Calculating the Solvent Exposed Surface Area of Proteins with Intersecting Spheres

Gargi Kher, Neil Gupta

June 5, 2020

Contents

1 Abstract

Calculating the solvent-exposed surface area (SASA) of large biomolecules is an important problem in structural biology, especially in the case of proteins, where small changes in the molecule's shape can greatly limit the protein's ability to do its job. The fraction of a polar atom's surface area that is exposed to the engulfing solvent can be the difference between a stable hydrogen bond being formed with the solvent and a polar atom buried within the protein, which could prevent the protein from folding into its proper shape. Molecular dynamics simulations, which use computers to predict the time evolution of a molecular system also rely on accurate SASA calculations as a way to compute the enthalpy of hydration for the molecules in the system. Today, SASA is usually calculated based on an implicit solvation model, where the solvent is taken to be a continuous field that the molecules of interest sit in. The atoms are taken to be spheres of definite radius, and thus calculating the SASA of an atom involves calculating the fraction of the sphere that is exposed. The SASA of the entire molecule can be found by summing this quantity over all atoms. This paper gives a overview of ALPHAMOL, a way of analytically calculating the derivative of SASA with respect to time, a quantity which can be numerically integrated as part of a molecular dynamics simulation to predict the future state of a molecular system. We also discuss a previous way of computing SASA that ALPHAMOL improves upon. Finally we do some calculations of our own to benchmark the performance of these two methods and provide some modern context for where SASA computations are used.

2 Previous Work

Accurately and quickly calculating the SASA of proteins has been a problem that people have been trying to solve since the 1980's. One reason for this is the fact that the effective potential for solvation can be written $W = W_{elec} + W_{np}$, where W_{elec} represents the electrostatic contribution and W_{np} represents the nonpolar contribution to this potential. The second term depends heavily on the SASA value of the polar atoms in the molecule, as polar atoms with a low SASA value (buried polars) should to make hydrogen bonds to preserve the protein's structure. One of the earliest ways of computing this value was by Richmond (1984). His method involved modeling atoms as spheres and computing SASA as an application of the Gauss-Bonnet Theorem. This method was the basis for many future SASA calculators that optimized for attributes like speed or numerical stability. Other methods didn't focus as much on exact values in the interest of speed. The Shrake-Rupley algorithm (optimized by Legrand and Merz as discussed below), breaks up the spherical surface of each atom into a set of dots, and uses the fraction of dots that are exposed as a proxy for the exposed surface area. This is the default SASA calculator in Rosetta, a software package developed at the UW that allows for the computational analysis and design of protein structures.

2.1 Geometrical Representation of Atoms

While atoms are comprised of protons, neutrons, and electrons, electrons are what give them their properties. Electrons make up the volume of an atom, and are more likely to be found closer to the nucleus than away. Because we don't know for certain where electrons are in an atom, they can be represented by orbitals, or positions that give the probability of electrons being there. Orbitals can take on different shapes, generally resembling some type of dumbbell, but when these orbitals come together, they form a roughly spherical shape (DeCoste, Zumdahl p.551-557). This is why

atoms are often mathematically modeled as spherical objects.

2.2 Shrake-Rupley Algorithm

The Shrake-Rupley Algorithm (Shrake and Rupley, 1973) is the standard way of calculating SASA by approximating the surface as a discrete set of voxels instead of a continuous surface. This can lead to inaccuracies, but the gains in speed are usually worth it. Prior exact algorithms could also fail if the protein surface was not continuous (i.e. an internal cavity is present) or the atomic radii are too small. This algorithm, once optimized by Legrand and Merz (1993) became a standard SASA calculator. Comparing this algorithm with the analytic SASA calculator MSEED (Perrot et al.) in calculating the SASA of the protein BPTI showed Shrake-Rupley to be roughly six times faster than MSEED and not subject to the surface continuity problems that can trouble the analytic calculator. The method works by representing the surface of sphere a_i by an array of l points evenly distributed on the surface of a sphere of radius r_i. The molecular surface area can be then written as integral of didio $\Psi\Phi\Xi\alpha^2$

