\mathbf{g}=\nabla S=(S_x, S_y, S_z); \quad (54)$$

$$\mathbf{n}=-\frac{\mathbf{g}}{|\mathbf{g}|}. \quad (55)$$

The Hessian matrix, $\mathbf{H}$, is given by

$$\mathbf{H}=\begin{bmatrix} \frac{\partial^2 S}{\partial^2 x} & \frac{\partial^2 S}{\partial x \partial y} & \frac{\partial^2 S}{\partial x \partial y} \\ \frac{\partial^2 S}{\partial x \partial y} & \frac{\partial^2 S}{\partial^2 y} & \frac{\partial^2 S}{\partial y \partial z} \\ \frac{\partial^2 S}{\partial x \partial z} & \frac{\partial^2 S}{\partial y \partial z} & \frac{\partial^2 S}{\partial^2 z} \end{bmatrix}. \quad (56)$$

The two principal curvatures can be evaluated by the following procedure.

1. Calculate matrix $\mathbf{P} = \mathbf{I} - \mathbf{nn}^{\mathbf{T}}$, here $\mathbf{I}$ is the identity matrix and $\mathbf{T}$ denotes the transpose;
2. Evaluate matrix $\mathbf{G}=\mathbf{I}-\frac{\mathbf{PHP}}{|\mathbf{g}|}$,
$$\mathbf{G}=(g_{ij})_{(i,j=1,3)} \quad (57)$$
3. Calculate the trace t and Frobenius norm f of matrix $\mathbf{G}$;
$$t=g_{11}+g_{22}+g_{33}; \quad (58)$$
$$f=\|\mathbf{G}\|=\sqrt{\sum_i\sum_j g_{ij}^2}; \quad (59)$$

$$\kappa_1=\frac{t+\sqrt{2f^2-t^2}}{2}; \quad (60)$$

$$\kappa_2=\frac{t-\sqrt{2f^2-t^2}}{2}. \quad (61)$$

When the two principal curvatures are available, the Gaussian curvature K and mean curvature H can be easily calculated,

$$K=\kappa_1\kappa_2; \quad (62)$$

$$H=\frac{\kappa_1+\kappa_2}{2}. \quad (63)$$

Essentially, the Hessian matrix method generates the same results as the above algorithm derived from the first and second fundamental form.

Numerical test for analytical cases: We use the second order central difference scheme to do the discretization. Two analytical examples are considered. We denote L_∞ and L_2 the L_∞ error and L_2 error.

Case 1. We set the domain as [−10, 10] × [−10, 10] × [−10, 10], and define a surface as

$$Z(x,y)=\frac{(x^2-y^2)(x^3-y^3)}{2000}. \quad (64)$$

Basically, the volumetric data $f(x, y, z)$ are defined as $f(x, y, z) = z - Z(x, y)$. Therefore, the analytical surface $Z(x,y)=\frac{(x^2-y^2)(x^3-y^3)}{2000}$ can be extracted by setting $f(x, y, z) = 0$. The analytical expressions for Gaussian curvature and mean curvature can be calculated,

$$K=\frac{z_{xx}z_{yy}-z_{xy}^2}{(1+z_x^2+z_y^2)^2}, \quad (65)$$

$$H=\frac{(1+z_x^2)z_{yy}-2z_xz_yz_{xy}+(1+z_y^2)z_{xx}}{2(1+z_x^2+z_y^2)^{\frac{3}{2}}}. \quad (66)$$

The numerical results are demonstrated in Fig. 4. Tables 3 and 4 give the error and associated convergence order. As we use the second order finite difference scheme to evaluate the derivatives, the second order accuracy is obtained. We also tested the Hessian matrix method, it generates the same results.

Case 2. In this case the domain is set as [−10, 10] × [−10, 10] × [−10, 10], and a surface is defined as

$$Z(x,y)=\frac{(x^3-y^3)}{30}. \quad (67)$$

The volumetric data $f(x, y, z)$ are defined as $f(x, y, z) = z - Z(x, y)$. Therefore, the analytical surface can be extracted by setting $f(x, y, z) = 0$. We can calculate the analytical solution for the Gaussian curvature and the mean curvature using Eqs. 65 and 66. Fig. 5 demonstrates the numerical results. The error and associated convergence order are listed in Tables 5 and 6. The Hessian matrix method gives the same results and both methods achieve the second order accuracy.

Numerical test for protein data: Having validated the two methods for curvature evaluation, we apply them to calculation of the structural features of proteins. Three protein structures are considered, two are from EMDB, i.e., EMD5273 and EMD5020, and the other one is from PDB with ID:1PPL. Figures 6, 7 and 8 demonstrate our results. All the protein surfaces generated in this section are extracted with the isovalue C=0.5. The data size for 1PPL is 146 × 117 × 97. EMD5273 and EMD5020 have the same data size of 100 × 100 × 100. As curvature evaluation algorithms involve only simple interpolation, the computation cost is very small. On a PC with Pentium 4 CPU 3.60GHz and 1.00 GB RAM, the computation times are about 4.2, 2.1 and 2.2 second for proteins EMD5273, EMD5020 and 1PPL, respectively.

These curvatures describe the concave and convex properties of the protein surface, see Appendix C. It is well known that in drug design and protein-protein interaction, the surface geometry is of significant importance [16]. Usually, the geometry of the drug should match to a concave region of the protein just like the key and lock relation. The same applies when two proteins interact with each other, or when a substrate binds to the active site of an enzyme. The quantitative measurement of curvatures has a great potential for further modeling and analysis of the geometric impact on biomolecular interactions.

Gaussian curvature characterizes the topological property of a surface. When integrated over the surface, Gaussian curvature gives rise to the information of the genus number, which is, loosely speaking, the number of “holes” in the biomolecule. The genus number can be applied to systems like ion channel proteins, whose open state and close state have different genus numbers. From the minimum curvature and shape index, we can obtain a rather clear picture of concave regions. Actually, the concaveness can be quantitatively characterized by the values of minimum curvature and shape index. Similarly, the convexity can be parameterized by the maximum curvature and shape index. The curvedness provides the information about the amplitude of the curvature, e.g., a large value usually indicates a sharp edge and/or corner.

The traditional MS suffers from geometric singularities, for which curvatures are undefined. Computationally, near geometric singularities, curvatures tend to have much dramatic local variations and the accuracy of computational results is reduced. In our MMS and surface generated from geometric and potential driven geometric flows, the geometric singularities are removed, and the surface is smooth with less local curvature uctuations. Further, a multiresolution model is proposed in our recent work [22]. Obtained with an adjustable parameter, a family of multiresolution surfaces can be designed to reduce local curvature variations. Consequently, concave regions and convex regions reflect the molecular morphology, instead of local atomic characteristics. In differential geometry based multiscale multiresolution models, the electrostatic potential is also coupled with the molecular surface generation. With polar and nonpolar areas defined by electrostatic potential, and concave and convex regions evaluated by the above curvature schemes, our approaches have a great potential for the prediction of protein active sites and/or binding sites.

2.5 Polarized curvature and binding site prediction

Based on the above curvature analysis and electrostatic characterization, it is clear that a potential protein binding site should be both electrostatically favorable and geometrically favorable. To combine these compatibilities, we propose polarized curvatures as the products of electrostatic potentials and curvatures. Specifically, the maximal curvature κ_1 and minimal curvature κ_2 are employed to construct their products with positive electrostatic surface potential Φ^+ and negative electrostatic surface potential Φ^-. Large amplitudes of these products indicate four different potential binding sites as summarized in Table 7. For example, a large amplitude of $\Phi^+ \times \kappa_1$ on a certain region of a protein surface indicates a potential binding site for a negatively charged protein or virus, while a large amplitude of $\Phi^+ \times \kappa_2$ indicates a potential binding site for a negatively charged small ligand. Similar behavior can be stated for the products of $\Phi^- \times \kappa_2$ and $\Phi^- \times \kappa_1$.

Figure 9 demonstrates the effectiveness of our proposed polarized curvature analysis. The top row illustrates the electrostatic surface maps and (small ligand) binding sites of four proteins. The bottom row displays the predictions of polarized curvatures ($\Phi \times \kappa_2$). In these cases, the minimal curvature (κ_2) is used to predict the concave regions of protein surfaces for potential binding sites of small ligands. The protein on the left chart is positively charged at its binding site and the rest of proteins are all negatively charged at their binding sites. The polarized curvatures shown in the bottom row give correct predictions for all binding sites.

In our future work, we will combine the polarized curvature analysis and the binding affinity analysis readily available in our multiscale solvation model [12] for more accurate prediction of protein-ligand binding, protein-DNA specificity and protein-protein interactions. We will make this approach automatic and robust.

3 Conclusion

Geometric modeling has widespread applications in the visualization, analysis and characterization of the macromolecules. For proteins, their structural features are intrinsically associated with their functions and molecular mechanisms. The exploration of the geometric features of a protein molecular surface enhances our understanding of molecular morphology and molecular mechanism, and allows significant applications to drug design and protein-protein interaction. This is particularly true when the geometric modeling is associated with the electrostatic analysis. The present work offers expository investigation and comprehensive summary of tools, algorithms and methodologies for geometric modeling of macromolecules.

Our study is based on two major biomolecular structure sources collected from experiments: the Protein Data bank (PDB) and the Electron Microscopy Data Bank (EMDB). The PDB contains information about structures obtained mainly by using X-ray crystallography and NMR spectroscopy at the atomic level resolution. Whereas, EMDB provides information mainly about multiproteins, organelles, viruses, and subcellular complexes obtained mostly from cryo-Electron Microscopy (cryo-EM) at the molecular level resolution. In the present paper, based on data from the PDB and the EMDB, related geometric modeling methods, software packages and visualization tools are provided and discussed in great detail.

The protein data from the PDB are in atomic resolution, so that crucial information like atom positions, van der Walls radius and partial charges can be obtained either directly or indirectly. Different definitions of the macromolecular surface have been proposed and constructed based on experimental data. However, the resulting surfaces usually suffer from geometric singularities (i.e., tips, cusps and self-intersecting facets) and violate the energy

minimization principle, due to the fact that they are just *ad hoc* divisions of the protein and its surroundings. The minimal molecular surface (MMS) is proposed as a surface that minimizes the surface free energy. This variational formulation based biomolecular surface fulfills the principle of energy minimization, while producing a smooth surface through Laplace-Beltrami flows obtained from the Euler-Lagrange equation. As the solvation process is of fundamental importance to biomolecular systems, it should be considered in the surface modeling. By adding the solvation energy, which is composed of nonpolar and polar parts, into the total free energy functional and by using the Euler-Lagrange equation, the geometric and potential driven Laplace-Beltrami flow is formulated. Essentially, the external potential term incorporates various solvation effects, except the surface tension. Further, in different types of biomolecular systems, other related effects, such as chemical potential and fluid flow, are accounted in external potential terms as well. In this paper, we explore all the surface generation related geometric aspects, including surface modeling, computational methods, algorithms and techniques.

The data from the EMDB, in contrast, is in a volumetric format and usually without detailed atomic information. These data often have a poor signal to noise ratio (SNR) and a noise reduction process is required. High order geometric PDEs can suppress the high-frequency components efficiently. In this paper, for the first time, the high order geometric PDEs are applied to the EMD noise removal. With the suitable PDE order and iteration time, the noise is drastically reduced, while image features are preserved.

