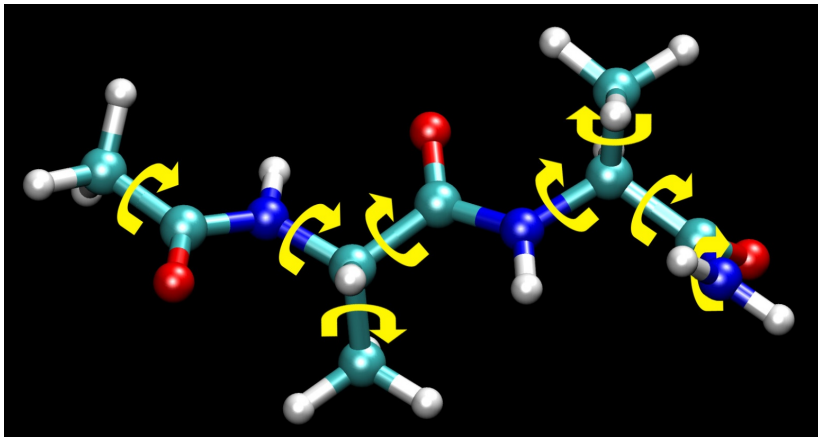


CS612 - Algorithms in Bioinformatics

Sampling

May 1, 2023

From a Rigid Ligand to a Flexible Ligand



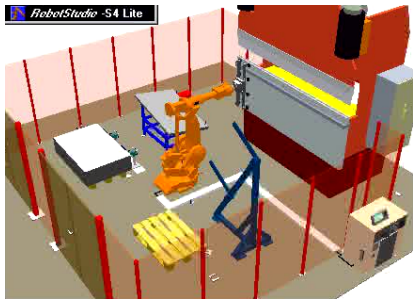
Torsional (Dihedral) Degrees of Freedom (DOF)

Kinematics

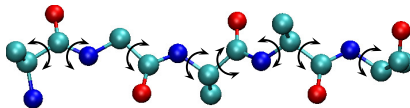
- Kinematics is a branch of classical mechanics that describes the motion of points, bodies (objects), and systems of bodies (groups of objects) without considering the forces that caused the motion.
- A kinematics problem begins by describing the geometry of a system and the initial conditions of any known values of position, velocity and/or acceleration of points in the system.
- Then, using geometric methods, the position, velocity and acceleration of any unknown parts of the system can be determined.
- Forward kinematics is the use of the kinematic equations of a robot to compute the position of the end-effector from specified values for the joint parameters.
- In protein motion, the problem becomes computing the new locations of the atoms given a set of dihedral rotations.

Robotics-inspired Approach to Protein Flexibility

- Similarity between proteins and robots: exploration of complex high-dimensional space
- Similarity exploited to sample conformations with spatial constraints



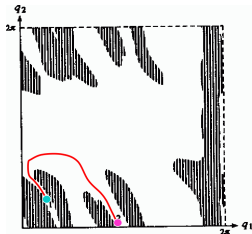
Articulated manipulator



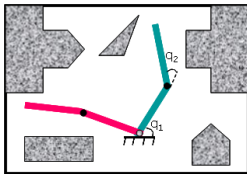
Protein Extended Backbone

Robotics-inspired Approach to Protein Flexibility

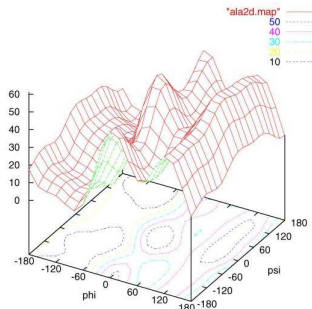
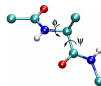
- Exploration of protein conformational space has parallels in robotics
- 0/1 collisions for robots versus energy field for proteins



