INSTITUTE OF PHYSICS PUBLISHING PHYSICAL BIOLOGY

Phys. Biol. **2** (2005) S127–S136 doi:10.1088/1478-3975/2/4/S07

Constrained geometric simulation of diffusive motion in proteins

Stephen Wells, Scott Menor, Brandon Hespenheide and M F Thorpe

Department of Physics and Astronomy, Arizona State University, Tempe, AZ 85287-1504, USA

E-mail: mft@asu.edu

Received 8 June 2005
Accepted for publication 27 July 2005
Published 9 November 2005
Online at stacks.iop.org/PhysBio/2/S127

Abstract

We describe a new computational method, *FRODA* (framework rigidity optimized dynamic algorithm), for exploring the internal mobility of proteins. The rigid regions in the protein are first determined, and then replaced by ghost templates which are used to guide the movements of the atoms in the protein. Using random moves, the available conformational phase space of a 100 residue protein can be well explored in approximately 10–100 min of computer time using a single processor. All of the covalent, hydrophobic and hydrogen bond constraints are maintained, and van der Waals overlaps are avoided, throughout the simulation. We illustrate the results of a *FRODA* simulation on barnase, and show that good agreement is obtained with nuclear magnetic resonance experiments. We additionally show how *FRODA* can be used to find a pathway from one conformation to another. This directed dynamics is illustrated with the protein dihydrofolate reductase.

M This article features online multimedia enhancements

1. Introduction

The ability of proteins to change their conformation is often vitally important to their function as biological machines. An enzyme, for example, must be able to perform the motions associated with binding a substrate, catalyzing a reaction and releasing the product, reliably thousands of times per second. The study of protein motion, and in particular of the relation between protein structure and protein function, is thus of fundamental importance in biochemistry and molecular biology. Simulation techniques can build upon experimental data, to reveal dynamical structural details that are hard or impossible to obtain experimentally.

The most widely-used approach to computing protein motion is molecular dynamics simulations (MD), which makes use of a classical energy function and then solves Newton's equations of motion for the atoms in the protein. All $3N$ degrees of freedom are included, where N is the number of atoms in the system. This method has been extensively used in the past 25 years and has led to many insights into protein motion [1–5]. Nevertheless, this approach continues to have three principal difficulties. Firstly, it is very computationally demanding; accurate trajectories for the atoms can only be obtained using a very short time step (∼femtosecond), which means that many simulation steps are required. To simulate even 10 ns of biological time requires computer resources of CPU-weeks. Some of the most biologically relevant motions take place on a timescale of milliseconds to seconds, which is much longer than the nanosecond scale typically available in MD. A few simulations, and even folding trajectories, have been obtained up to the microsecond scale for small proteins; however, these required very significant computer resources [6]. Secondly, the quality of the results is dependent on the force-field used [1–5] which are determined phenomenologically using input data from many proteins near the native state. Herculean efforts have been made over the years to improve these potentials, and to include the surrounding water and electrostatic forces [6–11], but there appears to be an inherent limit to this procedure, probably because of the limits of transferability of interactions between different environments. Thirdly, as has become apparent recently, the larger structures now being solved by x-ray crystallography, NMR and cyro-EM, mean that there is a developing need to look at larger proteins, protein complexes, viral capsids, etc. Despite efforts to speed up MD by temperature tempering [12–14] and other techniques [15], this is proving to be a difficult task and new approaches are urgently needed. Progress is being made with coarse-graining methods

such as the Gaussian network model [16–19]. In this paper, we introduce a new method, which focuses on maintaining constraints and geometry, which we believe will prove to be helpful.

The high computational demands of MD arise from the fact that all modes, from the lowest to the highest frequencies, must be simulated, with the high-frequency modes requiring a very short time step ∼femtoseconds. The most relevant motions on biological timescales, however, are not the highest-frequency vibrations but the low-frequency modes associated with large-amplitude diffusive motions. It is these diffusive and low-frequency motions that we focus on in this paper. Other techniques, such as essential dynamics [20] have also done this, but are very computationally intensive, as they analyze trajectories obtained from MD.

An alternative to the dynamical approach of Newton is the constraint-based approach of Lagrange. The Lagrangian formulation of classical mechanics [21] is entirely equivalent to that of Newton. Here we define the constraints which an acceptable conformer of the protein must obey (e.g. maintenance of covalent bond lengths and angles, excluded volumes, hydrogen bonds and hydrophobic tethers) and then search for new conformers within this constraint space. Rather than producing dynamical trajectories, we obtain collections of possible conformations of the protein, which define a trajectory in conformational space that respects the stereochemistry. Our measure of the progression of motion is not time but rather distance. That is, the mobility or extent of the motion is monitored conveniently by the root-mean-square deviation (RMSD) from a reference conformer. Thus the protein is seen as having geometrically defined pathways in the $3N$-dimensional conformational space, where N is the number of atoms. Within this conformational phase space, there are allowed regions and disallowed regions, which can be thought of as having potentials of zero and infinity respectively. *FRODA* (framework rigidity optimized dynamic algorithm) finds continuous pathways as it traverses the allowed regions.

In this paper, we present the details of this method of *geometric simulation* for proteins, with examples. The method is *atom-led* in the sense that the principal variables are the atomic positions rather than the dihedral angles. The present method avoids most of the difficulties associated with an earlier attempt at geometrical simulation called *ROCK* (rigidity optimized conformational kinetics), which emphasized multiple ring closure [22]. *FRODA* is about 100 to 1000 times faster than *ROCK*, and treats all atoms equivalently, whether they are in rings or not, main-chain or side-chain. *FRODA* allows for evaluation of non-covalent interaction constraints, such as steric clashes and hydrophobic tethers, simultaneously with the geometric constraints. Hydrogen bonds, after they have been identified, are treated in a similar way to covalent constraints. Constraints are enforced using rigid *ghost templates* defined using the various fixed constraints, while variable dihedral angles are handled by the juxtaposition of adjacent templates, as described in more detail below. The algorithm produces ring closure implicitly, and no distinction need be made between main and side chain atoms. The algorithm is O(N), making it suitable for application to large proteins, and rapid in execution, so that motions of several Å in RMSD can be achieved using minutes of CPU time for typical proteins with ∼100 residues.

