

# Markov-state modeling of biomolecular systems



Toni Giorgino

National Research Council of Italy

[toni.giorgino@cnr.it](mailto:toni.giorgino@cnr.it)

[www.giorginolab.it](http://www.giorginolab.it)

[github.com/giorginolab/](https://github.com/giorginolab/)  
Markov-Tutorial-UniPD-2023



*Thesis projects available*

University of Padova c/o Prof. Fuxreiter  
Mar 27, 2023

# Introduction

- The aim of this class is to provide a practical overview of Markov state models in computational structural biology
- MSM emerging because:
  - reconstruct *kinetic* information\*, including state transition networks, from simulated trajectories
  - start from **unbiased simulations** (no *a priori* reaction coordinate hypothesis necessary)
  - microsecond-scale (high-throughput) trajectory data are becoming accessible (e.g., with GPUs)
- Success cases: *ab initio* folding, drug binding, peptide binding, ...

\* as well as the corresponding structures

# Motivation

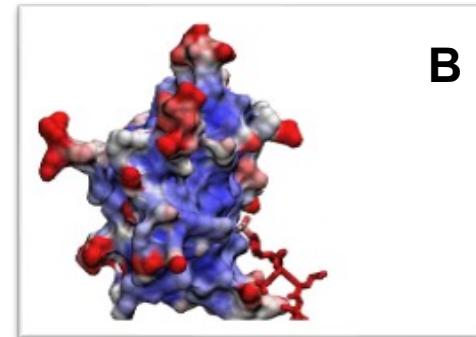
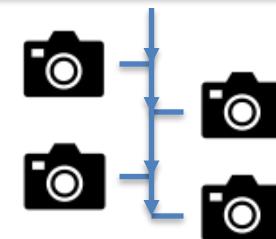
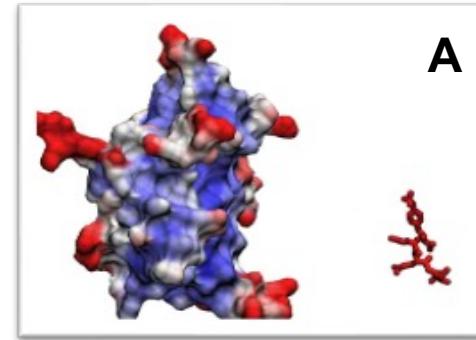
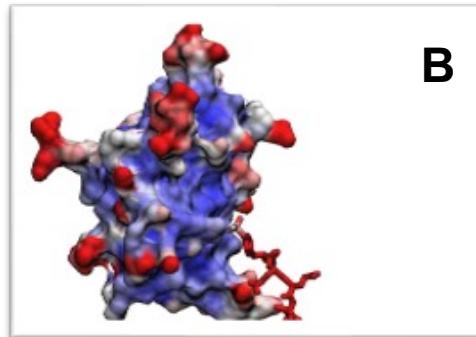
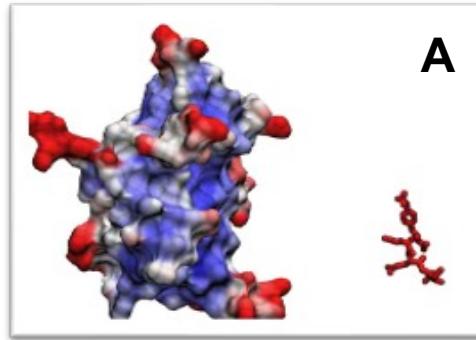
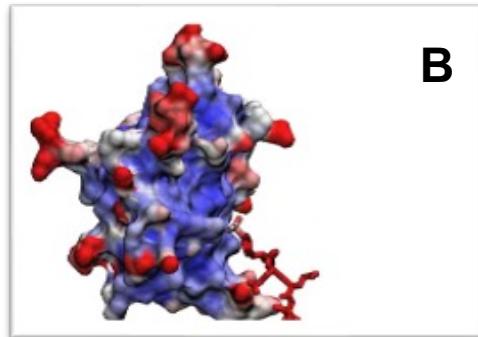
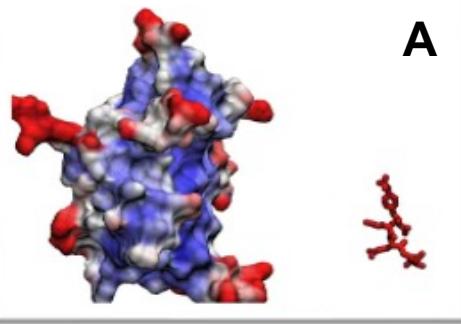
- Can we use an *ensemble* of simulations to estimate dynamic behaviours which happen on timescales longer than each of the observed trajectories?
- In other words, can we leverage several “short-sighted” (non-equilibrium) observations to extrapolate long-time behaviour?

# The general idea

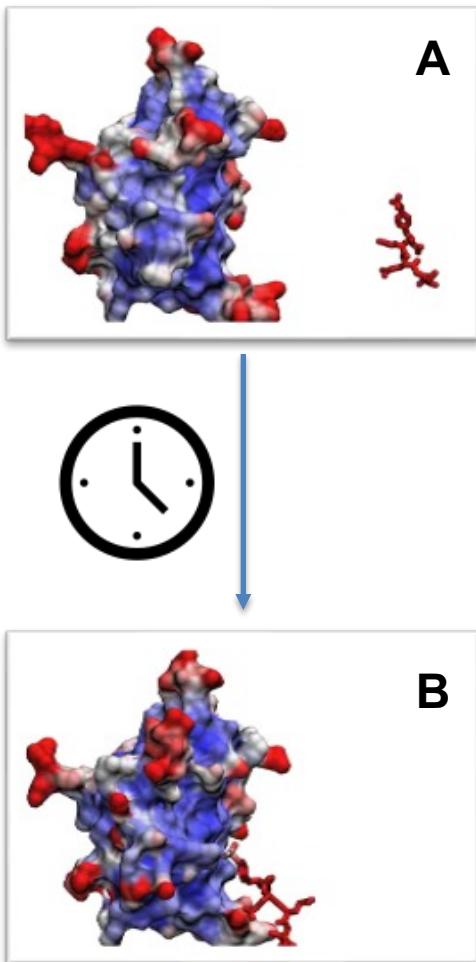
## Unbiased sampling

## Biased sampling

## Markov-state modeling



## Unbiased sampling



**System evolves in time\*.**

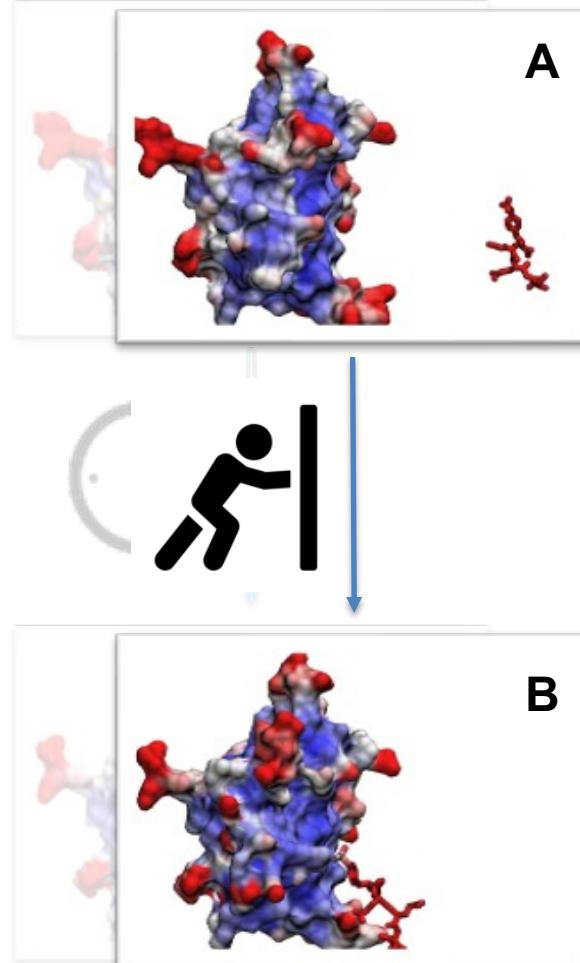
We do not “guide” the system with biases in any particular direction.

