NOLB: Nonlinear Rigid Block Normal Mode Analysis Method

Alexandre Hoffmann, Sergei Grudinin

NOLB : Nonlinear rigid block normal mode analysis method.

Alexandre Hoffmann[†,‡,¶] and Sergei Grudinin[*,†,‡,¶]

†*University of Grenoble Alpes, LJK, F-38000 Grenoble, France*
‡*CNRS, LJK, F-38000 Grenoble, France*
¶*Inria, F-38000 Grenoble, France*

E-mail: Sergei.Grudinin@inria.fr

This document is the unedited Author's version of a Submitted Work that was subsequently accepted for publication in Journal of Chemical Theory and Computation, copyright © American Chemical Society after peer review. To access the final edited and published work see http://dx.doi.org/10.1021/acs.jctc.7b00197.

Abstract

We present a new conceptually simple and computationally efficient method for nonlinear normal mode analysis called NOLB. It relies on the rotations-translations of blocks (RTB) theoretical basis developed by Y.-H. Sanejouand and colleagues[1,2]. We demonstrate how to physically interpret the eigenvalues computed in the RTB basis in terms of angular and linear velocities applied to the rigid blocks and how to construct a nonlinear extrapolation of motion out of these velocities. The key observation of our method is that the angular velocity of a rigid block can be interpreted as the result of an implicit force, such that the motion of the rigid block can be considered as a pure rotation about a certain center.

We demonstrate the motions produced with the NOLB method on three different molecular systems and show that some of the lowest frequency normal modes correspond to the biologically relevant motions. For example, NOLB detects the spiral sliding motion of the TALE protein, which is capable of rapid diffusion along its target DNA. Overall, our method produces better structures compared to the standard approach, especially at large deformation amplitudes, as we demonstrate by visual inspection, energy and topology analyses, and also by the MolProbity service validation. Finally, our method is scalable and can be applied to very large molecular systems, such as ribosomes.

Standalone executables of the NOLB normal mode analysis method are available at https://team.inria.fr/nano-d/software/nolb-normal-modes/. A graphical user interface created for the SAMSON software platform will be made available at https://www.samson-connect.net.

Introduction

Normal mode analysis (NMA) is an old and well established technique[3] that has recently found many new applications in the field of structural biology and structural bioinformatics[4]. The internal motions of a protein have been a topic of great interest for a long time. One reason for this interest is the fact that some of these motions are known to play an important role in protein functions[4–8]. While molecular dynamics (MD) can nowadays accurately predict these motions, it is typically very computationally expensive, whereas NMA is relatively cheap and easily allows us to either extract the so-called *essential dynamics* of the protein from MD tra-

jectories[9], or to compute some low-frequency *collective motions* for a single structure[7,10–12]. These low-frequency motions are particularly interesting to the structural biology community because they are commonly assumed to give more insight into protein function and dynamics[4,13].

Another application of NMA is the description and prediction of conformational transitions in molecular systems. More precisely, these transitions are approximated in a low-dimensional conformation space composed of some lowest normal modes[14–18]. For example, NMA has been successfully applied to protein-ligand docking problems[19,20], protein-protein docking problems[21–25] and others[26,27]. In these applications, NMA is usually used as a supplement to the rigid docking search method[19], and allows to handle the flexibility, and even the conformational changes of proteins with only a few additional degrees of freedom (DOF). We should, however, mention that the relevance of the NMA in protein-protein docking problems is still debatable[24,28].

NMA uses a quadratic approximation of the potential energy, and thus it produces linear deformations of the initial structure, which are accurate only for small-amplitude motions. Larger amplitudes can destroy, for example, the secondary structure and break interatomic bonds, when NMA is applied to a protein. An obvious circumvention for this problem will be to take smaller amplitude steps and iteratively recompute and diagonalize the Hessian matrix from the updated positions. This procedure would indeed produce a more realistic deformation of the initial structure thanks to the nonlinearity of the obtained deformation. However, such an approach requires multiple diagonalization steps, which may be computationally expensive for some of the applications. Thus, multiple attempts were made to introduce nonlinear deformations without the need of multiple diagonalizations.

For example, in mechanical engineering, a Taylor expansion of the normal modes vectors[29] was introduced, but this method requires to compute the derivatives of the normal modes which is a rather complex operation. Another method from structural dynamics considers the expression of normal modes as a function of "master coordinates" and reformulates the equation of motion as an over-determined system of partial differential equations (PDEs)[30], which can be solved using a Galerkin method, for example[31]. To our knowledge, the latter method has only been demonstrated on small systems and solving a large over-determined system of PDEs could be prohibitively costly when applied to a complex system like a protein. In structural biology the most straightworward way to introduce nonlinear motions in NMA is to express the stiffness matrix in the torsion angle subspace[11,32–36]. While NMA in this subspace preserves both bond lengths and angles[34–39], there is a negative long-range effect that limits the applicability of the method to small deformation amplitudes. Indeed, a small change of one dihedral angle causes a propagation of Cartesian displacements along the main chain of a protein or other biopolymer with an increasing amplitude of deformation. Truly, changes in torsional angles result in nonlinear Cartesian trajectories of atoms, and Bray et al.[40] demonstrated that NMA performed in this subspace and projected to curved paths in Cartesian coordinates describes protein conformational changes more accurately than linear Cartesian motions.

In this paper we present a conceptually simple and intuitive scheme for *nonlinear normal mode analysis* that extrapolates a motion computed from *instantaneous linear and angular velocities* to large amplitudes. The scheme can be considered as an evolution of the widely used rotations-translations of blocks (RTB) method presented below. Thus it will be easy to integrate this scheme with the existing software tools based on the RTB approach.

The Method

Harmonic oscillator model

In order to briefly explain the NMA method, let us consider a molecular system with N_a atoms at an equilibrium position $q_0 \in \mathbb{R}^{3N_a}$. We call

$V : \mathbb{R}^{3N_a} \mapsto \mathbb{R}$ the potential energy of our molecular system. We aim at analytically computing the molecular vibration of our system around its equilibrium. In order to do so we introduce $q : \mathbb{R}^+ \mapsto \mathbb{R}^{3N_a}$, a small time-dependent molecular displacement of our system around q_0. The basic idea of the NMA method is to represent the potential energy V in the vicinity of q_0 by its quadratic approximation and to *analytically* solve the Newton's equation of motion,

$$M(\ddot{q} + \ddot{q_0}) + \nabla V(q_0 + q) \approx M\ddot{q} + Kq = 0, \quad (1)$$

where M is the diagonal mass matrix, and K is the *Hessian matrix* of the potential energy V evaluated at the equilibrium position q_0. We should note that in classical mechanics K is traditionally called the *stiffness matrix*. A standard method to analytically solve this kind of ordinary differential equation (ODE) is to uncouple it by computing the square matrix of eigenvectors L and the diagonal matrix of eigenvalues Λ of the *mass-weighted stiffness matrix* $K_w = M^{-1/2}KM^{-1/2}$,

$$K_w = L\Lambda L^T. \quad (2)$$

We call $\eta : \mathbb{R}^+ \mapsto \mathbb{R}^{3N_a}$ the projection of q into the eigenspace of K_w and we call $(\lambda_i)_{i=0\ldots3N_a}$ the diagonal values in Λ. Then, left multiplication of the ODE 1 by $L^T M^{1/2}$ gives the following system of uncoupled equations,

$$\begin{aligned} \eta &= L^T M^{1/2} q \\ \ddot{\eta_i} + \lambda_i \eta_i &= 0 \quad i = 1 \ldots 3N_a. \end{aligned} \quad (3)$$

which can be solved by the classical ODE theory. In classical mechanics and engineering, an eigenvector associated with a harmonic oscillator described above is called a *normal mode* with the corresponding oscillating frequency given as $\sqrt{\lambda_i}$. The potential energy associated with a harmonic oscillation of this mode is thus linearly proportional to λ_i. Note that since the potential energy V is expanded around its local minimum, all $\lambda_i \geq 0$. Typically, the five or six first eigenvalues are null as they correspond to the rigid body motions, which lie in the kernel of K_w. Since for most of the applications we are only interested in collective motions, it seems natural to only consider a few of the lowest non-trivial normal modes.

