An Intuitive Approach to Measuring Protein Surface Curvature

Ryan G. Coleman,[1,2] Michael A. Burr,[2] Diane L. Souvaine,[2] and Alan C. Cheng[1,*]
[1]*Research Technology Center, Pfizer Global Research and Development, Cambridge, Massachusetts*
[2]*Department of Computer Science, Tufts University, Medford, Massachusetts*

ABSTRACT **A natural way to measure protein surface curvature is to generate the least squares fitted (LSF) sphere to a surface patch and use the radius as the curvature measure. While the concept is simple, the sphere-fitting problem is not trivial and known means of protein surface curvature measurement use alternative schemes that are arguably less straightforward to interpret. We have developed an approach to solve the LSF sphere problem by turning the sphere-fitting problem into a solvable plane-fitting problem using a transformation known as geometric inversion. The approach works on any arbitrary surface patch, and returns a radius of curvature that has direct physical interpretation. Additionally, it is flexible in its ability to find the curvature of an arbitrary surface patch, and the "resolution" can be adjusted to highlight atomic features or larger features such as peptide binding sites. We include examples of applying the method to visualization of peptide recognition pockets and protein conformational change, as well as a comparison with a commonly used solid-angle curvature method showing that the LSF method produces more pronounced curvature results. Proteins 2005; 61:1068–1074.** © **2005 Wiley-Liss, Inc.**

Key words: curvature; peptide binding sites; molecular surface; computational geometry

INTRODUCTION

Protein surface shape and curvature are key aspects of protein function and recognition. Peptide interactions to macromolecules often occur at structural clefts in proteins[1] and RNA,[2] and available algorithms for finding binding sites essentially locate concave pockets.[3–5] Protein surface curvature may influence the hydrophobic effect that is important in understanding protein folding,[6–8] and large-scale comparisons of protein surface topologies can help in protein function annotation by detecting functional similarities based on surface similarity of active sites.[9]

The major methods for quantitatively measuring protein surface curvature can be classified by their use of solid-angle or differential geometry approaches. Both types of methods use the concepts of molecular surface (MS) and solvent-accessible surface first defined by Richards,[10] and depicted in Figure 1(A). The solvent-accessible surface is defined by the center of a spherical water probe that is exhaustively rolled over the protein surface, while the MS is defined by the protein surface that the water probe touches, with inclusion of the reentrant surface depicted by the top water probe in Figure 1(A). While neither the solvent-accessible surface nor the MS is a truly smooth surface, the MS is smoother and is typically used in calculating curvature.

Solid-angle approaches such as the classic method of Connolly calculate curvature by placing a sphere with its center on the molecular surface and measuring the solid angle,[11] which can be defined by the surface area of the sphere portion lying inside the protein divided by the sphere's total surface area [illustrated in Fig. 1(B)]. GRASP[12] and SURFCV[13] use similar approaches. Solid-angle methods are fast but, as illustrated in Figure 1(B), ignore the topology of the surface inside the sphere since only the points where the placed sphere and protein surface intersect are accounted for in the calculations.

The other major approach to curvature calculation involves methods from differential geometry that normally require a mathematically-defined and continuous surface, from which essentially the greatest and smallest curvatures, known as the principal curvatures, of the surface can be calculated analytically. The principal curvatures are averaged to yield an average curvature, or multiplied to yield a Gaussian curvature.[14] These methods assume a continuous and differentiable surface, which is not found for protein molecular surfaces because of torus cusps and creases resulting from the intersection of molecular surface elements. Duncan and Olson used a Gaussian representation of protein atoms in part to try to overcome this issue.[14] SURFRACE circumvents this issue by taking an average of the curvatures of each continuous section (e.g., torus or sphere) of the non-differentiable surface,[15] as illustrated in Figure 1(C). Zachmann et al. elegantly compute principal curvatures from a numerically fitted

Abbreviations: LSF, least squares fitted; MS, molecular surface.

Ryan Coleman's present address is Genomics and Computational Biology Graduate Group, University of Pennsylvania School of Medicine, Philadelphia, PA 19104.