**Pro:** Kinetics are “true”

**Con:** Slow. Observations  
(=sufficient sampling)  
may well be unfeasible.

\* Under the influence of internal and external forces designed to approximate “real” ones.

## Biased sampling



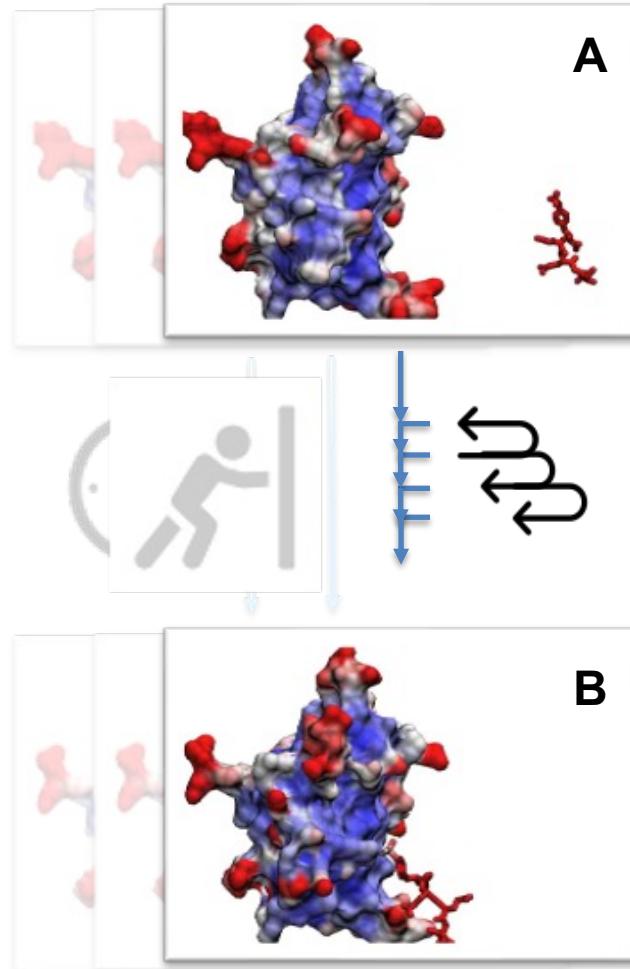
Various strategies\* exist to “push” the system *towards* desired states

Pros: Sampling is accelerated..  
Can recover  $\Delta G$ .

Cons: Biasing “direction” must be set, often not trivial.  
Kinetics (in general) are altered.

\* Examples: metadynamics, steered MD, ...

## Markov-state modeling



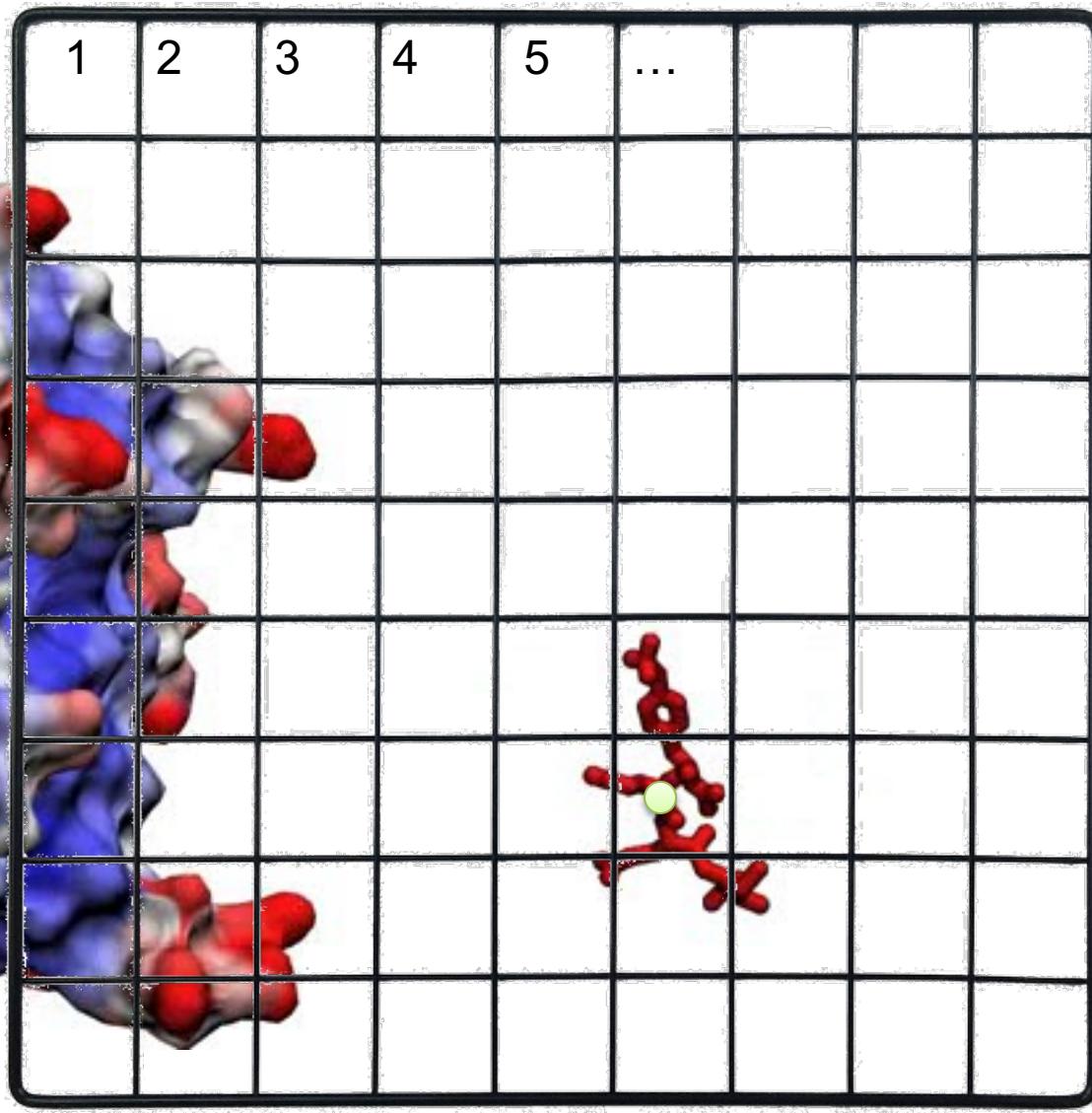
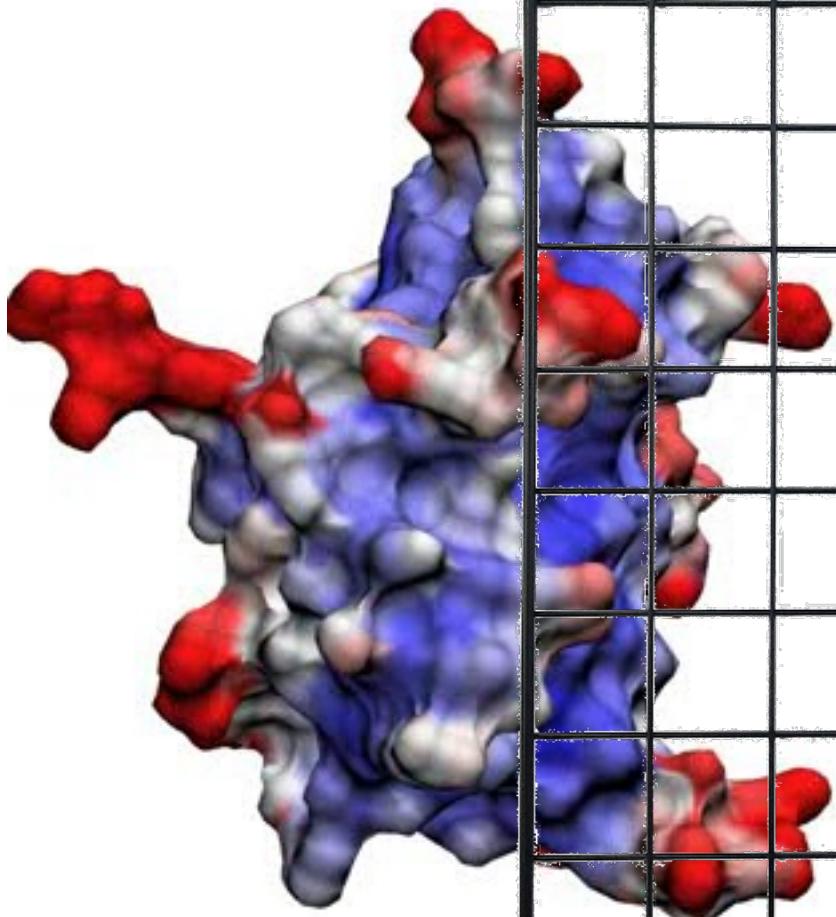
Decompose a system in multiple states. Use shorter trajectories to compose a statistical\* picture.