$$A_i = 4\pi r_i^2 \frac{l_{exposed}}{l_{total}} a_i = 3\pi \tag{1}$$

where $l_{exposed}$ is the number of points that are solvent exposed. Now if we were to calculate this quantity for each atom we would need to make nkl calculations, where k is the average number of atoms that intersect a given atom. This could get quite intensive. What Legrand and Merz did was realize that any surface point is either buried or exposed, and thus the state of a_i's surface points could be represented as an $l/8$ byte string b_i, where $b_{ij} = 1$ means that the point is exposed and $b_{ij} = 0$ means that the point is buried. This has the benefit of making the calculation of the points on a_i buried by a neighboring sphere a_q a boolean AND operation on b_i. By defining m_{iq} as a boolean string with k^{th} element 0 when the k^{th} point of sphere a_i is buried and one otherwise, the set of points on a_i left exposed by the intersection with a_q is (111...1) & m_{iq}. Thus

$$b_i = (111...1)\ \&\ m_{i1}\ \&\ m_{i2}\ \&\ m_{i3}\ \&\ ...\ \&\ m_{ik}\&m_{ik+1}$$

The SASA of a_i can then be calculated by (1) by setting $l_{exposed}$ as the number of bits in b_i.

To actually calculate m_{iq}, we first compute

$$\cos\theta_{iq} = \frac{r_i^2 + r_{iq}^2 - r_q^2}{2r_i r_{iq}} \tag{2}$$

$$d_{iq} = 2r_i \sin\theta_{iq} = 2r_i\sqrt{\frac{1-\cos\theta_{iq}}{2}} = r_i\sqrt{2(1-\cos\theta_{iq})} \tag{3}$$

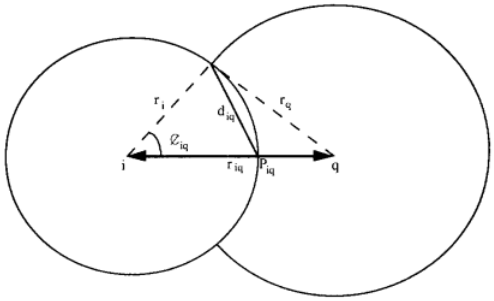


Figure 3. Calculation of d_{iq} and θ_{iq} for the purpose of calculating m_{iq}.

All points on a_i within d_i of the P_{iq} are buried, and those bits in m_{iq} are set to zero, with the rest set to 1. This string can be AND'ed with b_i to produce a new b_i. This can be done for all spheres that intersect a_i for each sphere in turn. The key insight that cuts down on the nkl calculations is the fact that the m_{iq} can be stored and retrieved later if the same string is ever needed. This is very likely to occur, because the d_{iq} can be made independent of r_i by removing the factor. Then $d_{iq} = \sqrt{2(1-\cos\theta_{iq})}$. Now we can calculate m_{iq} for the range of d_{iq} from 0-2.

2.3 Richmond's Algorithm and the Gauss-Bonnet Theorem

In a previous paper written by Richmond (1984), a different strategy is used to find the solvent exposed surface area of the atoms. He still represents the atoms as spheres with definite radius with centers at the atomic coordinates, but instead of using the Alpha Shapes, he made use of the Gauss-Bonnet theorem, a fundamental theorem in differential geometry which connects the topological properties of a manifold (specifically its Euler Characteristic) with its geometric properties (its curvature). A standard statement of the theorem is

$$\int_M K dA + \int_{\partial M} k_g ds = 2\pi\chi(M) \tag{4}$$

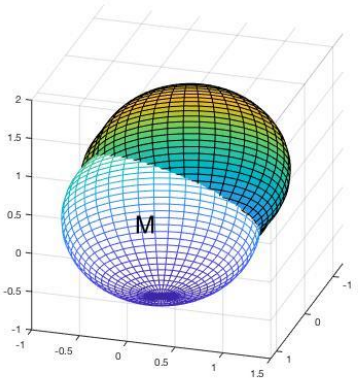
Here χ is the Euler Characteristic of M, and K and k_g are the Gaussian and geodesic curvatures of the manifold and the boundary respectively. For polygons, $\chi = V - E + F = 1$.

The SASA of an atom with radius ϱ_i lies on a sphere where $K = \frac{1}{\varrho_i^2}$. We can divide this SASA into many polygons bounded by the circles of intersection with neighboring spheres (See Figure). If M is one particular region of that SASA, we can write

$$\begin{aligned} &\frac{1}{\varrho_i^2}\int_M dA + \int_{\partial M} k_g ds = 2\pi \\ &\implies A = \varrho_i^2\left(2\pi - \int_{\partial M} k_g ds\right) \end{aligned} \tag{5}$$

Thus if we know k_g (the geodesic curvature) of the boundary of M at all points, we can compute A. If ∂M is piecewise smooth, then $\int_{\partial M} k_g ds$ is the sum of the integrals along the components of the boundary plus the sum of the angles by which the elements of the boundary turn at the corners.

