Overview & Applications

T. Lezon

Hands-on Workshop in Computational Biophysics
Pittsburgh Supercomputing Center
22 May, 2014
Simulations still take time

Coarse-grained Elastic Network Models are fast


Lane et al. 2013
The structural data explosion

Multiple structures for a single sequence

Dynamics may be inferred from structural data.

Exploiting the PDB since 2010

- High-throughput analysis of structural data
- Application Programming Interface (API) for development of tools
- Suitable for interactive usage

**ProDy** identifies, retrieves, aligns, and analyzes (PCA) structures matching input sequence

Bakan, Meireles & Bahar. Bioinformatics
An Interactive Tool

Languages
- Python: 80%
- C: 13%
- Other: 7%

IP[y]: IPython Interactive Computing

ProDy
Protein Dynamics & Sequence Analysis

matplotlib
NumPy

ProDy, updated May 20, 2014
more at Ohloh
An evolving suite of tools

ProDy
Protein Dynamics & Sequence Analysis

Evol
Bridging Evolution & Dynamics

DruGUI
Druggability Suite for VMD

NMWiz
Normal Mode Wizard for VMD

Principal Component Analysis
Elastic Network Models
Normal Mode Analysis
Trajectory Analysis

Multiple Sequence Alignment
Correlated Mutation Analysis
Structural Evolution

Computational Drug Discovery
Binding Site Prediction
Affinity Estimation

Call ProDy from VMD
Normal Mode Visualization
Elastic Network Model

- Useful for finding global equilibrium motions of proteins
- Employs harmonic potential about native state
- Coarse-grained (Cα-only description)
- Residue pairs are connected via springs
- Normal modes are found analytically
Elastic Network Model

- Useful for finding global equilibrium motions of proteins
- Employs harmonic potential about native state
- Coarse-grained (Cα-only description)
- Residue pairs are connected via springs
- Normal modes are found analytically
Elastic Network Model

- Useful for finding global equilibrium motions of proteins
- Employs harmonic potential about native state
- Coarse-grained (Cα-only description)
- Residue pairs are connected via springs
- Normal modes are found analytically
Approximating the protein free energy landscape

Rough and funnel-shaped energy landscape that is function of atomic coordinates

\[ V(\mathbf{r}) = V(\mathbf{r}^0) + \sum_i \left( \frac{\partial V}{\partial r_i} \right)_{\mathbf{r}^0} \Delta r_i + \frac{1}{2} \sum_{ij} \left( \frac{\partial^2 V}{\partial r_i \partial r_j} \right)_{\mathbf{r}^0} \Delta r_i \Delta r_j + \frac{1}{6} \sum_{ijk} \left( \frac{\partial^3 V}{\partial r_i \partial r_j \partial r_k} \right)_{\mathbf{r}^0} \Delta r_i \Delta r_j \Delta r_k + ... \]

\[ \mathbf{r} = (x_1, y_1, z_1, \ldots, x_N, y_N, z_N)^T \]

\[ \Delta \mathbf{r} = \mathbf{r} - \mathbf{r}^0 \]

\[ V(\mathbf{r}) \approx \frac{1}{2} \sum_j \left( \frac{\partial^2 V}{\partial r_i \partial r_j} \right)_{\mathbf{r}^0} \Delta r_i \Delta r_j \]

\[ = \frac{1}{2} \Delta \mathbf{r}^T \mathbf{H} \Delta \mathbf{r} \]
Equations of Motion

\[ \mathbf{M} \Delta \ddot{\mathbf{r}} = -\mathcal{H} \Delta \mathbf{r} \]

\[
\begin{align*}
\mathbf{q} &= \mathbf{M}^{1/2} \Delta \mathbf{r} \\
\mathbf{H} &= \mathbf{M}^{-1/2} \mathcal{H} \mathbf{M}^{-1/2}
\end{align*}
\]

\[
\ddot{\mathbf{q}} = -\mathbf{M}^{-1/2} \mathcal{H} \mathbf{M}^{-1/2} \mathbf{q}
\]

\[
= -\mathbf{H} \mathbf{q}
\]

\[
\begin{align*}
\mathbf{q}(t) &= \mathbf{q} e^{-i \omega t} \\
\dot{\mathbf{q}}(t) &= -\omega^2 \mathbf{q}(t)
\end{align*}
\]

\[ \mathbf{H} = \mathbf{V} \Lambda \mathbf{V} \]

\[
\Lambda_{ij} = \omega_i^2 \delta_{ij}
\]