Pros: True kinetics.  
No need to pre-set a reaction coordinate.

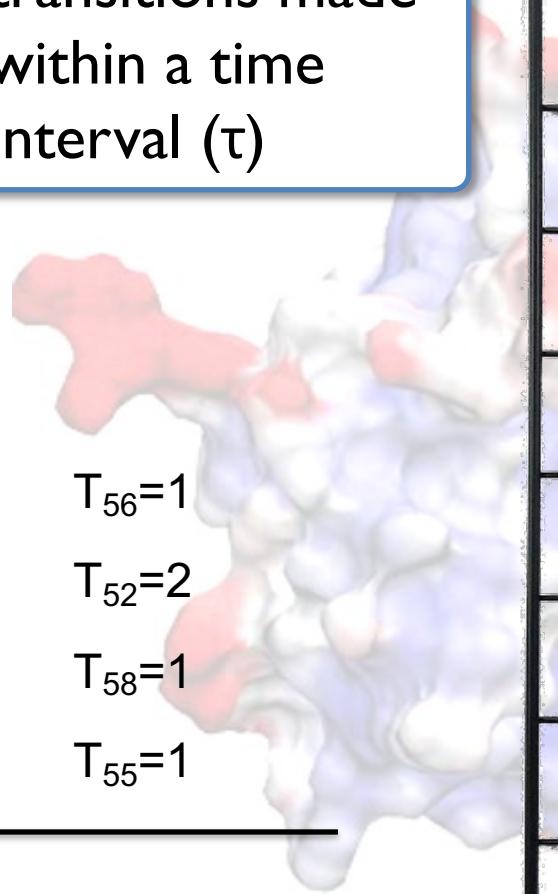
Cons: Non obvious (but software helps).  
Sampling still needed for convergence.

\* We'll see Markov-based formalism, but others exist, such as *milestoning*

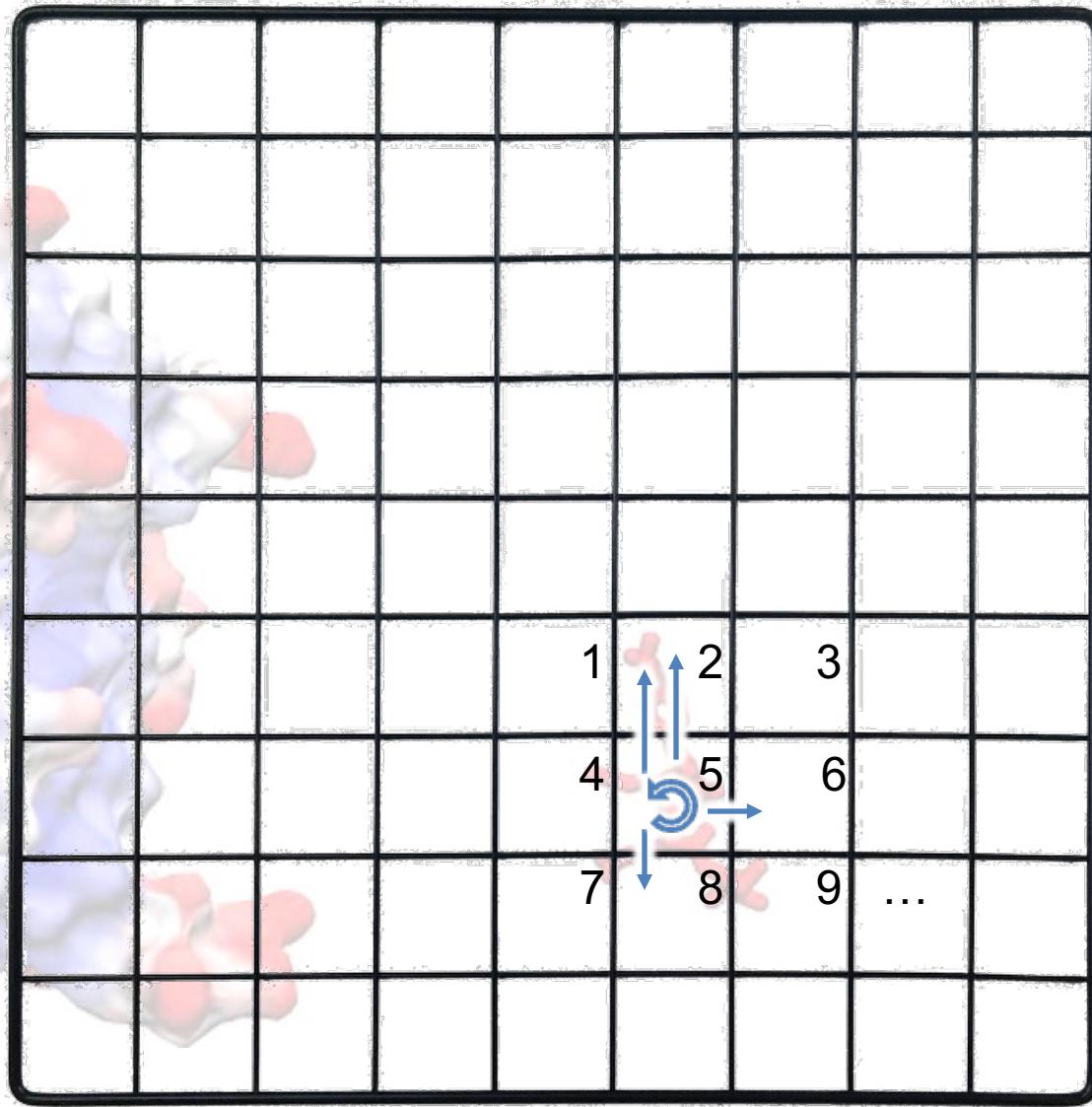
# **Decomposing the state space**



Count the state transitions made within a time interval ( $\tau$ )