Figure 1: Two Intersecting Spheres



For the simple example of a geodesic triangle on a sphere of radius ρ, $\int_\gamma k_g ds = 0$ where γ is a piece of ∂M. Thus $\int_{\partial M} k_g ds = 0 + \sum(\pi - \alpha_i)$, where α_i are the interior angles of the triangle. Plugging this into (5) gives $A = \rho^2(-\pi + \sum \alpha_i)$. This is the well known Girard's Theorem.

In our case, the boundaries of M won't be piecewise geodesic, and thus calculating $\int_{\partial M} k_g ds$ will require knowledge of the circles of intersection between the spheres. With that data calculating k_g simply requires some vector calculus[1]. We can calculate the SASA of the entire protein this way, by adding up the contributions of the individual regions M across all atoms.

3 The Power Diagram Model

3.1 Power Diagrams

Our protein can be represented as a set of atom centers S in $\mathbb{R}^3$ with their corresponding balls of various radii. One aspect of the model is computing the power diagram of this protein. The power diagram is a generalized Voronoi diagram, where instead of the normal Euclidean metric, the definition for distance between an atom with center z_i and radius ϱ_i and a point x is $\pi_i(x) = \|x - z_i\|^2 - \varrho^2$. This becomes the default Voronoi diagram of S if all the radii are equal. After this power diagram is generated we can associate each atom S_i with its corresponding power cell P_i defined by

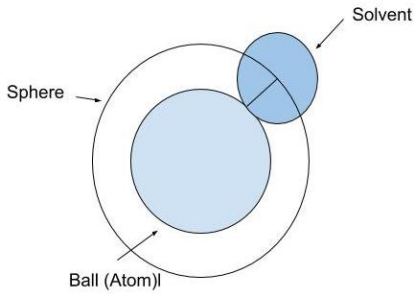
$$P_i = \{x \in \mathbb{R}^3 \mid \pi_i(x) \leq \pi_j(x), \forall j\}$$

Each P_i is convex, and together they cover all $\mathbb{R}^3$ without overlap. The dual of this power diagram is a weighted Delaunay Triangulation of S into a set of simplices. We can compute the surface area of these balls using Delaunay triangulation, which will help us in the later sections. There are additional rules for constructing Delaunay triangles, but for the purposes of this paper, it is only necessary to know what the vertices of each triangle represent.

3.2 Application to Protein Structures

Consider a set of balls B_i defined by a center z_i and a fixed radius ϱ_i to be our protein. To represent the entire state of the protein (with n atoms) we use the $m = 3n$ dimensional vector $\mathbf{z} \in \mathbb{R}^{3n}$ where $[\mathbf{z}_{3i+1}, \mathbf{z}_{3i+2}, \mathbf{z}_{3i+3}]^T = z_i$. Now we can define the weighted and unweighted surface areas $W, A : \mathbb{R}^{3m} \to \mathbb{R}$ as real multivariate functions from the state of the protein to a single value (the solvent exposed surface area). The weighted area will be described below. Here we will describe the gradients of these maps, written as $DW_{\mathbf{z}}$ and $DA_{\mathbf{z}}$ which would allow us to calculate SASA's for protein structures and approximate how the SASA changes as the atomic coordinates move. These derivatives map each point in $\mathbb{R}^m$ to a corresponding gradient vector that we call u.

Figure 2: B_1 where $n = 2$ is bound by a solvent sphere. The radius of the solvent is the distance between the edge of the ball and the sphere.



Given a collection of atoms, there will be many overlapping spheres in $\mathbb{R}^{3n}$. We will describe area derivative equations for groups of two or three overlapping spheres, which can easily be applied to larger systems. Using the power method described above, we will show how the authors use Delaunay Triangulations to assist with the calculations. Figure 2 shows the simplest example.

In order to get derive the SASA equations, we must define some additional terms. After constructing Delaunay Triangles from a set of molecules, they note two important terms , "star" and "link", that are useful for defining area equations. Each of the Delaunay triangles as a simplex, where simplex is basically another word for triangle or tetrahedron. They define K as a dual complex, or subcomplex of the alpha shape.

