Hierarchical and multi-resolution representation of protein flexibility

Yong Zhao\(^1\), Daniel Stoffler\(^2\) and Michel Sanner\(^1,\ast\)

\(^1\)Department of Molecular Biology, TPC26, The Scripps Research Institute, La Jolla, CA, USA and
\(^2\)F. Hoffmann-La Roche Ltd, Pharmaceuticals Division, CH-4070 Basel, Switzerland

ABSTRACT
Motivation: Conformational rearrangements during molecular interactions are observed in a wide range of biological systems. However, computational methods that aim at simulating and predicting molecular interactions are still largely ignoring the flexible nature of biological macromolecules as the number of degrees of freedom is computationally intractable when using brute force representations.

Results: In this article, we present a computational data structure called the Flexibility Tree (FT) that enables a multi-resolution and hierarchical encoding of molecular flexibility. This tree-like data structure allows the encoding of relatively small, yet complex sub-spaces of a protein’s conformational space. These conformational sub-spaces are parameterized by a small number of variables and can be searched efficiently using standard global search techniques. The FT structure makes it straightforward to combine and nest a wide variety of motion types such as hinge, shear, twist, screw, rotameric side chains, normal modes and essential dynamics. Moreover, the ability to assign shapes to the nodes in a FT allows the interactive manipulation of flexible protein shapes and the interactive visualization of the impact of conformational changes on the protein’s overall shape. We describe the design of the FT and illustrate the construction of such trees to hierarchically combine motion information obtained from a variety of sources ranging from experiment to user intuition, and describing conformational changes at different biological scales. We show that the combination of various types of motion helps refine the encoded conformational sub-spaces to include experimentally determined structures, and we demonstrate searching these sub-spaces for specific conformations.

Contact: sanner@scripps.edu

Supplementary information: Supplementary Data are available at Bioinformatics online.

1 INTRODUCTION

Molecular flexibility is an intrinsic property of biomolecules. It is crucial for such functions as enzymatic catalysis (Bennett and Steitz, 1978; Remington et al., 1982), regulation of protein activity (Perutz, 1970, 1989), transport of metabolites (Anderson et al., 1990; Spurlino et al., 1991), etc. Highly flexible proteins have been either identified or implicated in diseases such as AIDS (HIV gp41) and scrapie (Chan et al., 1997). Understanding the fundamental nature of biomolecular flexibility is important not only for relating structural information to biological function, but also for the development of novel therapeutics (Teague, 2003).

Many studies on protein flexibility have been carried out, both by experimental and computational approaches. Evidence of molecular flexibility can be observed in experimental data such as atomic positional occupancies and multiple conformations in X-ray crystallography; conformational changes between bound and unbound structures in allosteric proteins; and in ensembles of conformations from nuclear magnetic resonance (NMR) experiments. It is important to bear in mind that observed conformational differences may be due to intrinsic motion, disorder and/or to extrinsic experimental uncertainty.

Simulation of the dynamic behavior of biological molecules such as molecular dynamics (MD) and Monte Carlo methods provides a very detailed, sometimes overwhelming, representation of macromolecular flexibility. The trajectories and conformations reflect the combination of a large variety of motions occurring at different time scales and amplitudes. Moreover, domain motions usually occur at time scales that are beyond today’s typical MD simulation capabilities (Hansson et al., 2002). Various methods for deriving and predicting motion information, i.e. mathematical transformations describing conformational changes, have been developed. These methods fall roughly into two categories: the methods predicting collective motions (i.e. most atoms can move relative to each other) and the methods describing the motion of rigid domains in a macromolecule.

