Computational Structural Biology

Jarrod A. Smith <jarrod.smith@vanderbilt.edu>  Biochemistry 300
High Resolution Structural Biology

NMR Spectroscopy  ⇔  X-ray Crystallography

Computation

Determine experimentally or model 3D structures of biomolecules
Computational Structural Biology Tools

• Structure calculations
  • Building Xray and NMR structures from experimental data

• Protein sequence analysis
  • Pairwise alignments
  • Multiple sequence alignments
  • Secondary structure prediction

• Protein tertiary structure prediction
  • Protein folding
  • Homology modeling
  • Fold recognition

• Molecular mechanics

• Data visualization
  • Solvent Accessibility
  • Electrostatics
  • Mutations
Protein sequence analysis

• Pairwise sequence alignment
  • Frequently used to search for sequence similarity to a target - e.g. BLASTp
  • Not very robust for comparing sequences with little (but still structurally or functionally relevant) sequence similarity.

• Multiple sequence alignment
  • Suited for in-depth analysis of conserved positions
  • With many simultaneous sequences, one can make reliable conclusions about structure/function.
    • Conserved residues/segments
    • Loop regions in the corresponding structures
    • Consensus sequences that define functional domains
Secondary structure prediction

- Predict which regions of the sequence are α-helix, β-sheet, or coil
- Empirical methods: based on probability of local secondary structure statistically compiled from known structures
- Geometric/Pattern recognition methods:
  - Helices tend to have hydrophobic residues at i, i+3, i+4, i+7
  - Sheets tend to have hydrophobic residues at i, i+2, i+4, i+6, etc. and polar residues at i+1, i+3, i+5, etc.
Accuracy of secondary structure prediction

- Best results are obtained if with a sequence alignment followed by application of several different methods on all the sequences to get a consensus.
  - PSI-pred, JPRED, DSC, PHD, ZPRED, nnPredict, BMERC, SSP
- One can expect up to ~70% accuracy if using the latest tools and multiple-sequence alignments
- Up to ~80% accuracy in combination with tertiary structure prediction
Protein tertiary structure prediction: Why attempt it?

- **Speed:** To help keep up with the growing pool of sequenced genes that have unknown function and/or structure
  - Potential for high-throughput: Structural Genomics/Proteomics

- **Utility:** Pick up the slack where Xray and NMR technologies are limited
  - Many important proteins do not crystallize
  - Practical size limitations with NMR (60-80Kd?)
1968: The protein folding problem

• Given a protein’s AA sequence, what is its 3-dimensional fold, and how does it get there?

• Cyrus Levinthal’s paradox of protein folding (1968):

  Let each amino acid in a 100-residue protein have only 10 possible conformations. This leads to $10^{100}$ possible conformations to be searched by the protein. Given average bond rotation frequencies, approximately $10^{14}$ conformations/second can be explored. It will take $10^{86}$ seconds, or $10^{78}$ years for this protein to examine all possible conformations.

• Clearly this is not how proteins really fold!
2005: The protein folding computational “grand challenge”

- In 1999, IBM Research announced the “Blue Gene” project. A $100M project to be completed by 2005.
  Build a 1 petaflop supercomputer (a million billion floating point operations per second) to work on the protein folding problem. Fastest supercomputer in the world as of Nov. 2004 is 71 teraflops, or 0.071 petaflop.

- Still, this computer will “only” do $10^{15}$ operations per second.

- Clearly brute-force conformational searches are not how to go about predicting protein structure computationally, either.
Protein tertiary structure prediction methods

1. QQYTA KIKGR
2. TFRNE KELRD
3. FIEKF KGR

Algorithm

- Ab-initio protein folding – “The Holy Grail”
- Homology modeling
- Fold recognition/threading
Ab-Initio protein folding

• Find the tertiary structure of a protein from the primary sequence using an algorithm based on first principles of protein chemistry

• General strategy is to devise a method to “score” (energy function) protein conformations and then search for the conformation with the best score (lowest energy).

