Novel Computer Program for Fast Exact Calculation of Accessible and Molecular Surface Areas and Average Surface Curvature

OLEG V. TSODIKOV,[1,*] M. THOMAS RECORD, JR.,[1] YURI V. SERGEEV[2]
[1]*Department of Chemistry, University of Wisconsin-Madison, Madison, Wisconsin 53706*
[2]*OGCSB/NEI/NIH, 10/10B10, 9000 Rockville Pike, Bethesda, Maryland 20892-1860*

Received 18 September 2001; Accepted 18 December 2001
DOI 10.1002/jcc.10061

Abstract: New computer programs, SurfRace and FastSurf, perform fast calculations of the solvent accessible and molecular (solvent excluded) surface areas of macromolecules. Program SurfRace also calculates the areas of cavities inaccessible from the outside. We introduce the definition of average curvature of molecular surface and calculate average molecular surface curvatures for each atom in a structure. All surface area and curvature calculations are analytic and therefore yield exact values of these quantities. High calculation speed of this software is achieved primarily by avoiding computationally expensive mathematical procedures wherever possible and by efficient handling of surface data structures. The programs are written initially in the language C for PCs running Windows 2000/98/NT, but their code is portable to other platforms with only minor changes in input-output procedures. The algorithm is robust and does not ignore either multiplicity or degeneracy of atomic overlaps. Fast, memory-efficient and robust execution make this software attractive for applications both in computationally expensive energy minimization algorithms, such as docking or molecular dynamics simulations, and in stand-alone surface area and curvature calculations.

 J Comput Chem 23: 600–609, 2002

Key words: accessible surface area; Gauss-Bonnet theorem; molecular surface; average curvature; cavities

Introduction

Interactions between biological macromolecules such as proteins and nucleic acids depend very strongly on features of interacting surfaces, such as steric complementarity, and the content of hydrophobic or polar surface area. Macromolecular stability and solubility of proteins are determined by how elements of the macromolecular surface interact with solvent and small solutes present in solution. Therefore, macromolecular surface is one of the most important concepts in analyzing macromolecular structure and function.

The first rigorous definition of solvent-accessible surface was introduced by Lee and Richards.[1] Accessible surface (AS) is defined by the center of a solvent molecule, modeled as a hard probe sphere, when it rolls over the surface of a macromolecule, modeled as a collection of hard van der Waals spheres. Later, Richards[2] introduced the concept of molecular (solvent excluded) surface (MS) and solvent-excluded volume. MS is defined by the part of the surface of the solvent probe sphere that faces the macromolecule (see below). Recent examples of biophysical applications of solvent accessible and MS area calculations include protein folding[3,4] and parameterization of heat capacity changes,[4] macromolecular docking,[5] enzyme catalysis,[6] calculation of solvation energies,[7] small solute effects,[8] and molecular dynamics.[9]

Modern structural techniques such as NMR or X-ray crystallography can yield molecular structures at very high resolution. Often, in calculations involving energy minimization, one needs to compute changes in surface area that result from many very small iterative perturbations in the structure. In order to preserve the high level of atomic detail in surface area calculations, numerical methods need to employ very fine discretization of the surface, which makes them computationally costly. In addition, numerical methods fail to accurately take into account the presence of internal cavities inaccessible from the outside and other topological features of the surface. Therefore, both fast and accurate analytic calculations of molecular parameters are indispensable.

*Present address: Department of Biological Chemistry and Molecular Pharmacology, Harvard Medical School, Bld. C-2 Room 226, 240 Longwood Ave., Boston, MA 02115; e-mail: oleg_tsodikov@hms.harvard.edu

Correspondence to: O.V. Tsodikov

Contract/grant sponsor: National Institutes of Health; contract/grant number: GM23467

Several algorithms and their software implementations for analytical calculation of AS and MS areas have been designed in the last two decades. The algorithm developed by Richmond[10] uses the Gauss-Bonnet theorem to calculate AS areas. Connolly[11] used the same Gauss-Bonnet approach and the formalism described by do Carmo[12] and extended the calculation to MS areas. In his algorithm, the calculation is performed sequentially over all elements of the MS. Gibson and Scheraga[13] developed a closed-form analytical method for AS area based on areas of overlap of multiple spheres. A recently published algorithm[14] uses the Gauss-Bonnet theorem and space transformation concepts to calculate AS area. Another elegant analytical approach derived from the alpha shape theory[15] was applied to calculations of MS and AS areas. Program MSMS[16] employed a very fast algorithm involving a binary tree search to find overlapping atoms, but avoiding calculation of many inaccessible vertices. Such a minimalist approach, however, does not allow one to calculate cavities large enough to accommodate a solvent probe but inaccessible from the outside. These recent algorithms based on dual space concepts and convex hull calculations are fast because they avoid calculation of vertices that do not form the outer surface.

The algorithm described in this study is based on the Gauss-Bonnet theorem and uses the idea of correspondence between ASs and MSs. AS is geometrically simpler and therefore its elements are computed first. Then, a very speed- and memory-efficient method of sequentially computing the elements of MS and areas of both MS and AS in the same run concludes the area calculation. Due to the multiple overlap of atomic spheres and overall irregular structure of a macromolecule, calculations of its surface area are both topologically and computationally challenging. The algorithm resolves complex atomic overlaps, including a situation in which more than three atomic spheres intersect at the same point. We implemented a physically reasonable way of treating self-intersecting surfaces, which pose another topological complication.

The empirically determined computational time for calculating protein accessible surface areas by our algorithm scales as N for $N < 2500$ and as N^2 for $N > 2500$, where N is the number of atoms in the structure. Even though some recently published algorithms using dual transformations reach optimal theoretical asymptotic computational efficiency ($\sim N\ln N$), they are quite mathematically complex and use computationally expensive geometrical calculations. This increases calculation time for smaller structures. The algorithm presented here does not use dual space transformations; it explicitly checks burial of each vertex. However, this procedure and subsequent surface calculation are performed in a very computationally efficient way, which makes our algorithm comparable or faster than those using dual space concepts. Because calculations on different atoms can be performed simultaneously, the algorithm can be readily adapted for parallel computation. We expect that combining ideas used in this algorithm with those that are implemented in the recently published methods of surface area calculation cited above will lead to yet further improvements in computational efficiency of surface area calculations.