Normal mode analysis (NMA), Gaussian network model (GNM) and essential dynamics (ED) belong to the first category and are briefly presented here. The normal modes are essentially vectors in mass-weighted Cartesian coordinates, calculated from the second derivative matrix of the potential energy at a local minimum. Previous studies have shown that the low-frequency normal modes of proteins, usually with frequencies <30 cm\(^{-1}\), can represent many observed atomic displacements (Brooks and Karplus, 1985; Hayward et al., 1997; Hinsen et al., 1999; Tama et al., 2000). The GNM is a statistical mechanical theory originally developed by Flory and co-workers (Flory, 1976) for polymer networks. Protein structures can be modeled by GNM as elastic networks (Tirion, 1996), i.e. Ca atoms are used to represent amino acids and uniform springs connect the Ca pairs within a distance cutoff. Vibrational vectors and frequencies can then be derived from the connectivity matrix of inter-residue contacts to describe the protein flexibility. In ED, a covariance matrix is built from atomic
coordinates. Principle component analysis (PCA) of this covariance matrix provides the squared magnitude (eigenvalues) and principle components (eigenvectors). Top principle components often correspond to the most significant conformational changes. A three-dimensional (3D) conformation of a protein can be constructed by combining the eigenvectors. ED was used to generate multiple feasible conformations in the context of the protein–peptide docking (Mustard and Ritchie, 2005).

The second group of methods partitions a macromolecule into domains that move relative to each other as rigid bodies. Several software packages have been developed to partition macromolecules into domains and extract motion information automatically, such as Domain Parser (Xu et al., 2000), DynDom (Hayward and Lee, 2002), HingeFinder (Wriggers and Schulten, 1997), Domain Finder from the MMTK package (Hinsen, 2000) and Protein Domain Parser (Alexandrov and Shindyalov, 2003). The latter method predicts a domain and motion information from a single structure, whereas other programs require two protein conformers.

Motion information can also be obtained from databases. The database of macromolecular movements (Gerstein and Krebs, 1998) classifies the proteins according to the type of motion (hinge, shear, twist and other) that describe the conformational changes observed in crystallographic structures. This database focuses only on the observed dominant motion, whereas finer level motions, such as side chain rotamer flipping, are below the level of detail represented in this database.

Various studies on macromolecular flexibility have been carried out, with motion information derived at different resolutions of flexibility, including collective backbone motions, domain motions and side-chain flexibility. Hinsen et al. employed NMA to identify the flexible domains in proteins, as well as the low-frequency domain motion information (Hinsen, 1999; Hinsen et al., 1999). Tama et al. (2000) use the rotation-translations blocks method to divide the protein into residue blocks and construct low-frequency normal modes for large macromolecules by linearly combining the rotation and translation of these blocks. Krebs et al. (2002) applied the NMA in a database framework to determine the linear combination of modes that best approximated the direction of an observed motion. Leach (1994) docked ligands into flexible proteins with discrete side chains.

The methods described above focus on a fixed resolution of flexibility, i.e. the motion is defined at the level of individual atoms, amino acid side chains (rotamers) or rigid domains. These representations of the macromolecular flexibility are often hardwired in custom applications; hence, evaluating new motion descriptors or testing their combination and nesting would require access to the source code and advanced programming. Moreover, such extensions to a specific program restrict the new model of protein flexibility to that particular application. The limitation of current approaches in representing macromolecular flexibility for protein–peptide docking have been described in a recent survey (Bonvin, 2006) which emphasized the need for combining exiting approaches.

Here we present a novel data structure called the Flexibility Tree (FT). We are not proposing an alternative to existing methods for partitioning a molecule into rigid domains or computing motion descriptors for these domains, but rather a general platform that enables the easy combination and integration of descriptors of macromolecular flexibility and the nesting of motions occurring at different scales. We describe the construction of FTs for three proteins, using various sources of information and demonstrate that these FTs cover conformational spaces containing known experimental structures and can be searched efficiently. Finally, we discuss the use of FTs for a range of applications.