RTB model

In the past, many methods were developed to reduce the dimensionality of the original NMA problem. For example, pioneering works of Noguti and Gõ [41] and Levitt et al. [33], and later of Ma et al. [35] and Chacón et al. [36] explored the NMA approach in internal coordinates, Hinsen [13] proposed NMA in a reduced Fourier basis, and finally, many researchers considered coarse-graining [1,2,42–44]. One of the first and most popular methods to coarse-grain the initial system is the RTB approach introduced by Durand et al. [1] and further developed by Tama et al. [2] and Li and Cui [26]. In this method, individual or several consecutive amino residues are considered as rigid blocks that can only exhibit rotational and translational motions [1,2]. A very detailed description of the RTB method was made by Lezon et al. [45], where the authors provided both the mathematics behind the method along with a nice physical interpretation of the model. We should note that recently an extension to the RTB method called cluster-NMA has been proposed [42,46]. It follows the same philosophy as the RTB method, but considers real rigid block rotations about C_α atoms. The RTB method allows to compute a fairly accurate approximation to the normal modes of a large molecular system, typically a protein, in a reasonable amount of time. The main idea of the method is to describe the system as n rigid blocks, and to present the infinitesimal internal motions of the system as a set of infinitesimal rotations and translations of these rigid blocks. The transition from the RTB coordinate system with $6n$ DOFs to the all-atom coordinate system with $3N$ DOFs is performed by an orthogonal projection matrix $P \in \mathbb{R}^{3N \times 6n}$. The normal modes in the RTB coordinates are then computed by calculating the eigenvectors of the matrix $P^T K_w P$.

Let us now briefly explain how the projection matrix P is obtained. We start from the conservation laws of the linear momentum and the angular momentum of a rigid block consisting

of N_b atoms written in mass-weighted coordinates,

$$\begin{aligned} \sqrt{M_b}\dot{\tilde{q}} &= \sum_{k=1}^{N_b} \sqrt{m_k}\dot{q}_k \quad \text{for a translation} \\ \mathrm{I}^{1/2}\,\dot{\tilde{q}} &= \sum_{k=1}^{N_b} \sqrt{m_k}(q_k \times \dot{q}_k) \quad \text{for a rotation} \end{aligned}, \tag{4}$$

where M_b is the total mass of the rigid block, I is the rigid block's inertia tensor, $\tilde{q}$ is the blocks's displacement, m_k is the mass of the k^{th} atom of the block, and q_k is the displacement of the k^{th} atom of the block. The elements constituting P^T, the matrix projecting an all-atom motion q into a motion of rigid block $\tilde{q}$ are then obtained by differentiating (4) with respect to $\dot{q}_k$ [45]. This leads to translation $\mathrm{P_t}$ and rotation $\mathrm{P_r}$ matrices of size $3N_b \times 3$ each, computed for each of the rigid blocks and written through their k 3×3 square components,

$$\begin{aligned} \mathrm{P_t}_k &= \sqrt{\frac{m_k}{M_b}}\,\mathrm{E}_3 \quad \text{for a translation} \\ \mathrm{P_r}_k &= -\sqrt{m_k}(\mathrm{I})^{-1/2}[r_k - r^{COM}]_\times \quad \text{for a rotation} \end{aligned}, \tag{5}$$

where k is one of N_b atom indices, r_k is the position of the corresponding atom in the block, and r^{COM} the position of the block's center of mass (COM). The rigid block's displacement (δ, θ) 6-vector is then obtained by summing up the displacements in the RTB coordinate frame,

$$\begin{aligned} \delta &= \sum_{k=1}^{N_b} \mathrm{P_t}_k^T\, q_k \quad \text{for a translation} \\ \theta &= \sum_{k=1}^{N_b} \mathrm{P_r}_k^T\, q_k \quad \text{for a rotation} \end{aligned}. \tag{6}$$

Having written these equations, we can write the projection matrix P as a diagonal block matrix,

$$P = \begin{pmatrix} \mathrm{P_t}^1 & \mathrm{P_r}^1 & & \\ \ddots & \ddots & \ddots & \\ & & \mathrm{P_t}^n & \mathrm{P_r}^n \end{pmatrix}. \tag{7}$$

This projection allows, in principle, to drastically reduce the size of the stiffness matrix, which made it possible to study large biomolecular assemblies [2,26,47]. We want to emphasize that the rotational part in (7) is an *infinitesimal rotation* of a rigid block about its center of mass. Linearization of this rotation, as we will show below, distorts the interatomic distances and produces unrealistic molecular conformations for large deformation amplitudes. Below we will introduce a new method to circumvent this problem.

RTB normal modes

The main idea of our method is the *nonlinear extrapolation* of the instantaneous directions of motion described by the normal modes computed in the RTB subspace. These normal modes are calculated by the diagonalization of the RTB-projected mass-weighted stiffness matrix,

$$P^T K_w P = \tilde{L}\tilde{\Lambda}\tilde{L}^T, \tag{8}$$

where $\tilde{L}$ is the matrix composed of the RTB normal modes with the corresponding diagonal eigenvalue matrix $\tilde{\Lambda}$. This equation can be rewritten as

$$K_w \approx (P\tilde{L})\tilde{\Lambda}(P\tilde{L})^T. \tag{9}$$

Comparing this equation with (2) we can obtain the all-atom normal modes L^w (in mass-weighted coordinates) as a projection of RTB normal modes $\tilde{L}$ according to

$$L^w = P\tilde{L}. \tag{10}$$

The original RTB method used this equation to compute a linear deformation of the original molecular structure along directions L^w. However, we are going to use the structure of the projection matrix P to compute a more natural nonlinear deformation.

To proceed further, first of all we should remind that the size of an RTB normal mode vector L_i^w is $6n$, with n being the number of rigid blocks. Each six consecutive coordinates in this vector correspond to a certain rigid block, with the first three coordinates providing the *instantaneous displacement* of the rigid block's COM, and the second three coordinates providing the *instantaneous axis of rotation* of the rigid block,

passing though its COM. Introducing time dependence of the variables and taking the time derivative of the RTB normal mode vector, we can interpret its components as *instantaneous linear velocities* and *instantaneous angular velocities* of individual rigid blocks. The final step of our method is the extrapolation of rigid blocks' motion for large amplitudes starting from their instantaneous linear and angular velocities.

Extrapolation of the instantaneous motion

In the standard NMA method, molecular vibrations in a multi-dimensional harmonic oscillator are all uncoupled and can be found by solving (3), which gives a sinusoidal function as a time-dependent vibration in normal coordinates[3]. For our applications, however, it is sufficient to only consider the maximum amplitude of the deformation a, which, in principal, can be a sinusoidal function in time as well. Then, for a rigid block with mass M_b and inertia tensor I, given its instantaneous linear velocity $\vec{v}_w$ and its instantaneous angular velocity $\vec{\omega}_w$ expressed in mass-weighted coordinates, we first compute these in the non-mass weighted coordinates as follows,

$$\begin{aligned} \vec{v} &= M_b^{-1/2}\vec{v}_w \\ \vec{\omega} &= \mathrm{I}^{-1/2}\,\vec{\omega}_w \end{aligned}. \tag{11}$$

Then, the translational increment in the rigid block's position $\Delta\vec{x}$ and the angular increment in its orientation $\Delta\phi$ can be computed as

$$\begin{aligned} \Delta\vec{x} &= a\vec{v} \\ \vec{n} &= \vec{\omega}/||\vec{\omega}||_2, \\ \Delta\phi &= a||\vec{\omega}||_2 \end{aligned} \tag{12}$$

where the rigid block's rotation is described with a unit axis $\vec{n}$ passing though its COM and an angle ϕ. We see that the motion of the rigid block can now be described as a rotation about an axis $\vec{n}$ by an angle $\Delta\phi$, denoted as $R(\Delta\phi, \vec{n})$, followed by a translation by a vector $\Delta\vec{x}$. These rotation and translation operations are computed starting from instantaneous angular and linear velocities. The question is, however, how