The constraint information is generated by performing rigidity analysis on the protein using the *pebble game* approach [23, 24] as used in the *FIRST* code [25]. A detailed description of the body-bar formalism used here is provided in [26]. The rigidity analysis and geometric simulation software is available in the *FIRST*/*FRODA* package is written entirely in ANSI C++. We note that the word *FRODA* is not new, as a King Froda who lived around 500 AD is mentioned in Beowulf [27][1].

2. Rigidity analysis (*FIRST*)

Since rigidity analysis provides us with the input information necessary to perform geometric simulation, we here briefly review the concepts of network rigidity and the pebble game algorithm for identifying rigid clusters. We consider a protein as a network in which all covalent bond lengths and angles are fixed (constrained), and the covalent double bonds are locked (constrained). Constraints are also assigned to hydrophobic interactions and hydrogen bonds, which are determined by using the local chemistry and geometry as input [25]. Changes in the shape of the protein occur by changes in dihedral angles of rotatable bonds. Rigidity analysis, using the pebble game and *FIRST*, determines which dihedral angles are rotatable and which are locked. Those dihedral angles that are rotatable, usually do so in a coherent way with other rotatable dihedral angles in the vicinity, and rarely independently, except sometimes on side chains and at termini. While certain bonds are considered locked *a priori*—for example the C=N double bond in the peptide backbone—the rigidity of the three-dimensional folded protein is determined by the constraints introduced by hydrogen bonds and hydrophobic tethers.

Determining the rigidity of the protein is then a matter of balancing degrees of freedom against constraints. In some regions of the protein the degrees of freedom will outnumber the constraints, rendering these regions flexible. In other regions the degrees of freedom and the constraints exactly balance each other, making these regions isostatically rigid (i.e. with exactly the right number of constraints and no more). Finally, in some regions there are more constraints than there are degrees of freedom, making these regions over-constrained or stressed and containing redundant constraints. Just as the good design of buildings demands that sufficient redundancy be present, so it is with proteins, especially in the core region [25]. Rigid regions will have flexible hinges between them, and these are important in determining the geometrical pathways that will be found by *FRODA*. It is

[1] *Beowulf* was written around 1000 AD by an anonymous author. King Froda lived around 500 AD. "Beowulf gives his uncle the king not mere gossip of his journey, but a statesmanlike forecast of the outcome of certain policies at the Danish court. Talk of interpolation here is absurd. As both Beowulf and Hygelac know, -and the folk for whom the Beowulf was put together also knew, -Froda was king of the Heathobards (probably the Langobards, once near neighbors of Angle and Saxon tribes on the continent), and had fallen in fight with the Danes."

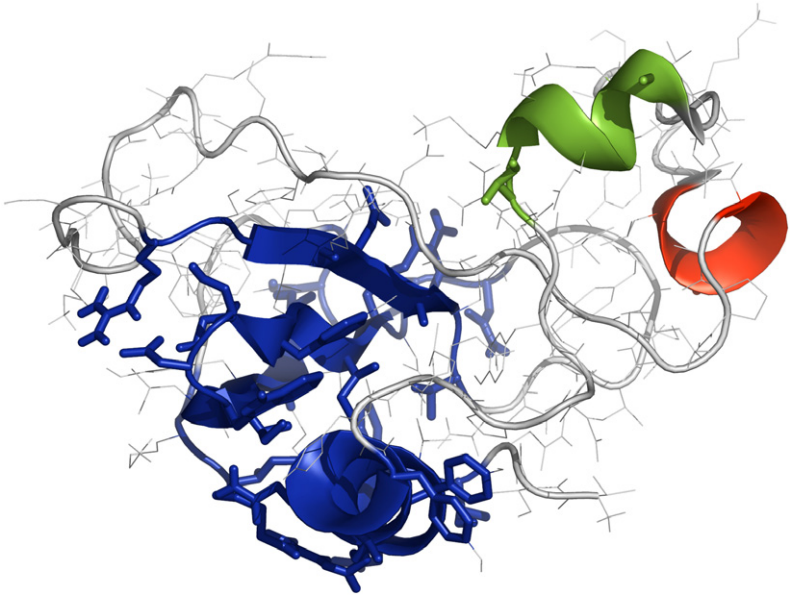

Figure 1. An example of a rigid cluster decomposition using the pebble game in *FIRST* on the protein barnase (PDB code: 1A2P [30]), showing the largest rigid regions in solid colors (blue, green and red). For most proteins, the sizes of the rigid clusters vary from hundreds to thousands of atoms for the largest rigid cluster (the 'rigid core' of the protein, in blue), through tens to hundreds of atoms for mobile rigid clusters such as alpha-helices, down to groups of three atoms in the most flexible parts of a chain.

important to note that the pebble game algorithm in *FIRST* determines hinges from a static perspective—nothing actually moves, but the potential for motion has been identified. This is much like identifying the hinges on a door, without moving the door. The actual motion of the door and the angle the door can move through is determined by making movements and determining if the motion is free within a certain range or if there is a chair or other obstruction in the way which restricts the amplitude of the motion of the door. It is this latter dynamical task that we focus on here with the *FRODA* algorithm.

This matching of degrees of freedom to constraints cannot be done as a global average, or the distinction between over-constrained and flexible regions would not be apparent. Nor is it purely local, as an atom may be part of a collective mode involving many atoms but only a few degrees of freedom. The pebble game [23–25] is an algorithm for distributing the degrees of freedom belonging to the atoms ('pebbles') over the bonds ('constraints') so as to determine the rigidity. Here we treat the protein as a 'body-bar' graph [26]; so that each atomic site is a body with six degrees of freedom, while each type of constraint brings with it a number of 'bars' (lost degrees of freedom) ranging from 2 for a hydrophobic tether to 5 for a single covalent bond or hydrogen bond and 6 for a locked (double, peptide/aromatic) covalent bond. The pebble game consists of distributing pebbles across bars according to certain rules so that the protein divides into regions with excess bars (stressed), regions with exactly as many bars as pebbles (isostatic) and regions with excess pebbles (flexible). The pebble game is a fast, integer algorithm scaling as order $O(N)$ in the number of sites, for most cases, and $O(N^{1.2})$ near a phase transition [28], where allosteric effects routinely spread across the entire protein. The decomposition of a protein, with ~100 residues, into rigid and flexible regions using *FIRST*, such as that shown in figure 1, takes only a few seconds and the method has successfully been applied to systems as large as an entire viral capsid with nearly half a million atoms [26].