adapted from J.-C.
Latombe, Stanford



adapted from P. Smith,
KSU

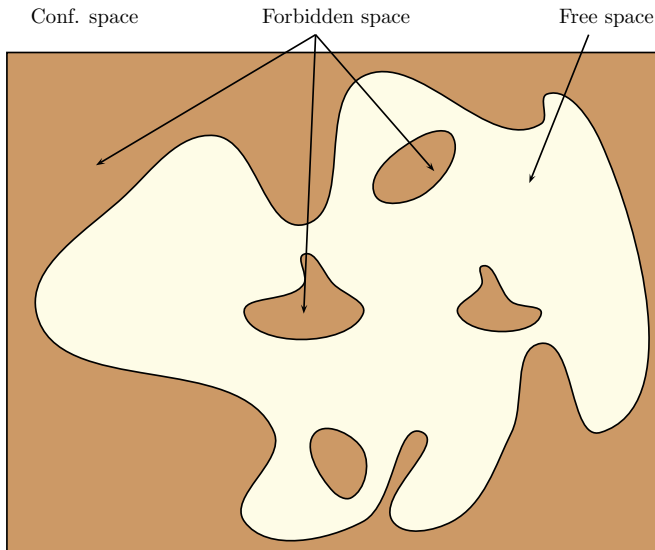


Robotics-inspired Approach to Protein Flexibility

- Dimensionality of configuration space
 - DOFs (rigid-body transformations and DOFs of the ligand)
 - Too many DOFs mean that the configuration space of the ligand is high-dimensional and difficult to search
 - Similar issue when planning motions for an articulated robotic chain in a cluttered environment
- Geometric complexity of the free space
 - Difficult to determine whether a ligand conformation and specific position and orientation result in a good fit
 - Similar issue for an articulated robot

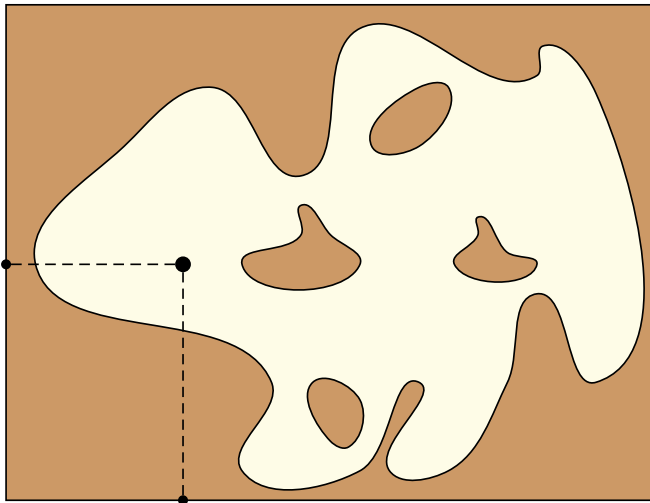
Address: Plan motions in the configuration space but compute in workspace (protein surface or cavity)!

Probabilistic Roadmap Motion Planning (PRM)



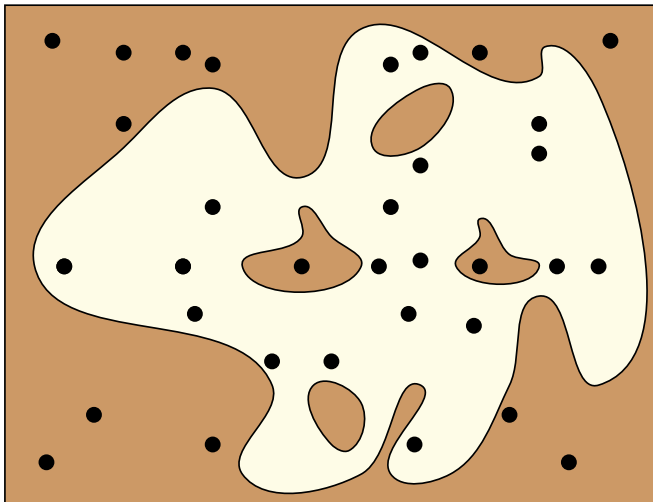
Probabilistic Roadmap Motion Planning (PRM)