In addition to quantity and composition of surface area, effects of surface curvature have been considered in calculations of thermodynamic contributions to solvation of nonpolar surfaces.[17,18] In docking of two macromolecular structures parameterized by curvature, geometrically complementary surfaces at the interface are required to have average curvatures that are equal in magnitude but opposite in sign. Therefore, this curvature criterion can be used as a necessary (but not sufficient) complementarity condition, thus allowing one to eliminate a subset of noncomplementary arrangements. We apply the concept of average surface curvature, which in classical physics is used to determine free energy of macroscopic surface tension, to molecular surface. Our computer program calculates molecular surface areas and respective average curvatures for each atom or a group of atoms in the structure.

Theory and Algorithm

Geometry of AS and MSs

We start by giving a brief overview of the concepts introduced by Connolly.[11] MS can be represented as a union of elements of three types. Points of contact between the probe sphere and atomic spheres form convex surface elements when the probe sphere touches only one atom. This part of the surface is also known as contact surface. Convex elements are joined smoothly by saddle surfaces formed by the part of the surface of the probe sphere facing the macromolecule (re-entrant part) when it contacts simultaneously two atomic spheres, and by convex surfaces formed by the re-entrant part of the probe sphere when it simultaneously contacts three atomic spheres. Molecular surface defined this way is smooth and therefore can be differentiated with respect to atomic coordinates or radii.

AS is defined by all positions of the center of the probe sphere. This surface is simpler because it consists of only convex parts. In fact, AS is MS of the structure in which radii of all atoms are incremented by the solvent probe radius, and the surface is then calculated by setting the probe radius equal to zero. However, unlike MS, AS is not differentiable as a whole because the convex surfaces in it are not joined smoothly. Nevertheless, these two surfaces are topologically equivalent. Convex surfaces of AS correspond to convex surfaces of MS, arcs that border convex surfaces of AS correspond to the saddles of MS, and the points of joining of the arcs (vertices) correspond to the convex surfaces of MS. This equivalence allows one to compute the elements of MS from AS without additional computational difficulty.

Calculation of the AS Graph

AS area can be represented as a three-dimensional graph, the nodes (vertices) of which are joined by the arcs formed by intersecting atomic spheres. The arcs surround convex surface elements. Each closed group of arcs forms a cycle. The cycles are oriented so that the accessible convex surface element is on the left side when moving along the cycle.[11] The first half of the algorithm, the calculation of this graph, is described in this section. Relevant mathematical expressions and nomenclature are given in Table 1. The terminology is analogous to that used by Connolly[11] except the expressions are converted to describe AS rather than MS, in terms of incremented atomic radii (see below).

First, the algorithm SurfRace increments each atomic radius ($r_{i,\mathrm{act}}$) by the probe radius ($r_i = r_{i,\mathrm{act}} + r_p$). From here on, the

Table 1. Parameters of Accessible and Molecular Surfaces.

Atomic coordinates	$\vec{a}_i$
Incremented radii	r_i
Interatomic distance	$d_{ij} = \lvert\vec{a}_j - \vec{a}_i\rvert$
Torus axis unit vector	$\vec{u}_i = (\vec{a}_j - \vec{a}_i)/d_{ij}$
Torus center	$\vec{t}_{ij} = \frac{1}{2}(\vec{a}_j + \vec{a}_i) + \frac{1}{2}(\vec{a}_j - \vec{a}_i)\frac{r_i^2 - r_j^2}{d_{ij}}$
Contact circle radius	$r_{ij} = \frac{1}{2}\frac{\sqrt{(r_i + r_j)^2 - d_{ij}^2}\ \sqrt{d_{ij}^2 - (r_i - r_j)^2}}{d_{ij}}$
Base triangle angle	$\omega_{ijk} = \arccos(\vec{u}_{ij} \cdot \vec{u}_{ik})$
Base plane normal vector	$\vec{u}_{ijk} = \frac{\vec{u}_{ij} \times \vec{u}_{ik}}{\sin \omega_{ijk}}$
Torus-basepoint unit vector	$\vec{u}_{tb} = \vec{u}_{ijk} \times \vec{u}_{ij}$
Base point	$\vec{b}_{ijk} = \vec{t}_{ij} + \frac{[\vec{u}_{ik} \cdot (\vec{t}_{ik} - \vec{t}_{ij})]}{\sin \omega_{ijk}}$
Vertex height	$h_{ijk} = \sqrt{r_i^2 - \lvert\vec{b}_{ijk} - \vec{a}_i\rvert^2}$
Vertex positions	$\vec{v}_{ijk} = \vec{b}_{ijk} \pm h_{ijk}\vec{u}_{ijk}$
Concave arc plane normal vector	$\vec{n}_{ijk} = (\vec{v}_{ijk} - \vec{t}_{ij}) \times \vec{u}_{ijk}/r_{ij}$
Concave triangle angle	$\beta_v = \arccos(\vec{n}_{ijk} \cdot \vec{n}_{ikj})$
Euler characteristic	χ = 2-number of cycles
Saddle wrap angle	$\varphi_s = \arccos(\vec{n}_{ijk} \cdot \vec{n}_{ijl})$, when $\vec{n}_{ijk} \times \vec{n}_{ijl} \cdot \vec{u}_{ij} < 0$ $\varphi_s = 2\pi - \arccos(\vec{n}_{ijk} \cdot \vec{n}_{ijl})$, when $\vec{n}_{ijk} \times \vec{n}_{ijl} \cdot \vec{u}_{ij} \geq 0$
Saddle width angle	$\theta_{si} = \arctan[\vec{u}_{ij} \cdot (\vec{t}_{ij} - \vec{a}_i)/r_{ij}]$

calculation is performed with these incremented atomic spheres. Next, the program analyzes atomic contacts. There are four possible spatial positions of each pair of atomic spheres.

1. The spheres have no common points and do not lie within each other.
2. The spheres touch each other, that is, have only one common point.
3. The spheres intersect, that is, have a common circle of points (contact circle[11]).
4. One sphere lies entirely within the other sphere.