*Correspondence to: Alan Cheng, Pfizer Research Technology Center, 620 Memorial Dr., Cambridge, MA 02139. E-mail: alan.c.cheng@pfizer.com

Received 14 November 2004; Accepted 29 May 2005

Published online 18 October 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/prot.20680

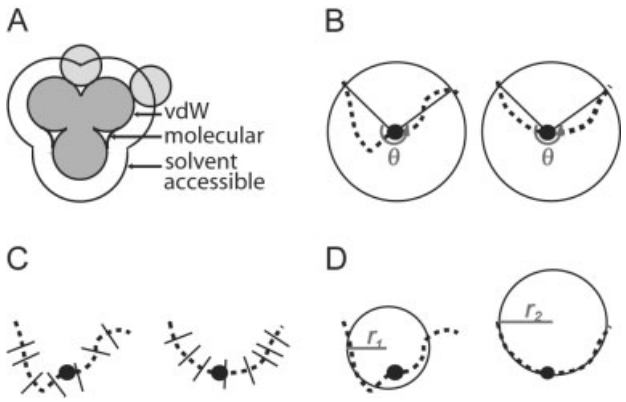

Fig. 1. Comparative illustration of methods discussed. **A:** Definition of solvent accessible surface, molecular surface, and van der Waals surface. The two light gray circles represent water probe spheres, and the dark gray area represents the area occupied by protein atoms. **B:** Illustration of the solid-angle curvature calculation method, which involves placing a sphere on a surface point and using the portion of the sphere lying inside the surface, represented by a dashed line, to generate a value that represents curvature. **C:** Illustration of the SURFRACE method, which takes an average of the curvatures of each continuous section of the non-differentiable surface. **D:** Illustration of the LSF sphere method, which involves fitting a sphere to the surface.

paraboloid.[16] Differential geometry approaches are generally considered to be more accurate than solid-angle methods because they attempt to take into account the surface patch topology, although the most appropriate curvature may differ depending on the application.

Perhaps the most obvious and straightforward approach for measuring surface curvature is to find the least squares-fitted (LSF) sphere to the surface, or, more precisely, the sphere that best fits a given surface as defined by a least sum of squared distances, as illustrated in Figure 1(D). Such an approach has the advantages of differential geometry methods, while additionally providing a quantitative curvature measure that is straightforward to apply and has direct physical interpretation. A sphere can be fitted to any surface, avoiding issues caused by differential geometry requirements for a smooth, differentiable surface. And the curvature can be taken from the radius of the LSF-sphere, instead of averaging curvatures over multiple sections of the non-differentiable surface or deriving a radius of curvature from a fitted parabaloid. Because the sphere is fitted to the *whole* surface, such a curvature measure takes into account the nuances of the surface topology, unlike solid-angle methods. The advantage of fitting to the whole surface in comparison to the solid angle approach is illustrated in Figure 1. The two surfaces result in the same solid-angle curvature in Figure 1(B) despite having topologies that suggest different curvatures, while the LSF sphere approach shown in Figure 1(D) allows differentiation of the two surfaces.

The LSF sphere approach is the most direct approach, but is not known to have been implemented for protein curvature calculation, most likely because fitting a sphere to an arbitrary surface turns out to be a difficult problem. While fitting of planes to a set of points can be easily done by finding the smallest eigenvalue and associated eigenvector of a symmetric, positive, semi-definite matrix,[17] there is no known simple matrix construction for sphere fitting that allows direct evaluation, despite extensive research. Known methods for least squares sphere fitting involve an initial guess followed by iterative searching or sampling methods.[18–22] Such methods suffer from potentially long or unlimited run times and convergence to suboptimal solutions, analogous to finding local minima instead of a global one. Here, we present a method to determine an intuitive curvature measure by transforming the sphere-fitting problem into a plane-fitting problem using a geometric transformation known as inversion. We apply the method for enhancing visualization of proteins and protein conformational changes. While proteins are the focus of this report, this method is general for any type of macromolecule.

MATERIALS AND METHODS

Surface Representation

Analytic representation of the macromolecular surface involves first constructing the three-dimensional weighted Delaunay tessellation of the biomolecule and then subtracting the alpha shape complex,[23] as implemented by Koehl in POCKET[24] and first described by Liang and coworkers.[25] We have validated the surface areas calculated by our surface representation with results from NACCESS[5] and POCKET. Curves representing boundaries between atoms are then calculated to allow reconstruction of the surface. The solvent-accessible surface is a union of sphere sections, while the molecular surface additionally includes torus and sphere sections that map the reentrant surface [Fig. 1(A)]. Each surface piece is evenly painted by points using a spiral dot placement algorithm[26,27] followed by removal of points outside the boundary curves. In this work, we pick points within a local radius to define the patch for curvature measurement, and we restrict the patch to a continuous surface such that patches are not included if they are within the distance range but are not connected.

Other approaches are available for generation of surface points, including the Connolly approach[28] and one based on icosahedrons.[13] The approach we chose gives us flexibility in varying the density of surface points, interconverting between a discretized surface and an analytic surface, and varying the size of patches.

