Molecular Surface Decomposition using Graphical Modeling

Maharavo Randrianarivony[1], Guido Brunnett[2]

[1]Institut für Informatik, Christian-Albrechts-Universität Kiel
[2]Fakultät für Informatik, Technische Universität Chemnitz
maharavo@informatik.tu-chemnitz.de

Abstract. We aim at decomposing molecular surfaces which are in the form of Van der Waals model into a structure adapted for mesh-free modeling. We focus on the generation of spherical patches by using rational Bézier surfaces and homogeneous coordinates. We can achieve exact global continuity by using generalized stereographic projection. We support the theoretical descriptions by practical decompositions of a few molecular surfaces.

1 Introduction

A lot of efforts have been done for mesh-based methods [1] in chemical modeling [2]. In contrast, mesh-free molecular structure is not yet well developed. Having patch representation is advantageous for both visualization and simulation tasks. Among others, OpenGL and ACIS have built-in functions for treating rational spline patches by providing only the required parameters as arguments. Further, NURBS primitives are available in many rendering softwares like POV-RAY. Additionally, generating a multiresolution setting is immediate when a four-sided splitting as we present below is available. Multiresolution structure is well known of being efficient for both fast graphical tasks and numerical simulations. Hence, we want to contribute in the development of algorithms for treatments of mesh-free molecular geometries. Each atom in a molecule is represented as an imaginary sphere whose radius corresponds to the Van der Walls radius [2, 3]. Let us denote by $\mathcal{B}(\omega, \rho)$ the ball of center ω and radius ρ. Consider N balls $\mathcal{B}_k := \mathcal{B}(\Omega_k, \rho_k)$ whose union is connected. The molecular surface is described by $S := \partial \left[\bigcup_{k=1}^{N} \mathcal{B}_k \right]$ which represents a closed surface. We want to tessellate S into m four-sided patches F_i such that the splitting $S = \bigcup_{i=1}^{m} F_i$ is conforming: every two nondisjoint patches F_i and F_j share either a complete edge or a single corner. Our goal is to create functions γ_i such that $F_i = \gamma_i([0,1]^2)$. Additionally, we require global continuity meaning that for two adjacent patches F_i and F_j, there is an affine mapping A such that $\gamma_i(s) = \gamma_j(A(s))$ for all $s \in \partial[0,1]^2$ (Fig. 1).

Relevant works are as follows. From a list of intersecting spheres, one generates the B-rep structures as in [1]. Generation of trimmed surfaces has been treated in [4] with application in mesh. For mesh-free methods, we have used

transfinite interpolations in [5] but the mappings are not necessarily globally continuous.

2 Materials and Methods

The description of the molecular surface S as in section 1 is good for CSG where the only geometric primitives are spheres. Since B-Rep structure with parametrizations is more convenient for the surface decompositions, we first convert that CSG representation into B-rep structure. Since the method of doing that is almost standard, we do not present it here. After that conversion, we have trimmed surfaces $\Gamma_k = \sigma_k(\mathcal{D}_k)$ where $\mathcal{D}_k$ is a multiply connected planar domain and $\sigma_k : \mathcal{D}_k \subset \mathbf{R}^2 \longrightarrow \mathcal{B}_k$ represents the stereographic projection with respect to $\mathcal{B}_k$. Since we have now a set of parametric trimmed surfaces [6], we can split them into four-sided patches. The detail of such a decomposition task can be found in [5]. Hence, we focus on the creation of the mappings from the unit square to the spherical patches while achieving global continuity. Although most methods here are described for the unit sphere $\mathbf{U}$, they can be generalized to any sphere. An element of the projective space $\mathbf{E}^3$ with homogeneous coordinates $[w : x : y : z]$ will have the cartesian coordinates $(x/w, y/w, z/w)$. We will need $\mathbf{P} := \{[w : x : y : 0] \in \mathbf{E}^3 \text{ with } w, x, y \in \mathbf{R}\}$ which corresponds to the plane $z = 0$ in cartesian coordinates. The creation γ_i uses the next projections from [3]. First, the hyperbolic projection ϑ transforms a point $[e_0 : e_1 : e_2 : e_3]$ to a point in $\mathbf{P}$ given by $\vartheta(\mathbf{e}) := \left[e_0^2 + e_3^2 : e_0 e_1 - e_2 e_3 : e_1 e_3 + e_0 e_2 : 0\right]$. Second, the stereographic projection is defined from $\mathbf{P}$ to $\mathbf{U}$ by $\sigma(\mathbf{e}) := \left[e_0^2 + e_1^2 + e_2^2 : 2e_0 e_1 : 2e_0 e_2 : e_1^2 + e_2^2 - e_0^2\right]$. Finally, the generalized stereographic projection is $\delta = \sigma \circ \vartheta$. For a point $\mathbf{q} = [q_0 : q_1 : q_2 : q_3]$, we define $\bar{\mathbf{q}} := [q_2 : q_3 : -q_0 : -q_1]$ and $\mathbf{q}^\perp := [-q_3 : q_2 : -q_1 : -q_0]$. Our approach is based on the fact that the image by δ of a rational Bézier [7] of degree n is a rational Bézier of degree $2n$ which is drawn on $\mathbf{U}$. Note that the converse holds true [3].
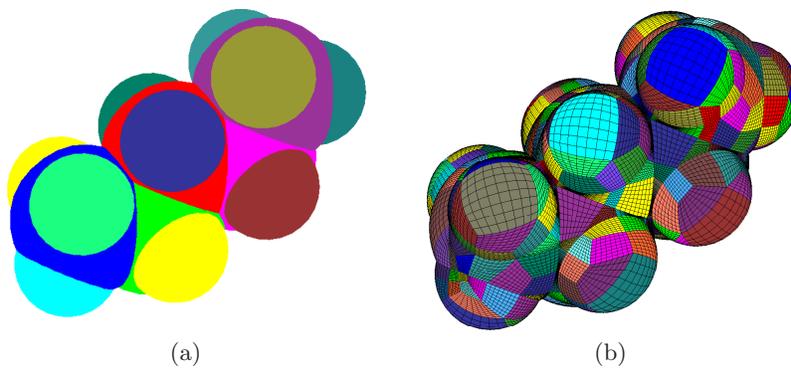