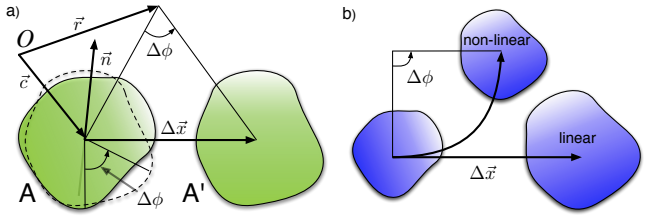

Figure 1: A motion of a rigid block. a) A rigid block A with the COM at $\vec{c}$ is rotated about an axis $\vec{n}$ by an angle $\Delta\phi$, and then translated by a vector $\Delta\vec{x}$. The result of the first rotation is shown as a dashed outline. The result of the complete motion is denoted as A'. This motion can be also represented as a pure rotation about a new center $\vec{r}$ by the same angle $\Delta\phi$. The origin of the coordinate system is denoted as O. b) Schematic comparison of the linear and nonlinear motion extrapolation methods. Please notice the final positions of the centers of mass and the size of the rigid blocks.

do we interpret this motion at large deformation amplitudes a? We should note that, generally, rotation and translation do not commute and thus the order of these operations matter. However, this is not the case for infinitesimal rotations. At large amplitudes, we are free to choose the order of the operations for our extrapolation, and for simplicity we stick to the order when the rotation is applied first.

If no external forces act on a rigid block, due to the momenta conservation laws it will be moving along a straight line. However, our rigid blocks interact with each other and thus they are always subject to implicit forces. Therefore, it is reasonable to assume that the angular velocity of a rigid block is the result of a rotation about certain center. This is the *key observation* of our method. Now, let us represent the motion of the rigid block as a *pure rotation* about a new rotation center $\vec{r}$, as it is shown in Figure 1a. Without loss of generality we assume that $\vec{r}$ is orthogonal to $\vec{n}$, i.e. $\vec{r}\cdot\vec{n} = 0$. The new rotation center $\vec{r}$ can be found from the following identity,

$$R(\Delta\phi, \vec{n})(\vec{A}-\vec{c})+\vec{c}+\Delta\vec{x} = R(\Delta\phi, \vec{n})(\vec{A}-\vec{r})+\vec{r}, \tag{13}$$

where $\vec{A}$ are atomic positions of the rigid block,

and $\vec{c}$ is its COM vector. It is easy to demonstrate that this equation has a solution only when vectors $\Delta\vec{x}$ and $\vec{n}$ are orthogonal to each other, which is not, generally, the case in 3D. To provide a unique solution, we represent the increment in the rigid block's position $\Delta\vec{x}$ as a sum of two orthogonal vectors,

$$\Delta\vec{x} = \Delta\vec{x}_{\perp} + \Delta\vec{x}_{||}, \tag{14}$$

where $\Delta\vec{x}_{\perp}$ is orthogonal to $\vec{n}$ and $\Delta\vec{x}_{||}$ is collinear to $\vec{n}$. If only the orthogonal $\Delta\vec{x}_{\perp}$ translation is used for the extrapolation of motion, then the unique solution of (13) is given as

$$\vec{r} = \vec{c} + \Delta\vec{x}_{\perp}/2 + (\vec{n} \times \Delta\vec{x}_{\perp})/(2\tan\frac{\Delta\phi}{2}). \tag{15}$$

We can see that the center of the extrapolated rotation $\vec{r}$ depends on the amplitude of the rotation angle $\Delta\phi$, which is not suitable for our reasons. However, at infinitely small amplitudes, it converges to the following value,

$$\vec{r}_0 \underset{a\to 0}{=} \vec{c} + (\vec{n} \times \vec{v}_{\perp})/||\vec{\omega}||_2, \tag{16}$$

which we will use for the motion extrapolation. Finally, extrapolated positions $\vec{A}'$ of a rigid block with initial atomic positions at $\vec{A}$ are given as

$$\vec{A}' = R(\Delta\phi, \vec{n})(\vec{A} - \vec{r}_0) + \vec{r}_0 + \Delta\vec{x}_{||}. \tag{17}$$

Figure 1b schematically shows a comparison between the proposed nonlinear extrapolation method and the standard linear extrapolation. Please note the final positions of the rigid blocks. Please also note that the linear extrapolation method does not preserve the shape of the rigid block. Mathematically speaking, this happens because the underlying transformation matrix represents a linearized rotation and has a non-unity determinant.

Potential function

In principle, our approach can be used with any potential function. However, to omit the need of initial system's energy minimization, we have chosen an all-atom anisotropic network model (ANM)[12,48], which is a type of elastic network model where the initial structure is always at equilibrium. The assessment of several elastic network models can be found, i.e., in Fuglebakk et al.[49],[50]. We specifically chose the all-atom model to make sure that torques acting on the rigid blocks are accurately computed. The all-atom ANM has the following potential function,

$$V(q) = \sum_{d_{ij}^0 < R_c} \frac{\gamma}{2}(d_{ij} - d_{ij}^0)^2, \tag{18}$$

where d_{ij} is the distance between the i^{th} and the j^{th} atoms, d_{ij}^0 is the reference distance between these atoms, as found in the original structure, γ is the stiffness constant, and R_c is a cut-off distance, typically between 8 Å and 15 Å. The stiffness matrix corresponding to this potential function is composed of the following blocks[4,12,48],

$$\begin{aligned} H_{ij} &= -\frac{\gamma}{(d_{ij}^0)^2}\begin{pmatrix} (x_{ij}^0)^2 & x_{ij}^0 y_{ij}^0 & x_{ij}^0 z_{ij}^0 \\ y_{ij}^0 x_{ij}^0 & (y_{ij}^0)^2 & y_{ij}^0 z_{ij}^0 \\ z_{ij}^0 x_{ij}^0 & z_{ij}^0 y_{ij}^0 & (z_{ij}^0)^2 \end{pmatrix} \quad i \neq j \\ H_{ii} &= -\sum_{j\neq i} H_{ij} \end{aligned}, \tag{19}$$

where $x_{ij} = x_i - x_j$, $y_{ij} = y_i - y_j$, and $z_{ij} = z_i - z_j$. In practice, to rapidly compute this matrix, we use an efficient neighbor search algorithm[51]. We should note that for large systems the atomic representation is not needed if we are only interested in a few slowest collective motions. Thus, in principle, our method can also use a simplified model, i.e., with only C_α atoms provided that each rigid block will be composed of at least four particles to have non-singular inertia tensors and non-zero torques.

Test cases

To assess the method, we have selected three types of tests. Fist, we chose three molecular systems for the visual inspection of the motions. These systems are the T7 large terminase (pdb code 4bij), the TAL effector PthXo1 bound to its DNA target (pdb code 3ugm), and the cytoplasmic domain of a bacterial chemore-

ceptor from thermotoga maritima (pdb code 2ch7). Our second test is the energy comparison between the linear and nonlinear deformations along some low-frequency modes at different deformation amplitudes. For this test we have selected four structures of molecular systems from those provided in the 2015/2016 Cryo-EM Model Challenge[52]. These are the structure of the T7 large terminase described above, the structure of the human γ-secretase (pdb code 5a63), the structure of the capsaicin receptor TRPV1 (pdb code 3j9j), and the structure of the TRPV1 ion channel (pdb code 3j5p). Finally, in the third test we measured the memory and CPU consumption of our method with five molecular structures of increasing size ranging from 4,630 of atoms to 284,479 of atoms. These are the structure of the cytoplasmic domain of a bacterial chemoreceptor from thermotoga maritima (pdb code 2ch7 with 4,630 of atoms excluding hetero atoms), the structure of the human γ-secretase (pdb code 5a63 with 9,646 of atoms excluding hetero atoms), the structure of the T7 large terminase (pdb code 4bij with 18,855 of atoms excluding hetero atoms), the structure of the photosystem II complex (pdb code 5b5e, 40,908 of atoms excluding hetero atoms), and the structure of the E. coli 70S ribosome (pdb code 5j8a, 284,479 of atoms excluding hetero atoms). We should mention that the last structure is one of the largest that the protein data bank[53] currently contains.