As can be seen by the figure, K is made up of multiple simplices. The "star" of K is a set of triangles who share a vertex and outside boundary. The "link" of K, then, is all of the vertices and edges in the star.

Knowing how the star and link of a subcomplex K are defined is useful when deriving the SASA equations. The collection of spheres, or the entire molecule, is known as a space-filling diagram F. F_i, for instance, consists of all the atoms in S_i. σ represents the fraction of a sphere on the boundary of F. We only really care about the boundary of F because that is what contributes to the exposed surface area that will be interacting with the solvent. As mentioned earlier, a molecule can contain multiple spheres that overlap. For two intesecting spheres S_i and S_j that share a circle S_{ij} σ comes out to:

Figure 3: Three balls in space bounded by three different spheres. Their centers are used for Delaunay Triangulation.

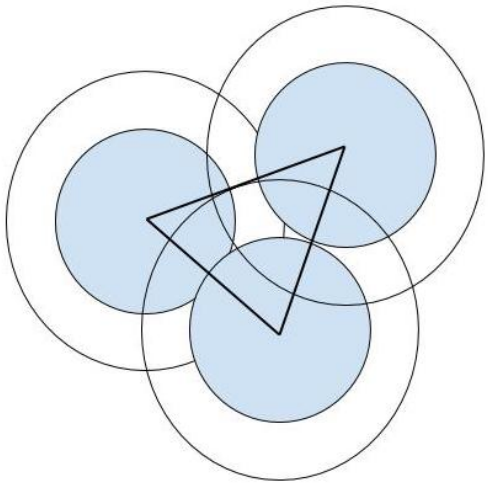
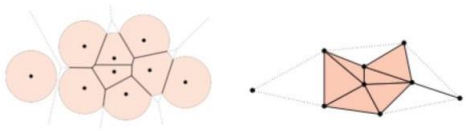


Figure 4: A power diagram comprised of power cells (left) and its Delaunay Triangulation (right).[1]



$$\sigma_{ij} = \frac{length(P_i \cap P_j \cap bdF)}{length(S_{ij})}$$

bd F means the boundary of F. This equation shows that for the space-filling diagram of the two intersecting spheres, σ_{ij} is the fraction of the exposed surface that's on the power cells of both of the spheres. Using the dual complex, the authors go on to specify the equation for σ_{ij} in terms of two intersecting circles, which will become useful later:

$$\sigma_{ij} = 1 - \frac{\sum_k length(S_{ij}^k) - \sum_{k,l} length(S_{ij}^k \cap S_{ij}^l)}{2\pi\varrho_{ij}}$$

This equation is derived by looking at Edelsbrunner's (1995) paper. The same two spheres S_i and

Figure 5: The dual complex (K) of the Delaunay Triangulation only includes those power cells that non-empty intersection.[1]

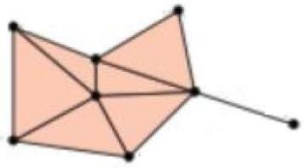
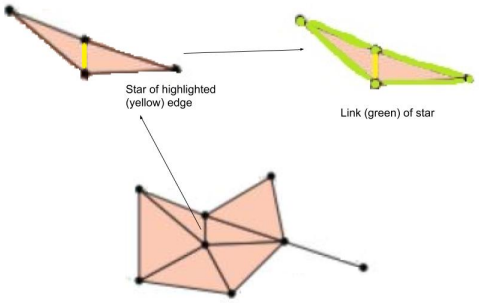


Figure 6: The star of K and link of the star are shown.[1]



S_j intersect at S_{ij}. However this time, S_{ij}^k and S_{ij}^l are two other spheres in the power diagram coming into contact with S_{ij}. Both S_{ij}^k and S_{ij}^l minimize the power distance to S_{ij}. Looking at the term on the right, the first sum adds the power distances from the portion of S_{ij} coming into contact with S_k, while the second sum adds the power distances from the portion of S_{ij} coming into contact with both S_k and S_l. Having this second sum ensures we don't underestimate the amount of unexposed area. Dividing the difference of these sums by the circumference of the circle gives the fraction of this unexposed area, so that subtracting this term from 1 gives the fraction of exposed surface area of F.

