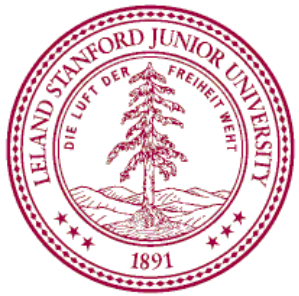
Folding@Everywhere

Computational Biochemistry in the New Era of HPC

Imran Haque
Department of Computer Science
Stanford University

http://cs.stanford.edu/people/ihaque
http://folding.stanford.edu



Folding@home distributed computing

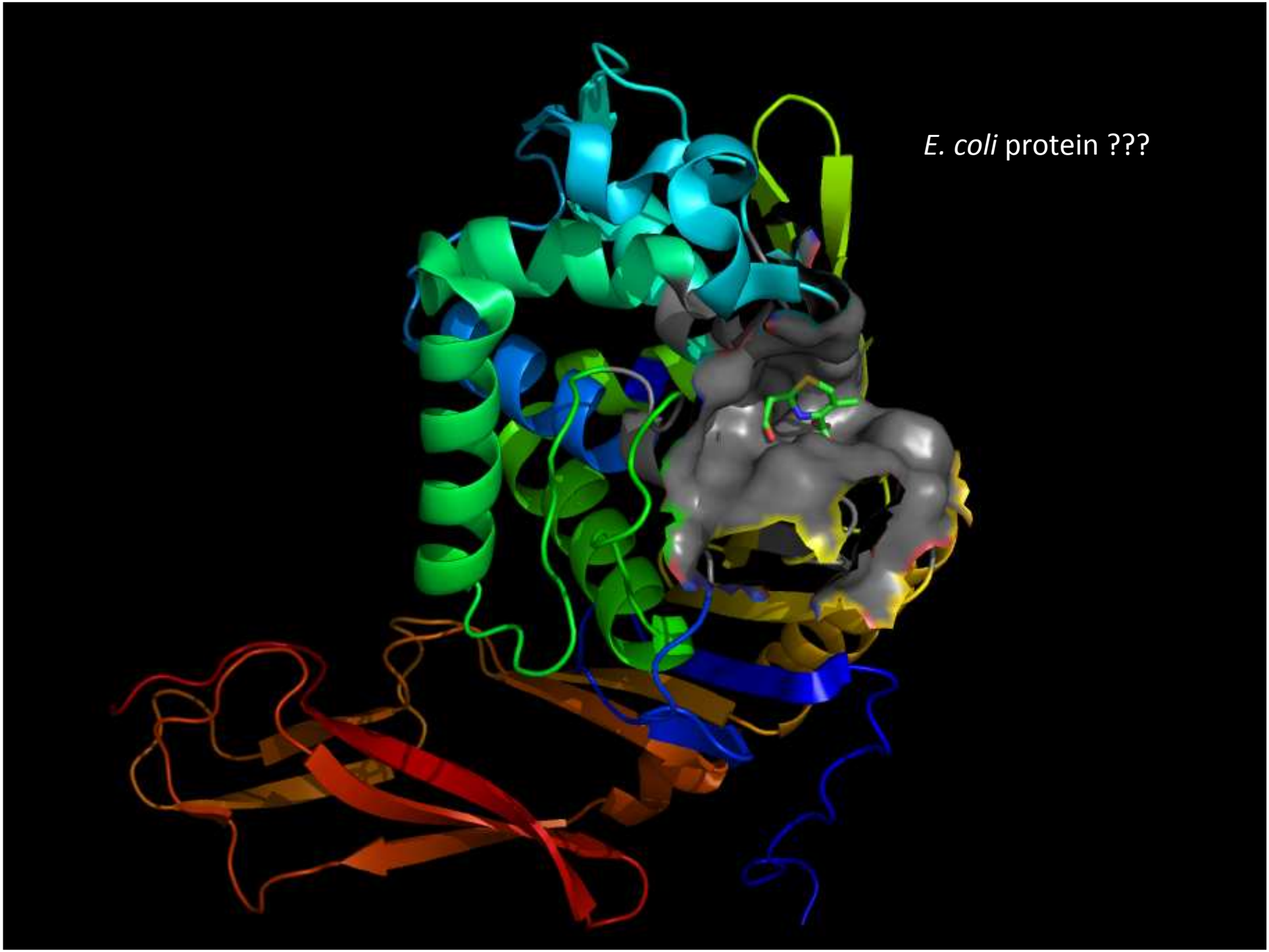
Hyperience, 24 Nov 2010

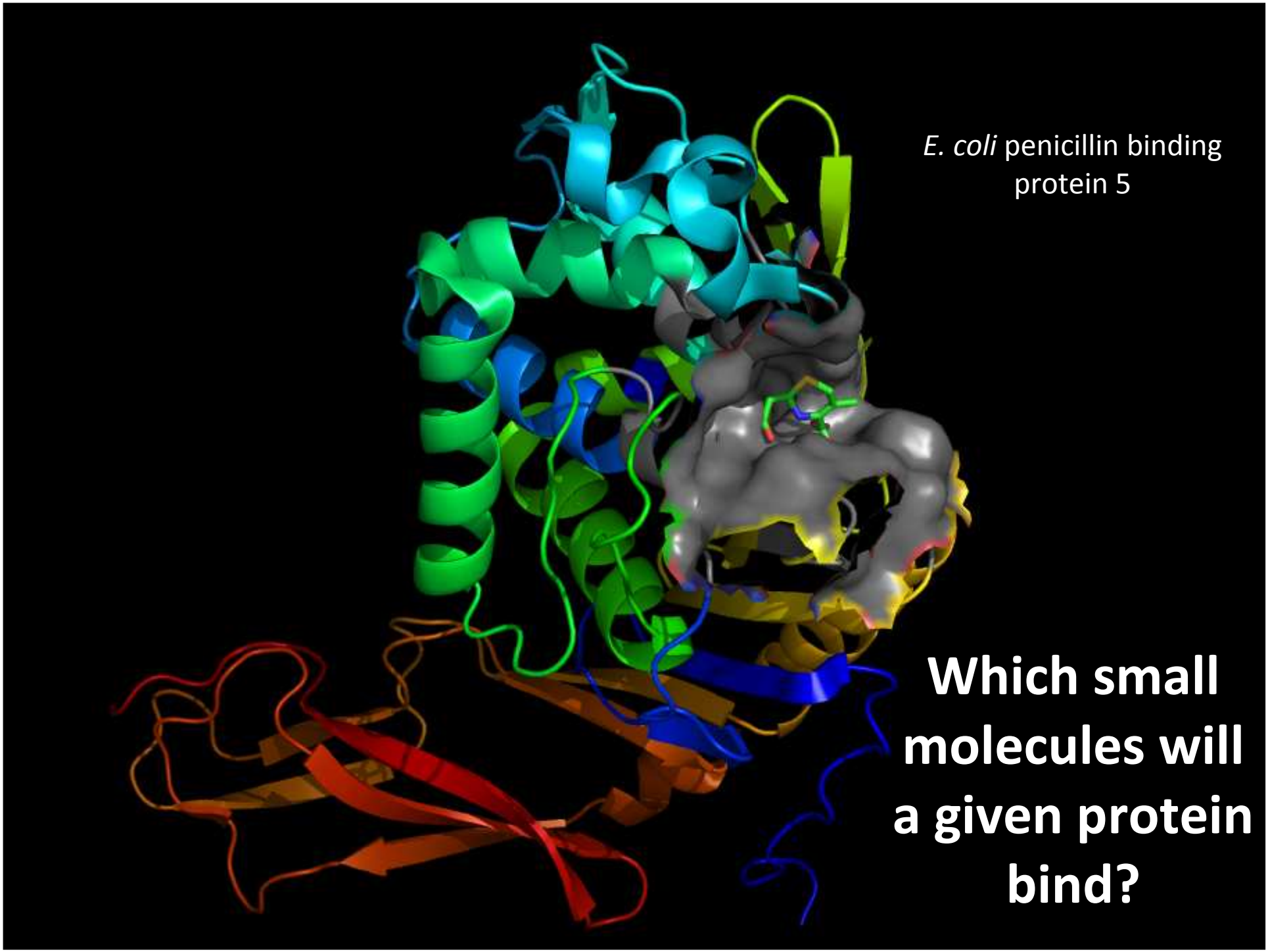
Conclusions

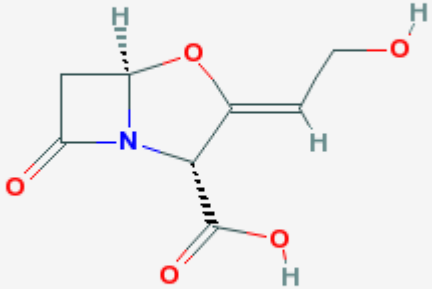
- Future HPC will be driven by **heterogeneous architectures** and (even more) **massive parallelism**

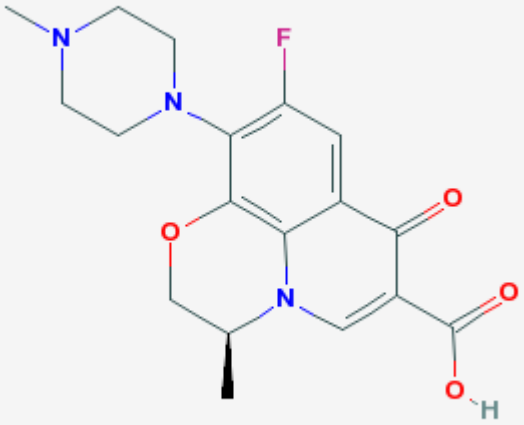
- Applications need both **systems- and algorithms-level redesign** to be effective on next-generation HPC

- **Our work shows a possible direction**: GPU rewrites and entirely new algorithms driving cheminformatics and physical simulation


E. coli protein ???

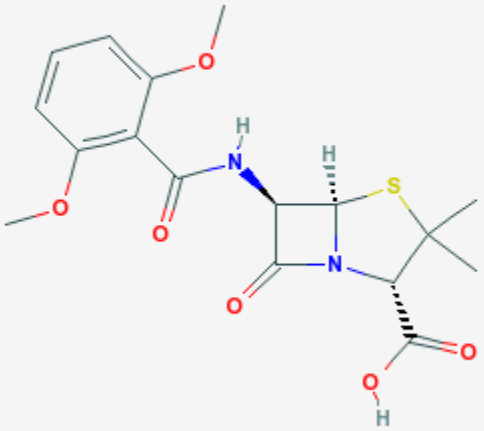

E. coli penicillin binding protein 5
Which small molecules will a given protein bind?

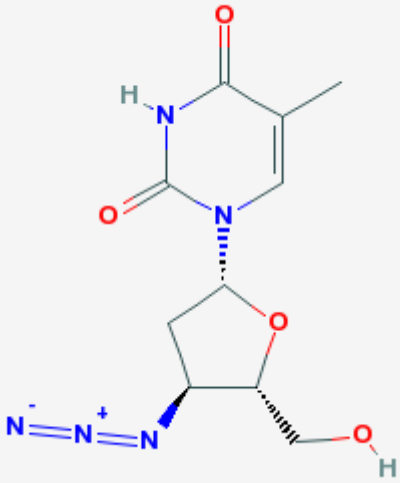


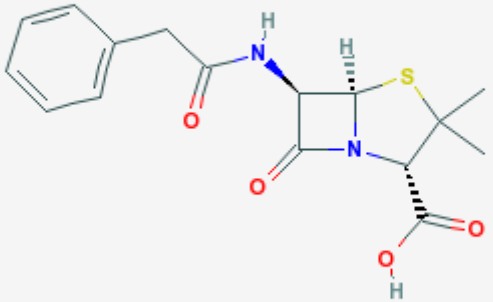


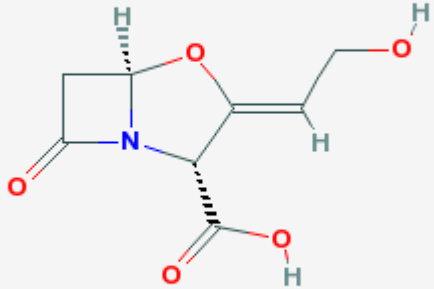
What do these compounds do?