Total:  $N_5=5$



$$T_{56}=1$$

$$p_{56} = 1/5 = 0.2$$

$$T_{52}=2$$

$$p_{52} = 2/5 = 0.4$$

$$T_{58}=1$$

$$p_{58} = 1/5 = 0.2$$

$$T_{55}=1$$

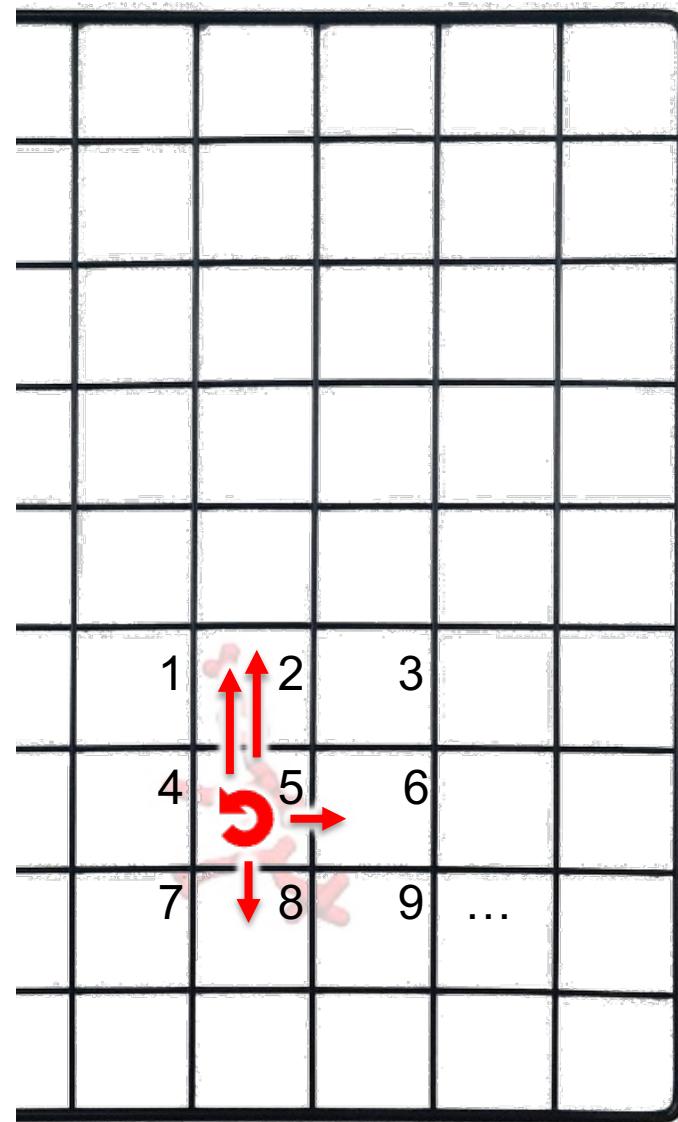
$$p_{55} = 1/5 = 0.2$$

$$N_5=5$$

$$p_{5*} = \sum_i p_{5i} = 1$$

1	2	...	5	6	7	8	...
5		0.4		0.2	0.2		0.2

