An Adaptive Grid Based Method for Computing Molecular Surfaces and Properties

Chandrajit Bajaj and Vinay Siddavanahalli

November 3, 2006

Abstract

We present an adaptive grid based algorithm to compute a family of relevant molecular surfaces. Molecular interfaces are important in simulations and visualization involving biomolecules. The Richards surface has traditionally been used as a good approximation to the surface, and defined as the surface formed by the inner facing part of a solvent probe atom rolling along the van der Waals surface of the molecule. Computing and representing this surface has traditionally involved complex geometrical data structures like alpha shapes. Adaptive and uniform trilinear grids are commonly used in various simulations involving interactions of molecules or computation of electrostatics and other energy terms. We make use of this grid directly to compute the Molecular Surface and properties like area, volume, curvatures, surface atoms and other surfaces. We compare geometrical and biochemical properties with other methods as a validation.

1 Molecular Surface Definitions

Explicit surface definitions as the interface between the solvent and proteins have been given since 1970s. Since it is easier to handle implicitly defined models mathematically, different implicit approximations to these surfaces have been developed.

1.1 van der Waals and Lee Richards Surface Definitions

The most common model for molecules is as a collection of atoms represented by spheres, with radii equal to their van der Waals radii. The surface of the set of spheres is known as the van der Waals surface. Lee and Richards introduced the concept of accessibility to the solvent. Proteins are not isolated, but commonly present in solutions, especially water. Also, the van der Waals surface contained too many internal atoms and patches which are not accessible by the solvent. Hence, Lee and Richards gave a new definition for the molecular surface as the surface accessible to the solvent [32]. They modeled water molecules as spheres with radius 1.4Å, and considered the locus of the center of one such 'probe', as it rolled along the protein surface as the Solvent Accessible Surface (SAS). Richards then gave a more commonly used definition for molecular surface as a set of contact and reentrant patches in [42]. A probe solvent sphere, rolling over the atoms of a protein defines a region in which none of its points pass through. The boundary of this volume is continuous and defines a new molecular surface. This surface is composed of convex patches where the probe touches the atom surfaces, concave spherical patches when the probe touches more than 2 atoms simultaneously and toroidal patches when the probe rolls between two atoms. Connolly called this as an alternative definition of the SAS surface in [18], but is now commonly known as the Solvent Contact Surface (SCS), or Solvent Excluded Surface (SES) or simply the Molecular Surface. These surfaces, for a 3 atom example is shown as a 2D cross section in figure 1. We also provide analogous volume definitions for each surface.
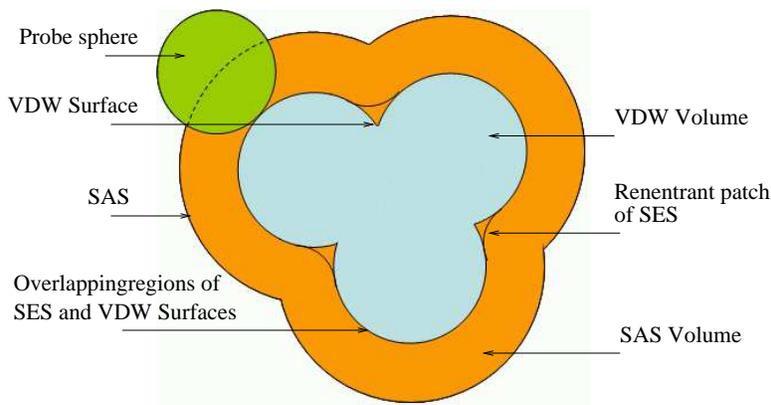
Figure 1: The different molecular surfaces and regions are shown for a 3 atom model in 2D. The SAS surface is the locus of the center of the rolling probe sphere. The VDW surface is the exposed union of spheres representing atoms with their van der Waals radii and contains the VDW volume. The lower side of the rolling probe defines the smooth SES which contains parts of the VDW surface and reentrant patches. We also define the SAS volume as the region between the SAS and SES. The region between the SAS and VDW volumes is later refered to as the SES volume.

2 Adaptive Grid Based SES Construction

We provide a new algorithm to compute the molecules SES and other related properties, which we will use in the docking framework. Our main goals are efficient representation for operations required during docking, accurate surface definition and ease of implementation.

2.1 Related Work

Since Richards introduced the SES definition, a number of techniques have been devised to compute the surface, both static and dynamic, implicit and explicit. Connolly introduced two algorithms to compute the surface. First, a dot based numerical surface construction and second, an enumeration of the patches that make up the analytical surface (See [17], [18] and his PhD thesis). In [50], the authors describe a distance function grid for computing surfaces of varying probe radii. Our data structure contains approaches similar to their idea. A number of algorithms were presented using the intersection information given by voronoi diagrams and the alpha shapes introduced by Edelsbrunner [23], including parallel algorithms in [49] and a triangulation scheme in [1]. Fast computations of SES is described in [46] and [45], using Reduced sets, which contains points where the probe is in contact with three atoms, and faces and edges connecting such points. Non Uniform Rational BSplines (NURBs) descriptions for the patches of the molecular surfaces are given in [6], [5] and [7]. You and Bashford in [53] defined a grid based algorithm to compute a set of volume elements which make up the Solvent Accessible Region.

3 Signed Distance Function based Family of Surfaces

We define a volume function Φ and use its contours to provide a family of molecular surfaces. Consider the union of atoms of the molecule $\cup B$. Inflate each atom b in this set by the probe radius (solvent radius) r_p to give the new complex $\cup B_{r_p}$. Let its boundary be Γ_B. Let Φ define the signed distance function of ΓB, such that the interior (closer to van der Waals) is given a positive sign. Let all regions within the atom (see [53] for definitions) be given a constant positive high value H.

Observations and lemmas:

- Isosurfaces S_I with isovalues $I : 0 \leq I \leq H$ form a family of surfaces.

- $\Gamma_B = S_0$, as defined by Lee and Richards, is the SAS of the molecule.

- S_{r_p} is the SES.

- $S_{x \to H^-}$ is the van der Waals surface.

- $\{x : 0 \leq \Phi(x) \leq H\}$ defines a volume exclusion function, which can be convenient to use in electrostatic computations.

- The region $\{x : -r_p \leq \Phi(x) \leq r_p\}$ has a high probability for the presence of surface atoms of a protein docked to the current one.

The above observations point to the obvious advantages in using such a definition for our molecular structure representation for docking. Let us further examine some of them in detail.

$\Gamma_B = S_0$ is the SAS, and S_{r_p} is the SES: By definition of the SAS, it is the locus of the center of the probe as it rolls over the spherical atoms of the protein. But it should be noted that the grid based definition also includes holes, which may be removed if necessary. The SES surface is always defined by points on the probe. It is in fact the boundary of the region accessible to any part of the probe radius. Hence, it is always at a constant distance of r_p away from the locus of the center. Therefore, our third observation follows. Again, holes are included in our definition and need to be removed if required.

$\{x : 0 \leq \Phi(x) \leq H\}$ provides a volume exclusion function: Volume exclusion functions are used in setting up dielectric constant for electrostatic computations. The twin requirements of smoothness at the boundary and accuracy in modeling the SES are not met by many of the definitions in practise today. Our definition is provides a 'sufficiently' smooth function around the SES (Φ is smooth in the radial direction), and contains the SES within it.

Isosurfaces S_I with isovalues $I : 0 \leq I \leq H$ form a family of surfaces At the extremes isovalues, we have the SAS and the VDW surfaces, and the SES lies in between them at an isovalue of r_p. We show the SES surface and other surfaces surrounding it in figure 5.

Interface of docked ligand is in the region $\{x : -r_p \leq \Phi(x) \leq r_p\}$: For good shape complementarity, as observed in docked complexes, atoms of the ligand must lie close to the surface of the protein. The above 'skin' definition provides a functional representation for such a region, as it defines the region where a probe sphere is in touch with the protein.

4 Algorithm

Let us consider a grid G in which the molecule is embedded to have a maximum and minimum grid spacing, h_{max}, h_{min}. Let the dimension with the lower resolution be N^3. Initially, the grid is uniformly divided using h_{max} as the grid spacing. Then:

- **Top down subdivision:**
  1. Insert each atom $b_i, i = 1..M$ into G, subdividing if necessary.
  2. With each insertion, update locally, points $\vec{p} \in G$ as belonging to V_{SAS}, V_{VDW}.
  3. Compute the boundaries S_{VDW}, S_{SAS}.

- **Bottom up collapse**
  1. New points created by the previous steps in the grid, and buried in atoms interiors are collapsed to make the grid sparser in a bottom up fashion.