Hydrogen bonds are identified with an energy scale based on their geometry, with energies ranging from 0 down to -10 kcal mol^{-1}. A user-defined energy cutoff determines which bonds to include and which not, with the default being, -1 kcal mol^{-1}. A 'dilution plot' can be produced showing how the rigidity of the protein depends on the cutoff, with a lower cutoff producing more and smaller rigid clusters [29]. This allows for the selection of relevant motions; for example, if the relative motion of two domains in a protein is of interest, the cutoff should be chosen so as to make those domains two separate rigid bodies.

3. Geometric simulation (*FRODA*)

While flexibility and mobility are closely connected concepts, they are not identical. For example, a structure such as an alpha helix may be rigid (i.e. its shape does not change during motion) but mobile relative to the core of the protein, so long as it is flanked by flexible regions to act as hinges; in the same way that a door is a rigid object but is mobile relative to its frame. As we will now describe, geometric simulation is an efficient method to obtain the mobility that results from flexibility.

The central concept of geometric simulation is the replacement of interatomic potentials by fictitious rigid bodies (the *ghost templates*). Atoms are bound, not to each other, but to the vertices of these rigid ghost templates. Constraints on bond lengths and angles are enforced by an iterative process in which the ghost templates are fitted to the atomic positions and then each atom is fitted to the position(s) of the vertex or vertices to which it belongs. The *position* and *orientation* of a

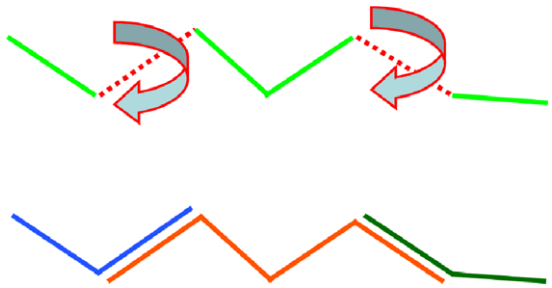

Figure 2. A polymer chain with two rotatable dihedral angles (upper sketch) is associated with three overlapping ghost templates (lower sketch). Each group of mutually rigid atoms belongs to a common ghost template, and atoms associated with a rotatable bond belong to more than one ghost template.

ghost template are given by a least-squares fit to the positions of the atoms belonging to it. Since the ghost template is rigid, it has only six degrees of freedom regardless of the number of atoms it shadows. In the atom-fitting step, each atom considers only the positions of one or a few vertices in the ghost templates, rather than many atoms having to be taken into account. The ghost template is thus a very economical way of representing a large number of interlocking interatomic constraints. This approach was originally developed to simulate rigid-unit motion in framework mineral structures [31, 32], where the ghost templates were all identical, for example corner-sharing SiO_4 tetrahedra in quartz. Here the ghost templates can vary from two bonds connecting three atoms (see figure 2) up to hundreds of thousands of atoms that define a rigid virus capsid [26].

In simulating the flexible motion of a protein, we constrain bond lengths and bond angles, while permitting some dihedral angles to vary. Such constraints are represented by overlapping ghost templates along the rotatable bond. Consider the case of an ethane molecule at high temperature, with rotation permitted about the C–C sigma bond as shown in figure 3, which involves a single degree of freedom. In figure 3(*a*), the molecule is represented by (figure 3(*b*)) two ghost templates overlapping along the bond, so that each hydrogen atom is associated with a single vertex while each carbon atom is associated with two vertices, one in each ghost template. If we perform a random displacement of every atomic position (figure 3(*c*)), we break all of the constraints. To reintroduce them, we fit the ghost templates to the new positions of the atoms (figure 3(*d*)), and then refit the atoms to the ghost templates (figure 3(*e*)). At this stage the ghost templates and atoms are not yet co-located, indicating that the constraints have not yet been re-established. Over successive iterations of these steps—fitting the ghost templates to the atomic positions (figure 3(*f*)) and then the atoms to the ghost templates (figure 3(*g*))—the molecule returns to a sterically correct and new conformation (figure 3(*h*)) which respects all the constraints to within a user-defined tolerance, e.g. no mismatch with the ghost templates greater than 0.1 Å, or better if required. The higher the tolerance, the more CPU time required.

Essentially, each ghost template represents the constrained form of a part of the protein. If all the atoms belonging to a ghost template are close to their vertices, then the constraints on that part of the protein are satisfied. When all atoms and ghost templates are closely matched, then we have a protein conformer which satisfies all the constraints. This implicitly produces ring closure without having to explicitly solve for a consistent set of dihedral angles. Indeed, the algorithm is not explicitly aware of whether atoms are members of rings or not. This is a crucial reason why this procedure is more efficient, and hence computational faster, than that used in *ROCK* [22].