Least Squares Curvature Algorithm

Finding the best-fitted sphere requires simultaneous minimization of distances and definition of a center given the restriction to a sphere. A transformation known as geometric inversion[29,30] is used in generating an LSF sphere to a surface. An inversion sphere of radius k can be defined for any inversion point (p,q,r), and all other points (x_i,y_i,z_i) can be transformed around the inversion sphere as follows:

$$x_i \mapsto \frac{k^2(x_i - p)}{(x_i - p)^2 + (y_i - q)^2 + (z_i - r)^2} + p$$

$$y_i \mapsto \frac{k^2(y_i - q)}{(x_i - p)^2 + (y_i - q)^2 + (z_i - r)^2} + q$$

$$z_i \mapsto \frac{k^2(z_i - r)}{(x_i - p)^2 + (y_i - q)^2 + (z_i - r)^2} + r$$

The transformation takes the inversion point (p, q, r) to infinity. In practice, this point is treated as a special case. We use a unit inversion sphere ($k = 1$) here. However, other definitions of k, and even potentially setting a larger k for larger surfaces, may increase the accuracy of the method. We make use of the geometric inversion property of being self-dual, meaning that a point transformed twice returns to the same point. Another property we take advantage of is that inversion of points that lie on a sphere that passes through the inversion point results in a set of points that lies on a plane [depicted for the two-dimensional case in Fig. 2(A)]. It follows that a set of points that lie approximately on a sphere, where the LSF sphere passes through the inversion point, will lie near a plane under inversion. We note that geometric inversion was also used by Yeates for a different problem in protein modeling, that of finding spheres that simultaneously touch four atoms in a protein and do not intersect the surface.[31] We use inversion to find a LSF sphere, where the fit is determined by the sum of the smallest distances from the ideal sphere to each data point. The best-fit line to points in two dimensions, or the best-fit plane to points in three dimensions, can be determined by finding the smallest eigenvalue and corresponding eigenvector of a symmetric, positive, semi-definite matrix,[18,19] which can be done in constant time. Due to the inversive transformation, the closest point to the origin on the plane found represents the furthest point from the origin when inverted back to normal space. The origin and this furthest point defines the diameter of the sphere since they both lie on the sphere [see left panel of Fig. 2(A)]. Because the inverted space shifts the spatial relationships between points, we use a weight of d_i^4 for the plane LSF, where d_i is the distance for each point, i, from the inversion point.

The algorithm is depicted in Figure 2(B). Making the reasonable assumption that the LSF sphere passes through at least one of the data points, the set of surface points is then fitted around each surface point to generate a set of possible solutions, and the fitted sphere with the least sum of squares is kept as the best fitted sphere solution. The sphere radius is mathematically referred to as the "radius of curvature." The complete algorithm that includes the transformation, plane fitting, and inversion about each point, is detailed below and results in an overall complexity of $O(n^2)$.

Algorithmic complexity for each step is given on the right.

1. Define a set of points P to find the least sum of squares sphere to.
2. For each point $p_i \in P$:
   a. Let p_i be the inversion point (p, q, r) and points $\{x_i, y_i, z_i\}$ be all other points in P. $O(1)$
   b. Invert $\{x_i, y_i, z_i\}$ using the inversion defined in the methods to generate points t_i. $O(n)$
   c. Find the least sum of squares plane fit to the points t_i. $O(1)$
   d. Find the point on the plane closest to p_i. Call this point a. $O(1)$
   e. Transform a using the inversion defined in the methods to generate a'. $O(1)$
   f. Define the sphere center, c_i, as the average of p_i and a'. $O(1)$
   g. Define the radius for the sphere given center c_i. $O(1)$
   h. If the least sum of squares is lower than the previous best fit, keep c_i and the radius. $O(1)$
3. Output the best found center and radius.

To determine whether the surface is convex or concave, we calculate the distance between the center of the LSF sphere and the relevant atom center, and assign the surface as concave if the distance is less than the diameter of the LSF sphere, and convex otherwise. Algorithms were implemented in Java, and use Java3D libraries for vector math and JAMA libraries for solving eigenvalue problems.

