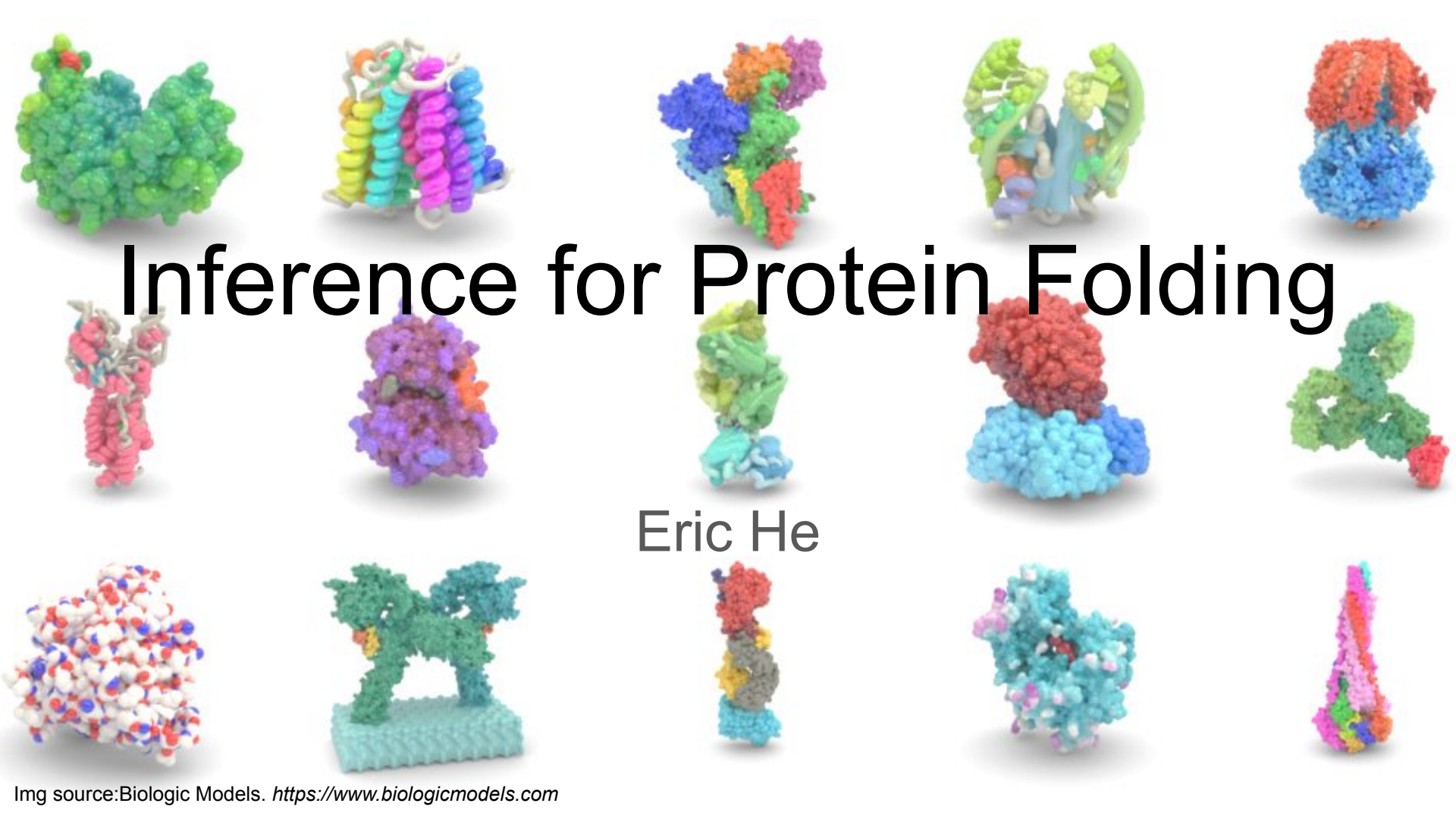
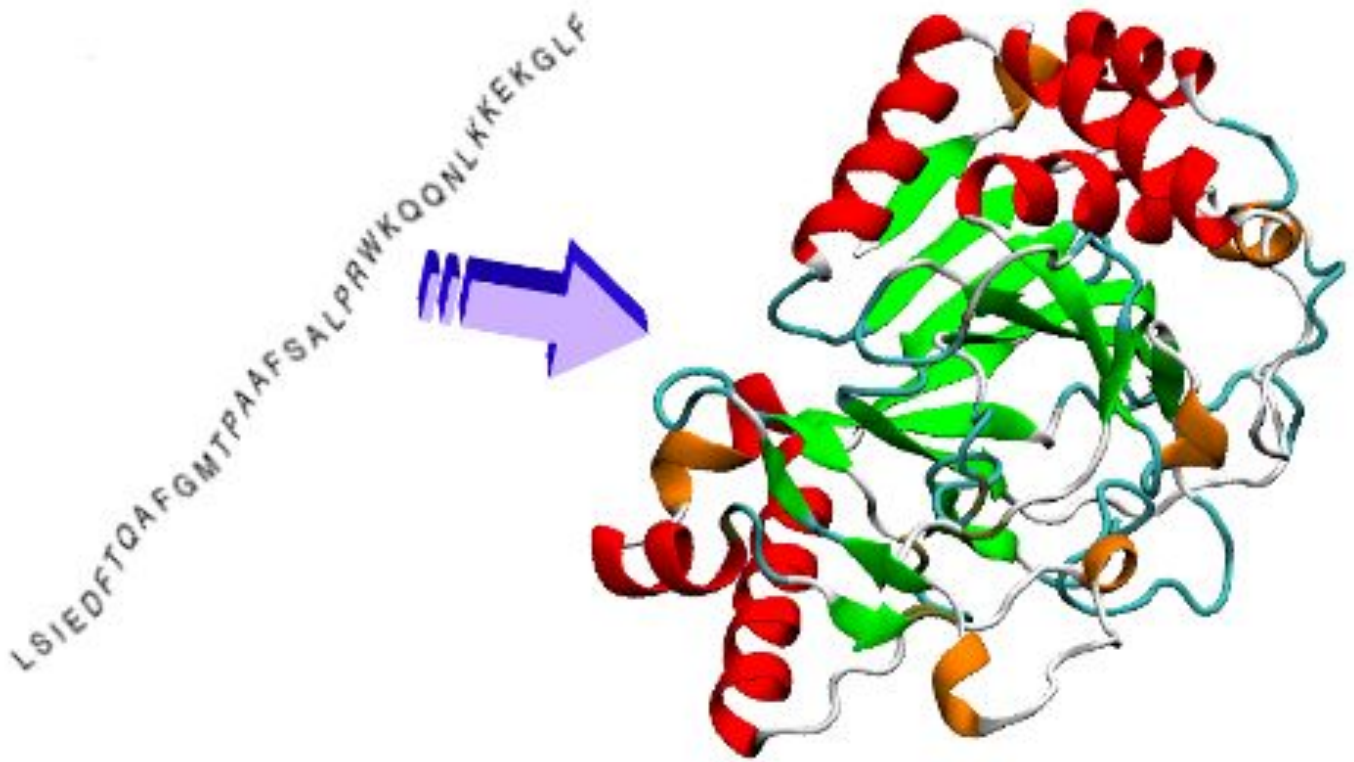