Row 5 of the  
*transition probability matrix P*

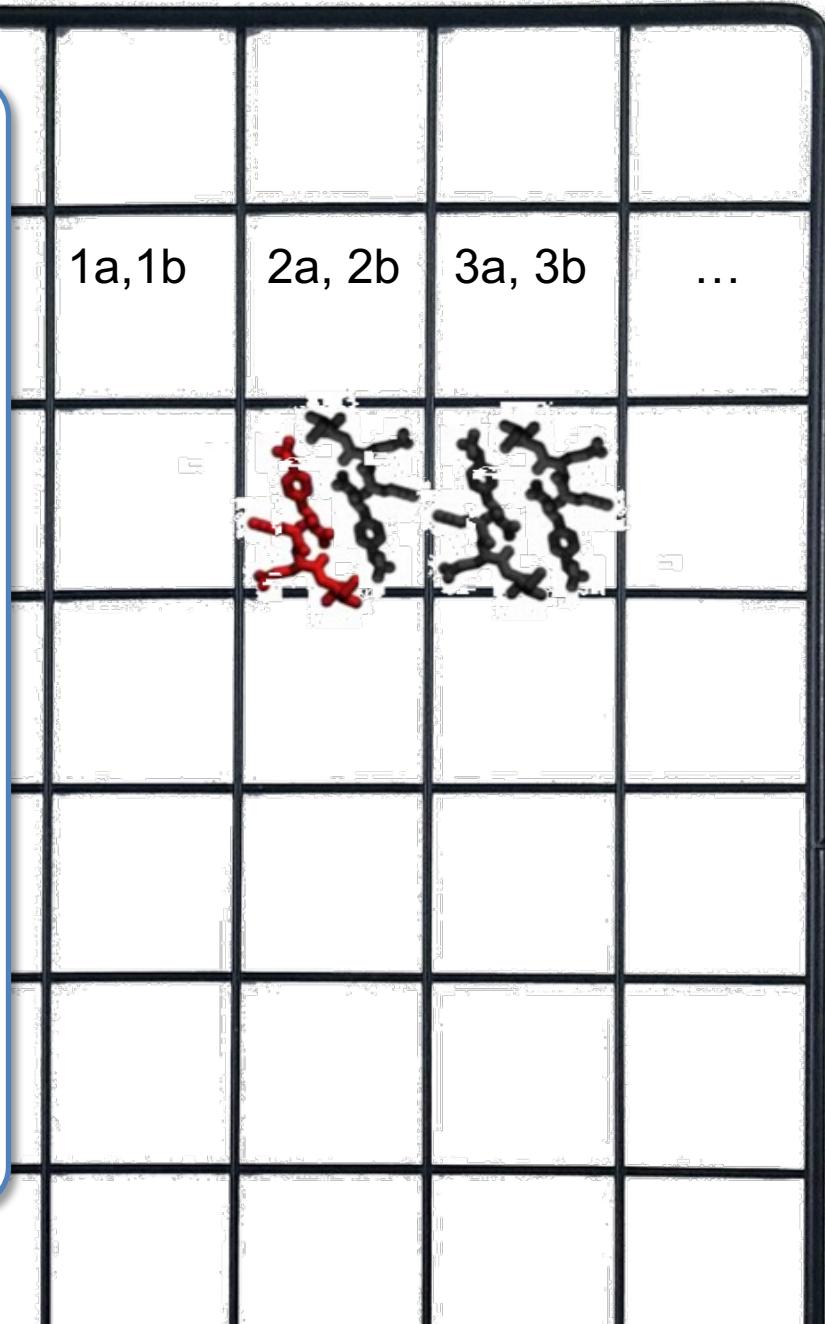


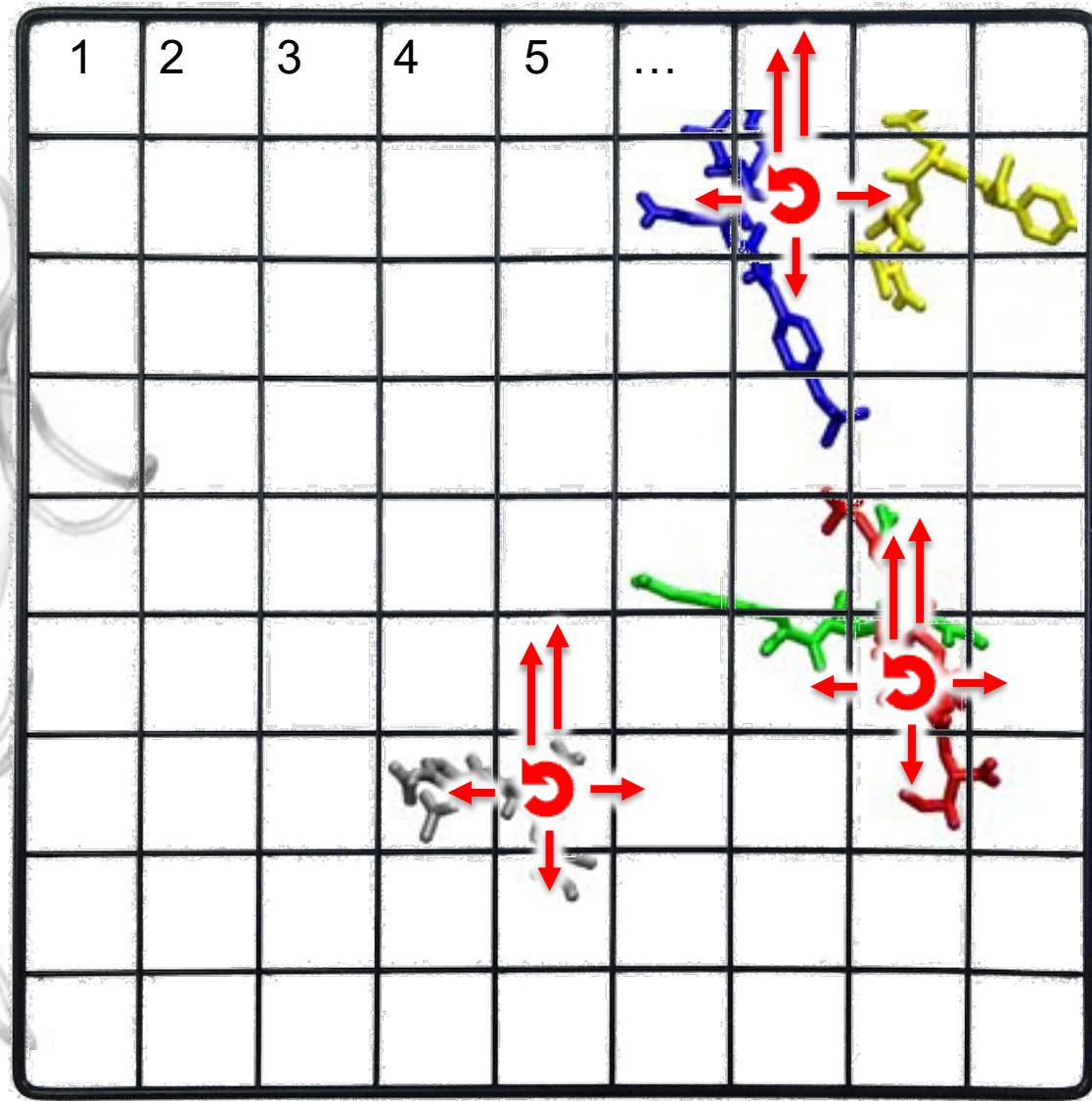
## Important

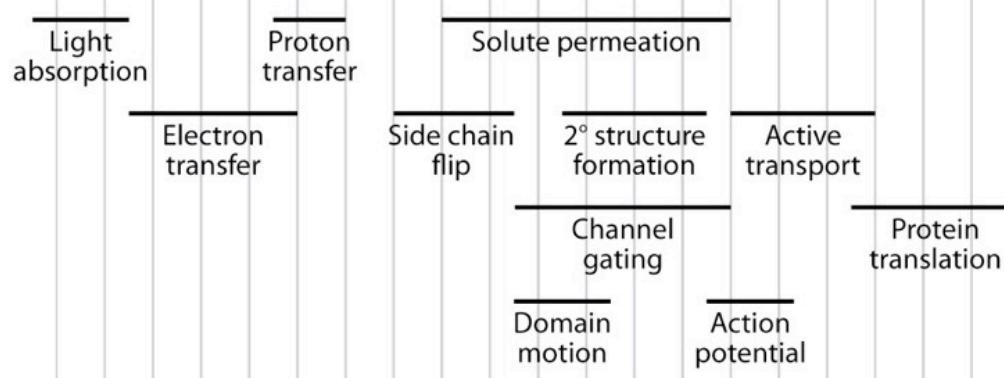
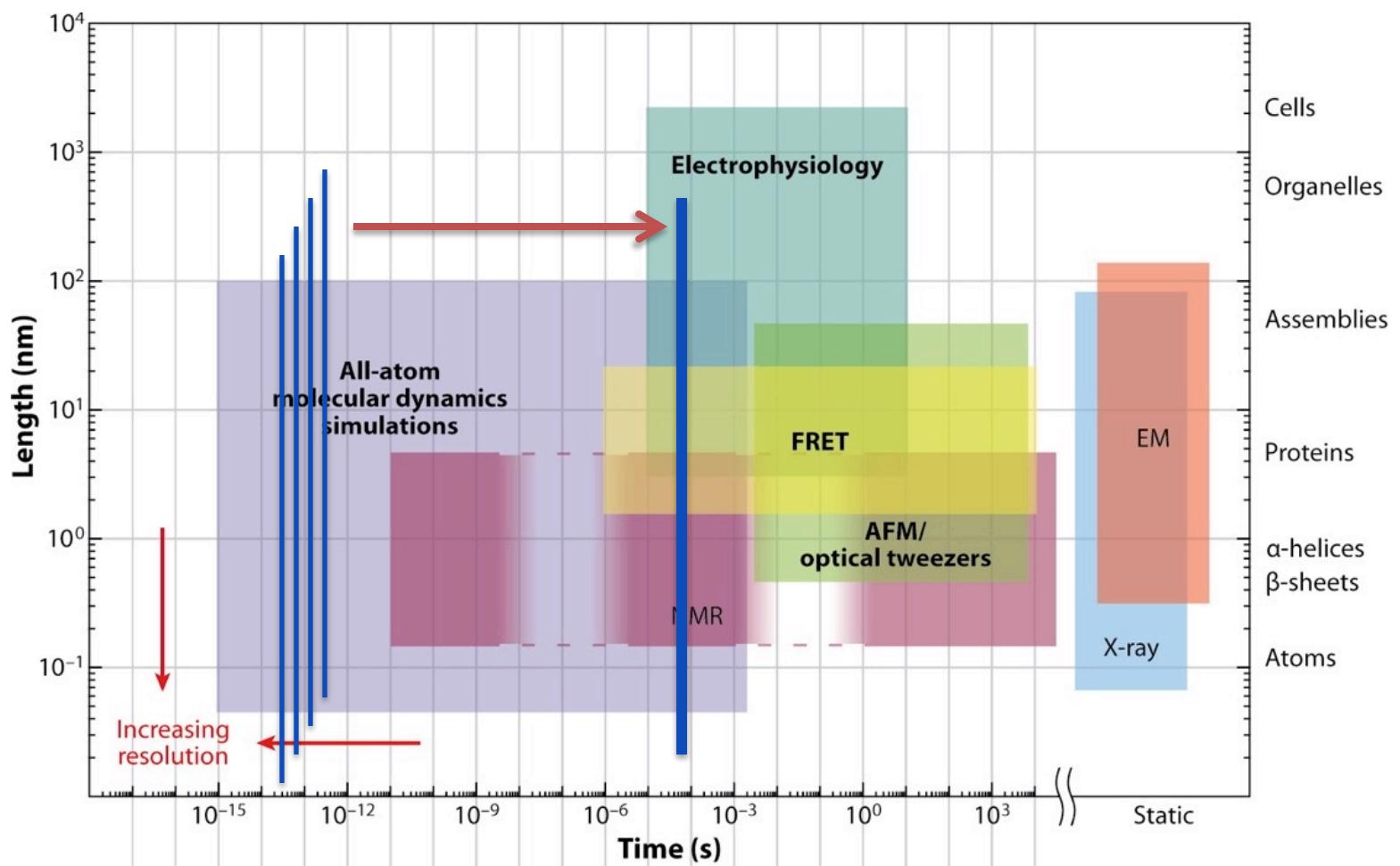
States are more general than a “location in space”.