Protein phosphatase 1B (PTP-1B) apo (2HNP) and pTyr co-complex (1PTY) structures[32,33] were taken from the protein databank (PDB),[34] and superimposed using MOE-ProSuperpose (version 2003.02, Chemical Computing Group). Practical applications are likely to involve patch sizes defined by radii of 2–4Å. In our Java implementation, PTP-1B structures required approximately one and a half hours using 2Å radius patches, and 5 h using 4Å radius patches on a 2.4-GHz Pentium 4 machine running Linux and using surface dot densities of 25 dots/Å^2. The algorithm scales quadratically with the number of points, and decreasing the dot density to 10 dots/Å^2 allows calculation of curvatures for 2Å radius patches in about 20 min for the PTP-1B structures. The method is straightforward to parallelize, and future work on code optimization can help decrease the run-time. While we find simple Euclidean distances work well to define the patches, a slightly more sophisticated approach may involve measuring the minimum distance between two points along the vertices of the triangulated surface.

Generation of Protein Surface Curvature Graphics

To generate protein surfaces colored by curvature [Fig. 2(C)], PDB file B-factors were first replaced with atomic curvature values, and then visualized by coloring the surface by B-factor. Molecular surfaces [Fig. 2(C, D)] were generated using the Pfizer Molecular Visualization Tool. Curvature values used in the B-factor column are reciprocal radius values, scaled to the range 1–70, with 35 representing planarity. Specifically, the values we plot here are $\left(52 \cdot \frac{1}{r} + 35\right)$, where positive r values represent convex surfaces, and negative r values represent concave surfaces, and r is the radius of curvature. This scaling of

