Elastic models of conformational transitions in macromolecules

Moon K. Kim [a], Gregory S. Chirikjian [a,*], Robert L. Jernigan [b,1]

[a] *Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA*
[b] *Laboratory of Experimental and Computational Biology, Molecular Structure Section, CCR, NCI, NIH, Bethesda, MD 20892-5677, USA*

Abstract

We develop a computationally efficient and physically realistic method to simulate the transition of a macromolecule between two conformations. Our method is based on a coarse-grained elastic network model in which contact interactions between spatially proximal parts of the macromolecule are modelled with Gaussian/harmonic potentials. To delimit the interactions in such models, we introduce a cutoff to the permitted number of nearest neighbors. This generates stiffness (Hessian) matrices that are both sparse and quite uniform, hence, allowing for efficient computations. Several toy models are tested using our method to mimic simple classes of macromolecular motions such as stretching, hinge bending, shear, compression, ligand binding and nucleic acid structural transitions. Simulation results demonstrate that the method developed here reliably generates sequences of feasible intermediate conformations of macromolecules, since our method observes steric constraints and produces monotonic changes to virtual bond angles and torsion angles. A final application is made to the opening process of the protein lactoferrin.
© 2002 Elsevier Science Inc. All rights reserved.

Keywords: Conformational transitions; Intermediate structure; Elastic network model; Animation; Macromolecules

1. Introduction

As the number of solved structures of macromolecules is rapidly increasing, we frequently need to relate such structures to one another. Many proteins have multiple conformations (some are called "open" and "closed" forms) [1]. Conformational transitions between two forms are often important to understand the relationship between structure and function. In other words, some motions are requisite to the way in which a structure performs a particular function such as catalysis, regulation, transport, and binding of ligands [2]. Hence, comprehending conformational transitions can be useful for understanding biological mechanisms; however, directly computing the transition pathways has proven difficult. This problem of elucidating transition pathway can be viewed as a more limited problem than the protein folding problem.

On the other hand, it is also popular in molecular graphics to attempt to visualize conformational transitions. Obviously, one of the best ways is through animations such as digital movies (e.g. AVI or MPEG). Animations usually are produced by inserting images of intermediate conformations between the two end conformations at the extrema.

Since these two are usually the only known conformations, the hypothetical intermediate conformations are visualized in sequence to produce an animation, but the realism of these produced forms can be a problem.

In a number of recent papers in this area, Vonrhein et al. showed movies of macromolecular motions by a linear interpolation between the atomic coordinates of the two end conformations in Cartesian space [3]. The severe deficiency of this method is that the bond lengths and angles of the intermediate conformations can be unrealistic and in several cases protein chains even pass through one another. To get around this problem, Gerstein and Krebs applied proper restraints and minimized energy of each intermediate conformation to correct for molecular stereochemistry and to enforce rules of molecular structure [4,5].

An alternative interpolation approach that has been taken is to use internal coordinates such as bond lengths, bond angles, and torsion (dihedral) angles instead. Kleywegt and Jones implemented this approach to construct intermediate conformations with the LSQMAN program [6,7]. Ideally, this approach would produce realistic bond lengths and torsion angles; however, this method also has some problems. Even if we construct intermediate conformations by interpolating torsion angles between those of the two end conformations, while holding bond lengths and angles fixed, we will often get impossible pathways for several reasons. First, with these constraints it may not be possible for the solved conformation from one end to reach the other end
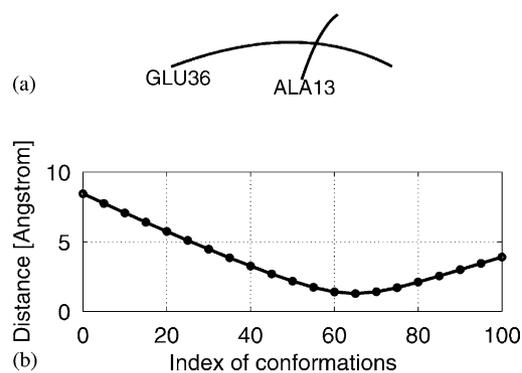
Fig. 1. An example of torsional angle interpolation between two lac repressor headpiece structures (named 1LCC and 1LCD, Protein Data Bank). During the conformational transition from 1LCC to 1LCD, the α-carbon of ALA13 and the α-carbon of GLU36 come too close together, ≤1 Å, in (a). This unrealistic relative distance between two atoms is shown in (b).

Fig. 2. Representation of protein structure as an elastic network. The backbone trace is shown in dark lines. The grey lines represent the spring connections between α-carbons within a cutoff distance of 8 Å for lac repressor headpiece structure (1LCC).

in Cartesian space because X-ray crystallography data for the two end conformations do not always have exactly the same values of internal variables such as bond lengths and bond angles. Therefore, we must either refine the two end conformations so as to have consistent values of internal variables, except for torsion angles, before interpolating in torsion angle space or permit the interpolation of all internal variables simultaneously in order to avoid this problem. A further limitation is that in the process of generating intermediate conformations some parts of the molecule could come too close to other parts in order to achieve a smooth simulated pathway, which would produce highly unfavorable states in the sense of high energy interactions or steric clashes. Fig. 1 shows a case where a particular pair of α-carbons in a protein come too close to each other during conformational transitions using internal coordinate interpolation so that it gives rise to high repulsive energy peaks simply because of Van der Waals repulsions between non-bonded atoms. A third problem occurs for the specific rotation angles. Individual values are not all equi-energetic. Consequently some forms take on higher energies, and the intermediate forms generated could have inordinately high values, even when lower energy alternative pathways exist.

There is a further more complex issue—in some cases a large transition may require complete denaturation of part of the structure. This complexity is beyond the scope of the present paper. However, it is conceivable that the present model computations could indicate where the denaturation is required, if intermediates lose sufficient numbers of interacting pairs. If no good pathway is achievable with the present approach, it would indicate a more complex pathway, such as one having large distortions rather than a relatively smooth pathway. These large distortions could of course include denaturing parts of the chain.

