Numerical path integration technique for the calculation of transport properties of proteins

Eun-Hee Kang and Marc L. Mansfield
Department of Chemistry and Chemical Biology, Stevens Institute of Technology, Hoboken, New Jersey 07030, USA

Jack F. Douglas
Polymers Division, National Institute for Standards and Technology, Gaithersburg, Maryland 20899, USA
(Received 28 February 2003; revised manuscript received 9 September 2003; published 31 March 2004)

We present a new technique for the computation of both the translational diffusivity and the intrinsic viscosity of macromolecules, and apply it here to proteins. Traditional techniques employ finite element representations of the surface of the macromolecule, taking the surface to be a union of spheres or of polygons, and have computation times that are $O(m^3)$ where m is the number of finite elements. The new technique, a numerical path integration method, has computation times that are only $O(m)$. We have applied the technique to approximately 1000 different protein structures. The computed translational diffusivities and intrinsic viscosities are, to lowest order, proportional respectively to $N_R^{-1/3}$ and N_R^0, where N_R is the number of amino acid residues in the protein. Our calculations also show some correlation with the shape of the molecule, as represented by the ratio m_2/m_3, where m_2 and m_3 are, respectively, the middle and the smallest of the three principal moments of inertia. Comparisons with a number of experimental results are also performed, with results generally consistent to within experimental error.

DOI: 10.1103/PhysRevE.69.031918 PACS number(s): 87.15.Aa, 87.15.Vv

INTRODUCTION

The transport properties of proteins are important both in understanding biological processes and in molecular characterization. Therefore, methods for estimating or computing such properties from the native structure have been an important area of study. The most common computational approach represents the x-ray crystal or NMR solution structure of the molecule as a union of spheres (hydrodynamic ''beads'') that interact via some form of the Oseen hydrodynamic interaction [1–28]. A related approach consists of constructing a molecular boundary surface and solving the appropriate integral equation with finite elements distributed over the surface [29–31]. Some authors have also employed size or mass correlations [32–36], brute-force molecular dynamics simulations [37], or predictions based on effective spheres or ellipsoids [20,21,35].

A new technique for the computation of the translational diffusion coefficient and of the intrinsic viscosity is now available. The main advantage is that it is generally faster than the older techniques. The older techniques have computation times that are $O(m^3)$, where m is either the number of hydrodynamic beads or the number of finite elements used to represent the surface of the molecule. As we will show below, the new technique is $O(m)$. It takes advantage of a three-fold analogy between different physical problems. Because of analogies between hydrodynamics and electrostatics, the translational diffusivity and the intrinsic viscosity of a molecule are analogous, respectively, to the capacity and to the polarizability tensor that would be possessed by a perfect conductor of precisely the same shape as the molecule [38–43]. A particular angular averaging of the hydrodynamic forces converts the problem to a boundary value problem in electrostatics. (The Oseen tensor, averaged over orientations, becomes the Green's function of a point charge.) The analogy is only approximate, but has been demonstrated to be accurate to about 2% and 5%, respectively, for the diffusivity and the intrinsic viscosity of a large number of objects [38,40]. These electrostatics problems can, in turn, be solved by numerical path integration techniques that involve summing over random walk trajectories in the space outside the object [38,39,41,44–47]. As a result, we can, in a single simulation of random walk trajectories, obtain estimates of the translational diffusivity, the intrinsic viscosity, the electrostatic capacity, and the electrical polarizability tensor, although these last two properties are not those of the molecule itself, but those that would be possessed by a perfect conductor of the same shape as the molecule [46].

A powerful algorithm for performing the numerical path integrations is presented in the following section. The justification for the algorithm is already given in the literature, and so will not be repeated here [38,39,42–44,46,47]. In Sec. III we report its application to the calculation of the translational diffusivity and the intrinsic viscosity of over 1000 protein structures downloaded from the Protein Data Bank [48]. We find that the translational diffusivity is correlated with the size of the protein, varying approximately as the $-1/3$ power of the residue number. Weaker correlations of both the diffusivity and the intrinsic viscosity with the shape of the protein as manifested through ratios of the principal moments of inertia are also found: Elongated structures tend to have smaller diffusivities and larger intrinsic viscosities than spherical ones. Experimental diffusivities and intrinsic viscosities are available for a number of proteins; for these we find generally good agreement with our estimates.

ZENO ALGORITHM

The necessary calculation can be formulated as a boundary value problem on the surface of the molecule. We let $\mathbf{\Omega}$ represent the surface. The algorithm employs a sphere, radius R, called the launch sphere, which encloses $\mathbf{\Omega}$. The surface

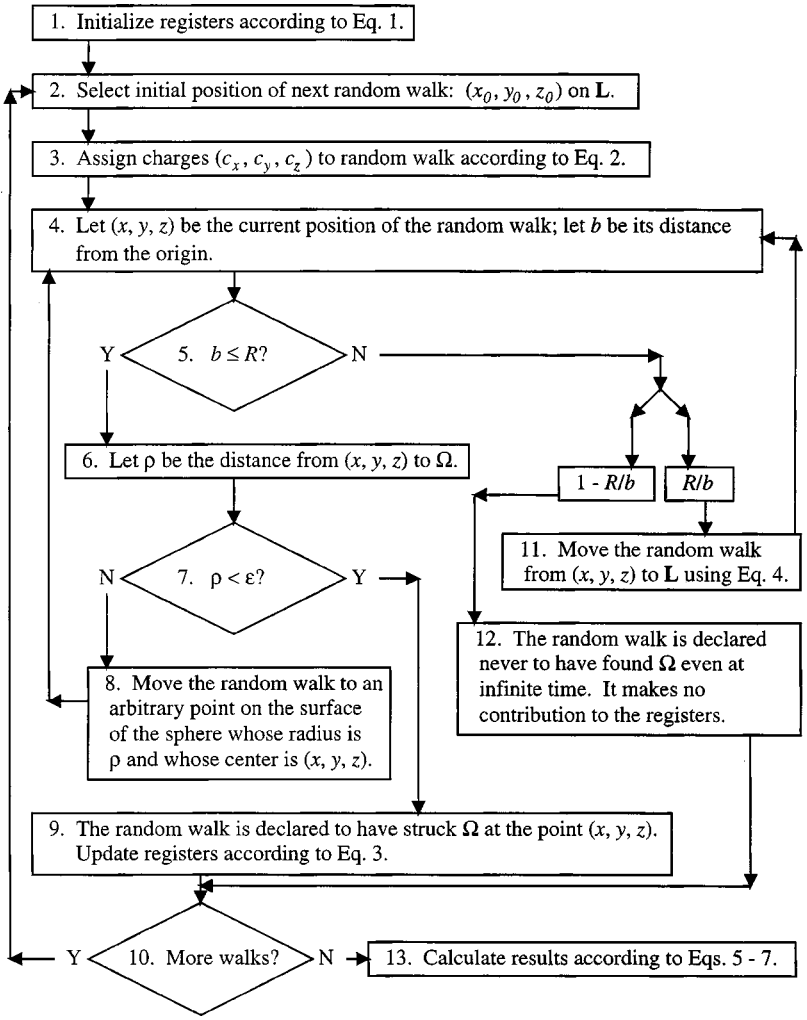

FIG. 1. Flow chart of the Zeno algorithm used to calculate translational diffusion coefficients and intrinsic viscosities. Note the "fork in the road," encountered on the NO path leaving box 5. One path or the other is chosen at random, one with probability R/b, the other with probability $1-R/b$.

of the launch sphere is designated **L**. Any arbitrary sphere that completely encloses **Ω** can serve as the launch sphere, and there is no need for **L** and **Ω** to be concentric. Nevertheless, the best statistics are obtained if we use the smallest possible launch sphere. Cartesian coordinates are defined relative to the center of **L**, unless otherwise noted. We refer to the technique as the Zeno algorithm, and summarize the algorithm in the following paragraphs and in a flow chart in Fig. 1.