Inference for Protein Folding

Eric He

Img source:Biologic Models. *https://www.biologicmodels.com*

Goal: predict the 3-D “tertiary structure” given the “primary structure” of amino acids



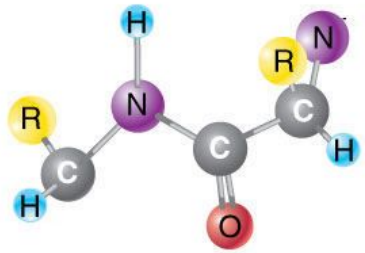
Img source:Wei Qian (2017). *Graph Matching, Pattern Learning, and Protein Modeling*

The physics of protein folding are complex

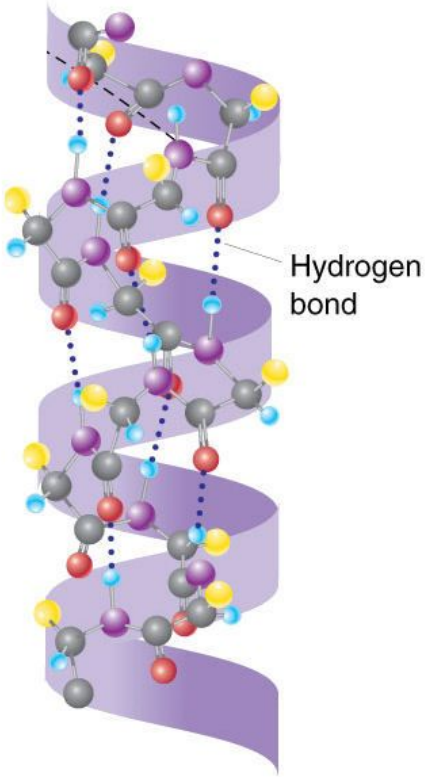
It took 100 days for the purpose-built supercomputer *Anton* to simulate 100 milliseconds of folding