Case 4 is not chemically possible and is not considered in the algorithm. Case 2 is treated identically to case 1, that is, the common point is simply disregarded. This makes physical sense because atomic positions are not fixed and the solvent probe will be able to pass through such a pair of atoms. Therefore, every two atoms in the structure will be treated as either case 1 or 3. For each atom i, the program finds all atoms intersecting atom i. In order to speed up this step computationally, we consider only atoms for which all three conditions $|x_i - x_j| < r_i + r_j$, $|y_i - y_j| < r_i + r_j$, $|z_i - z_j| < r_i + r_j$ (where x_i, y_i, z_i are atomic coordinates) are satisfied. For each of these atoms, the program checks the overlap condition $|\vec{a}_i - \vec{a}_j|^2 < (r_i + r_j)$.[2] In this condition, comparing the squares of distances is faster than comparing distances themselves, which requires using the more computationally expensive square root calculation. All atoms contacting atom i and the distances between them and atom i are stored in their respective arrays. Further analysis of triple intersections is conducted with these atoms. If atoms i and j intersect (case 2) then another atom $k \neq i,j$ intersecting atom i may or may not intersect atom j. If atom k intersects both atoms i and j the contact circles formed by atoms i and j and by atoms i and k may or may not intersect. If the two contact circles intersect, then two potential vertices are formed. The algorithm then checks whether either vertex is buried inside any atom intersecting atom i other than atoms j and k. If the vertex is not buried it then belongs to the AS graph. The coordinates of this vertex are stored in the node coordinate array. The ordered triplet of atom numbers i,j,k ($i < j < k$) is stored in the vertex atom number array.

If no other contact circle on atom i intersects the contact circle formed by atoms i and j, then the program checks whether an arbitrary point on this circle is buried inside any atom contacting atom i other than atom j. If this point is buried inside some atom, then the entire contact circle is buried by construction. If this point is not buried, then the contact circle is a part of the AS graph. The program records the coordinates of the point on the contact circle into the contact circle coordinate array. The ordered pair of atoms (i,j: $i < j$) forming this circle is stored into the contact circle atom number array.

When index i has run over all atoms in the structure, all vertices and contact circle of AS have been recorded in the respective arrays. In addition, these arrays will contain vertices and contact circles that belong to the surfaces of cavities, inaccessible from outside, large enough to accommodate the probe sphere. The algorithm resolves the outer surface (AS) from cavity surface(s) in the process of constructing the AS graph, as described below.

The program builds the AS graph using the stored information on vertices and contact circles. The AS graph construction starts with finding the first atom accessible from the outside. The program takes the atom whose $z_i + r_i$ is the largest. The point $(x_i,y_i,z_i + r_i)$ on its surface is accessible; it is the "north pole" of the structure.

Table 2. Expressions for Molecular Surface Area and Its Derivative with Respect to Displacement along a Normal Vector.

Convex:	$A_+ = r_i^2(2\pi\chi + \sum_s \varphi_s \sin\theta_{si} - \sum_v \beta_v)$
	$\left.\frac{dA_+}{dh}\right\|_{h=0} = \frac{2A_+}{r_i}$
Saddle:	$A_{si} = \varphi_s(r_p r_{ij}\theta_{si} - r_p^2 \sin\theta_{si}),\ r_p < r_{ij}$
	$A_{si} = \varphi_s[r_p^2(2\sin\theta_1 - \sin\theta_{si}) - r_p r_{ij}(2\theta_1 - \theta_{si})],\ r_p \geq r_{ij}$
	$\theta_1 = \arccos(r_{ij}/r_p)$
	$\left.\frac{dA_{si}}{dh}\right\|_{h=0} = \varphi_s(2r_p \sin\theta_{si} - r_{ij}\theta_{si}),\ r_p < r_{ij}$
	$\left.\frac{dA_{si}}{dh}\right\|_{h=0} = \varphi_s[r_p(-4\sin\theta_1 + 2\sin\theta_{si}) + r_{ij}(2\theta_1 - \theta_{si})],\ r_p \geq r_{ij}$
Concave:	$A_- = r_p^2(\sum_v \beta_v - \pi)$
	$\left.\frac{dA_-}{dh}\right\|_{h=0} = -\frac{2A_-}{r_p}$

The next step of the calculation constructs all cycles for this atom. The algorithm finds all vertices formed by the atom and records them in the array of vertices of the atom of interest. Starting from an arbitrary vertex, the program finds the next vertex connected to it by a properly oriented arc. If more than one vertex occurs at the same vertex point this degeneracy is resolved at this step by finding the right arc. The indices of the first and the second vertex are then recorded into the arc (edge) array. The number of the atom forming this edge with the first atom is recorded in the separate array. The process continues until the coordinates of the last vertex coincide with those of the starting vertex, thus building a complete cycle. The vertices forming this cycle are marked out of the list of vertices and recorded in the cycle array. The same process continues until all the vertices belonging to the first atom are marked out and therefore all properly oriented cycles will be built. Contact circles formed by the atom are cycles themselves. At the next step, the program determines all cycles that surround the convex surface element containing the north pole, that is, those that are accessible from the outside. Here we use the procedure developed by Connolly.[11]

1. The matrix of external-internal relationship between each pair of cycles A and B is determined, using the stereographic projection.[11,12]
2. The criterion that two cycles A and B border the same element if and only if A and B are internal with respect to each other and are internal to any other cycle C internal to both A and B. For the first atom, the north pole point is considered a cycle belonging to the convex surface element of interest.

The information about the cycles found in the previous step is entered into the expressions for area and average curvature calculation (Table 2). All vertices participating in this calculation are then marked off. Next, the algorithm moves on to the next atom, sharing an arc or a contact circle with the previous atom. It is essential to move from atom to atom in the most efficient way, storing all the necessary information and erasing the information that will not be used anymore in order to eliminate redundancy in the search for the next move. The edge graphs serve as a guide in moving from atom to atom. Whenever calculation for a given surface element is completed, the algorithm looks for remaining elements in the edge array. The edge atom in that element then corresponds to the next atom to perform the calculation on. After building each edge on this atom, the algorithm checks all remaining edges in the edge array and if it finds the same edge, the program erases it. If the algorithm determines that the edge has not been recorded yet, it records its parameters in the edge arrays. Therefore, each edge is processed only twice, as minimally necessary, because it joins two accessible surface elements. When an edge is encountered first it is recorded into the edge array and the first surface element is calculated, and when it is encountered the second time, the edge is erased and the second surface element is calculated. By construction, when no edges are left in the edge array, the AS (MS) calculation is completed.