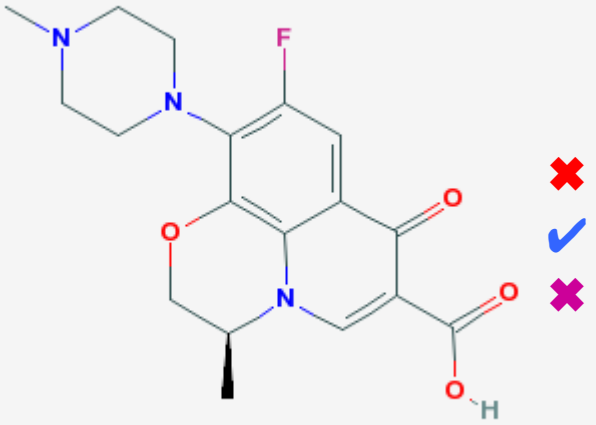
- **inhibit penicillin binding proteins?**
- **kill bacteria?**
- **kill viruses?**





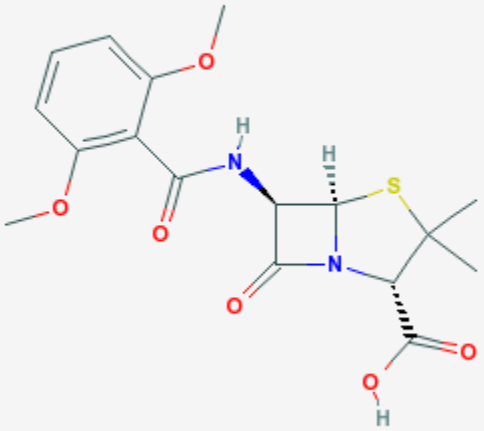


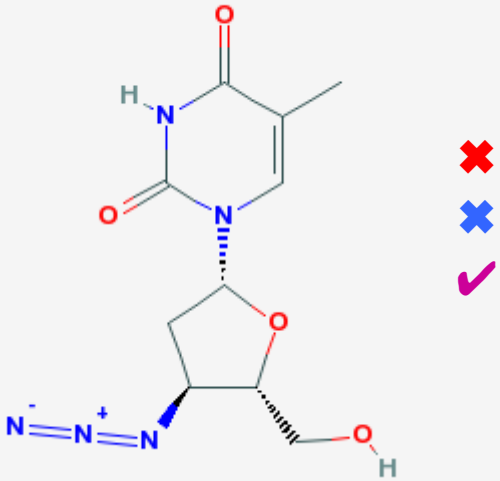


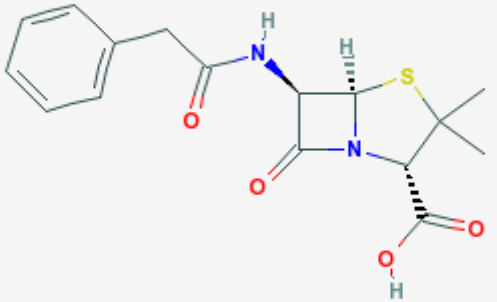


What do these compounds do?

- inhibit penicillin binding proteins?
- kill bacteria?
- kill viruses?