A series of recent papers have demonstrated the usefulness of coarse-grained models including only α-carbons as point masses representing residues and a simplified harmonic

potential for considering internal interactions between neighboring residues as shown in Fig. 2. Such models are suitable to describe the global motions of complex systems of small proteins or single proteins having more than several thousand residues [8–10].

In this paper, we generate feasible pathways for conformational transitions using the simplest potential and coarse-grained models. The key idea is to interpolate evenly the distance between spatially proximal parts of the macromolecule in both conformations within the context of the elastic network model. Since we interpolate relative distance between spatially close residues, unrealistic conformations and steric clashes become less likely. This method is a reasonable compromise between oversimplified linear interpolation in Cartesian or internal coordinates and computationally expensive methods such as MD simulations and represents another significant example of the utility of elastic model to treat protein conformations. We address several fitting strategies to visualize the simulated pathways smoothly. Several toy models are presented using the method developed here, and the conformational transition of lactoferrin is also simulated.

2. Method

2.1. System modelling

In this section, we derive a discrete mechanical model of small conformational changes for macromolecules about an equilibrium conformation. In the discussion that follows, we formulate the method in the context of protein structure. Extensions to nucleic acids or other biomolecular structures are obvious, and are also illustrated with toy models later in the paper.

We label the mass of the ith residue in the protein chain as m_i, and model the interaction between residues i and j with a linear spring having stiffness $k_{i,j}$. Given the full set

of masses, stiffnesses and equilibrium positions we derive the global mass matrix and the global stiffness matrix.

2.1.1. Standardizing coordinates

Once structures of macromolecules have been obtained from the Protein Data Bank (http://www.rcsb.org/pdb/) [1], the first step is to define a set of n representative atoms (e.g. α-carbons in the case of proteins). The position of the ith atom at time t is denoted

$$\boldsymbol{x}_i(t) = [x_i(t), y_i(t), z_i(t)]^{\mathrm{T}} \in \mathbb{R}^3. \tag{1}$$

This is defined relative to a reference frame fixed in space at the center of mass of all the representative atoms and oriented along the principal moments of inertia. That is, the positions used for this calculation are those for the static (undeformed) model data. The center of mass coordinates are given by

$$\boldsymbol{x}_{\mathrm{cm}}(0) = \frac{\sum_{k=1}^{n} m_k \boldsymbol{x}_k(0)}{\sum_{k=1}^{n} m_k} \tag{2}$$

and the moment of inertia tensor is

$$I(0) = \sum_{k=1}^{n} m_k [(\boldsymbol{x}_k^{\mathrm{T}}(0)\boldsymbol{x}_k(0))\mathbb{E}_3 - \boldsymbol{x}_k(0)\boldsymbol{x}_k^{\mathrm{T}}(0)], \tag{3}$$

where $\mathbb{E}_3$ is the 3×3 identity matrix. We assume that

$$\boldsymbol{x}_{\mathrm{cm}}(0) = \boldsymbol{0} \tag{4}$$

and

$$I(0) = \mathrm{diag}[I_1, I_2, I_3] = \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix}. \tag{5}$$

Given another initial set of points $\{\boldsymbol{x}_k'(0)\}$ that do not satisfy these properties, we can define

$$\boldsymbol{x}_k(0) = R^{\mathrm{T}} \boldsymbol{\chi}_k(0), \tag{6}$$

where

$$\boldsymbol{\chi}_k(0) = \boldsymbol{x}_k'(0) - \boldsymbol{x}_{\mathrm{cm}}'(0) \tag{7}$$

and R is the rotation matrix such that

$$I(0) = R^{\mathrm{T}} I'(0) R, \tag{8}$$

where

$$I'(0) = \sum_{k=1}^{n} m_k [(\boldsymbol{\chi}_k^{\mathrm{T}}(0)\boldsymbol{\chi}_k(0))\mathbb{E}_3 - \boldsymbol{\chi}_k(0)\boldsymbol{\chi}_k^{\mathrm{T}}(0)]. \tag{9}$$

By the proper choice of R that has normalized eigenvectors of $I'(0)$ as columns, we can ensure that $I(0)$ is diagonal.

2.1.2. Elastic network model

The total kinetic energy in a network of n point masses has the form

$$T = \frac{1}{2} \sum_{i=1}^{n} m_i \|\dot{\boldsymbol{x}}_i(t)\|^2. \tag{10}$$

We define $\boldsymbol{\delta}_i(t)$ as a vector of small displacements

$$\boldsymbol{x}_i(t) = \boldsymbol{x}_i(0) + \boldsymbol{\delta}_i(t). \tag{11}$$

The global mass matrix for the whole network is the matrix M such that

$$T = \frac{1}{2} \dot{\boldsymbol{\delta}}^{\mathrm{T}} M \dot{\boldsymbol{\delta}} \tag{12}$$

where

$$\boldsymbol{\delta} = [\boldsymbol{\delta}_1^{\mathrm{T}}, \dots, \boldsymbol{\delta}_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{3n}. \tag{13}$$

In the present case, M is diagonal.

The total potential energy in a network of connected springs has the form

$$V = \frac{1}{2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} k_{i,j} \{\|\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t)\| \|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|\}^2 \tag{14}$$

and

$$k_{i,j} = \begin{cases} 1 & \text{if } \|\boldsymbol{x}_i - \boldsymbol{x}_j\| \leq d \\ 0 & \text{if } \|\boldsymbol{x}_i - \boldsymbol{x}_j\| > d \end{cases} \tag{15}$$

where d is a cutoff distance between atoms at time $t = 0$ and $k_{i,j}$ is the (i, j) element of k (called the "linking matrix" or "contact matrix"), which is assumed to be unity for all contacting pairs and zero for pairs not in contact, regardless of the residue types concerned.