Next, the program looks for remaining vertices in the vertex array. If such exist, they belong to cavities inaccessible from the outside. The calculation proceeds identically for each cavity, and ends for a given cavity when all edges get erased. When all vertices are erased, the calculation for a given structure terminates.

We implemented a different version of this algorithm in a program, FastSurf, which calculates only AS and MS area and average MS curvature but does not process the cavities. In this program, the vertices of only surface accessible atoms are calculated as the program encounters these atoms. The rest of the algorithm remains essentially the same. Therefore, FastSurf calculates vertices approximately $N^{1/3}$ times as fast as SurfRace at the expense of sacrificing cavity analysis.

Self-Intersecting Surfaces

The topological complexity of MS can lead to its self-intersections, which pose difficulties in visual representations of this surface

using computer graphics. However, this complexity does not appear to be relevant to area calculation. We propose a simple and physically meaningful way of treating self-intersecting parts of MS in area calculation. For thermodynamic considerations, translational entropy of water in the first hydration layer is determined by all possible locations of a water molecule in contact with protein surface. Therefore, areas of all parts of MS, including self-intersecting ones, should be summed independently, that is, as if they were not self-intersecting. In this sense, self-intersections are easily dealt with in area calculations, even though they still remain a challenge for graphical purposes. The expression for the saddle surface element[11] needs correction for the case when the saddle is self-intersecting ($r_p \geq r_{ij}$). According to this independent summation principle, we perform the integration over two parts ($\theta < \theta_1$ and $\theta_1 < \theta < \theta_{si}$, where θ_{si} and θ_1 are defined in Tables 1 and 2, respectively) and obtain the expression given in Table 2. The expressions for concave surfaces remain the same even if they are interpenetrating, and all areas calculated using these expressions are summed independently as though the areas do not overlap.

Average Surface Curvature

In this section we will apply the concept of average surface curvature to MS and derive analytic expression for average curvature calculation.

Consider a differentiable surface $\vec{r}_0 = \vec{r}_0(u, v)$. We will construct a one-parameter family of surfaces derived from this surface using the following transformation. Surfaces of this family are constructed by displacing the initial surface by h along a normal to each point. Then this family of surfaces is given by the equation $\vec{r} = \vec{r}_0(u, v) + h\vec{n}_0$, where

$$\vec{n}_0 = \frac{\frac{\partial \vec{r}_0}{\partial u} \times \frac{\partial \vec{r}_0}{\partial v}}{\left| \frac{\partial \vec{r}_0}{\partial u} \times \frac{\partial \vec{r}_0}{\partial v} \right|} \tag{1}$$

is a unit normal vector to surface $\vec{r}_0$. (Here and further, subscript "0" refers to surface $\vec{r}_0$.) The average surface curvature (H_0) of surface $\vec{r}_0$ at a given point is defined as the average of the two principal curvatures (k_1 and k_2) at that point:

$$H_0 = \tfrac{1}{2}(k_1 + k_2) \tag{2}$$

We will now consider the coefficients of the first fundamental form of surface $\vec{r}$ defined below, neglecting terms proportional to h^3 and higher powers of h:

$$E \equiv \left(\frac{\partial \vec{r}}{\partial u}\right)^2 = E_0 - 2hL_0 \tag{3}$$

where

$$L_0 = \frac{\partial^2 \vec{r}_0}{\partial u^2} \cdot \vec{n}_0$$

is a coefficient of the second fundamental form of surface $\vec{r}_0$. Analogously,

$$G \equiv \left(\frac{\partial \vec{r}}{\partial v}\right)^2 = G_0 - 2hN_0 \tag{4}$$

where

$$N_0 = \frac{\partial^2 \vec{r}_0}{\partial v^2} \cdot \vec{n}_0 \tag{5}$$

and

$$F \equiv \frac{\partial \vec{r}}{\partial u} \cdot \frac{\partial \vec{r}}{\partial v} = F_0 - 2hM_0 \tag{6}$$

where

$$M_0 = \frac{\partial^2 \vec{r}_0}{\partial u \partial v} \cdot \vec{n}_0 \tag{7}$$

Therefore, because average curvature H_0 is equal to the trace of the Weingarten mapping matrix[19]

$$H_0 = \frac{1}{2} \frac{L_0G_0 - 2M_0F_0 + N_0E_0}{E_0G_0 - F_0^2} \tag{8}$$

we obtain

$$EG - F^2 = (E_0G_0 - F_0^2)(1 - 4hH_0) \tag{9}$$

In the transformation from $\vec{r}_0$ to $\vec{r}$, each small element of surface $\vec{r}_0$ with surface area ΔA_0 corresponds to a small element of surface $\vec{r}$ with surface area ΔA. The ratio of these surface areas can be written in terms of the coefficients of the first fundamental form:[19]

$$\left(\frac{\Delta A}{\Delta A_0}\right)^2 = \frac{EG - F^2}{E_0G_0 - F_0^2} = 1 - 4hH_0 \tag{10}$$

which yields the expression for average surface curvature in terms of surface area

$$H_0 = \frac{1}{4h}\left[1 - \left(\frac{\Delta A}{\Delta A_0}\right)^2\right] \tag{11}$$

Multiplication of both sides of this equation by h and differentiation with respect to h yields

$$H_0 = -\frac{1}{4}\frac{d\left(\frac{\Delta A}{\Delta A_0}\right)^2}{dh} = -\frac{1}{2\Delta A_0}\frac{d\Delta A}{dh}\bigg|_{h=0} \tag{12}$$

In the limit of $\Delta A_0 \to 0$ the above expression is exact. For small finite ΔA_0, this expression can be taken as a definition of average curvature of molecular surface. This expression has a simple

physical interpretation. If one multiplies both sides of it by $\sigma\Delta A_0$, where σ is an arbitrary surface tension coefficient, then the left hand side will be equal to the force of surface tension and the right hand side is the gradient of surface energy.