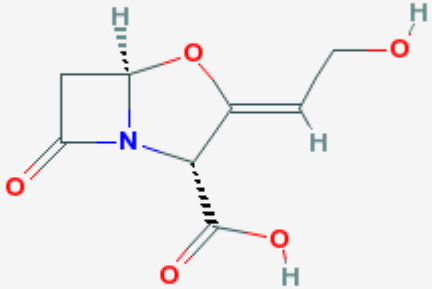






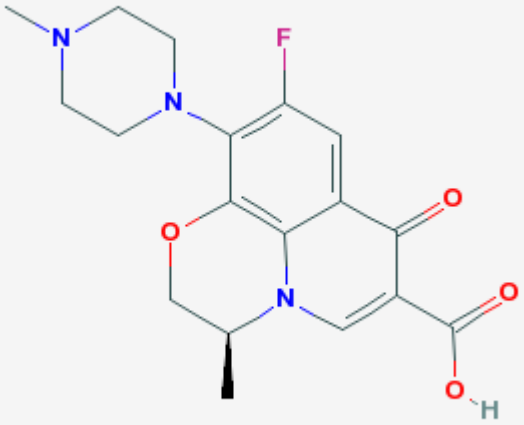
bisphenol A

estrogen mimic



clavulanic acid

beta-lactamase inhibitor

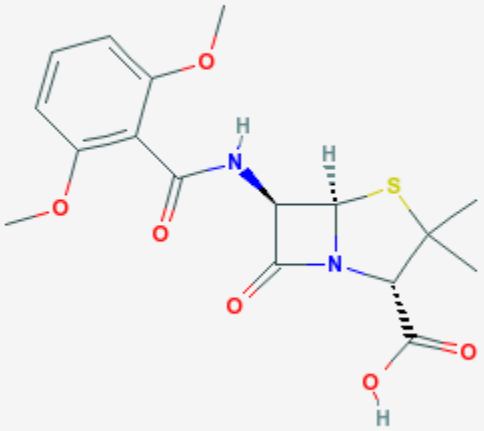


levofloxacin

DNA gyrase inhibitor

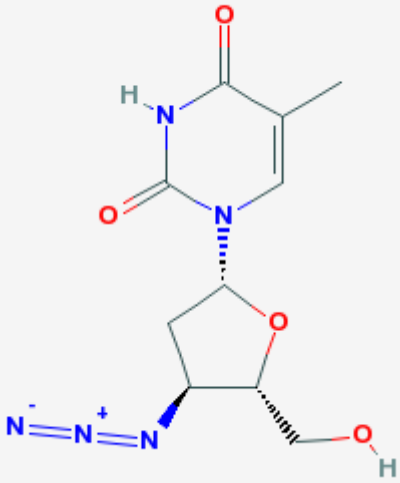
methicillin

beta-lactam antibiotic



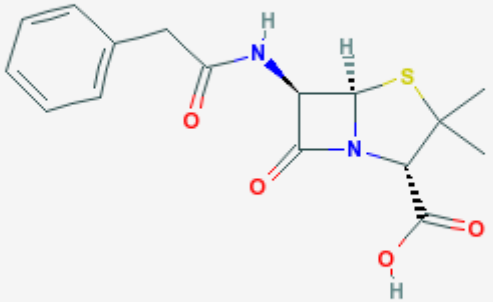
zidovudine

HIV RT inhibitor



penicillin G

beta-lactam antibiotic



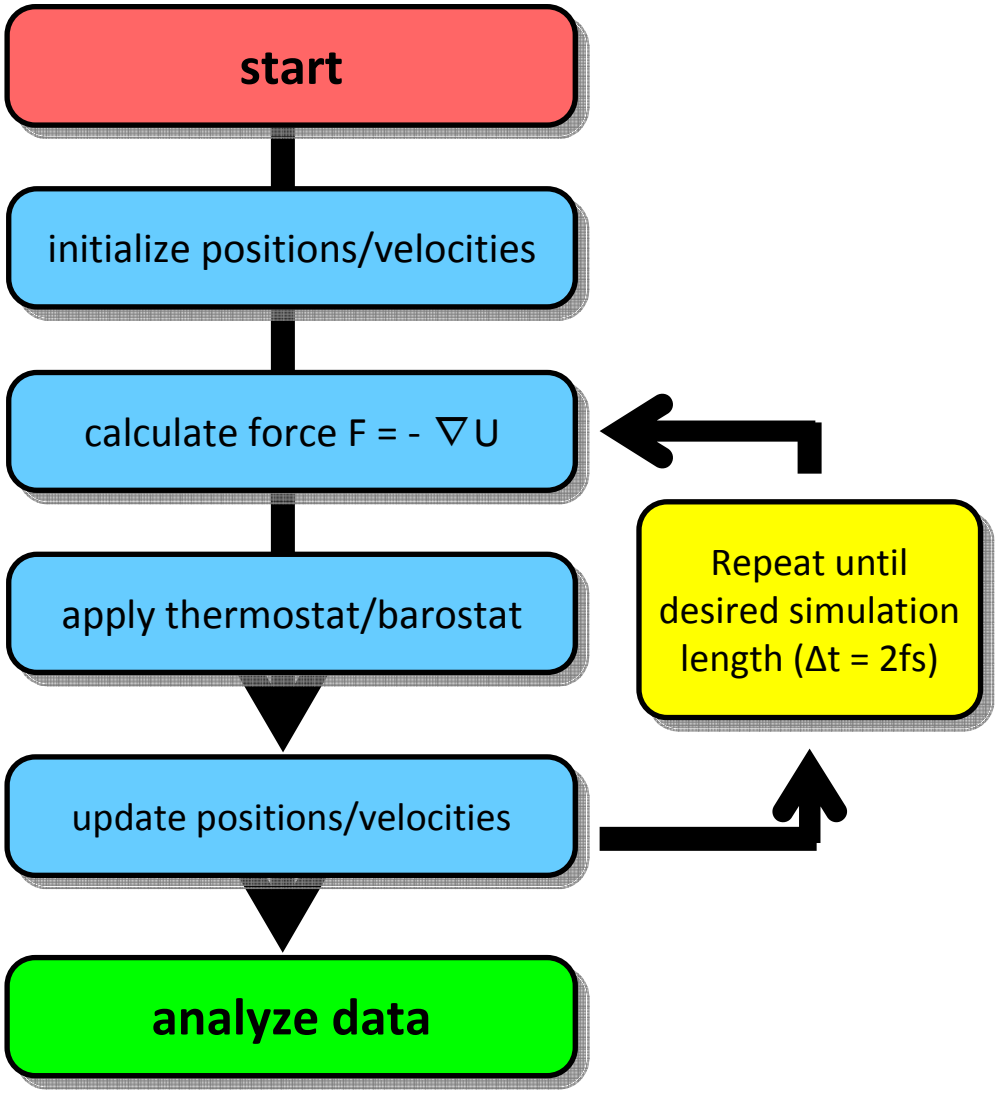
Chemical Biology - Methods

- Experimental assays: expensive, labor-intensive

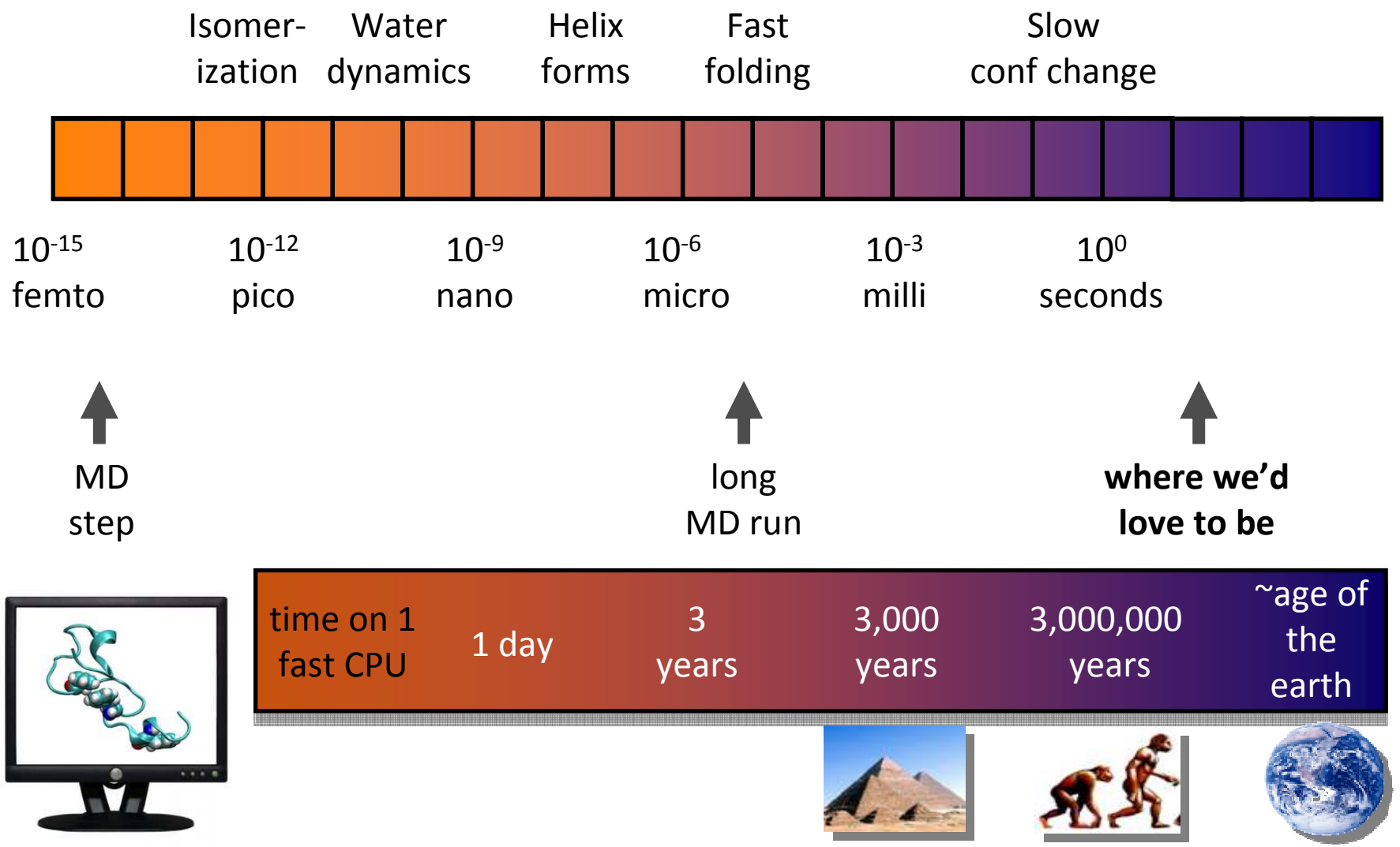
- **Computation**
  - **Physical simulation**

Physical Simulation

- MD = Numerical Integration of Newton's equation
- Dominant simulation method in computational biology and chemistry
- Can work with detailed (eg atomistic) or coarse grained models
- Detailed models needed for quantitative comparison to experiment



Physical Simulation - Timescales



Chemical Biology - Methods

- Experimental assays: expensive, labor-intensive

- **Computation**
  - **Physical simulation**
  - **Data mining**

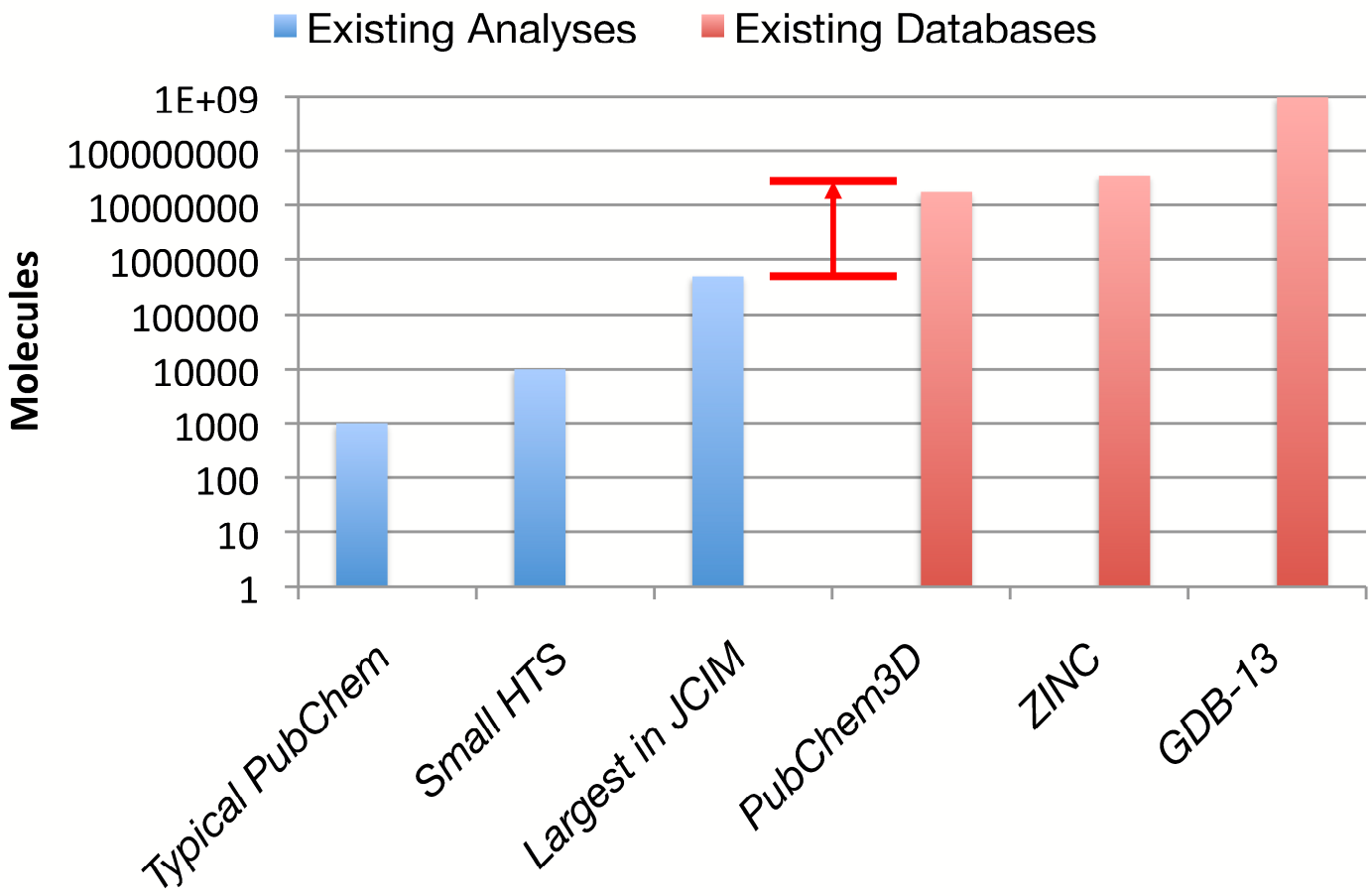
Chemical Databases

- A modern trend – giant **public** databases of chemical assay data
  - NCBI PubChem: 34,340 assays; 965,730 compounds
  - EBI ChEMBLdb: 8,054 targets; 600,625 compounds

- Companies releasing their internal databases
  - **GlaxoSmithKline**: Gamo et al. Thousands of chemical starting points for antimalarial lead identification. *Nature* **465,** 305-310 (20 May 2010).

- **Let's learn from this data and make predictions – chemical informatics or data mining!**

The Cheminformatics Gap



Computational analysis has not kept up with growth in chemical databases: the ***cheminformatics gap.***

Not just a linear gap

- Chemical similarity comparison is a common bottleneck in chemical algorithms
- How many similarities for N molecules?
  - **Virtual screening, k-means clustering: O(N)**
  - **Hierarchical clustering, network analysis: $O(N^2)$**
  - **LM hierarchical: $O(N^3)$**

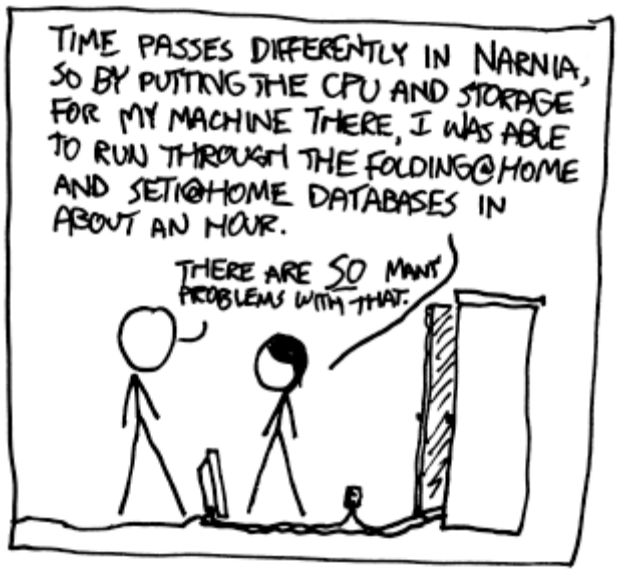
The gap is not just 10x-100x…

more like 100x – 1 million x!

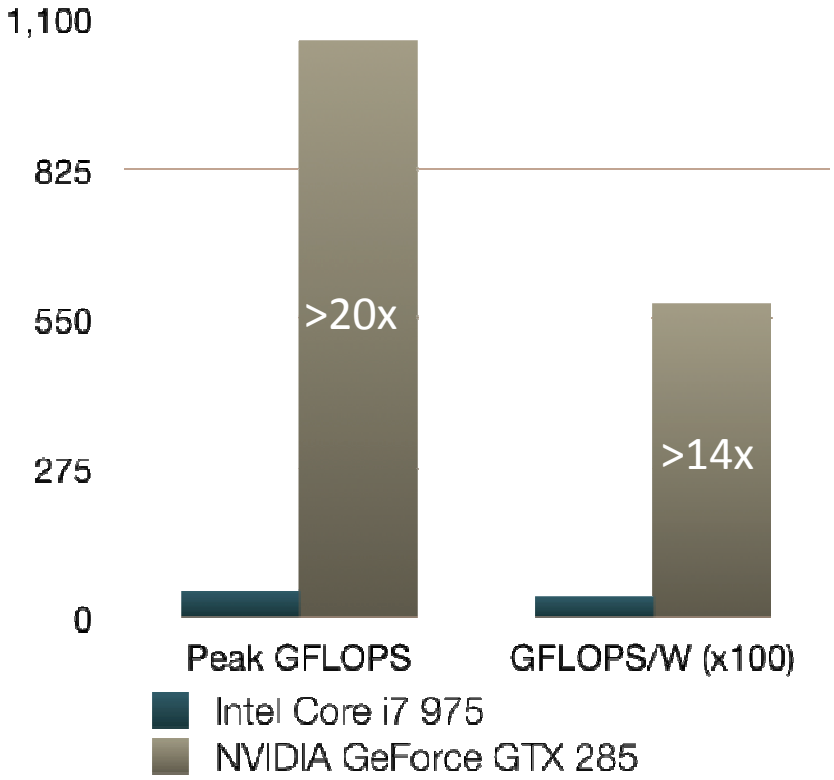
The Computational Barrier

For both physical simulation and data mining, we're about 1,000,000x short of where we'd like to be.

What can we do?