A large number, N, of random walkers are initiated from arbitrary points on the launch sphere. These begin walking and a certain fraction of them eventually adsorb onto **Ω**. The remainder never find **Ω**, even at infinite time, and are said to have wandered off to infinity. The statistics of the trajectories of these random walks give us the properties of interest. The following registers are used to accumulate statistics, and are initialized to zero at the outset:

$$K_x^+ = K_y^+ = K_z^+ = K_x^- = K_y^- = K_z^- = 0, \text{ initially,} \tag{1a}$$

$$V_{xx}^+ = V_{xy}^+ = V_{xz}^+ = V_{yx}^+ = \cdots = V_{xx}^- = V_{xy}^- = V_{xz}^- = V_{yx}^- = \cdots = 0, \quad \text{initially.} \tag{1b}$$

We let $(x_0, y_0, z_0) \in \mathbf{L}$ represent the initial point of the trajectory of one of the random walkers. Three "charges," c_x, c_y, c_z, each equal to ± 1, and termed the "x charge," the "y charge," and the "z charge," respectively, are assigned to each random walker [46]. The three charges are stochastic variables depending on (x_0, y_0, z_0). For c_x we have

$$c_x = +1 \quad \text{with probability } \frac{1}{2} + \frac{x_0}{2R},$$

$$c_x = -1 \quad \text{otherwise,} \tag{2}$$

with analogous definitions for c_y and c_z.

The algorithm requires calculation of the distance function ρ, representing the minimum distance from a point (x,y,z) outside **Ω** to **Ω**. If the random walker is currently found on or inside **L** and outside **Ω**, it is displaced to an

arbitrary point a distance ρ away from its current position, guaranteeing that it will still lie outside $\mathbf{\Omega}$, and then the process is repeated. Since the displacement is never large enough to bring the random walker in contact with the surface, it never actually hits $\mathbf{\Omega}$. However, it comes arbitrarily close as we find ourselves displacing the walker through progressively smaller and smaller ρ distances. This explains the name of the algorithm: The process suggests Zeno's paradox in which Achilles runs to overtake a tortoise, but can never quite catch up because we continually examine smaller and smaller time steps. Of course, Achilles and our computer both have better things to do than get caught in this infinite spiral, and so we declare that the random walker has made contact with $\mathbf{\Omega}$ whenever ρ falls below some preset ''skin thickness,'' ε.

Whenever the random walker hits $\mathbf{\Omega}$ at a point (x,y,z) we accumulate statistics on the event by updating the registers according to the following prescription:

$$\text{If } c_x=+1, \text{ then } K_x^+ \leftarrow K_x^+ + 1$$

$$\text{and } (V_{xx}^+, V_{yx}^+, V_{zx}^+) \leftarrow (V_{xx}^+, V_{yx}^+, V_{zx}^+) + (x,y,z). \quad (3a)$$

$$\text{If } c_x=-1, \text{ then } K_x^- \leftarrow K_x^- + 1$$

$$\text{and } (V_{xx}^-, V_{yx}^-, V_{zx}^-) \leftarrow (V_{xx}^-, V_{yx}^-, V_{zx}^-) + (x,y,z). \quad (3b)$$

$$\text{If } c_y=+1, \text{ then } K_y^+ \leftarrow K_y^+ + 1$$

$$\text{and } (V_{xy}^+, V_{yy}^+, V_{zy}^+) \leftarrow (V_{xy}^+, V_{yy}^+, V_{zy}^+) + (x,y,z). \quad (3c)$$

$$\text{If } c_y=-1, \text{ then } K_y^- \leftarrow K_y^- + 1$$

$$\text{and } (V_{xy}^-, V_{yy}^-, V_{zy}^-) \leftarrow (V_{xy}^-, V_{yy}^-, V_{zy}^-) + (x,y,z). \quad (3d)$$

$$\text{If } c_z=+1, \text{ then } K_z^+ \leftarrow K_z^+ + 1$$

$$\text{and } (V_{xz}^+, V_{yz}^+, V_{zz}^+) \leftarrow (V_{xz}^+, V_{yz}^+, V_{zz}^+) + (x,y,z). \quad (3e)$$

$$\text{If } c_z=-1, \text{ then } K_z^- \leftarrow K_z^- + 1$$

$$\text{and } (V_{xz}^-, V_{yz}^-, V_{zz}^-) \leftarrow (V_{xz}^-, V_{yz}^-, V_{zz}^-) + (x,y,z). \quad (3f)$$

In other words, K_x^+ counts the number of random walkers with positive x charge that hit the surface, while $(V_{xx}^+, V_{yx}^+, V_{zx}^+)$ is the vector sum of the points of contact of all random walkers with positive x charge, and similarly for all the other registers.

Whenever the random walker is found outside $\mathbf{L}$, there exists a finite probability that it may never return to $\mathbf{L}$, let alone hit $\mathbf{\Omega}$. This explains the ''fork in the road'' on the NO branch out of box 5 in Fig. 1. That construction means that control follows one path with probability R/b, and the other with probability $1-R/b$, where b is the distance between the current position of the random walker and the center of the launch sphere. Therefore, with probability $1-R/b$ the random walker is assumed to have escaped to infinity, while with probability R/b the random walker is returned to the surface of $\mathbf{L}$, to a site determined as follows: We perform a coordinate rotation $\mathbf{R}$ that temporarily places the random walker on the positive z axis; we select two floating point random numbers, r_1 and r_2, each distributed randomly on the interval (0, 1); new spherical-polar coordinates (r,θ,ϕ) for the point are assigned using Eq. (4); and finally we back-transform with the rotation $\mathbf{R}^{-1}$ [41]. The net result is to place the walker at an appropriate site on the surface of $\mathbf{L}$:

$$r=R, \quad (4a)$$

$$\theta=\cos^{-1}\left\{(2X)^{-1}\left[1+X^2-\left(\frac{(1-X)(1+X)}{1+(2r_1-1)X}\right)^2\right]\right\}, \quad X=R/b, \quad (4b)$$

$$\phi=2\pi r_2. \quad (4c)$$

Equation (4b) is derived from the familiar formula for the charge distribution on a conducting sphere induced by an external point charge, which also represents the distribution of the sites of first passage of random walkers from an external point to the surface of a sphere.