Configurations are sampled by picking coordinates at random



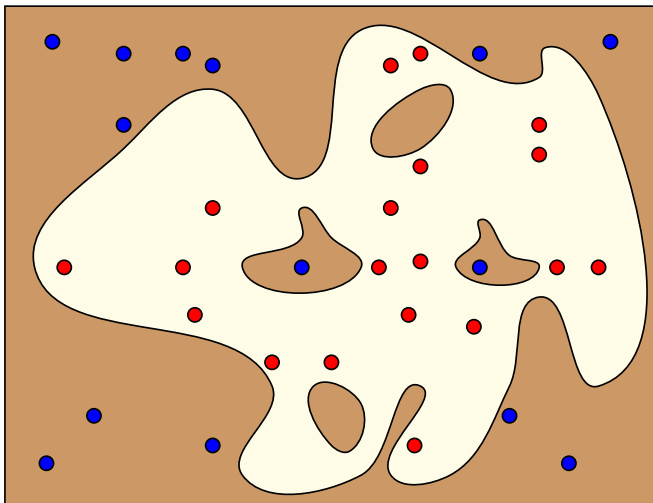
Probabilistic Roadmap Motion Planning (PRM)

Configurations are sampled by picking coordinates at random



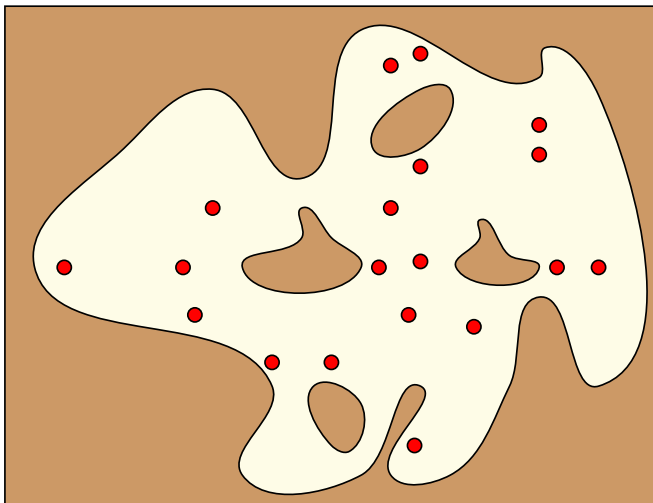
Probabilistic Roadmap Motion Planning (PRM)

Sampled configurations are tested for collision (in workspace!)



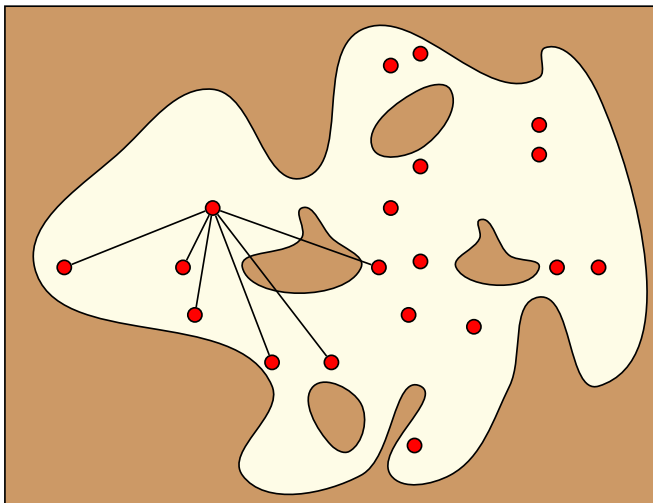
Probabilistic Roadmap Motion Planning (PRM)

The collision-free configurations are retained as “milestones”