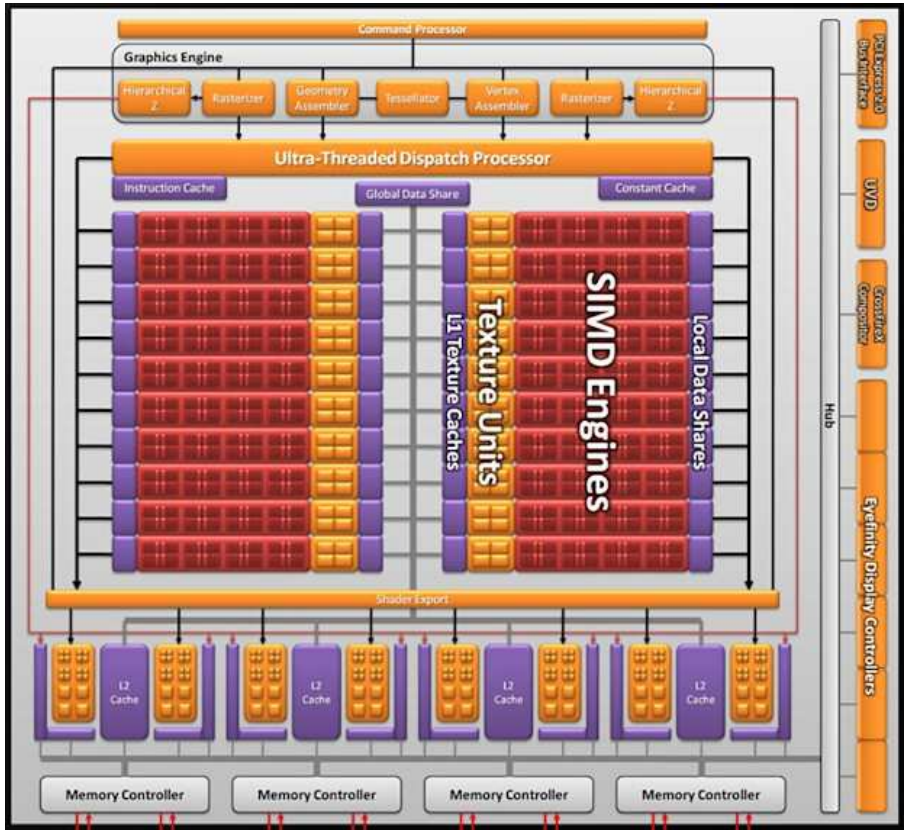


Why GPUs?



- GPUs have excellent peak throughput and efficiency
- BUT
  - Hard to program
  - Require inherent data parallelism
  - Often require complete rewrite
  - Questionable reliability

What's in a GPU?



NVIDIA GF100

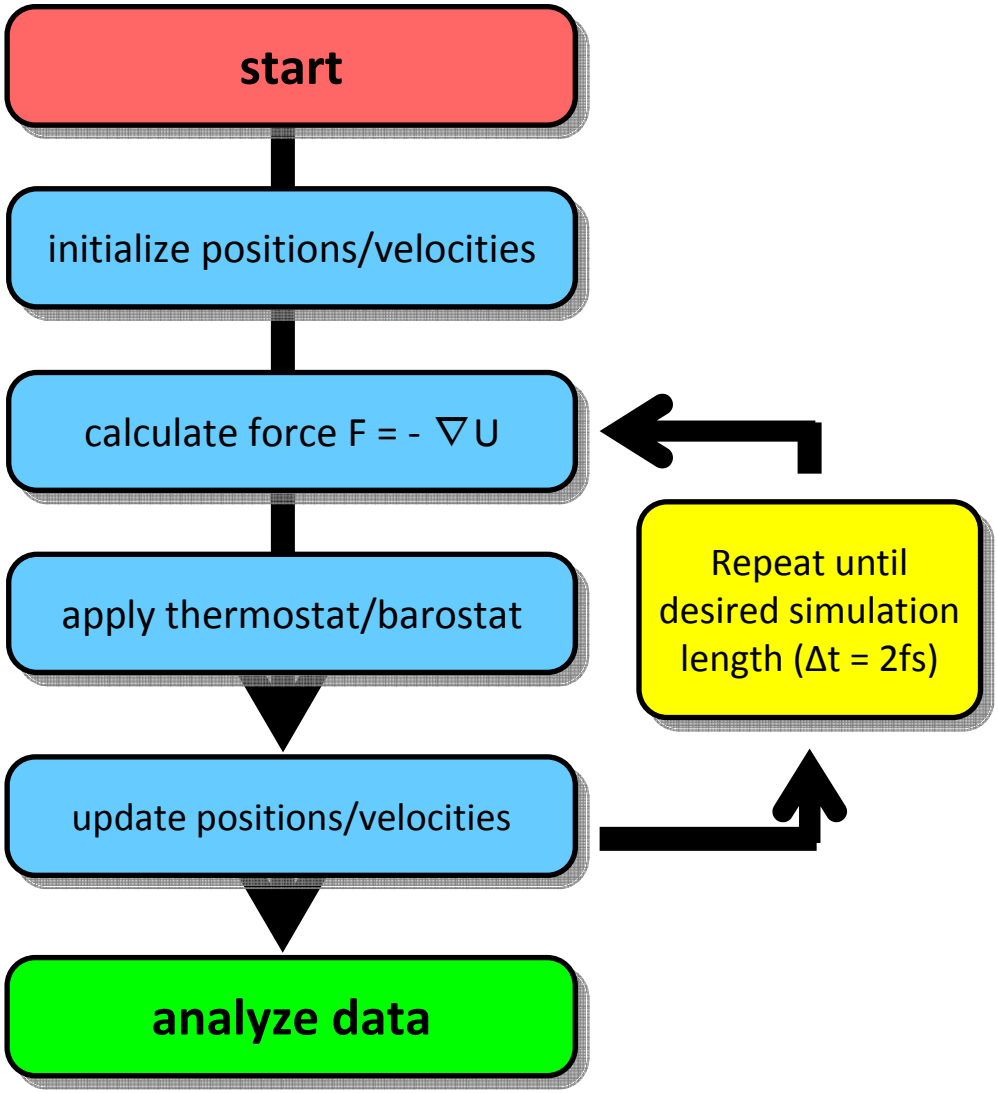
(GeForce GTX 480)

AMD Cypress

(Radeon HD 5870)

Physical Simulation

- Molecular dynamics is highly parallel
- Synchronization overhead decreases as system size increases
- Excellent fit for GPU acceleration



(Beauchamp, OpenMM team, Pande)

OpenMM – High Performance MD

Molecule	# atoms	ns/day	speedup*	GFLOP/s (GPU)	GFLOP/s (x86)
fip35	544	576	**128x**	311	657
villin	582	529	**136x**	328	692
lambda	1254	202	**255x**	547	1153
α-spectrin	5078	17	**735x**	805	1702

* GTX280-OpenMM vs Core 2 Duo 3GHz-AMBER (one core);
Fermi is ~2x faster!

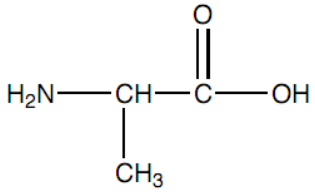
http://simtk.org/home/openmm

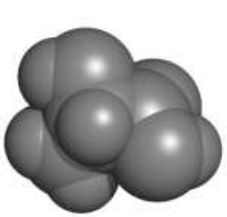
Friedrichs MS et al. *J. Comput. Chem.,* **2009,** *30*(6), pp 864-872
Luttman E et al. *J. Comput. Chem.*, **2009**, *30*(2), pp 268-274

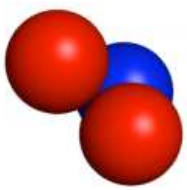
3 Views of Chemical Similarity

- 2D substructure:
- 3D shape:
- 3D chemotype:

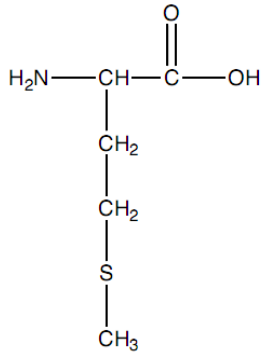
Alanine

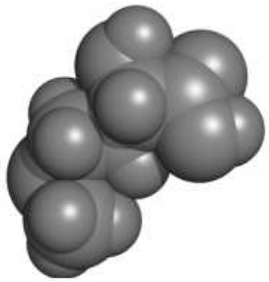


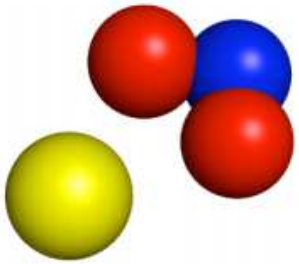




Methionine

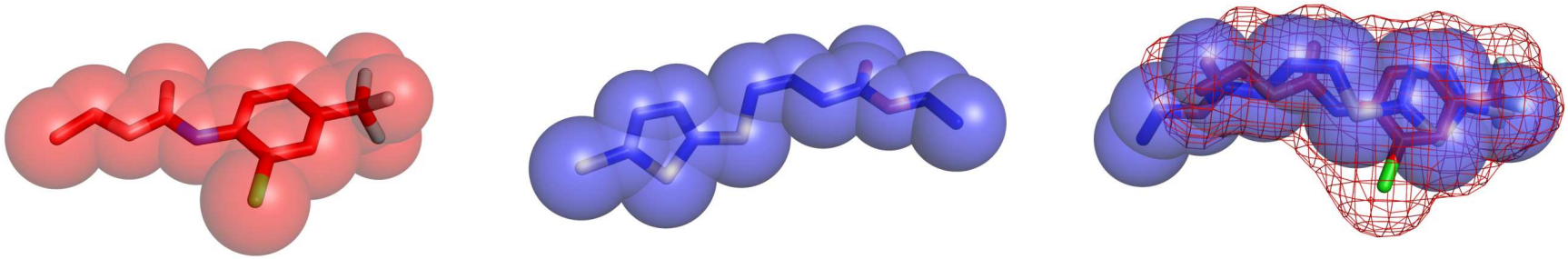






GPU-Accelerated 3D Similarity

- Molecular overlay optimization: used to find new active compounds from a database given one active "query" molecule



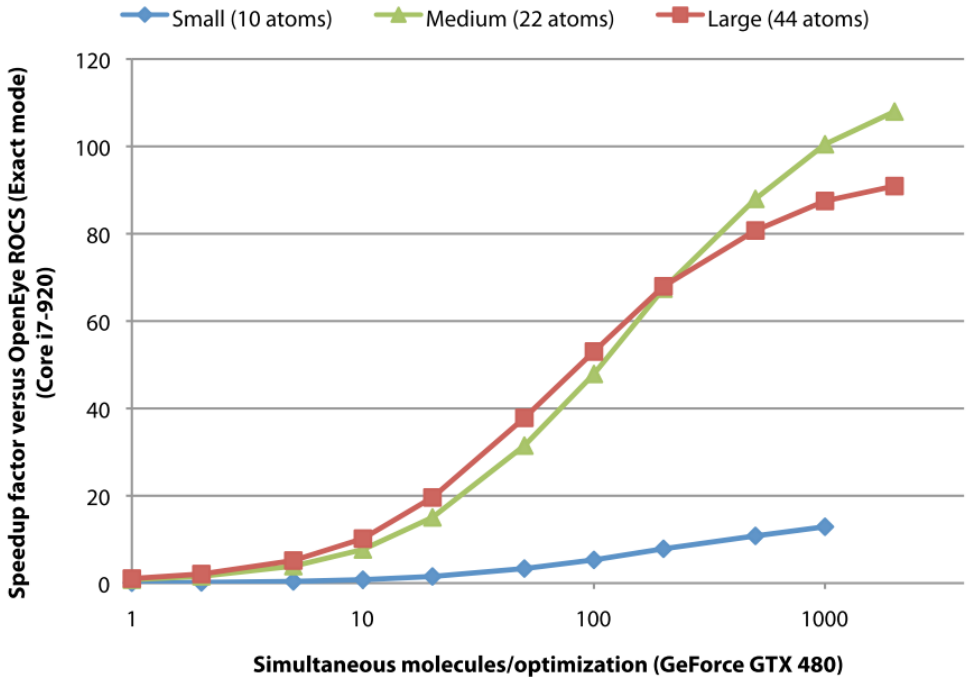
- Complexity O(MN): double-loop over all atom pairs
- DB = ~10M mol.; CPU = 100/sec = ~2 days/query
- *Use GPU to exploit parallelism of problem.*

http://simtk.org/home/paper

Haque IS and Pande VS. *J. Comput. Chem.,* **2010,** *31*(1), pp 117-132
Haque IS and Pande VS. in *GPU Computing Gems, vol 1.* **2010**

PAPER or PLASTIC, sir?