- For each point $\vec{p}$ around a point classified as S_{VDW}, search in a local region with extent r_p for a S_{SAS} boundary cell. Find the closest distance of the point from the S_{SAS} boundary.

Details on each step is given below.

1. **Vertex classification** Each atom is inserted into the grid. If we start with a single node in the adaptive mesh, subdivision is performed as we insert each atom in an adaptive manner. With the insertion of each atom, the vertices around the center of the atom are classified as belonging to inside the V_{SAS} or the V_{VDW}. Vertices classified as V_{VDW} are fixed while vertices marked V_{SAS} could be updated with the insertion of new atoms. We use the method described by [2] for sphere-cube intersection tests. The cost of this insertion is linear in the number of atoms and cubic in the resolution of the grid: $O(Mh_{max}^3)$.

2. **Boundary cell detection** We examine the classification of the eight corners of each cell of the grid. If some vertices belong to the inside of a volume and others to the outside, we mark the cell as a boundary cell. This operation is linear in the number of cells of the grid $O((N-1)^3)$. Each boundary cell is given an index.

3. **Adaptive subdivision of S_{SAS}** Each boundary cell which contains more than three atoms contributing to it is subdivided up to a user defined resolution. The index of each atom intersecting a cell is kept in a linked list associated with that cell. Using that list, we classify each subdivided vertex as belonging to the interior of the V_{SAS} or not. Using a technique similar to obtain boundary cells, we generate a list of finer boundary cells in the subdivided cells. The maximum cost of this operation is $O((N-1)^3(h_{max}/h_{min})^3)$, although the average case cost should be much smaller as only the boundary cells are involved.

4. **S_{SES} computation**

   The cells around each vertex in V_{SAS} and S_{VDW} is searched for the S_{SAS}. If there is a cell with only one intersecting atom, we find the exact distance from the vertex to the spherical patch of S_{SAS} in that cell and stored at the vertex. If a closer distance it is found, the stored distance is updated. If we are searching a cell which contained more than one atom, and hence subdivided, we just take the minimum distance from the center of all the subdivided cell to the vertex in question as the distance of the spherical patch in the cell to the vertex. The cost of this search will vary as r_p^3, the number of boundary van der Waals cells in the volume and the accuracy desired (as provided by h_{min}).

4.1 Spherical Patch Intersection

Let us define a sphere as having a center $\vec{c} = \{c_x, c_y, c_z\}$ and radius r. Define a cube with points $\vec{a}_1, .., \vec{a}_8$

Equation of arc of intersection of sphere and face of cube The intersection is always an arc of a circle. We will obtain the center, radius of the circle and the intersection points. We will consider only a face parallel to the xy plane. Other cases should follow similarly. The point of projection of $\vec{c}$ to the plane containing the face is $\vec{p} = \{c_x, c_y, \texttt{z coordinate of face}\}$. This point is the center of the circular arc. The radius using Pythagoras theorem is $\sqrt{r^2 - dist(\vec{p}, \vec{c})^2}$. The intersection points on the edges, if any is now the intersection of this circle with the line containing the edge, and checking whether the points lie within the edge.

Shortest distance of point to a circular arc Let the point be $\vec{p}$, the center, radius of the arc be $\vec{p}_1, \vec{r}$ and the two end points of the arc be $\vec{p}_2$ and $\vec{p}_3$.

Lemma The shortest distance of a point $\vec{q}$ in a plane to a circular arc in the plane is:

- Point is outside the infinite sector defined by the arc. The shortest distance is : $|\vec{r} - dist(\vec{q}, p_1)|$.

- Otherwise, the shortest distance $= min(dist(q, p_2), dist(q, p_3))$.

Let $\vec{q}$ be the projection of $\vec{p}$ to the plane containing the circle. Let d_1 be the $dist(\vec{p}, \vec{q})$ and d_2 be the shortest distance of the arc from $\vec{q}$. Thus, the shortest distance from $\vec{p}$ to the circular arc is $\sqrt{(d_1)^2 + (d_2)^2}$.

Shortest distance of point to a spherical patch Here the spherical patch is in a cube, bounded by circular arcs. Consider the circle a boundary arc is part of. The center of the sphere and this circle will form an infinite cone. Hence the collection of boundary arcs form a collection of infinite cones.

Lemma The shortest distance of a point $\vec{p}$ to a spherical patch in a cube is:

- Point is inside each of the infinite cones. The shortest distance is : $|\vec{r} - dist(\vec{p}, \vec{c})|$.

- Otherwise, the shortest distance is the minimum of the shortest distances of the point to each of the bounding arcs.

4.2 Complexity

For M atoms (including B boundary atoms), smallest grid spacing h, grid length N, VDW radius r and solvent radius r_p, the timing complexity is

- SDF initialization: $O(N^3)$

- Insertion of atoms: $O(M(\frac{2(r+r_p)}{h})^3)$

- Boundary atom detection:

  - Uniform grid traversal: $O(N^3)$
  - Sphere traversal: $O(M(\frac{2(r+r_p)}{h})^3)$
  - Octree traversal: $O(log(N^3)B) \leq O(log(N^3)M)$

- Patch voxel distance computation: $O(M(\frac{2(r+r_p)}{h})^6 C)$, C is cost(dist(patch, voxel))

- Isocontouring for visualization: $O(N^3)$

4.3 Self Intersections in Patch Complex Model

A patch complex (consisting of convex, concave and toroidal patches) can be derived using our adaptive grid structure and SAS sphere intersection enumeration. But the patch complex is known to have problems of bad intersections. According to lemma 3, 4, 5, 6 and 7 from Bajaj et al [9], there are only two possible self intersections that occur in the commonly used rolling ball model:

- A toroid can self intersect with itself (Figure 10(a) in [9]).

- A concave patch can intersect with another in the case of a 3 atom model (Figure 9 in [9]).

In figure 2, we show the surface computed when two atoms are present, and moved close till they form a single surface patch. In the case of surfaces computed from the rolling ball model, we would instead get a self intersecting toroidal patch when the gap between the atoms becomes smaller than the diameter of the probe radius. This can be computed by looking at all pairs of intersecting SAS spheres, which is already given in our adaptive grids. To examine the intersection of two concave patches, we look at the three atoms model as shown in figure 3. Again, we get similar results compared to [9]. This case occurs when there are three intersecting SAS atoms, and can be enumerated by our grid.

(a) The toroidal patch is disjoint and there is no self intersection.

(b) As the atoms come closer, a well defined toroidal patch is created.
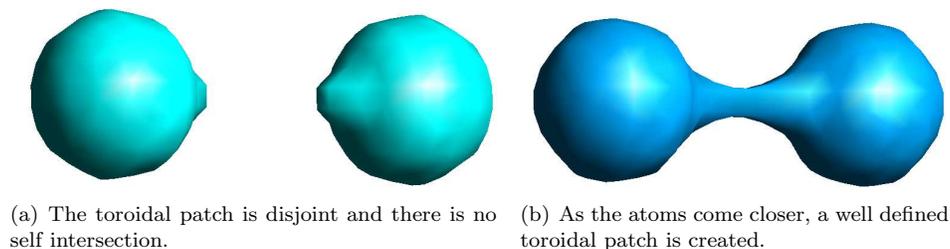
Figure 2: The solvent excluded surfaces of two atoms which come closer.

(a) The 2 concave patches are disjoint and there is no wrong intersection.

(b) As the atoms come closer, a well defined patch, similar to the approximations in [9] is created.

(c) At mutually closer distances, the topology changes and the center hole disappears.
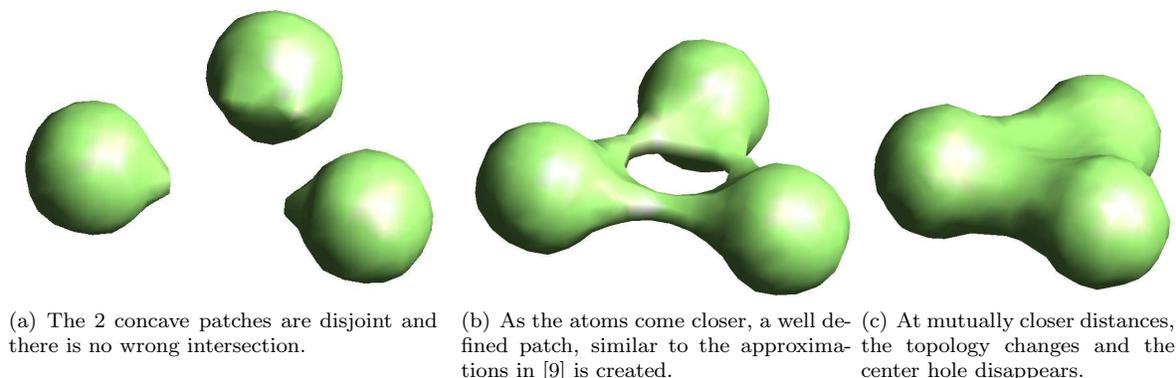
Figure 3: The solvent excluded surfaces of three atoms which come closer.