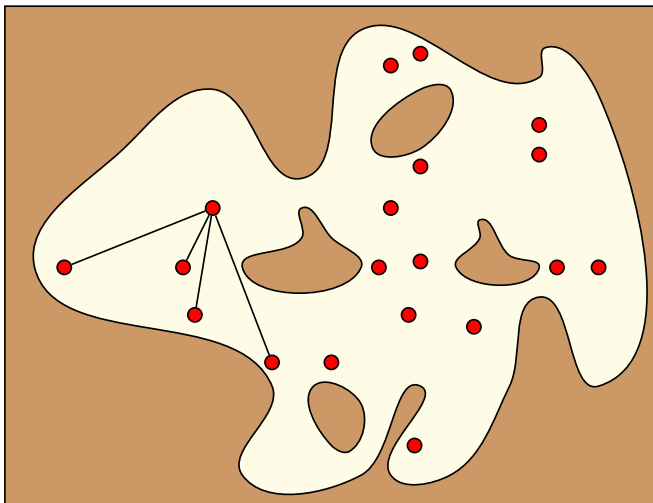
Probabilistic Roadmap Motion Planning (PRM)

Each milestone is linked by straight paths to its k-nearest neighbors



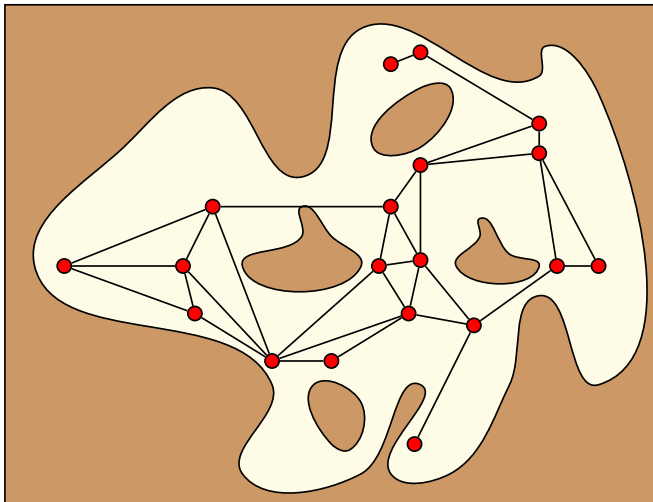
Probabilistic Roadmap Motion Planning (PRM)

Each milestone is linked by straight paths to its k-nearest neighbors



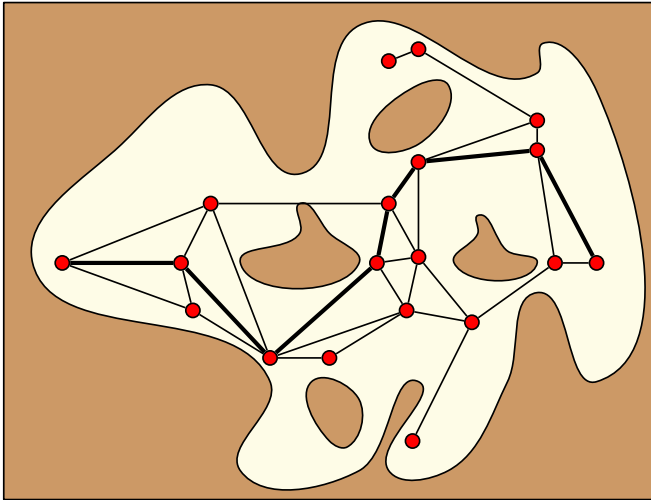
Probabilistic Roadmap Motion Planning (PRM)

The collision-free links are retained to form the PRM



Probabilistic Roadmap Motion Planning (PRM)

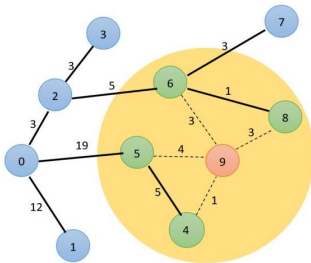
Finding paths in the map.