• Both the “scoring problem” and the “sampling problem” are difficult problems to solve
The conformational sampling problem in ab-initio protein folding

- Seemingly hopelessly large space to search
- Cleverly choose the conformations you will sample
  - Lattice models
  - Probabilistic maps based on real structures
  - Dead End Elimination
  - Fragment Replacement

Alanine dipeptide probability map
The scoring problem in ab-initio protein folding

- Scoring needs to be fast (lots of conformations to test!)
- Scoring needs to be accurate
- Typical strategy: score in two (or more) stages

1. Conformational search
   - 10^9 Conformers
   - Fast scoring: 10^2/s per CPU
   - 10^4 Conformers
   - Decoys + native-like
   - 11 days on 10 CPUs

2. Decoys + native-like
   - 10^4 Conformers
   - Accurate scoring: 10/hr per CPU
   - Native-like
   - 4 days on 10 CPUs

3. Conformational search
   - 10^9 Conformers
   - Accurate scoring: 10/hr per CPU
   - Native-like
   - 11 millennia on 10 CPUs
Scoring criteria in ab-initio protein folding

- Statistical criteria
  - Clustering

- Geometric criteria
  - Helix-helix packing arrangement
  - Helix-sheet packing arrangement

- Thermodynamic criteria
  - Buried hydrophobic surface area
  - Solvent accessible surface area
  - Density
  - Electrostatic energy

- Empirical criteria
  - Pair potentials - statistical analysis of pairwise peptide-peptide contacts in real structures
  - Fitting to unassigned NMR data
Do ab-initio methods work?

• CASP – Critical Assessment of techniques for protein Structure Prediction
  • Bi-annual, large-scale experiment to blind-test current prediction techniques
  • X-ray and NMR community is called upon to submit sequences for which structures are about to be released
  • Structure prediction community builds models of them by cutoff date
  • After cutoff date, the real structures are submitted to PDB and the predictions are compared

- A few successes, but the majority of ab-initio predictions were wrong.
- The successes were in the 6-9Å RMSD range for Ca trace.
- CASP5 (2002) - 3.5Å RMSD

**T087 - PPase (Domain 2: 202-307)**

Native

Model 3

**RMSD = 6.2 Å (85 Cα)**

PDB code 1I74
pyrophosphatase from *Streptococcus Mutans*
Ab-intio prediction summary

• Most folds come out wrong - overcome by generating more and testing with experimental data!

• Sidechain info often not included

• Even the most correct models are not suitable for detailed work like drug design or docking studies.

• Still too costly and difficult to be a practical methodology for structure determination, but progress is being made.
Homology or “comparitive” modeling

• This methodology takes advantage of the fact that evolutionarily related proteins (high sequence similarity) often have very similar tertiary structures

• Core framework is often the same
  • The core of highly sequence-homologous proteins often share similar secondary structure elements packed into the same overall tertiary fold

• Loop regions may be different
  • The loops that interconnect the core secondary structure elements can vary widely even among closely homologous proteins
Steps to building a homology model

- Search for homologous proteins of known 3D structure
  - Run a BLAST search against all sequences in the PDB
  - Assess the BLAST hits for degree of homology. 35% or greater sequence homology is usually sufficient for a reasonably reliable result
  - The more known structures with homologous sequences, the better

- Build a multiple sequence alignment using the sequences of the known structure(s) and your sequence of interest
  - HMMer, SAM, ClustalW, MSA, AMPS
  - If pre-made alignments exist for a particular family of proteins, use them. They are often hand-edited by experts.
Steps to building a homology model (cont.)

- Construct a framework model from the sequence alignment, typically by copying main chain and side chain coordinates for the homologous residues from the parent structure(s).

- Build in the side chains for non-homologous residues
  - Sidechain rotamer libraries coupled with molecular mechanics energies (energy minimization).