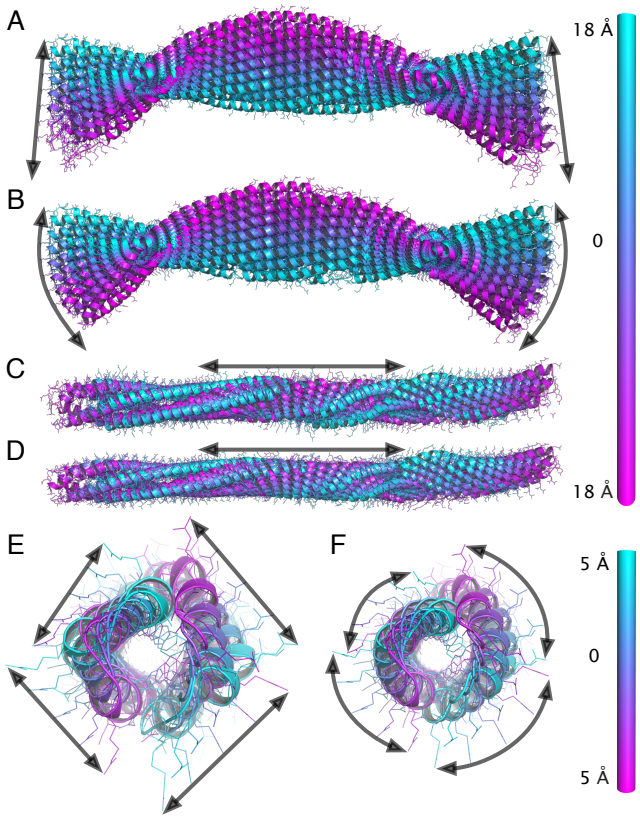

Figure 2: Comparison of linear (A, C, E) and nonlinear (B, D, F) motion extrapolations of a coiled coil protein (pdb code 2ch7). Three types of motions are shown, bending (A, B), stretching (C, D), and twisting (E, F). Several snapshots at different deformation amplitudes are superposed to each other. These are colored according to the values of the overall deformation, as measured by the root-mean-square deviation (RMSD). The colorbars show the RMSD with respect to the initial position. The arrows follow the trajectories of individual atoms.

Results and discussion

Visual inspection of the nonlinear motions

For the first test we have computed some lowest-frequency normal modes for several molecular systems and present the difference between the linear and the nonlinear extrapolation approaches, as it is described below. The first molecular system demonstrates three basic types of internal motions (see Figure 2) and the other two systems illustrate some biologically relevant motions (see Figure 3). Overall, Figures 2 and 3 clearly demonstrate that the nonlinear extrapolation produces visually better and physically more realistic motions than the standard approach. We should mention that in this test we used a single residue as a rigid block. We have additionally performed experiments with a larger number of residues per block, up to 10, and the results are very similar with the same conclusions as stated below.

There are, generally, three basic types of internal motions that a molecular system may exhibit. These are bending, stretching and

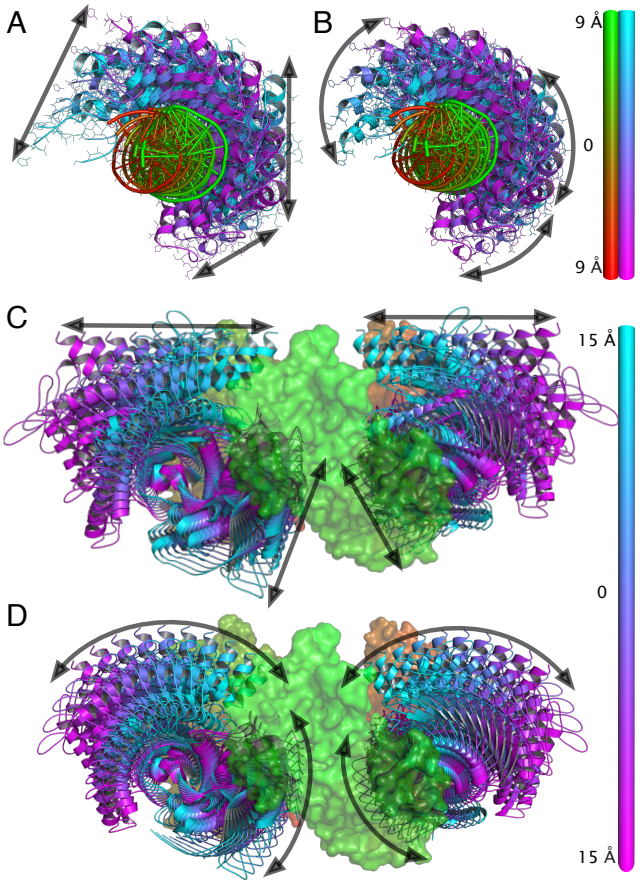

Figure 3: Comparison of linear (A, C) and nonlinear (B, D) motions computed for two molecular systems. Several snapshots at different deformation amplitudes are superposed to each other. These are colored according to the values of the overall deformation, as measured by the RMSD. The colorbars show the overall RMSD with respect to the initial positions. The arrows follow the trajectories of individual atoms. (A, B). Sliding of a DNA-binding protein (colored from cyan to purple) in the groove of the surface of the DNA (pdb code 3ugm). (C, D). Motion of two subunits of a terminase pentamer protein (pdb code 4bij). Three other subunits are shown in the surface representation. The 5-fold symmetry axis points towards the top of the figure.

twisting. All of these motions can be clearly seen with symmetric elongated rod-like objects. Therefore, for the first illustration we have chosen a coiled-coil water-soluble protein from the cytoplasmic domain of a bacterial chemoreceptor (pdb code 2ch7). For this protein, we have computed its ten lowest normal modes and specifically selected those that correspond to the described basic types of motions. Then, we have computed the linear and nonlinear motion extrapolations at different amplitudes. These are presented in Figure 2. The difference between the two types of extrapolations is especially apparent for motions with a large portion of involved rotation. For example, Figures 2A-B show a bending type of motion and Figures 2E-F show a twisting motion. For these two types of motions the difference between the two extrapolation approaches is visually clear. This is because for these types of motions the translational component is typically negligible with respect to the rotational component, which is given as a pure rotation of rigid blocks about a certain center. Thus, the nonlinear extrapolation produces a very different result at large deformation amplitudes. However, for the stretching motion, which is shown in Figures 2C-D, there is no noticeable visual difference between the two types of motion extrapolation. This is because in this case the motion is mostly represented by its translation component and there is almost no difference between the two extrapolation approaches.

Another interesting type of motion where the nonlinear extrapolation produces a noticeable different result is the spiral sliding of a transcription activator-like effector (TALE) protein in a surface groove of its DNA target. This motion, both using linear and nonlinear extrapolations at large amplitudes, is shown in Figures 3A-B. Here, we can see very similar motions of the DNA molecule (colored from green to red), while the extrapolated motions of the TALE protein (colored from cyan to purple) look more physically realistic in the nonlinear case. We should note that the maximum overall RMSD, as measured for the linear extrapolation, is about 9 Å. At such large deformation amplitudes, the linear extrapolation significantly perturbs the structure, as can be illustrated by broken covalent bonds. We should also emphasize that this sliding motion, as computed by the NOLB analysis around the system's equilibrium position, is biologically relevant, as has been recently demonstrated by the

direct observation of TALE protein dynamics[54]. More precisely, the TALE proteins are capable of rapid diffusion along DNA using a combination of sliding and hopping.

Finally, as the last example, we have chosen a pentameric assembly of terminase proteins with the C_5 cyclic symmetry. The terminase is a powerful motor that converts ATP hydrolysis into mechanical movement of the DNA[55]. Similar to the previous examples, we have computed the lowest normal modes for the whole assembly and chosen the one that is responsible for the opening and closing of the channel in the middle of the assembly. More precisely, here each of the five subunits rotates symmetrically such that the channel in the middle changes its shape. Figures 3C-D show the difference between both the linear and the nonlinear extrapolations of this motion. In order to make the figure more comprehensible, we show the motion of only two out of five subunits, colored from cyan to purple according to the amplitude of the deformation. The three remaining subunits (shown in surface representation) are static. Again, we can see that at large amplitudes the nonlinear extrapolation looks more physically realistic than the linear one. Similar to the previous example, this motion composed of symmetric rotations of each of the five subunits, as computed by the NOLB analysis, is biologically relevant and has been noticed during the cryo-electron microscopy reconstruction of the T7 large terminase[55]. More precisely, the five terminase subunits rotate to adapt the channel in such a way that it can accommodate the guest DNA.