5 Operations Supported by the Adaptive Grid

1. **Surface atoms detection**

   Surface atoms are defined as those within a certain distance from the Molecular Surface. To obtain these atoms, we first compute the Molecular Surface. Next we search locally around each atom to find the distance of the atom from the surface. This operation is linear in the number of atoms and cubic in the resolution of the grid.

2. **Population of skin region**

   We define the skin region of one molecule as the region belonging to the probe as it rolls on the surface, and defined as Solvent Accessible Surface 2 Volume (V_{SAS2}). We define the skin implicitly as a set of spheres packing the region. The packing density is itself chosen to approximately equal the packing of the atoms belonging to the molecular surface. The region is defined over a trilinear grid in which the molecule is embedded. The grid spacing h is chosen to preserve the features of the molecule. Assuming that the interatomic distance is $\sim 1\mathring{A}$, we can use $h = 0.5\mathring{A}$. By finding the boundary vertices of the SAS, we can obtain potential centers for the skin spheres. A packing algorithm decides, based on the packing density required, if a potential center should contain an atom or not.

3. **Area volume computations**

   We use primal contouring to define the surface and volumes. The area under the surface is approximated by piecewise linear elements of the isocontour. The volume is approximated by the volume enclosed by that piecewise linear approximation. This cost is linear in the size of the grid.

4. **Curvature and normal computations**

3 atoms showing different surfaces and regions. S_{SAS}: dark blue, V_{SAS}: pink, S_{SES}: red, V_{SES}: light blue, S_{VDW}: yellow and V_{VDW}: green
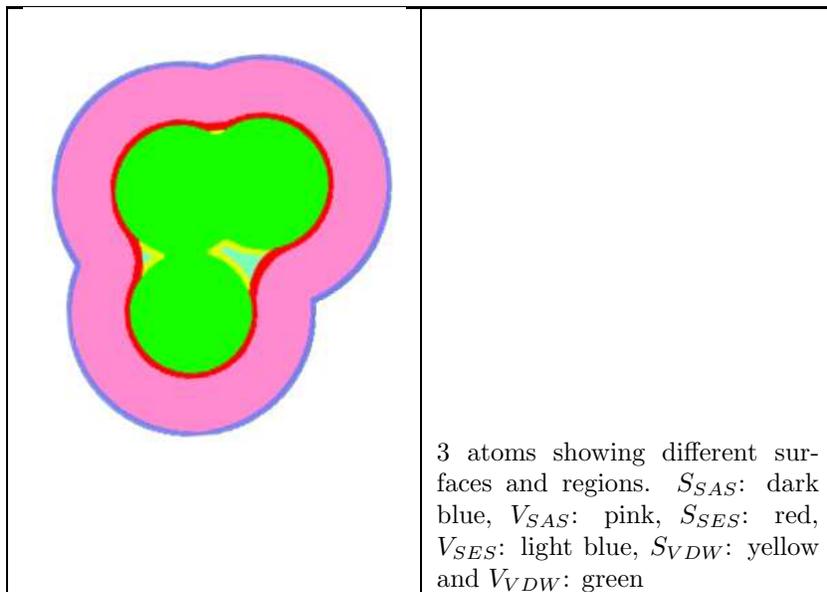
Table 1: A 2D cross section of the adaptive grid classification on a 3atom model.

These differential properties are computed using a two step process. Initially, when we propagate the distance from the S_{SAS}, we also store whether the nearest patch is the intersection of one, two or more spheres. In each case, we can analytically provide the answer to the curvatures. For example, for a sphere with radius r, the Mean and Gaussian curvatures are $-1/r$ and $1/r^2$ respectively. In the second step, we compute the derivatives from the isocontour. At points where the two vary significantly, we choose to keep the value provided by the differencing scheme as the signed-distance algorithm used is only an approximation.

6 Results

Region classification and construction of molecular surfaces Before we provide timing, geometric and functional properties and skin, surface regions, we present the results of our classification and signed distance function on a 3 atom model in figure 1. Using a relatively high resolution grid of 128^3, we classify grid points depending on the volume and surface it is part of, giving priorities of surface class over volumes and SES class over other surfaces. The figure is a 2D cross section of a volume rendering of the classified volume.

The Solvent Excluded Surface The solvent excluded surface is obtained as an isocontour with value equal to probe radius. In figure 4, we show colored visualizations of four different molecules.

Family of surfaces In figure 5, we show four different surfaces computed from the adaptive grid, at four different isovalues. The myoglobin molecule, 101m.pdb, is used as a test case. At a distance of 0, we get the SAS surface, which is the union of spheres model, with each atom represented as a sphere with radius equal to the sum of its radius and a probe radius. In this example, we used a probe radius of 1.4Å. As we go further away, we get a smooth deformation of the SAS surface to the SES surface, as shown in the different figures. Since we are interested in the SES, we do not compute further in practise, but

PDB Id	Number of atoms	time (64^3)	time (128^3)
3sgb	1912	11	85
1brc	2197	6	58
2ptc	2243	6	53
2kai	2267	7	74
3tpi	2313	6	54
1tab	2387	9	72
1ppf	2520	7	63
4cpa	2739	10	85
1mkw	4844	8	60

Table 2: Times (in seconds) taken to compute the adaptive grid based surfaces and volume regions for different initial grids which are adaptively subdivided to a depth of 3.

in theory, higher isovalues will take us closer to the van der Waals surface. This example shows the utility of our method as a volume exclusion function for computing electrostatics, which needs a smooth decay at the SES boundary.

Timing   The cost of the algorithm depends on the depth of the adaptive grid, the resolution of the initial base grid and the size of the molecule. In table 2, we provide the time taken to compute the properties on the grid, including surfaces and demarking volumetric regions for different molecules and grid sizes. As the number of atoms increase, the time taken increases, but the fixed output grid size reduces the number of relevant search points within the SAS and VDW regions. Hence there is no direct correlation seen between the two. If the grid resolution can be chosen depending on the molecule size, then the time would increase monotonically with the number of atoms for molecules with similar distribution of atoms (say for a set of globular proteins).

Surface atoms detection   The surface atoms of three proteins from the complexes, anti-idiotypic fab (1iai.pdb), hemagglutinin (2vir.pdb) and bob-white quail lysosyme (1bql.pdb) are visualized in figure 6. The interior atoms are colored by the residue they belong to while the outer surface atoms all have an orange color. We show a cutoff of the three molecules to reveal the surface and interior.

SAS² skin region construction   From the same above three complexes (1iai,2vir and 1bql), we extract the second protein and compute the skin regions (see figure 7) defined by the volume where the probe is present and touching the molecule. This region is used later in docking as it represents a volume where the interface atoms from the docking protein have a high probability of being present.

7   Geometric and Functional Comparison of Molecular Surfaces

Results from the surface construction algorithms are compared using both geometric and electrostatic properties. In each case, we compare the values to either exact known analytical solutions, or output from standard software. Four different surfaces are constructed:

1. Solvent excluded surface, constructed as an isocontour with isovalue 1.4 from our adaptive grid.

2. A sum of Gaussians, computed using a discrete set of radii, with rate of decays (B) of:

(a) **B = −2.3** : This value is used in the literature as one which gives a good approximation of the volume enclosed [43].

(b) **B = −1.0** : By reducing the value of B, the Gaussian is made smoother, leading to a lower resolution map.

(c) **B = −3.0** : We look at a sharper Gaussian to study the trend of properties as we move about the standard value of -2.3.

Datasets: We use a set of 71 complexes. Properties are computed and compared of both the complex and its two individual proteins. Hence, overall, we use 213 proteins in our dataset. The complex's Id in the figures is appended with a 'C', while the two proteins in it are appended with '1' and '2'.

7.1 Geometric comparison

The geometry of the surface plays an important role in its structural and functional properties. Two of the geometric properties we consider are areas and volumes:

- **Area of the surface:** The area of the surface is computed using a simple approximation by summing the area of triangles contained in the isocontours. MSMS [45] computes a analytical solution to the solvent excluded surface area. We compare our results with theirs.