This is nothing more than the sum of elastic energy for each spring stretched or compressed from its equilibrium length. Springs represent close residues, all interacting in identical ways, and the elastic potential energy follows a harmonic potential, appropriate for small deviations from equilibrium. In general, Eq. (14) is a non-linear function of the deformations even though the springs are linear. However, when we assume that the deformations are small, V becomes a classical quadratic potential energy function [15]. In order to see this, the quantity inside the summations in Eq. (14) can be written as

$$V_{i,j} = \frac{1}{2} k_{i,j} \{\|(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)) + (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))\| \\ - \|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|\}^2 \tag{16}$$

which can then be separated into two parts

$$V_{i,j} = \frac{1}{2} k_{i,j} (V_{i,j}^{(1)} + V_{i,j}^{(2)}) \tag{17}$$

where $V_{i,j}^{(1)}$ has the squared terms of the expansion and $V_{i,j}^{(2)}$ has the cross terms :

$$\begin{aligned} V_{i,j}^{(1)} &= \|(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)) + (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))\|^2 + \|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|^2 \\ &= \|\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t)\|^2 + 2\|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|^2 + 2(\boldsymbol{x}_i(0) \\ &\quad - \boldsymbol{x}_j(0))^{\mathrm{T}} (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t)) \end{aligned} \tag{18}$$

and

$$V_{i,j}^{(2)} = -2\|(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)) + (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))\|$$
$$\cdot \|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|. \tag{19}$$

Eq. (19) can be simplified for small values of $\|\boldsymbol{\delta}_i(t)\|$ and $\|\boldsymbol{\delta}_j(t)\|$ using the Taylor series approximation

$$\|\boldsymbol{x} + \boldsymbol{\delta}\| \approx \|\boldsymbol{x}\| + \frac{\boldsymbol{x} \cdot \boldsymbol{\delta}}{\|\boldsymbol{x}\|} + \frac{1}{2} \frac{\boldsymbol{\delta}^{\mathrm{T}} A(\boldsymbol{x}) \boldsymbol{\delta}}{\|\boldsymbol{x}\|} \tag{20}$$

where

$$A(\boldsymbol{x}) = \mathbb{E}_3 - \frac{\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}}{\|\boldsymbol{x}\|^2}. \tag{21}$$

Hence, we write

$$V_{i,j}^{(2)} = -2\|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|^2 - 2(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))(\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))$$
$$- (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))^{\mathrm{T}} A(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))(\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t)). \tag{22}$$

We can now write Eq. (17) as

$$V_{i,j} \approx \frac{1}{2} k_{i,j} (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))^{\mathrm{T}}$$
$$\times [\mathbb{E}_3 - A(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))](\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t)). \tag{23}$$

If we let $G_{i,j} \in \mathbb{R}^{3 \times 3}$ be defined as

$$G_{i,j} = k_{i,j}[\mathbb{E}_3 - A(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))]$$
$$= k_{i,j} \frac{(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))(\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0))^{\mathrm{T}}}{\|\boldsymbol{x}_i(0) - \boldsymbol{x}_j(0)\|^2}, \tag{24}$$

then for small deflections, the total potential energy (14) can be written in the form

$$V = \frac{1}{2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t))^{\mathrm{T}} G_{i,j} (\boldsymbol{\delta}_i(t) - \boldsymbol{\delta}_j(t)). \tag{25}$$

Note that $G_{i,j} = G_{j,i}$ and $G_{i,j}^{\mathrm{T}} = G_{i,j}$. The stiffness matrix for the whole network is the matrix K such that

$$V = \frac{1}{2} \boldsymbol{\delta}^{\mathrm{T}} K \boldsymbol{\delta}. \tag{26}$$

The matrix K consists of an $n \times n$ array of 3×3 symmetric blocks. Let $K_{i,j}$ denote the i,jth block for $i, j \in [1, \ldots, n]$. If mass i is not connected to mass j, then the i,jth block is a 3×3 zero matrix. Generally, if $i \neq j$,

$$K_{i,j} = -G_{i,j}. \tag{27}$$

When $i = j$, the result is

$$K_{i,i} = \sum_{k=1}^{i-1} G_{k,i} + \sum_{k=i+1}^{n} G_{i,k} = \sum_{k \neq i} G_{k,i}. \tag{28}$$

Finally, we can obtain the equation of motion of the protein composed of n residues as

$$M\ddot{\boldsymbol{\delta}} + K\boldsymbol{\delta} = \boldsymbol{0}. \tag{29}$$

Since we use Cartesian coordinates, the elements of M are of the form $M_{i,j} = m_i \delta_{i,j}$ (i.e. M is diagonal). If

$m_i = m$, then normal modes are the eigenvectors of K. Normal modes generated using this coarse-grained model can be used to evaluate potential motions about a single equilibrium conformation of a large molecule with relatively little computational cost.

2.2. Incremental formation of an intermediate conformation

We derive here an incremental formulation to generate intermediate conformations along a putative pathway between two forms defined as end constraints. The key idea is to interpolate between two values of the distances between spatially proximal α-carbons, which are artificially connected with virtual bonds in the elastic network model [8–10]. While the relationship between molecular conformations and the distances between atoms in conformations has been studied extensively [16], our goal is to generate intermediate conformations by finding small changes in α-carbon positions that result from inducing correspondingly small changes in inter-residue distances.