2 METHODS

2.1 Design of the FT data structure

A FT is a tree-like data structure (Fig. 1) composed of tree node objects (FT nodes), simply referred to as nodes. Each node represents a set of atoms moving as a rigid body at a certain level of approximation. We refer to this set of atoms as the molecular fragment represented by the FT node. A fragment can be as large as a macromolecule or a domain, or as small as a side chain or even a single atom. The root node represents a macromolecule or macromolecular assembly. It is recursively divided into children nodes representing molecular fragments moving relative to each other. Each node can be assigned its own motion information. As the macromolecule is recursively partitioned into fragments (e.g. domains, secondary structures and side chains), more detailed motion information can be specified. For instance, a possible FT for HIV protease is depicted in Figure 1A. We partitioned the homodimer into two monomers at the first level of the tree, thus allowing the specification of a motion for one chain relative to the other. The ‘flap’ loops move relative to the ‘core’ region. Hence, each chain is further partitioned into a core and flap domain allowing the specification of the motion of the flaps relative to their respective core domains. More subtle flexibility can be encoded. For instance, the Arg8 side chain in each chain may adopt alternative conformations from a rotamer library. This illustrates the ability to selectively add fine grained motion details at user-specified locations.

Motion information is specified using an extensible set of motion objects. We defined two types of motion objects: inter-node motions and intra-node motions. Inter-node motions move the molecular fragment as a rigid body, whereas intra-node motions deform the molecular fragment corresponding to a node. Examples of inter-node motion include ‘hinge’ defined by a rotation axis and an angle, ‘shear’ with a direction and an amplitude, and general 4 × 4 transformation matrices. A second set of motion objects defines intra-node motions. Such objects may be used to describe spatial positions for the atoms within a molecular fragment. These positions can be obtained from sources such as rotamer libraries, MD, X-ray crystallography, NMR, ED and NMA. Currently available motion descriptors along with their variables and parameters are listed in the Supplementary Material.
Complex motions can be created by combining motion objects. A combined motion is essentially a list of motion objects applied in sequence. For example, the six degrees of freedom for position and orientation can be represented by combining a point rotation with a translation motion. Another example is the combination of ‘restricted rotation’, and ‘box translation’ to create a ‘local perturbation’ motion, as illustrated in Figure 2. Combining a local perturbation motion with a hinge for instance allows the inclusion of the neighborhood of the hinge in the sub-space encoded by the FT.

The motion of a child node is expressed in a coordinate system provided by a reference node. For instance, side chain atoms may move relative to the backbone, whereas amino acids in a flexible loop might move relative to (i.e. be constrained by) the position of the previous amino acid in the chain. If an intra-node motion is applied to the reference node, each child node must specify anchor points used to define the reference coordinate system in which motions are expressed. Each FT node maintains its own transformation matrix, which is calculated from the motion parameters and variables. The local transformation matrix in a node is the product of the reference matrix and its own transformation matrix. This local transformation matrix can, in turn, be used as a reference matrix for other nodes in the tree. When motion variables are modified, local transformation matrices are recursively rebuilt and applied to the atomic coordinates of the molecular fragments, thus leading to a new conformation. Motion objects can be randomized (i.e. motion variables randomly adopt new values within the allowed ranges) to generate random transformations. A FT corresponds to randomizing each motion object in the tree and results in a new conformation of the molecule.

A shape can be associated with a FT node in order to visualize the corresponding molecular fragment. It can be as simple as lines, sticks and balls, spheres, ellipsoids, boxes, or as complex as a convex hull or a molecular surface (Sanner et al., 1996).

Similarly to motions, shapes are nested and combined recursively in the tree. When the shape of a child node is evaluated in the context of its parent, the child’s position-independent shape can be modulated using the motion information. This process is referred to as the ‘shape-motion convolution’ (Fig. 3). The motion-modulated shapes from all the children nodes are then combined to produce the parent’s shape. The overall shape of the molecule is built by repeating this process recursively (shown in Fig. 3, arrows). The user can control this process by assigning various convolution and combination operators to the FT nodes. This shape nesting and combination capability allows for multi-resolution representation of flexible molecular shapes. Modifications to motion variables directly impact the shape.