Img source:Matt Simmons

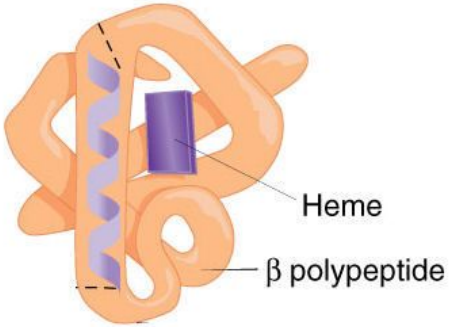
Empirical data reveals information on emergent folding structure



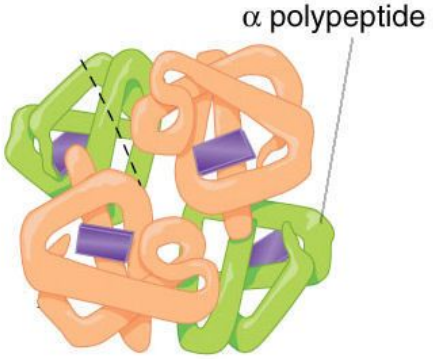
(a) Primary structure



(b) Secondary structure

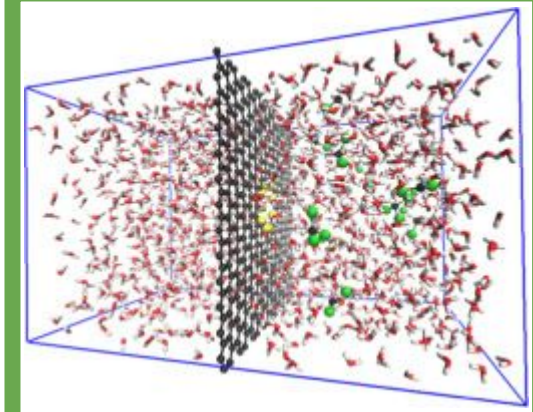


c) Tertiary structure

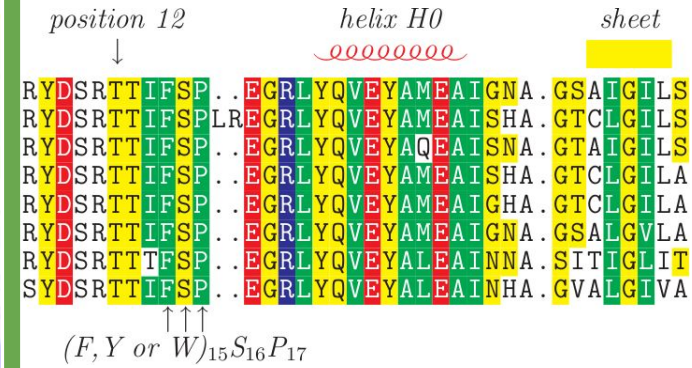


(d) Quaternary structure

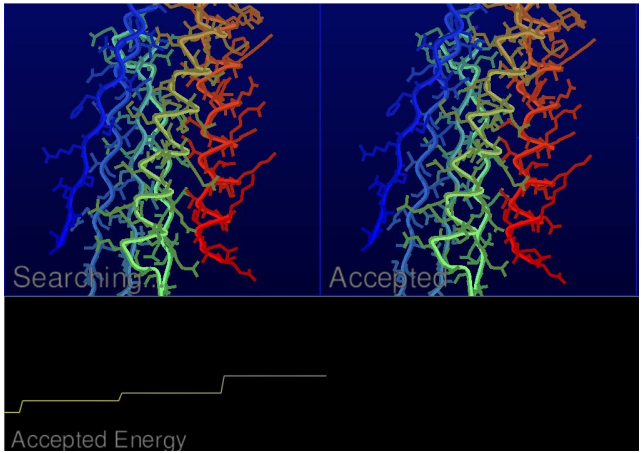
Probabilistic methods dominate all aspects of the folding problem