Curvature properties indicate the concave or convex regions, which are likely to be the potential binding sites or active sites. Within the framework of the Cartesian representation, we tested second order computational algorithms for curvature evaluation. Six different curvature descriptors, including Gaussian curvature, mean curvature, maximum curvature, minimum curvature, shape index, and curvedness, are employed for the first time to the two types of protein surfaces, variational surfaces generated from PDB data and surfaces extracted from denoised EMDB data. An interesting feature of our work is that the curvature analysis for surfaces generated from our variational model is paired with the electrostatic analysis resulted from the same model. Such a feature enables us to introduce polarized curvatures for the screen of protein-ligand binding and protein-protein interaction sites. We demonstrate that newly proposed polarized curvatures give rise to a good prediction of protein-ligand binding sites.

Acknowledgments

This work was supported in part by NSF grants IIS-0953096, IIS-1302285 and DMS-1160352, and NIH grant R01GM-090208. GWW acknowledges the Mathematical Biosciences Institute for hosting valuable workshops.

Appendices

A Mathematical models for surface generation

In this appendix, the theoretical models for surface generation are presented based on two experimental data sources, the PDB and the EMDB.

A.1 Minimal molecular surface

The minimal molecular surface (MMS) was introduced to construct surfaces free of geometric singularities via the variational principle [5, 7]. Although the original derivation was pursued in the Lagrangian formulation, we offer an Eulerian description in the present work. In this approach, a hypersurface S is defined to represent the biomolecular surface. Basically, we assign each point with coordinate (x, y, z) a value $S(x, y, z)$, which represents

the domain information. It can be viewed as a characteristic function of the solute domain. It is also known as a surface function. In Fig. 10, we illustrate this surface function by its 1D projection. By using geometric measure theory, the surface energy functional can be expressed as [64]

$$G_{\text{surface}}=\gamma\text{Area}=\int_{\mathbb{R}^3}\gamma|\nabla S|d\mathbf{r}, \quad (68)$$

where γ is the surface tension. As it is convenient for us to set up the total free functional as a 3D integral in $\mathbb{R}^3$, we make use of the concept of a mean surface area [64, 13] and the coarea formula [21] on a smooth surface

$$\text{Area}=\int_0^1\int_{S^{-1}(c)\cap\Omega}d\sigma dc=\int_\Omega|\nabla S(\mathbf{r})|d\mathbf{r},\Omega\subset\mathbb{R}^3. \quad (69)$$

Here hypersurface function S is distributed between 0 and 1, $S^{-1}(c)$ represents the inverse function of S and Ω is defined as the whole domain. The variation of Eq. (68) with respect to S leads to the vanishing of surface-tension weighted mean curvature $\nabla\cdot\left(\gamma\frac{\nabla S}{|\nabla S|}\right)=0$. The energy minimization of Eq. (68) can be realized by the introduction of an artificial time to obtain a generalized Laplace-Beltrami equation

$$\frac{\partial S}{\partial t}=|\nabla S|\nabla\cdot\left(\frac{\gamma\nabla S}{|\nabla S|}\right), \quad (70)$$

The final MMS, which is free of geometric singularity, is obtained by extracting an iso-surface from the steady state hypersurface function. During each iteration, we keep the value in the van der Waals surface enclosed domain unchanged. The computational details are described in Section 2. A cross section of the characteristic function S for protein 1PPL is demonstrated in Fig. 11. It can be seen that all contour lines between 0 and 1 can be chosen to extract isosurfaces. The MMS is often chosen to be $S = 0.5$, which has been found to be near optimal.

A.2 Variational multiscale models

To account for the local variations near the biomolecular surfaces due to interactions between solvent and solute molecules, solvation models are needed. It is well known that water contributes to more than half of the human cell mass. Essentially all the biological processes in a human cell occurred in aqueous environment. Therefore, solvation process is of tremendous importance for biomolecular systems. Phenomenologically, solvation processes can be described as the creation of a cavity in the solvent, the hydrogen bond breaking and formation at the solvent-solute interface, the surface reconstruction of the solute molecule, and entropy effect due to the solvent-solute mixing. Solute-solvent interactions are usually described by solvation energy, which can be decomposed into polar and nonpolar contributions. Due to the extensive sampling of the explicit methods, implicit approaches are commonly employed for solvation analysis and computation. Among them, the scaled particle theory (SPT) describes the surface free energy and the mechanical work of creating a cavity of the solute size in the solvent [57, 44]. To account for the solvent-solute dispersive interaction, an improved nonpolar solvation free energy functional is defined as

$$G_{\text{nonpolar}}=\gamma\text{Area}+p\text{Vol}+\int_{\Omega_s}Ud\mathbf{r},\mathbf{r}\in\mathbb{R}^3, \quad (71)$$

where p is the hydrodynamic pressure, and U denotes the solvent-solute non-electrostatic interactions, such as the van der Waals interaction. The final integration is over the whole solvent domain Ω_s. We define S as the characteristic function of the solute domain as in Section A.1. Keep in mind that the biomolecular surface can be viewed from two aspects, that is, it can be represented as a sharp interface by extracting certain iso-values from the hypersurface function, or it can be treated as a smooth interface between the solvent and the solute. The volume term can be easily defined as

$$\text{Vol}=\int_{\Omega_m} d\mathbf{r}=\int_{\Omega} S(\mathbf{r})d\mathbf{r}, \quad (72)$$

where Ω_m is the solute domain. On the other hand, $1 - S$ can be viewed as the characteristic function of the solvent domain. Therefore, the last term in Eq. (71) can be rewritten as

$$\int_{\Omega_s} U d\mathbf{r}=\int_{\Omega}(1 - S(\mathbf{r}))U d\mathbf{r}. \quad (73)$$

When there are multiple species in the aqueous environment, under the assumption of pairwise solvent-solute interactions, U can be expressed as a summation over all the interactions of the solvent species with each solute atom near the interface [68]

$$U=\sum_{\alpha}\rho_{\alpha}U_{\alpha} \quad (74)$$

$$U_{\alpha}=\sum_{j}U_{\alpha j}(\mathbf{r})+\sum_{\beta}U_{\alpha\beta}(\mathbf{r}), \quad (75)$$

where ρ_α ($\mathbf{r}$) is the density of αth solvent component, and U_α is the interaction potential of αth solvent species, which can be further divided into the summation of $U_{\alpha j}$, namely, the interaction potentials between the jth atom of the solute and the αth component of the solvent, and the summation of $U_{\alpha\beta}$, i.e., the interaction potentials between the αth component of the solvent and the βth component of the solvent.

Therefore, the total nonpolar solvation free energy can be rewritten as

$$G_{\text{nonpolar}}=\int_{\Omega}\left[\gamma|\nabla S|+pS+(1 - S)\sum_{\alpha}\rho_{\alpha}U_{\alpha}\right] d\mathbf{r}. \quad (76)$$

The Poisson-Boltzmann theory is of great significance in the analysis of polar solvation free energy. When there are multiple charge species in the aqueous surrounding, their concentration profiles can be represented by Boltzmann distributions [81]. For example, density of the αth solvent component can be expressed as an integral equation of density functional theory (DFT) type

$$\rho_{\alpha}=\rho_{\alpha 0}e^{-\frac{q_{\alpha}\Phi+U_{\alpha}-\mu_{\alpha 0}}{k_B T}}, \quad (77)$$

where k_B is the Boltzmann constant, T is the temperature, $\rho_{\alpha 0}$ denotes the reference bulk concentration of the αth charged species, and q_α denotes the valence of the αth charged species and Φ is the electrostatic potential. The term $\mu_{\alpha 0}$ is a relative reference chemical potential. Physically, for different charge species, i.e., $\rho_\alpha \neq \rho_\beta$, given that $\rho_{\alpha 0} = \rho_{\beta 0}$, $\mu_{\alpha 0}$ describes the difference in their equilibrium concentrations. The Boltzmann distribution is

an efficient way to avoid more expensive description of the solvent. Note that Eq. (77) is a statement that the chemical potential vanishes at equilibrium [68].

Using the Boltzmann distribution, the polar solvation free energy can be expressed as [64, 68]

$$G_{\text{polar}}=\int_{\Omega}\left\{S\left[-\frac{\varepsilon_m}{2}|\nabla\Phi|^2+\Phi\rho_m\right]+(1-S)\left[-\frac{\varepsilon_s}{2}|\nabla\Phi|^2-k_BT\sum_{\alpha}\rho_{\alpha 0}\left(e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}-1\right)\right]\right\}d\mathbf{r}, \quad (78)$$

where ε_s and ε_m are the dielectric constants of the solvent and solute, respectively, and ρ_m represents the fixed charge density of the solute. It is important to realize that the polar/nonpolar decomposition is by no means unique but rather arbitrary [11]. The Boltzmann distribution in Eq. (77) has the term U_α, and takes into account the energy of this solvent-solute non-electrostatic interactions. One should not double count these interactions when summarizing all the energy contributions. Thus, the total free energy functional for the solvation system can be described as

$$G^{\text{PB}}_{\text{total}}[S,\Phi]=\int_{\Omega}\left\{\gamma|\nabla S|+pS+S\left[-\frac{\varepsilon_m}{2}|\nabla\Phi|^2+\Phi\rho_m\right]+(1-S)\left[-\frac{\varepsilon_s}{2}|\nabla\Phi|^2-k_BT\sum_{\alpha}\rho_{\alpha 0}\left(e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}-1\right)\right]\right\}d\mathbf{r}. \quad (79)$$

The total solvation free energy in Eq. (79) is expressed as a functional of the surface function S and the electrostatic potential Φ. By applying the variational principle to minimize the total solvation free energy with respect to Φ and S, we obtain two equations

$$\frac{\delta G^{\text{PB}}_{\text{total}}}{\delta\Phi}\Rightarrow\nabla\cdot([(1-S)\varepsilon_s+S\varepsilon_m]\nabla\Phi)+S\rho_m+(1-S)\sum_{\alpha}q_\alpha\rho_{\alpha 0}e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}=0, \quad (80)$$

and

$$\frac{\delta G^{\text{PB}}_{\text{total}}}{\delta S}\Rightarrow-\nabla\cdot\left(\gamma\frac{\nabla S}{|\nabla S|}\right)+p-\frac{\varepsilon_m}{2}|\nabla\Phi|^2+\Phi\rho_m+\frac{\varepsilon_s}{2}|\nabla\Phi|^2+k_BT\sum_{\alpha}\rho_{\alpha 0}\left(e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}-1\right)=0. \quad (81)$$

From Eq. (80), the generalized Poisson-Boltzmann equation is obtained

$$-\nabla\cdot(\varepsilon(S)\nabla\Phi)=S_{\rho_m}+(1-S)\sum_{\alpha}q_\alpha\rho_{\alpha 0}e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}, \quad (82)$$

where $\varepsilon(S) = (1 - S)\varepsilon_s + S\varepsilon_m$ is the generalized permittivity function. Based on our earlier related work [7, 64, 11, 12], we introduce an artificial time to Eq. (81) and obtain a generalized Laplace-Beltrami equation

$$\frac{\partial S}{\partial t}=|\nabla S|\left[\nabla\cdot\left(\gamma\frac{\nabla S}{|\nabla S|}\right)+V_1\right], \quad (83)$$

where the potential driven term V_1 is given by

$$V_1=-p+\frac{\varepsilon_m}{2}|\nabla\Phi|^2-\Phi\rho_m-\frac{\varepsilon_s}{2}|\nabla\Phi|^2-k_BT\sum_{\alpha}\rho_{\alpha 0}\left(e^{-\frac{q_\alpha\Phi+U_\alpha-\mu_{\alpha 0}}{k_BT}}-1\right). \quad (84)$$

The generalized Laplace-Beltrami equation (83) describes the surface formation and evolution under the geometric and potential driven flow. The potential flow part consists of

both the polar and nonpolar energy effects, and the solvation process is naturally involved in the surface generation. The final solvent-solute interface can be extracted from the steady state, and will be free from geometric singularities.