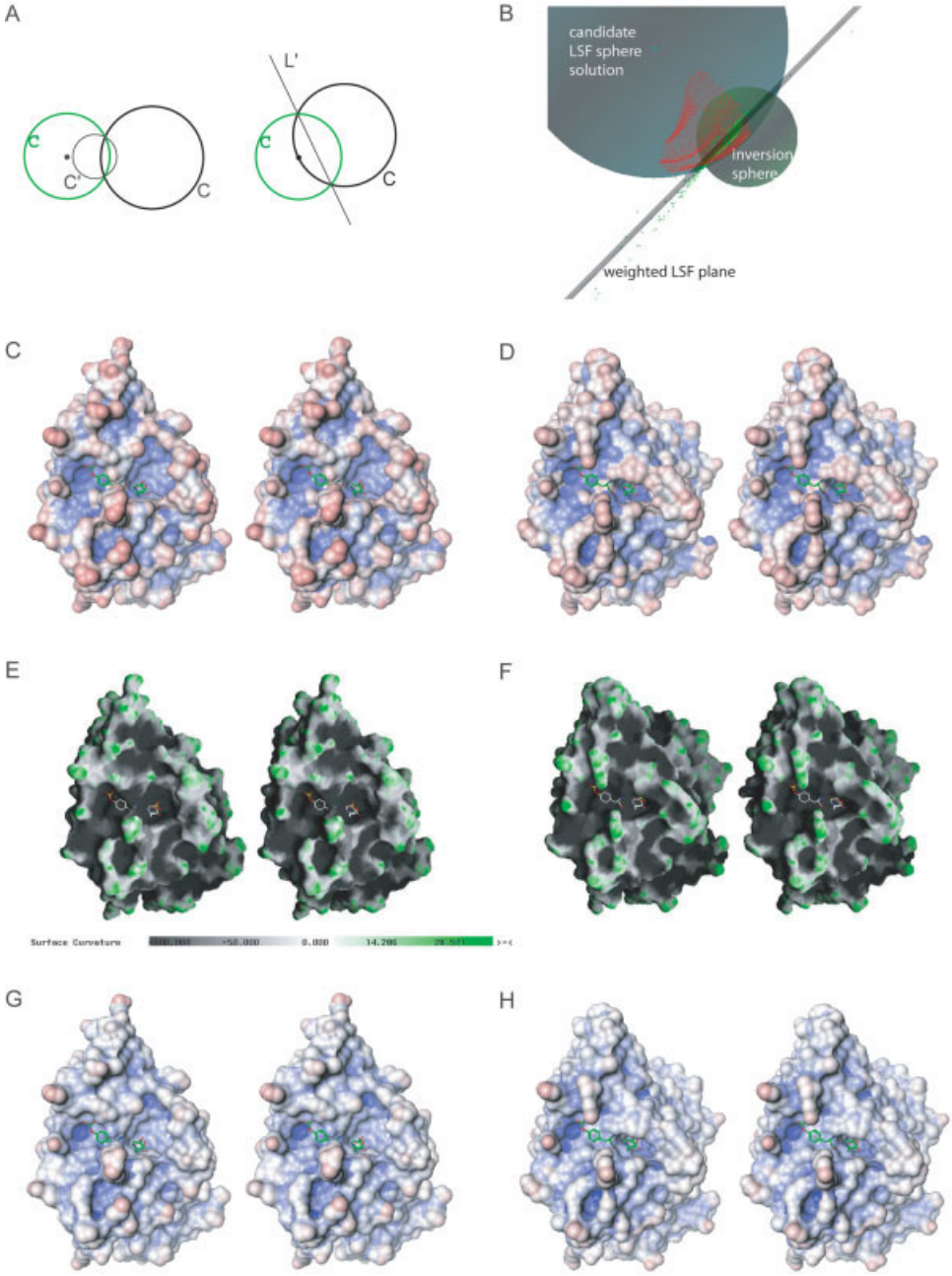

Fig. 2. LSF-sphere curvature. **A:** Geometric inversion of a circle. The image of a circle, C, under inversion in C (green) is a circle, C′, if C does not pass through the center of C, or a line, L′, if C passes through the center of C. **B:** Finding a best least squares fit sphere to a given surface patch (red) involves inversion of points representing the surface patch using an inversion sphere (green) around a candidate point to generate the inverted points (green), finding the best least-squares fit plane (depicted as a black line) to the inverted points, and inversion of the fitted plane to give a candidate sphere (cyan). This process is repeated for every point on the surface patch, and the best least squares sphere is retained. **C,D:** Stereo rendering of PTP-1B peptide binding face colored by curvature for the unbound (PDB ID: 2HNP) and bound (PDB ID: 1PTY) structures, respectively. Curvature is colored from red (highly convex) to white (flat) to blue (highly concave), where atomic curvature is calculated using a 4Å radius. The pTyr's from the bound structure have been mapped onto the apo-structure as a reference. **E,F:** Curvature figures generated in GRASP for the unbound and bound structures, respectively. In GRASP, the curvature is colored from dark gray (score of −100) to white (score of 0) to dark green (score of around 30). **G,H:** To facilitate comparison of the two methods, curvature numbers extracted from GRASP output are plotted onto the structure surfaces and colored using the same scale as in C and D.

values is purely for visualization purposes; the radius of curvature values we calculate is physically interpretable but must be scaled to a suitable B-factor range.

The GRASP solid angle method[12,13] was used to make figures for comparison purposes [Fig. 2(E–H)]. Curvature values calculated at each surface vertex using default settings were transformed into curvature values for each atom by first associating the surface points with the closest atom and then transforming these values into a single value using the average value. We visually validated that the average value provides a reasonably accurate representation of the computed surface curvature, and we also provide the GRASP generated figures side-by-side for comparison. The GRASP curvature values were then transformed to B-factor column values by linearly scaling the GRASP output of −100 to 100 to values of 1 to 70, where a GRASP output of 0 mapped to a value of 35 representing planarity, and graphics were generated in the Pfizer Molecular Visualization Tool.

RESULTS

We use a geometric transformation known as inversion to solve a long-standing problem of finding a LSF sphere to an arbitrary surface. A rapid alpha shape–based method defines the protein surface, from which a dot surface is generated. The problem of finding the LSF sphere to the dot surface is solved by taking the problem into inverted space, where it becomes a simpler plane-fitting problem. The LSF plane is transformed using geometric inversion to real space, yielding the LSF sphere. The sphere radius then can be used to define the curvature (mathematically termed the *radius of curvature*). The LSF-sphere method is illustrated in Figure 2 (A, B) and described in detail in Materials and Methods. We compare the algorithmic details and theoretical running times of the method to the iterative method and a hyperplane-fitting method in a separate work.[35] We focus here on developing and validating the geometric inversion LSF approach for protein structure analysis. We use PTP-1B in our validation because it provides a large range of shallow and deep pocket topologies, but we note that we have also applied the method to over 200 proteins from the PDB without issue.

We first validated that the LSF fit in inverted space has a direct correlation in "real" space. Since points close to the inversion point end up far away in inversive space (see Materials and Methods), a weighting dependent on the distance in "real" space is needed. Our weighting of d^4 is by analogy to Strandlie et al.[36] For 25,000 randomly selected surface patches of 2–6 Å radius covering the protein PTP-1B, we found that the residual sum of squares from the plane fit in inverted space follows that of the sphere fit in "real" space with a correlation coefficient of $r^2 = 0.93$ ($r = 0.97$). While the weighting scheme is not perfect, the correlation is nevertheless quite good, and for smaller patches the correlation is higher.

To validate the sphere fitting, we applied the method to 700 ideal spheres with radii ranging from 0.1 to 7Å in increments of 0.1Å and dot densities of 10 to 100 dots/Å^2 in increments of 10, and found the LSF solution radii were essentially perfect, with the worst error being 2×10^{-12}Å. Randomly selected sets of 10 dots were also sufficient for finding the radii with similarly insignificant errors.