Topology comparison

As we have discussed above, the nonlinear normal modes approach demonstrated *visually* better results on all the tested examples. However, both linear and nonlinear extrapolation methods result in physically unrealistic local geometries at large deformation amplitudes. Thus, an additional energy minimization is typically required to relax the locally disturbed molecular geometries. Therefore, in this test we estimate the computational difficulty of such

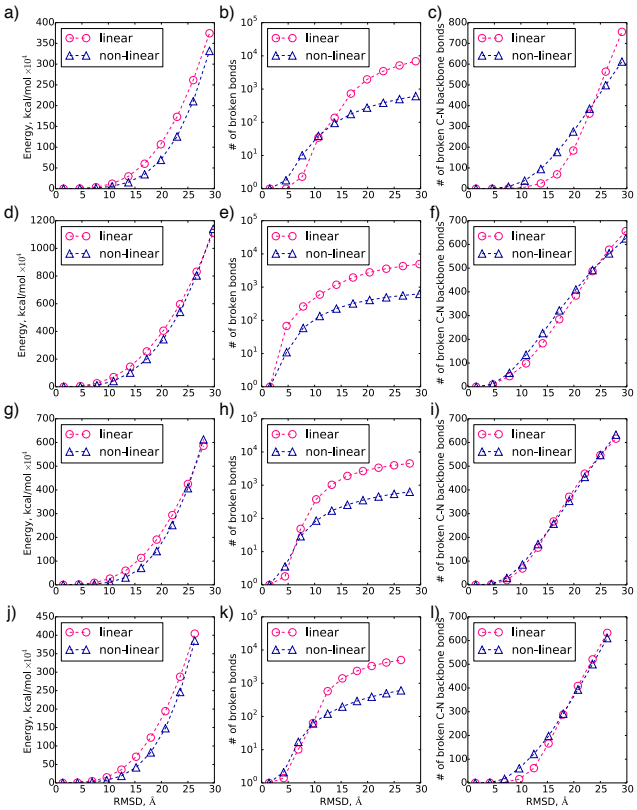

Figure 4: Comparison of linear and nonlinear deformations averaged over the 10 lowest normal modes computed for the following systems, (a-c) 4bij, (d-f) 5a63, (g-i) 3j9j, and (j-l) 3j5p. In (a,d,g,j) the bond harmonic energy as a function of the deformation amplitude is shown. In (b,e,h,k) the total number of broken bonds as a function of the deformation amplitude is shown (in a log scale). In (c,f,i,l) the number of broken bonds between individual amino acids is shown as a function of the deformation amplitude. See the main text for details.

a minimization, which should be proportional to the deformation energy of the final structure. More precisely, we assume that the covalent bonds in the initial molecular structure are represented by harmonic springs with a force constant of 500 kcal/(mol Å^2), which is a typical value in classical force fields[56,57], and we also assume that the total potential energy in the system is given by the sum of the bond contributions.

For this test, we measured the potential energy of the molecular structures generated by both linear and nonlinear extrapolations at var-

ious deformation amplitudes. Figures 4(a,d,g,j) show potential energy for several molecular structures averaged over ten lowest normal modes as a function of the overall RMSD of the final structure with respect to the initial one. We can see that for all the systems the nonlinear normal modes approach produces geometries with a lower bond energy than the standard linear NMA method, at least for deformations that do not exceed 25 Å in RMSD. This means that, in principle, it will be computationally more efficient to optimize the structures produced by the NOLB approach compared to the standard one.

To extend the analysis of the produced molecular topologies, we compared the number of broken covalent bonds in the final molecular structures. We define a covalent bond between two atoms as broken if its length exceeds the sum of the corresponding van der Waals radii multiplied by a factor of 0.6. Figures 4(b,e,h,k) show the total number of broken covalent bonds for the two approaches and clearly demonstrate that the linear extrapolation perturbs local molecular geometries much more compared to the NOLB method. Indeed, we can see that the gap between the two curves increases when the deformation amplitudes become larger. However, since the NOLB NMA approach relies on the rigid body dynamics and all the individual amino acids are treated as rigid blocks, we additionally compared the number of broken covalent bonds between individual amino acids for the two extrapolation approaches. Figures 4(c,f,i,l) show these comparisons. For this case we can see that at small deformation amplitudes, the NOLB method breaks more covalent bonds, which should be expected. At large deformation amplitudes, however, the NOLB method performs better than the standard approach. Nonetheless, we should only consider the total number of broken bonds, or the total deformation energy of the system. In all the cases, as Figure 4 demonstrates, the NOLB NMA approach produces much better results compared to the standard method. In Supporting Information we also provides individual tables that list the data for each of the normal modes individually for all the described molecular structures.

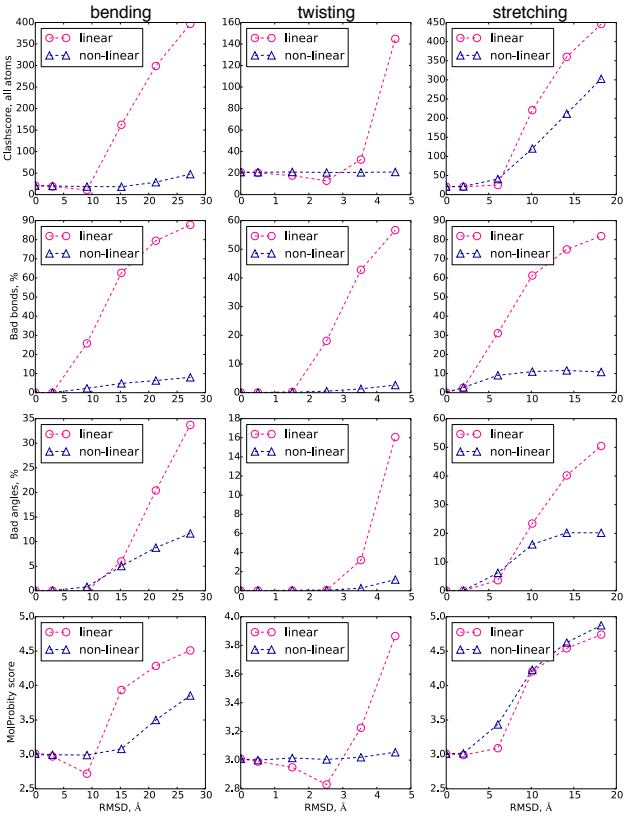

Figure 5: Comparison of linear and nonlinear deformations of the coiled-coil cytoplasmic domain of a bacterial chemoreceptor (pdb code 2ch7) assessed by the MolProbity server.[58] Results for the bending motion are shown in the left column, for the twisting motion are shown in the middle column, and for the stretching motion are shown in the right column. Multiple MolProbity statistics are plotted as a function of the deformation amplitude. The 'clashscore' is the number of serious clashes (atomic overlap ≥ 0.4 Å) per 1,000 atoms. Bad bonds and angles are those that are further away than four standard deviations from the expected values. The MolProbity score is a log-weighted combination of the clashscore, the percentage of not favored Ramachandran angles, and the percentage of bad side-chain rotamers, giving one number that reflects the crystallographic resolution at which those values should be expected.