(a)   (b)

Fig. 1. Molecular surface (a) and decomposition into foursided patches (b)

2.1 Circular boundary parametrization

We want to parametrize a circular arc $\mathcal{C}$ with fixed endpoints as an image of δ. There are already plenty of methods for representing a circular arc as rational Bézier but we explicitly need that it is an image of δ because we intend to obtain global continuity. We suppose that we have four equispaced points $\mathbf{Q}_i$ belonging to $\mathcal{C}$ which is contained in the unit sphere $\mathbf{U}$. We are given also two points $\mathbf{b}_0 \in \mathbf{E}^3$ and $\mathbf{b}_2 \in \mathbf{E}^3$ which are transformed by δ to the first and the last points: $\delta(\mathbf{b}_0) = \mathbf{Q}_0$ and $\delta(\mathbf{b}_2) = \mathbf{Q}_3$. We search a conic $\mathbf{x}$ which maps by δ to the circle $\mathcal{C}$. In other words, we need the internal node $\mathbf{b}_1 \in \mathbf{E}^3$ such that $\mathcal{C}$ is the image of the quadratic Bézier curve $\mathbf{x}(t) = \mathbf{b}_0 B_0^2(t) + \mathbf{b}_1 B_1^2(t) + \mathbf{b}_2 B_2^2(t)$ with $\mathbf{b}_i = [w_i : x_i : y_i : z_i]$. The internal points $\mathbf{Q}_1$ and $\mathbf{Q}_2$ should correspond to the parameter values $t_1 := 1/3$ and $t_2 := 2/3$ as $\delta[\mathbf{x}(t_i)] = \mathbf{Q}_i$. Toward that end, we choose a preimage $\mathbf{q}_i$ of $\mathbf{Q}_i$ by δ. Since $\delta^{-1}(\mathbf{Q}_i) = \{\lambda \mathbf{q}_i + \mu \mathbf{q}_i^\perp : \lambda, \mu \in \mathbf{R}\}$, we have $< \bar{\mathbf{q}}_i, \mathbf{x}(t_i) >= 0$ and $< \bar{\mathbf{q}}_i^\perp, \mathbf{x}(t_i) >= 0$ which imply:

$$< \bar{\mathbf{q}}_i, \mathbf{b}_1 >= K(\bar{\mathbf{q}}_i, t_i) \qquad < \bar{\mathbf{q}}_i^\perp, \mathbf{b}_1 >= K(\bar{\mathbf{q}}_i^\perp, t_i) \qquad \text{for} \quad i = 1, 2 \qquad (1)$$

where $K(\mathbf{q}, t) := -\frac{1}{B_1^2(t)} \left[< \mathbf{q}, \mathbf{b}_0 > B_0^2(t) + < \mathbf{q}, \mathbf{b}_2 > B_2^2(t) \right]$. The four homogeneous coordinates of $\mathbf{b}_1$ are the solution of the linear system given by (1).