One advantage of our method is that the density of dots is easily adjusted to increase computational speed. In practice, there is a lower limit where the method is not reliable because of poor representation of the surface. To test the convergence of curvatures calculated for different dot densities, LSF spheres were found for the 25,000 random patches using surface dot densities ranging from 10 to 60 dots/Å^2, in increments of 5 dots/Å^2. The patch is defined by a set of points on a continuous surface within a user-defined radius, which in this case was 4Å. For dot densities of 10–60 dots/Å^2, radius of curvature values for 97.1% of surface patches converge to within 1%. For dot densities of 20–60 dots/Å^2, curvature radius values for 98.0% of surface patches converge to within 1%, and 99.8% converge to within 2.5%. The rare (0.2%) occurrence of poor convergence appears to be due to a small subset of saddle surfaces that are particularly sensitive to dot density. Thus, different dot densities can be used depending on the accuracy and robustness desired, and dot densities of 20–25 dots/Å^2 are very robust. We have, thus, found that the least squares curvature method performs robustly and with high accuracy.

DISCUSSION

We have pointed out that the most straightforward approach to measuring protein curvature is to solve for the least-squares fitted sphere to the surface patch of interest. The curvature is easily interpreted because it is simply the radius of the sphere, and the sphere itself can be visualized. We have developed a method that solves the LSF sphere problem using a direct rather than iterative approach, by making the reasonable assumption that the LSF sphere passes through at least one surface point. The method is sensitive to the surface topology unlike solid-angle methods, while it avoids the need for the indirect approaches of differential geometry-utilizing methods. Some available programs using differential geometry approaches for curvature calculation report a small but significant failure rate that is likely in part due to the complexity of the approaches. The algorithm presented here is relatively simple, which leads to its robustness.

Currently the main drawback of our method is the quadratic scaling of the algorithm. We have used high-resolution dot densities of 25 dots/Å^2 that result in 5–6-h run times for typical proteins of ~40 kD. While use of lower resolution dot densities leads to a less accurate surface representation for curvature calculations (see Results), we note that the commonly used 1–5 dots/Å^2 does reduce run times to a couple minutes. The advantage of the method is that the sphere fitting is done in a guaranteed time, unlike iterative search methods. The LSF approach described here is robust, as detailed in the Results, and allows calculation of curvature values for an arbitrary surface.

The ability to vary the surface patch size allows measurement of curvature at different "resolutions." For instance, a 4Å radius patch can highlight macroscopic features such as deep clefts for side-chain recognition, while a 2Å patch allows identification of more nuanced atomic-level features. In Figure 2, PTP-1B is colored according to curvature using 4Å patches, allowing for easy visualization of protein topology. Peptide recognition often involves clefts, and coloring the PTP-1B apo structure in this way reveals two connected clefts, colored blue in Figure 2(C), which turn out to be the two binding pockets for bisphosphorylated substrate peptides.[32, 33] While automated methods are available for finding binding sites, they are typically only useful for finding single deep clefts that are prioritized based on a calculated volume or buriedness.[3–5] Peptides and peptidic ligand binding sites frequently involve less well-defined clefts that automated methods are not designed to find. Visual location of binding sites can be effective in these cases, and can be substantially aided by projection of quantitative curvature values.

Another illustrative application involves visualization of conformational change. While overlay of protein backbones can reveal architectural changes between two protein isoforms, it is more difficult to visualize topology changes. Since protein function is strongly influenced by its shape, a method for visualizing shape changes can be useful in understanding protein function. In Figure 2(D), we show the bound structure of PTP-1B co-complexed with two phosphotyrosines[33] and colored by curvature. Visual comparison with the apo structure shown in Figure 2(C) reveals a striking topological change due to phosphotyrosine binding. This approach to visualizing conformational change is simple and has a clear interpretation.