This expression allows one to calculate average curvature for molecular surface. In order to derive the one-parameter surface family $\vec{r}$ defined above from a given molecular surface, one needs to apply two translations: increase the radius of each atom by h and decrease the probe radius by h. Then, the molecular surface built using the new atomic and probe radii will be a surface, which belongs to the family $\vec{r}$. MS area can be partitioned among all atoms in the structure in the following natural way: each atom is assigned its convex surface area element(s) (A_+), the area of a concave element (A_-) is divided by three and each participating atom is assigned one-third of this area, each saddle is divided by the plane from which the integration in the saddle area (A_{si}) calculation begins. MS area assigned to each atom is

$$A = A_+ + A_{si} + A_-/3 \tag{13}$$

where expressions for A_+, A_-, and A_{si} are given in Table 2. Because this area assignment is invariant of the displacement of the surface along the normal to each point, average surface curvature of each atom can be calculated by differentiating the area expressions with respect to h:

$$H = -\frac{1}{2A}\left(\frac{dA_+}{dh} + \frac{dA_{si}}{dh} + \frac{1}{3}\frac{dA_-}{dh}\right)\Bigg|_{h=0} \tag{14}$$

The two surface translations used in the construction are

$$\frac{dr_p}{dh} = -1 \text{ and } \frac{dr_i}{dh} = 1 \tag{15}$$

which yield expressions for area derivatives given in Table 2. The program partitions MS area and calculates the average curvature of each atom using these expressions.

The above definition of H can be easily generalized for several (n) atoms, using commutation of differentiation and summation, which yields the expression for the average curvature of n atoms:

$$H_n = \frac{\sum_{i=1}^{n} H_i A_i}{\sum_{i=1}^{n} A_i} \tag{16}$$

Therefore, the average curvature of a group of atoms is defined as a weighted average of their average curvatures with their areas as weights.

Computational Features and Characteristics of Surface Area Calculations by Surface Racer

We performed a series of AS and MS area calculations using the program FastSurf on a set of protein structures with a wide range of sizes. (The maximal allowed number of atoms in a structure handled by SurfRace is limited only by the size of user RAM.) This set included protein structures from a recent article by Liang et al.[15] to facilitate comparison with published algorithms. Table 3 shows the results of these area calculations. Both AS and MS areas are in excellent agreement with the output of existing algorithms. Calculation times of AS area for several structures with a wide range of sizes are plotted in Figure 1. As $N \to \infty$, the calculation time increases as N.[2] However, for $N < 2500$, the computation time is a linear function of N with a zero intercept (see the inset in Fig. 1 and Table 3), which is faster than the maximal speed asymptotic time dependence ($\sim N\ln N$ [16]). In this range, the computational speed of AS area calculation is comparable to or faster than the fastest program known to the authors, MSMS.[16] A combination of double and single real number precision as well as handling of most common degenerate cases makes our program more robust than MSMS; our program did not complete the calculation for only one out of about 100 different structures. We noticed that the frequency of various degenerate cases depends strongly on the choice of the set of van der Waals radii. This was the case for intersections of more than three incremented spheres at two very close points, such that the computational round-off error is on the order of the distance between the points. We found that degeneracies were minimized with the Richards[2] radii set. The sets in which many different atoms have identical radii, such as a set from Richmond and Richards,[20] result in higher frequency of degenerate cases, leading to program crashes or infinite looping (one out of 20–30 structures).

The program SurfRace performs the analysis of cavities and their AS and MS areas and average curvatures. Here "accessible" area is the locus of the center of a probe sphere put inside a cavity and rolling over its surface. In Table 4, we list the number of cavities and their AS areas for a set of proteins and compare them with results reported by Rashin et al.[21] computed using an approximate algorithm. We compare only proteins whose Protein Data Bank[22] (PDB; http://www.rcsb.org/pdb) coordinate files contain no nonwater HETATM (PDB term) records. Often cavities are just large enough to accommodate one probe sphere, therefore the exact number of them depends strongly on the choice of probe and atomic van der Waals radii. We used a probe radius of 1.2 Å and the set of van der Waals radii used by Rashin et al.[21] to facilitate comparison. Results shown in Table 4 are in excellent agreement with those of Rashin et al. [21] For two proteins (4pti[23] and 2lyz[24]) our program found additional cavities that the approximate algorithm missed due to their small size, reflected by very similar respective surface areas. A recent study based on exact calculation of surface areas and volumes[25] reported discrepancies between the output of their program VOLBL and Rashin et al. [21] The difference between the sets of van der Waals radii used in these articles can readily explain observed differences.

Calculations of Average Curvature of MS

Several studies have pointed out that thermodynamic and structural characteristics of solvation of protein surface depend not only on the extent of solvent-exposed surface area but also on local surface curvature. Taking into account a simple curvature-depen-

Table 3. Comparison of AS and MS Areas Computed Using FastSurf, ANAREA, and MSDOT.

	ASA, Å^2		MSA, Å^2		
Protein[a]	FastSurf	ANAREA[b]	FastSurf	MSDOT[b]	FastSurf ASA Calculation Time (s)[c]
1eca	7234	7132	6825	6850	1.10
1nxb	4083	4059	3433	3403	0.46
1rbc	6755	6733	5968	5959	0.84
1rbe	6702	6651	5893	5873	0.84
1rbf	6695	6660	5977	5939	0.86
1rbg	6713	6650	5962	5877	0.83
1rbh	6728	6667	6003	5907	0.88
1rbi	6690	6655	5860	5812	0.84
2act	9124	9193	8238	8920	1.41
2cha	11068	11073	10302	10796	1.65
2lyz	6730	6766	5981	6249	0.88
2ptn	9410	9466	8675	9381	1.41
2rns	6713	6691	5880	5782	0.84
2sn3	4214	4235	3476	3529	0.41
3cyt	11846	11741	11012	10798	1.75
3rn3	6952	6908	6155	6036	0.87
4pti	3963	4020	3215	3346	0.38
5mbn	8417	8433	7765	8048	1.27
1arb	9696	9706	9070	9777	1.53
1cau	20245	20172	19103	19856	3.63
1cse	12918	12777	12132	12366	2.35
1ecd	7203	7113	6805	6820	1.11
1icm	7598	7520	7362	7390	1.05
1mbd	8478	8501	7869	8152	1.25
1plc	5128	5113	4350	4368	0.61
1rro	6061	6000	5384	5380	0.82
1thm	10010	9914	9272	9455	1.72
1ycc	6627	6600	6094	5961	0.82
3sdh	14326	13747	14266	13766	2.73
4gcr	9040	8964	8375	8472	1.36
5p21	8673	8639	7776	8202	1.16