To complete the analysis, we have also evaluated the quality of several selected structures using a popular MolProbity server.[58] For this evaluation we chose three types of deformations

of a coiled coil cytoplasmic domain of a bacterial chemoreceptor presented in Fig. 2, namely, bending, twisting, and stretching. MolProbity is a structure validation web service widely used to evaluate the quality of X-ray or NMR structures. For the analysis it uses a variety of physics- and knowledge-based algorithms. Figure 5 presents the computed MolProbity statistics. More precisely, it shows the amount of serious clashes (with atomic overlap ≥ 0.4 Å), the percentage of statistically abnormal bonds and angles, and finally, the cumulative 'MolProbity score', which reflects the crystallographic resolution at which these structures should be expected. As before, we can see that at large deformation amplitudes the NOLB method produces consistently better structures than the standard linear approach. This conclusion is true for all studied types of motions. At small deformation amplitudes, the linear NMA approach performs slightly better if we consider the total number of serious clashes in the structures. Interestingly enough, this number can even decrease compared to the crystallographic structure, presumably because of its moderate resolution.

The presented examples demonstrate that the NOLB approach is able to generate structures with a fewer number of geometric distortions compared to the linear NMA method. However, after a certain amplitude of deformation, our method will also produce topological artifacts. This amplitude will generally depend on the type of motion, or, more technically, on the amount of the involved rotation compared to the translation (see eqs. 14 and 17). For a pure rotational motion, for example, trajectories of all the rigid blocks will be located on certain circles and thus the maximum geometrical distortion of the structure will be always bounded by the circles radii regardless the deformation amplitude. Figure 2F gives a fare approximation of such a motion. For the other extreme case of a pure translational motion, there will be no difference between the two approaches and the distortions produced by the NOLB method will be the same as in the standard approach, as it is shown in Fig. 2D.

We would like to conclude this section mentioning that structural distortions presented above are not a serious obstacle for the applicability of the Cartesian NMA approaches. Indeed, the produced molecular structures can be straightforwardly optimized using standard techniques, for example, gradient-based minimizers and classical force-fields. However, as we hinted above, it will be computationally more efficient to optimize a structure produced by the NOLB approach compared to the linear one due to a typically lower energy of the NOLB structure. Also, at large NMA deformation amplitudes, the result of such an optimization for the linear technique will be generally different from the one of the nonlinear technique. Thus, the presented NOLB approach is a computationally cheap alternative to the other NMA methods when large deformation amplitudes are required.

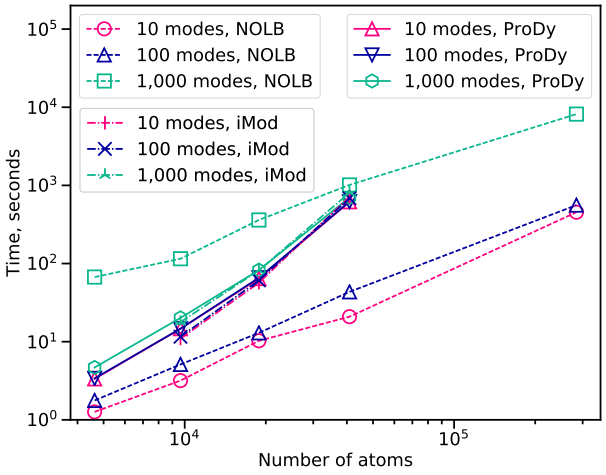

Figure 6: Total time taken by the NOLB, ProDy and iMod methods to compute first 10, 100, and 1,000 normal modes for five molecular structures as a function of their size in a log-log scale. Several data points are missing because ProDy failed on the largest system and iMod failed on the smallest and the largest systems. See the main text for details.

Memory and CPU consumption

Finally we demonstrate the scalability of our method on five molecular structures of various sizes and geometries, as we have described

Table 1: Memory consumption of the NOLB NMA method on the tested molecular structures. All the computations were performed using the double precision variables. We set the interatomic interaction cutoff to 10 Å. The number of atoms is listed without the heteroatoms. The size of the matrices is given as the number of rows (or columns) they contain.

Name	PDB code	Number of atoms	All-atom Hessian size	RTB Hessian size	Memory required
Chemoreceptor	2ch7	4,630	13,890	3,702	123 Mb
Human γ-secretase	5a63	9,646	28,938	7,338	310 Mb
Terminase	4bij	18,855	56,565	14,220	570 Mb
Photosystem II	5b5e	40,908	122,724	31,494	1,3 Gb
70S ribosome	5j8a	284,479	853,437	123,804	9,3 Gb

in more detail above. We should specifically mention that these results only demonstrate the performance of our RTB NMA implementation. The subsequent nonlinear analysis of the motions takes only a marginal piece of the total time, which can be ignored. More technically, our method uses sparse data representation and the Lanczos scheme to find a subset of eigenvectors of the Hessian matrix. As a reference, we also provide results of other state-of-the-art NMA methods. These are the RTB module of the ProDy package[59] and the iMod method that performs NMA in internal coordinates.[36] Both of these methods operate with dense matrices and use LAPACK routines for the partial diagonalization. ProDy computes a subset of eigenvectors of a real symmetric matrix, whereas iMod seeks for a subset of eigenvectors of the generalized symmetric definite eigenvalue problem. We should mention that we also tested the original RTB NMA implementation of Yves-Henri Sanejouand and colleagues,[1,2] but it turned out to be much slower than the other tested methods because of the full Hessian diagonalization. Also, the CHARMM program used to have a generalized RTB method called block-normal-modes (BNM),[26] but it disappeared from the recent CHARMM releases and we could not assess its performance. We present the numerical results measured on a MacBook Pro Mid 2015 laptop with a 2.8 GHz Intel Core i7 processor and 16 GB 1600 MHz DDR3 RAM. The same interaction cutoff value of 10 Å was used in all the tested methods. The rigid blocks in both ProDy and NOLB were constructed on a single residue basis. For the iMod method, we chose all the dihedral angles as degrees of freedom. Table 1 lists the memory consumption of the NOLB method on the tested structures. We can see that even the structure of the E. coli 70S ribosome with ∼300,000 of atoms, which is one of the largest in the protein data bank, can be computed with our method on all the modern computers. Figure 6 shows the total execution time of the NOLB, ProDy and iMod methods to compute the first 10, 100, and 1,000 normal modes for five systems of increasing size. We should mention that the NOLB method spends almost all of its time in the diagonalization of the Hessian matrix, thus its total time can be generally attributed solely to the diagonalization procedure. Also, in these tests, we have disabled the output of the computed normal modes, as this might take a significant portion of time. Overall, the timing for our method scales linearly with the size of the molecular structure and nonlinearly with the number of the computed normal modes. Regarding the other two methods, we can draw several observations. First of all, in terms of speed ProDy and iMod are very similar to each other despite the fact that one uses the RTB model in the Cartesian space, while the other uses model representation in the internal coordinates. Second, the performance of these two methods is almost independent of the number of requested modes. We should mention that iMod failed on the smallest chemoreceptor system outputting zero eigenvectors, so we removed these data from the plot. Finally, both methods failed with the segmentation fault on the largest system during the computation of the Hessian matrix. Therefore, we repeated the test removing all the RNA

chains from the ribosome molecule, such that the final structure contained only 90,587 atoms, but the two methods failed again. To conclude, if only a few normal modes are required (up to 100), then the sparse iterative scheme based on the Lanczos diagonalization algorithm seems to be advantageous over the other strategies. The difference becomes very significant for mid- to large-size systems starting at about 20,000 of atoms. On the other hand, if all the modes are required, then the dense diagonalization methods are much more effective. Finally, for molecular systems of a very large size starting from about 100,000 of atoms, only the sparse method implemented in NOLB completed the job. We should mention here that, of course, more aggressive coarse-graining schemes can be used for large systems such that dense diagonalization methods will be very efficient as well. Also, our test case is far from being exhaustive and more rigorous comparisons of different diagonalization techniques can be found elsewhere, for example in a recent study from the authors of iMod,[60] where they drew the same conclusions regarding the advantage of the iterative Krylov subspace techniques. Overall, this test demonstrated that modern NMA algorithms compute the slowest normal modes for mid-size molecular systems in a very reasonable time, typically in less than a minute, and in many cases these are computed in several seconds almost at the interactive rates.