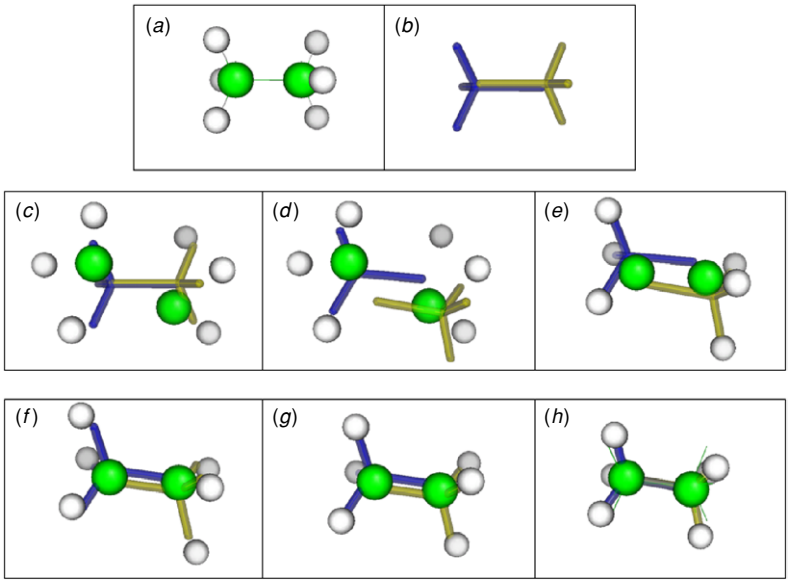

Figure 3. The motion of an ethane molecule as determined by geometric simulation. (*a*) Initial atomic positions; (*b*) ghost templates; (*c*) random atomic displacement; (*d*) fitting of ghost templates to atoms; (*e*) refitting of atoms to ghost templates; (*f*) and (*g*) further iterations of (*d*) and (*e*); (*h*) until a new conformer is found.

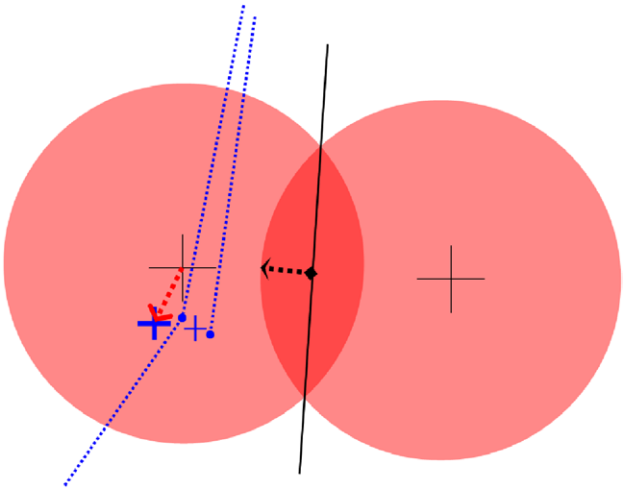

Figure 4. Showing how steric overlap is handled. When atoms come into contact, the overlap influences the movement of the atoms during the relaxation. In the above case, the atom on the left is tethered to sites (blue dots) in two ghost templates (shown as blue dotted lines). The center of the atom, shown as a large black cross, would move to the small light blue cross in order to match the midpoint of its two associated ghost template sites. However, the steric overlap (black dashed arrow) causes it to move further away from the touching atom, so that its new position is at the larger blue cross, and the resultant motion is shown by the dashed red arrow.

Table 1. Showing the van der Waals radii for chemical elements found in proteins, and also the cutoff currently used as acceptable in *FRODA*, which is 62% of the full radius. The option exists within *FRODA* to assign smaller radii to polar hydrogen atoms if desired.

Species	Full radius	Cutoff radius
H	1.0	0.62
C	1.7	1.05
N	1.55	0.96
O	1.4	0.87
S	1.8	1.17
P	1.8	1.17
Mg	1.5	0.93
Si	2.1	1.30
Mn	1.4	0.87

While the ghost templates enforce covalent and hydrogen-bond constraints, there are other constraints that must also be respected—in particular, every atom carries with it an excluded volume. This corresponds to an inequality rather than an equality, in the sense that two atoms have a minimum acceptable separation but no maximum, and so is treated differently. Steric overlap is handled by adding a small bias to the atomic motion during the fitting process, such that atoms which come into contact move to avoid each other, as shown in figure 4. The maximum degree of overlap in an acceptable conformer, as a fraction of the sum of van der Waals radii, is a user-defined parameter in *FRODA*, with a value currently set of 0.62, for consistency with the cutoff radii used in *ROCK*. Table 1 shows, for each of the common elements found in proteins, its full radius (with two atoms being considered in contact if their distance is less than the sum of full radii) and cutoff radius currently used (a conformer is not acceptable if any pair of atoms are closer than the sum of their cutoff radii).

We find that maintenance of steric constraints during fitting ensures that Ramachandran angles are well maintained, e.g. all variable dihedral angles are rarely found in the 'forbidden' regions of the Ramachandran ϕ–ψ plot [33]. Again this is in contrast to *ROCK*, where stereochemistry was checked only after solving for dihedral angles, so that new conformers were quite often rejected as they lie in forbidden regions of the Ramachandran plot.

Hydrophobic tethers require two atoms to remain close to each other within a cutoff (taken in *FIRST*/*FRODA* to be the sum of the atomic radii plus 0.5 Å), while permitting relatively free movement within that cutoff. That is, hydrophobic interactions do not bring with them the strict bond length and angle constraints of a covalent or hydrogen bond. We model hydrophobic interactions similarly to steric interactions using small biases during fitting. If a pair of atoms, identified as hydrophobically-tethered in the initial structure, begin to move apart beyond the cutoff, biases are added to their motion bringing them closer together again. If the atoms are found to be mutually rigid by the pebble game, then of course their relative positions will be maintained by a ghost template, so that no special steric or hydrophobic handling is required.

To simulate a protein, therefore, we first use the rigidity analysis in *FIRST* to determine the rigid clusters and the variable dihedral angles. A ghost template is then created 'covering' every rigid cluster, with the ghost templates overlapping across the rotatable bonds. We then explore the phase space of conformational variation by performing random displacements of all atoms (e.g. typically by around 0.1 to 0.4 Å) followed by several cycles of atom-ghost template fitting so as to restore all constraints to within tolerances. Once a new acceptable conformer is found, it is reported, and then the new coordinates are taken as the starting point for the next set of random displacements. If the geometric simulation does not converge within some reasonable number of cycles—as is possible if a conformation lies near an extreme of the range of allowed motions—then the process restarts from an earlier conformation. The flow of the algorithm is shown in figure 5 and the exploration of the allowed conformational space is illustrated in figure 6. Usually we keep the largest rigid cluster immobile during the simulation and measure motion relative to this 'rigid core'; however, this is not essential to the method and all atoms can be made mobile, which would correspond to adding six degrees of freedom to the rigid core, representing its six rigid body modes.