2.2 Implementation

The FlexTree package is implemented in the Python programming language (www.python.org) as an extensible set of objects in the object-oriented programming sense. It is designed to facilitate the combination of motions and the addition of new motion descriptors. The component-based, modular architecture allows the incorporation of FT in a wide variety of applications. We have integrated the FlexTree package using our molecular viewer PMV (Sanner, 2005) and the visual programming environment Vision (Sanner et al., 2002). A FT can be constructed interactively using Vision (Fig. 4C) by dragging and dropping Vision-nodes (i.e. rectangular boxes in Fig. 4C) corresponding to FT nodes and connecting them using the mouse. Likewise, attributes of each FT node (e.g. molecular fragment, motion and shape) exist as Vision-nodes and can be created and assigned to a particular FT node using the mouse.

3 EXPERIMENTS

In this section, we demonstrate how to encode a small but specific conformational sub-space. We present three examples based on flexibility information obtained from various sources. In the first example, DynDom (Hayward and Lee, 2002) takes two adenylate kinase (AK) conformations (open, closed) to derive domain hinge motions. We built a FT to move one domain toward the closed conformer. Its trivial to add a local perturbation to move the rigid body even closer to the known conformer. In the second example, ProFlex needs only one conformation to predict the flexibility of P38. We combined ED with flexible side chains to describe the collective motion near the sb5 binding pocket. By modifying motion variables, our FT covers the sb5-bound conformation and the 14e-bound conformation. The last example demonstrates the encoding of induced fit for protein kinase A (PKA). Previous study has shown that a balanol cannot dock into the adenosine pocket (Cavasotto and Abagyan, 2004). Combining hinge and...
RMSD to the target conformation. Only a hinge motion applied to ATP-lid domain. (DynDom, the RMSD can be reduced to 1.69 Å. By adding a local perturbation before the hinge motion is applied. (RMSD = 1.69 Å) it can further be reduced to 0.65 Å which is close to the best rigid body fit identified by DynDom, we built a FT allowing the ATP-lid domain (residues 117–159) in open and closed conformation is 15.28 Å.

The open (orange) and closed (blue) conformations of adenylate kinase are superimposed using backbone atoms of the core domains (residues 3–29, 64–116 and 160–212). The ATP-lid domains (117–159) are depicted as ribbons whereas the Ca traces of the core and the AMP-binding domains (30–63) are shown as tubes. The hinge axis identified by DynDom is depicted as a red arrow. (A) The RMSD of backbone atoms in the ATP-lid domain is 15.28 Å before the hinge motion is applied. (B) Using the hinge predicted by DynDom, the RMSD can be reduced to 1.69 Å. By adding a local perturbation it can further be reduced to 0.65 Å which is close to the best rigid body fit (RMSD = 0.56 Å). (C) Screen shot of a FT in the Vision-based GUI. Here only a hinge motion applied to ATP-lid domain. (D) A local perturbation is combined to the hinge motion to explore the neighborhood and reduce the RMSD to the target conformation.

Local perturbation, together with flexible side chains, we started with adenosine-bound structure and approached the balanol-bound structure. We successfully docked balanol to this ‘deformed’ PKA conformation, demonstrating that the FT encodes the induced conformational changes that is critical for adenosine/balanol binding.

A simple FT
Kinases regulate many aspects that control cell growth, movement and death. Disregulated kinase activity is a frequent cause of disease, particularly cancer. AK catalyzes the reaction ATP + AMP → 2ADP. Hayward used the DynDom package to partition the AK into three domains and derive two hinges describing the conformational changes that is critical for adenosine/balanol binding.