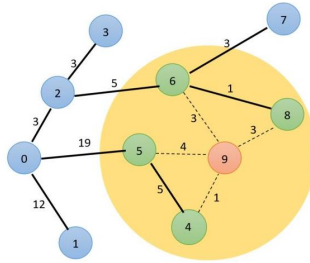
Application – RRT

- Construct a tree rooted at the start configuration by using random samples from the search space.
- Create a new sample q_r randomly; in our case, a new molecular conformation.
- Find the sample's nearest neighbor q_{near} on the tree.
- Try to connect the nearest neighbor to the new sample by an edge. In our case, we incrementally rotate the neighbor's relevant backbone dihedral angles in the direction of the new conformation.
- Stop when hitting a high energy barrier or when the newly created sample is reached. The stopping point is q_{new} .
- Add q_{new} to the tree with q_{near} as its parent.
- Stop when the algorithm reaches the distance within the set threshold from the target.

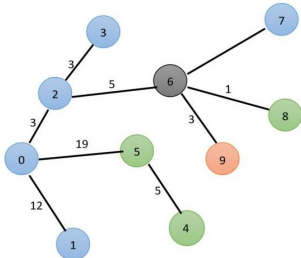
- For every node, the tree distance from the root is recorded.
- After the nearest neighbor is found on the tree, a neighborhood of vertices in a fixed radius from the new node is examined, and if a node with a cheaper overall cost is found, it replaces the nearest neighbor as the node's parent.
- In addition, the paths are rewired following adding the new node. After a vertex has been connected to the cheapest neighbor, the neighbors are checked to see if being rewired to the newly added vertex will decrease their cost.
- If the cost indeed decreases, the neighbor is rewired to the newly added vertex. This step makes the path smoother and less jagged looking.



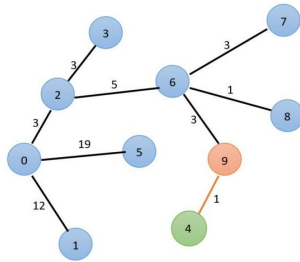
(1) Finds nearby neighbors of the new node



(3) Checks costs for rewiring



(2) Chooses the best (least cost) parent for the new node



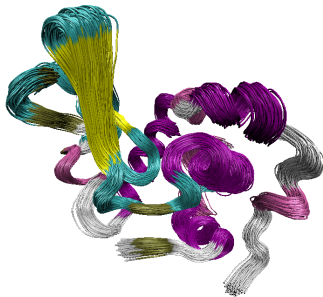
(4) Rewired tree

Inverse Kinematics (IK)

- Inverse kinematics is the problem of finding the right values for the underlying degrees of freedom of a chain.
- In the case of a protein chain these degrees of freedom of the dihedral angles, so that the chain satisfies certain spatial constraints.
- For example, in some applications, it is necessary to find rotations that can steer certain atoms to desired locations in space.
- The applications of inverse kinematics to protein structure include mainly loop modeling and generating ensembles of structures.
- In this case - manipulate the rotational degrees of freedom of a loop region to find possible loop conformations that attach to the rest of the protein.

Modeling Loops Using Inverse Kinematics

- Goal: Model the ensemble of conformations of a protein.
- It is known that proteins are not rigid but fluctuate about an ensemble of structures under equilibrium conditions.
- Focus mostly on loop regions, as they are the most flexible ones.



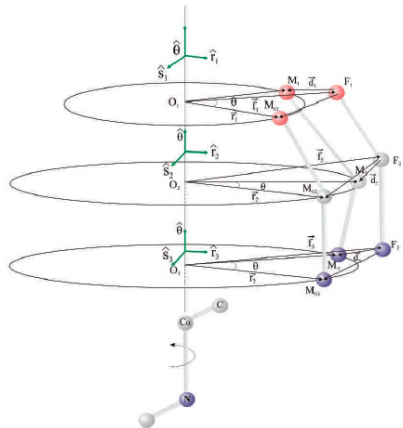
Modeling Loops Using Inverse Kinematics

- Inverse kinematics: Manipulate the degrees of freedom of an articulated chain to satisfy some end-constraints.
- In this case - manipulate the rotational degrees of freedom of a loop region to find possible loop conformations that attach to the rest of the protein.
- Cyclic Coordinate Descent (CCD): solve for and rotate one dihedral at a time.