Suppose that we have atomic coordinates of the two end conformations of the same protein denoted by $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{\chi}_i\}$, respectively. One can build two elastic network models (one for each conformation). We introduce a cost function as

$$C(\boldsymbol{\delta}) = \frac{1}{2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} k_{i,j} \{\|\boldsymbol{x}_i + \boldsymbol{\delta}_i - \boldsymbol{x}_j - \boldsymbol{\delta}_j\| - l_{i,j}\}^2. \tag{30}$$

An intermediate conformation is defined by the value of $\boldsymbol{\delta}$ that minimizes this cost when all the other parameters are held constant. The linking matrix k is the "union" of the two linking matrices for $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{\chi}_i\}$ in the sense that $k_{i,j}$ has value 1 whenever residues i and j are within the cutoff range in either conformation. Alternatively, one can use the "common" linking matrix in which $k_{i,j}$ has value 1 only when residues i and j are within the cutoff range in both conformations. However, in our experience this does not produce a transition pathway that makes one end form converge to the other, because residues which are not connected to one another in the sense of the common linking matrix are not confined to move to the targeted form in this approach. In contrast, if we use the union linking matrix to generate a transition pathway, both of the two end forms are considered as constraints in order to make intermediate conformations converge to the targeted form.

The value $l_{i,j}$ is the targeted distance between i and j. $l_{i,j}$ can be chosen as

$$l_{i,j} = (1 - \alpha)\|\boldsymbol{x}_i - \boldsymbol{x}_j\| + \alpha\|\boldsymbol{\chi}_i - \boldsymbol{\chi}_j\| \tag{31}$$

where α is the coefficient that represents the extent to which the conformation has moved away from $\{\boldsymbol{x}_i\}$ towards $\{\boldsymbol{\chi}_i\}$. For example, when $\alpha = 0.5$, the desired conformation is the one with inter-residue distances at the average of conformations $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{\chi}_i\}$. Using the "union" linking matrices

confines the intermediate conformations to the interval between the two end conformations. Our goal is to find values of $\boldsymbol{\delta}$ that minimize Eq. (30). Eq. (30) can be expanded in a Taylor series for small values of $||\boldsymbol{\delta}_i(t)||$ and $||\boldsymbol{\delta}_j(t)||$ as

$$C_{i,j} = \tfrac{1}{2}k_{i,j}(C_{i,j}^{(1)} + C_{i,j}^{(2)} + C_{i,j}^{(3)}), \tag{32}$$

where

$$C_{i,j}^{(1)} = (\boldsymbol{\delta}_i - \boldsymbol{\delta}_j)^{\mathrm{T}} \left[\mathbb{E}_3 - l_{i,j} \frac{A(\boldsymbol{x}_i - \boldsymbol{x}_j)}{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|} \right] (\boldsymbol{\delta}_i - \boldsymbol{\delta}_j), \tag{33}$$

$$C_{i,j}^{(2)} = 2\left(1 - \frac{l_{i,j}}{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}\right)(\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}(\boldsymbol{\delta}_i - \boldsymbol{\delta}_j), \tag{34}$$

and

$$C_{i,j}^{(3)} = (\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}(\boldsymbol{x}_i - \boldsymbol{x}_j) - 2l_{i,j}\|\boldsymbol{x}_i - \boldsymbol{x}_j\| + l_{i,j}^2. \tag{35}$$

If we define $G'_{i,j} \in \mathbb{R}^{3\times 3}$ as

$$G'_{i,j} = k_{i,j}\left[\mathbb{E}_3 - l_{i,j}\frac{A(\boldsymbol{x}_i - \boldsymbol{x}_j)}{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}\right] \tag{36}$$

then

$$\frac{1}{2}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} k_{i,j}C_{i,j}^{(1)} = \frac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\Gamma\boldsymbol{\delta}. \tag{37}$$

Generally, if $i \neq j$,

$$\Gamma_{i,j} = -G'_{i,j}. \tag{38}$$

When $i = j$, the result is

$$\Gamma_{i,i} = \sum_{k=1}^{i-1} G'_{k,i} + \sum_{k=i+1}^{n} G'_{i,k} = \sum_{k\neq i} G'_{k,i}. \tag{39}$$

Let $\boldsymbol{v}_{i,j} \in \mathbb{R}^{1\times 3}$ be

$$\boldsymbol{v}_{i,j} = 2k_{i,j}\left(1 - \frac{l_{i,j}}{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}\right)(\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}}. \tag{40}$$

Then

$$\frac{1}{2}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} k_{i,j}C_{i,j}^{(2)} = \frac{1}{2}\boldsymbol{\gamma}\boldsymbol{\delta} \tag{41}$$

where

$$\boldsymbol{\gamma} = [\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_n] \in \mathbb{R}^{1\times 3n} \tag{42}$$

and

$$\boldsymbol{\gamma}_i = -\sum_{k=1}^{i-1}\boldsymbol{v}_{k,i} + \sum_{k=i+1}^{n}\boldsymbol{v}_{i,k} = \sum_{k\neq i}\boldsymbol{v}_{i,k}. \tag{43}$$

Let B be

$$\frac{1}{2}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} k_{i,j}C_{i,j}^{(3)} = B. \tag{44}$$

Finally, Eq. (30) is simplified as

$$C(\boldsymbol{\delta}) = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} C_{i,j} = \frac{1}{2}\boldsymbol{\delta}^{\mathrm{T}}\Gamma\boldsymbol{\delta} + \frac{1}{2}\boldsymbol{\gamma}\boldsymbol{\delta} + B. \tag{45}$$

We get the minimum of $C(\boldsymbol{\delta})$ over $\boldsymbol{\delta}$ as

$$\frac{\partial C(\boldsymbol{\delta})}{\partial \boldsymbol{\delta}} = \Gamma\boldsymbol{\delta} + \frac{1}{2}\boldsymbol{\gamma}^{\mathrm{T}} = \boldsymbol{0}. \tag{46}$$

We note that $\Gamma \in \mathbb{R}^{3n\times 3n}$ always has three zero eigenvalues corresponding to translation modes, because a translated version of $\boldsymbol{\delta}$ satisfying Eq. (46) can also minimize the cost function. That is, the solution to Eq. (46) is not unique. To solve this problem, one can either assume a particular point is fixed in space so that Γ can be reduced to a non-singular (invertible) matrix or add the constraint of linear momentum conservation so that

$$\sum_{i=1}^{n} m_i\boldsymbol{\delta}_i = \boldsymbol{0}. \tag{47}$$

Note that here we treat $m_i = 1$.