The algorithm *FRODA* is rapid and scales roughly linearly with the number of moving objects. Each conformer differs from the previous one by a small RMSD of ∼0.1 Å, but over some thousands of iterations large flexible motions can be explored with some portions of the protein moving quasi-continuously by several Ångstroms. Eventually the RMSD relative to the initial conformer will saturate, indicating that the full range of motion allowed by the chosen set of constraints has been explored. Since the time for conformer generation is very short (0.1–1 s on a single processor) for proteins of order 100 residues, such an exploration requires only 10 to 100 min or so of CPU time on a single processor. This represents a tremendous saving of time and computational resources compared to MD.

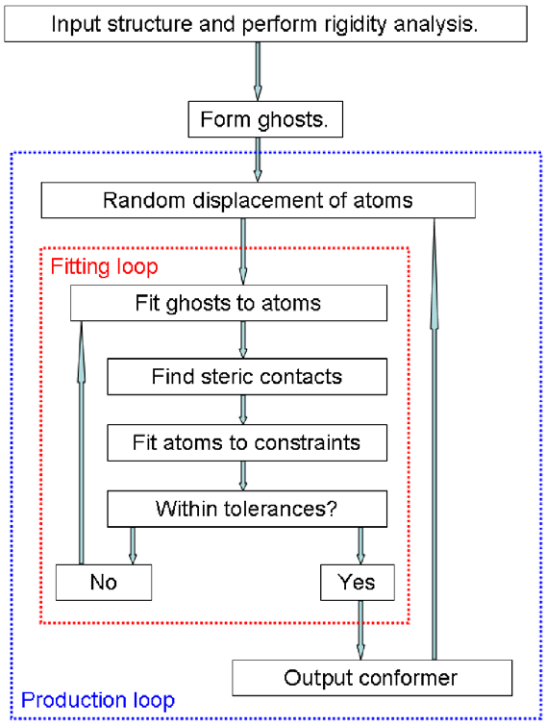

Figure 5. Showing the program flow for *FIRST*/*FRODA*. An initial rigidity analysis leads to the creation of the ghost templates. Each iteration of the *FRODA* production loop involves an initial random displacement of the atoms followed by multiple iterations of the fitting loop, in which the geometric and steric constraints are enforced.

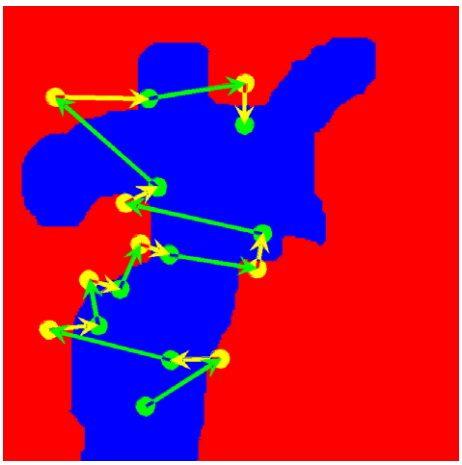

Figure 6. Searching the landscape of flexible motion with geometric simulation. Here we show a sketch of a two-dimensional slice through the $3N$-dimensional phase space of the protein, divided into allowed (blue) and forbidden (red) regions. Successive random moves (green arrows) followed by enforcement of the constraints (yellow arrows) produce a set of conformers (green circles) that explore the allowed region.

Since the very special functional properties of proteins have been produced by evolution, we may expect the allowed pathways of large-scale diffusive motion to be quite simple and largely deterministic. This would be quite different for an arbitrary molecular cluster of a few thousand atoms, which would be expected to have very chaotic and complex geometrical pathways riddling phase space.

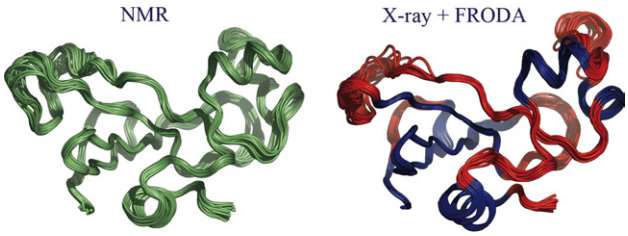

Figure 7. Showing a comparison between the NMR ensemble of conformers for barnase (PDB code: 1BNR) [35] and the set of *FRODA* conformers generated from a single x-ray crystallographic structure (PDB code: 1A2P) [34]. Both panels contain 20 conformers. The *FRODA* conformers are selected from a much larger set and well spaced apart.

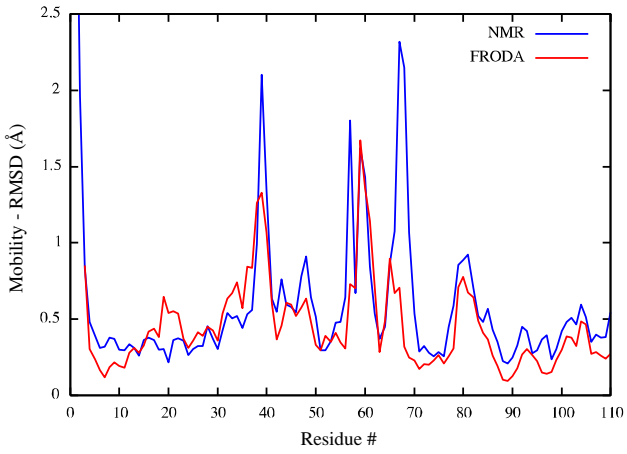

Figure 8. Plotting the mobility (C_α RMSD) of each residue, as determined using data from figure 7, and showing more clearly that *FRODA* captures the main features of the mobility of the protein, when compared with the NMR data.

4. Example of undirected motion: barnase

Barnase is a 110 residue single-domain exotoxin with ribonuclease activity, produced by *Bacillus amyloliquefaciens*. Barnase can be co-expressed and co-folded in *E. coli* with an inhibitor, barstar. Without barstar, barnase is lethal. The function of barnase has not been elucidated but it is presumed to function as a digestive enzyme or as an exotoxin against predators or competitors [34].