- Use GPUs to accelerate 3D shape-only (PAPER) or shape+color (PLASTIC) comparison: **100x speedup**



- PLASTIC: 15000 alignments/sec/GPU

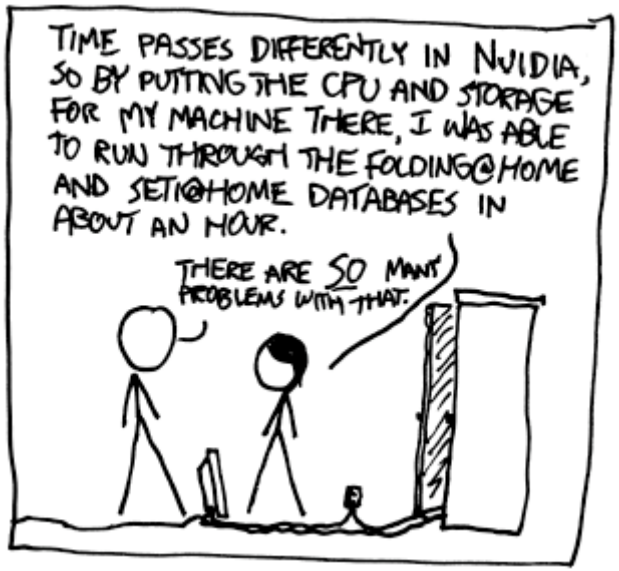
http://simtk.org/home/paper

Haque IS and Pande VS. *J. Comput. Chem.,* **2010,** *31*(1), pp 117-132

Haque IS and Pande VS. in *GPU Computing Gems, vol 1.* **2010**

The Computational Barrier

GPUs get us a factor 100-1000x. **Problem solved?**



Systems work is just the beginning – we also need **new algorithms** to bridge the rest of the gap. These algorithms will **rely on domain knowledge.**

Cheminformatics: a storage challenge

- Speeding up a $O(N^2)$ algo 100x is not enough:

Problem size	CPU time	Storage needed
10 mols	1 ms	1 kB
10K mols	1 min	1 GB
100K mols	1 day	1 TB
10M mols	**3 yr**	**1 PB**
1B mols	**30K yr**	**10K PB**

- Computing on existing-scale datasets requires entire datacenters' worth of storage.

A Modest Proposal

- Let's calculate all the pairwise similarities for compounds in PubChem3D (N = 17M) based on 3D shape and 2D chemical similarity
- Using CPUs
  - 3D: OpenEye ROCS: 150/sec/core = ***30,000 cpu-years***
    1 PB for matrix
- Add GPUs:
  - 3D: PAPER: 15K/sec/gpu = ***300 gpu-years***
    Still 1 PB disk

SCISSORS: Math for Fun and Profit

- Many molecular similarity methods report similarity as a Tanimoto score
- How can we use the mathematical structure of Tanimotos to gain insight into the metrics and **calculate them faster**?

Classical vector Tanimoto returns value in [-1/3, 1] for a pair of vectors A, B in terms of their inner products

$$T_{AB} = \frac{\langle A, B \rangle}{\langle A, A \rangle + \langle B, B \rangle - \langle A, B \rangle}$$

Tanimoto equation can be rearranged to get inner product in terms of Tanimoto and vector magnitudes

$$\langle A, B \rangle = \frac{T_{AB}}{1 + T_{AB}} \left(\langle A, A \rangle + \langle B, B \rangle \right)$$

Haque IS and Pande VS. *J. Chem. Inf. Model.,* **2010** *50*(6), pp1075-1088.

SCISSORS: Derivation

- Assume molecules can be represented as vectors in $\mathbf{R}^N$
- Simple assumptions on <A,A> and <B,B> get us <A,B>

$$\langle A, B \rangle = \frac{2\,T_{AB}}{1 + T_{AB}}$$

- Given a matrix G of inner products, want matrix M with molecule vectors along rows

$$MM^T = G$$

- G is real-symmetric, so use eigenvalue decomposition

$$G = MM^T = VDV^T$$

$$M = VD^{\frac{1}{2}}$$

Haque IS and Pande VS. *J. Chem. Inf. Model.,* **2010** *50*(6), pp1075-1088.

SCISSORS: The key

- Select a small number **k** of molecules (**k** << **N**) to act as a “basis set”
- Do all-pairs comparison on basis set and decompose to molecule matrix **M**
- For each new “library” molecule **x**, run slow method only against basis set. Place inner products in a vector and solve for vector rep of **x** by least-squares:

$$M\vec{x} = T$$

- All-pairs: now only O(**kN**) slow computations!

Haque IS and Pande VS. *J. Chem. Inf. Model.,* **2010** *50*(6), pp1075-1088.

Hardly Even a Request…

- 3D: Using PAPER+SCISSORS (basis size=2700)

  17M * 2700 / 15000 = 35 gpu-day +

  17M * 17M / 600M = 5 gpu-day

  274,000x speedup (vs 30 000 cpu-yr)

- Storage: 17GB for SCISSORS

  33,000 x reduction

- Computations that required all of FAH can now be done on a single (well-equipped) desktop

Scalability and Resilience

- Proposed exascale initiative roadmap suggests *dramatically* higher concurrency levels

	2009	**2011**	**2015**	**2018**
FLOP/s	2 Peta	20 Peta	200 Peta	1000 Peta
Total concurrency	225,000	3,200,000	50,000,000	1,000,000,000
MTTI	Days	Days	Days	O(1 day)

- FAH data corroborate short MTTI for new GPU archs.

- Need **scalable, resilient** algorithms for physical sim

C Engelmann, Oak Ridge Natl. Lab

Haque IS and Pande VS. Proc. CCGrid 2010.

Limitations of traditional parallel MD

- Parallelism by spatial decomposition
  - each CPU gets assigned atoms
  - calculates the force for “its” atoms
  - communication between boxes
- Challenge
  - how to break up the problem for billions of processors when you only have millions of atoms?
  - What do you do when you only have thousands?!?!?
- What about scaling to billions of processors?
  - **can’t have # processors > # atoms**
  - **machine may not even run long enough to checkpoint/restart**

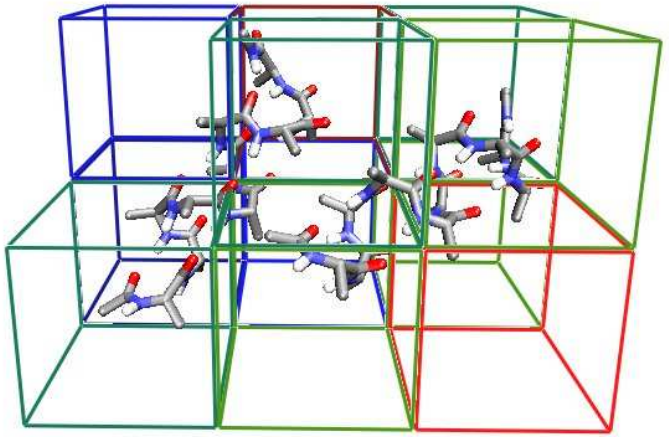
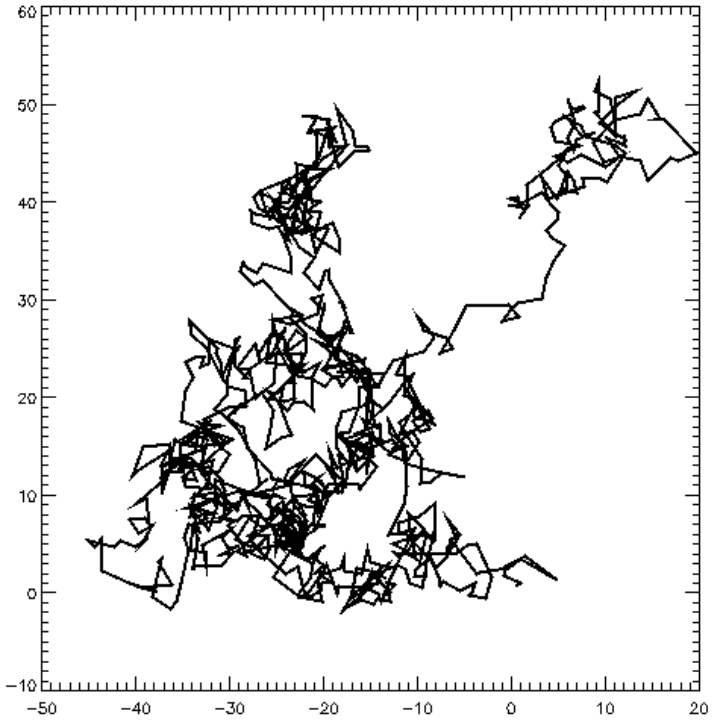
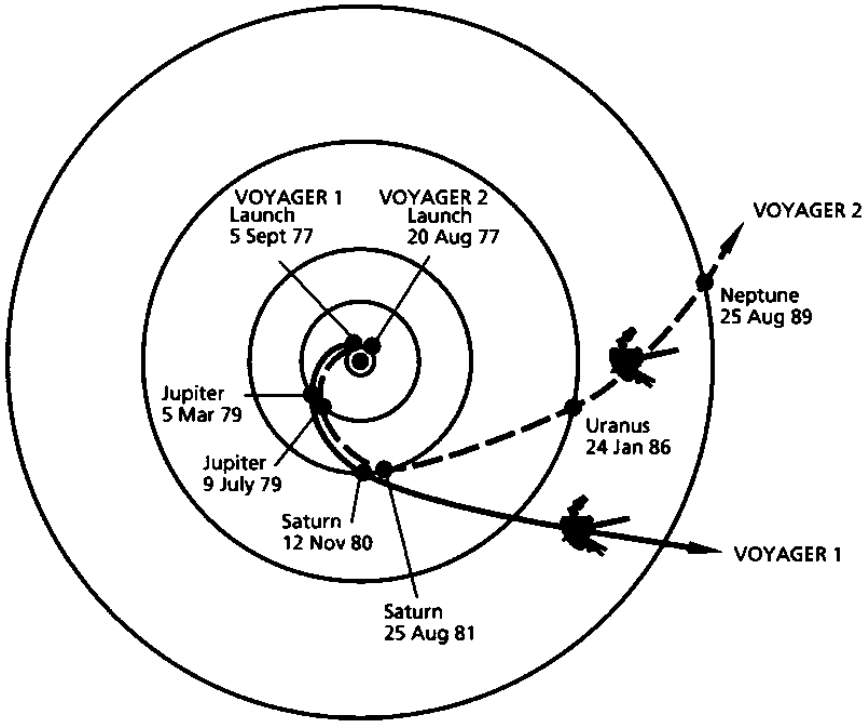


figure from http://www.ks.uiuc.edu/Research/Algorithms/

Anton from D. E. Shaw

How to think of MD simulations





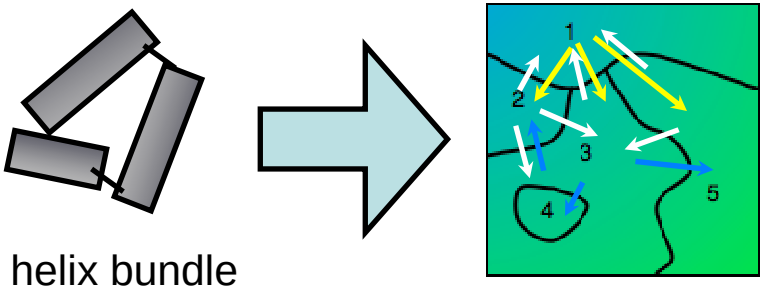
YES!