Canutescu A. A., and Dunbrack R. L. Protein Science 12, 2003

CCD for Inverse Kinematics

- **Goal:** find optimal values to simultaneously steer the three backbone atoms of the end of the fragment to their target positions.
- Current positions before rotation - M_0 , after rotation M and target positions F .
- S is the sum of squared distances between current positions and target positions
- Steering these three atoms to their target positions requires minimizing S .



CCD for Inverse Kinematics

- S is defined as:

$$S = |\vec{F}_1 M_1|^2 + |\vec{F}_2 M_2|^2 + |\vec{F}_3 M_3|^2$$

Where

$$\vec{F}_1 M_1 = \vec{O}_1 M_1 - \vec{O}_1 F_1$$

- Notice that it is a 2D rotation around the plane defined by the \hat{r} and \hat{s} local axes.
- The squared norm of the vector $M - F$ (denoted FM) has this value for each of the three atoms, so we can sum the three contributions to S .
- We can express the rotation with respect to the \hat{r} and \hat{s} plane as:

$$\vec{O}_1 M_1 = r_1 \cos \theta \hat{r}_1 + r_1 \sin \theta \hat{s}_1$$

- r_1 is the vector between O and M_{01} , which we want to rotate by θ .

CCD for Inverse Kinematics

- From the previous equations above it follows that:

$$\vec{F}_i M_i = r_i \cos \theta \hat{r}_i + r_i \sin \theta \hat{s}_i - \vec{f}_i \equiv \vec{d}_i, i = 1, 2, 3$$

- Calculating the squared distances between the moving atoms and the fixed target atoms, we obtain:

$$|\vec{d}_i|^2 = r_i^2 + f_i^2 - 2r_i \cos \theta (\vec{f}_i \cdot \hat{r}_i) - 2r_i \sin \theta (\vec{f}_i \cdot \hat{s}_i)$$

Putting it all together, we can express S as the sum of the squared distances above.

- Differentiating with respect to θ gives us:

$$\frac{dS}{d\theta} = \frac{d|\vec{d}_1|^2}{d\theta} + \frac{d|\vec{d}_2|^2}{d\theta} + \frac{d|\vec{d}_3|^2}{d\theta}$$

where

$$\frac{d|\vec{d}_i|^2}{d\theta} = 2r_i \sin \theta (\vec{f}_i \cdot \hat{r}_i) - 2r_i \cos \theta (\vec{f}_i \cdot \hat{s}_i)$$

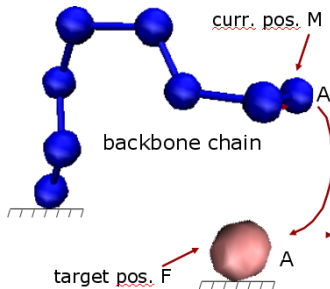
- After a little bit of math, S can be written as:

$$S = a - \sqrt{b^2 + c^2} \cos(\theta - \alpha)$$

- S is minimum when $\theta = \alpha$. Now we have explicit values for sine and cosine.
- Notice that the Time complexity is linear time on the number of DOFs to solve for all dihedrals of a chain.

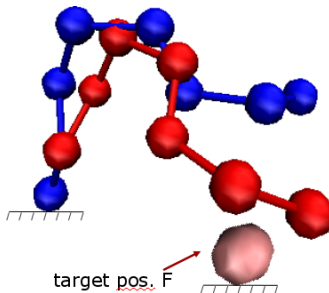
Modeling Loops Using Inverse Kinematics

- Cyclic Coordinate Descent:
solve for and rotate one
dihedral at a time
- Given: atom at current
position M, target position F
- Goal: Solve for dihedral θ
s.t. $|F - M|^2 = S(\theta) < \varepsilon$
threshold
- Time complexity: Linear
time on the nr. DOFs to
solve for all dihedrals of a
chain