The computation continues until a total of N random walk trajectories have been generated. Once this occurs, control shifts to box 13, and results are computed according to the following equations. Let

$$t=\frac{K_j^+ + K_j^-}{N} \text{ (independent of } j), \quad (5a)$$

$$u_j=\frac{K_j^+ - K_j^-}{N}, \quad (5b)$$

$$v_{ij}=\frac{V_{ij}^+ + V_{ij}^-}{N}, \quad (5c)$$

$$w_{ij}=\frac{V_{ij}^+ - V_{ij}^-}{N}. \quad (5d)$$

The electrostatic capacity and all nine components of the electrostatic polarizability tensor are given by

$$C=tR \quad \text{(capacity)}, \quad (6a)$$

$$\alpha_{ij}=12\pi R^2(w_{ij}-u_j v_{ij}/t) \quad \text{(polarizability tensor)}, \quad (6b)$$

$$\langle\alpha\rangle=\left(\frac{1}{3}\right)(\alpha_{xx}+\alpha_{yy}+\alpha_{zz}) \quad \text{(mean polarizability)}. \quad (6c)$$

In a large number of calculations on diverse bodies (either exact or by finite elements), the ratios R_h/C and $M[\eta]/\langle\alpha\rangle$ prove respectively to be within about 2% of 1 and 5% of 0.79, where R_h is the hydrodynamic radius, $[\eta]$ is the intrinsic viscosity, and M is the mass of the body [38,40]. Therefore, we can apply the following approximations to determine hydrodynamic properties:

$$R_h \cong C \quad \text{(hydrodynamic radius)}, \quad (7a)$$

$$D \cong kT/(6\pi\eta C) \qquad \text{(translational diffusion coefficient)}, \tag{7b}$$

$$[\eta] \cong \frac{0.79\langle\alpha\rangle}{M} \qquad \text{(intrinsic viscosity)}, \tag{7c}$$

where η is the solvent viscosity. Finally, if we define R_η to be the radius of the sphere that has the same intrinsic viscosity as the macromolecule, then we obtain

$$R_\eta \cong 0.42\langle a\rangle^{1/3}. \tag{7d}$$

The calculation of the electrostatic properties is rigorous in the limits $N\to\infty$ and $\varepsilon\to 0$, while finite N and nonzero ε are expected to generate relative errors of magnitude $N^{-1/2}$ and ε/C, respectively. We have found that values of N around 10^6 guarantee accuracies of about 4 significant figures in the capacity and 3 figures in the polarizability, and that values of $10^{-7}<\varepsilon/R<10^{-5}$ still permit the algorithm to perform in reasonable times [46].

The only aspect of the algorithm that depends explicitly on $\mathbf{\Omega}$ is the calculation of the distance function ρ in box 6 of Fig. 1. Therefore, the algorithm is very versatile: We use two different plug-in procedures, one for initialization (e.g., setting up an array of beads or finite elements that represent $\mathbf{\Omega}$), and the other for calculation of ρ. All $\mathbf{\Omega}$-specific features are assigned to these two procedures, and we can switch easily from one $\mathbf{\Omega}$ to another by plugging in the appropriate procedures. The algorithm is also fast: Traditional approaches to these boundary value problems are either to divide the surface up into finite elements consisting of small polygonal regions or else to represent the object as a union of spheres. Solution then typically requires the inversion of an order m matrix where m is the number of finite elements or of spheres, so computation time is $O(m^3)$. The bottleneck of the Zeno algorithm for complex shapes is computation of the distance function ρ. But treating the surface as a collection of m finite elements or the body as the union of m spheres leads to computation times that are $O(m)$, since we obtain the minimum distance to the surface by computing the minimum distance to each one of the elements and taking the smallest of all these. Another drawback of the traditional approach occurs when the molecule is represented as a union of spheres interacting via the Oseen hydrodynamic interaction. The traditional approach is problematical if the spheres overlap [9], but not the approach described here.

COMPUTATIONS

X-ray crystal or NMR solution structures from the Protein Data Bank (PDB) were used in our calculations [48]. The initial download consisted of all files in the February 2000 PDB-Select subset, of which there are about 1200 [49,50]. A single structure was extracted from each file. In those instances for which a single file contains more than one structure, e.g., a PDB submission containing multiple NMR models, we selected the last model of the record. The protein is represented as a union of spheres, each of radius 5 Å, and centered on the C_α atom of each amino acid residue, and we let N_R represent the number of residues. Figure 2 displays graphical images of several of these representations, and includes a catalase, PDB code 1 cf9, $N_R\approx 2900$ [Fig. 2(a)]; a myoglobin, PDB code 1a6m, $N_R\approx 150$ [Fig. 2(b)]; a histone, PDB code 1aoi, $N_R\approx 800$ [Fig. 2(c)]; and a β2-glycoprotein, PDB code 1qub, $N_R\approx 320$ [Fig. 2(d)]. Since successive C_α's are never more than 10 Å apart, the spheres for successive residues obviously overlap, and there is also substantial overlap between the spheres representing neighboring nonbonded residues. However, it has been suggested that the 5 Å radius is appropriate to model the hydration shell around the protein [19]. The problem of computing ρ for such a model reduces to computation of the distance to the surface of each sphere, and taking the minimum over all spheres.

Protein conformations consisting of more than one chain, i.e., proteins with quaternary structure, are treated as a unit; this is equivalent to assuming that the chains are associated in that conformation while in solution. However, in a small number of cases (ca. 30) the union of the C_α spheres forms a disconnected set. For example, the PDB includes nucleic acid-protein complexes with a single double helix bound to several proteins. Our procedure selects only the C_α coordinates, and once the nucleic acids are discarded, the remaining protein molecules occasionally are not in contact. We could obviously apply this technique to such complexes by including the nucleic acid, but in this study we chose against this. Therefore any PDB structure that yields a disconnected set of spheres is discarded from the sample set.

The algorithm was applied to all the remaining structures, using a skin thickness, ε, of 0.001 Å and launching a total of $N=10^6$ random walks at each protein structure. The computation time is approximately linear in both N and N_R. This study was carried out on several Pentium III machines with clock speeds of 800 Mhz, and the CPU time necessary for each calculation is approximately $(2.3\times 10^{-8}\ \text{min})NN_R$. (For example, a million random walks launched at a protein of 100 residues requires about 2 min of CPU time.)

As shown in Eq. (7c), accurate computations of $[\eta]$ require accurate molecular weights, which can introduce discrepancies: For example, we can expect that the apo and halo forms of most proteins have very nearly the same values of $\langle\alpha\rangle$, since this depends only on the space filled by the molecule. However, $[\eta]$ could vary somewhat depending on the mass of the guest moieties present in the halo form. Rather than concern ourselves with this issue, we have chosen to consider the product $M[\eta]\equiv V_h$, the "hydrodynamic volume," which is, as shown in Eq. (7c), directly proportional to $\langle\alpha\rangle$, and can be computed without knowledge of the molecular mass.

We find that the capacity and the mean polarizability correlate strongly with the cube root and the first power of N_R, respectively; see Fig. 3. These correlations are expected for globular proteins since capacity and polarizability are comparable, respectively, to the size and volume. Equations (7a)–(7d) then imply that the translational diffusivity and the intrinsic viscosity vary approximately as $N_R^{-1/3}$ and N_R^0, respectively.