2.2 Spherical transfinite interpolation

Let us consider four circular arcs C_k represented as rational Bézier curves which are drawn on the unit sphere $\mathbf{U}$ and which enclose a four-sided spherical surface $\mathcal{H}$ such that $C_k(t) = \sum_{i=0}^{2n} \mathbf{c}_i^k B_i^{2n}(t)$ with $\mathbf{c}_i^k = [w_i^k : x_i^k : y_i^k : z_i^k]$. We assume that C_k are the images of four Bézier curves D_k of degree n by δ. That is, we have $C_k(t) = \delta[D_k(t)]$ with $D_k(t) = \sum_{i=0}^{n} \mathbf{d}_i^k B_i^n(t)$ such that there is coincidence at the corners as $\mathbf{d}_0^1 = \mathbf{d}_0^4$, $\mathbf{d}_n^1 = \mathbf{d}_0^2$, $\mathbf{d}_n^2 = \mathbf{d}_n^3$, $\mathbf{d}_0^3 = \mathbf{d}_n^4$.

What we are searching is not simply a rational Bézier surface which interpolates those curves but a spherical transfinite interpolant $\mathbf{X}$ residing on $\mathbf{U}$:

$$
\begin{aligned}
\mathbf{X}(u, 0) &= C_1(u), \qquad \mathbf{X}(u, 1) = C_3(u) \qquad \forall u \in [0, 1] \\
\mathbf{X}(0, v) &= C_4(v), \qquad \mathbf{X}(1, v) = C_2(v) \qquad \forall v \in [0, 1] \\
\mathbf{X}(u, v) &\in \mathbf{U} \qquad \forall (u, v) \in [0, 1]^2
\end{aligned}
\qquad (2)
$$

The desired patch $\mathbf{X}$ will be represented as a rational Bézier surface $\mathbf{X}(u, v) = \sum_{i=0}^{2n} \sum_{j=0}^{2n} \mathbf{b}_{ij} B_i^{2n}(u) B_j^{2n}(v)$ with $\mathbf{b}_{ij} = [w_{ij} : x_{ij} : y_{ij} : z_{ij}]$. That problem amounts to searching for some rational Bézier $\mathbf{Y}$ of degree n which maps by δ to $\mathbf{X}$ as $\mathbf{X}(u, v) = \delta[\mathbf{Y}(u, v)]$ with $\mathbf{Y}(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{n} \mathbf{a}_{ij} B_i^n(u) B_j^n(v)$. On account of the boundary conditions (2), the problem is reduced to the determination of the internal homogeneous control points $\mathbf{a}_{ij}$ of $\mathbf{Y}$. They are found by specifying that the image rational Bézier $\mathbf{X}$ interpolates some given internal points $Q_k \in \mathcal{H} \subset \mathbf{U}$ at $(u_k, v_k) \in [0, 1]^2$: $\mathbf{X}(u_k, v_k) = \delta[\mathbf{Y}(u_k, v_k)] = Q_k$ for all $k = 1, ..., m := 2(n-1)^2$. Let $\mathbf{q}_k$ be a preimage of Q_k by the mapping δ. Since we want to achieve $\mathbf{Y}(u_k, v_k) = \mathbf{q}_k$ for all $k = 1, ..., m$, we have two equalities: $\langle \bar{\mathbf{q}}_k, \mathbf{Y}(u_k, v_k) \rangle = 0$, $\langle \bar{\mathbf{q}}_k^\perp, \mathbf{Y}(u_k, v_k) \rangle = 0$, $\forall k = 1, ..., m$. By denoting the

set of indices (i, j) for internal and boundary control points of $\mathbf{Y}$ by $\mathcal{J}$ and $\mathcal{B}$ respectively, we obtain

$$\sum_{(i,j)\in\mathcal{J}} \langle \bar{\mathbf{q}}_k, \mathbf{a}_{ij} \rangle B_i^n(u_k) B_j^n(v_k) = -\langle \bar{\mathbf{q}}_k, \sum_{(i,j)\in\mathcal{B}} \mathbf{a}_{ij} B_i^n(u_k) B_j^n(v_k) \rangle \qquad (3)$$

Similar computations yield relations for $\bar{\mathbf{q}}_k^\perp$. Since $\mathbf{a}_{ij}$ for $(i, j) \in \mathcal{B}$ are known, the right hand side of (3) is completely specified. Therefore, this leads to some linear system of equations having $\mathbf{a}_{ij}$ with $(i, j) \in \mathcal{J}$ as unknowns.