No

http://simtk.org/home/msmbuilder

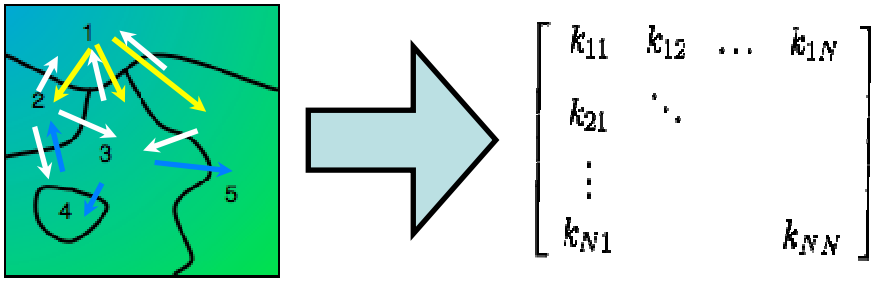
A statistical approach to simulation

1. Sample metastable states:
automatic algorithms to adaptively sample
and identify metastable states
via a ***kinetic*** clustering mechanism
(avoid one/low dimensional R.C.'s)

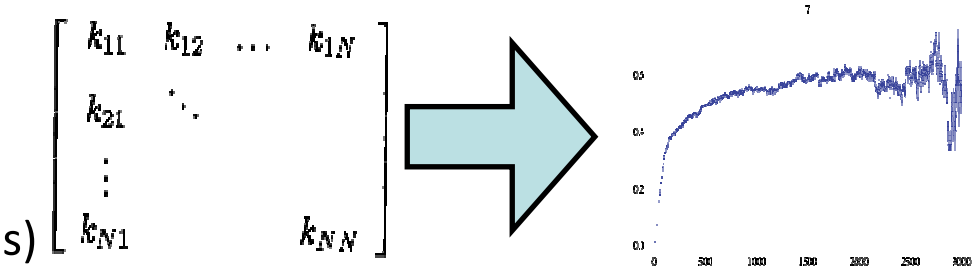


3 helix bundle

2. Build transition matrix:
use MD to sample transition probabilities
(ideally adaptively -- which allows MSMs to
be more efficient than very long runs)



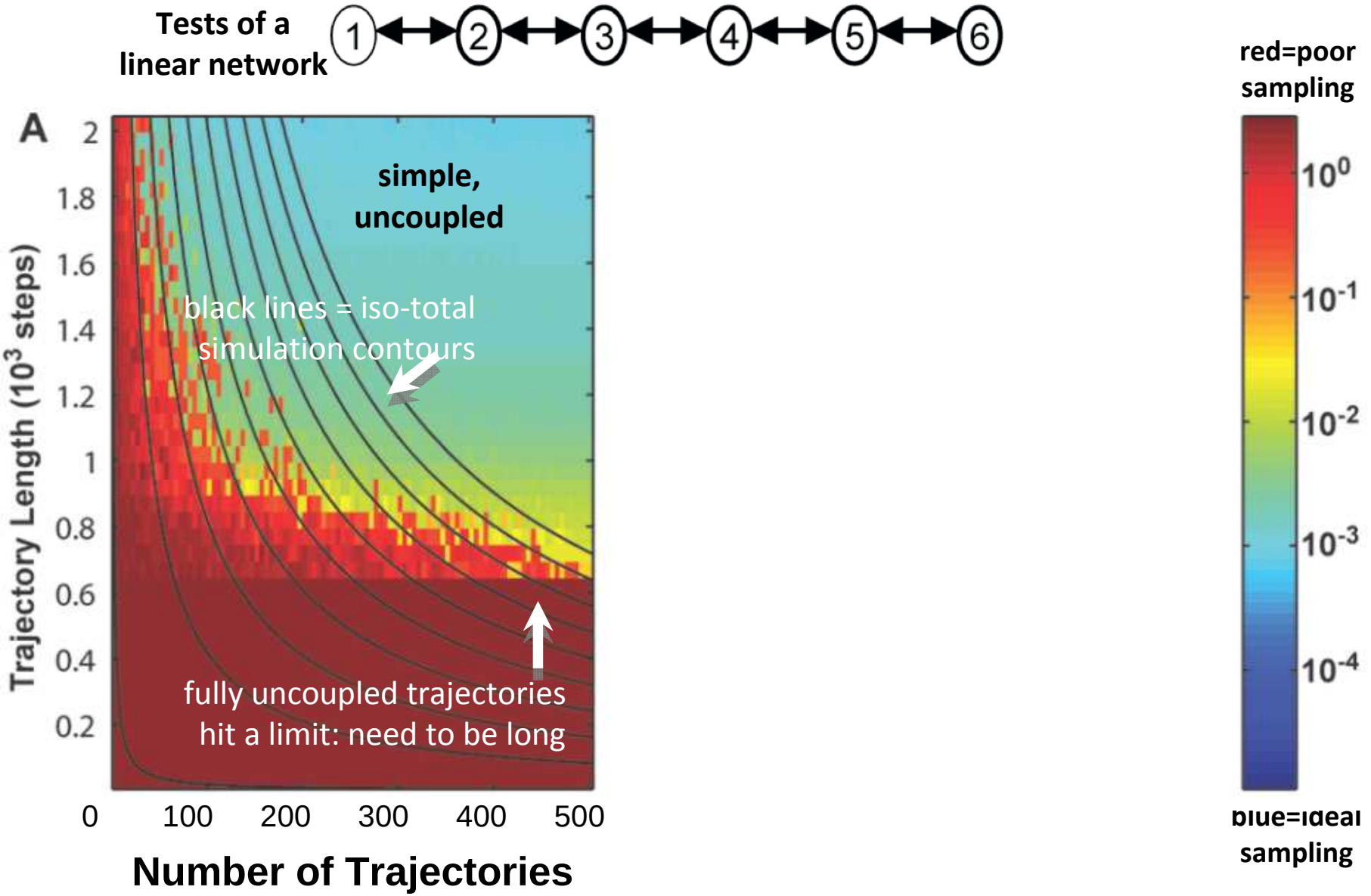
3. Use transition matrix:
transition matrix contains everything to
predict structure, thermodynamics, and
kinetics (built-in analysis via lumped MSM's)



also see the work of: Caflisch, Chodera, Deuflhard, Dill, Hummer, Noé, Pande, Pitera, Singhal-Hinrichs, Roux, Schütte, Swope, Weber

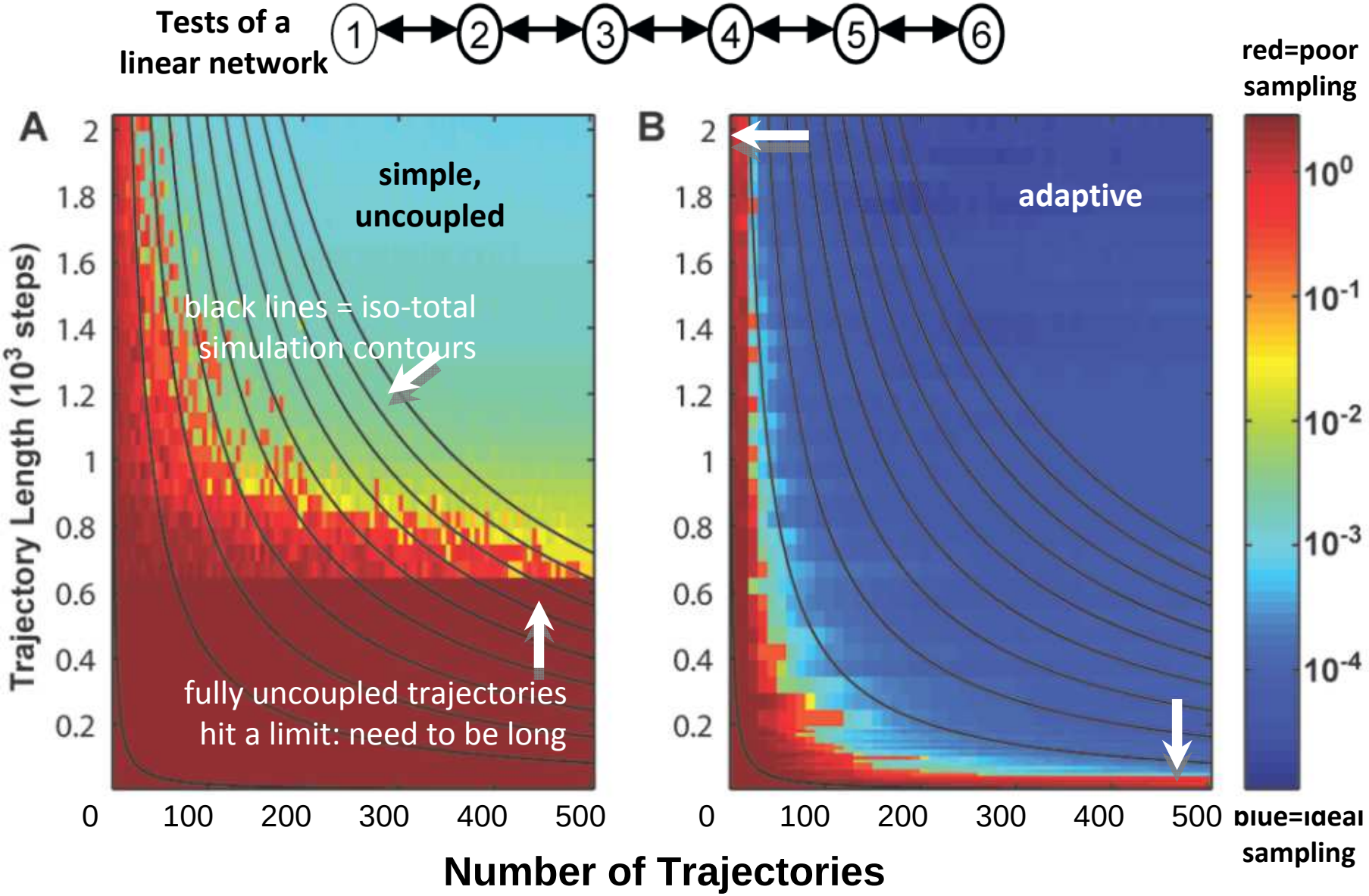
(Bowman, Pande)