Markov chains for molecular dynamics



Hidden Markov models for multiple sequence alignment



Monte Carlo minimization of empirical potentials

these are a few applications… there are many more

Img source:doi.org/10.1016/j.ces.2015.01.048

Img source:http://www.bioinformatics.org/strap/

Img source:Rosetta@Home

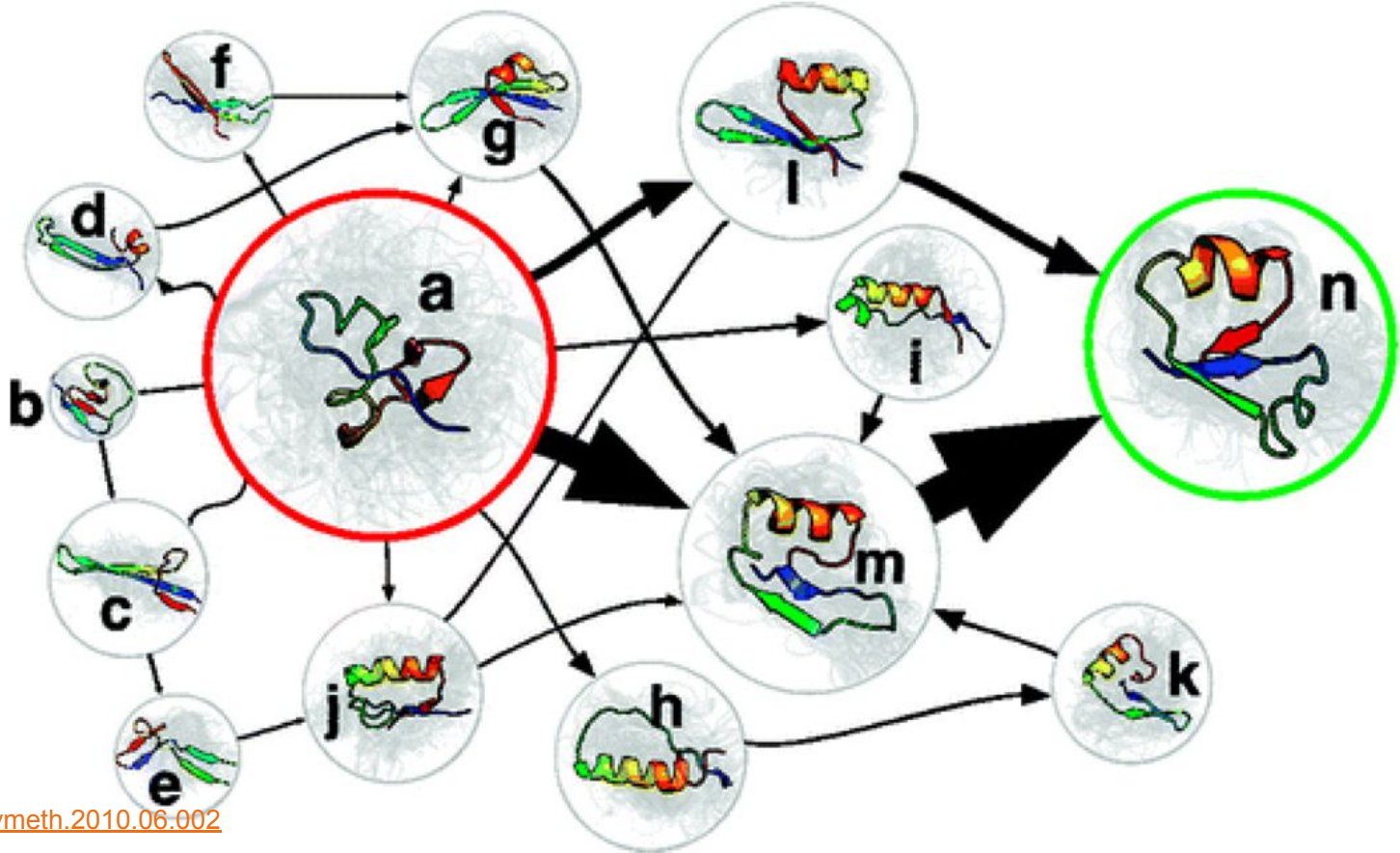
Molecular Dynamics simulations are used to construct Markov State Models of protein states