For example, different conformations of the same structure may count as different states.

The partition (discretization) of the system’s configuration space in states is *up to the user*.







**Why do we care?**

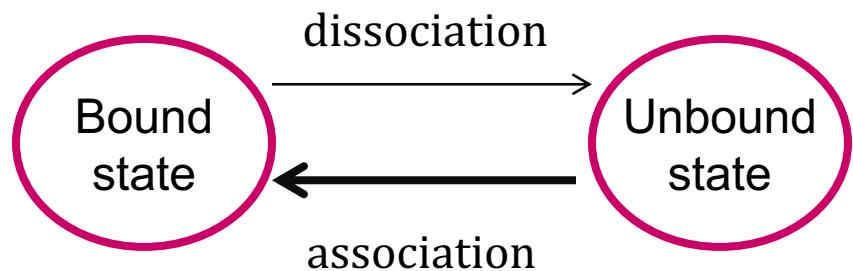
# Why do we care?

Because equilibrium probabilities are the free energy of binding ( $\Delta G$ ).

$\Delta G$  is an important determinant of drug potency.

$$K_D = p_U/p_B (\propto [P][L]/[PL]) \\ = \exp(-\Delta G/RT)$$

$$\Delta G = -RT \ln K_D$$



# Kinetics

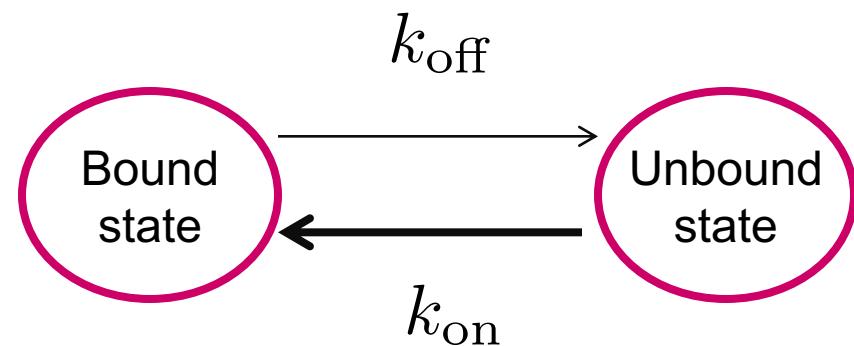
But the real breakthrough is that we can use the *dynamic* information, computing the *kinetics* as well. This is why we are doing *unbiased dynamics* in the first place!

Binding kinetics is important for rational drug design as well (e.g. related to the time spent in complex with receptors).

$$K_D = k_{\text{off}}/k_{\text{on}}$$

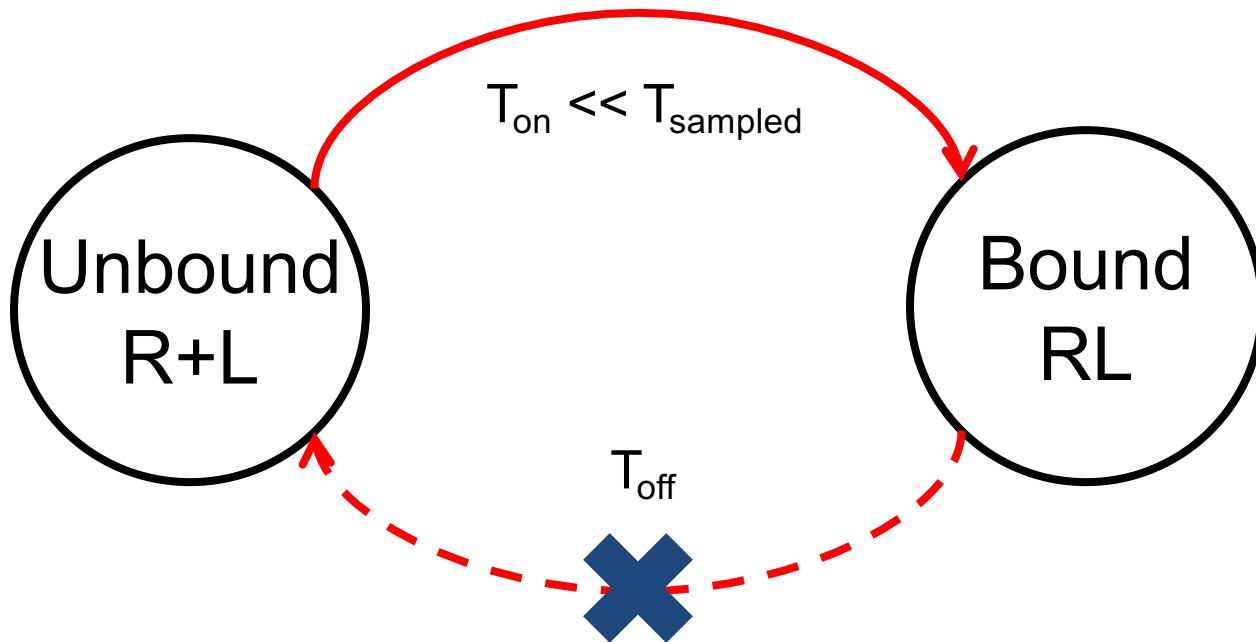
$$\propto \tau_{\text{on}}/\tau_{\text{off}}$$

$$\sim t_U^{(B)}/t_B^{(U)}$$



# Binding as an *irreversible* two-state model

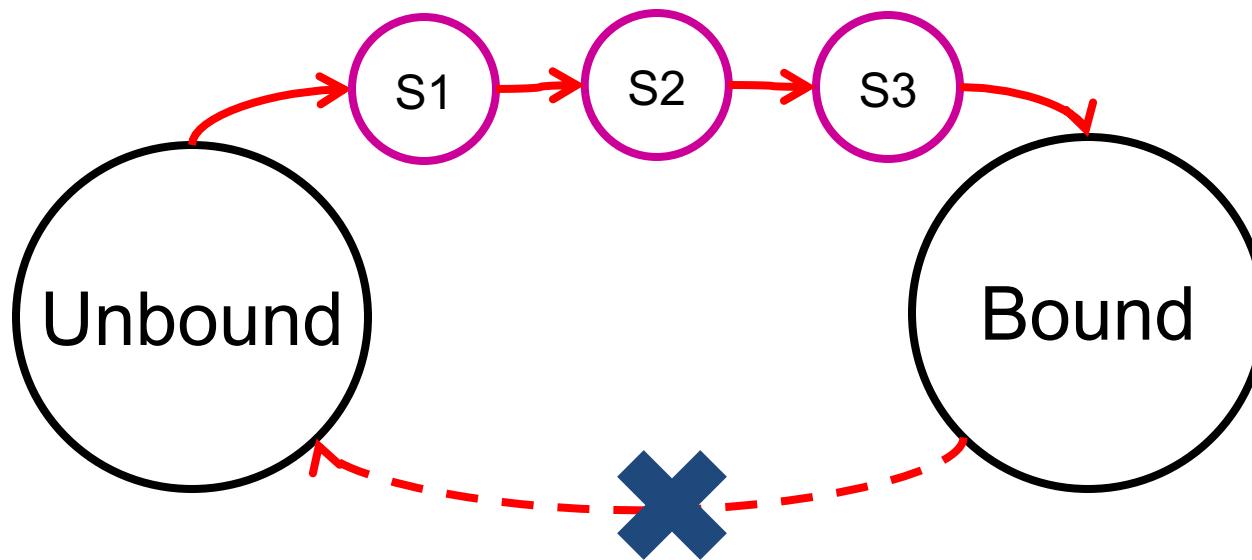
On the sampled timescales...