Noted that by appropriate selections of the evolution time, one can generate a multiresolution representation of biomolecular surfaces. Such a multiresolution representation can be tuned to emphasize desirable features, namely, the binding sites of different scales, on a protein surface.

B High order geometric PDEs for the treatment of noisy EMDB data

The cryo-EM data in the EMDB suffer from the poor signal-to-noise ratio (SNR). Noise reduction is an indispensable process for the visualization, analysis and modeling of the cryo-EM data. Many elegant methods and techniques have been proposed or introduced, including wavelet transform techniques [58], bilateral filtering [59, 33, 42], and iterative median filtering [60], nonlinear anisotropic diffusion equations [27, 26] and Beltrami flow [24]. As high order geometric PDEs can suppress high-frequency components more efficiently [63], here we apply them to edge-preserving noise reduction of the EMD.

The arbitrarily high order nonlinear PDEs can be constructed using Fick's law [63]. Basically, we can design a nonlinear hyperflux,

$$\mathbf{j}_q = -d_q(u(\mathbf{r},t),|\nabla u(\mathbf{r},t)|,t)\nabla\nabla^{2q}u(\mathbf{r},t), q=0,1,2,\cdots \quad (85)$$

where $\mathbf{r}\in\mathbb{R}^n$, $\nabla=\frac{\partial}{\partial \mathbf{r}}$, $u(\mathbf{r}, t)$ can be viewed as the surface functions, and $d_q(u(\mathbf{r}, t), |\nabla u(\mathbf{r}, t)|, t)$ are edge sensitive diffusion coefficients. The high order diffusion PDEs can be expressed as

$$\frac{\partial u(\mathbf{r},t)}{\partial t} = -\sum_q \nabla\cdot\mathbf{j}_q + e(u(\mathbf{r},t),|\nabla u(\mathbf{r},t)|,t), q=0,1,2,\cdots \quad (86)$$

where $e(u(\mathbf{r}, t), |\nabla u(\mathbf{r}, t)|, t)$ is a nonlinear operator. It can be viewed as an external potential term and can be used to influence the behavior of the hyperflux. We can obtain the Perona-Malik equation by setting $q = 0$. The original data are used as an initial condition with suitable boundary conditions. The diffusion coefficients can be designed to be inhomogeneous or anisotropic, so as to remove the noise more efficiently while preserving the features. The Gaussian distribution is a widely used choice,

$$d_q(u(\mathbf{r},t),|\nabla u(\mathbf{r},t)|,t) = d_{q0}\exp\left[-\frac{|\nabla^q u|^2}{2\sigma_q^2}\right], \quad (87)$$

where d_{q0} reflects the noise amplitude, σ_0 and σ_1 are variance of u and ∇u in the neighborhood of a certain position $\mathbf{r}$,

$$\sigma_q^2(\mathbf{r}) = \overline{|\nabla^q u - \overline{\nabla^q u}|^2}(q=0,1). \quad (88)$$

The notation $\overline{Y(\mathbf{r})}$ stands for the average of $Y(\mathbf{r})$ in the neighborhood around position $\mathbf{r}$. High order nonlinear diffusion PDEs have been applied to many areas [63, 37, 31, 30].

Another option is the high order geometric PDEs [4],

$$\frac{\partial S}{\partial t}=(-1)^q\sqrt{g(|\nabla\nabla^{2q}S|)}\nabla\cdot\left(\frac{\nabla(\nabla^{2q}S)}{\sqrt{g(|\nabla\nabla^{2q}S|)}}\right)+V(S,|\nabla S|),q=0,1,2,\cdots \quad (89)$$

where $g(|\nabla\nabla^{2q}S|) = 1 + |\nabla\nabla^{2q}S|^2$ is the generalized Gram determinant. When $q = 0$, we have the potential driven mean curvature flow [4]

$$\frac{\partial S}{\partial t}=|\nabla S|\nabla\cdot\left(\frac{\nabla S}{|\nabla S|}\right)+V(S,\nabla S). \quad (90)$$

This equation is also related to the geometric flow equations derived from solvation analysis in Section A.2.

Computationally, due to the stiffness of high order nonlinear PDEs, the construction of suitable algorithms is an important issue. The common approach is alternating direction implicit (ADI) schemes [70, 4]. Through the splitting of the spatial dimensions, they managed to improve the efficiency and stability.

C Geometric features

Geometric features include surface area, volume, Gaussian curvature, mean curvature, shape index, and curvedness. The evaluation of surface area and surface enclosed volume is quite straightforward in the Lagrangian representation. However, in the Eulerian representation, their accurate evaluation is non-trivial. The details of the evaluation in the Eulerian representation have been described in Section 2.4.

The curvatures, on the contrary, involve more intrinsic or extrinsic information than area or volume. When a 2D surface is embedded in the 3D space, curvatures at each point measure how fast the normal direction changes from this point to nearby points. A normal plane at a certain point is spanned by the normal vector and a chosen tangent vector. This plane intersects with the surface along a planar curve, which has a unique curvature at the given point. Therefore, for this point, different curvatures can be obtained along different tangent directions. Among them, two principal curvatures, the maximum curvature and the minimum curvature are the maximum and the minimum values of these curvatures, respectively. The tangent directions related to principal curvatures are called principal directions. Note that these two tangent directions are always orthogonal to each other. From the point of view of eigenvalue decomposition of the shape operator matrix (Jacobian of the normal field), principal curvatures are just eigenvalues, and the principal directions are associated eigenvectors. The sign of the curvature can be defined as follows: bending towards the normal direction indicates negative curvature and bending away form the normal direction implies a positive curvature.

Gaussian curvature and mean curvature are popular shape descriptors as they are invariants under the rotation of the tangent plane. The Gaussian curvature is defined as the product of the two principal curvatures $\kappa_1\kappa_2$, while the mean curvature is defined as the average of the two principal curvatures $\frac{\kappa_1+\kappa_2}{2}$. Based on the signs of the Gaussian curvature and mean curvature, the local shape near a certain point can be classified into eight different surface types: pit, valley, saddle valley, flat, minimal surface, saddle ridge, ridge and peak. Table 8 demonstrates these different situations and summarizes the details of the definition. There is an intuitive way to understand this special classification — we locally approximate a surface in a specific coordinate system aligned to the principal directions and the normal direction (Darboux frame) as

$$z=\frac{1}{2}\begin{pmatrix} x & y \end{pmatrix}\begin{pmatrix} -\kappa_1 & 0 \\ 0 & -\kappa_2 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}$$

where the values of two principal curvatures κ_1 and κ_2 are chosen from the representative set $\{-1, -\frac{1}{2}, 0, \frac{1}{2}, 1\}$ in the sequence of ascending values. For instance, for the surface type of pit, their values are both -1.

Koenderink and van Doorn developed a single-value, angular measure to describe a local surface type instead of the principal curvatures [35]. It is called the shape index and defined as:

$$s=\frac{2}{\pi}\arctan\left(\frac{\kappa_1+\kappa_2}{\kappa_1-\kappa_2}\right).$$

The shape index describes the relation between the principal curvatures and gives an intuitive representation of the local surface shape. An important surface parameter complementary to the shape index is called curvedness, defined as:

$$c=\sqrt{\frac{\kappa_1^2+\kappa_2^2}{2}}.$$

These two surface indication parameters are also used in image processing, computer-aided design, and experimental data segmentation. In the present work, we introduce them to the biomolecular surface analysis.

D Matlab code for MRC data extraction

A Matlab code is given for extracting the voxel value information from the MRC (or CCP4) data. The output data are stored in ".dat" format for further use. The example here is for "EMD1048" with the input data "emd_.map" and output data "EMD1048.dat". A detail specification of the MRC file format can be found on the website http://ami.scripps.edu/software/mrctools/mrc_specification.php.

```
function a= readMRCfile(fname)
% readMRCfile readMRCfile (fname)
[fid,message]=fopen('emd_1048.map','r');
if fid == -1
error('can''t open file');
a= -1;
return;
end
nx=fread(fid,1,'long');
ny=fread(fid,1,'long');
nz=fread(fid,1,'long');
type= fread(fid,1,'long');
fprintf(1,'nx= %d ny= %d nz= %d type= %d', nx, ny,nz,type)
```

```
% Seek to start of data
status=fseek(fid,1024,-1);
% Shorts
if type== 1
a=fread(fid,nx*ny*nz,'int16');
end
%floats
if type == 2
a=fread(fid,nx*ny*nz,'float32');
end
fclose( fid);
a= reshape(a, [nx ny nz]);
if nz == 1
a= reshape(a, [nx ny]);
end
bb=reshape(a,[nx*ny*nz,1]);
save('EMD1048.dat','bb', '-ASCII');
```

References

1. Baker, NA. Biomolecular applications of Poisson-Boltzmann methods. In: Lipkowitz, KB.; Larter, R.; Cundari, TR., editors. Reviews in Computational Chemistry. Vol. volume 21. Hoboken, NJ: John Wiley and Sons; 2005.
2. Baker, NA.; Bashford, D.; Case, DA. Implicit solvent electrostatics in biomolecular simulation. In: Leimkuhler, B.; Chipot, C.; Elber, R.; Laaksonen, A.; Mark, A.; Schlick, T.; Schutte, C.; Skeel, R., editors. New Algorithms for Macromolecular Simulation. Springer; 2006.
3. Baker NA, Sept D, Joseph S, Holst MJ, McCammon JA. Electrostatics of nanosystems: Application to microtubules and the ribosome. Proceedings of the National Academy of Sciences of the United States of America. 2001; 98(18):10037–10041. [PubMed: 11517324]
4. Bates PW, Chen Z, Sun YH, Wei GW, Zhao S. Geometric and potential driving formation and evolution of biomolecular surfaces. J. Math. Biol. 2009; 59:193–231. [PubMed: 18941751]
5. Bates PW, Wei GW, Zhao S. The minimal molecular surface. arXiv:q-bio/0610038v1, [q-bio.BM]. 2006
6. Bates, PW.; Wei, GW.; Zhao, S. Midwest Quantitative Biology Conference. Mackinac Island, MI: Mission Point Resort; 2006 Sep-Oct. The minimal molecular surface.
7. Bates PW, Wei GW, Zhao S. Minimal molecular surfaces and their applications. Journal of Computational Chemistry. 2008; 29(3):380–391. [PubMed: 17591718]
8. Bertonati C, Honig B, Alexov E. Poisson-Boltzmann calculations of nonspecific salt effects on protein-protein binding free energy. Biophysical Journal. 2007; 92:1891–1899. [PubMed: 17208980]
9. Bobenko, AI.; Schröder, P. Discrete willmore flow; Symp. on Geometry Processing; 2005 Jul. p. 101-110.
10. Chen D, Chen Z, Chen C, Geng WH, Wei GW. MIBPB: A software package for electrostatic analysis. J. Comput. Chem. 2011; 32:657–670.
11. Chen Z, Baker NA, Wei GW. Differential geometry based solvation models I: Eulerian formulation. J. Comput. Phys. 2010; 229:8231–8258. [PubMed: 20938489]
12. Chen Z, Baker NA, Wei GW. Differential geometry based solvation models II: Lagrangian formulation. J. Math. Biol. 2011; 63:1139–1200. [PubMed: 21279359]
13. Chen Z, Wei GW. Differential geometry based solvation models III: Quantum formulation. J. Chem. Phys. 2011; 135(194108)
14. Chen Z, Zhao S, Chun J, Thomas DG, Baker NA, Bates PB, Wei GW. Variational approach for nonpolar solvation analysis. Journal of Chemical Physics. 2012; 137(084101)