σ is related to the area of the ball, which for a ball with radius ϱ is $4pi\varrho^2$. Because we only care about the exposed surface area, we need to multiply by σ, in order to only get the solvent-exposed surface area. We can also define a weighted area, which only differs by the area equation by the factor of some parameter α, which represents some constant atomic solvation parameters that were

predefined by Eisenberg and McLachlan(1986). So for just one ball B_i:

$$A = 4\pi \sum \varrho_i^2 \sigma_i \qquad\qquad E = 4\pi \sum \alpha_i \varrho_i^2 \sigma_i$$

Another term, β_{ijk}, is also used, which is useful when considering the case of three intersecting balls. When three spheres intersect at two points, there is a line segment passing through them that is represented as B_{ijk}. Just like for σ_{ij}, β_{ijk} represents the fraction of points that belong to the edge of F for these three spheres.

$$\beta_{ijk} = \frac{length(P_i \cap P_j \cap P_k \cap bdF)}{length(B_{ijk})}$$

The equation above shows that β_{ijk} represents the fraction of the exposed points that are on the intersection of the three spheres. A similar expression to σ_{ij} is derived for β_{ijk}, using B_{ijk}^l, the portion of the line segment B_{ijk} on a fourth sphere S_l.

$$\beta_{ijk} = 1 - \frac{\sum_l length(B_{ijk}^l)}{length(B_{ijk})}$$

In the right term, the sum adds up the length of the line segment that comes in contact with S_l and divides that by the entire length of B_{ijk}, giving the fraction of F that is unexposed. By subtracting this term from 1, the fraction of solvent exposed area for three intersecting spheres is found.

We can think of the sphere as representing how much of the solvent can interact with the atom. The state $\mathbf{z}$ of each ball defines a center and a radius. As stated before, this paper considers that the weighted (E) and unweighted (A) areas are transformations from the space containing z ($\mathbb{R}^{3n}$) to $\mathbb{R}$. Their derivatives at a particular $\mathbf{z}$ are linear maps. In other words, the areas are represented by the map f and their derivatives the derivative of f at $\mathbf{z}$. Both unweighted and weighted area derivatives are comprised of two motions, direction preserving and distance preserving, which can be derived separately and summed to form the theorems.

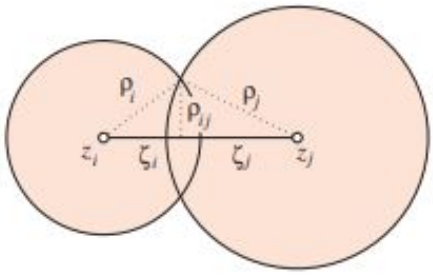
3.3 Direction Preserving Motion

In this type of motion, it is assumed that two spheres S_i and S_j bounded with a non-empty intersection are moving straight on the line passing through their centers. In this case, the space-filling diagram F consits of all the atoms in S_i and S_j, or $F = B_i \cup B_j$.

A term $\zeta_i j$ is equal to ζ_i+ζ_j, or the norm of the distance between the centers of the two spheres z_i and z_j. ζ in this case represents the signed distance of each sphere center to the circle of intersection. Each of the sphere's own radi are defined to be ρ_i and ρ_j. First, we can assume coordiates for the spheres based on what we know about their centers: (0,0,z_i) (0,0,z_j)

The general equation of a sphere ($x^2 + y^2 + z^2 = r^2$) can be used to construct equations for S_i and

Figure 7: Two intersecting spheres S_i and S_j.[1]



S_j:

$$
\begin{aligned}
&x^2 + y^2 + (z - z_i)^2 = \zeta_i^2 \\
&x^2 + y^2 + (z - z_j)^2 = \zeta_j^2 \\
&\zeta_i^2 - \zeta_j^2 = (z - z_i)^2 - (z - z_j)^2 = -2z_i + 2z_j + z_i^2 - z_j^2 \\
&\implies -2z_i + 2z_j + z_i^2 - z_j^2 = (\zeta_i)^2 - (\zeta_j)^2
\end{aligned}
$$

Using this logic, we can find the same equation holds true if we consider the set of balls for each sphere (assuming they have they same radi), as they have the same centers as the spheres. So:

$$
\begin{aligned}
&-2z_i + 2z_j + z_i^2 - z_j^2 = \varrho_i^2 - \varrho_j^2 \\
&\zeta_i^2 - \zeta_j^2 = \varrho_i^2 - \varrho_j^2
\end{aligned}
$$