3.1 A simple FT
Kinases regulate many aspects that control cell growth, movement and death. Disregulated kinase activity is a frequent cause of disease, particularly cancer. AK catalyzes the reaction ATP + AMP → 2ADP. Hayward used the DynDom package to partition the AK into three domains and derive two hinges describing the conformational changes between an open conformation (4AKE: A) and a ligand-induced closed conformation (1ANK: B) (Hayward, 2004). Here, we followed the same protocol but used an alternative closed conformation (2ECK: B). DynDom partitions the kinase into three rigid domains. Figure 4A shows the open and closed conformation superimposed using the backbone atoms of the core domains identified by DynDom. The root mean square deviation (RMSD) between the backbone atoms of the ATP-lid domains (residues 117–159) in open and closed conformation is 15.28 Å.

Using the open conformation, the hinge and domain definitions identified by DynDom, we built a FT allowing the ATP-lid domain to move relative to the core domain. The best achievable RMSD between the closed conformation and the conformations in this tree is 1.69 Å at a hinge angle of 52.6°. Yet, the rigid body fit of the backbone atoms of the ATP-lid domains in the open and closed conformation has a RMSD of 0.56 Å, suggesting that the hinge alone is a coarse approximation of the conformational change. It takes only a couple of mouse clicks in the Vision-based graphical user interface (GUI) (Fig. 4C) to combine a local perturbation motion with the hinge motion (Fig. 4D). The best RMSD found after 1000 random sampling of this new FT was 0.65 Å (Fig. 4B). This illustrates the ease with which FTs can be altered, as well as the ability of local perturbation to explore the neighborhood of narrow conformational path.

3.2 A FT for MAPK P38 using a single structure
14e (Stelmach et al., 2003) and sb5 (Wang et al., 1998) are known ligands of the mitogen-activated protein kinase (MAPK) P38. MAPKs are serine/threonine-specific protein kinases that respond to extracellular stimuli (mitogens) and regulate various cellular activities, such as gene expression, mitosis, differentiation and cell survival/apoptosis. We built a FT using the sb5-bound conformation (1BMK) and encoded both backbone and side chain motion obtained from analyzing that single structure. Then we verified that FT can reach the 14e-bound conformation. Note that in this example the knowledge of the 14e-bound conformation was not used to define the flexibility of P38.

First, we removed the sb5 ligand from the crystal structure (1BMK) and added polar hydrogens using AutoDockTools (Sanner, 2005). Then the ‘flexibility and rigidity analysis’ from ProFlex (Jacobs et al., 2001) was carried out. We excluded all hydrogen bonds with energy > −0.1 kcal/mol. Flexible backbone atoms predicted by ProFlex are marked black (Fig. 5A). The flexible atoms near the sb5-binding site cluster into three loops: Gly31-Ser37, Thr106-Asp112 and Leu167-His174. We used the CONCOORD package (de Groot et al., 1997) to generate alternative conformations for each of these loops and performed a PCA of the Ca atoms conformations. We ignored Leu167-His174 loop because no significantly different conformation was generated by CONCOORD for this loop. We used the top five principle components to describe the collective motions of amino acids in loops A and B and made the side chains in these loops flexible. The corresponding FT data structure of P38 is shown in Figure 5B. A genetic algorithm was then used to optimize the motion variables (i.e. individual weights of the five principal components deforming the loop backbones and the rotamers) in order to approximate the 14e-bound conformation. The best result has a RMSD of 1.14 Å with the target structure, which is better than the rigid superimposition of the loops, RMSD = 1.29 Å.