- **Volume enclosed by the surface:** The volume enclosed is computed by summing partial or full contributions of voxels in the grid. If $n, n \leq 8$ grid corners are included in the volume, then $n/8$ of the voxels volume is counted.

In figures 8 and 9, we plot the areas (in Å^2) of five different surfaces (our four and the numerical value computed by MSMS) with the analytical solvent excluded surface area given by MSMS. While using MSMS, we set the probe radius to 1.4 and do not change the default values of grid spacing etc. From the figures, we see that the adaptive grid and the MSMS numerical values are just slightly lower than the analytical value. Among the sum of Gaussian surfaces, a rate of decay of 1.0, which provides a smooth surface was seen to be a good approximation, while sharper Gaussians tends to increase the surface area due to the increase in 'bumps' and 'valleys'.

Volumes (in Å^3) enclosed by the solvent excluded surface are compared in figures 10 and 11. The adaptive grid algorithm and sums of sharper Gaussians, including rates of decay of $-2.3, -3$, are seen to be close to the numerical value computed by MSMS. Unfortunately, the volume under the smoother surface defined by a Gaussian sum with decay rate of -1.0 is seen to overestimate the volume, as expected.

8 Conclusion

The new adaptive grid algorithm computes a family of surfaces for a given molecule, regions of interest and aids in computing areas, volumes and other geometrical properties. Unlike traditional techniques of computing molecular surfaces that involve complex data structures, we show how smooth approximations with no self intersections can be obtained using a more conventional adaptive grid data structure.

References

[1] N. Akkiraju and H. Edelsbrunner. Triangulating the surface of a molecule. *Discrete Applied Mathematics*, 71(1-3):5–22, 1996.

[2] J. Arvo. A simple method for box-sphere intersection testing. pages 335–339, 1990.

[3] D. Attali and H. Edelsbrunner. Inclusion-exclusion formulas from independent complexes. In *SCG '05: Proceedings of the twenty-first annual symposium on Computational geometry*, pages 247–254, New York, NY, USA, 2005. ACM Press.

[4] C. Bajaj, P. Djeu, V. Siddavanahalli, and A. Thane. Texmol: Interactive visual exploration of large flexible multi-component molecular complexes. In *VIS '04: Proceedings of the conference on Visualization '04*, pages 243–250, Washington, DC, USA, 2004. IEEE Computer Society.

[5] C. Bajaj, H. Y. Lee, R. Merkert, and V. Pascucci. Nurbs based b-rep models for macromolecules and their properties. In *Proceedings of the fourth ACM symposium on Solid modeling and applications*, pages 217–228. ACM Press, 1997.

[6] C. Bajaj, V. Pascucci, A. Shamir, R. Holt, and A. Netravali. Multiresolution molecular shapes. Technical report, TICAM, Univ. of Texas at Austin, Dec. 1999.

[7] C. Bajaj, V. Pascucci, A. Shamir, R. Holt, and A. Netravali. Dynamic maintenance and visualization of molecular surfaces. *Dis. App. Math.*, 127(1):23–51, 2003.

[8] C. Bajaj and V. Siddavanahalli. Fast error-bounded surfaces and derivatives computation for volumetric particle data. ICES 06-03, U.T. Austin, 2006.

[9] C. Bajaj, G. Xu, R. Holt, and A. Netravali. Nurbs approximation of a-splines and a-patches. *International Journal of Computational Geometry and Applications*, 13(5):359–389, November 2003.

[10] J. F. Blinn. A gen. of alg. surface drawing. *ACM TOG*, 1(3):235–256, 1982.

[11] J.R. Blinn, D.C. Rohrer, and G.M. Maggiora. Field-based similarity forcing in energy min. and mol. matching. *Pac. Symp. on Biocomp.*, 4:415–425, 1999.

[12] S. Boys. Electronic wave functions, i: A general method of calculation for the stationary states of any molecular system. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 200(1063):542–554, February 1950.

[13] R. Bryant, H. Edelsbrunner, P. Koehl, and M. Levitt. The area derivative of a space-filling diagram. *Discrete & Computational Geometry*, 32(3):293–308, 2004.

[14] F. Cazals, F. Chazal, and T. Lewiner. Molecular shape analysis based upon the morse-smale complex and the connolly function. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*, pages 351–360, New York, NY, USA, 2003. ACM Press.

[15] F. Cazals and F. Proust. On the topology and the geometry of (voronoi) molecular interfaces. part i: algorithmsy. Inria technical report, INRIA, October 2004.

[16] F. Cazals and F. Proust. Revisiting the description of protein-protein interfaces. part ii: Experimental study. Inria technical report, INRIA, February 2005.

[17] M. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16:548–558, 1983.

[18] M. Connolly. Solvent-accessible surfaces of proteins and nucleic acids. *Science*, 221(4612):709–713, 19 August 1983.

[19] M. Connolly. Molecular surfaces: A review. *Network Science*, 14, 1996.

[20] B. Duncan and A. Olson. Approximation and characterization of molecular surfaces. *Biopolymers*, 33(2):219–229, February 1993.

[21] B. Duncan and A. Olson. Shape analysis of molecular surfaces. *Biopolymers*, 33(2):231–238, February 1993.

[22] H. Edelsbrunner and P. KoehlDagger. The weighted-volume derivative of a space-filling diagram. *PNAS*, 100(5):2203–2208, March 2003.

[23] H. Edelsbrunner and E. Mucke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1):43–72, 1994.

[24] Eran Eyal and Dan Halperin. Dynamic maintenance of molecular surfaces under conformational changes. In *SCG '05: Proceedings of the twenty-first annual symposium on Computational geometry*, pages 45–54, New York, NY, USA, 2005. ACM Press.

[25] R. Fraczkiewicz and W. Braun. Exact and efficient analytical calculation of the accessible surface areas and their gradients for macromolecules. *Journal of Computational Chemistry*, 19(3):319–333, 1998.

[26] R. Gabdoulline and R. Wade. Analytically defined surfaces to analyze molecular interaction properties. *J. of Molecular Graphics*, 14(6):341–353, December 1996.

[27] E. Garduno and G. Herman. Implicit surface visualization of reconstructed biological molecules, September 2003.

[28] A. C. Good and W. G. Richards. Rapid evaluation of shape similarity using gaussian functions. *Journal of Chemical Information and Computer Sciences*, 33(1):112–116, February 1993.

[29] J. Grant and B. Pickup. A gaussian description of molecular shape. *Journal of Physical Chemistry*, 99:3503–3510, 1995.

[30] W. Im, D. Beglov, and B. Roux. Continuum solvation model: electrostatic forces from numerical solutions to the poisson-boltzmann equation. *Computer Physics Communications*, 111:59–75, 1998.

[31] Jan J. Koenderink. *Solid shape.* MIT Press, Cambridge, MA, USA, 1990.

[32] B. Lee and F. Richards. The interpretation of protein structures: estimation of static accessibility. *Journal of Molecular Biology*, 55(3):379–400, February 1971.

[33] N. Max. Computer representation of molecular surfaces. *Journal of Medical Systems*, 6(5):485–499, October 1982.

[34] J. Mestres, D. Rohrer, and G. Maggiora. Mimic: A molecular-field matching program. exploiting applicability of molecular similarity approaches. *Journal of Computational Chemistry*, 18(7):934–954, 1997.

[35] J. Mestres, D. Rohrer, and G. Maggiora. A molecular field-based similarity approach to phar. pattern recognition. *J. of Mol. Grap. and Mod.*, 15:114–121, 1997.

[36] Paul G. Mezey. The shape of molecular charge distributions: Group theory without symmetry. *Journal of Computational Chemistry*, 8(4):462–469, June 1987.

[37] Paul G. Mezey. Shape group studies of molecular similarity: Shape groups and shape graphs of molecular contour surfaces. *Journal Journal of Mathematical Chemistry*, 2(4):299–323, October 1988.

[38] Paul G. Mezey. *Shape in Chemistry; An introduction to molecular shape and topology.* VCH Inc, 1993.

[39] Daniel Potts and Gabriele Steidl. Fast summation at nonequispaced knots by nffts. *SIAM Journal on Scientific Computing*, 24(6):2013–2037, 2003.

[40] G. Purvis and C. Culberson. On the graphical display of molecular elec. force-fields and gradients of the electron density. *J. of Mol. Grap.*, 4:88–92, 1986.

[41] N. Ray, X. Cavin, J. Paul, and B. Maigret. Intersurf: dynamic interface between proteins. *J. of Molecular Graphics and Modelling*, 23(4):347–354, January 2005.