15. Cheng LT, Dzubiella J, McCammon AJ, Li B. Application of the level-set method to the implicit solvation of nonpolar molecules. Journal of Chemical Physics. 2007; 127(8)
16. Cipriano G, Phillips GN Jr, Gleicher M. Multi-scale surface descriptors. IEEE Transactions on Visualization and Computer Graphics. 2009; 15:1201–1208. [PubMed: 19834190]
17. Connolly ML. Depth buffer algorithms for molecular modeling. J. Mol. Graphics. 1985; 3:19–24.
18. Corey RB, Pauling L. Molecular models of amino acids, peptides and proteins. Rev. Sci. Instr. 1953; 24:621–627.
19. Droske M, Rumpf M. A level set formulation for willmore flow. Interfaces and Free Boundaries. 2004; 6(3):361–378.
20. Eisenhaber F, Argos P. Improved strategy in analytic surface calculation for molecular systems: Handling of singularities and computational efficiency. J. Comput. Chem. 1993; 14:1272–1280.
21. Federer H. Curvature Measures. Trans. Amer. Math. Soc. 1959; 93:418–491.
22. Feng X, Xia K, Tong Y, Wei G-W. Geometric modeling of subcellular structures, organelles and large multiprotein complexes. International Journal for Numerical Methods in Biomedical Engineering. 2012; 28:1198–1223. [PubMed: 23212797]
23. Feng X, Xia KL, Chen Z, Tong YY, Wei GW. Multiscale geometric modeling of macromolecules II: Lagrangian representation. Journal of Computational Chemistry. 2013 in press.
24. Fernandez J. Tomobflow: feature-preserving noise filtering for electron tomography. BMC Bioinformatics. 2009; 178:1–10.
25. Fernandez J, Li S, Lucic V. Three-dimensional anisotropic noise reduction with automated parameter tuning: Application to electron cryotomography. Current Topics in Artificial Intelligence. 2007; volume 4788:60–69.
26. Fernandez S, Li JJ. An improved algorithm for anisotropic nonlinear diffusion for denoising cryo-tomograms. J. Struct. Biol. 2003; 144:152–161. [PubMed: 14643218]
27. Frangakis AS, Hegerl R. Noise reduction in electron tomographic reconstructions using nonlinear anisotropic diffusion. J. Struct. Biol. 2001; 135:239–250. [PubMed: 11722164]
28. Geng W, Wei GW. Multiscale molecular dynamics using the matched interface and boundary method. J Comput. Phys. 2011; 230(2):435–457. [PubMed: 21088761]
29. Geng W, Yu S, Wei GW. Treatment of charge singularities in implicit solvent models. Journal of Chemical Physics. 2007; 127:114106. [PubMed: 17887827]
30. Gilboa G, Sochen N, Zeevi YY. Forward-and-backward diffusion processes for adaptive image enhancement and denoising. IEEE Transactions on Image Processing. 2002; 11(7):689–703. [PubMed: 18244666]
31. Gilboa G, Sochen N, Zeevi YY. Image sharpening by flows based on triple well potentials. Journal of Mathematical Imaging and Vision. 2004; 20(1–2):121–131.
32. Gogonea V, Osawa EE. Implementation of solvent effect in molecular mechanics. 1. model development and analytical algorithm for the solvent -accessible surface area. Supramol. Chem. 1994; 3:303–317.
33. Jiang W, Baker ML, Wu Q, Bajaj C, Chiu W. Applications of a bilateral denoising filter in biological electron microscopy. J. Struct. Biol. 2003; 144:114–122. [PubMed: 14643214]
34. Kindlmann G, Whitaker R, Tasdizen T, Möller T. Curvature-based transfer functions for direct volume rendering: methods and applications. Proc. IEEE Visualization. 2003
35. Koenderink JJ, van Doorn AJ. Surface shape and curvature scales. Image and Vision Computing. 1992 Oct.10(8):557–564.
36. Lee B, Richards FM. The interpretation of protein structures: estimation of static accessibility. J Mol Biol. 1971; 55(3):379–400. [PubMed: 5551392]
37. Lysaker M, Lundervold A, Tai XC. Noise removal using fourth-order partial differential equation with application to medical magnetic resonance images in space and time. IEEE Transactions on Image Processing. 2003; 12(12):1579–1590. [PubMed: 18244712]
38. Marenich AV, Cramer CJ, Truhlar DG. Perspective on foundations of solvation modeling: The electrostatic contribution to the free energy of solvation. Journal of Chemical Theory and Computation. 2008; 4(6):877–887.

39. Mumford D, Shah J. Optimal approximations by piecewise smooth functions and associated variational problems. Communications on Pure and Applied Mathematics. 1989; 42(5):577–685.
40. Osher S, Fedkiw RP. Level set methods: An overview and some recent results. J. Comput. Phys. 2001; 169(2):463–502.
41. Osher S, Sethian J. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. Journal of computational physics. 1988; 79(1):12–49.
42. Pantelic RS, Rothnagel CY, Huang R, Muller D, Woolford D, Landsberg MJ, McDowall A, Pailthorpe B, Young PR, Banks J, Hankamer B, Ericksson G. The discriminative bilateral filter: An enhanced denoising filter for electron microscopy data. J. Struct. Biol. 2006; 155:395–408. [PubMed: 16774838]
43. Perona P, Malik J. Scale-space and edge-detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1990; 12(7):629–639.
44. Pierotti RA. A scaled particle theory of aqueous and nonaqeous solutions. Chemical Reviews. 1976; 76(6):717–726.
45. Prabhu NV, Zhu P, Sharp KA. Implementation and testing of stable, fast implicit solvation in molecular dynamics using the smooth-permittivity finite difference Poisson-Boltzmann method. Journal of Computational Chemistry. 2004; 25(16):2049–2064. [PubMed: 15481091]
46. Richards FM. Areas, volumes, packing, and protein structure. Annual Review of Biophysics and Bioengineering. 1977; 6(1):151–176.
47. Rocchia W, Sridharan S, Nicholls A, Alexov E, Chiabrera A, Honig B. Rapid grid-based construction of the molecular surface and the use of induced surface charge to calculate reaction field energies: Applications to the molecular systems and geometric objects. Journal of Computational Chemistry. 2002; 23:128–137. [PubMed: 11913378]
48. Roux B, Simonson T. Implicit solvent models. Biophysical Chemistry. 1999; 78(1–2):1–20. [PubMed: 17030302]
49. Rudin, LI.; Osher, S.; Fatemi, E. Nonlinear total variation based noise removal algorithms. Proceedings of the eleventh annual international conference of the Center for Nonlinear Studies on Experimental mathematics : computational issues in nonlinear science; Elsevier North-Holland, Inc; Amsterdam, The Netherlands, The Netherlands. 1992. p. 259p. 268
50. Sanner MF, Olson AJ, Spehner JC. Reduced surface: An efficient way to compute molecular surfaces. Biopolymers. 1996; 38:305–320. [PubMed: 8906967]
51. Sethian JA. Evolution, implementation, and application of level set and fast marching methods for advancing fronts. J. Comput. Phys. 2001; 169(2):503–555.
52. Sharp KA, Honig B. Electrostatic interactions in macromolecules - theory and applications. Annual Review of Biophysics and Biophysical Chemistry. 1990; 19:301–332.
53. Simonett G. The willmore flow for near spheres. Differential and Integral Equations. 2001; 14:1005–1014.
54. Simonson T. Macromolecular electrostatics: continuum models and their growing pains. Current Opinion in Structural Biology. 2001; 11(2):243–252. [PubMed: 11297935]
55. Sochen N, Kimmel R, Malladi R. A general framework for low level vision. Image Processing, IEEE Transactions on. 1998; 7(3):310–318.
56. Soldea O, Elber G, Rivlin E. Global segmentation and curvature analysis of volumetric data sets using trivariate b-spline functions. IEEE Trans. on PAMI. 2006; 28(2):265–278.
57. Stillinger FH. Structure in aqueous solutions of nonpolar solutes from the standpoint of scaled-particle theory. J. Solution Chem. 1973; 2:141–158.
58. Stoschek A, Hegerl R. Denoising of electron tomographic reconstructions using multiscale transformations. J. Struct. Biol. 1997; 120:257–265. [PubMed: 9441931]
59. Tomasi C, Manduchi R. Bilateral filtering for gray and color images. Proc. ICCV. 1998; 98:839–846.
60. van der Heide P, Xu XP, Marsh BJ, Hanein D, Volkmann N. Efficient automatic noise reduction of electron tomographic reconstructions based on iterative median filtering. J. Struct. Biol. 2007; 158:196–204. [PubMed: 17224280]

61. Wang Y, Wei GW, Yang S-Y. Partial differential equation transform – Variational formulation and Fourier analysis. International Journal for Numerical Methods in Biomedical Engineering. 2011; 27:1996–2020. [PubMed: 22207904]
62. Wang Y, Wei GW, Yang S-Y. Mode decomposition evolution equations. Journal of Scientific Computing. 2012; 50:495–518. [PubMed: 22408289]
63. Wei GW. Generalized Perona-Malik equation for image restoration. IEEE Signal Processing Lett. 1999; 6:165–167.
64. Wei GW. Differential geometry based multiscale models. Bulletin of Mathematical Biology. 2010; 72:1562–1622. [PubMed: 20169418]
65. Wei G-W. Multiscale multiphysics and multidomain models I: Basic theory. Journal of Theoretical and Computational Chemistry. 2013; 12(8):1341006.
66. Wei GW, Jia YQ. Synchronization-based image edge detection. Europhysics Letters. 2002; 59(6): 814–819.
67. Wei GW, Sun YH, Zhou YC, Feig M. Molecular multiresolution surfaces. arXiv:math-ph/0511001v1. 2005:1–11.
68. Wei G-W, Zheng Q, Chen Z, Xia K. Variational multiscale models for charge transport. SIAM Review. 2012; 54(4):699–754. [PubMed: 23172978]
69. Willmore, TJ. Riemannian Geometry. USA: Oxford University Press; 1997.
70. Witelski TP, Bowen M. ADI schemes for higher-order nonlinear diffusion equations. Applied Numerical Mathematics. 2003; 45(2–3):331–351.
71. Witkin A. Scale-space filtering: A new approach to multi-scale description. Proceedings of IEEE International Conference on Acoustic Speech Signal Processing. 1984; 9:150–153.
72. Xia KL, Wei GW. Three-dimensional MIB galerkin method for elliptic interface problems. submitted to Journal of Computational Physics. 2013
73. Xia KL, Zhan M, Wei GW. MIB galerkin method for elliptic interface problems. Journal of Computational Physics. 2013
74. Xie D, Jiang Y, Brune P, Scott LR. A fast solver for a nonlocal dielectric continuum model. SIAM Journal on Scientific Computing. 2012; 34:B107–B126.
75. Xu G, Pan Q, Bajaj CL. Discrete surface modeling using partial differential equations. Computer Aided Geometric Design. 2006; 23(2):125–145. [PubMed: 19830268]
76. Yu SN, Geng WH, Wei GW. Treatment of geometric singularities in implicit solvent models. Journal of Chemical Physics. 2007; 126:244108. [PubMed: 17614538]
77. Yu SN, Wei GW. Three-dimensional matched interface and boundary (MIB) method for treating geometric singularities. J. Comput. Phys. 2007; 227:602–632.
78. Yu ZY, Holst M, Cheng Y, McCammon JA. Feature-preserving adaptive mesh generation for molecular shape modeling and simulation. Journal of Molecular Graphics and Modeling. 2008; 26:1370–1380.
79. Zhang Y, Bajaj C, Xu G. Surface smoothing and quality improvement of quadrilateral/hexahedral meshes with geometric flow. Communications in Numerical Methods in Engineering. 2009; 25:1–18. [PubMed: 19829757]
80. Zhao S. High order matched interface and boundary methods for the helmholtz equation in media with arbitrarily curved interfaces. J. Comput. Phys. 2010; 229:3155–3170.
81. Zheng Q, Wei GW. Poisson-Boltzmann-Nernst-Planck model. Journal of Chemical Physics. 2011; 134:194101. [PubMed: 21599038]
82. Zheng Q, Yang SY, Wei GW. Molecular surface generation using PDE transform. International Journal for Numerical Methods in Biomedical Engineering. 2012; 28:291–316. [PubMed: 22582140]
83. Zhou YC, Wei GW. On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method. J. Comput. Phys. 2006; 219(1):228–246.
84. Zhou YC, Zhao S, Feig M, Wei GW. High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources. J. Comput. Phys. 2006; 213(1):1–30.