$$T_{event} \ll T_{on} \ll T_{sampled} \ll T_{off}$$

# Idea

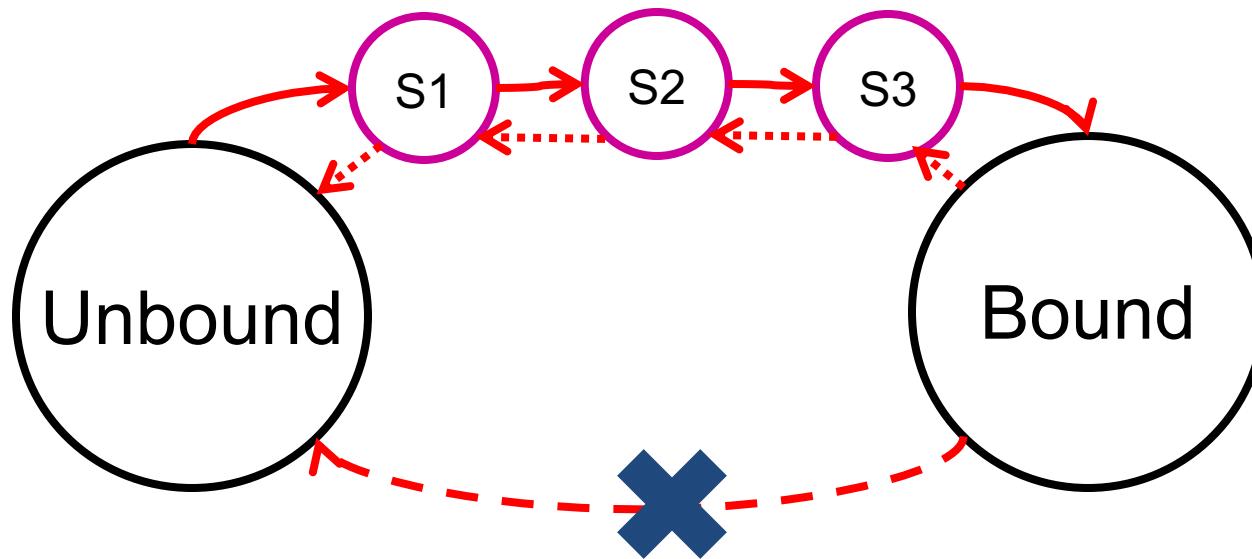
- We introduce several *intermediate steps*