[42] F. Richards. Areas, volumes, packing, and protein structure. *Annual Review of Biophysics and Bioengineering*, 6:151–176, June 1977.

[43] D. Ritchie. Evaluation of protein docking predictions using hex 3.1 in capri rounds 1 and 2. *Proteins: Structure, Function, and Genetics*, 52(1):98–106, July 2003.

[44] M. Sanner and A. Olson. Real time surface reconstruction for moving molecular fragments. *Pacific Symposium on Biocomputing*, 2:385–396, 1997.

[45] M. Sanner, A. Olson, and J. Spehner. Fast and robust computation of molecular surfaces. In *Proceedings of the eleventh annual symposium on Computational geometry*, pages 406–407. ACM Press, 1995.

[46] M. Sanner, A. Olson, and J. Spehner. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, 38(3):305–320, March 1996.

[47] John C. Shelley, Mee Y. Shelley, Robert C. Reeder, Sanjoy Bandyopadhyay, and Michael L. Klein. A coarse grain model for phospholipid simulations. *The Journal of Physical Chemistry B*, 105(19):4464–4470, 2001.

[48] Y. Tsai. Rapid and accurate computation of the distance function using grids. *J. Comput. Phys.*, 178(1):175–195, 2002.

[49] A. Varshney and F. Brooks. Fast analytical computation of richards's smooth molecular surface. In *VIS '93: Proc. 4th conf. on Vis. '93*, pages 300–307, 1993.

[50] R. Voorintholt, M. T. Kosters, G. Vegter, G. Vriend, and W. G. Hol. A very fast program for visualizing protein surfaces, channels and cavities. *Journal of Molecular Graphics*, 7(4):243–245, December 1989.

[51] P. Duane Walker, Gustavo A. Arteca, and Paul G. Mezey. A complete shape characterization for molecular charge densities represented by gaussian-type functions. *Journal of Computational Chemistry*, 33(2):231–238, February 1991.

[52] S. Wodak and J. Janin. Analytical approximation to the accessible surface area of proteins. *Proceedings of the National Academy of Sciences of the United States of America*, 77(4):1736–1740, April 1980.

[53] T. You and D. Bashford. An analytical algorithm for the rapid determination of the solvent accessibility of points in a three-dimensional lattice around a solute molecule. *Journal of Computational Chemistry*, 16(6):743–757, 1995.

[54] Y. Zhang, G. Xu, and C. Bajaj. Quality meshing of implicit solvation models of biomolecular structures. *The special issue of Computer Aided Geometric Design (CAGD) on Geometric Modeling in the Life Sciences*, 2006.
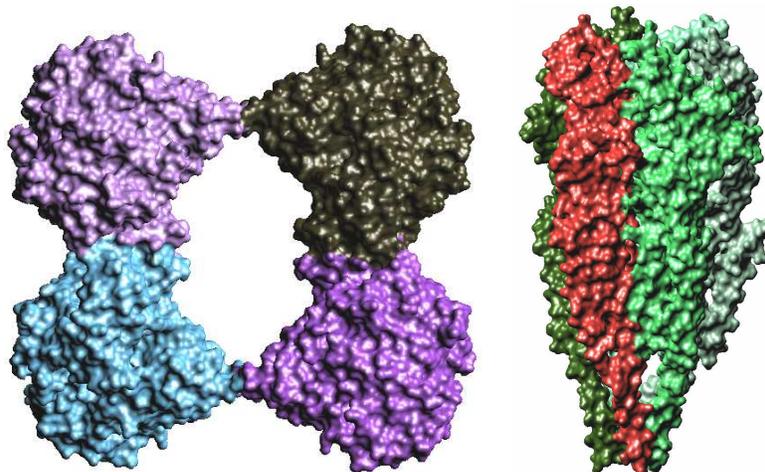
(a) An acetylcholine esterase (1C2B.PDB). It is shown in its tetramer form. Each unit, containing 4172 atoms each, is colored with a different color.

(b) The nicotinic acetylcholine receptor with over 14,000 atoms (2BG9.PDB). It has 5 chains, shown in different colors.
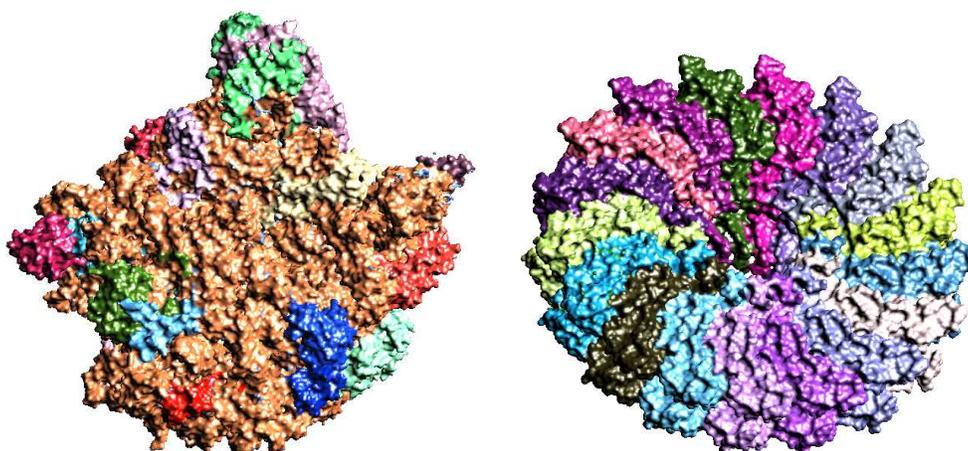
(c) The large ribosomal subunit (1JJ2.PDB) has almost 100,000 atoms. The main RNA chain (in brown) and other chains are shown.

(d) The tobacco mosaic virus, a helical virus (1EI7.PDB). The repeating subunits, each containing 2806 atoms, are shown.

Figure 4: The solvent excluded surfaces of four different molecules.
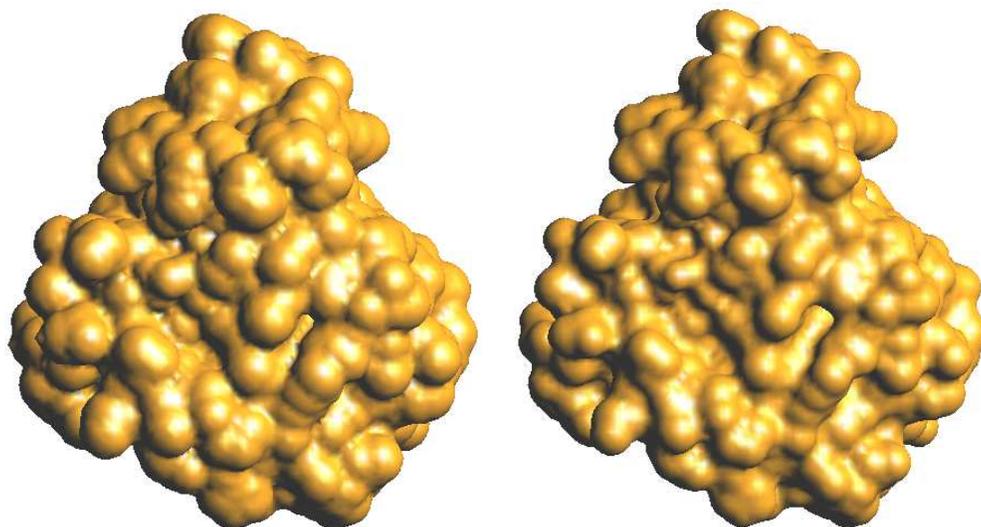
(a) Isovalue=0 yields the SAS

(b) Intermediate surface at isovalue 0.8
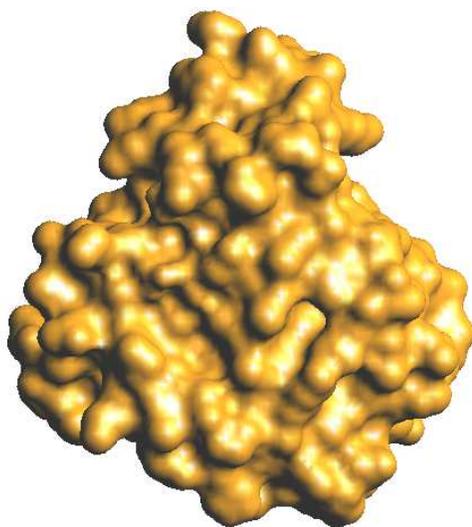
(c) Intermediate surface at isovalue 1.1
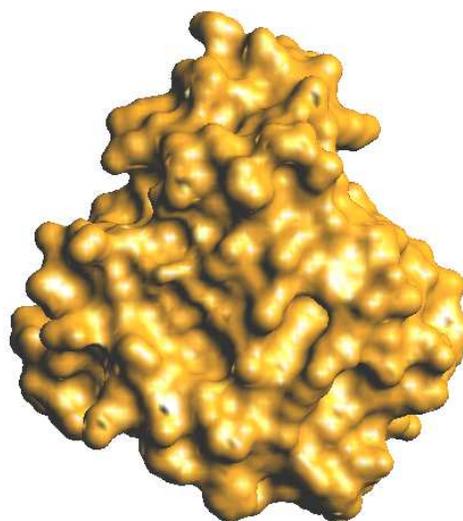
(d) At isovalue 1.4 (probe radius), we obtain the SES.