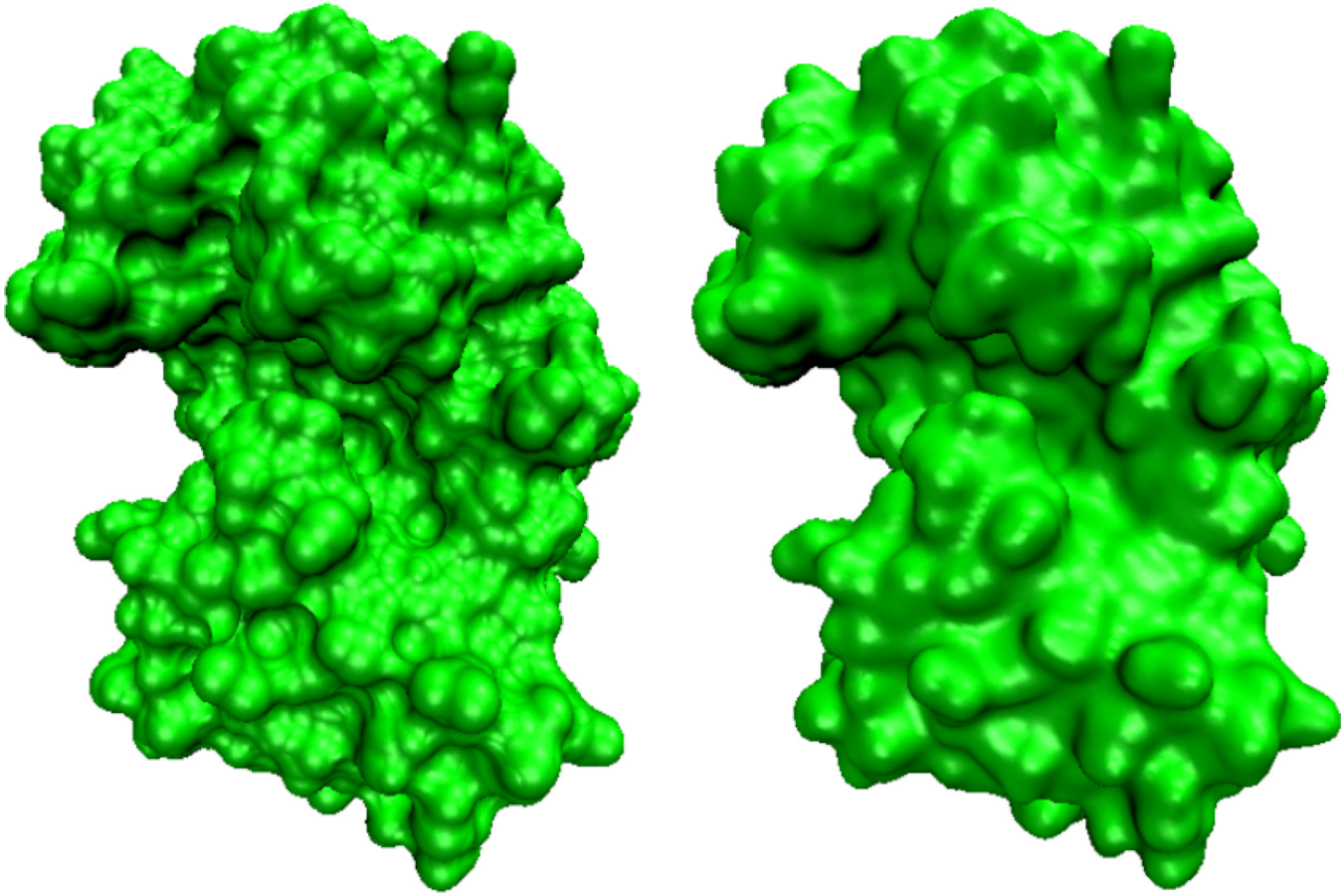


Figure 1.
Comparison of the solvent excluded surface (Left) of protein 1PPL and surface generated by the Laplace-Beltrami flow with $V_1 = 0$ (Right). The latter is free from geometric singularity. In the generation of 1PPL MMS, an outer layer of 1.7Å is used to immerse the protein in solvent. The comptational domain for protein 1PPL is [−14.7, 57.8]*[−16.7,41.3]*[−8.2,39.8]. We set the grid size to be 0.5 Å, and 100 iterations are carried out.

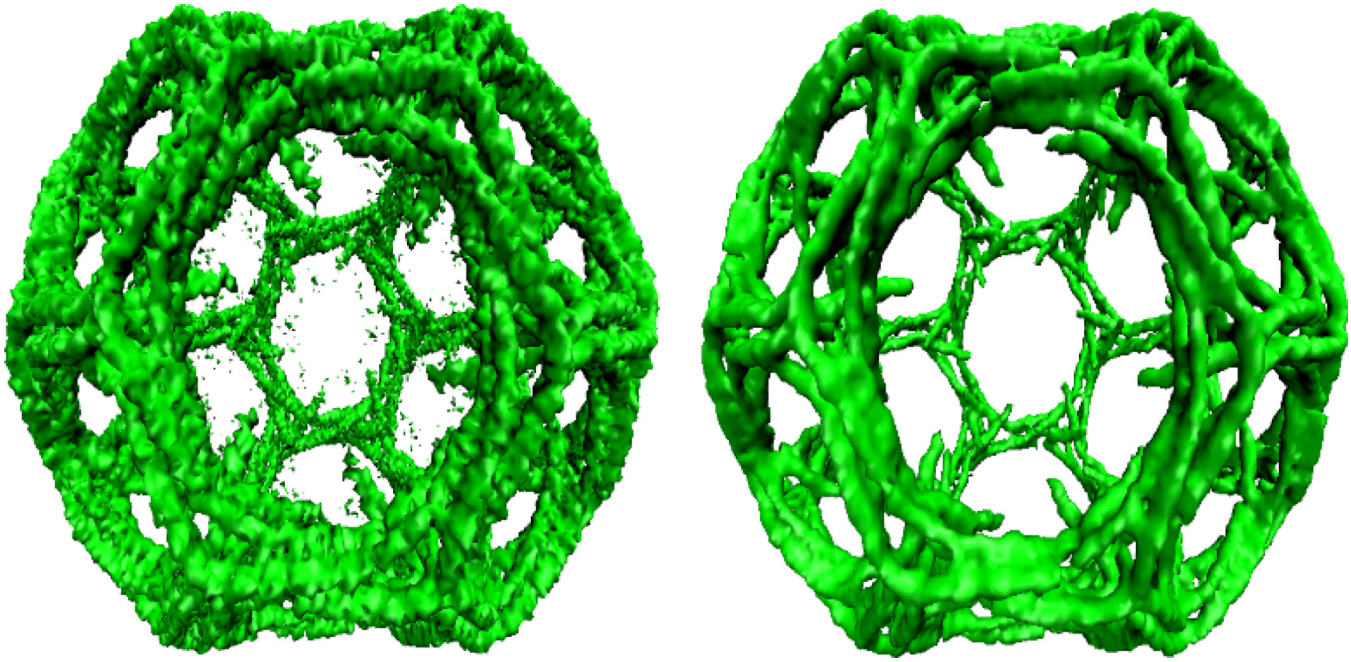


Figure 2.
Noise reduction for emd5119. The left chart shows the noisy image from the original density maps; The right chart shows the image free from the noise.

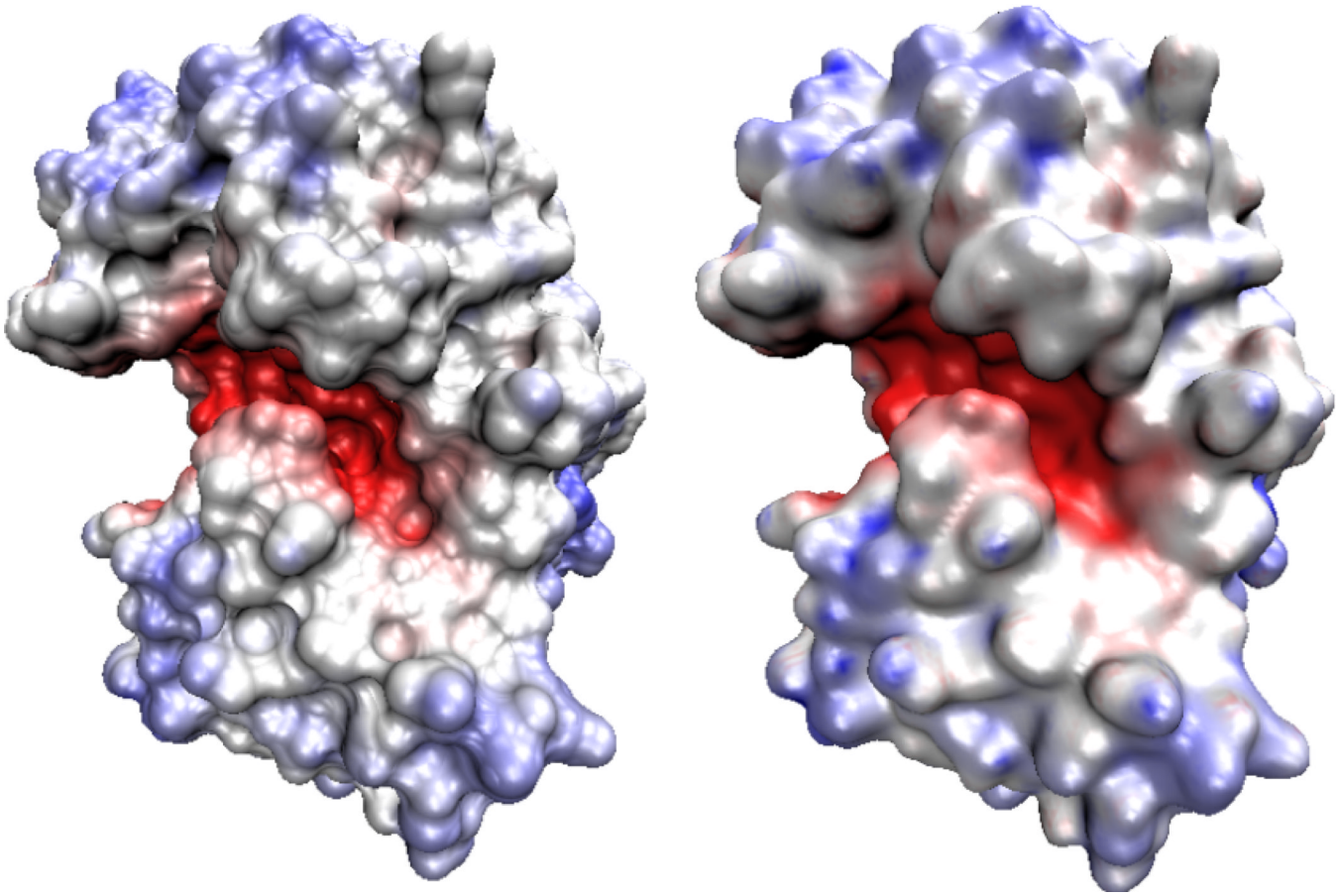


Figure 3.
Comparison of electrostatic potential distributions on a molecular surface (Left) and a surface generated from a generalized Laplace Beltrami equation (Right) for protein 1PPL.