In all calculations for this table, we used the set of van der Waals radii from Richmond and Richards[20] and a probe radius of 1.2 Å to facilitate comparison with other programs.

[a]The protein structures are: 1eca, hemoglobin aquo met,[28] 1nxb, neurotoxin b,[29] 1rbc, ribonuclease S,[30] 1rbe, ribonuclease S M13F mutant,[30] 1rbf, ribonuclease S M13G mutant,[30] 1rbg, ribonuclease S M13I mutant,[30] 1rbh, ribonuclease S M13L mutant,[30] 1rbi, ribonuclease S M13V mutant,[30] 2act, actinidin,[31] 2cha, α chymotryprin A,[32] 2lyz, lysozyme,[24] 2ptn, trypsin,[33] 2rns, ribonuclease S,[34] 2sn3, scorpion neurotoxin,[35] 3cyt, cytochrome c,[36] 3rn3, ribonuclease A,[37] 4pti, trypsin inhibitor,[23] 5mbn, myoglobin,[38] 1arb, achromobacter protease I,[39] 1cau, {PRIVATE "TYPE=PICT;ALT=iota"}carrageenan,[40] 1cse, subtilisin carlsberg,[41] 1ecd, hemoglobin deoxy,[28] 1icm, intestinal fatty acid binding protein complexed with myristate,[42] 1mbd, myoglobin,[43] 1plc, plastocyanin,[44] 1rro, rat oncomodulin,[45] 1thm, thermitase,[46] 1ycc, cytochrome c,[47] 3sdh, hemoglobin I,[48] 4gcr, gamma-B crystallin,[49] 5p21, c-H-Ras p21 protein.[50]

[b]The areas are taken from Liang et al.[15]

[c]The calculation times do not include input-output. All calculations were performed using FastSurf on a Dell Optiplex Gxa equipped with a Pentium II 266 MHz processor.

dent correction factor in the surface tension coefficient allows one to reconcile at least a part of the apparent discrepancy between transfer energies of a hydrocarbon series into aqueous solution and macroscopic surface energies at the water-oil interface.[17] The scaled particle theory[26] or simulations of transfer experiments using a simple water model[18] indicate that solvation thermodynamics are strongly curvature dependent, although in a more complex way, largely dictated by the propensity of surface waters to maximize the number of their hydrogen bonds. Packing efficiency of both surface atoms and water molecules also depends on local curvature.[27] Our program provides a way to compute rigorously average curvatures of each surface atom, thus providing a parameterization, which can be used in calculation of curvature-corrected solvation energies as well as in docking algorithms.

As an example of average MS curvature calculation, we used eq. (16) to calculate the curvatures of nonpolar, polar, and total

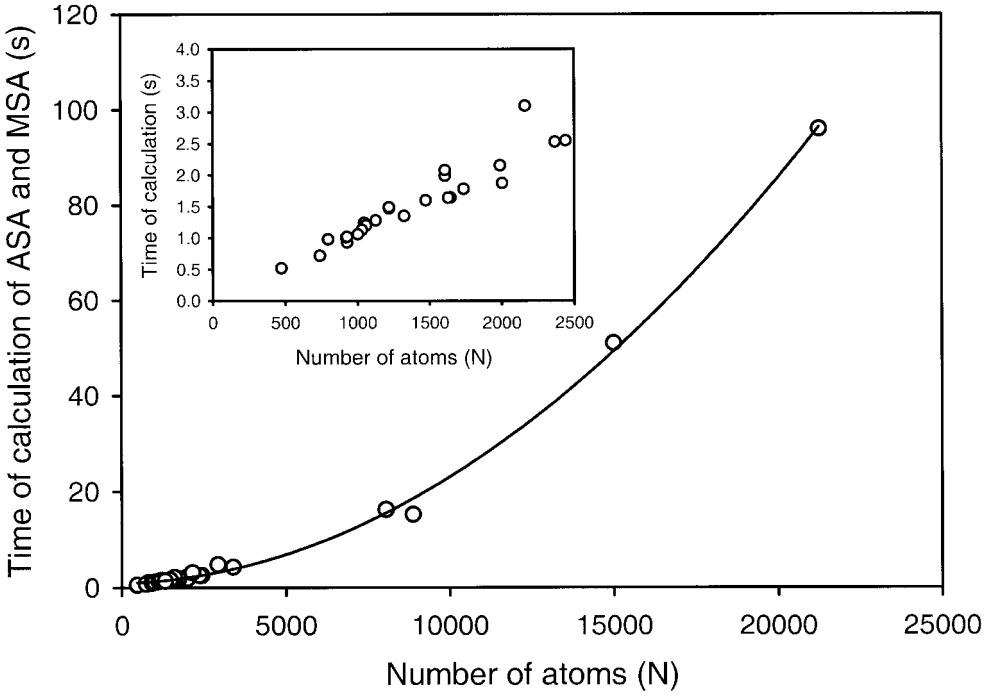

Figure 1. Times of computation of accessible and molecular surface areas for a set of protein structures from the Protein Data Bank. The calculations were performed on a Dell Optiplex GXa equipped with a Pentium II 266 MHz processor and running Windows NT 4.0. The times do not include the processes of input and output of data, each performed only once at the beginning and at the end, respectively. The computation times for the subset of these structures, smaller than 2500 atoms, are shown in the inset. All calculations were performed using program FastSurf with the set of van der Waals radii from Richards[2] and probe radius of 1.4 Å.