We apply this method to a simple toy model in the plane and obtain the smooth pathway from the left to the right conformation as shown in Fig. 3. During the conformational transitions, the spring on the left side disappears and the new spring appears on the right side instead. Several intermediate conformations are more flexible (unstable) than either of the two end conformations due to the loss of stiffness. In the equilibrium of a real protein system,
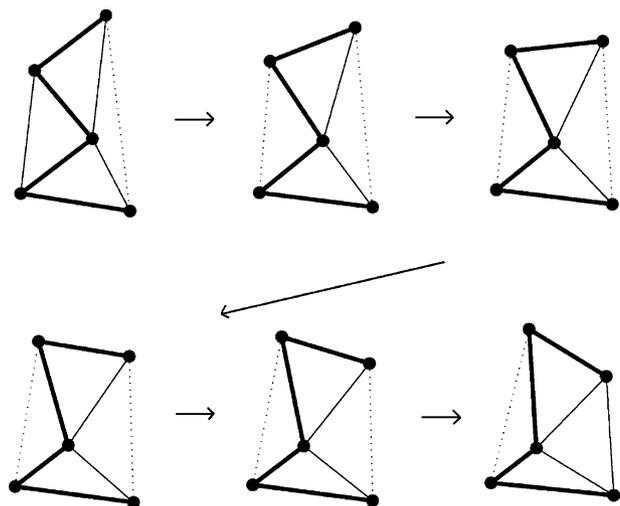
Fig. 3. Pathway illustration of a toy model. Conformations change from the top left to bottom right. Ninety-nine intermediate conformations are obtained incrementally using distance interpolation and four intermediate conformations are illustrated here. Both end conformations have seven contacts. The union linking matrix has eight. Several intermediate conformations have only six contacts, making them more flexible than either of the two end conformations. The backbone chain is illustrated as bold lines and dashed lines represent contacts that are absent. The "union" linking matrix includes all contacts present in either end-point conformation.

changes in the surroundings of a protein might trigger conformational changes from one equilibrium to another. In the present case, intermediate states can be less stable and more flexible.

2.3. Computational complexity

It is known that the dynamic behavior of the elastic network model of a protein can vary with cutoff values. Large cutoff values give rise to an increase in the number of interacting pairs and a greater cohesiveness. Consequently, systems become stiffer, the amplitudes of fluctuations decrease, and motions are usually more cooperative. Also for relatively short cutoff values, it is possible to obtain more than six zero eigenvalues corresponding to rigid-body modes in normal mode analysis, and there can be extremely large amplitude fluctuations along particular directions for particular residues [8]. Likewise, our interpolation method, which is basically derived from a matrix similar to the stiffness matrix, is sensitive to cutoff values and the geometry of a given protein structure. Short cutoff values strongly force the residues to be in contact with local neighbors only. It can sometimes cause unrealistic results that lead to discontinuous motions. To remove such behavior, one could adopt larger cutoff values. However, a denser linking matrix can tremendously increase computation time for generating intermediate transitions in large protein models composed of several thousand residues.

We introduce a new approach to assure having uniformly sparse linking matrices. The method reduces computational costs for the whole interpolation process and also guarantees realistic results. To reach this goal, the linking matrix can be created by imposing a cutoff on the number of residue contacts, instead of a cutoff distance. Namely, we can connect residues to their neighboring residues in order from the closest one, increasing the distance gradually until the fixed limiting number is reached, regardless of the actual distance of the last connection. This assures obtaining a linking matrix that is sparse and uniform because all residues will have the same number of connections.

2.4. Visualization

Animations of conformational transitions are more comprehensible than a series of static pictures and are particularly useful for teaching [17]. We incrementally generate 99 intermediate conformations between the two end point conformations using the distance interpolation method proposed in this paper. In the implementation, we calculate δ to minimize our cost function in Eq. (30) using steps of $\alpha = 0.01$. Then we get the first intermediate conformation denoted by $\{x_i^1\}$, which is between $\{x_i\}$ and $\{\chi_i\}$, and differs from $\{x_i\}$ by only 1%. That is,

$$x_i^1 = x_i + \delta_i \tag{48}$$

where x_i^1 is the ith residue of $\{x_i^1\}$. Likewise for the conformation $\{x_i^2\}$,

$$x_i^2 = x_i^1 + \delta_i \tag{49}$$

where δ is the solution of Eq. (46) when $\alpha = 0.02$. The remaining conformations can be obtained in the same way. We use several Matlab functions such as "getframe" and "movie" to build an animation from the solved conformations. Static pictures accumulate to create movie frames. Finally, these can be converted to common digital movie formats.

Our interpolation method does not account for the absolute position of individual atoms in space but the distance between interacting pairs. Hence, the Cartesian position of atoms of an intermediate conformation is not unique in space. For this reason, several movies produced by this method clearly show that the solved conformations starting from one conformation do not converge to the spatial position and orientation of other conformation, even though the shape is sequentially interpolated quite well. Several rigid-body superposition methods are useful to correct this problem, as discussed below.