Figure 5: Our signed distance function based definition yields a family of surfaces which we can extract using a novel adaptive grid based algorithm.

(a) Hemagglutinin   (b) Anti-Idiotypic Fab   (c) Bob-white quail lysosyme
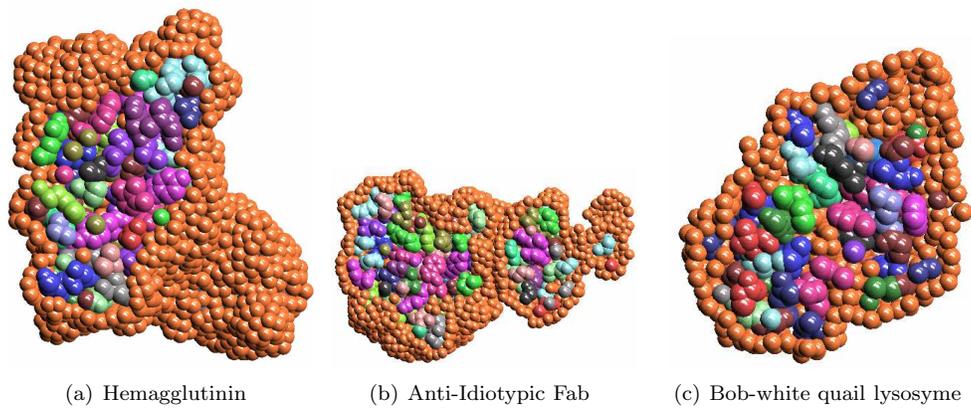
Figure 6: Surface atoms of three complexes shown in orange over the interior atoms which are colored by their residue type.