Newton

Mass-weighted coordinates

Mass-weighted Hessian

Oscillatory solution
Anisotropic Network Model

\[ V(\mathbf{r}) = \frac{\gamma}{2} \sum_{i=1}^{N} \sum_{j>i} \left( |\mathbf{r}_{ij}| - |\mathbf{r}_{ij}^0| \right)^2 \Theta \left( R_c - |\mathbf{r}_{ij}^0| \right) \]

Harmonic Step function

\[
\left( \frac{\partial^2 V}{\partial x_i \partial y_j} \right)_{\mathbf{r}^0} = -\frac{x_i^0 y_j^0}{|\mathbf{r}_{ij}^0|^2}
\]

Hessian is calculated directly from structure

\[
\mathbf{H}_{ij} = -\frac{\gamma}{(R_{ij}^0)^2} \begin{bmatrix}
    (x_{ij}^0)^2 & x_{ij}^0 y_{ij}^0 & x_{ij}^0 z_{ij}^0 \\
    x_{ij}^0 y_{ij}^0 & (y_{ij}^0)^2 & y_{ij}^0 z_{ij}^0 \\
    x_{ij}^0 z_{ij}^0 & y_{ij}^0 z_{ij}^0 & (z_{ij}^0)^2
\end{bmatrix}
\]

ENMs are flexible!

• Selection of the ENM force constants
  • Cutoff distance
  • Functional form
• Changing the potential directly
• Adding rigidity
• Including external elements, or looking at only a subsystem
• Altering network topology
Flexible force constants

\[ \gamma(r_{ij}) = \begin{cases} \gamma & |r_{ij}| < R_c \\ 0 & \text{otherwise} \end{cases} \]


\[ \gamma(r_{ij}) = \gamma_0 \exp \left( -\frac{|r_{ij}|^2}{r_0^2} \right) \]


\[ \gamma(r_{ij}) = \begin{cases} c_0 + c_1 |r_{ij}| & r_{ij} < R_0 \\ c_2 |r_{ij}|^{-6} & \text{otherwise} \end{cases} \]


\[ \gamma(r_{ij}) = |r_{ij}|^{-2} \]

Yang et al. PNAS 106 (2009).
Optimizing force constants

• Download NMR structures from PDB
• Calculate residue MSFs for each protein
• Assign ENM topology
• Optimize force constants to reproduce structural dynamics
• Search for trends in force constant values with structure

fetchPDB()
calcMSF()
buildHessian()
Flexible force constants

Fine-tuning force constants

- Distance-dependence
- 1st neighbors
- 2nd neighbors
- H bonds

Learn more at prody.csb.pitt.edu
Improving correlation between ENM and B factors

Possible solution: Add crystal contacts

“Tip Effect”
Explicit membrane models

As the environment fluctuates randomly, the effective motion of the system is given by

\[ V_{\text{eff}} (s) = \frac{1}{2} \Delta s^T \left( H_{ss}^{-1} \right) \Delta s \]

\[ H_{ss}^{-1} = H_{ss} - H_{SE} \left( H_{EE} \right)^{-1} H_{ES} \]

learn more at prody.csb.pitt.edu
Improving correlation between ENM and B factors

Crystal contacts help, but do not eliminate tip effect. What about lattice vibrations?
Global transitions

Outward Facing (OF)  Inward Facing (IF)

Global transitions

Single subunit showing the transport domain moving across the membrane

Rotations-Translations of Blocks

H_{RTB} = P \times H^{AA} \times P^T

Smaller Hessian can be more easily diagonalized...

H: ANM Hessian (3 rows/cols per residue)
P: Projection matrix from all-residue space to rigid block space
H^{RTB}: RTB Hessian (no internal motions of blocks)
V'^{AA}: Approximate ANM motions

...and modes projected back into all-residue space

RTB.buildHessian()
Exploring structural transitions: Glutamate transporter

ANM predicts large radial motions of the trimer. Can we invent a better model?

$$H_{ij} = -\frac{\gamma}{(R_{ij}^0)^2} \begin{bmatrix}
(x_i^0)^2 & x_i^0 y_i^0 & x_i^0 z_i^0 \\
x_i^0 y_i^0 & (y_i^0)^2 & y_i^0 z_i^0 \\
x_i^0 z_i^0 & y_i^0 z_i^0 & (z_i^0)^2
\end{bmatrix}$$

