A Simple Surface Representation Scheme for Rigid Body Docking

Vipin K. Tripathi, Bhaskar Dasgupta, Kalyanmoy Deb, Hari K. Voruganti

Department of Mechanical Engineering, Indian Institute of Technology Kanpur, India-208016
vipinkt,dasgupta,deb,hari@iitk.ac.in

ABSTRACT

Reconstructing a topologically consistent and geometrically accurate model of surface from observed data is an important problem in geometric reverse engineering. Molecular docking is a new area which has an application of geometric reverse engineering in surface representation of protein molecules. In rigid body docking, molecules are considered as rigid bodies. In this paper, a new approach which is simple and efficient is proposed for modeling the surface patches of protein molecules. The approach aims at representing the huge data of molecular surface patches with a few control points. In the process, an inverse problem in B-spline surface fitting is solved using Cholesky decomposition or singular value decomposition of an input data matrix. The input to the scheme is molecular surface point data arranged on topologically rectangular grid. The results of testing the scheme on different types of surface point data sets prove its effectiveness.

Keywords: surface modeling, docking, geometric reverse engineering, b-splines, cholesky decomposition

1 Introduction

Molecular docking is a problem of maximizing the molecular interactions. Computational molecular docking refers to the algorithms which attempt to obtain the best binding conformation of two interacting molecules [1]. Docking is computationally difficult because there are many alternate binding locations and orientations in which two molecules can bind with each other. Information of the best binding conformation is useful in many applications like rational drug design, cellular pathways, macromolecular assemblies, protein folding, etc. Docking of two molecules is shown in Fig. 1.

A common classification divides docking methods into two classes, rigid body docking and flexible ligand docking. Rigid body docking methods consider molecules as rigid bodies. A rigid body docking procedure has three elements: (i) representation of the molecular shape, (ii) conformational space search and (iii) ranking of potential solutions. Shape representation is the most important aspect of docking because the other two elements depend upon the method used for molecular shape representation.

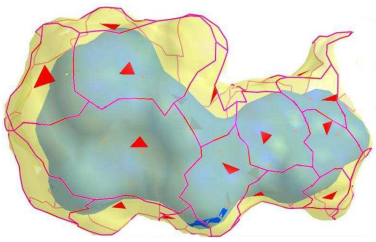

Figure 1: Docking of two molecules

For molecular shape representation, a geometric approach initiated by Kuntz [2] and further elaborated and rationalized by Connolly [3] is normally adopted. This approach considers molecular surface as a network of connected patches of different types such as convex, concave, flat, etc. as shown in Fig. 1. The patches of two different molecular surfaces are matched with each other at different points and in different orientations to obtain the best binding conformation under certain scoring criterion [4].

This paper describes a simple and efficient method for modeling of surface patches of the molecules. The method is based upon solving an inverse problem in B-spline surface fitting. Despite a number of approaches tried in the literature [5,6,7,8,9], solving the inverse problem remains a difficult and computationally expensive problem. A simplified formulation is developed in this study to solve the inverse problem which makes use of Cholesky decomposition or singular value decomposition of an input data matrix [10]. The input to the scheme is surface point data arranged on a topologically rectangular grid.

2 Surface Representation

For modeling of surface patches of the molecule, B-spline surface representation technique is used. B-spline surface is defined in terms of a characteristic polyhedron as shown in Fig. 2. The shape of the surface approximates the polyhedron [11]. The advantages of this surface are the local control of surface and independence of the degree of surface from number of control points [12].

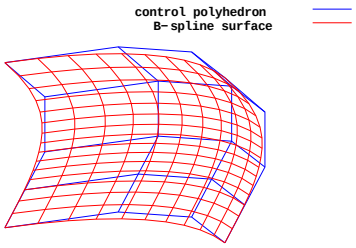

Figure 2: B-spline surface

A tensor product B-spline surface is formulated as

$$\mathbf{Q}(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} \mathbf{P}_{ij} N_{i,k}(u) N_{j,l}(v) \qquad (1)$$

where

$\mathbf{P}_{ij}$ = $(n+1)\times(m+1)$ vertices of defining the polyhedron,

$N_{i,k}(u)$, $N_{j,l}(v)$ = Blending functions calculated recursively by Equations 2& 3.