As an example of the exploration of the flexible motion of a protein, we present a *FRODA* simulation of barnase. The rigidity analysis was performed by enforcing all covalent and hydrophobic constraints, together with those hydrogen bonds with energies lower than –1 kcal mol^{-1}. We then carried out several thousand iterations of geometric simulation until the RMSD saturated, at which point we conclude that we have explored all accessible regions of phase space. This process took about 50 CPU-min. The results are shown in figures 7 and 8, where they are compared with NMR results.

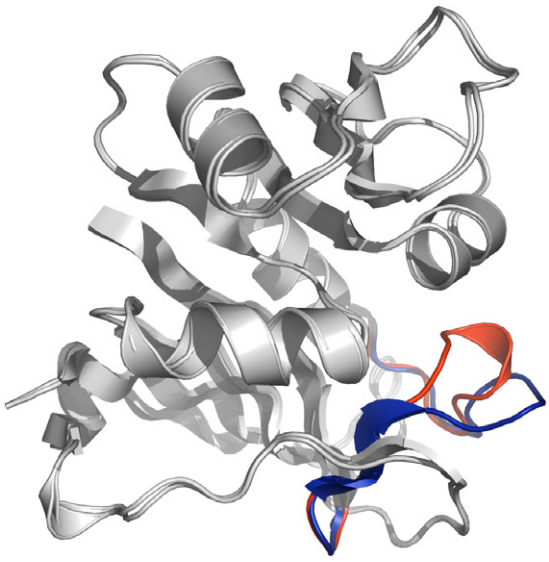

Figure 9. Overlay of two crystal forms of dihydrofolate reductase, showing the mobile M-20 loop in the closed (red; PDB code: 1RX6) and occluded (blue; PDB code: 1RX1) position [36].

The background in figure 8 is too low for the *FRODA* results when compared to the NMR data, because of the suppression of the high-frequency motions, which would be expected to add an additional background of around 0.2 Å. This background is expected to be rather featureless. The overall agreement between the NMR and *FRODA* mobilities is good, both as regards the parts of the protein that are mobile and the relative magnitudes of the motions. The small differences that exist (for example, *FRODA* slightly overestimates the motion around residue 20 and underestimates that around resiudes 68–70) are partially attributable to small mismatches between the constraints obtained geometrically from the crystal structure by *FIRST* and those which are active in solution during the NMR experiment, and also to residual errors in the experimental and simulation data.

5. Example of conformational change: DHFR

Dihydrofolate reductase (DHFR) is an enzymatic protein which catalyzes the reduction of 7,8-dihydrofolate (DHF) or folate to 5,6,7,8-tetrahydrofolate, which plays a key role in the biosynthesis of purines and thymidylate, which are essential components of DNA. The enzyme DHFR is present in all living organisms. Analysis [36] shows that there are three DHFR conformations observed in crystallographic structures: the open (PBD code: 1RX1), closed (PDB code: 1RX6) and occluded (PDB code: 1RA9) conformations [37], and are all well resolved with 2.0 Å resolution. The three conformations are similar, with slight differences in the orientation of the upper domain, and significant differences in the loop region of residues 14–24, which is conventionally called the M-20 loop, and shown in figure 9.

The algorithm as described so far involves random atomic moves followed by cycles of geometric fitting to re-enforce the constraints. It is also possible to add directional biases (decreasing the RMSD to the target) to the atomic motions, so that they are not completely random. This allows us to explore the conformational pathway between two conformers. If multiple conformers of a protein are known—for example from crystallization under different pH conditions, or with and without substrates bound—then the coordinates of one conformer can be treated as a target. Geometric simulation

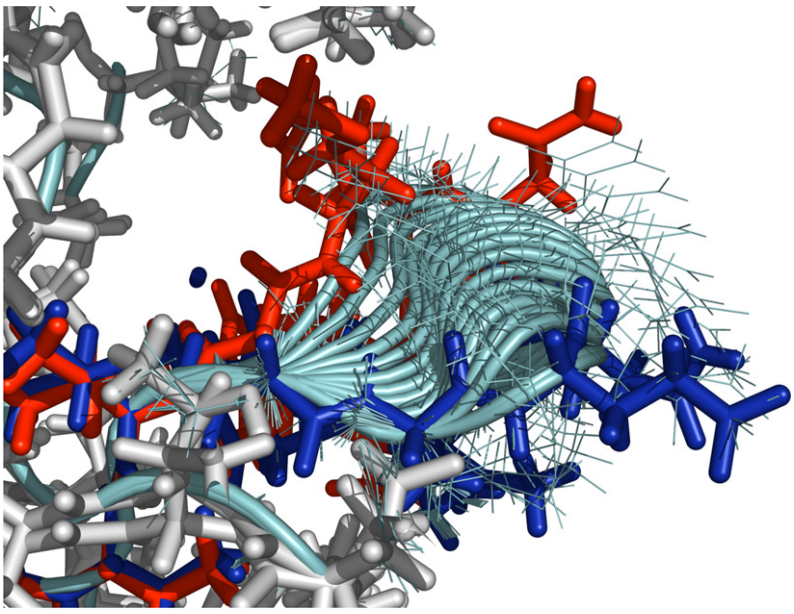

Figure 10. Showing a series of *FRODA* steps superimposed from a directed geometric simulation of the M-20 loop motion in DHFR. Atomic steps were biased toward the coordinates of the closed form (red) from the occluded form (blue). The motion is not a linear morph; each intermediate frame is a geometrically and sterically valid conformer.