- Build in the main chain and side chain atoms for sites where insertions/deletions are present in the sequence alignment

- Commonly used software - Swissmodel, Modeller, Homology (InsightII), SCWRL, MOE

- Model improvement
  - Experimental Data?
  - Energy minimization/Molecular dynamics
Accuracy of homology models
Protein fold recognition (AKA “threading”)

• By combining secondary structure prediction with fold recognition, there is hope to find the fold for a protein that has no significant homology to known structures.

• Proteins often adopt similar folds, even though they share no significant sequence or functional similarities.
  • Nature may be limited to a finite number of functional folds under biologically relevant conditions

• Fold recognition turns the protein folding problem around…
  • Instead of “Given this sequence, what’s the best structure”, fold recognition asks, “Given these known structures, which one does my sequence fit the best?”
Protein fold recognition

- Thread sequence (with or without secondary structure prediction data) through known structures and find the best match
- Good match == buried hydrophobic residues and exposed polar residues, good agreement with secondary structure prediction, and low pair potentials (hydrophobic-hydrophobic vs. charge-hydrophobic interactions)

Sequence:

```
1  IQRGH YTERK AAELT RTIVG
21  VVEAC HSLGV MHRDL ENFLF
41  VSKHE DSLLK TIDFG LSMFF
61  KPDDV FTDVV GSPYY VAPEV
81  LRKRY GPEAD VWSAG VIVYI
101 LLSGV PPFWA ETEQG IFE
```
Success of protein fold recognition?

- The best success rates have been between 40%-50%
- Even when fold family has been correctly identified, the sequence is likely to be “out of register” with the fold
- Many folds have close relatives that are impossible to distinguish between at this level
- You might have a new fold
- This type of methodology keeps getting better as more new folds are discovered
**Molecular Mechanics**

- Molecular mechanics is a methodology used to model molecules using a classical, physical description of bonds, angles, etc.

- Simple equations are coupled with empirically determined/validated parameters to describe the energy cost of deviating from ideal geometry – **MM force field**

- Some common MM applications
  - Energy minimization - a method for finding the nearest local energy minimum given a starting conformation.
  - Molecular dynamics - a method for simulating molecular motions by coupling Newton’s equation of motion (classical mechanics) with an MM force field.
  - Simulated annealing - Search 3-space for global energy minimum
  - AMBER, CHARMM, Discover, Gromos, NAMD, Xplor
Molecular Mechanics Force Fields

- A force field is a combination of force constants, parameters, and an energy function that are used together to describe the energy of a molecule.
- The force field parameters are empirically or computationally determined from small model compounds (individual amino acids and nucleotides in the case of packages designed to model biomolecules).
- The two most useful force fields for simulations of biomolecules are CHARMm (Harvard) and AMBER (UCSF).
A molecular mechanics potential energy function (force field)

\[
V_{\text{total}} = \sum_{\text{bonds}} K_r (r - r_{eq})^2 + \sum_{\text{angles}} K_\theta (\theta - \theta_{eq})^2
+ \sum_{\text{dihedrals}} K_\phi \left(1 + \cos(n \phi)\right)
+ \sum_{i<j} \left[ \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^6} \right] \quad \text{Van der Waals}
+ \sum_{i<j} \left[ \frac{q_i q_j}{\varepsilon r_{ij}} \right] \quad \text{Electrostatic}
\]
Bond term

$$\sum_{bonds} K_r (r - r_{eq})^2$$
Angle term

\[ \sum K_\theta (\theta - \theta_{eq})^2 \]
Dihedral term

\[ \sum_{\text{dihedrals}} K_\phi (1 + \cos(n\phi)) \]
Non-bonded terms (van der Waals)

\[
\sum_{i<j} \left[ \frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} \right]
\]
Non-bonded terms (electrostatics)