(a) Immunoglobulin from influenza virus   (b) Idiotypic fab of virus neutralizing antibody   (c) Hylel 5 Fab
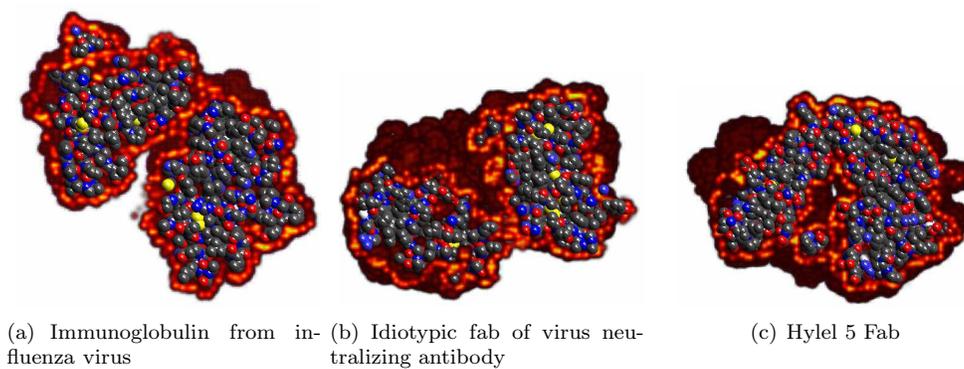
Figure 7: Surface atoms of three complexes shown in orange over the interior atoms which are colored by their residue type.
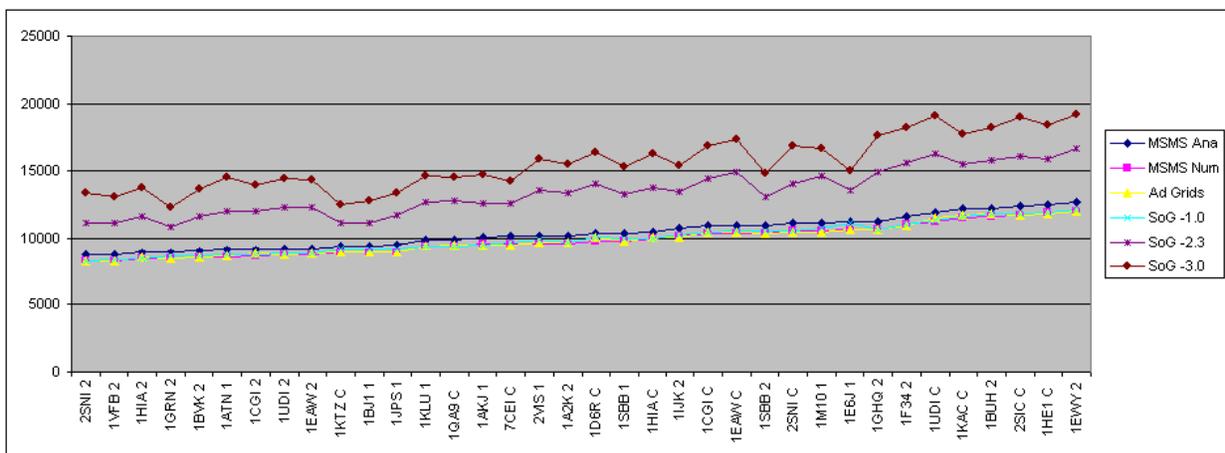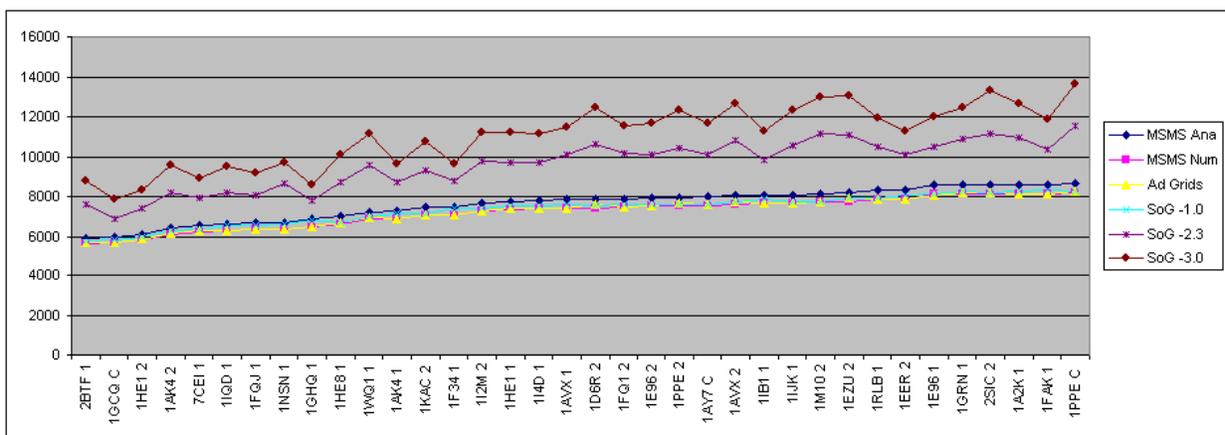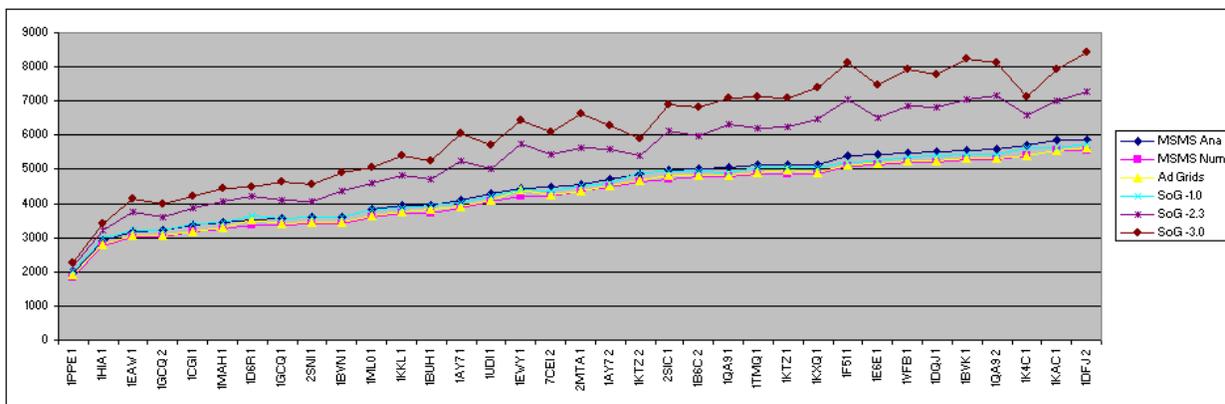
Figure 8: Area comparisons 1,2,3. We compare the areas (in Å^2) of surfaces of molecules computed using our adaptive grid algorithm (yellow), the analytical surface area by MSMS (dark blue), MSMS numerical surface area (pink) and the sum of Gaussians method with three different rates of decay (light blue, purple and brown).
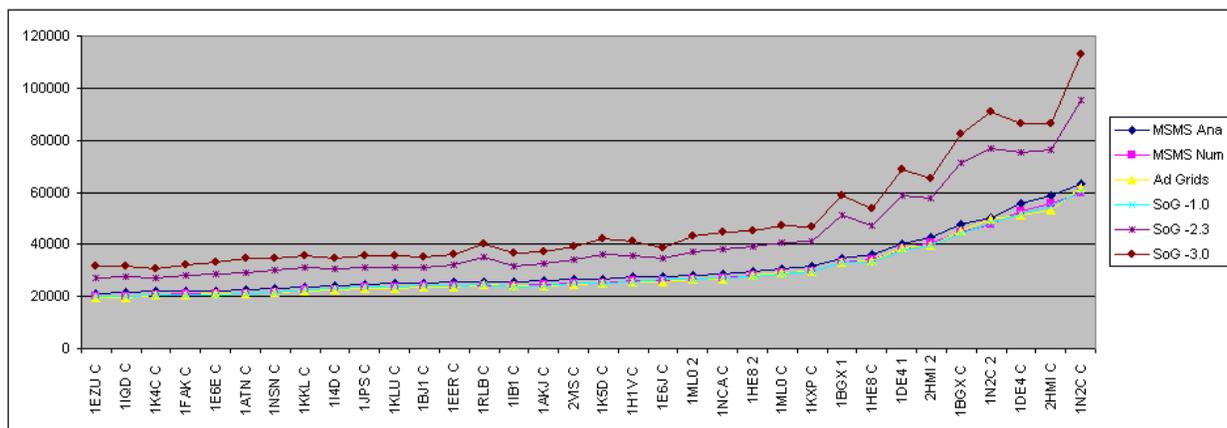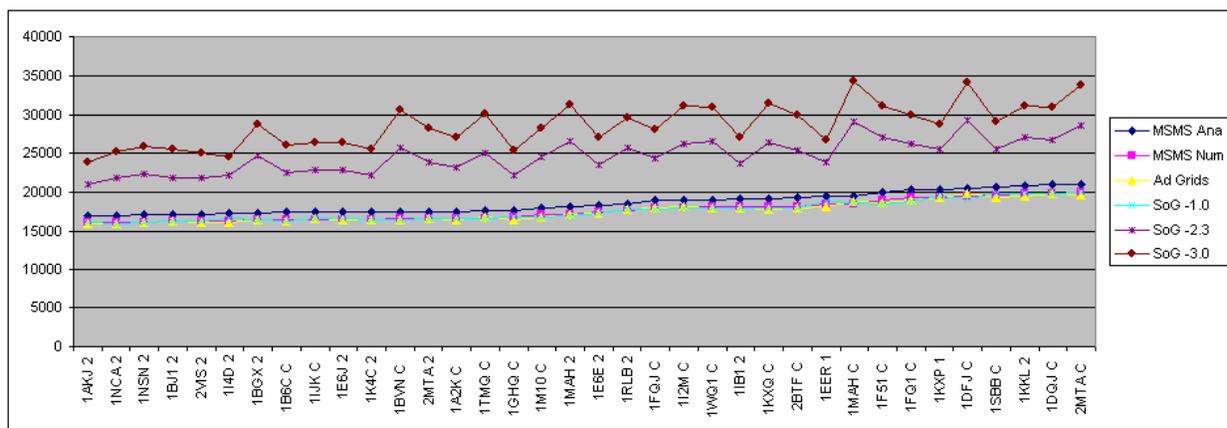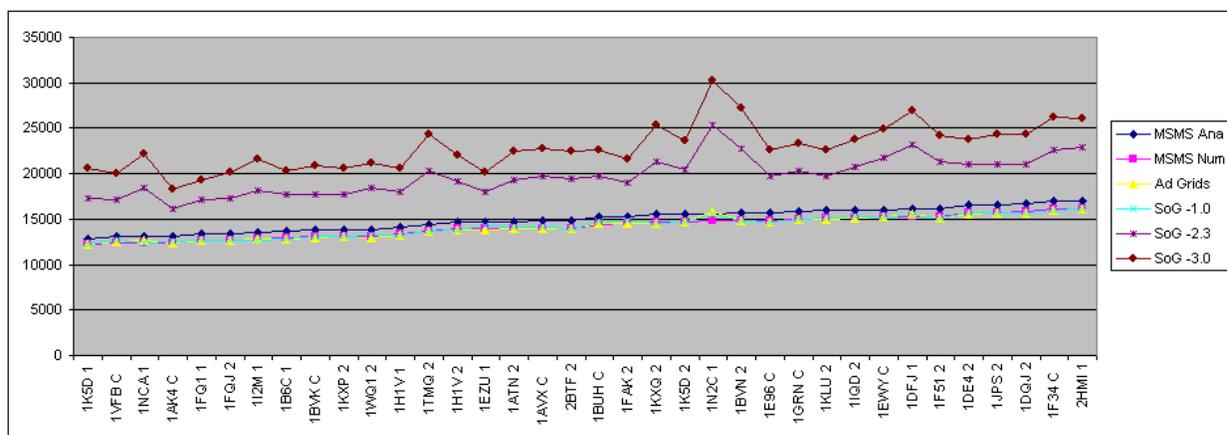
Figure 9: Area comparisons 4,5,6: We compare the areas (in Å^2) of surfaces of molecules computed using our adaptive grid algorithm (yellow), the analytical surface area by MSMS (dark blue), MSMS numerical surface area (pink) and the sum of Gaussians method with three different rates of decay (light blue, purple and brown).