Of particular interest is the way that each individual pro-

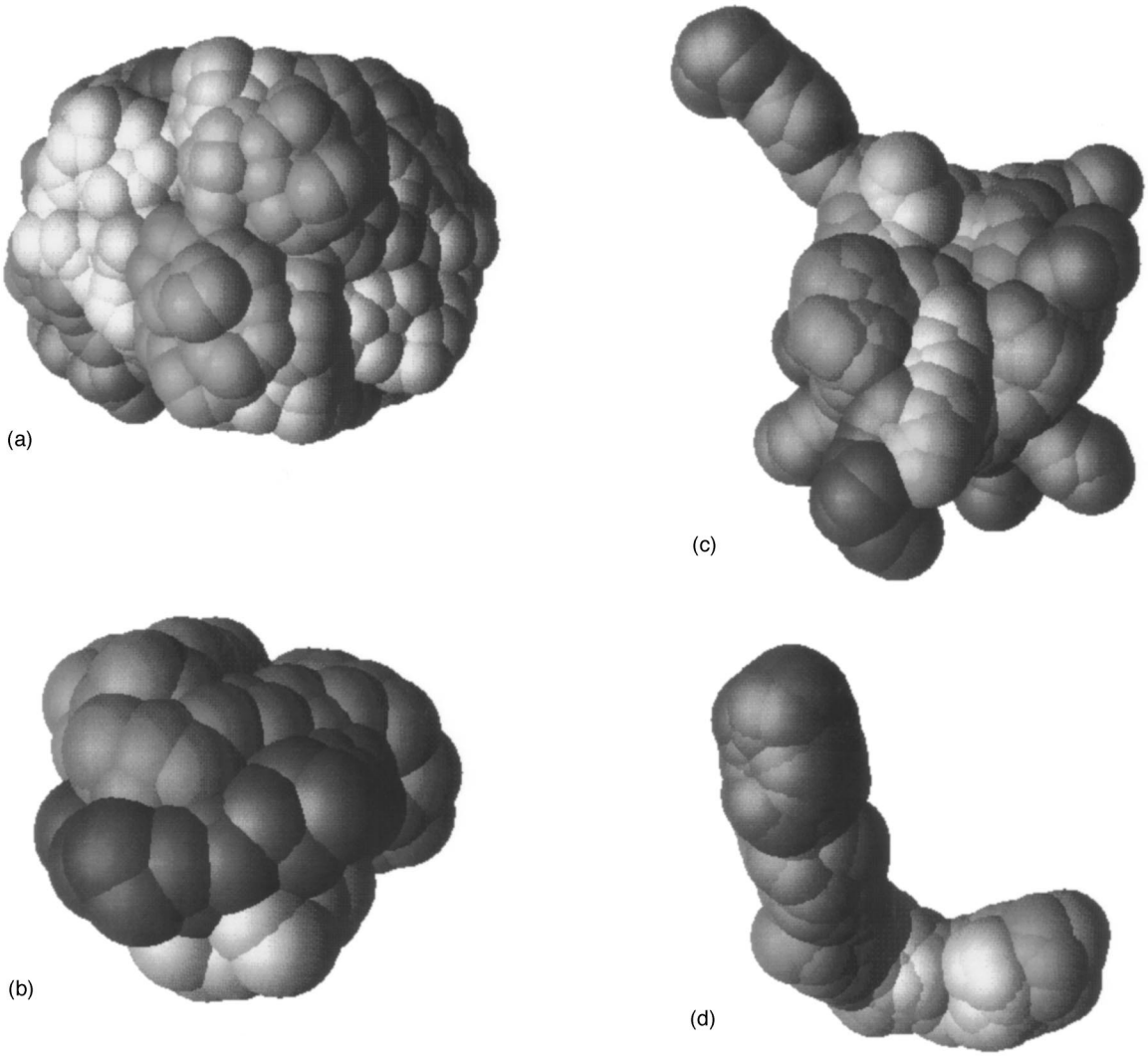

FIG. 2. Each protein is modeled as a union of spheres of radius 5 Å, one sphere per amino acid residue, centered on the C_α atom of the residue. (a) A catalase, PDB code 1cf9; (b) a myoglobin, PDB code 1a6m; (c) histone, PDB code 1aoi; and (d) a β2-glycoprotein, PDB code 1qub.

tein fluctuates about this average N_R behavior. We study this by considering the quantities

$$G = C/N_R^{1/3} \tag{8}$$

and

$$H = \langle \alpha \rangle / N_R . \tag{9}$$

For most proteins in the sample set, G lies between 3.7 and 5.0 Å, while H lies between about 650 Å^3 and 1500 Å^3. The averages of G and H over all molecules in the sample set are $\langle G \rangle = 4.10$ Å and $\langle H \rangle = 910$ Å^3, respectively, which leads to the following low-order approximations:

$$R_h \approx (4.10 \text{ Å}) N_R^{1/3}, \tag{10}$$

$$M[\eta] \approx (719 \text{ Å}^3) N_R . \tag{11}$$

The values of G and H relative to their means carry all further information about the shape of the protein. For example, Table I displays the G and H values of the four proteins shown in Fig. 2, and underscores the progression in G or H as we proceed from globular to elongated proteins. We characterize the shape of the protein by calculating the principle moments of inertia, m_1, m_2, and m_3, with $m_1 > m_2 > m_3$. For spheroidal shapes we have $m_1 \approx m_2 \approx m_3$, while for prolate shapes we have $m_1 > m_2 \approx m_3$, and for oblate $m_1 \approx m_2 > m_3$. Highly prolate shapes are rare among the native proteins. Consequently, m_1/m_2 is reasonably close to 1 for all members of the sample set. However, correlations between either G or H and m_2/m_3 are observed. Figure 4 shows how the distribution of G varies with m_2/m_3. The following formula adequately represents the drift in average G as a function of m_2/m_3:

$$G \approx [3.84 + 0.116(m_2/m_3)] \text{ Å}. \tag{12}$$

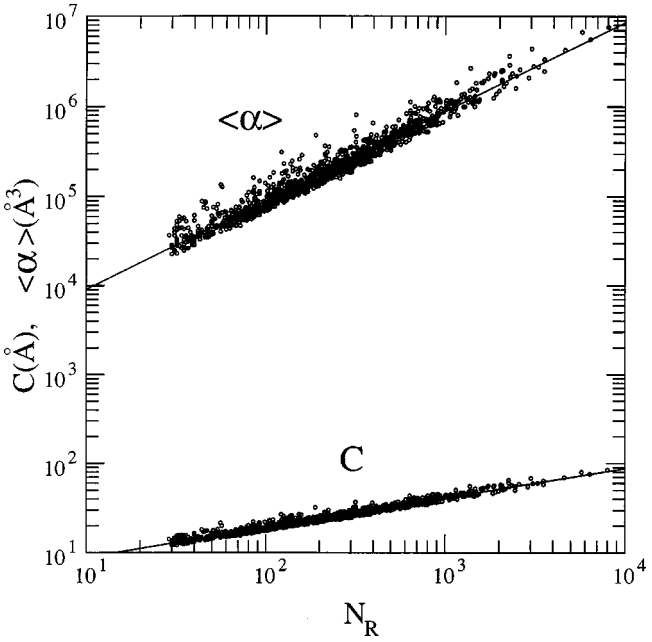

FIG. 3. Log-log plot of the capacitance, C, and the mean polarizability, $\langle\alpha\rangle$, as a function of N_R, the number of residues, for all proteins in the sample set. The best-fit lines have slopes 0.331 and 0.992, respectively.