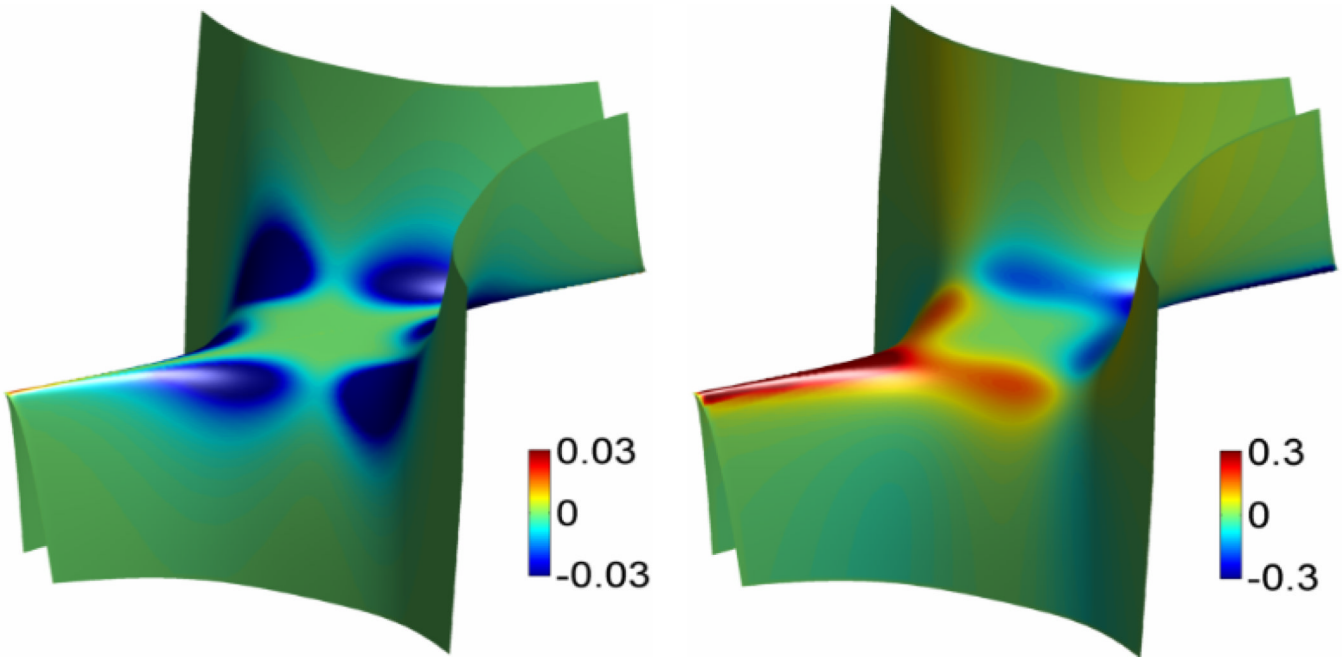


Figure 4.
Computational results of Gaussian curvature and mean curvature for test Case 1.

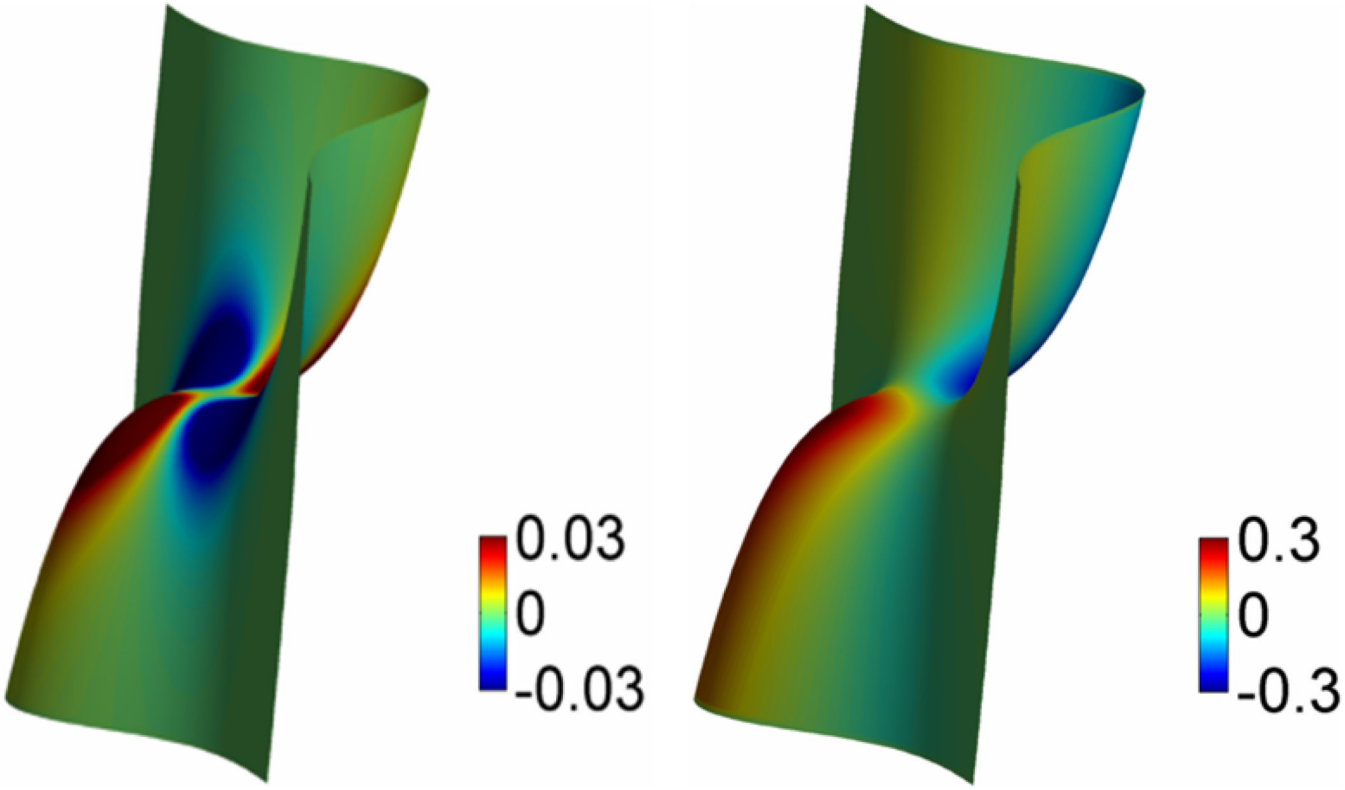


Figure 5.
Computational results of Gaussian curvature and mean curvature for test Case 2.

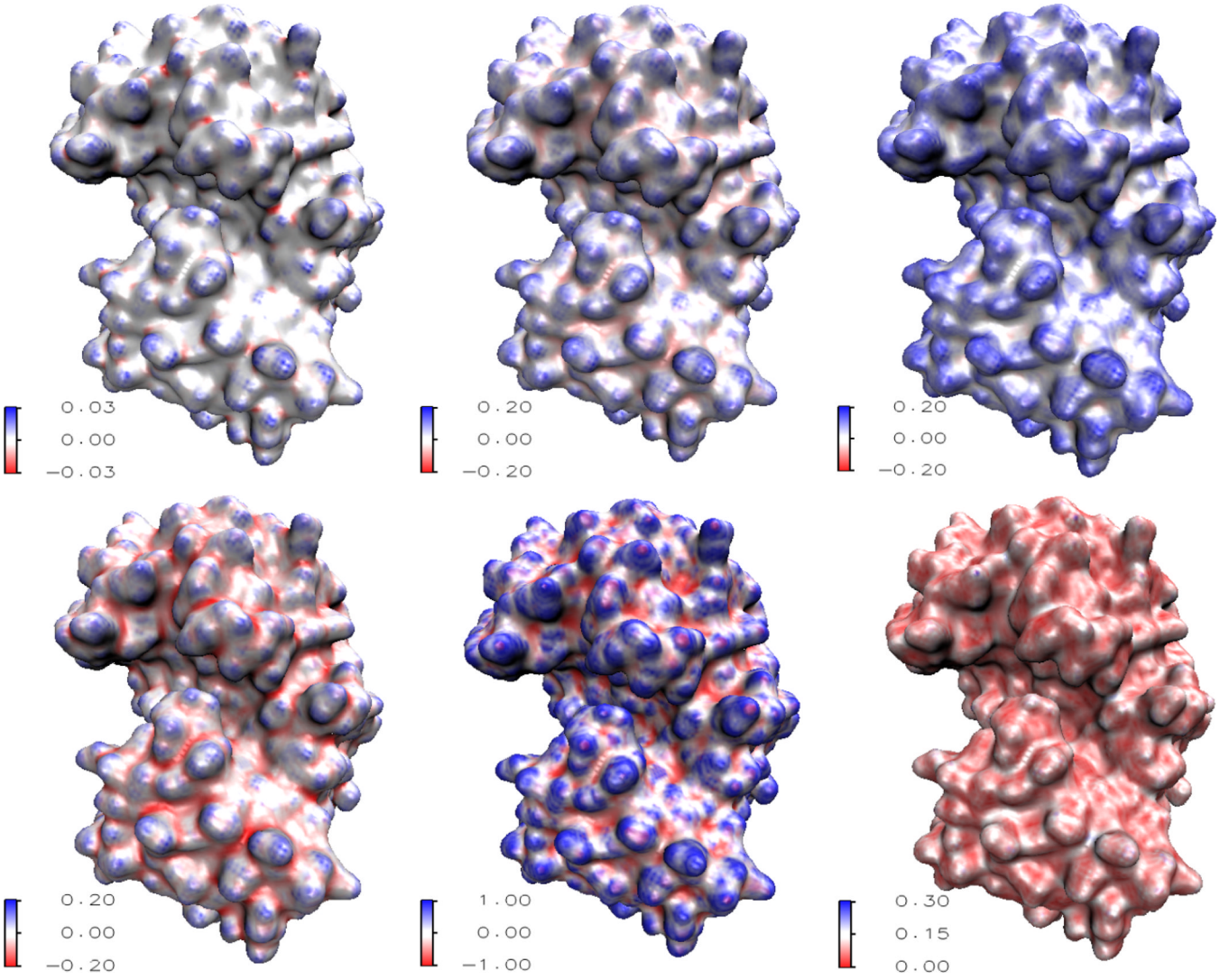


Figure 6.
Curvature analysis of Protein 1PPL. From top left to bottom right: Gaussian curvature, mean curvature, maximum curvature, minimum curvature, shape index, and curvedness.