17
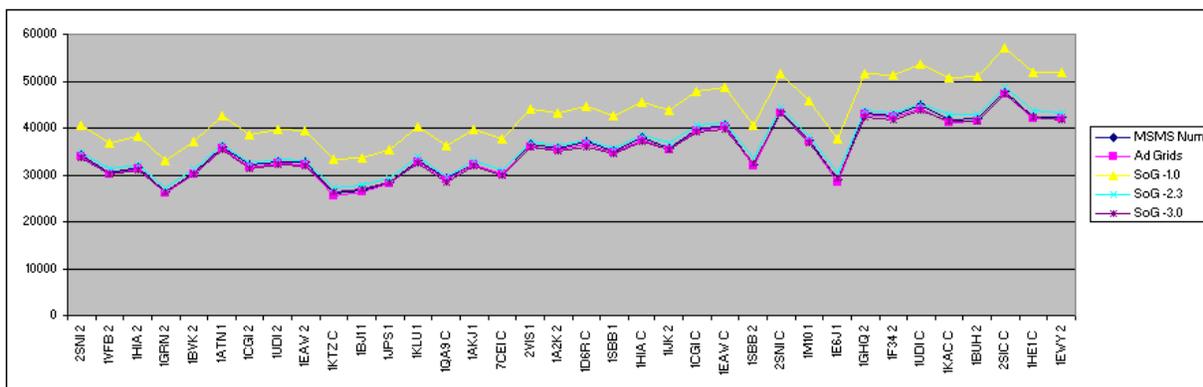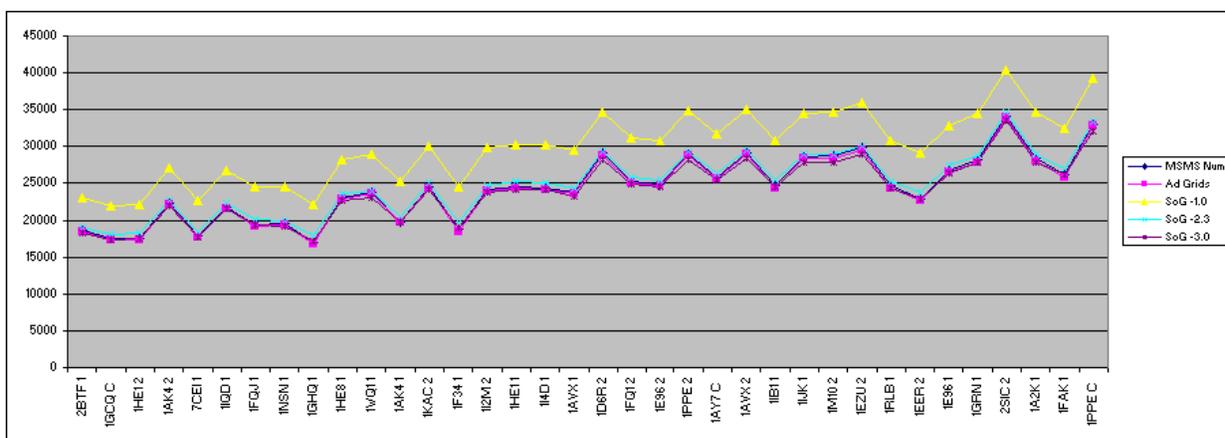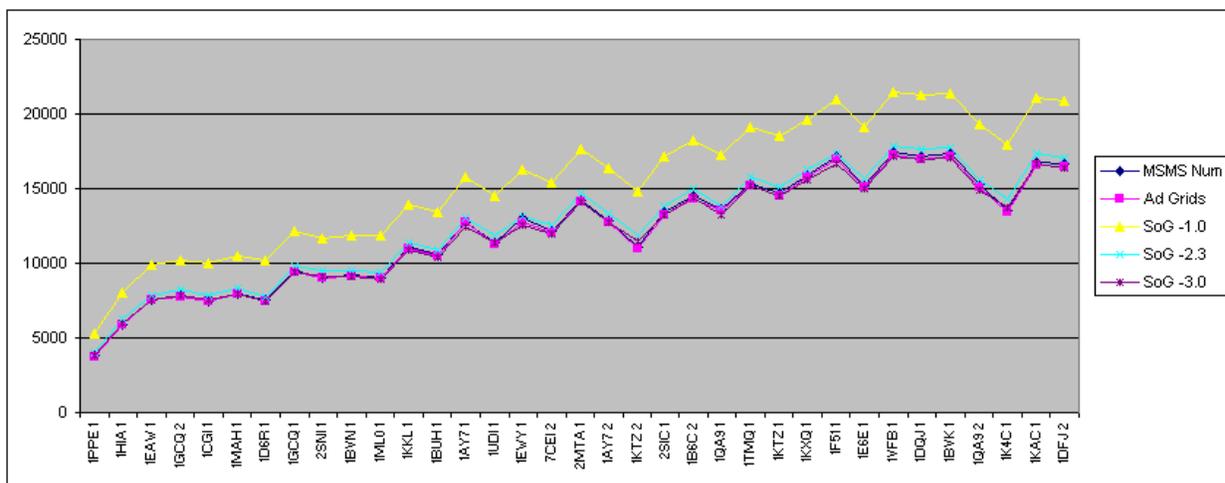
Figure 10: Volume comparisons 1,2,3: We compare the volumes (in Å^3) enclosed by surfaces of molecules computed using our adaptive grid algorithm (pink), MSMS numerical volume (dark blue) and the sum of Gaussians method with three different rates of decay (yellow, light blue and brown).

18
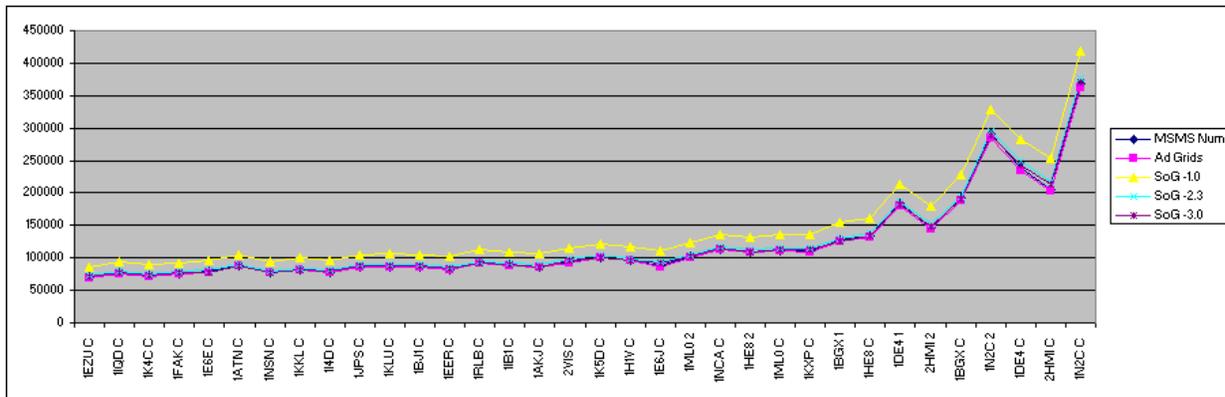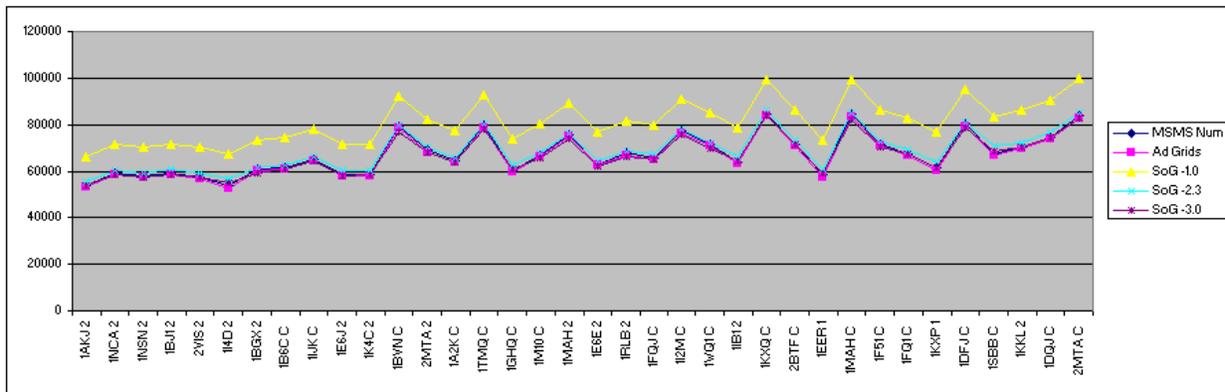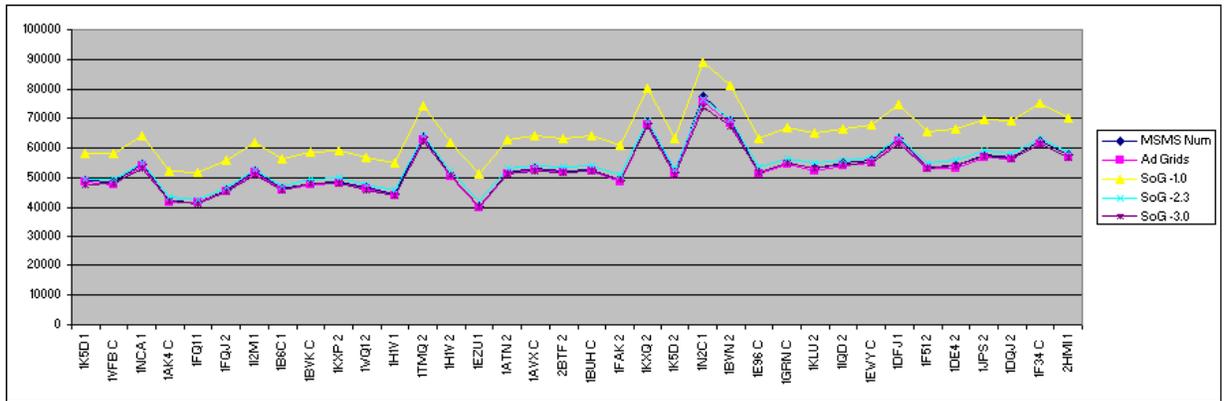
Figure 11: Volume comparisons 4,5,6: We compare the volumes (in Å^3) enclosed by surfaces of molecules computed using our adaptive grid algorithm (pink), MSMS numerical volume (dark blue) and the sum of Gaussians method with three different rates of decay (yellow, light blue and brown).