Modified k-means clustering discretizes conformations into microstates

Lumping function ***L*** assigns macrostates to microstates

Macrostate transition matrix ***T***

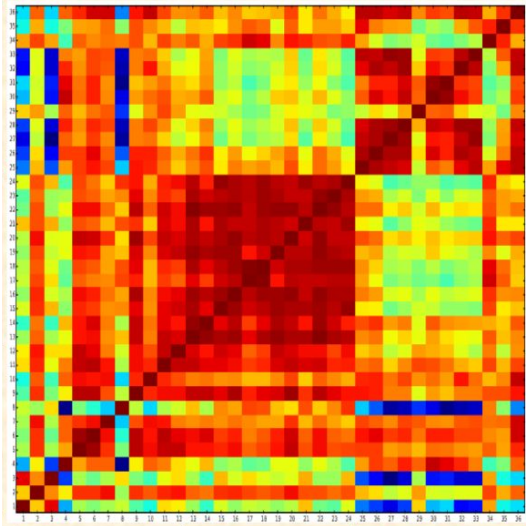
Microstate emission matrix ***Θ***



Img source:https://doi.org/10.1016/j.ymeth.2010.06.002

Testing the Markov assumption on the macrostate transition matrix T

Leaving time test



The time to leave state *i* should follow a geometric distribution with parameter T_{ii}

Eigenvalue decay test

$$\text{Eigenvalues}(A) = \lambda$$
$$\text{Eigenvalues}(A^t) = \lambda^t$$
$$\implies \lambda \sim \text{Exponential}$$

The non-unity eigenvalues of T_Δ should follow an exponential distribution w.r.t. the time step Δ

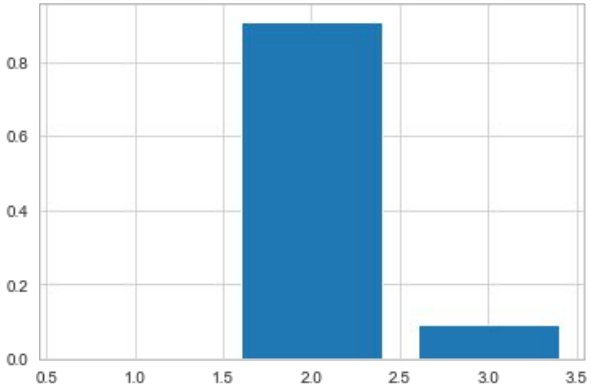
Img source:https://simssa.ca/blog/simssa-and-dream/

Dirichlet prior on the space of transition matrices satisfying detailed balance

$$C_i = \begin{bmatrix} 0 & 20 & 2 \end{bmatrix} \qquad P_i = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \qquad \mu_i = C_i + P_i$$

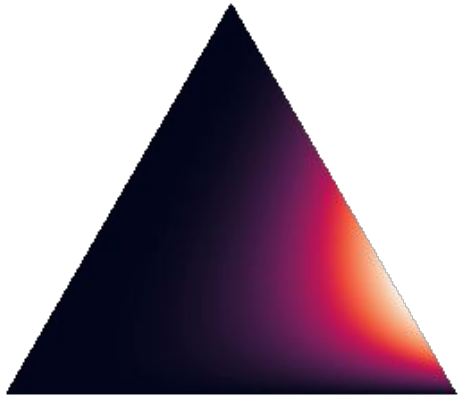
Naive MLE on C_i

$$T_i = \text{Multinoulli}\left(\frac{\{C_{ij}\}}{\sum_j C_{ij}}\right)$$



Dirichlet Prior

$$T_i = \text{Dirichlet}(\mu_i)$$



MLE With Detailed Balance

$$T_{ij} = \text{Multinoulli}\left(\frac{C_{ij} + C_{ji}}{\frac{|C_i|}{|T_i|} + \frac{|C_j|}{|T_j|}}\right)$$

Dirichlet Prior With Detailed Balance

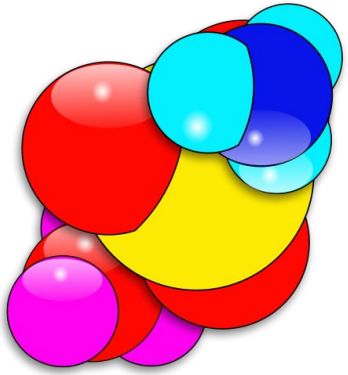
$$T_{ij} = \text{Very complicated!}$$

Adaptive sampling cuts computations by 99%

$$\lim_{|\mu|\to\infty} \text{Dirichlet}(\mu) \to \text{MVN}(m = \frac{\mu}{|\mu|}, \Sigma = \frac{|\mu|\mu - \mu\mu^T}{|\mu|(|\mu|+1)})$$

An MVN allows us to estimate the variance of a metric with respect to our uncertainty around the transition probabilities of a particular state. We focus our simulations on these variance-enhancing states.

Warm your room this winter by donating your spare computer time to computational biology projects



Rosetta@home

Protein Folding, Design, and Docking

Leading tertiary structure prediction software on CPU

FOLDING @HOME

GPU-based molecular dynamics experiments for protein folding

There are also computational astronomy and math projects, if you prefer those