2.4.1. RMS superposition

Traditional RMS superposition minimizes the RMS difference between two conformations. Consider two conformations of n residue positions in $\mathbb{R}^3$ denoted as $\{x_i\}$ and $\{\chi_i\}$ where the correspondence $x_i \leftrightarrow \chi_i$ is assumed for all $i = 1, \ldots, n$. The goal is to find the rigid-body motion (R, a) such that $\{\chi_i\}$ is moved to fit in the "best" way to $\{x_i\}$ where a is a translation vector and $R \in \mathbb{R}^{3 \times 3}$ is a rotation matrix. One way to define the best fit is to minimize the RMS error

$$E^2(R, a) = \sum_{i=1}^{n} \|x_i - (R\chi_i + a)\|^2. \tag{50}$$

The optimal rotation matrix that minimizes Eq. (50) when $a = 0$ is [18]

$$R = (XY^{\mathrm{T}} YX^{\mathrm{T}})^{1/2} (YX^{\mathrm{T}})^{-1} \tag{51}$$

where

$$X = [x_1, \ldots, x_n] \quad \text{and} \quad Y = [\chi_1, \ldots, \chi_n]. \tag{52}$$

It makes sense that a should be the zero vector in our case, since all intermediate conformations obtained by our method always have their origin at the center of mass as a result of the constraint in Eq. (47). Alternatively, one can compute the optimal estimate of R in two stages by first computing

$$\hat{R} = \left[\sum_{k=1}^{n} x_k \chi_k^{\mathrm{T}} \right] \left[\sum_{i=1}^{n} \chi_i \chi_i^{\mathrm{T}} \right]^{-1} \tag{53}$$

and then finding the closest rotation matrix R to $\hat{R}$. The solution is the orthogonal matrix in the polar decomposition of $\hat{R}$ as

$$R = \hat{R}(\hat{R}^{\mathrm{T}} \hat{R})^{-1/2}. \tag{54}$$

2.4.2. Incremental rotation method

Given two end conformations denoted as $\{x_i\}$ and $\{\chi_i\}$ and given that we have already computed R such that

$$\sum_{i=1}^{n} \|x_i - R\chi_i\|^2 \tag{55}$$

is minimized, if we make a small change to $\{\chi_i\}$ such that $\chi_i \rightarrow \chi_i + \delta_i$, we easily find ω such that the following is minimized.

$$C(\omega) = \sum_{i=1}^{n} \|x_i - R(\mathbb{E}_3 + \text{mat}(\omega))\chi_i\|^2 \tag{56}$$

where

$$\text{mat}(\omega) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{57}$$

and the rotational matrix is approximated as $\mathbb{E}_3 + mat(\omega)$. The result is quadratic in ω and can be written in the form

$$C(\omega) = \sum_{i=1}^{n} (\omega^T A_i^T A_i \omega + 2b_i^T A_i \omega + b_i^T b_i) \tag{58}$$

where

$$A_i = R\,\text{mat}(\chi_i) \in \mathbb{R}^{3\times3} \quad \text{and} \quad b_i = x_i - R\chi_i \in \mathbb{R}^{3\times1}. \tag{59}$$

Let

$$\Phi = \sum_{i=1}^{n} A_i^T A_i \quad \text{and} \quad \phi = \sum_{i=1}^{n} b_i^T A_i. \tag{60}$$

We can calculate ω such that

$$\frac{\partial C(\omega)}{\partial \omega} = 2\Phi\omega + 2\phi = 0. \tag{61}$$

When we align all intermediate conformations with the two end conformation using this method, R is initially set to $\mathbb{E}_3$ and the first intermediate conformation denoted as $\{x_i^1\}$ is substituted for $\{\chi_i\}$. We get the optimally superimposed set of residues denoted as $\{x_i^{a1}\}$ in the form

$$x_i^{a1} = R^1(\mathbb{E}_3 + \text{mat}(\omega^1))x_i^1 \tag{62}$$

where $R^1 = \mathbb{E}_3$ and ω^1 is the solution of Eq. (61). Note that when we optimize the second intermediate conformation, the given rotation matrix R^2 can be obtained as

$$R^2 = R^1(\mathbb{E}_3 + \text{mat}(\omega^1)) \tag{63}$$

In general,

$$R^{i+1} = R^i(\mathbb{E}_3 + \text{mat}(\omega^i)) \tag{64}$$

Periodically Eq. (54) may have to be used with $\hat{R} = R^i$ to ensure that R^i does not deviate from being a rotation matrix after many iterations.