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } t_i \le u < t_{i+1}, \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

$$N_{i,k}(u) = \frac{(u-t_i)N_{i,k-1}(u)}{t_{i+k-1}-t_i} + \frac{(t_{i+k}-u)N_{i+1,k-1}(u)}{t_{i+k}-t_{i+1}} \qquad (3)$$

where t_i = Knot values calculated by Eq. 4. For an open curve

$$t_i = \begin{cases} 0 & \text{if } i < k, \\ i-k+1 & \text{if } k \le i \le n, \\ n-k+2 & \text{if } i > n. \end{cases} \qquad (4)$$

where k = order of curve. Index i varies as $0 \le i \le n+k$ and the range of parametric variable u is $0 \le u \le n-k+2$.

Finding $\mathbf{Q}$, the surface data points, by an input of $(n+1)\times(m+1)$ number of control points which are vertices of polyhedron $\mathbf{P}_{ij}$, is known as the 'forward problem'.

3 The Inverse Problem

For modeling of surface patches, it is the inverse problem which is important, that is, given a known set of data points on a surface, determine the defining polyhedron for B-spline surface that best approximates the data. This, being inverse of the forward problem explained earlier, is known as the 'inverse problem'. Solving an inverse problem is useful as it yields a very small

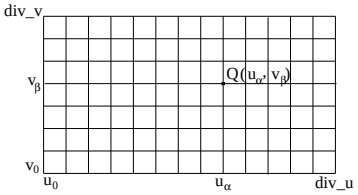

Figure 3: Surface data points on a rectangular grid

number of control points as compared to the given data points which are easy to store and process. B-spline surface formulation needs data points to be parameterized on a rectangular grid as shown in Fig. 3.

Writing out Eq. 1 for a single surface data point, say, point $\mathbf{Q}(u_\alpha, v_\beta)$ as shown in Fig. 3 yields

$$\mathbf{Q}(u_\alpha,v_\beta) = \begin{bmatrix} N_{0,k}(u_\alpha) & \cdots & N_{n,k}(u_\alpha) \end{bmatrix} \begin{bmatrix} \mathbf{P}_{ij} \end{bmatrix} \begin{bmatrix} N_{0,l}(v_\beta) \\ N_{1,l}(v_\beta) \\ \vdots \\ N_{m,l}(v_\beta) \end{bmatrix}. \qquad (5)$$

Writing an equation of this form for each data point on the rectangular grid yields a system of simultaneous equations. In the matrix form the result can be written as

$$[\mathbf{Q}] = [\mathbf{C^t}][\mathbf{P_{ij}}][\mathbf{D}] \qquad (6)$$

where $\mathbf{Q}$ is the matrix of data points. It has $(div\_u+1)\times(div\_v+1)$ elements in it. $\mathbf{C^T}$ is the matrix of row vectors on the right hand side of Eq. 5. It contains $(div\_u+1)\times(n+1)$ elements. $\mathbf{P}$ is a matrix of control points. It contains $(n+1)\times(m+1)$ elements. $\mathbf{D}$ is the matrix of column vectors on the right hand side of Eq. 5. It contains $(m+1)\times(div\_v+1)$ elements.

Eq. 6 is simplified to the standard form,

$$\mathbf{q} = [\mathbf{A}]\mathbf{p} \qquad (7)$$

where

$\mathbf{q}$ =Column vector of data points of dimension (div_u+1)*(div_v+1),1,

$\mathbf{p}$ =Column vector of control points of dimension (n+1)*(m+1),1,

[$\mathbf{A}$] =Matrix of dimension (div_u+1)*(div_v+1),(n+1)*(m+1).