protein surfaces for a subset of unrelated proteins from Table 5. The reciprocal of the average curvature of the total MS of a protein is equal to the radius of a sphere, which has the same MS as the protein. Because a typical globular protein is very compact and therefore roughly spherical in shape, this effective radius well approximates the size of the protein. Interestingly, in all cases where the average curvature of nonpolar MS is negative (convex surface), it is much smaller in magnitude (the surface is less convex) than that of polar MS, and in many instances the average curvature of nonpolar surface is positive, that is, this surface is, on average, concave. This difference is very significant because nonpolar MS area is larger than its polar counterpart and the average curvature of the entire protein surface is negative due to its overall convexity, as shown in Table 5. Therefore, as expected, average curvature of polar surface is always negative (Table 5). Because AS is displaced from MS along its normal by a probe radius, the difference between the average curvatures of nonpolar and polar surfaces should be reflected in their AS and MS areas. This is indeed the case. The larger ratios of MS to AS areas of nonpolar surfaces than those of polar surfaces (Table 5) indicate that a nonpolar surface is less convex (or more concave). Nonpolar MS areas are invariably very similar to nonpolar AS areas, whereas polar molecular areas are about 25% smaller than their accessible counterparts. This difference in average curvatures has not been previously reported in literature. It is a consequence of different energetics of interaction of polar and nonpolar protein surfaces with aqueous solvent. Even though the magnitudes of the average curvatures of polar and nonpolar MS are fairly small (<0.1 Å^{-1}), corresponding to the effective radius of 10 Å or larger, these values are averages of a very large number of curvatures of individual atoms, most of which are quite large in magnitude. For example, 70% of surface atoms of cytochrome c (PDB code 3cyt[36]) have

Table 4. Analysis of Inaccessible Cavities in Protein Interior.

Protein[a]	Molecular Mass	Number of Cavities[b]	Total AS Area,[b] Å^2
4pti	6,000	3 (2)	20 (20)
1nxb	7,000	0 (0)	0 (0)
1sn3	7,000	2 (2)	6 (6)
1rn3	11,000	5 (5)	3 (3)
2lyz	13,000	10 (8)	53 (50)

In all calculations using SurfRace for this table, we used the set of van der Waals radii from Rashin et al.[21] and a probe radius of 1.2 Å.[21]
[a]The protein structures include: 1sn3, scorpion neurotoxin,[50] 1rn3, ribonuclease A.[51] The other structures are described in Table III.
[b]The values in parentheses are taken from Rashin et al.[21]

Table 5. Average Curvature of MS.

Protein	Average Curvature, Å^{-1}	Nonpolar MS Area,[a] Å^2	Nonpolar MS Curvature, Å^{-1}	Polar MS Area,[a] Å^2	Polar MS Curvature, Å^{-1}
1nxb	−0.0713	1959 (0.88)	−0.0527	1349 (0.72)	−0.0984
1plc	−0.0628	2465 (0.98)	−0.0242	1800 (0.68)	−0.1156
2rnt	−0.0365	2546 (0.98)	−0.0117	2231 (0.76)	−0.0648
2ins	−0.0326	2891 (0.98)	−0.0027	1987 (0.74)	−0.0760
1rro	−0.0250	2941 (1.03)	0.0126	2303 (0.75)	−0.0730
1lzt	−0.0516	3046 (1.00)	−0.0127	2539 (0.72)	−0.0982
2lyz	−0.0445	3103 (1.00)	−0.0057	2624 (0.72)	−0.0904
1rbc	−0.0500	3182 (0.94)	−0.0286	2467 (0.75)	−0.0777
1ycc	−0.0158	3458 (1.01)	0.0058	2347 (0.77)	−0.0477
2act	−0.0411	4535 (0.99)	−0.0162	3332 (0.75)	−0.0751
2ptn	−0.0169	4624 (1.02)	0.0083	3685 (0.79)	−0.0484
1thm	−0.0256	4702 (1.05)	0.0116	4026 (0.76)	−0.0690
1kvd	−0.0139	5400 (0.99)	−0.0070	4655 (0.85)	−0.0220
2cha	−0.0101	5558 (1.06)	0.0239	4133 (0.75)	−0.0558
3app	−0.0056	6265 (1.02)	0.0078	5261 (0.81)	−0.0217

In all calculations for this table, we used the set of van der Waals radii from Richards[2] and a probe radius of 1.4 Å.
[a]The values in parentheses are the ratios of MS areas to their respective AS areas.

average curvatures larger than 0.1 Å^{-1} in magnitude. For such highly curved surfaces, concave surface will interact with fewer water molecules than convex surface of the same area, which may explain at least a part of the difference between the average curvatures of polar and nonpolar surfaces. A more quantitative analysis of the origins of the curvature difference is in progress.

In docking applications, the requirement of complementarity of the interface surfaces dictates that the average curvatures of interacting surfaces be opposite in sign and equal in magnitude. Preliminary calculations in this laboratory (Tsodikov, unpublished results) indicate that this criterion is satisfied to a good accuracy in several known protein-protein complexes. Therefore, we expect that average curvature will be a useful parameter to use in docking algorithms.

Conclusions

We report a novel program for calculating exact AS and MS (excluded) areas and average surface curvature of MS and interior cavities. The program is fast, robust, and portable to practically any existing computational platform. To our knowledge, it is the only completely independent exact AS and MS area calculation program that also analyzes cavities in the protein interior, which runs conveniently on PCs. We applied the concept of average curvature to MS and showed that this parameter is important in determining energetics of interactions of protein surface with solvent. We expect that our program will find widespread application in stand-alone calculation of surface area and curvature as well as in time-consuming energy minimization routines.

Programs SurfRace and FastSurf are available upon request from the corresponding author (web address: http://monte.biochem.wisc.edu/~tsodikov).

Acknowledgments

We thank Drs. Alexei Podtelezhnikov and Irina Shkel for helpful discussions and the reviewers for their suggestions.