3. Simulation results

All simulations are performed using Matlab programming and static pictures created are sequentially converted to AVI formats to ensure display in internet browser environments. We have pursued six toy models to test our interpolation method. Fig. 4 demonstrates results for four fundamental planar motions such as elongation, shear, hinge bending, and breathing. Fig. 5 shows two simple, but more complex
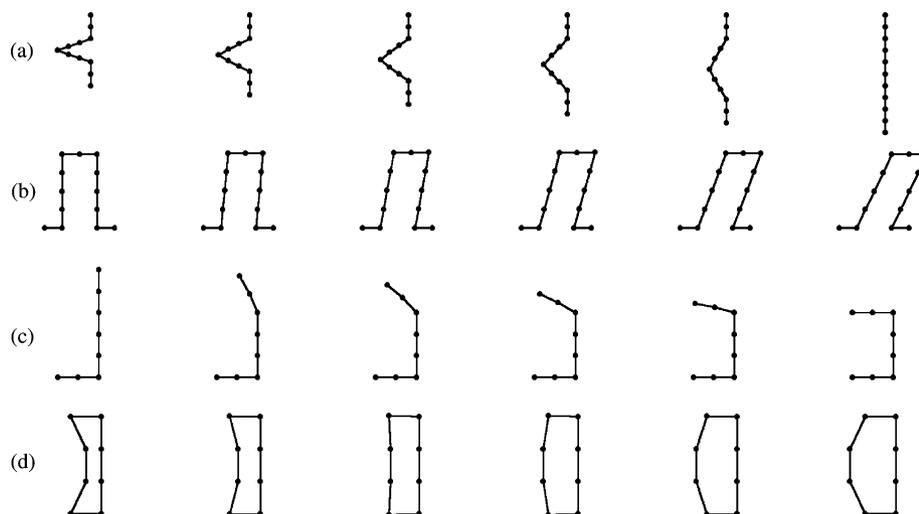
Fig. 4. Simulation results for toy protein models having simple motions. Two end conformations are given and certain points are fixed in the plane to animate relative motions. Ninety-nine intermediate conformations are produced using a constant cutoff distance. Four intermediate conformations are uniformly displayed to generate elongation in (a), shear in (b), hinge bending in (c), and breathing in (d). The model successfully represents all of these transitions.
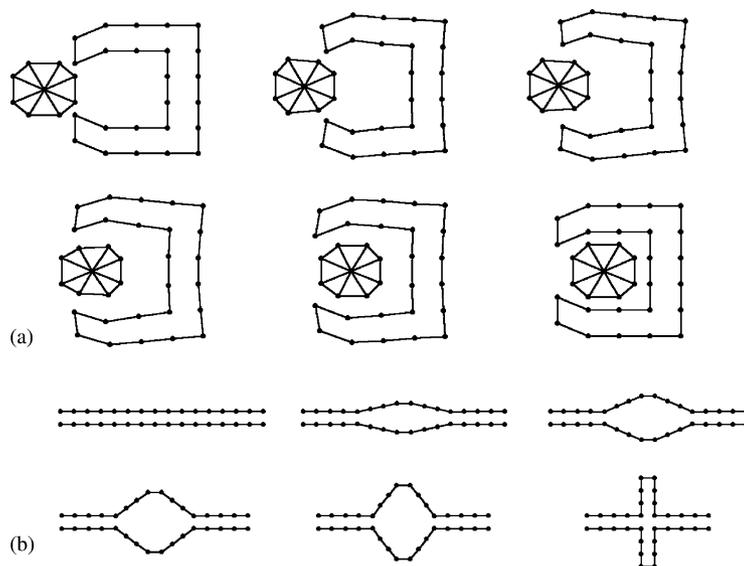
Fig. 5. Simulation of binding and refolding toy models. The toy model to mimic ligand binding is shown in (a). The docking side of the receptor opens to receive the incoming ligand as the ligand approaches and simultaneously the ligand deforms to facilitate entry. As the binding is completed, the receptor and ligand both return to their original conformations. Two parallel lines bow out to represent a nucleic acid transition from double helical B-DNA to a clover leaf junction (Holliday junction structure) in (b).
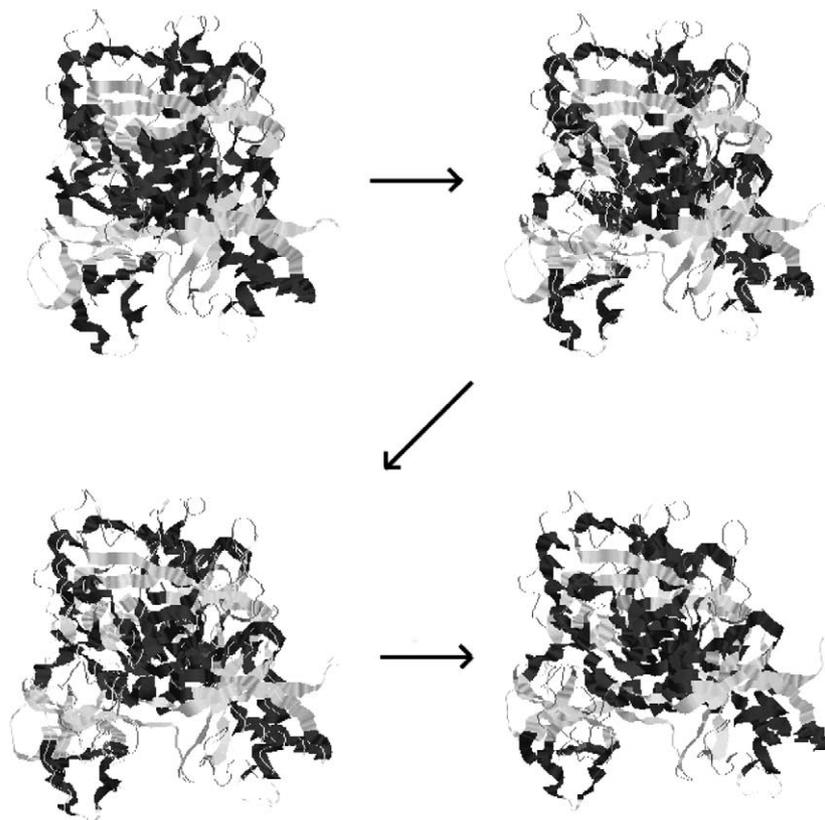
Fig. 6. Simulation of intermediate conformation between lactoferrin forms 1LFG ("closed") and 1LFH ("open") using a limiting contact number cutoff of 20. Here, 99 intermediate conformations are obtained incrementally using our interpolation method and two intermediate conformations are illustrated using RMS superposition. This shows movement of lactoferrin from the "closed" form to the "open" form. This figure was made with Rasmol.

toy models to mimic such motions as ligand binding and nucleic acid Holliday junction formation, respectively. Our method generates plausible pathways for those toy models. In addition, we have chosen a real protein structure having two different structures. Lactoferrin has "open" and "closed" forms, 1LFG and 1LFH, and we generate a feasible pathway between those two forms using our interpolation method.

Our uniform and sparse linking matrix generated by a contact number cutoff enables us to get a feasible pathway within relatively short computational time. Fig. 6 shows the simulation result of the conformational transition of lactoferrin which consists of 691 residues. This simulation illustrates the movement from the closed (diferric) form (1LFG) to the open (apo) form (1LFH). Fig. 7 presents virtual bond angles between sequential α-carbons. These small changes can be easily accommodated within the coarse-grained model since there is only one point per amino acid. Average distances between sequential alpha carbons are also shown in Fig. 7. These do not deviate far from 3.8 Å during transition. Our method observes steric constraints, regardless of the size of the protein.