Altered radial force constants:

$$H_{ij} = -(R_{ij}^0)^{-2} \begin{bmatrix}
(x_{ij}^0 \sqrt{Y_x})^2 & x_{ij}^0 y_{ij}^0 \sqrt{Y_x Y_y} & x_{ij}^0 z_{ij}^0 \sqrt{Y_x Y_z} \\
x_{ij}^0 y_{ij}^0 \sqrt{Y_x Y_y} & (y_{ij}^0 \sqrt{Y_y})^2 & y_{ij}^0 z_{ij}^0 \sqrt{Y_y Y_z} \\
x_{ij}^0 z_{ij}^0 \sqrt{Y_x Y_z} & y_{ij}^0 z_{ij}^0 \sqrt{Y_y Y_z} & (z_{ij}^0 \sqrt{Y_z})^2
\end{bmatrix}$$

$$H_{ij} = -\frac{\gamma}{(R_{ij}^0)^2} \begin{bmatrix}
(x_i^0)^2 & x_i^0 y_i^0 & c x_i^0 z_i^0 \\
x_i^0 y_i^0 & (y_i^0)^2 & c y_i^0 z_i^0 \\
c x_i^0 z_i^0 & c y_i^0 z_i^0 & (c z_i^0)^2
\end{bmatrix}$$

Exploring structural transitions: Glutamate transporter

ANM: Large radial motions

imANM
Explicit membrane models

As the environment fluctuates randomly, the effective motion of the system is given by

\[ V_{\text{eff}}(s) = \frac{1}{2} \Delta s^T \left( H^{ss}\right)^T \Delta s \]

\[ H^{ss}^{-1} = H^{ss} - H^{SE} (H^{EE})^{-1} H^{ES} \]
Thinking bigger: NPC


ENM of NPC

NPC Model

456 proteins:
57 proteins/spoke
30 unique proteins

gammaVariableCutoff()
reduceModel()
parseSparseMatrix()


\[ R_c = \begin{cases} 
12 \text{nm protein-protein} \\
8 \text{nm NE-NE} \\
10 \text{nm protein-NE} 
\end{cases} \]

\[ \gamma = \begin{cases} 
1.0 \text{ protein-protein} \\
10 \text{ NE-NE} 
\end{cases} \]

NE: 2070 discrete points
NPC slow modes

Mode 1

Mode 2

Scallop MS motion

protein axial position

29
## Tutorials

<table>
<thead>
<tr>
<th>Tutorials</th>
<th>Description</th>
<th>Go to Tutorial</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ProDy</strong></td>
<td>Learn how to use ProDy from the introductory ProDy tutorial or from the comprehensive API reference manual.</td>
<td></td>
</tr>
<tr>
<td><strong>Normal Mode Wizard</strong></td>
<td>Learn how to depict normal modes and generate animations of protein dynamics along them with NMeiz.</td>
<td></td>
</tr>
<tr>
<td><strong>Evol</strong></td>
<td>Learn how to identify conserved and coevolved residues and characterizing their dynamical properties.</td>
<td></td>
</tr>
<tr>
<td><strong>Drugability Suite</strong></td>
<td>Learn how to setup and analyze drugability simulations containing small organic molecules using DrugUI.</td>
<td></td>
</tr>
<tr>
<td><strong>Elastic Network Models</strong></td>
<td>Learn how to perform normal mode analysis and developing customized force constant functions.</td>
<td></td>
</tr>
<tr>
<td><strong>Ensemble Analysis</strong></td>
<td>Learn how to analyze large and heterogeneous ensembles of protein structures to infer dynamical properties.</td>
<td></td>
</tr>
<tr>
<td><strong>Structure Analysis</strong></td>
<td>Learn how to compare and align structures, identify ligand contacts, and extract ligands from PDB files.</td>
<td></td>
</tr>
<tr>
<td><strong>Trajectory Analysis</strong></td>
<td>Learn how to analyze simulation trajectories, in particular handling large trajectory files that don’t fit in memory.</td>
<td></td>
</tr>
<tr>
<td><strong>Conformational Sampling</strong></td>
<td>Learn how to generate alternate protein conformations along AMM modes and to refine them using NAMD.</td>
<td></td>
</tr>
</tbody>
</table>