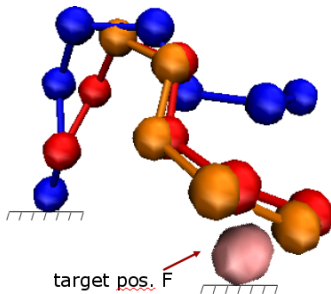
Modeling Loops Using Inverse Kinematics

- Cyclic Coordinate Descent:
solve for and rotate one
dihedral at a time
- Given: atom at current
position M , target position F
- Goal: Solve for dihedral θ
s.t. $|F - M|^2 = S(\theta) < \varepsilon$
threshold
- Time complexity: Linear
time on the nr. DOFs to
solve for all dihedrals of a
chain



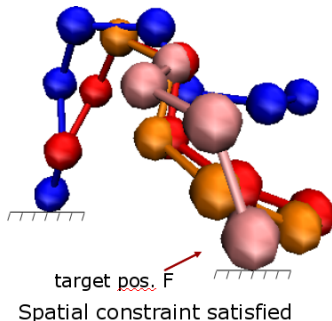
Modeling Loops Using Inverse Kinematics

- Cyclic Coordinate Descent:
solve for and rotate one
dihedral at a time
- Given: atom at current
position M , target position F
- Goal: Solve for dihedral θ
s.t. $\|F - M\|^2 = S(\theta) < \varepsilon$
threshold
- Time complexity: Linear
time on the nr. DOFs to
solve for all dihedrals of a
chain



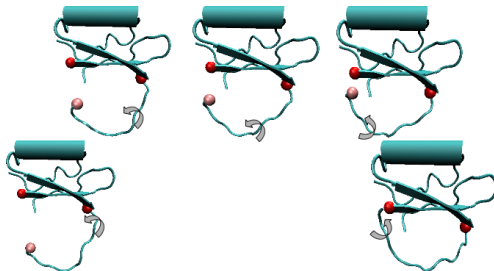
Modeling Loops Using Inverse Kinematics

- Cyclic Coordinate Descent:
solve for and rotate one
dihedral at a time
- Given: atom at current
position M , target position F
- Goal: Solve for dihedral θ
s.t. $\|F - M\|^2 = S(\theta) < \varepsilon$
threshold
- Time complexity: Linear
time on the nr. DOFs to
solve for all dihedrals of a
chain



Modeling Loops Using Inverse Kinematics

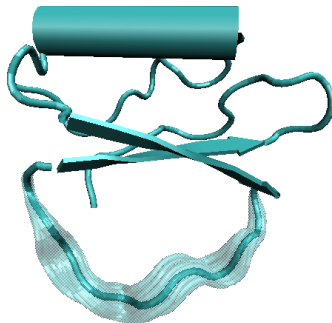
- Since there is redundancy, many solutions are feasible.
- Find rotations to satisfy spatial constraints on atoms Combine with energy minimization to obtain physical structures



Example: Chymotrypsin inhibitor 2

Equilibrium Fluctuations

More DOFs than spatial constraints can be exploited to generate fragment fluctuations



Example: Chymotrypsin inhibitor 2

Equilibrium Fluctuations

- Sample equilibrium fluctuations:
- Spatially constrained through Cyclic Coordinate Descent
- Energetically constrained to be feasible



Local Fluctuations in
 α -Lactalbumin

Boltzmann ensemble average

$$RMSD_x = \sum_{Confs} \frac{RMSD(C, C_{native}) e^{-\beta \Delta E_c}}{Q}$$

$$\Delta E_c = E_c - E_{native}$$

$$Q = \sum_{Confs} e^{-\beta \Delta E_c}$$

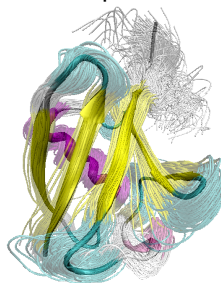
Equilibrium Fluctuations

α -Lactalbumin (α -Lac)



- 123 residues
- Hydrogen exchange protection factors available

Ubiquitin



- 76 residues NMR information on fluctuations available