To illustrate practical differences between the LSF method and a solid angle method, curvatures were computed in GRASP and displayed as shown in Figure 2(E, F). To facilitate visual comparison, in Figure 2(G, H) we also extracted the GRASP curvature values and scaled it in the same fashion as done in Figure 2(C, D) (see Materials and Methods). The deep clefts are as easily identifiable in the GRASP solid-angle method as in the LSF method. Overall, however, the GRASP surface appears to have less pronounced curvature, probably due to the solid-angle method only measuring the "edges," whereas the LSF method is sensitive to the whole surface patch. This less pronounced curvature can be seen in the histogram of curvature values shown in Figure 3. A non-linear scaling of the GRASP curvatures may give greater discrimination of differently curved surfaces, although this would further reduce the physical interpretability of the curvature values. We note, however, that the GRASP method has another interpretation that is not curvature per se, but is instead related to excluded water area.[13]

Other factors such as intermolecular interactions and entropic contributions are also important for understanding function and how conformational changes affect function, but shape is a basic and fundamental property that can be better visualized and understood using the physically intuitive curvature method we have described. While

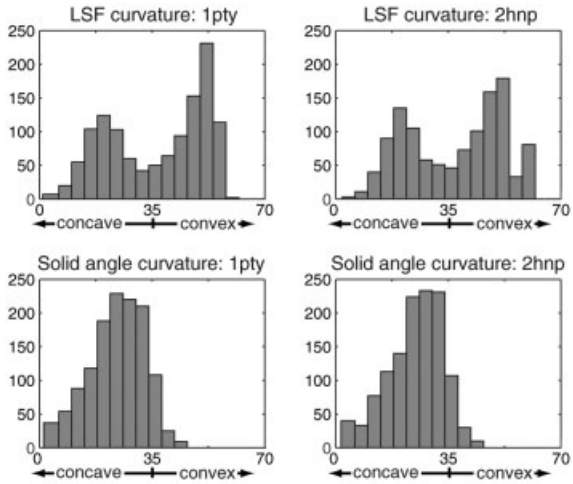

Fig. 3. Histograms of curvature values for PTP-1b apo- (PDB ID: 2HNP) and bound (PDB ID: 1PTY) structures. Solid-angle curvature values are from GRASP. To facilitate comparison, curvature values from both methods are scaled from 1 to 70, where 35 represents planarity, values >35 represent increasing convexity, and values <35 represent increasing concavity (further details provided in Materials and Methods). Plots made with Matlab 7.0 (Mathworks).

we focused here on two illustrative applications that involve projection of the quantitative calculated LSF curvature onto a protein surface, the method is general and can be used in multiple applications. Recent work by Tsuchiya et al. shows that different protein curvature measures can lead to improved prediction of protein-DNA binding interfaces,[37] and in another recent work, Joughin et al.[38] elegantly showed that computed surface curvatures from a differential geometry approach[39] can be used in conjunction with amino acid identities and computed electrostatic potentials to successfully identify phosphopeptide binding sites. It is conceivable that a physically intuitive curvature can enhance both the interpretability and performance of such approaches.

ACKNOWLEDGMENTS

We thank Kathryn Armstrong, Ken Dill, and the anonymous reviewers for helpful comments on the manuscript, Enoch Huang, S. Sridharan, Jason Hughes, Daniel Caffrey, and Qing Cao at Pfizer for discussions, Patrice Koehl for making POCKET available to us, and Kim Sharp for help in using GRASP.

REFERENCES

1. Laskowski RA, Luscombe NM, Swindells MB, Thornton JM. Protein clefts in molecular recognition and function. Prot Sci 1996;5:2438–2452.
2. Cheng AC, Calabro V, Frankel AD. Design of RNA-binding proteins and ligands. Curr Opin Struct Biol 2001;11:478–84.
3. Brady GP Jr, Stouten PF. Fast prediction and visualization of protein binding pockets with PASS. J Comput Aided Mol Des 2000;14:383–401.
4. Liang J, Edelsbrunner H, Woodward C. Anatomy of protein pockets and cavities: Measurement of binding site geometry and implications for ligand design. Prot Sci 1998;7:1884–1897.
5. Laskowski RA. SURFNET: A program for visualizing molecular surfaces, cavities, and intermolecular interactions. J Mol Graph 1995;13:323–330.