Conclusion

In this work we have presented a conceptually simple and computationally efficient method for nonlinear normal mode analysis. It relies on the rotation-translation of rigid blocks theoretical basis developed by Y.-H. Sanejouand and colleagues[1,2]. We have demonstrated how to physically interpret the eigenvalues computed in the RTB basis in terms of angular and linear velocities applied to the rigid blocks and how to construct a nonlinear extrapolation of motion using these velocities. The key observation of our method is that the angular velocity of a rigid block can be interpreted as the result of an implicit applied force, such that the motion of the rigid block can be considered as a pure rotation about a certain center. In principle, our method is independent of the force-field. However, we have only tested it with the anisotropic network model. We demonstrated the motions produced with the NOLB method on three different molecular systems, and noted that some of the lowest frequency normal modes correspond to the biologically relevant motions, as has been reported in literature. For example, NOLB has detected the spiral sliding motion of the TALE proteins, which are capable of rapid diffusion along DNA. It has also detected the simultaneous rotation of the five terminase subunits in a pentameric assembly, which allows it to accommodate the guest DNA. We have compared our method with the standard approach in Cartesian coordinates and have shown that the NOLB NMA produces better structures, especially at large deformation amplitudes, as has been also confirmed by the MolProbity service validation. We have also demonstrated that our method is scalable and can be applied to very large molecular systems, such as ribosomes. In the future it will be interesting to study how the NOLB normal modes describe protein conformational changes compared to other approaches.[40] Standalone executables of the NOLB NMA are available at https://team.inria.fr/nano-d/software/nolb-normal-modes/. A graphical user interface created for the SAMSON software platform will be shortly available at https://www.samson-connect.net.

Acknowledgement The authors thank Yassine Naimi and Stephane Redon for the creation of a graphical user interface for the NOLB method inside the SAMSON software platform. The authors thank Yves-Henri Sanejouand from Université de Nantes and Osni Marques from Lawrence Berkeley National Laboratory for their advice with diagonalization techniques. Finally, the authors thank Leonard Jaillet and Guillaume Pages for helpful discussions during the preparation of the manuscript.

Supporting Information Available: Supplementary Information pdf file is available free

of charge. It contains 8 tables that list the energy and number of broken bonds for deformations along the lowest 10 normal modes for 4 tested molecular structures. This material is available free of charge via the Internet at http://pubs.acs.org/.

References

(1) Durand, P.; Trinquier, G.; Sanejouand, Y.-H. a New Approach for Determining Low-Frequency Normal Modes in Macromolecules. *Biopolymers* **1994**, *34*, 759–771.

(2) Tama, F.; Gadea, F. X.; Marques, O.; Sanejouand, Y.-H. Building-Block Approach for Determining Low-Frequency Normal Modes of Macromolecules. *Proteins: Struct., Funct., Bioinf.* **2000**, *41*, 1–7.

(3) Wilson, E. B.; Decius, J. C.; Cross, P. C. *Molecular Vibrations: The Theory of Infrared and Raman Spectra*; McGraw-Hill, 1955.

(4) Bahar, I.; Lezon, T. R.; Bakan, A.; Shrivastava, I. H. Normal Mode Analysis of Biomolecular Structures: Functional Mechanisms of Membrane Proteins. *Chem. Rev.* **2010**, *110*, 1463–1497.

(5) Williams, R. J. P. the Conformation Properties of Proteins in Solution. *Biol. Rev.* **1979**, *54*, 389–437.

(6) McCammon, J. A.; Karplus, M. the Dynamic Picture of Protein Structure. *Acc. Chem. Res.* **1983**, *16*, 187–193.

(7) Karplus, M.; McCammon, J. A. Dynamics of Proteins: Elements and Function. *Annu. Rev. Biochem.* **1983**, *52*, 263–300.

(8) Bahar, I.; Chennubhotla, C.; Tobi, D. Intrinsic Dynamics of Enzymes in the Unbound State and Relation to Allosteric Regulation. *Curr. Opin. Struct. Biol.* **2007**, *17*, 633–640.

(9) Amadei, A.; Linssen, A. B. M.; Berendsen, H. J. C. Essential Dynamics of Proteins. *Proteins: Struct., Funct., Genet.* **1993**, *17*, 412–425.

(10) Brooks, B.; Karplus, M. Harmonic Dynamics of Proteins: Normal Modes and Fluctuations in Bovine Pancreatic Trypsin Inhibitor. *Proc. Natl. Acad. Sci. U.S.A.* **1983**, *80*, 6571–6575.

(11) Levitt, M.; Sander, C.; Stern, P. S. the Normal Modes of a Protein: Native Bovine Pancreatic Trypsin Inhibitor. *Int. J. Quantum Chem.* **1983**, *24*, 181–199.

(12) Atilgan, A. R.; Durell, S. R.; Jernigan, R. L.; Demirel, M. C.; Keskin, O.; Bahar, I. Anisotropy of Fluctuation Dynamics of Proteins with an Elastic Network Model. *Biophys. J.* **2001**, *80*, 505–515.

(13) Hinsen, K. Analysis of Domain Motions by Approximate Normal Mode Calculations. *Proteins: Struct., Funct., Genet.* **1998**, *33*, 417–429.

(14) Tama, F.; Sanejouand, Y.-H. Conformational Change of Proteins Arising from Normal Mode Calculations. *Protein Eng.* **2001**, *14*, 1–6.

(15) Kim, M. K.; Jernigan, R. L.; Chirikjian, G. S. Efficient Generation of Feasible Pathways for Protein Conformational Transitions. *Biophys. J.* **2002**, *83*, 1620–1630.

(16) Xu, C.; Tobi, D.; Bahar, I. Allosteric Changes in Protein Structure Computed by a Simple Mechanical Model: Hemoglobin T R2 Transition. *J. Mol. Biol.* **2003**, *333*, 153–168.

(17) Zheng, W.; Brooks, B. R.; Thirumalai, D. Low-Frequency Normal Modes That Describe Allosteric Transitions in Biological Nanomachines Are Robust to Sequence Variations. *Proc. Natl. Acad. Sci. U.S.A.* **2006**, *103*, 7664–7669.

(18) Kirillova, S.; Cortés, J.; Stefaniu, A.; Siméon, T. An NMA-Guided Path Planning Approach for Computing Large-Amplitude Conformational Changes in Proteins. *Proteins: Struct., Funct., Bioinf.* **2008**, *70*, 131–143.

(19) Zacharias, M. Rapid Protein-Ligand Docking Using Soft Modes from Molecular Dynamics Simulations to Account for Protein Deformability: Binding of FK506 to FKBP. *Proteins: Struct., Funct., Bioinf.* **2004**, *54*, 759–767.

(20) Lindahl, E.; Delarue, M. Refinement of Docked Protein-Ligand and Protein-DNA Structures Using Low Frequency Normal Mode Amplitude Optimization. *Nucleic Acids Res.* **2005**, *33*, 4496–4506.

(21) May, A.; Zacharias, M. Accounting for Global Protein Deformability During Protein–Protein and Protein–ligand Docking. *Biochim. Biophys. Acta* **2005**, *1754*, 225–231.

(22) May, A.; Zacharias, M. Energy Minimization in Low-Frequency Normal Modes to Efficiently Allow for Global Flexibility During Systematic Protein-Protein Docking. *Proteins: Struct., Funct., Bioinf.* **2007**, *70*, 794–809.

(23) Dobbins, S. E.; Lesk, V. I.; Sternberg, M. J. E. Insights into Protein Flexibility: The Relationship Between Normal Modes and Conformational Change upon Protein-Protein Docking. *Proc. Natl. Acad. Sci. U.S.A.* **2008**, *105*, 10390–10395.

(24) Moal, I. H.; Bates, P. A. SwarmDock and the Use of Normal Modes in Protein-Protein Docking. *Int. J. Mol. Sci.* **2010**, *11*, 3623–3648.

(25) Ruvinsky, A. M.; Kirys, T.; Tuzikov, A. V.; Vakser, I. A. Structure Fluctuations and Conformational Changes in Protein Binding. *J. Bioinform. Comput. Biol.* **2012**, *10*, 1241002.

(26) Li, G.; Cui, Q. a Coarse-Grained Normal Mode Approach for Macromolecules: An Efficient Implementation and Application Ca^{2+}-ATPase. *Biophys. J.* **2002**, *83*, 2457–2474.