4. Comparison of NMA with the network interpolation

In this section we compare results generated using the coarse-grained NMA approach of Section 2.1 with those of network interpolation. All-atom NMA is a widely used tool [11–14]. Coarse-grained NMA has the advantage of being computationally faster while still being able to accurately capture the behaviors of the low frequency modes. Let $\{v_i\}$ be the normalized eigenvectors computed from Eq. (29) assuming the harmonic motion $\delta(t) = e^{i\omega_i t} v_i$. The small displacement generated using the network interpolation model for $\alpha = 0.01$ from one of the end states is δ. Fig. 8 plots $F(i) = (\delta \cdot v_i)/\|\delta\|$. This shows the correlation between the behavior of conformational transition and non-rigid modes of the closed and open forms of lactoferrin. As can be seen, small harmonic motions about these equilibria are consistent

(a)

(b)

Fig. 7. Virtual bond angles and lengths in lactoferrin. The average values of virtual bond angles and lengths between sequential α-carbons in all intermediate conformations are shown in (a) and in (b), respectively. These ought not to deviate away far from 106° and 3.8 Å, respectively. Our interpolation method preserves steric constraints well.

(a)   <– lowest   Index of eigenvalues   –> highest

(b)   <– lowest   Index of eigenvalues   –> highest

Fig. 8. Decomposition of the normalized displacement from the network interpolation model over eigenvectors calculated from coarse-grained NMA. The normalized displacement which corresponds to the network interpolation result of 1% deviation from 1LFG to 1LFH is decomposed over the set of orthonormal eigenvectors obtained by NMA of ILFG in (a). Similarly, the decomposition result of the displacement from 1LFH to 1LFG over eigenvectors of 1LFH is shown in (b). Only the first 500 nonrigid lower modes are displayed. The strong concentration near lowest modes shows that small harmonic motions are consistent with the large motions predicted by the network interpolation model.

with the large motions predicted by the network interpolation model near the end conformations.

5. Conclusions

We develop a computationally efficient method for the realistic simulation of macromolecules having a transition between two conformations. Our method is based on a coarse-grained elastic network model. Using cutoffs to the number of nearest neighbors generates a stiffness matrix that is both sparse and uniform, hence, allowing for efficient computations. Simulation results for several toy models and an actual two state protein illustrate that the method developed here reliably generates sequences of feasible intermediate conformations and avoids steric conflicts. Our method is a reasonable compromise between oversimplified linear interpolations and computationally expensive methods such as MD and all atom NMA. Animations produced using this method are posted at http://custer.me.jhu.edu/proteins/movies.html. Among the large number of possible applications, we have demonstrated that one could use this method to analyze ligand binding or nucleic acid polymorphism.

References

[1] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, P.E. Bourne, The Protein Data Bank, Nucleic Acids Res. 28 (2000) 235–242.

[2] S. Subbiah, Protein Motions, R.G. Landes Company, Austin, 1996.

[3] C. Vonrhein, G.J. Schlauderer, G.E. Schulz, Movie of the structural changes during a catalytic cycle of nucleoside monophosphate kinases, Structure 3 (1995) 483–490.

[4] M. Gerstein, W. Krebs, A database of macromolecular motions, Nucleic Acids Res. 26 (1998) 4280–4290.

[5] W. Krebs, M. Gerstein, The morph server: a standardized system for analyzing and visualizing macromolecular motions in a database framework, Nucleic Acids Res. 28 (2000) 1665–1675.

[6] G.J. Kleywegt, T.A. Jones, Where freedom is given, liberties are taken, Structure 3 (1995) 535–540.

[7] G.J. Kleywegt, T.A. Jones, Phi/Psi-chology: Ramachandran revisited, Structure 4 (1996) 1395–1400.

[8] A.R. Atilgan, S.R. Durell, R.L. Jernigan, M.C. Demirel, O. Keskin, I. Bahar, Anisotropy of fluctuation dynamics of proteins with an elastic network model, Biophys. J. 80 (2001) 505–515.

[9] I. Bahar, R.L. Jernigan, Vibrational dynamics of transfer RNAs: comparison of the free and synthetase bound forms, J. Mol. Biol. 281 (1998) 871–884.

[10] I. Bahar, B. Erman, R.L. Jernigan, A.R. Atilgan, D.G. Covell, Examination of collective motions in HIV-1 reverse transcriptase. Examination of flexibility and enzyme function, J. Mol. Biol. 285 (1999) 1023–1037.

[11] S. Jaaskelainen, C.S. Verma, R.E. Hubbard, Conformational change in the activation of lipase: an analysis in terms of low-frequency normal modes, Protein Sci. 7 (1998) 1359–1367.

[12] F. Tama, Y.H. Sanejouand, Conformational change of proteins arising from normal mode calculations, Protein Eng. 14 (2001) 1–6.

[13] M.M. Tirion, D. Ben-Avraham, Normal mode analysis of G-actin, J. Mol. Biol. 230 (1993) 186–195.

[14] M.M. Tirion, D. Ben-Avraham, Normal modes analyses of macromolecules, Physica A 249 (1998) 415–423.

[15] G.S. Chirikjian, Derivation of spring mass models of protein dynamics. John Hopkins University, Baltimore, 2000.

[16] G.M. Crippen, T.F. Havel, Distance Geometry and Molecular Conformation, Wiley, New York, 1988.

[17] A.G. Booth, Visualizing protein conformational changes on a personal computer-alpha carbon pseudo bonding as a constraint for interpolation in internal coordinates space, J. Mol. Graph. Model. 19 (2001) 481–486.

[18] G.S. Chirikjian, A.B. Kyatkin, Engineering Applications of non-commutative Harmonic Analysis, CRC Press, Boca Raton, FL, 2001.