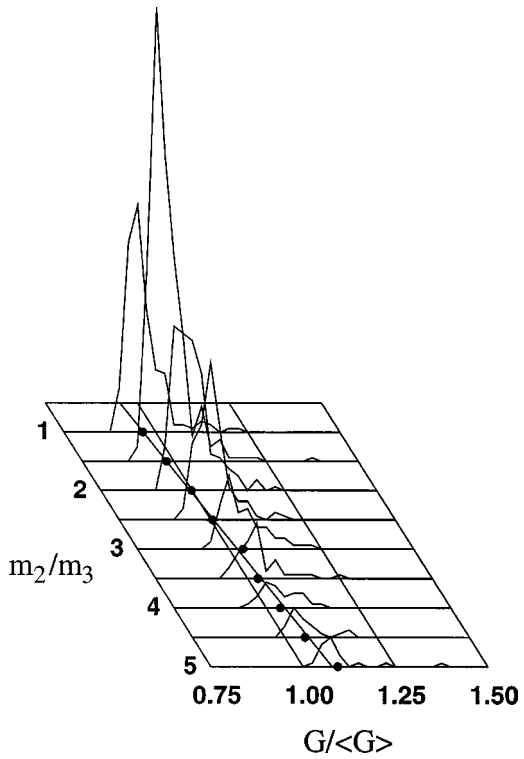

FIG. 4. The quantity G is directly related to the diffusion coefficient of the protein; see Eqs. (7) and (8). This displays the distribution of G over proteins in the sample set as a function of the shape parameter m_2/m_3. The symbols (●) and their best-fit line are traced out on the x-y plane of this diagram and show how G is correlated with protein shape. The equation for the best-fit line is given as Eq. (12).

Figure 5 shows similar data for H, with the following correlation:

$$H \approx [676 + 104(m_2/m_3)] \text{ Å}^3. \tag{13}$$

COMPARISONS WITH EXPERIMENT

In addition to the above calculations, we have performed comparisons with experimental results. Experimental values of either D, R_h, or $[\eta]$ for a number of different proteins were collected from the literature. We also downloaded structural data for the same or highly homologous proteins from the PDB and computed both C and $\langle\alpha\rangle$ by the technique described above. Translational diffusivities depend strongly on temperature, not so much through the temperature appearing in the numerator of the Stokes-Einstein equation as through the temperature dependence of the solvent viscosity appearing in the denominator; which of course also produces solvent dependence. On the other hand, unless the protein denatures, the hydrodynamic radius is more or less invariant with respect to changes in temperature or solvent. Therefore, in this section we draw comparisons between predicted and measured hydrodynamic radii rather than between translational diffusivities. Table II displays our results for a number of different proteins. The agreement is generally good; usually within experimental error. The results are also displayed in Fig. 6.

TABLE I. Calculated properties of several proteins. Note the progression in G or H values as we proceed from globular to elongated proteins. Brackets indicate powers of ten.

PDB code	N_R	C (Å)	$\langle\alpha\rangle$ (Å^3)	G (Å)	H (Å)
1cf9	2908	54.5	2.07 [+6]	3.8	710
1a6m	151	20.7	1.14 [+5]	3.9	750
1aoi	805	43.4	1.09 [+6]	4.7	1350
1qub	319	36.9	8.06 [+5]	5.4	2500

Comparisons of predicted and computed values of $V_h = M[\eta]$ are displayed in Table II and Fig. 7. In these comparisons, only literature citations giving both M and $[\eta]$ have been used, so that V_h could be calculated directly from the experiments. Agreement with experiment is again reasonable in most cases.

In some of the experiments summarized in Tables II and III, the diffusing particle is identified as the monomeric polypeptide, while in the crystal the same or a near homologue displays quaternary structure. In such cases, the monomer was modeled by manually extracting the coordinates of a single chain from the PDB. All such cases are identified in Tables II and III.

Tables II and III also show results for the tobacco mosaic virus. This was modeled as a single cylinder of length 3000 Å and diameter 180 Å. The skin thickness was set at 0.01 Å and 2 million random walks were employed in the calculation.

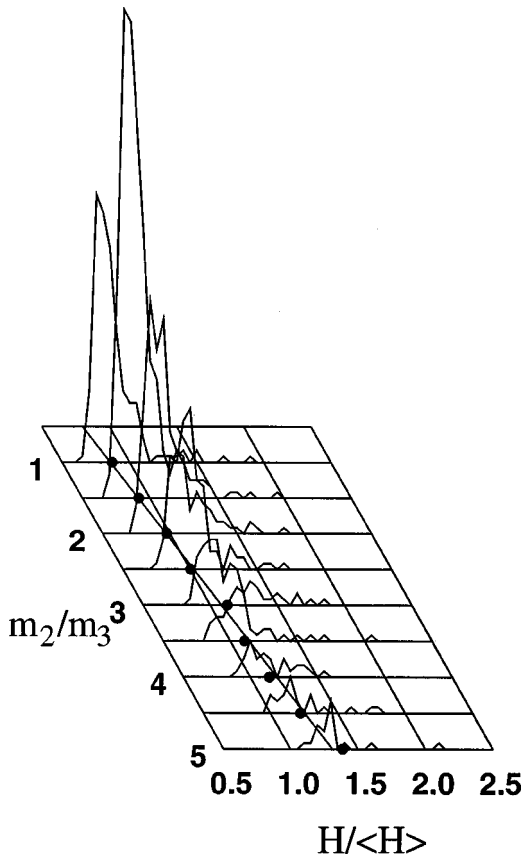

FIG. 5. The quantity H is related to the intrinsic viscosity; see Eqs. (7) and (8). This displays its distribution over proteins of the sample set. It is correlated with the shape parameter m_2/m_3. The best-fit line traced out on the *x-y* plane of this diagram also appears in the text as Eq. (13).

In addition to the examples listed in Tables II and III, the literature contains several empirical or semiempirical correlations that permit the estimation of either D or $[\eta]$ from other properties. As a further confirmation of our technique, we now attempt to predict these correlations directly from the results given above.

Motivated by the Stokes-Einstein equation and anticipating an approximate proportionality between $M^{1/3}$ and R_h, Young, Carroad, and Bell [61] proposed the following expression, in which the coefficient is determined empirically:

$$D=\left[8.34\times10^{-8}\left(\frac{\mathrm{cm}^2}{\mathrm{sec}}\right)\left(\frac{\mathrm{cP}}{\mathrm{K}}\right)\left(\frac{\mathrm{g}}{\mathrm{mol}}\right)^{1/3}\right]\frac{T}{\eta M_m^{1/3}}. \quad (14)$$

Here M_m is the molar mass. To obtain a similar expression from our results requires the proportionality constant between M_m and N_R, which is obviously just the average mass of one residue. Averaging over our data set, we obtain

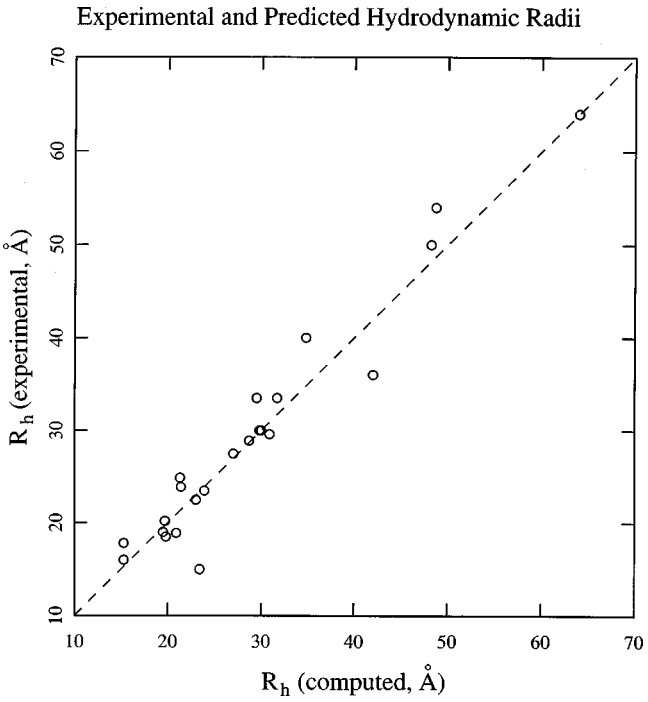

FIG. 6. Comparison of experimental and computed hydrodynamic radii. These data also appear in Table II.