6. Sharp KA, Nicholls A, Fine RF, Honig B. Reconciling the magnitude of the microscopic and macroscopic hydrophobic effects. Science 1991;252:106–109.
7. Chan HS, Dill KA. Solvation: Effects of molecular size and shape. J Chem Phys 1994;101:7007–7026.
8. Southall NT, Dill KA. The mechanism of hydrophobic solvation depends on solute radius. J Phys Chem B 2000;104:1326–1331.
9. Ferre F, Ausiello G, Zanzoni A, Helmer-Citterich M. SURFACE: A database of protein surface regions for functional annotation. Nucleic Acids Res 2004;32:D240–244.
10. Richards FM. Areas, volumes, packing, and protein structure. Annual Rev Biophys Bioeng 1977;6:151–76.
11. Connolly ML. Measurement of protein surface shape by solid angles. J Mol Graphics 1986;4:3–6.
12. Nicholls A, Sharp KA, Honig B. Protein folding and association: insights from the interfacial and thermodynamic properties of hydrocarbons. Proteins 1991;11:281–296.
13. Sridharan S, Nicholls A, Honig B. A new vertex algorithm to calculate solvent accessible surface area. Biophys J 1992;61:A174.
14. Duncan BS, Olson AJ. Shape analysis of molecular surfaces. Biopolymers 1993;33:231–238.
15. Tsodikov OV, Record MT Jr, Sergeev YV. Novel computer program for fast exact calculation of accessible and molecular surface areas and average surface curvature. J Comput Chem 2002;23:600–609.
16. Zachmann C-D, Heiden W, Schlenkrich M, Brickmann J. Topological analysis of complex molecular surfaces. J Comp Chem 1992;13:76–84.
17. Pearson K. On lines and planes of closest fit to systems of points in space. The Philosophical Magazine 1901;2:559–572.
18. Shakarji CM. Least squares fitting algorithms of the NIST algorithm testing system. J Res Natl Inst Stand Technol 1998;103:633–641.
19. Eberly D. Least squares fitting of data. http://www.magic-software.com/Documentation/LeastSquaresFitting.pdf; 2001.
20. Ahn SJ, Rauh W, Warnecke H-J. Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. Pattern Recog 2001;34:2283–303.
21. Ahn SJ, Rauh W, Cho HS. Orthogonal distance fitting of implicit curves and surfaces. IEEE Trans Pattern Anal Mach Intell 2002;24:620–638.
22. Atieg A, Watson GA. A class of methods for fitting a curve or surface to data by minimizing the sum of squares of orthogonal distances. J Comput Appl Math 2003;158:277–296.
23. Edelsbrunner H, Mucke EP. Three-dimensional alpha shapes. ACM Trans Graph 1994;13:43–72.
24. Koehl P. ProShape. http://csb.stanford.edu/koehl/ProShape; 2003.
25. Liang J, Edelsbrunner H, Fu P, Sudhakar PV, Subramaniam S. Analytical shape computation of macromolecules: I. Molecular area and volume through alpha shape. Proteins 1998;33:1–17.
26. O'Rourke J. Computational geometry column 18. Int J Comput Geom Appl 1993;3:107–113.
27. Saff EB, Kuijlaars A. Distributing many points on a sphere. Math Intell 1997;19:5–11.
28. Connolly ML. Plotting protein surfaces. J Mol Graphics 1986;4:93-96.
29. Brannan DA, Esplen MF, Gray JJ. Geometry. Cambridge University Press, Cambridge UK; 1999. 510 p.
30. Dobkin DP, Souvaine DL. Advances in robotics 1: algorithmic and geometric aspects of robotics. In: Schwartz T, Yap CK, editors. Computational geometry: a user's guide. Mahwah, NJ: Lawrence Erlbaum Associates; 1997. p 43–93.
31. Yeates TO. Algorithms for evaluating the long-range accessibility of protein surfaces. J Mol Biol 1995;249:804–815.
32. Barford D, Flint AJ, Tonks NK. Crystal structure of human protein tyrosine phosphatase 1B. Science 1994;263:1397–1404.
33. Puius YA, Zhao Y, Sullivan M, Lawrence DS, Almo SC, Zhang ZY. Identification of a second aryl phosphate-binding site in protein-tyrosine phosphatase 1B: a paradigm for inhibitor design. Proc Natl Acad Sci USA 1997;94:13420–13425.
34. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. The Protein Data Bank. Nucleic Acids Res 2000;28:235–242.
35. Coleman RG, Burr MA, Souvaine DL, Cheng AC. Transformations and algorithms for least sum of squares hypersphere fitting. Proc Can Conf Comp Geometry 2004;16:104–107.
36. Strandlie A, Wroldsen J, Fruhwirth R, Lillekjendlie B. Particle tracks fitted on the Riemann sphere. Comp Phys Comm 2000;131:905–908.
37. Tsuchiya Y, Kinoshita K, Nakamura H. Structure-based prediction of DNA-binding sites on proteins using the empirical preference of electrostatic potential and the shape of molecular surfaces. Proteins 2004;55:885–894.
38. Joughin BA, Tidor B, Yaffe MB. A computational method for the analysis and prediction of protein:phosphopeptide-binding sites. Prot Sci 2005;14:131–139.
39. Meyer M, Desbrun M, Schröder P, Barr AH. Discrete differential-geometry operators for triangulated 2-manifolds. In: Visualization and mathematics III. Springer Verlag, Heidelberg, Germany; 2003. p 34–58.