3.3 Induced fit in balanol/adenosine-bound pocket
cAMP-dependent protein kinase (cAPK), also known as PKA, regulates in concert with other kinases the majority of cellular pathways, especially those involved in signal transduction. Crystal structures of adenosine (1BKX and 1FM0) and balanol (1BX6) bound to cAPK are available. After superimposing the backbone atoms of the adenosine-bound structures (1BKX) with the balanol-bound structure (1BX6), a hinge-like displacement of the Gly-rich flap became evident. This is in agreement with the observation by Cavasotto and Abagyan (2004) who showed that balanol cannot
Fig. 5. (A) The flexibility analysis of backbone atoms near the sb5-binding site (predicted by ProFlex). Rigid atoms are depicted in gray whereas the flexible ones are colored in black. The native binding ligand sb5 is shown as sticks and balls. (B) The FT data structure for representing P38 flexibility: two loops are modeled with essential dynamics (E.D.). The side chains on the loops near the sb5 docking pocket are made flexible (except for glycines and alanines). These flexible side chains are modeled by rotamers (R).

Fig. 6. (A) The superimposition of balanol-bound structure (1BX6, black) and adenosine-bound conformation (1FMO, gray) of cAPK reveals a hinge motion of the Gly-rich flap which is crucial for balanol/adenosine binding. The side chains made flexible (Phe54 and Arg56) are shown as tubes. (B) FT encoding the hinge motion of the Gly-rich flap and rotameric side chains for Phe54 and Arg56. The motion variables $v_1$, $v_2$ and $v_3$ parameterize the conformational sub-space encoded by the tree and can be randomized to sample the sub-space or optimized for a given target conformation.

dock in the rigid adenosine pocket due to clashes with the Gly-rich flap (Gly50-Val57). We built a FT using the 1FMO bound structure of adenosine and encoded a hinge motion using a hinge axis defined by the two nitrogen atoms from Gly50 and Val57, as revealed by visual inspection (Fig. 6A). In order to eliminate the clashes between the moving flap and the rest of the kinase, we allowed two residues on the flap, Arg56 and Phe54, to adopt rotameric positions resulting in the FT data structure depicted in Figure 6B. This tree encodes several combined motions: the flap moves relative to the fixed region, whereas the two side chains move relative to the moving Gly-rich flap.

When the backbone atoms of 1FMO and 1BX6 are superimposed, the all-atoms RMSD between the flap atoms is 2.77 Å. This deviation has been reported to be sufficient to prevent cross-docking (Cavasotto and Abagyan, 2004). The rigid superimposition of the 1FMO flap over the same flap on 1BX6 yields a RMSD of 1.43 Å. The variables of the motion objects in the tree (i.e. hinge angle and rotamer index for movable side chains) were optimized by a genetic algorithm to reproduce the balanol-bound conformation (1BX6) leading to a RMSD of 1.98 Å which is better than the initial 2.77 Å. Adding a local perturbation motion allows this value to be reduced to 1.22 Å which is better than the rigid superimposition due to the rearrangements of flexible side chains. AutoDock (Morris et al., 1999) docks balanol to this cAPK conformation (energy = −12.02 kcal/mol, RMSD = 0.85 Å). This FT encodes the conformations that are crucial for balanol/adenosine binding.

4 DISCUSSIONS AND CONCLUSIONS

In this article, we describe the FT data structure allowing the hierarchical and multi-resolution representation of conformational changes in macromolecules. This approach is based on the hypothesis that protein flexibility can be represented by a hierarchical decomposition of molecular fragments moving relative to each other. Conformational changes that cannot be decomposed hierarchically would lead to a single-level tree. This is the worst case scenario in which this representation does not provide any particular advantage. However, side chain motion is invariably involved in protein-binding processes, and it has been suggested that hinge bending motion between domains is likely to universally exist (Ma et al., 2002). A large number of documented cases show domain motion between bound and unbound states (Gerstein and Krebs, 1998). Moreover, biological macromolecules themselves are organized in a hierarchical manner (from molecule to domains, chains, residues and atoms). This idea is also consistent with conclusions drawn by Teague in his recent review (Ma et al., 2002; Teague, 2003) where he points out that ‘Proteins can be considered as being composed of CHUs (compact hydrophobic units) connected by loops. The CHUs act as relatively rigid bodies that move with respect to each other.’ Hence, we are confident that the FT will be useful for encoding important conformational sub-spaces for a wide range of biological macromolecules.