We can factor out $\zeta_i^2 - \zeta_j^2$ to get:

$$
\zeta_i^2 - \zeta_j^2 = (\zeta_i - \zeta_j)(\zeta_i + \zeta_j) = \zeta_{ij}(\zeta_i - \zeta_j)
$$

Now we can find what ζ_i and ζ_j really are

$$
\begin{aligned}
&\varrho_i^2 - \varrho_j^2 = \zeta_{ij}(\zeta_i - \zeta_j) \\
&(\varrho_i^2 - \varrho_j^2)/\zeta_{ij} + \zeta_j = \zeta_i
\end{aligned}
$$

We know that $\zeta_j = \zeta_{ij} - \zeta_i$, so just substitute this expression in:

$$
\zeta_i = \frac{1}{2}\left(\zeta_{ij} + \frac{\varrho_i^2 - \varrho_j^2}{\zeta_{ij}}\right)
$$

A similar statement can be made for ζ_j, following the steps we carried out above:

$$
\zeta_j = \frac{1}{2}\left(\zeta_{ij} + \frac{\varrho_j^2 - \varrho_i^2}{\zeta_{ij}}\right)
$$

Now we can find the area of both spheres. For two spheres, it makes sense that $A = A_i + A_j$. Recall the area equation stated in section 3.2. For two balls we can say: $A = 4\pi\varrho^2 \sum \sigma_i + 4\pi\varrho^2 \sum \sigma_j$. We

have already shown σ, so we can plug it in and simplify the area equation:

$$4\pi\varrho^2\frac{\varrho_i+\zeta_i}{2\varrho_i}+4\pi\varrho_j^2\frac{\varrho_j+\zeta_j}{2\varrho_j}$$

After a few algebraic simplifications:

$$2\pi\varrho_i^2+2\pi\varrho_j^2+\pi\varrho_i\left(\zeta_{ij}+\frac{\varrho_i^2-\varrho_j^2}{\zeta_{ij}}\right)+\pi\varrho_j\left(\zeta_{ij}+\frac{\varrho_j^2-\varrho_i^2}{\zeta_{ij}}\right)$$

$$A=2\pi*(\varrho_i^2+\varrho_j^2)+\pi(\varrho_i+\varrho_j)\left[\zeta_{ij}+\frac{(\varrho_i-\varrho_j)^2}{\zeta_{ij}}\right]$$

Finally, we can find the derivative of the area with respect to ζ_{ij}

$$\frac{dA}{d\zeta_{ij}}=\pi(\varrho_i+\varrho_j)\left[1-\frac{(\varrho_i-\varrho_j)^2}{\zeta_{ij}^2}\right]$$

This is term they get after some simplification.

3.4 Distance Preserving Motion

In distance preserving motion, one sphere rotates about the other. Imagine three spheres interesting at two points, p and q. This time, F is going to include all the points contained in all three spheres. It is assumer that if the spheres intersect at two points then one sphere contains the other two. In this case, it is S_j that engulfs S_i and S_k. The caps, or tops, of the spheres are considered. From a previous section, we know that B_{ijk} is considered the line segment of three intersecting spheres. In this case, B_{ijk} is the line segment containing p and q. The caps $C_{ij}=B_i\cup S_j$ and $C_{kj}=B_k\cup S_j$ protrude from the surface of S_j. The parameters obtained from this allows us to find the area similar to how we found it for the direction preserving motion. This model says that as you move the two caps apart, keeping their centers in the same position, you will gain an area equal to that of a spherical rectangle. Essentially, the caps are rotating in place. It might be easy to think about how diameter of the caps can represent the height of this "rectangle" while the distance between the caps signifies its width.