Shorter trajectories can be *more* efficient



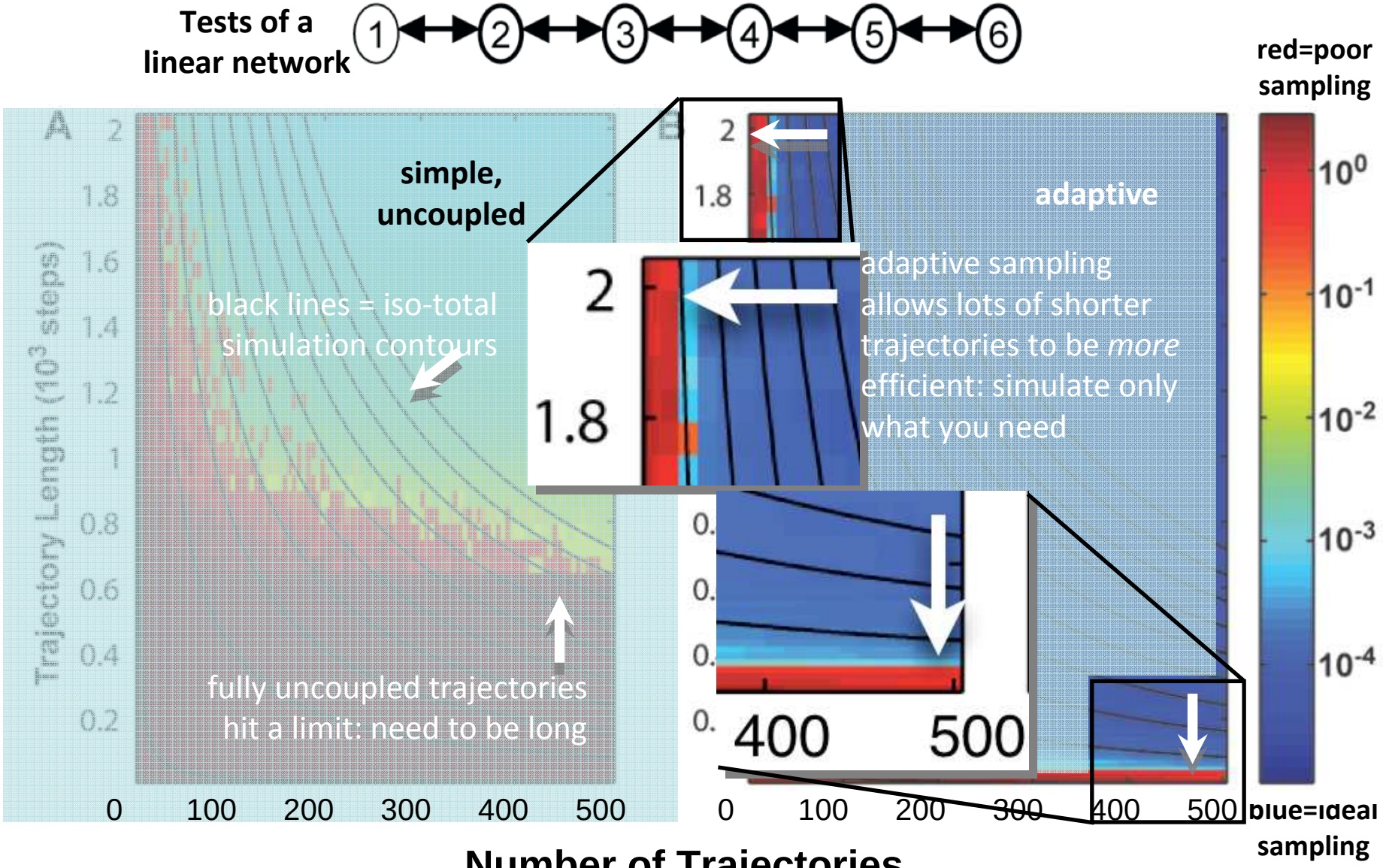
(Bowman, Pande)

Shorter trajectories can be *more* efficient



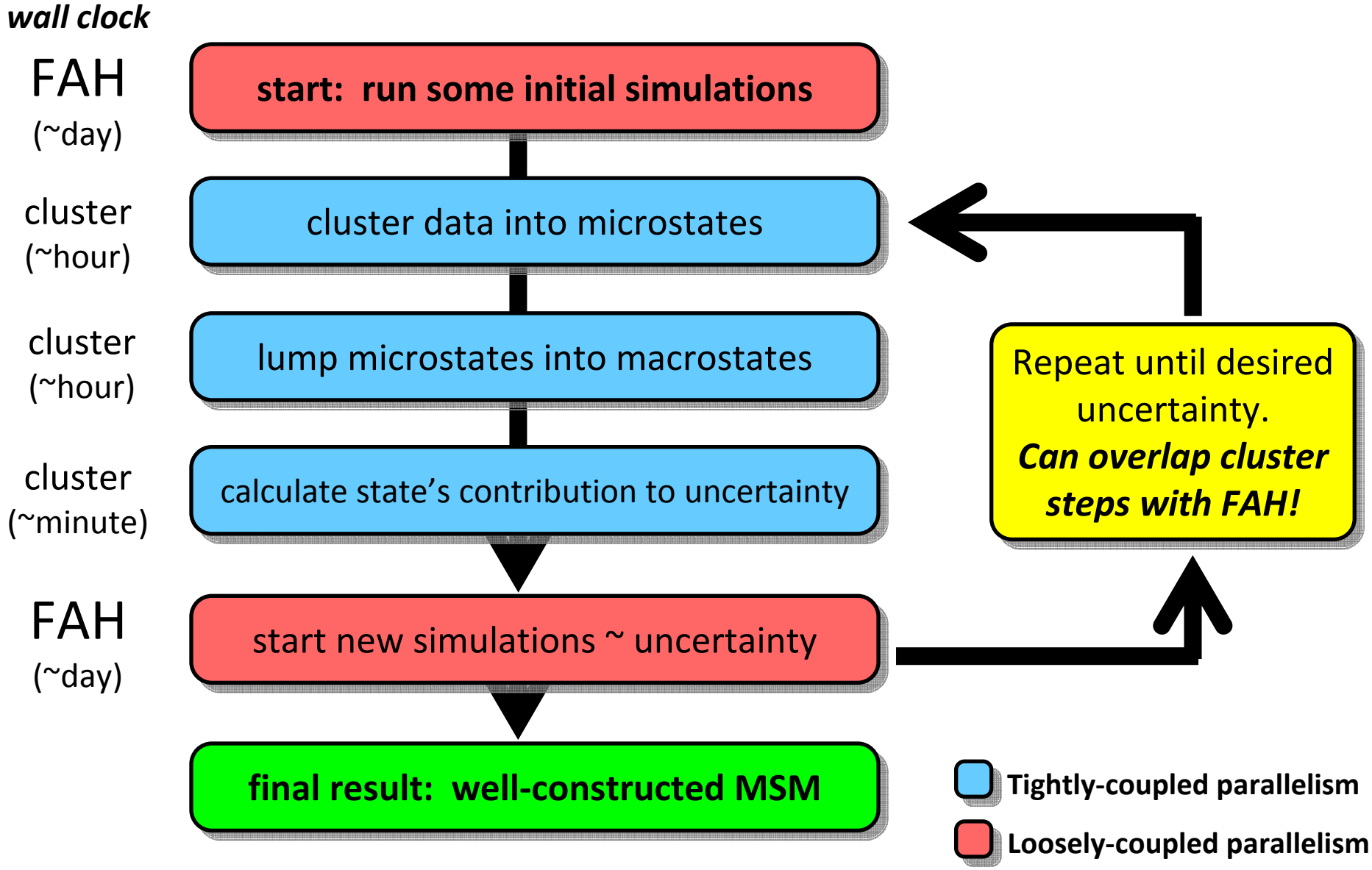
(Bowman, Pande)

Shorter trajectories can be *more* efficient



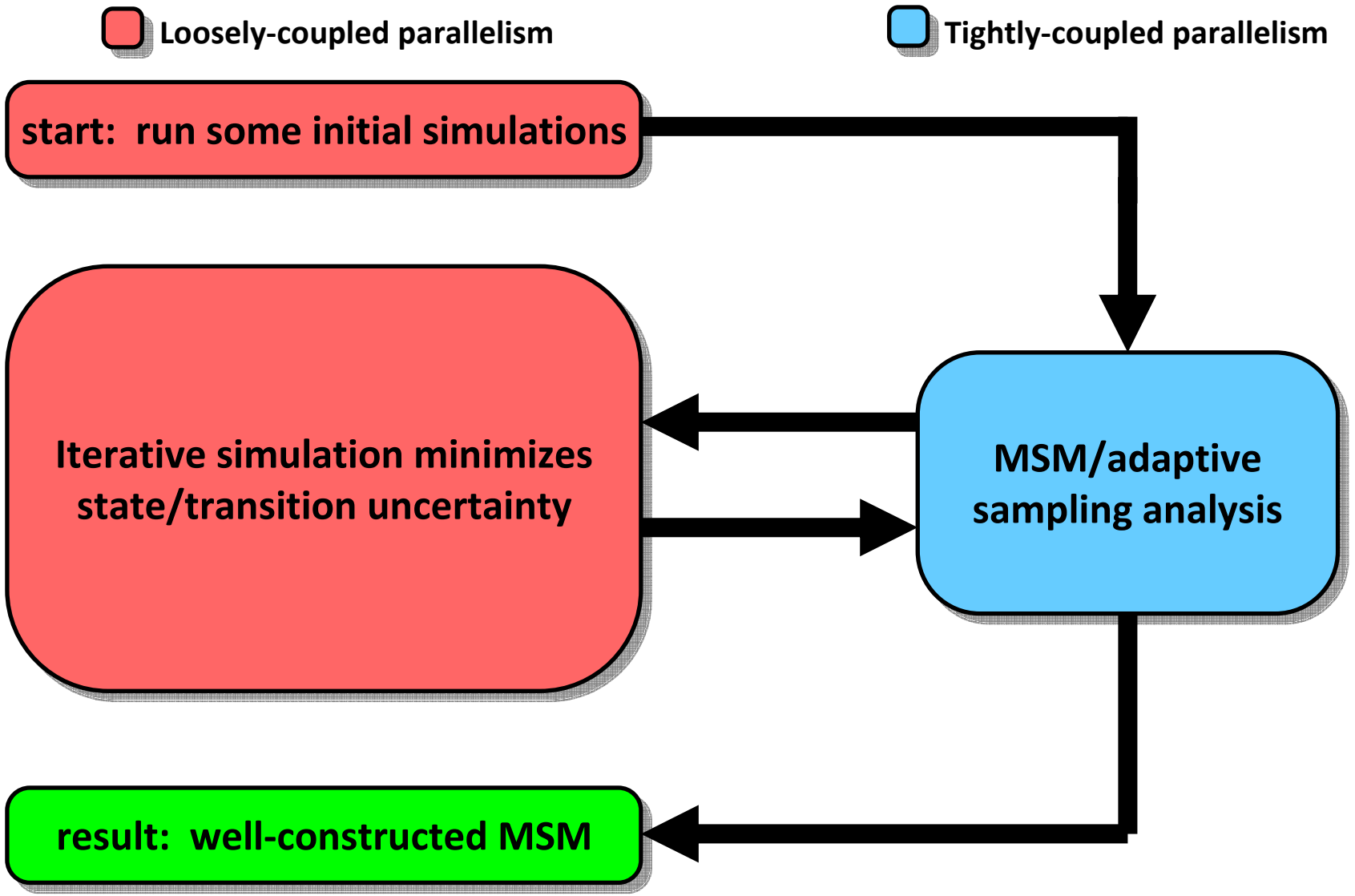
http://simtk.org/home/msmbuilder (Singhal, Bowman, Haque, Pande)

Adaptive Sampling – Parallel + Resilient



http://simtk.org/home/msmbuilder (Singhal, Bowman, Haque, Pande)