(27) Bastard, K.; Saladin, A.; Prévost, C. Accounting for Large Amplitude Protein Deformation During in Silico Macromolecular Docking. *Int. J. Mol. Sci.* **2011**, *12*, 1316–1333.

(28) Lensink, M. F. et al. Prediction of Homoprotein and Heteroprotein Complexes by Protein Docking and Template-Based Modeling: A CASP-CAPRI Experiment. *Proteins: Struct., Funct., Bioinf.* **2016**, *84*, 323–348.

(29) Idelsohn, S. R.; Cardona, A. a Load-Dependent Basis for Reduced Nonlinear Structural Dynamics. *Comput. Struct.* **1985**, *20*, 203–210.

(30) Shaw, S. W.; Pierre, C. Normal Modes for Non-Linear Vibratory Systems. *J. Sound Vib.* **1993**, *164*, 85–124.

(31) Pesheck, E.; Pierre, C.; Shaw, S. W. a New Galerkin-Based Approach for Accurate Non-Linear Normal Modes Through Invariant Manifolds. *J. Sound Vib.* **2002**, *249*, 971–993.

(32) Levitt, M. a Simplified Representation of Protein Conformations for Rapid Simulation of Protein Folding. *J. Mol. Biol.* **1976**, *104*, 59–107.

(33) Levitt, M.; Sander, C.; Stern, P. S. Protein Normal-Mode Dynamics: Trypsin Inhibitor, Crambin, Ribonuclease and Lysozyme. *J. Mol. Biol.* **1985**, *181*, 423–447.

(34) Kamiya, K.; Sugawara, Y.; Umeyama, H. Algorithm for Normal Mode Analysis with General Internal Coordinates. *J. Comput. Chem.* **2003**, *24*, 826–841.

(35) Lu, M.; Poon, B.; Ma, J. A New Method for Coarse-Grained Elastic Normal-Mode

Analysis. *J. Chem. Theory Comput.* **2006**, *2*, 464–471.

(36) Lopéz-Blanco, J. R.; Garzón, J. I.; Chacón, P. iMod: Multipurpose Normal Mode Analysis in Internal Coordinates. *Bioinformatics* **2011**, *27*, 2843–2850.

(37) Tirion, M.; ben Avraham, D.; Lorenz, M.; Holmes, K. Normal Modes As Refinement Parameters for the F-Actin Model. *Biophys. J.* **1995**, *68*, 5–12.

(38) Tirion, M. M. Large Amplitude Elastic Motions in Proteins from a Single-Parameter, Atomic Analysis. *Phys. Rev. Lett.* **1996**, *77*, 1905.

(39) Kovacs, J. A.; Cavasotto, C. N.; Abagyan, R. Conformational Sampling of Protein Flexibility in Generalized Coordinates: Application to Ligand Docking. *J. Comput. Theor. Nanosci.* **2005**, *2*, 354–361.

(40) Bray, J. K.; Weiss, D. R.; Levitt, M. Optimized Torsion-Angle Normal Modes Reproduce Conformational Changes More Accurately than Cartesian Modes. *Biophys. J.* **2011**, *101*, 2966–2969.

(41) Noguti, T.; Gõ, N. Dynamics of Native Globular Proteins in Terms of Dihedral Angles. *J. Phys. Soc. Jpn.* **1983**, *52*, 3283–3288.

(42) Schuyler, A. D.; Chirikjian, G. S. Efficient Determination of Low-Frequency Normal Modes of Large Protein Structures by Cluster-NMA. *J. Mol. Graphics Modell.* **2005**, *24*, 46–58.

(43) Ahmed, A.; Gohlke, H. Multiscale Modeling of Macromolecular Conformational Changes Combining Concepts from Rigidity and Elastic Network Theory. *Proteins: Struct., Funct., Bioinf.* **2006**, *63*, 1038–1051.

(44) Demerdash, O. N. A.; Mitchell, J. C. Density-Cluster NMA: A New Protein Decomposition Technique for Coarse-Grained Normal Mode Analysis. *Proteins: Struct., Funct., Bioinf.* **2012**, *80*, 1766–1779.

(45) Lezon, T. R.; Shrivastava, I. H.; Yang, Z.; Bahar, I. *Handbook on Biological Networks*; World Scientific Pub Co Pte Lt, 2009; pp 129–158.

(46) Schuyler, A. D.; Chirikjian, G. S. Normal Mode Analysis of Proteins: A Comparison of Rigid Cluster Modes with Cα Coarse Graining. *J. Mol. Graphics Modell.* **2004**, *22*, 183–193.

(47) Bahar, I.; Rader, A. J. Coarse-Grained Normal Mode Analysis in Structural Biology. *Curr. Opin. Struct. Biol.* **2005**, *15*, 586–592.

(48) Doruker, P.; Atilgan, A. R.; Bahar, I. Dynamics of Proteins Predicted by Molecular Dynamics Simulations and Analytical Approaches: Application to α-Amylase Inhibitor. *Proteins: Struct., Funct., Bioinf.* **2000**, *40*, 512–524.

(49) Fuglebakk, E.; Reuter, N.; Hinsen, K. Evaluation of Protein Elastic Network Models Based on an Analysis of Collective Motions. *J. Chem. Theory Comput.* **2013**, *9*, 5618–5628.

(50) Fuglebakk, E.; Tiwari, S. P.; Reuter, N. Comparing the Intrinsic Dynamics of Multiple Protein Structures Using Elastic Network Models. *Biochim. Biophys. Acta* **2015**, *1850*, 911–922.

(51) Artemova, S.; Grudinin, S.; Redon, S. a Comparison of Neighbor Search Algorithms for Large Rigid Molecules. *J. Comput. Chem.* **2011**, *32*, 2865–2877.

(52) Lawson, C. L.; Patwardhan, A.; Baker, M. L.; Hryc, C.; Garcia, E. S.; Hudson, B. P.; Lagerstedt, I.; Ludtke, S. J.; Pintilie, G.; Sala, R.; Westbrook, J. D.; Berman, H. M.; Kleywegt, G. J.; Chiu, W. EMDataBank Unified Data Resource for

3DEM. *Nucleic Acids Res.* **2016**, *44*, D396–D403.

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

(54) Cuculis, L.; Abil, Z.; Zhao, H.; Schroeder, C. M. Direct Observation of TALE Protein Dynamics Reveals a Two-State Search Mechanism. *Nat. Commun.* **2015**, *6*.

(55) Dauden, M. I.; Martin-Benito, J.; Sanchez-Ferrero, J. C.; Pulido-Cid, M.; Valpuesta, J. M.; Carrascosa, J. L. Large Terminase Conformational Change Induced by Connector Binding in Bacteriophage T7. *J. Biol. Chem.* **2013**, *288*, 16998–17007.

(56) MacKerell, A. D.; Banavali, N.; Foloppe, N. Development and Current Status of the CHARMM Force Field for Nucleic Acids. *Biopolymers* **2000**, *56*, 257–265.

(57) Wang, J.; Cieplak, P.; Kollman, P. A. How Well Does a Restrained Electrostatic Potential (RESP) Model Perform in Calculating Conformational Energies of Organic and Biological Molecules? *J. Comput. Chem.* **2000**, *21*, 1049–1074.

(58) Chen, V. B.; Arendall, W. B.; Headd, J. J.; Keedy, D. A.; Immormino, R. M.; Kapral, G. J.; Murray, L. W.; Richardson, J. S.; Richardson, D. C. MolProbity: All-Atom Structure Validation for Macromolecular Crystallography. *Acta Crystallogr., Sect. D: Biol. Crystallogr.* **2010**, *66*, 12–21.

(59) Bakan, A.; Meireles, L. M.; Bahar, I. ProDy: Protein Dynamics Inferred from Theory and Experiments. *Bioinformatics* **2011**, *27*, 1575–1577.

(60) López-Blanco, J. R.; Reyes, R.; Aliaga, J. I.; Badia, R. M.; Chacón, P.; Quintana-Ortí, E. S. Exploring Large Macromolecular Functional Motions on Clusters of Multicore Processors. *J. Comput. Phys.* **2013**, *246*, 275–288.