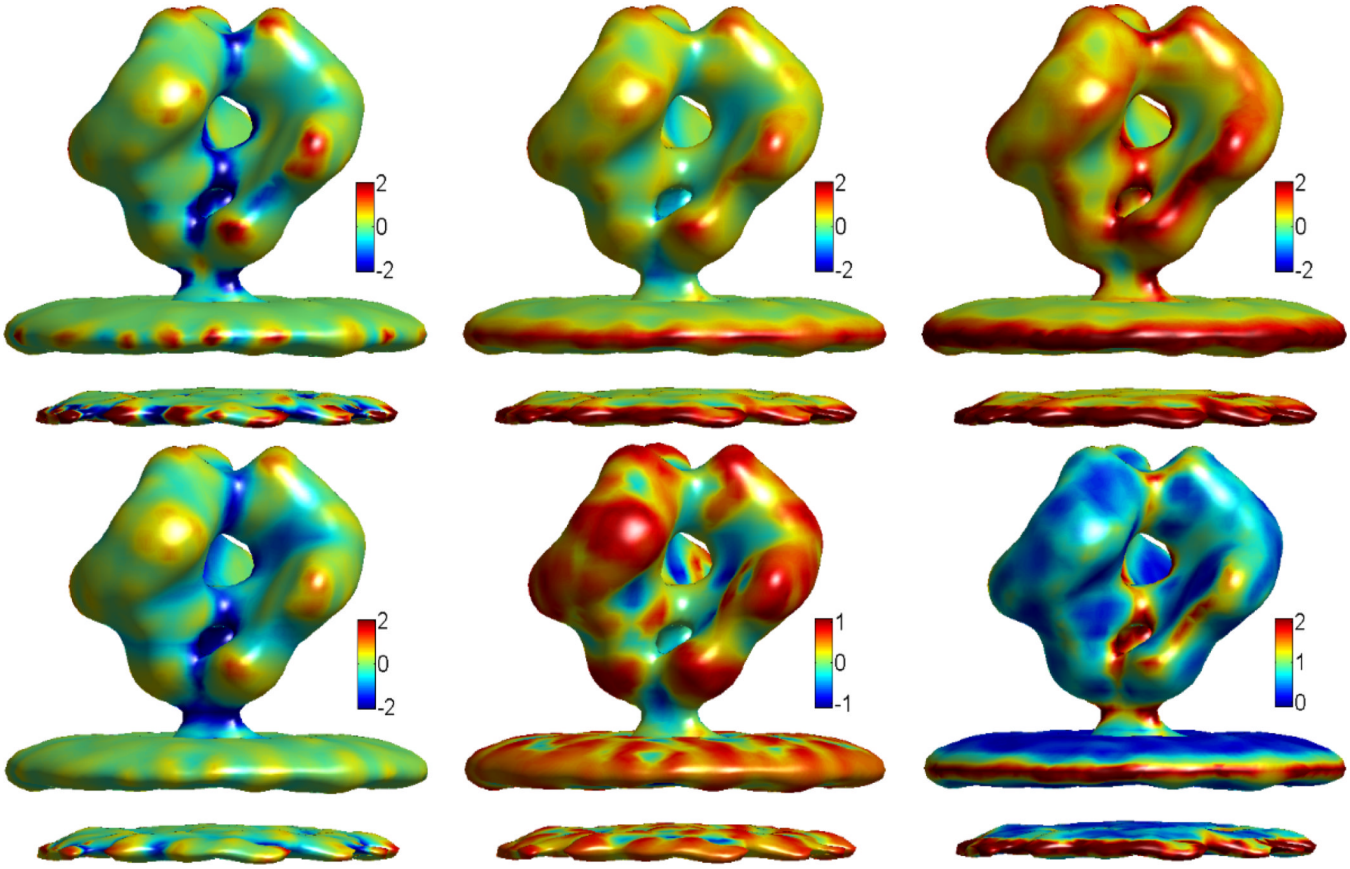


Figure 7.
EMD5273 curvature properties. From top left to bottom right: Gaussian curvature, mean curvature, maximum curvature, minimum curvature, shape index, and curvedness.

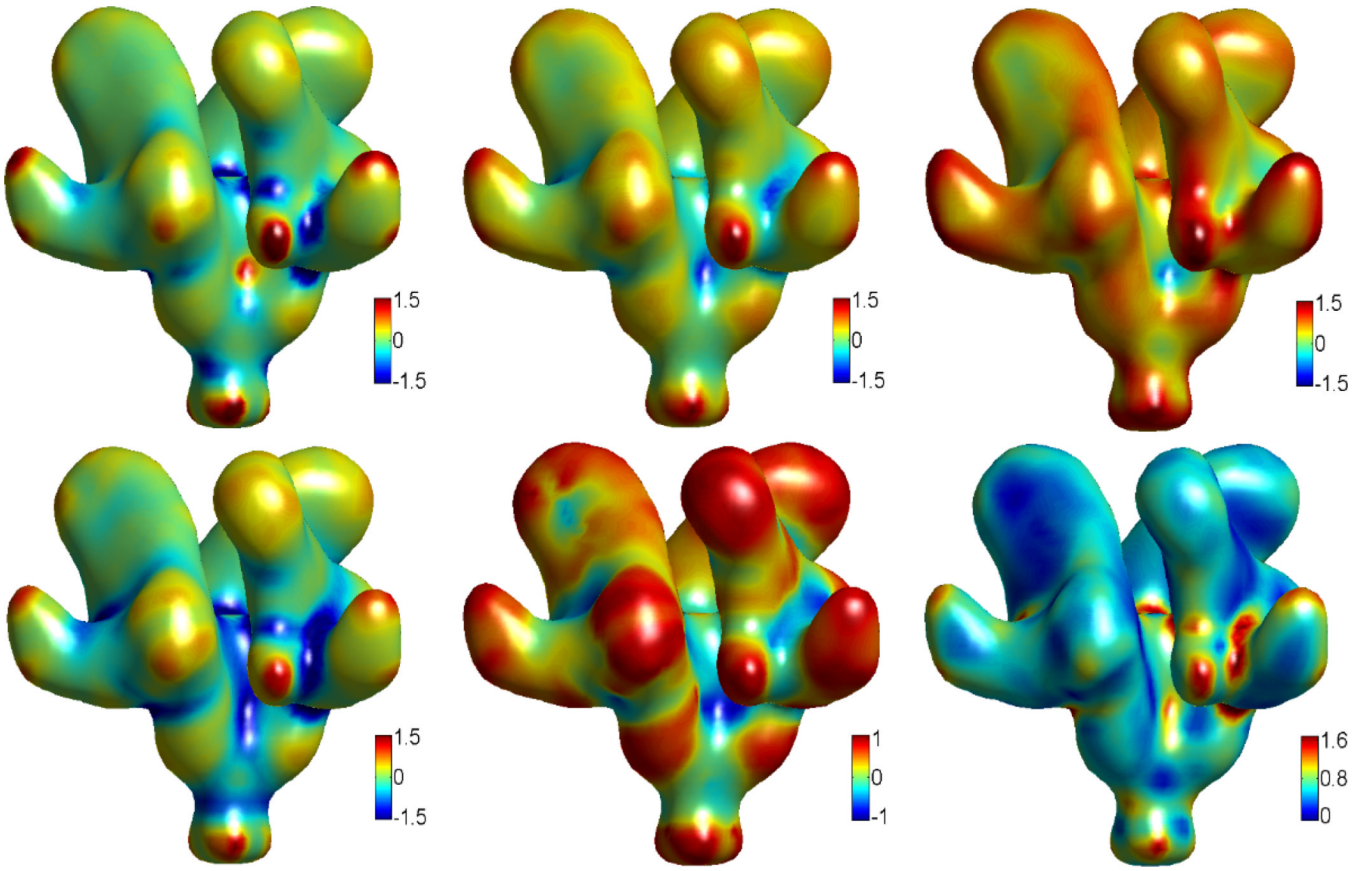


Figure 8.
EMD5020 curvature properties. From top left to bottom right: Gaussian curvature, mean curvature, maximum curvature, minimum curvature, shape index, and curvedness.

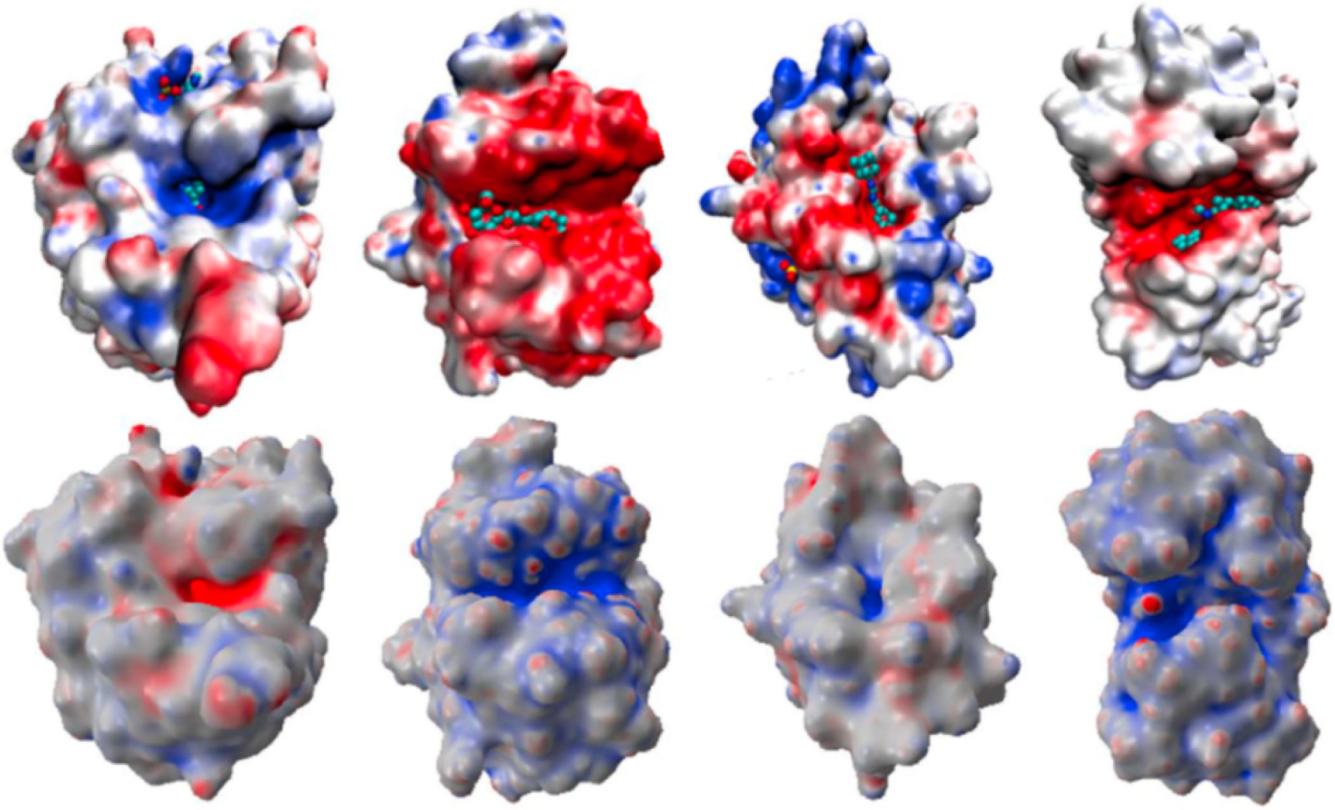


Figure 9.
Polarized curvature based binding site prediction for four proteins (Left to right:1ADS, 1BYH, 1EJN, 2WEB). Top row: Protein-ligand complexes displayed with electrostatic surface potential; Bottom row: Polarized curvature maps ($\Phi \times \kappa_2$) indicating the binding sites.