Adaptive Sampling – Parallel + Resilient

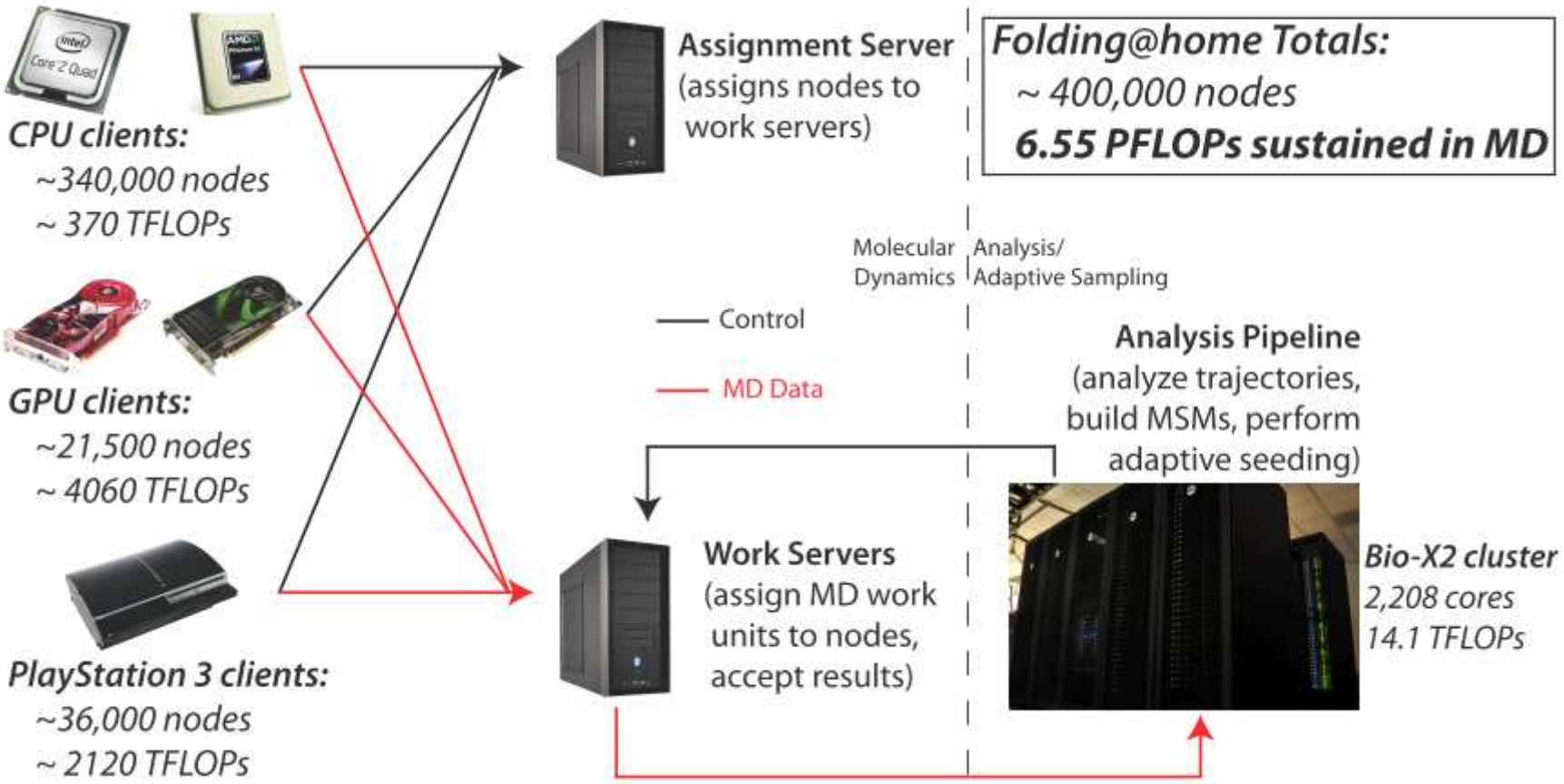


http://folding.stanford.edu (Pande Group)

Folding@home – Parallel + Resilient

Loosely-coupled parallelism

Tightly-coupled parallelism



“Real” Chemistry: States and Rates

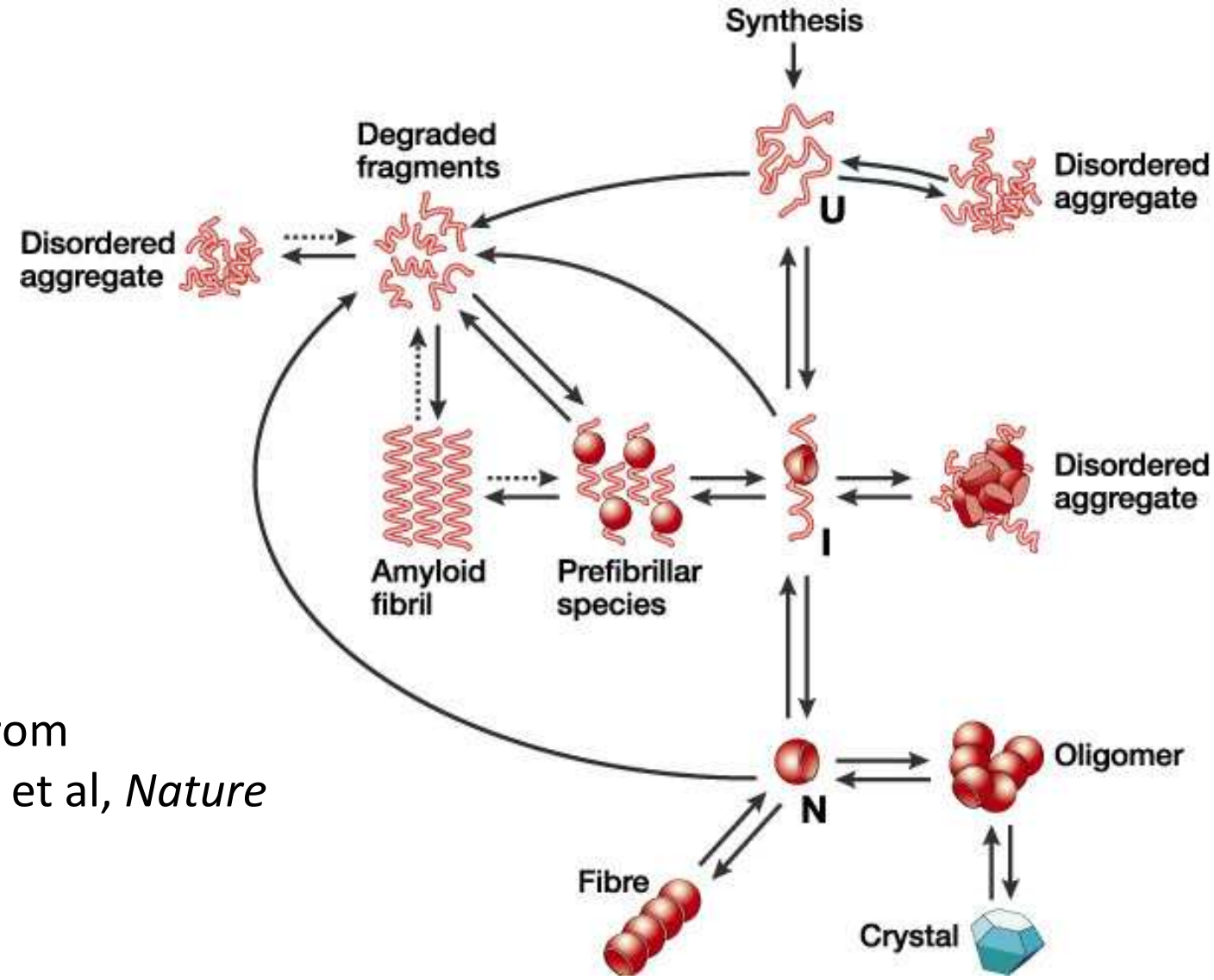
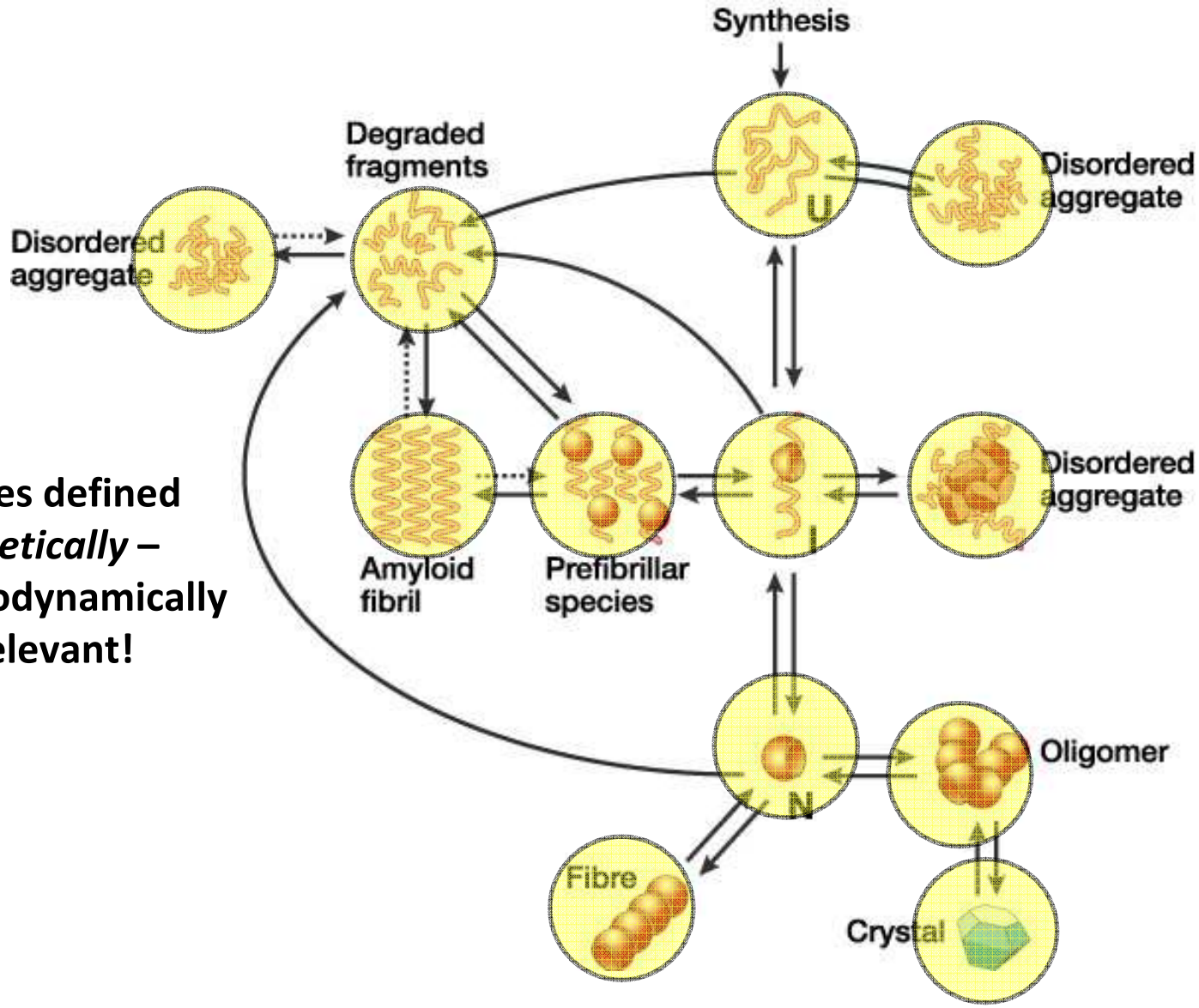


Figure from
Dobson, et al, *Nature*

http://simtk.org/home/msmbuilder

MSMs let us compute states and rates



States defined *kinetically* – thermodynamically relevant!

Acknowledgments

Stanford

- Vijay Pande (PI)
- Paul Novick
- Greg Bowman
- Kyle Beauchamp
- Randy Radmer
- Mark Friedrichs
- Peter Eastman

Collaborators

- John Chodera
- Del Lucent
- Pat Walters
- Kim Branson
- Erik Lindahl
- Michael Houston
- Scott LeGrand

- **Folding@home users**

VERTEX

AMD

NVIDIA.

Conclusions

- Future HPC will be driven by **heterogeneous architectures** and (even more) **massive parallelism**

- Applications need both **systems- and algorithms-level redesign** to be effective on next-generation HPC

- Our work shows a possible direction: GPU codes (**PAPER, OpenMM**) and new algorithms (**SCISSORS, MSMBuilder**) for cheminformatics and physical simulation

http://folding.stanford.edu

ihaque@cs.stanford.edu