Matrix [$\mathbf{A}$] is formed by the rule derived in this study as follows,

$$\mathbf{A}_{I,J} = d_{J,I}[\mathbf{C^t}] \qquad (8)$$

where
$\mathbf{A}_{(I,J)}$ $=(I,J)-th$ block of $\mathbf{A}$ matrix,
$\mathrm{d}_{(J,I)}$ $=(J,I)-th$ element of $\mathbf{D}$ matrix.

Now the inverse problem can be stated as: How to solve system $\{\mathbf{q}\}=[\mathbf{A}]\{\mathbf{p}\}$ for unknown $\mathbf{p}$ when $\mathbf{q}$ is given?

Solution procedure used here makes use of Cholesky decomposition based method to solve Eq. 7. Cholesky's method is a popular method of solving system of linear equations like $\mathbf{Ap}=\mathbf{q}$. If it works in a given case, it is about a factor of two faster than the alternative methods available for solving linear systems [15]. As Cholesky decomposition [13] works for a square, symmetric and positive definite matrix and the surface point data available in molecular docking problem will give a highly over-specified matrix $\mathbf{A}$, system $[\mathbf{A^T A}]\mathbf{p}=[\mathbf{A^T}]\mathbf{q}$ is solved instead of the system $\mathbf{Ap}=\mathbf{q}$. Matrix $[\mathbf{A^T A}]$ on the left hand side is a square matrix then.

If Cholesky decomposition fails due to matrix $[\mathbf{A^T A}]$ not being positive definite, a method based on singular value decomposition [16] is used which is applicable to a general rectangular $M \times N$ matrix.

4 Proposed Algorithm

Step 1: Input parameters n, m, k, l, div_u, div_v as explained earlier.

Step 2: Calculate knot vectors by Eq. 4.

Step 3: Calculate blending function $N_{i,k}(u)$ and $N_{j,l}(v)$ by Eqs. 2 and 3.

Step 4: Form matrices $[\mathbf{C^t}]$ and $[\mathbf{D}]$.

Step 5: Form matrix $[\mathbf{A}]$, from matrices $[\mathbf{C^t}]$ and $[\mathbf{D}]$.

Step 6: Input $\mathbf{q}$ vector.

Step 7: Form matrix $[\mathbf{A^T A}]$.

Step 8: Use Cholesky decomposition to solve system $[\mathbf{A^T A}]\mathbf{p}=[\mathbf{A^T}]\mathbf{q}$.

Step 9: If Cholesky decomposition fails, use singular value decomposition of matrix $\mathbf{A}$ to solve $|\ \mathbf{Ap}-\mathbf{q}\ |$ in least square sense.

5 Results and Discussion

Numerical implementation of the algorithm has been done in C programming language. Surface representation results are displayed in *gnuplot* in *linux* environment. For Cholesky decomposition and singular value decomposition the routines from GNU scientific library [14] have been used. Results of testing the algorithm on five different data sets are shown in the Figures 4 through 8. The values of parameters taken in the procedure are:

div_u = 20,	div_v = 30
n = 3,	m = 4
k=4,	l=4

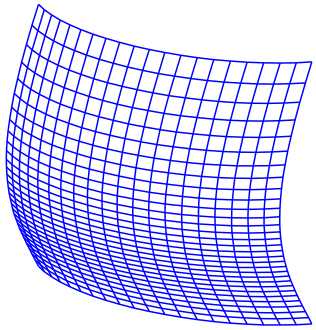

Figure 4: Surface patch 1

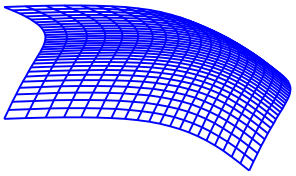

Figure 5: Surface patch 2

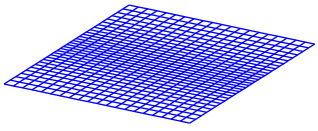

Figure 6: Surface patch 3

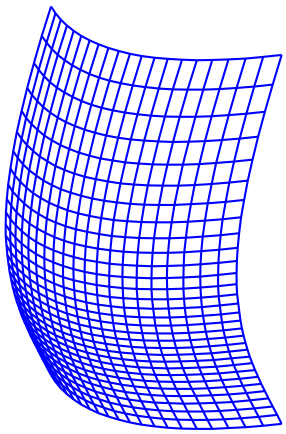

Figure 7: Surface patch 4

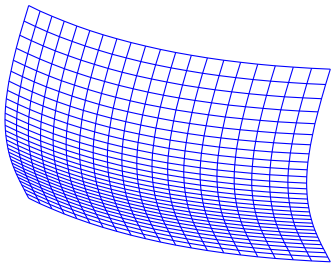

Figure 8: Surface patch 5

6 Conclusion

Testing the proposed scheme on different data sets justify its effectiveness for modeling the variety of surface patches. A bicubic B-spline surface with sufficiently small number of control points gives an optimal surface from the analysis, convergence and fairness point of view. Surface can be generated and manipulated easily using the control points. A Cholesky decomposition based technique is more efficient to solve the inverse problem while the singular value decomposition based method is more general in nature.

REFERENCES

[1] I. Halperin, B. Ma, H. Wolfson, and R. Nussinov, "Principles of docking: An overview of search algorithms and a guide to scoring functions," Proteins, 47, 409–433, 2002.

[2] I. Kuntz, J. Blaney, S. Oatley, R. Langridge and T. Ferrin, "A geometric approach to macromolecule-ligand interactions," J. of Mol. Biol., 161, 269–288, 1982.

[3] M. Connolly, "Analytical molecular surface calculation," J. of Appl. Cryst., 16, 548–558, 1983.

[4] R. Norel, D. Petrey, H. Wolfson, and R. Nussinov, "Examination of shape complementarity in docking of unbound proteins," Proteins, 35, 403–419, 1999.

[5] R.Frank and G. M. Nielson, "Scattered Data Interpolation and applications: A Tutorial and Survey," Geometric Modelling: Methods and Their Applications, 131–160, 1991.

[6] E. Arge, M. Daehlen and A. Tveito, "Approximation of Scattered Data Using Smooth Grid Functions," J. of Comput. and Appl. Maths., 59, 191–205, 1995.

[7] M. Eck and H. Hoppe, "Automatic Reconstruction of B-spline Surface of Arbitrary Topological Type," Proc., SIGGRAPH'96, Computer Graphics Proc., Ann. Conf. Series, 325–334, 1996.

[8] B. Baran and M. S. Celebi, "A Surface Reconstruction Approach for Scattered Data Using Gaussian Sphere," IJCI Proc. of Int. Conf. on Signal Processing,ISSN 1304-2386, 1, 2003.

[9] V. K. Tripathi, B. Dasgupta, and K. Deb, "An approach for surface generation from measured data points using B-splines," Int Conf on Systemics, Cybernetics and Informatics, Hyderabad, India, 578–581, 2005.

[10] V. K. Tripathi, "Optimization in shape matching in the context of molecular docking," Ph.D. Thesis, Dept. of Mechanical Engg., Indian Institute of Technology Kanpur, India, 2006.

[11] M. E. Mortenson, "Geometric Modeling," John Wiley & Sons, 1985.

[12] D. F. Rogers and J. A. Adams, "Mathematical Elements for Computer Graphics," McGraw-Hill Publishing Company, 1990.

[13] Erwin Kreyszig, "Advanced Engineering Mathematics," John Wiley and Sons, 2002.

[14] M. Galassi, J. Davies, J. Theiler, and F. Rossi, "GNU scientific library reference manual," 2003.

[15] W. M. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, "Numerical recipes: Art of scientific computing," Cambridge University Press, 1986.

[16] D. C. Lay, "Linear Algebra and Its Applications," Reading, MA: Addison-Wesley, 1996.