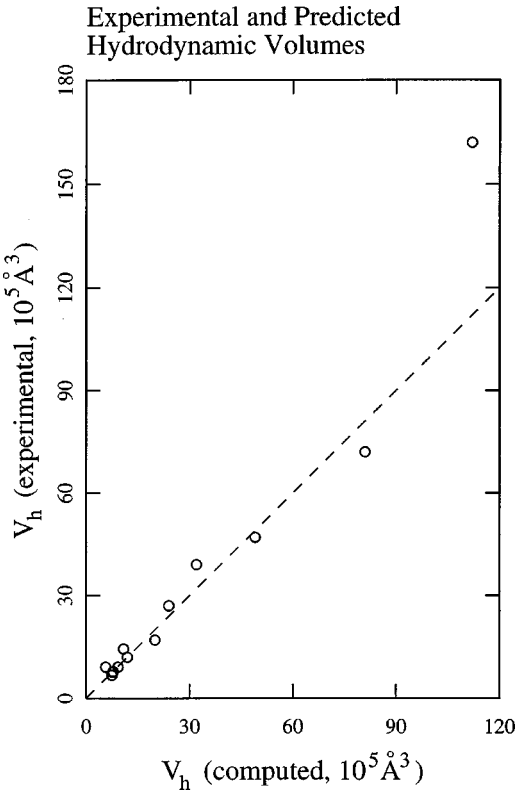

FIG. 7. Comparison of experimental and computed hydrodynamic volumes. These data also appear in Table III.

TABLE II. Comparison between calculated and experimental values of the hydrodynamic radius of proteins. The code of the PDB entry used in the calculation is shown. Experimental uncertainties are displayed whenever supplied in the experimental reference.

Protein	PDB code	R_h (calc) (Å)	R_h (expt) (Å)
434 repressor (1–63) [51]	2r63	15.3	17.8±0.5
bovine pancreatic trypsin inhibitor [37]	4pti	15.3	16
lysozyme [52–55]	6lyz	19.5	18.0±1.0
			19
			20.4±0.2
			18.6±1.0
α lactalbumin [36]	1a4v	19.7	20.2
ribonuclease *A* [36]	1a5q	19.8	18.3
			21
myoglobin [36]	101m	20.9	18.9
γ_D crystallin (monomer)[a] [56]	1e1p	21.3	24.9
γ_B crystallin [57]	4gcr	21.4	23.9±0.3
chymotrypsinogen *A* [36]	1chg	23.0	22.5
green fluorescent protein [58]	1emb	23.4	15±0.5
triose phosphate isomerase (probable monomer)[a,b] [59]	8tim	23.9	23.5
β lactoglobulin [52]	1beb	27.0	27.5±2.0
ovalbumin [36]	1ova	28.7	27.6
			29
			30
phosphoglycerate kinase [36]	1fw8	29.5	33.5
γ_D crystallin (dimer)[c] [56]	1e1p	29.8	30
IgG Fab′ *B*72.3 [8]	1bbj	30.0	30±1
triose phosphate isomerase (dimer) [59]	8tim	30.9	29.6±0.25
hemoglobin [36]	1a3n	31.7	31
			34
			35.5
flagellin [36]	1io1	34.8	40
Hexokinase[a] [36]	1bg3	42.0	36
catalase [36]	4b1c	48.2	48
			52
nitrogenase MoFe [22]	3min	48.7	54±3
ferritin [60]	1mfr	64	64
tobacco mosaic virus[d] [30]		482	490±50

[a]One chain was extracted manually from the PDB file to model the monomer.
[b]A species detected in the experiment was tentatively assigned as the monomer.
[c]The experiment included a synthetic crosslinker (bismaleimidohexane) to stabilize the dimer; this crosslinker was not directly modeled in the calculation.
[d]Tobacco mosaic virus was modeled as a cylinder of diameter 180 Å, length 3000 Å, and skin thickness 0.01 Å.

$$M_m = \left[110.0 \left(\frac{\mathrm{g}}{\mathrm{mol}} \right) \right] N_R . \tag{15}$$

Combining Eqs. (7a), (7b), (10), and (15):

$$D = \left[8.56 \times 10^{-8} \left(\frac{\mathrm{cm}^2}{\mathrm{sec}} \right) \left(\frac{\mathrm{cP}}{\mathrm{K}} \right) \left(\frac{\mathrm{g}}{\mathrm{mol}} \right)^{1/3} \right] \frac{T}{\eta M_m^{1/3}} . \tag{16}$$

By also using the shape dependence of Eq. (12), we obtain the following:

$$D = \left[9.14 \times 10^{-8} \left(\frac{\mathrm{cm}^2}{\mathrm{sec}} \right) \left(\frac{\mathrm{cP}}{\mathrm{K}} \right) \left(\frac{\mathrm{g}}{\mathrm{mol}} \right)^{1/3} \right] \times \left[1 + 0.0302 \left(\frac{m_2}{m_3} \right) \right]^{-1} \frac{T}{\eta M_m^{1/3}} . \tag{17}$$

The discrepancy between Eqs. (14) and (16) is less than 3%, and as Eq. (17) shows, at least some of this discrepancy might be attributed to differences in the shape ratio m_2/m_3

TABLE III. Comparisons between predicted and measured hydrodynamic volumes, $V_h = M[\eta]$, for M the molecular mass and $[\eta]$ the intrinsic viscosity. In many instances, the displayed experimental value is averaged over several experimental results.

Protein	PDB code(s)	V_h (calc)/10^5 Å^3	V_h (expt)/10^5 Å^3
neurophysin monomer[a] [21]	115c,115d	5.7	9.1
lysozyme [21]	6lyz	7.5	6.7
α-lactalbumin [36]	1av4	7.7	7.6
ribonuclease *A* [21]	1a5q	7.9	7.8
myoglobin[a] [21,36]	101m, 1azi	9.3	9.1
neurophysin dimer [21]	1npo	10.9	14.4
chymotrypsinogen *A* [21,36]	1chg	12	12
β-lactoglobulin [21]	1beb	20	17
Ovalabumin [21]	1ova	24	27
hemoglobin[a] [21,36]	1aoo,1a3n	32	39
transferrin[a] [21]	1aiv,1ovt	49	47
hexokinase[b] [36]	1bg3	81	72
catalase [21,36]	4b1c	112	162
tobacco mosaic virus[c] [21]		21900	19000

[a]The computed value represents an average over two different PDB files.
[b]One chain was extracted manually from the PDB file to model the monomer.
[c]Tobacco mosaic virus was modeled as a cylinder of diameter 180 Å, length 3000 Å, and skin thickness of 0.01 Å.

between our sample set and the experimental set used to obtain Eq. (14).