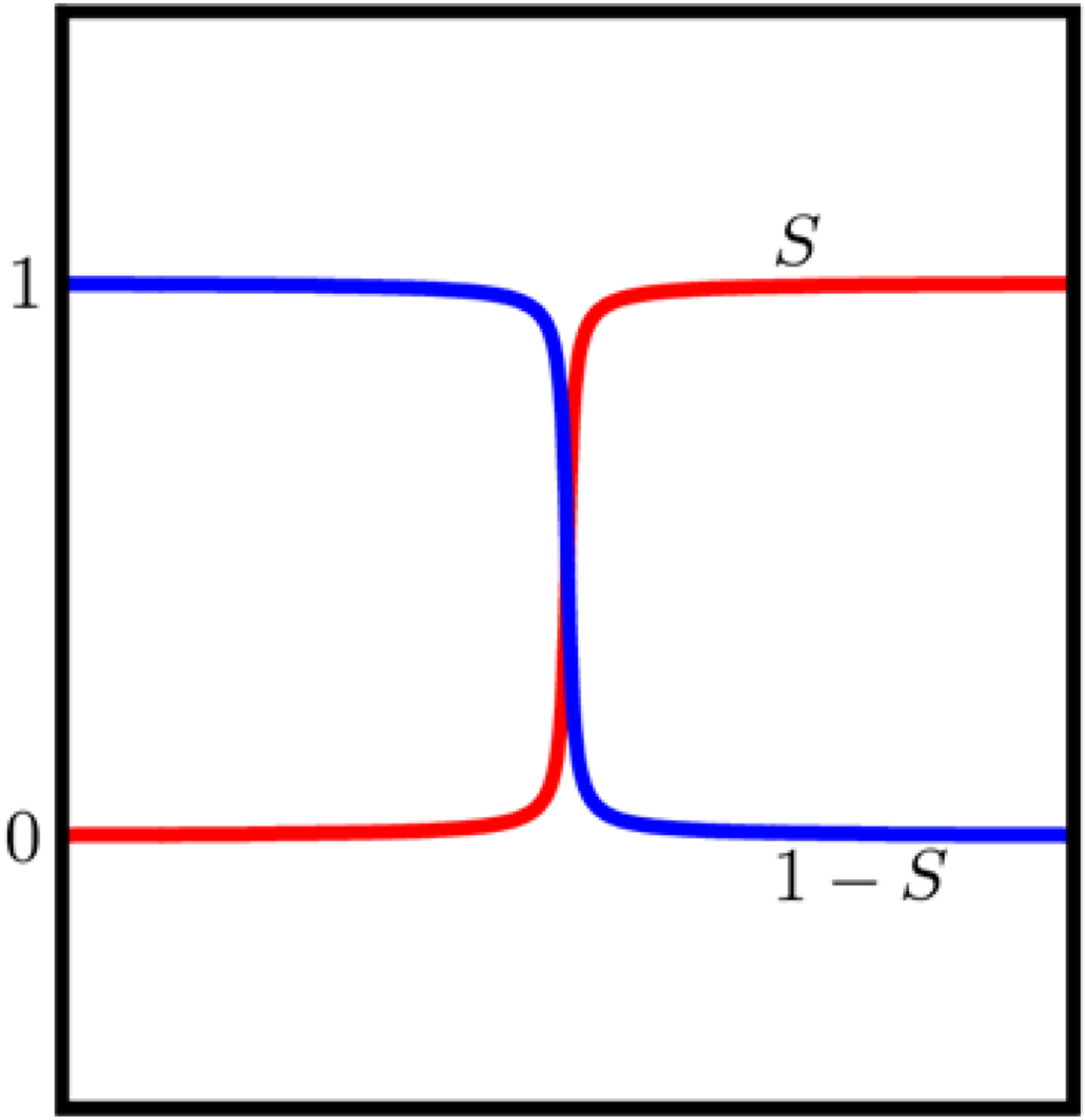


Figure 10.
Characteristic functions S and $1 - S$ projected in 1D.

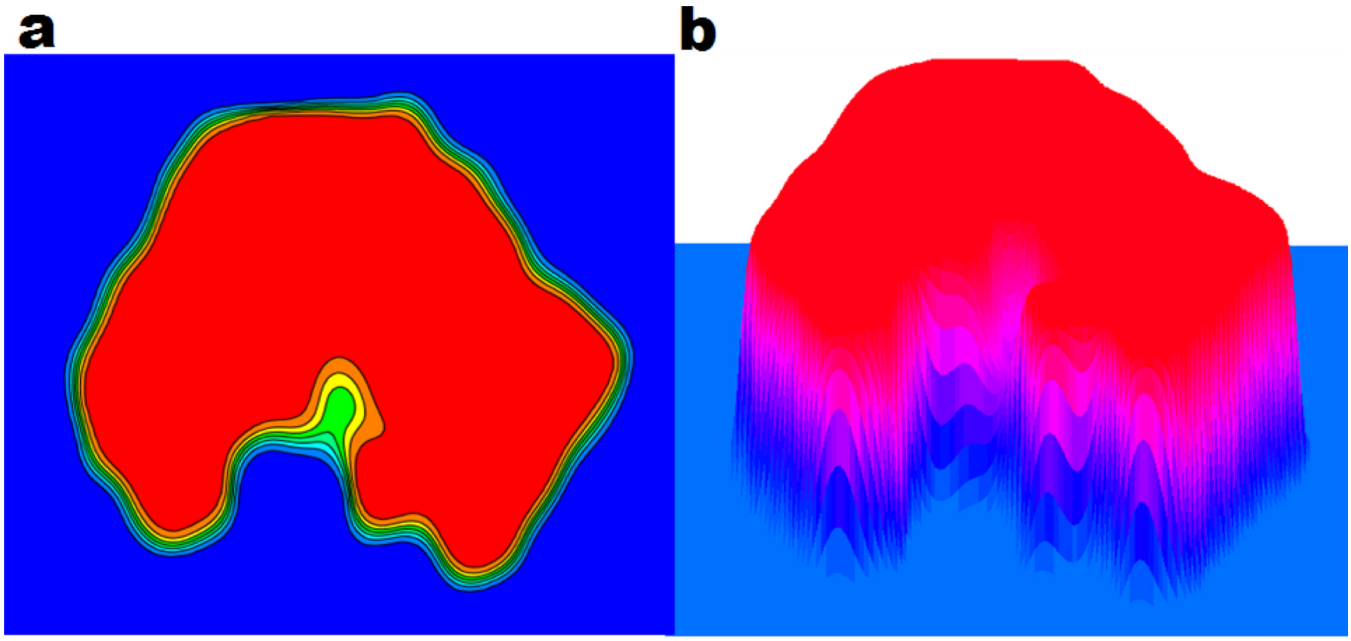


Figure 11.
A contour plot (a) and a mesh plot (b) of a cross section of the characteristic function *S* for protein 1PPL. The blue color and red color represent value 0 and 1, respectively in (a). All contour lines between 0 and 1 can be chosen to extract isosurfaces.

Table 1

Test of the convergence of the proposed method for Lagrangian to Eulerian transformation.

h	Error	Order
2.500e-1	7.985e-4	
1.250e-1	2.651e-4	1.59
6.250e-2	7.265e-5	1.87
3.125e-2	1.979e-5	1.88

Table 2

Test of the convergence of the proposed method for the surface area of a sharp interface in the Cartesian representation.

$N_x \times N_y$	Error	Order
21× 21	3.391	
41× 41	9.390e-1	1.85
81× 81	2.026e-1	2.21
161× 161	5.498e-2	1.88

Table 3

Numerical errors and convergence orders for calculating Gaussian curvature (Case 1).

$n_x \times n_y$	L_∞	Order	L_2	Order
21 × 21	1.483e-1		1.543e-2	
41 × 41	7.049e-2	1.07	4.280e-3	1.85
81 × 81	2.028e-2	1.80	8.494e-4	2.33
161 × 161	5.348e-3	1.92	1.816e-4	2.23

Table 4

Numerical errors and convergence orders for calculating mean curvature (Case 1).

$n_x \times n_y$	L_∞	Order	L_2	Order
21 × 21	4.498e-1		3.735e-2	
41 × 41	4.057e-2	3.47	2.667e-3	3.81
81 × 81	2.231e-3	4.18	5.262e-4	2.34
161 × 161	6.893e-4	1.69	1.343e-4	1.97

Table 5

Numerical errors and convergence orders for Gaussian curvature (Case 2).

$n_x \times n_y$	L_∞	Order	L_2	Order
11 × 11	2.682e-2		6.295e-3	
21 × 21	7.268e-3	1.88	1.420e-3	2.15
41 × 41	1.846e-3	1.98	3.528e-4	2.01
81 × 81	4.927e-4	1.91	8.751e-5	2.01

Table 6

Numerical errors and convergence orders for calculating mean curvature (Case 2).

$n_x \times n_y$	L_∞	Order	L_2	Order
11 × 11	4.451e-2		1.009e-2	
21 × 21	1.146e-2	1.96	2.415e-3	2.06
41 × 41	2.923e-3	1.97	5.942e-4	2.02
81 × 81	7.604e-4	1.94	1.470e-4	2.02

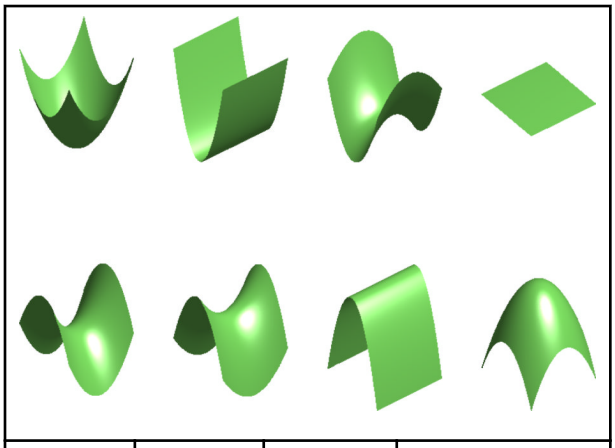
Table 7

Polarized curvatures as binding indicators of protein surfaces. Maximal curvature (κ_1), minimal curvature (κ_2), positive electrostatic surface potential (Φ^+) and negative electrostatic surface potential (Φ^-) are combined to indicate potential binding sites.

	$\kappa_1 > 0$	$\kappa_2 < 0$
$\Phi^+ > 0$	site for negatively charged protein	site for negatively charged small ligand
$\Phi^- < 0$	site for positively charged protein	site for positively charged small ligand

Table 8

The figure gives surface types classified according to Gaussian curvature (K) and mean curvature (H). From top left to bottom right: pit, valley, saddle valley, flat, minimal surface, saddle ridge, ridge and peak, respectively. The table lists surface types based on signs of Gaussian curvature (K) and mean curvature (H).



	$K > 0$	$K = 0$	$K < 0$
$H > 0$	peak	ridge	saddle ridge
$H = 0$	none	flat	minimal surface
$H < 0$	pit	valley	saddle valley