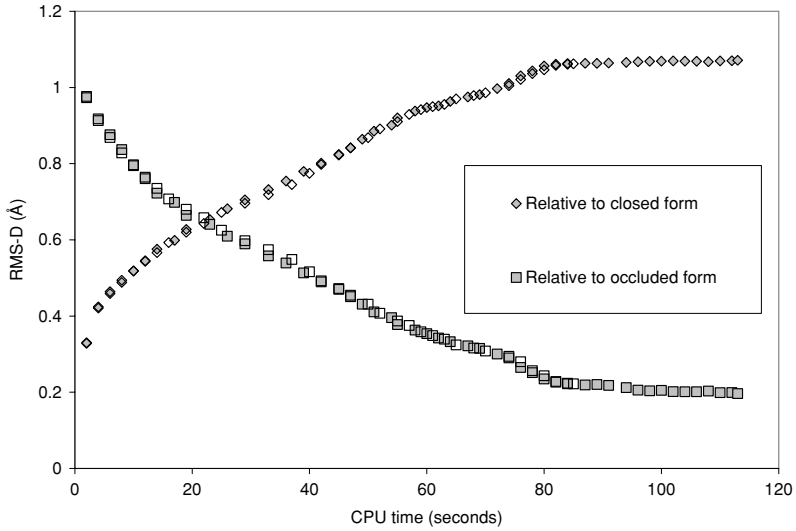

Figure 11. Main-chain atom RMSD in Å of conformers relative to the closed and occluded structures of DHFR [36, 37] as a function of CPU-time for DHFR. Initial coordinates were taken from the closed form and motion was targeted toward the occluded form. The simulation finished when a conformer was found to be within 0.2 Å RMSD of the target. Two independent runs are shown by the open and closed symbols. Note that each run took less than 2 min.

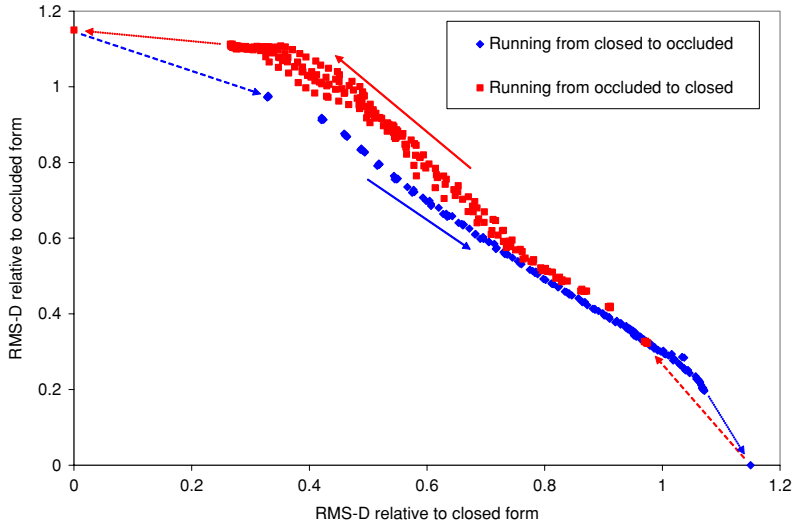

Figure 12. Main-chain atom RMSD in Å of conformers from closed and occluded structures of DHFR [36, 37]. Paths are shown for both the closed to occluded (blue diamonds) and occluded to closed (red squares) directions; five independent runs are overlaid for each series. Note that the pathways from the occluded to the closed forms display some variation in the final stages (top left). The end points are shown by symbols on the axes. Note that the initial part of each pathway is shown by a dashed arrow, and was traversed quickly. The last (missing) part of the pathway is shown by the dotted arrow.

then generates a pathway from one conformer to the other. The random step provides a degree of simulated annealing which allows the protein to find its way past small local hindrances. The flexibility relevant to the change is found by enforcing only those hydrogen-bond and hydrophobic constraints which are *common* to both the initial and target conformations of the protein.

A directed simulation of the conformational change of dihydrofolate reductase from the 'occluded' to the 'closed' crystal forms is shown in figure 10. The simulation took less than 5 min to approach the target conformer within 0.2 Å main-chain RMSD. The random step limits the ultimate accuracy of the fit; however, when a conformer is found that is close to the target, the residual distance may be covered by directed motion without random steps. In figure 11, we show the CPU-time progression between the two conformers for two directed runs.

A question that arises in the simulation of pathways for conformational change is, what degree of variation is possible in the pathway?—that is, whether the path is highly

constrained (a narrow channel) or only loosely constrained (a broad highway). Since we can run multiple pathway simulations rapidly using geometric simulation, we can start to explore this question of pathway variation.

Figure 12 shows the main-chain RMSDs to the closed and occluded forms during multiple targeted runs. We note that the simulations running from the occluded to the closed form show a small degree of variation in the later stages, indicating variations in the exact pathway of conformational change in the loop. Simulations running in the opposite direction do not show this variation, presumably because the variable region lies at the beginning of the path (near the closed form) and the directed steps always follow the same trajectory. The implication is that examination of both directions for conformational change is necessary in order to assess variability in the pathway. The last (missing) part of the pathways is due to the finite size of the random steps coupled with small experimental differences in the measured covalent bond lengths and angles in the two closed and occluded conformers.

6. Program availability

The geometric simulation algorithm for proteins has been programmed as a module called *FRODA*—framework rigidity optimized dynamic algorithm. The combined *FIRST*/*FRODA* code for rigidity analysis and flexible motion simulation is written entirely in ANSI C++ and is available gratis to academic users via Flexweb (flexweb.asu.edu). The source code can be downloaded or the program can be run interactively using flexweb. Note that the graphics used in figures 1, 9 and 10 were generated with PyMOL [38].

7. Conclusions

An alternative to the molecular dynamics (Newtonian) approach to protein simulation is the constraint-based (Lagrangian) generation of conformers. Rigidity analysis using the pebble game in *FIRST* sets and analyses the set of constraints, determines rigid clusters and the flexible joints between them. Geometric simulation is a technique for enforcing the constraints using ghost template bodies. This allows the use of the atomic coordinates as primary variables and the simultaneous enforcement of geometric and steric constraints, avoiding the use of dihedral angles as variables, and thereby enforcing ring closure implicitly. The resulting algorithm is O(N) and rapid in execution, as no effort is spent on simulating high-frequency motions and no potential is evaluated or differentiated. As a result the flexible motion of a typical protein can be investigated in 10 to 100 min. This represents a saving of many orders of magnitude, up to a million in some cases, relative to conventional molecular dynamics simulations.