Again motivated by the Stokes-Einstein formula and anticipating a proportionality between R_h and R_g, Tyn and Gusek [36] obtained the following empirical relationship by fitting experimental D-R_g data:

$$D = \left[5.78 \times 10^{-8} \left(\frac{\mathrm{cm}^2}{\mathrm{sec}} \right) \left(\frac{\mathrm{cP\,\AA}}{\mathrm{K}} \right) \right] \frac{T}{\eta R_g}. \tag{18}$$

A comparable equation based on our calculations requires a relationship between R_g and N_R. It is very common, in computing R_g of macromolecules, to represent entire monomers or residues by a single point, but one should remember that this engenders errors comparable to the size of one monomer unit. Such an error is usually negligible for random coils, but is a significant fraction of the radius of all but the largest globular proteins. Therefore, we calculate the radius of gyration of the proteins in our sample set by Monte Carlo integration over a continuous volume represented by the union of 5 Å radius spheres centered on each C_α, obtaining the results shown in Fig. 8. These results are well summarized by the expression

$$R_g = (3.42\ \text{Å}) N_R^{1/3}. \tag{19}$$

(We have also computed R_g by summing over the discrete set of points corresponding to the center of each C_α. This yields an artifactual exponent of 0.37, attributable to the fact that the sum over discrete points is a more severe approximation for small proteins. Arteca has also reported exponents somewhat larger than 1/3 [62], but this also appears to be attributable to a summation over discrete points.)

Combining Eqs. (7a), (7b), (10), and (19) yields

$$D = \left[6.11 \times 10^{-8} \left(\frac{\mathrm{cm}^2}{\mathrm{sec}} \right) \left(\frac{\mathrm{cP\,\AA}}{\mathrm{K}} \right) \right] \frac{T}{\eta R_g}, \tag{20}$$

which agrees with Eq. (18) to better than 6%.

A useful empiricism for the intrinsic viscosity of globular proteins is that $[\eta]$ is typically in the range 2.5 to 6.0 cm^3/g

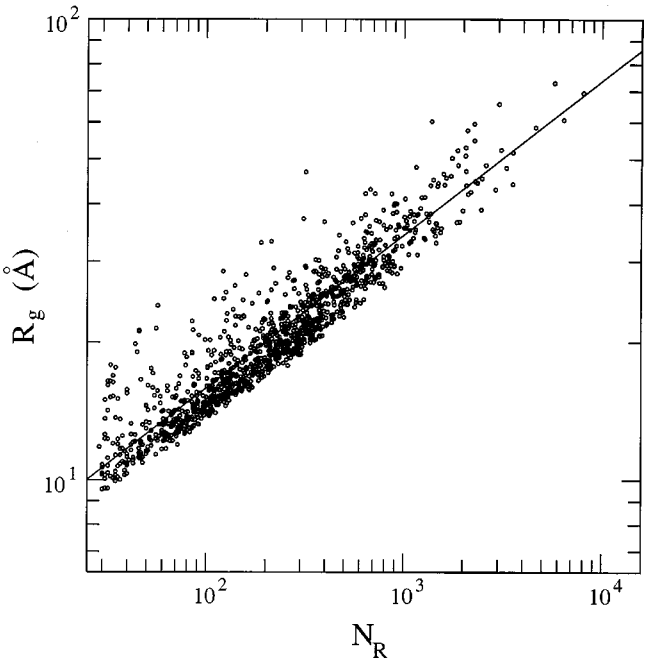

FIG. 8. Log-log plot of radius of gyration R_g vs N_R for all proteins in the sample set. The least-squares line (shown) has slope 0.331.

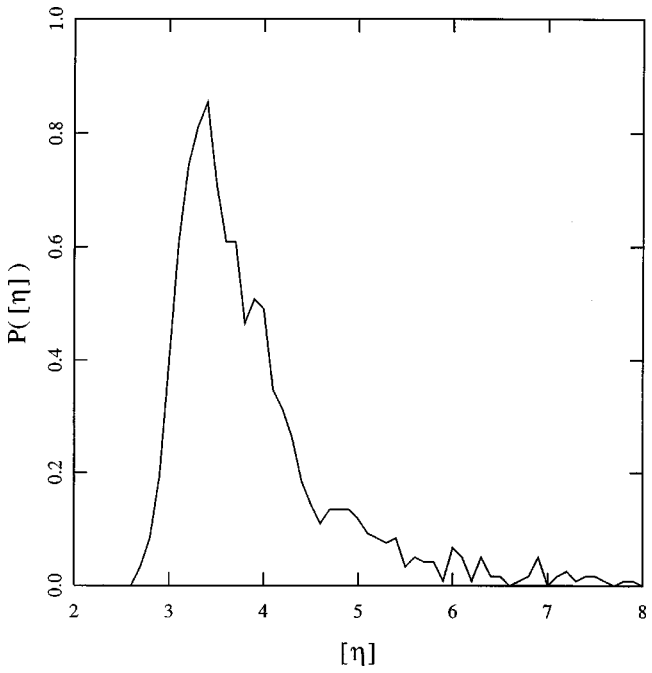

FIG. 9. Probability density of computed intrinsic viscosities of all proteins in the sample set.

and independent of molecular weight [21]. If we combine Eqs. (11) and (15), we obtain

$$[\eta] = 3.93 \ \ \mathrm{cm}^3/\mathrm{g} \tag{21}$$

and including the shape dependence of Eq. (13) gives

$$[\eta] = \left(3.70 \frac{\mathrm{cm}^3}{\mathrm{g}}\right)\left(1 + 0.154 \frac{m_2}{m_3}\right). \tag{22}$$

Figure 9 shows the probability density of $[\eta]$ as calculated from our sample set. Note that our estimates almost always fall within the empirical 2.5 to 6.0 cm^3/g range.

DISCUSSION AND CONCLUSIONS

We have presented a very efficient algorithm for the calculation of the translational diffusivity and the intrinsic viscosity of molecules, and applied it to a large number of protein structures. Comparisons with experiment are usually quite good. The technique, a numerical path integration, is asymptotically rigorous for certain boundary value problems in electrostatics. It is only approximate for the analogous hydrodynamic boundary value problems; however, it has been shown to give high accuracy in all situations in which it has been tested [38,40].

The approach presents several advantages over traditional techniques. First, computation time is $O(m)$, rather than $O(m^3)$, where m is the number of finite elements or beads employed to represent the boundary. Second, models constructed from hydrodynamic beads interacting via the Oseen tensor are fraught with problems when the beads overlap [9]; no such problems are encountered with this technique. Third, although not done here, we can construct models from beads of various sizes and shapes. Again, this is not possible with traditional hydrodynamic treatments.

We obtain the general result that the hydrodynamic radius, the hydrodynamic volume, and the radius of gyration vary on average as $N_R^{1/3}$, N_R, and $N_R^{1/3}$, respectively, consistent with a model of globular proteins as compact, space-filling objects. (See Figs. 3 and 8.). These results are consistent with the well-known empiricisms that diffusivities of proteins are inversely proportional either to the cube root of the molecular mass or to the radius of gyration [see Eqs. (16) and (20)] and that intrinsic viscosities are independent of molecular mass. And although the calculation confirms these empiricisms, it also demonstrates departures from these laws when the proteins are elongated.

Arteca has reported subtle departures from the $N_R^{1/3}$ law for the radius of gyration [62]. He reports that effective exponents change somewhat at about $N_R \approx 300$ and attributes this to statistical changes in the α-helix and β-strand content as N_R increases. We find similar trends in both the hydrodynamic radius and volume, and plan a full report at some future date.