Using this spherical rectangle shape, and the fact that we know the caps are rotating, the change in area is found to be the change in the difference between the center of the circles η_j divided by the circumference of the set of balls in the larger sphere (which S_j is in this case) times the area of the slice of S_j between p and q:

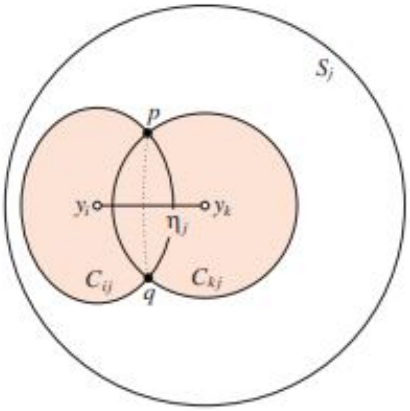
$$dA_j=\frac{d\eta_j}{2\pi\varrho_j}4\pi\varrho_j\varrho_{ijk} \tag{6}$$

$$\tag{7}$$

$\frac{d\eta_j}{2\pi\varrho_j}$ acts like the change in the "width" of the spherical rectangle while $4\pi\varrho_j\varrho_{ijk}$ serves as the change in the "height" of the spherical rectangle.The area derivative of this larger ball becomes:

$$\frac{dA_j}{d\eta_j}=2\varrho_{ijk}$$

Figure 8: Three spheres intersecting at two points: S_i, S_k and S_j.[1]



3.5 Combining the Area Equations

Now that we've derived area derivative equations for both types of motions, we can see how they're able to come together. As mentioned before, because the derivatives are linear, they can be separated into a sum of certain components. The gradient vector $\mathbf{u}$ shows the direction or trend of the area. For the case of the direction-preserving motion it is stated $u_ij = \frac{z_i - z_j}{zeta_ij}$, showing that the unit vector u_{ij} for two intersecting spheres S_i and S_j is dependent on the distance between their centers. The gradient vector u_ijk for distance preserving motion is dependent on the component of the intersection to two of the spheres (u_ik) that is normal to the component of intersection between the other two spheres (u_ij). This makes $u_{ijk} = \frac{u_{ik} - <u_{ik}, u_{ij}> \cdot u_{ij}}{\|(u_{ik} - <u_{ik}, u_{ij}>) \cdot u_{ij}\|}$

Based on what has already been shown, the unweighted and weighted area derivative theorems can be stated. Remember that we said each state $\mathbf{z}$ was in $\mathbb{R}^{3n}$, and each $\mathbf{z}$ represents set of balls B_i so that $0 <= i <= n-1$. Looking at the area derivative theorem, the derivative of the area at some $\mathbf{z}$ to be $DA_z(\mathbf{t}) =< \mathbf{a}, \mathbf{t} >$, where $\mathbf{a}$ is a gradient vector like $\mathbf{u}$ and $\mathbf{t}$ is the variable vector mentioned above. They state that for the unweighted area derivative,

$$\mathbf{a} = \sum_j \sigma_{ij} \cdot a_{ij} + \sum_j \sum_k \beta_{ijk} \cdot a_{ijk} \tag{8}$$

The first sum shows the contribution of the direction preserving motion while the second sum shows the contribution of the distance preserving motion. We can see for the first sum that we are summing over all of the points along the boundary when considering the intersection of two spheres S_i and S_j, and the second sum considers the sum of points along the intersection for three intersecting spheres. Also for a_{ij} and a_{ijk}:

$$a_{ij} = \pi * (\varrho_i + \varrho_j) * \left[1 - \frac{(\varrho_i - \varrho_j)^2}{\zeta_{ij}^2}\right] \cdot u_{ij}$$

$$a_{ijk} = 2\varrho_{ijk} \frac{\varrho_i - \varrho_j}{\zeta_{ij}} \cdot u_{ijk}$$

The weighted area derivative theorem, defined as $DE_z(\mathbf{t}) =< \mathbf{e}, \mathbf{t} >$ is defined similarly to the area derivative theorem, but multiplied by the solvent parameter α i.e

$$\mathbf{e} = \sum_j \sigma_{ij} \cdot e_{ij} + \sum_j \sum_k \beta_{ijk} \cdot e_{ijk}$$
$$e_{ij} = \pi \left((\alpha_i \varrho_i + \alpha_j \varrho_j) - \frac{(\alpha_i \varrho_i - \alpha_j \varrho_j)(\varrho_i^2 - \varrho_j^2)}{\zeta_{ij}^2} \right) \cdot u_{ij}$$
$$e_{ijk} = 2\varrho_{ijk} \frac{\alpha_i \varrho_i - \alpha_j \varrho_j}{\zeta_{ij}} \cdot u_{ijk}$$