References

1. Lee, B.; Richards, F. M. J Mol Biol 1971, 55, 379.
2. Richards, F. M. Annu Rev Biophys Bioeng 1977, 6, 151.
3. Spolar, R. S.; Record, M. T., Jr. Science 1994, 263, 777.
4. Livingstone, J. R.; Spolar, R. S.; Record, M. T., Jr. Biochemistry 1991, 30, 4237.
5. Jackson, R. M.; Sternberg, M. J. J Mol Biol 1995, 250, 258.
6. LiCata, V. J.; Allewell, N. M. Biochemistry 1997, 36, 10161.
7. Raschke, T. M.; Tsai, J.; Levitt, M. Proc Natl Acad Sci USA 2001, 98, 5965.
8. Courtenay, E. S.; Capp, M. W.; Anderson, C. F.; Record, M. T., Jr. Biochemistry 2000, 39, 4455.
9. Das, B.; Meirovitch, H. Proteins 2001, 43, 303.
10. Richmond, T. J. J Mol Biol 1984, 178, 63.
11. Connolly, M., J. Appl. Cryst 1983, 16, 548.
12. do Carmo, M. P. Differential Geometry of Curves and Surfaces. Prentice-Hall: Englewood Cliffs, NJ, 1976.
13. Gibson, K. D.; Scheraga, H. A. Mol Phys 1987, 62, 1247.
14. Fraczkiewicz, R.; Braun, W. J Comp Chem 1998, 19, 319.
15. Liang, J.; Edelsbrunner, H.; Fu, P.; Sudhakar, P.V.; Subramaniam, S. Proteins 1998, 33, 1.
16. Sanner, M. F.; Olson, A. J.; Spehner, J. C. Biopolymers 1996, 38, 305.
17. Sharp, K. A.; Nicholls, A.; Fine, R. F.; Honig, B. Science 1991, 252, 106.
18. Southall, N. T.; Dill, K. A. J Phys Chem 2000, 17, 1326.
19. Dineen, S. Multivariate Calculus and Geometry; Springer: 1998.
20. Richmond, T.; Richards, F. M. J Mol Biol 1978, 119, 537.
21. Rashin, A.; Iofin, M.; Honig, B. Biochemistry 1986, 25, 3619.
22. Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.;

Weissig, H.; Shindyalov, I. N.; Bourne, P. E. Nucleic Acids Res 2000, 28, 235.
23. Marquart, M.; Walter, J.; Deisenhofer, J.; Bode, W.; Huber, R. Acta Crystallogr 1983, B39, 480.
24. Diamond, R. J Mol Biol 1974, 82, 371.
25. Liang, J.; Edelsbrunner, H.; Fu, P.; Sudhakar, P. V.; Subramaniam, S. Proteins 1998, 33, 18.
26. Floris, F. M.; Selmi, M.; Tani, A.; Tomasi, J. J Chem Phys 1997, 107, 6353.
27. Gerstein, M.; Chothia, C. Proc Natl Acad Sci USA 1996, 93, 10167.
28. Steigemann, W.; Weber, E. J Mol Biol 1978, 127, 309.
29. Tsernoglou, D.; Petsko, G. A.; Hudson, R. A. Mol Pharmacol 1978, 14, 710.
30. Varadarajan, R.; Richards, F. M. Biochemistry 1992, 31, 12315.
31. Baker, E. N.; Dodson, E. J. Acta Crystallogr 1980, A36, 559.
32. Birktoft, J. J.; Blow, D. M. J Mol Biol 1972, 68, 187.
33. Walter, J.; Steigemann, W.; Singh, T. P.; Bartunik, H.; Bode, W.; Huber, R. Acta Crystallogr 1982, B38, 1462.
34. Zhao, B.; Carson, M.; Ealick, S. E.; Bugg, C. E. J Mol Biol 1992, 227, 239.
35. Takano, T.; Dickerson, R. E. Proc Natl Acad Sci USA 1980, 77, 6371.
36. Howlin, B.; Moss, D. S.; Harris, G. W. Acta Crystallogr 1989, A45, 851.
37. Takano, T. Methods and Applications in Crystallographic Computing; Oxford University Press: 1984; p 262.
38. Tsunasawa, S.; Masaki, T.; Hirose, M.; Soejima, M.; Sakiyama, F. J Biol Chem 1989, 264, 3832.
39. Ko, T.-P.; Day, J.; McPherson, A. Acta Crystallogr 2000, D56, 411.
40. Bode, W.; Papamokos, E.; Musil, D. Eur J Biochem 1987, 166, 673.
41. Eads, J.; Sacchettini, J. C.; Kromminga, A.; Gordon, J. I. J Biol Chem 1993, 268, 26375.
42. Phillips, S. E.; Schoenborn, B. P. Nature 1981, 292, 81.
43. Guss, J. M.; Bartunik, H. D.; Freeman, H. C. Acta Crystallogr 1992, B48, 790.
44. Ahmed, F. R.; Rose, D. R.; Evans, S. V.; Pippy, M. E.; To, R. J Mol Biol 1993, 230, 1216.
45. Teplyakov, A. V.; Kuranova, I. P.; Harutyunyan, E. H.; Vainshtein, B. K.; Frommel, C.; Hohne, W. E.; Wilson, K. S. J Mol Biol 1990, 214, 261.
46. Louie, G. V.; Brayer, G. D. J Mol Biol 1990, 214, 527.
47. Royer, W. E., Jr. J Mol Biol 1994, 235, 657.
48. Najmudin, S.; Nalini, V.; Driessen, H. P. C.; Slingsby, C.; Blundell, T. L.; Moss, D. S.; Lindley, P. F. Acta Crystallogr 1993, D49, 223.
49. Pai, E. F.; Krengel, U.; Petsko, G. A.; Goody, R. S.; Kabsch, W.; Wittinghofer, A. EMBO J 1990, 9, 2351.
50. Almassy, R. J.; Fontecilla-Camps, J. C.; Suddath, F. L.; Bugg, C. E. J Mol Biol 1983, 170, 497.
51. Borkakoti, N.; Moss, D. S.; Stanford, M. J.; Palmer, R. A. J Crystallogr Spectrosc Res 1984, 14, 467.