Since the algorithm makes no distinction between main-chain and side-chain motions, or indeed between the motions of large proteins and small flexible molecules, geometric simulation suggests itself as a powerful tool in drug design, including virtual screening. Many possible conformers of a protein and its substrate can be generated rapidly for testing with standard docking approaches [39–41]. Because this technique is rapid, it opens up the possibility of doing real time interactive manipulations of protein structures, which will be useful in feasibility studies for experiments and subsequent analysis.

Acknowledgments

We should like to acknowledge support from the NIH under grant number GM067249 and from the Biodesign Institute at Arizona State University. We should like to thank Ming Lei and Maria Zavodszky for continuing discussions on geometric simulation.

References

[1] Brooks B R, Bruuccoleri R E, Olafson B D, States D J, Swaminathan S and Karplus M 1983 *J. Comput. Chem.* **4** 187–217
[2] Case D A *et al* 1999 *AMBER 6* (San Francisco, CA: University of California)
[3] Pearlman D A, Case D A, Caldwell J W, Ross W R, Cheatham T E, Debolt S, Ferguson G, Seibel G and Kollman P A 1995 *Comput. Phys. Commun.* **91** 1–41
[4] Kale L, Skeel R, Bhandarkar M, Brunner R, Gursoy A, Krawetz N, Phillips J, Shinozaki A, Varadarajan K and Schulten K 1999 *J. Comput. Phys.* **151** 283–312
[5] Humphrey W, Dalke A and Schulten K 1996 *J. Mol. Graph* **14** 33–8
Humphrey W, Dalke A and Schulten K 1996 *J. Mol. Graph* **14** 27–8
[6] Duan Y and Kollman P A 1998 *Science* **282** 740–4
[7] Gilson M K, McCammon J A and Madura J D 1995 *J. Comput. Chem.* **16** 1081–95
[8] Procacci P, Marchi M and Martyna G J 1998 *J. Chem. Phys.* **108** 8799–803
[9] Vorobjev Y N, Almagro J C and Hermans J 1998 *Proteins* **32** 399–413
[10] Luise A, Falconi M and Desideri A 2000 *Proteins* **39** 56–67
[11] Borgis D and Hynes J T 1991 *J. Chem. Phys.* **94** 3619–28
[12] Andricioaei I, Straub J E and Voter A F 2001 *J. Chem. Phys.* **114** 6994–7000
[13] Stolovitzky G and Berne B J 2000 *Proc. Natl Acad. Sci. USA* **97** 11164–9
[14] Faken D B, Voter A F, Freeman D L and Doll J D 1999 *J. Phys. Chem.* A **103** 9521–6
[15] Mousseau N, Derreumaux P, Barkema G T and Malek R 2001 *J. Mol. Graph Model* **19** 78–86
[16] Brooks B and Karplus M 1983 *Proc. Natl Acad. Sci. USA* **80** 6571–5
[17] Tama F and Sanejouand Y H 2001 *Protein Eng.* **14** 1–6
[18] Tama F, Gadea F X, Marques O and Sanejouand Y H 2000 *Proteins* **41** 1–7
[19] Atilgan A R, Durell S R, Jernigan R L, Demirel M C, Keskin O and Bahar I 2001 *Biophys. J.* **80** 505–15
[20] Amadei A, Linssen A B and Berendsen H J 1993 *Proteins* **17** 412–25
[21] Goldstein H, Poole C P and Safko J L 2002 *Classical Mechanics* 3rd edn (San Francisco, CA: Addison-Wesley)
[22] Lei M, Zavodszky M I, Kuhn L A and Thorpe M F 2004 *J. Comput. Chem.* **25** 1133–48
[23] Jacobs D J and Thorpe M F 1995 *Phys. Rev. Lett.* **75** 4051–4
[24] Jacobs D J and Thorpe M F 1996 *Phys. Rev.* E **53** 3682–93
[25] Jacobs D J, Rader A J, Kuhn L A and Thorpe M F 2001 *Proteins* **44** 150–65

[26] Hespenheide B M, Jacobs D J and Thorpe M F 2004 *J. Phys.: Condens. Matter* **16** S5055–64
[27] Gummere F B 1910 *Beowulf Harvard Classics* (New York: P F Collier & Sons) vol 49; chapter 28
[28] Rader A J, Hespenheide B M, Kuhn L A and Thorpe M F 2002 *Proc. Natl Acad. Sci. USA* **99** 3540–5
[29] Hespenheide B M, Rader A J, Thorpe M F and Kuhn L A 2002 *J. Mol. Graph Model* **21** 195–207
[30] Martin C, Richard V, Salem M, Hartley R and Mauguen Y 1999 *Acta Crystallogr.* D **55** 386–98
[31] Wells S, Dove M and Tucker M 2004 *J. Appl. Crystallogr.* **37** 536–44
[32] Sartbaeva A, Wells S A and Redfern S A T 2004 *J. Phys.: Condens. Matter* **16** 8173–89
[33] Ramakrishnan C and Ramachandran G N 1965 *Biophys. J.* **5** 909–33
[34] Hartley R W 1989 *Trends Biochem. Sci.* **14** 450–4
[35] Bycroft M, Ludvigsen S, Fersht A R and Poulsen F M 1991 *Biochemistry* **30** 8697–701
[36] Sawaya M R and Kraut J 1997 *Biochemistry* **36** 586–603
[37] Reyes V M, Sawaya M R, Brown K A and Kraut J 1995 *Biochemistry* **34** 2710–23
[38] DeLano W L 2005 The PyMol Molecular Graphics System http://www.pymol.org
[39] Jones G, Willett P, Glen R C, Leach A R and Taylor R 1997 *J. Mol. Biol.* **267** 727–48
[40] Rarey M, Kramer B, Lengauer T and Klebe G 1996 *J. Mol. Biol.* **261** 470–89
[41] Goodsell D S and Olson A J 1990 *Proteins* **8** 195–202