Single-level models of protein flexibility (e.g. only domain motion) are over-simplified. In a recent review on protein–protein docking (Bonvin, 2006), Bonvin summarized the state-of-the-art approaches in implicit and explicit treatment of flexibility in docking. For large conformational changes involving backbone atoms, the author stated that ‘Most probably, there will not be a unique solution; rather, it will be the proper combination of approaches for representing conformation changes and flexibility at several levels that will lead to success.’ This is precisely what the FT enables: combining the motions at various resolutions to encode a specific conformational sub-space. We believe that the ability to mix large conformational changes and local rearrangements, will be important for modeling molecular interactions. Moreover, the ability to selectively add atomic-level motions such as rotameric side chains, in selected locations helps keep the computational complexity under control. User can focus on large-scale motion first, combine more small scale motions when necessary. The relevant flexibility can then be represented with limited number of variables.

Detailed motion information might not be necessary everywhere in the protein. We defined various inter-node motions to describe the relative motion of rigid bodies and various intra-node motions defining conformational changes of a molecular fragment. These motion descriptors include analytical descriptions (hinges, rotations, translations, screws, normal modes, ED, etc.) and discrete descriptions (rotamers, discrete sets of coordinates, etc.), and we have demonstrated their seamless and easy combination. New motion descriptors can readily be added to the existing set and there is no limit as to how these descriptors can be combined. For example, a particular loop in a protein can be modeled with ED or normal modes whereas another loop in the same protein can be described using discrete conformers. These loops can be
inside domain moving relative to each other, and they can contain user-selected flexible side chains. This versatility is an important characteristic which enables the targeted incorporation of flexibility information available from a wide variety of sources.

The tree-like structure of the FT leads naturally to the multi-resolution encoding of flexibility, allowing the representation of large-scale motions at the top of the tree and more subtle local rearrangements further down the tree for user-defined fragments of the macromolecule. It provides a mechanism for the selective encoding of specific conformational sub-spaces using computationally tractable parameterization. We have shown several examples of conformational space encoding. The first example shows the ease of combining local perturbation motions to cover more conformational space. In the P38 case, the flexibility information were derived from the ProFlex package, using single structure, whereas in the cAPK example, the hinge motion information was based on user intuition, which is sufficient to enable previously failed cross-docking.

This data structure should be useful for a variety of applications. We are currently developing a ligand–receptor docking program that uses the FT for representing protein flexibility. We are also exploring the use of FTs for docking flexible proteins into cryo-EM density maps, using a cross-correlation function as scoring function. Another potential application of FT could be the orchestration of individual proteins in large complexes such as molecular motors, viral capsids or even cells. The component-based software design of the macromolecule. It provides a mechanism for the selective incorporation of flexibility which enables the targeted incorporation of flexibility information available from a wide variety of sources.

Another potential application of FT could be the orchestration of individual proteins in large complexes such as molecular motors, viral capsids or even cells. The component-based software design of the macromolecule. It provides a mechanism for the selective incorporation of flexibility which enables the targeted incorporation of flexibility information available from a wide variety of sources.

The source code of the FlexTree package is freely available for academic usage. The FlexTree package can be found at http://www.scripps.edu/~sanner/FlexTree. Moreover, the programs PMV and Vision providing GUI for building and manipulating FTs are freely available (http://www.scripps.edu/~sanner/software) in source form. A movie clip showing the conformational sampling of HIV protease is included in the Supplementary Material. The movie frames were generated using PMV, Vision and FlexTree.

ACKNOWLEDGEMENTS

We thank Arthur Olson for fruitful discussions and the financial support from NIH Grant BISTI, GM65609 (2003–2007) and NIH RR08605 (2004–2009) to Michel Sanner. This is manuscript 18089-MB from the Scripps Research Institute.

Conflict of Interest: none declared.

REFERENCES