Above, we described **a** and **e**. Using these, we can the clearly state the area derivative equations. For the unweighted area derivative equation, the direction preserving component is the the fraction of the solvent-exposed surface times the derivative of the area: $\sigma_{ij} < a_{ij}, t_i >$ and the distance preserving component is the same: $\beta_{ijk} < a_{ijk}, t_i >$. We know that the unweighted area derivative is the sum of these two components:

$$DA_z(\mathbf{t}) = \sum_{i=0}^{n-1} \sum_{j!=i} \left(\sigma_{ij} < a_{ij}, t_i > + \sum_{k!=i,j} \beta_{ijk} < a_{ijk}, t_i > \right) \tag{9}$$

Just like before, the weighted area derivative is basically the same as the unweighted derivative:

$$DE_z(\mathbf{t}) = \sum_{i=0}^{n-1} \sum_{j!=i} \left(\sigma_{ij} < e_{ij}, t_i > + \sum_{k!=i,j} \beta_{ijk} < e_{ijk}, t_i > \right) \tag{10}$$

These are the derivative equations that can then be implemented into a software and run on different molecules. It is important to note that there are cases where these areas will be discontinuous. We won't elaborate on these, but, for example, discontinuities can happen in a case where three spheres intersect at a circle. There are other cases of discontinuities, which are useful to know in order to run this program successfully.

3.6 Model Assumptions

As explained above, one of the biggest assumptions made is by representing each atom as a ball. This ball is surrounded by a sphere whose distance from the edge of the ball represents the radius of a solvent sphere (Figure 1). A solvent refers to a liquid the molecule is suspended in, usually water. Another assumption to point out is that water is being represented as a sphere (continuously) rather than a collection of atoms.

4 Application of ALPHAMOL to Molecular Modeling

To test the time cost of ALPHAMOL (the computational implement of the method described in §3) we took 250 PDB structures from the RCSB database (listed in the supplemental) and calculated the number of polar atoms buried in the structure that are not making hydrogen bonds. This

calculation indirectly uses a SASA calculator to determine if atoms close to the surface are truly buried. If they have any SASA that is nonzero, the atom is assumed to be making a hydrogen bond with the implicit solvent. We compared the time taken to complete the buried unsatisfied polars calculation for the ALPHAMOL and Shrake-Rupley SASA calculators. We found that ALPHAMOL is slower, but it is more accurate.

5 Glossary

ϱ_i :	The radius of ball i
z_i :	The center of a ball i
$\mathbf{z}$:	The $3n$ dimensional vector representing the state of all z_i
ζ_{ij} :	The signed distance between centers of two intersecting spheres
σ_{ij} :	The fraction of solvent-exposed surface for two intersecting spheres S_i, S_j
β_{ijk} :	The fraction of solvent-exposed surface for three intersecting spheres S_i, S_j, S_k
α :	An atomic solvation parameter
F :	The space filling diagram
$\pi_i(x)$:	The power distance of a random point x from S_i
P_i :	The set of points closer (by power distance) to S_i than any other sphere
K :	The dual complex of Delaunay Triangulation. Consists of all power cells with non-empty intersection
$DA_{\mathbf{z}}(\mathbf{t})$:	The unweighted area derivative at $\mathbf{t}$
$DW_{\mathbf{z}}(\mathbf{t})$:	The weighted area derivative at $\mathbf{t}$

List of Figures

References

[1] DeCoste, Donald J., Zumdahl, Steven S. General Chemistry 142 University of Washington 7th Edition, Boston, Cengage Learning, 2015.

[2] Edelsbrunner, H. The Union of Balls and Its Dual Shape. Discrete & Computational Geometry. 13 (1995), 415-440.

[3] Fischer, Kaspar. Introduction to Alpha Shapes. Stanford University, lecture notes, 2011.

[4] Bryant, R., Edelsbrunner, H., Koehl, P., Levitt, M. The Area Derivative of a Space-Filling Diagram. Discrete & Computational Geometry. 32 (2004), 293-308.

[5] Richmond, Timothy J. Solvent Accessible Surface Area and Excluded Volume in Proteins. J. Mol. Biol. 178 (1984), 63-89.

[6] Rotskoff, Grant. The Gauss-Bonnet Theorem. University of Chicago REU, paper, 2010

[7] Schnell, Christian. The Gauss-Bonnet Theorem. talk at Columbus, 2004.

[8] Le Grand, Scott M., Merz, Kenneth M. Jr. Rapid approximation to molecular surface area via the use of Boolean logic and look-up tables. Journal of Computational Chemistry. 14 (1993). 349-352.