- Coulomb’s law with a cutoff is the simplest option
  \[
  \sum_{i<j} \frac{q_i q_j}{\varepsilon r_{ij}}
  \]

- Many advances have been made in this area
  - Long-range electrostatics in periodic systems (solvent) are important, but significantly raising the coulomb cutoff is computationally quite expensive
  - Particle Mesh Ewald (PME) summation (AMBER & CHARMm)
  - Generalized Born continuum solvent model (AMBER)
  - Fast Multipole Method (CHARMm)
  - Extended Electrostatics Model (CHARMm)
Energy Minimization

• The most basic application of MM

• Given a set of starting conditions (coords + force field), travel to the nearest local energy minimum
  • Adding hydrogens to a crystal structure
  • “Removing” kinetic energy from snapshots of an MD simulation for use as starting coordinates elsewhere
  • Quick clean-up during interactive model building

• Two popular algorithms:
  • Steepest descent - very robust
  • Conjugate gradient - better convergence
**Molecular Dynamics (MD) Simulations**

- One of the principle tools for studying time-dependent behavior of biological molecules
- Widely used to investigate the structure, dynamics, and thermodynamics of proteins and nucleic acids
  - Conformational substates
  - Thermodynamic properties
  - Protein folding pathways
- Used in NMR and Xray structure determination
  - Simulated annealing
How does MD work?

• MD simulates the atomic motions of a molecule using classical mechanics

• Atoms are modeled as point charges with mass $m$ under the influence of the force field (energy function), $V$

• Newton’s equation of motion is integrated at each step of the simulation for each atom in the system to determine their positions, $r$

\[
F = m_i a = m_i \frac{d^2 r_i}{dt^2}
\]

\[
F = - \frac{dV}{dr_i}
\]

\[
- \frac{dV}{dr_i} = m_i \frac{d^2 r_i}{dt^2}
\]
Time scale for molecular motions of interest

- Short-range motions (0.01 to 5 Å, 10^{-15} to 10^{-1} s)
  - Bond stretching, angle bending
  - Sidechain motions
  - Loop motions

- Rigid body motions (1 to 10 Å, 10^{-9} to 1 s)
  - Helix/Domain motions
  - Subunit motions

- Long-range motions (>10 Å, >10^{-6} s)
  - Binding events
  - Folding/unfolding
Achievable simulation time scales for molecular dynamics

- Complete simulation of protein/nucleic acid motions requires time steps small enough to capture the fastest motions (bond and angle vibrations)
- One time step typically equals $10^{-15}$ s simulation time
- Observing the slower motions (e.g. domain motions and folding events) therefore takes a lot of steps
- Practical considerations
  - Typical protein (200 AA with solvent) on a typical CPU takes about 1s of real time to compute 1 fs simulation time
  - This currently limits MD simulations to the nanosecond range for typical protein simulations (one month of computer time for ~4ns)
  - Up to microsecond range for small peptide simulations
Microscopic to macroscopic: statistical mechanics

• MD simulations generate microscopic information (atomic positions and velocities)

• One often wants to explore macroscopic properties of a system using these microscopic simulations
  • Changes in binding free energy of a drug candidate
  • Thermodynamic properties of conformational substates

• Relating this microscopic information to macroscopic observables requires statistical mechanics
Ensemble averaging

• An experiment measures a bulk property of the sample over a large ensemble of molecules, each one in its own conformation, energy state, etc.

• Statistical mechanics describes experimental observables in terms of a weighted average over the entire ensemble present in the sample.

• In MD there is no ensemble - there is only one molecule.
Time averaging

- The Ergodic hypothesis of statistical mechanics states that
  \[ \langle A \rangle_{\text{ensemble}} = \langle A \rangle_{\text{time}} \]

- If you let a single molecule evolve indefinitely, it will eventually explore all the states present instantaneously in an ensemble.

- If you are careful to run MD long enough, you can collect snapshots of the simulation over time and calculate bulk properties by averaging over them.