3 Results

We have implemented routines in C/C++ and OpenGL in order to decompose some molecular surfaces. The numerical results can be found in Table 1 where the input was taken from PDB files. Note that the runtime measurement has been performed on a machine with processor Intel Core 2.16GHz running Windows Vista. For a triangle having vertices $T := [\mathbf{x}_p, \mathbf{x}_q, \mathbf{x}_r]$ belonging to the sphere $\mathcal{B}(\Omega_k, \rho_k)$, we introduce $\varepsilon(T) := \left| \text{dist} \left(\frac{1}{3}[\mathbf{x}_p + \mathbf{x}_q + \mathbf{x}_r], \Omega_k \right) - \rho_k \right|$. The criterion that quantifies the surface invariability of a mesh $\mathcal{M}$ is $\varepsilon(\mathcal{M}) := \frac{1}{\text{card}(\mathcal{M})} \sum_{T\in\mathcal{M}} \varepsilon(T)$. The patches which are obtained by the mappings from the unit squares are illustrated in Fig. 2.

4 Discussion

The main advantage of our mesh-free method is that it requires few patches in comparison to mesh-based methods. For a surface invariance $\varepsilon(\mathcal{M})$ of order
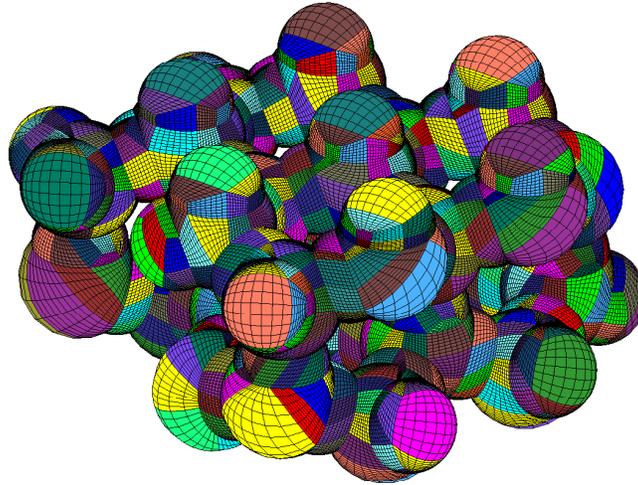

Fig. 2. Decomposition with 1284 four-sided patches

Table 1. Decomposing molecular surfaces: comparison with meshes

Molecule	Nb atoms	Nb patches	Time (patch)	Nb triangles	$\varepsilon(\mathcal{M})$	Time (mesh)
Benzene	12	160	2.26 sec	18615	5.56e-004	4.40 sec
Fullerene	60	360	2.45 sec	102380	6.88e-004	9.09 sec
Propane	11	231	9.65 sec	21927	3.53e-004	11.73 sec
Petane	17	388	11.77 sec	38764	3.31e-004	15.67 sec
Ice	84	1284	71.97 sec	290732	4.96e-004	77.22 sec

10^{-4}, several thousands of triangles are required. Further, only 9 control points for each patch are necessary to completely store the results. Additionally, the patches are exact as opposed to meshes where piecewise linear approximation is required. The runtime depends in fact on the molecular models. In general, the runtime for mesh-free geometric preparation is slightly faster than mesh-based one. But sometimes, the difference is tangible as in the case of fullerene where the surface structure is almost uniform. On the other hand, for every two neighboring patches, the lines incident upon the interface match very well as illustrated in Fig. 2. That is, the global continuity that we discussed in the above theory is obtained. As for the simulation side, there are efficient numerical solvers which accept mesh-free models directly. Those solvers do not require the patch representation to be converted to meshes. A concrete example is the Wavelet-Galerkin solver which needs only the control points for the generation of multiscale bases.

References

1. Laug P, Borouchaki H. Surface grid generation on intersecting spheres. Proc 6th International Conference Numerical Grid Generation. 1998; p. 717–26.
2. Cances E, Mennucci B. New applications of integral equations methods for solvation continuum models: ionic solutions and liquid crystals. J Math Chemistry. 1998;23:309–26.
3. Dietz R, Hoschek J, Jüttler B. An algebraic approach to curves and surfaces on the sphere and on other quadrics. Comput Aided Geom Des. 1993;10(3):211–29.
4. Laug P, Borouchaki H. Molecular surface modeling and meshing. Proc 10th International Meshing Round Table of Sandia National Laboratory. 2001; p. 31–41.
5. Randrianarivony M. Geometric processing of CAD data and meshes as input of integral equation solvers, PhD thesis. Chemnitz: Technische Universität Chemnitz; 2006.
6. Brunnett G. Geometric design with trimmed surfaces. Comput Suppl. 1995;10:101–15.
7. Albrecht G. An algorithm for parametric quadric patch construction. Comput. 2004;72:1–12.