[1] J. Antosiewicz and D. Porschke, J. Phys. Chem. **93**, 5301 (1989).
[2] V. Bloomfield, W. O. Dalton, and K. E. van Holde, Biopolymers **5**, 135 (1967).
[3] V. Bloomfield, K. E. van Holde, and W. O. Dalton, Biopolymers **5**, 149 (1967).
[4] V. A. Bloomfield and D. P. Filson, J. Polym. Sci., Part C: Polym. Symp. **25**, 73 (1968).
[5] O. Byron, Methods Enzymol. **321**, 278 (2000).
[6] B. Carrasco and J. Garcia de la Torre, Biophys. J. **75**, 3044 (1999).
[7] B. Carrasco and J. Garcia de la Torre, J. Chem. Phys. **111**, 4817 (1999).
[8] B. Carrasco, J. Garcia de la Torre, O. Byron, D. King, C. Walters, S. Jones, and S. E. Harding, Biophys. J. **77**, 2902 (1999).
[9] B. Carrasco, J. Garcia de la Torre, and P. Zipper, Eur. Biophys. J. **28**, 510 (1999).
[10] D. P. Filson and V. A. Bloomfield, Biochemistry **6**, 1650 (1967).
[11] J. Garcia de la Torre and V. A. Bloomfield, Biopolymers **16**, 1747 (1977).
[12] J. Garcia de la Torre and V. A. Bloomfield, Biopolymers **16**, 1765 (1977).
[13] J. Garcia de la Torre and V. A. Bloomfield, Biopolymers **16**, 1779 (1977).
[14] J. Garcia de la Torre and V. A. Bloomfield, Biopolymers **17**, 1605 (1978).
[15] J. Garcia de la Torre and V. A. Bloomfield, Q. Rev. Biophys. **14**, 81 (1981).

[16] J. Garcia de la Torre and B. Carrasco, Eur. Biophys. J. **27**, 549 (1998).
[17] J. Garcia de la Torre, S. Navarro, M. C. Lopez Martinez, F. G. Diaz, and J. J. Lopez Cascales, Biophys. J. **67**, 530 (1994).
[18] J. Garcia de la Torre, B. Carrasco, and S. E. Harding, Eur. Biophys. J. **25**, 361 (1997).
[19] J. Garcia de la Torre, M. L. Huertas, and B. Carrasco, Biophys. J. **78**, 719 (2000).
[20] S. E. Harding, Biophys. Chem. **5**, 69 (1995).
[21] S. E. Harding, Prog. Biophys. Mol. Biol. **68**, 207 (1997).
[22] T. Hellweg, W. Eimer, E. Krahn, K. Schneider, and A. Müller, Biochim. Biophys. Acta **1337**, 311 (1997).
[23] J. J. Müller, D. Zirwer, G. Damaschun, H. Welfle, K. Gast, and P. Plietz, Stud. Biophys. **96**, 103 (1983).
[24] B. Spotorno, L. Piccinini, G. Tassara, C. Ruggiero, M. Nardini, F. Molina, and M. Rocco, Eur. Biophys. J. **25**, 373 (1997).
[25] D. C. Teller, E. Swanson, and C. de Haën, Methods Enzymol. **61**, 103 (1979).
[26] M. M. Tirado Garcia, M. A. Jiménez Rios, and J. M. Garcia Bernal, Int. J. Biol. Macromol. **24**, 19 (1990).
[27] R. M. Venable and R. W. Pastor, Biopolymers **27**, 1001 (1988).
[28] P. Zipper and H. Durchschlag, Biochem. Soc. Trans. **26**, 726 (1998).
[29] S. A. Allison and V. T. Tran, Biophys. J. **68**, 2261 (1995).
[30] D. Brune and S. Kim, Proc. Natl. Acad. Sci. U.S.A. **90**, 3835 (1993).
[31] K. S. Chae and A. M. Lenhoff, Biophys. J. **68**, 1120 (1995).
[32] H. Durchschlag and P. Zipper, Prog. Colloid Polym. Sci. **107**, 43 (1997).
[33] H. Durchschlag and P. Zipper, Biochem. Soc. Trans. **26**, 731 (1998).
[34] T. F. Kumosinski and H. Pessen, Arch. Biochem. Biophys. **219**, 89 (1982).
[35] J. J. Müller, Biopolymers **31**, 149 (1991).
[36] M. T. Tyn and T. W. Gusek, Biotechnol. Bioeng. **35**, 327 (1990).
[37] P. E. Smith and W. F. van Gunsteren, J. Mol. Biol. **236**, 629 (1994).
[38] J. F. Douglas, Adv. Chem. Phys. **102**, 121 (1997).
[39] J. F. Douglas and A. Friedman, in *IMA Series on Mathematics and its Applications*, edited by A. Friedman (Springer, New York, 1995), Vol. 67, p. 166.
[40] J. F. Douglas and E. J. Garboczi, Adv. Chem. Phys. **91**, 85 (1995).
[41] J. A. Given, J. B. Hubbard, and J. F. Douglas, J. Chem. Phys. **106**, 3761 (1997).
[42] J. B. Hubbard and J. F. Douglas, Phys. Rev. E **47**, 2983 (1993).
[43] H.-X. Zhou, Biophys. J. **69**, 2286 (1995).
[44] J. F. Douglas, H.-X. Zhou, and J. B. Hubbard, Phys. Rev. E **49**, 5319 (1994).
[45] B. A. Luty, J. A. McCammon, and H.-Z. Zhou, J. Chem. Phys. **97**, 5682 (1992).
[46] M. L. Mansfield, J. F. Douglas, and E. J. Garboczi, Phys. Rev. E **64**, 061401 (2001).
[47] H.-X. Zhou, A. Szabo, J. F. Douglas, and J. B. Hubbard, J. Chem. Phys. **100**, 3821 (1994).
[48] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, Nucleic Acids Res. **28**, 235 (2000).
[49] U. Hobohm and C. Sander, Protein Sci. **3**, 522 (1994).
[50] U. Hobohm, M. Scharf, R. Schneider, and C. Sander, Protein Sci. **1**, 409 (1992).
[51] V. Yu. Orekhov, D. M. Korzhnev, K. V. Pervushin, E. Hoffmann, and A. S. Arseniev, J. Biomol. Struct. Dyn. **17**, 157 (1999).
[52] S. Beretta, G. Chirico, and G. Baldini, Macromolecules **33**, 8663 (2000).
[53] A. Bonincontro, A. De Francesco, and G. Onori, Chem. Phys. Lett. **301**, 189 (1999).
[54] V. Calandrini, D. Fioretto, G. Onori, and A. Santucci, Chem. Phys. Lett. **324**, 344 (2000).
[55] J. J. Grigsby, H. W. Blanch, and J. M. Prausnitz, J. Phys. Chem. B **104**, 3645 (2000).
[56] N. Asherie, J. Pande, A. Lomakin, O. Ogun, S. R. A. Hanson, J. B. Smith, and G. B. Benedek, Biophys. Chem. **75**, 213 (1988).
[57] J. Pande, A. Lomakin, B. Fine, O. Ogun, I. Sokolinski, and G. Benedek, Proc. Natl. Acad. Sci. U.S.A. **92**, 1067 (1995).
[58] U. Kubitscheck, O. Kückmann, T. Kues, and R. Peters, Biophys. J. **78**, 2170 (2000).
[59] C. J. Morgan, D. K. Wilkins, L. J. Smith, Y. Kawata, and C. M. Dobson, J. Mol. Biol. **300**, 11 (2000).
[60] P. Schuck, Biophys. J. **78**, 1606 (2000).
[61] M. E. Young, P. A. Carroad, and R. L. Bell, Biotechnol. Bioeng. **22**, 947 (1980).
[62] G. A. Arteca, Phys. Rev. E **51**, 2600 (1995).