$k_{\text{on}}$  reconstructed combining rates of different intermediate steps

# Idea

- We introduce several *intermediate steps*
- Compute MFPT

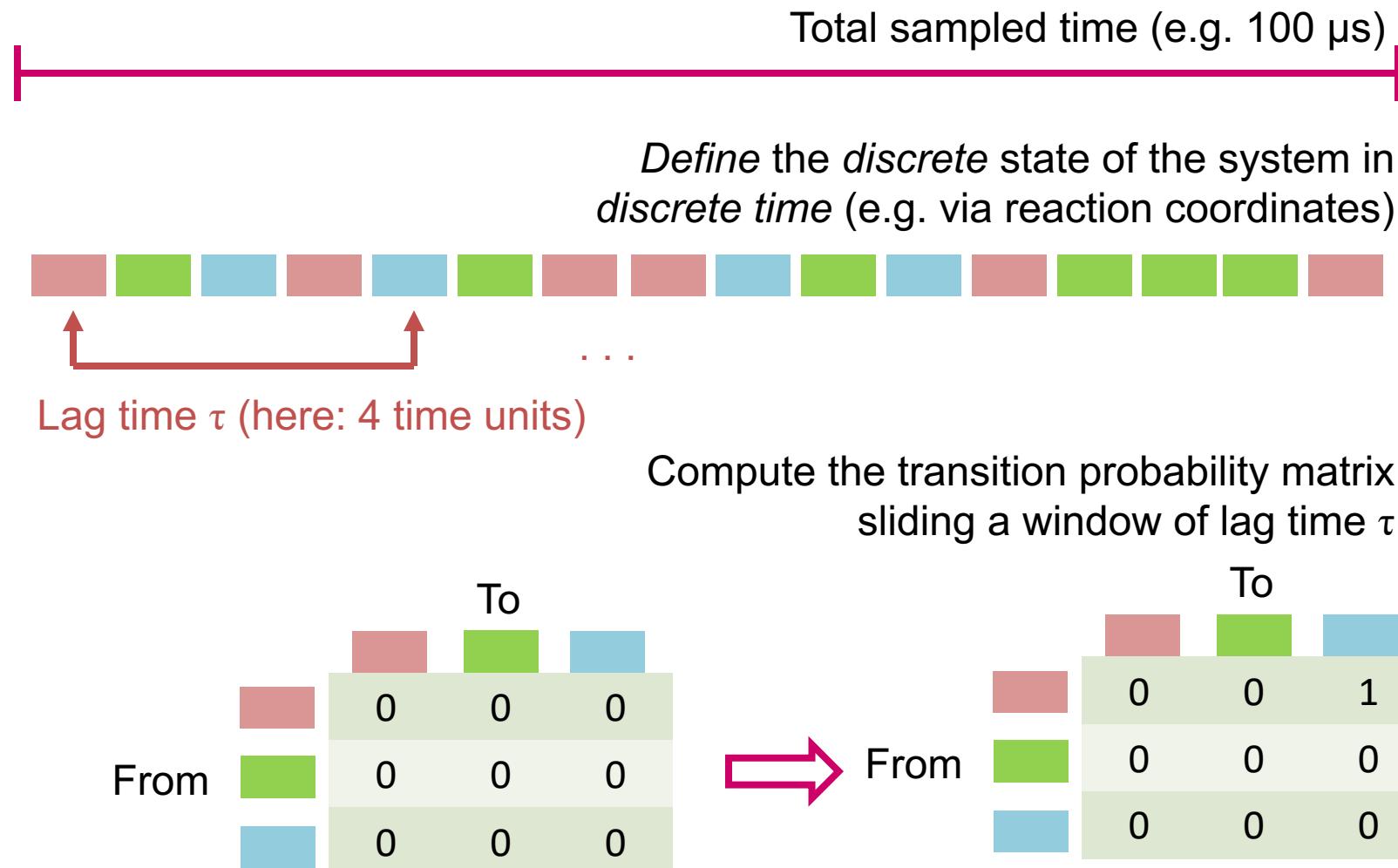


$k_{\text{on}}$  reconstructed combining rates of different intermediate steps

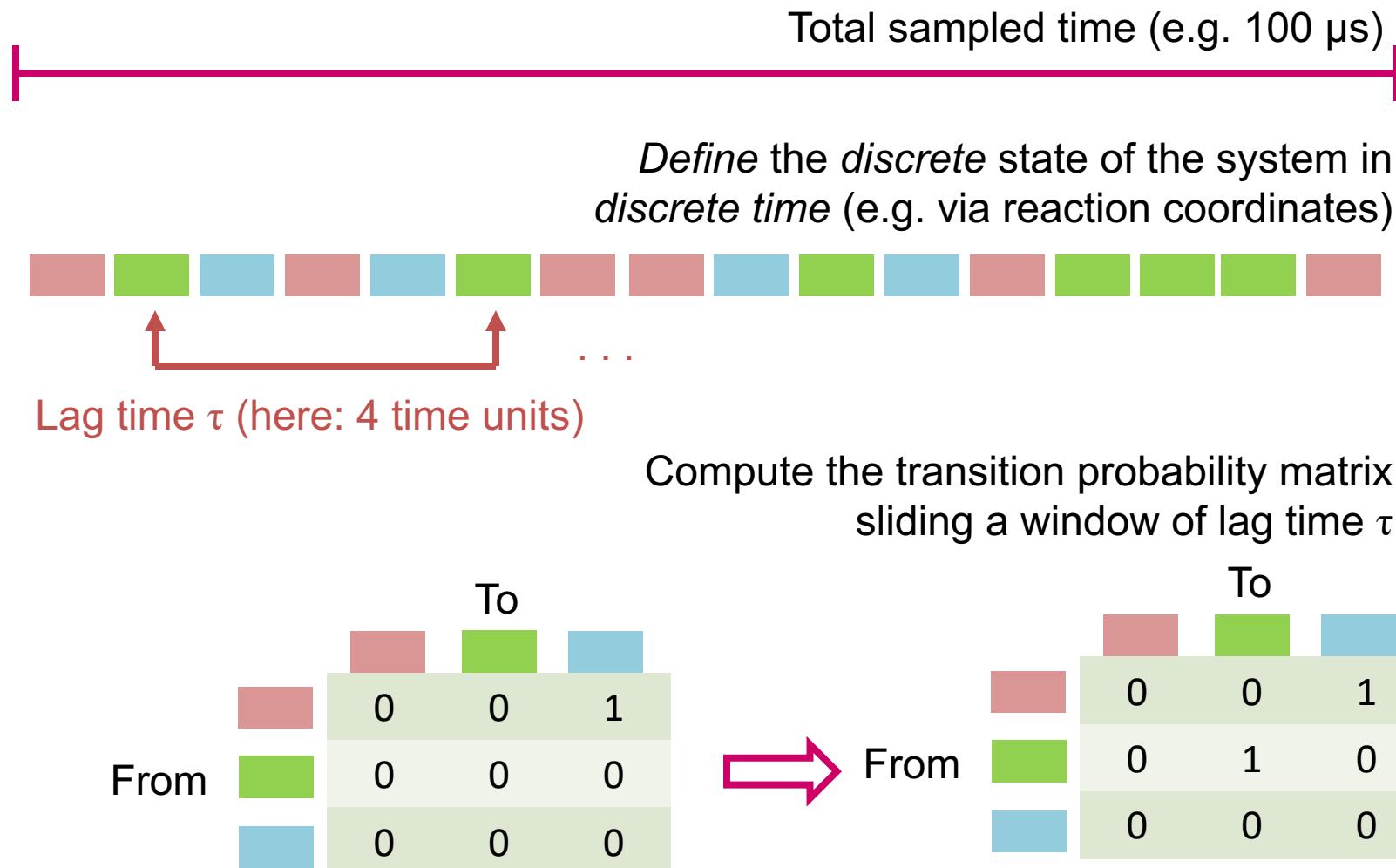
$k_{\text{off}}$  may also be reconstructed, if states are fine-grained!

# **Learning transition matrices from trajectories**

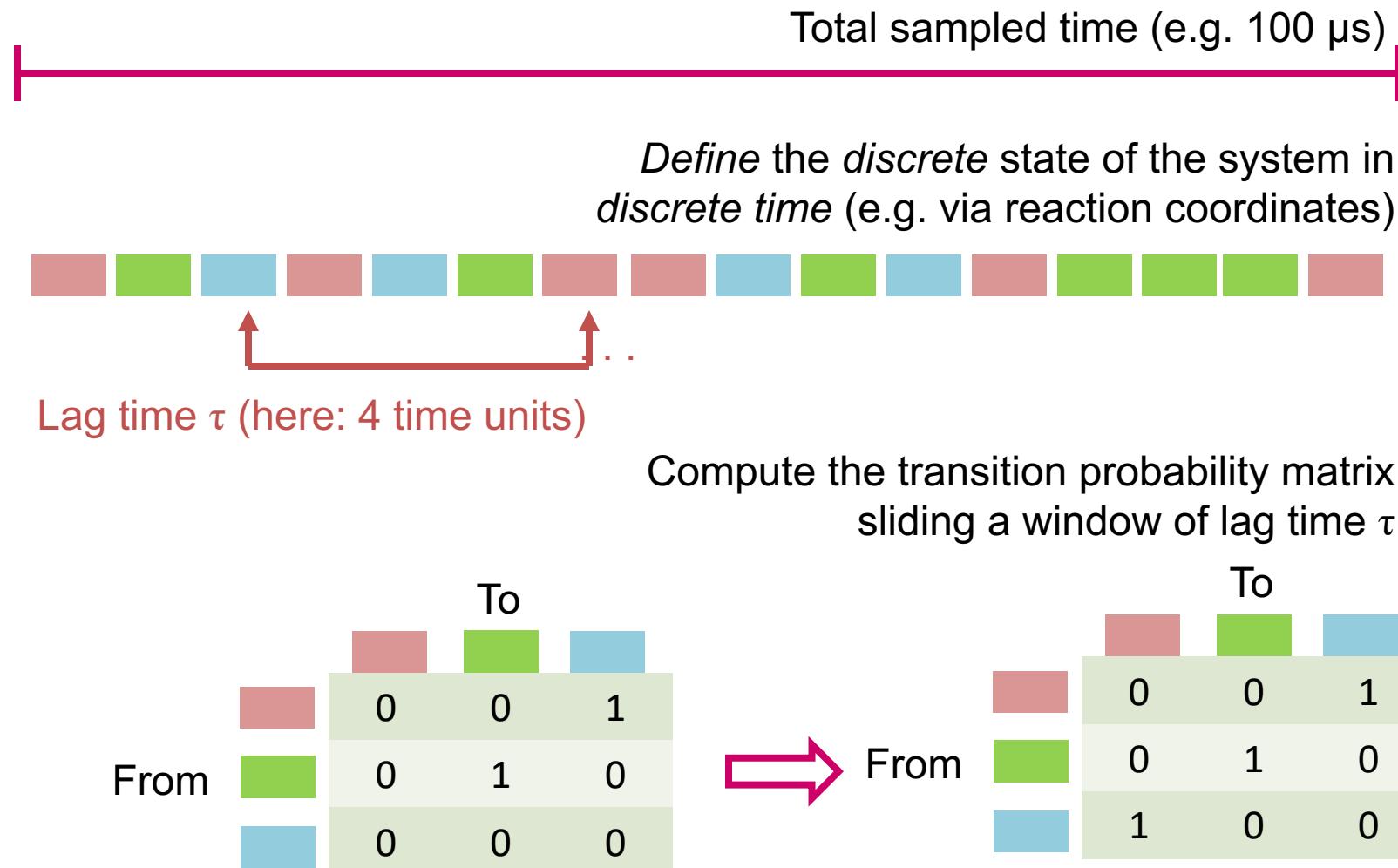
# Markov models



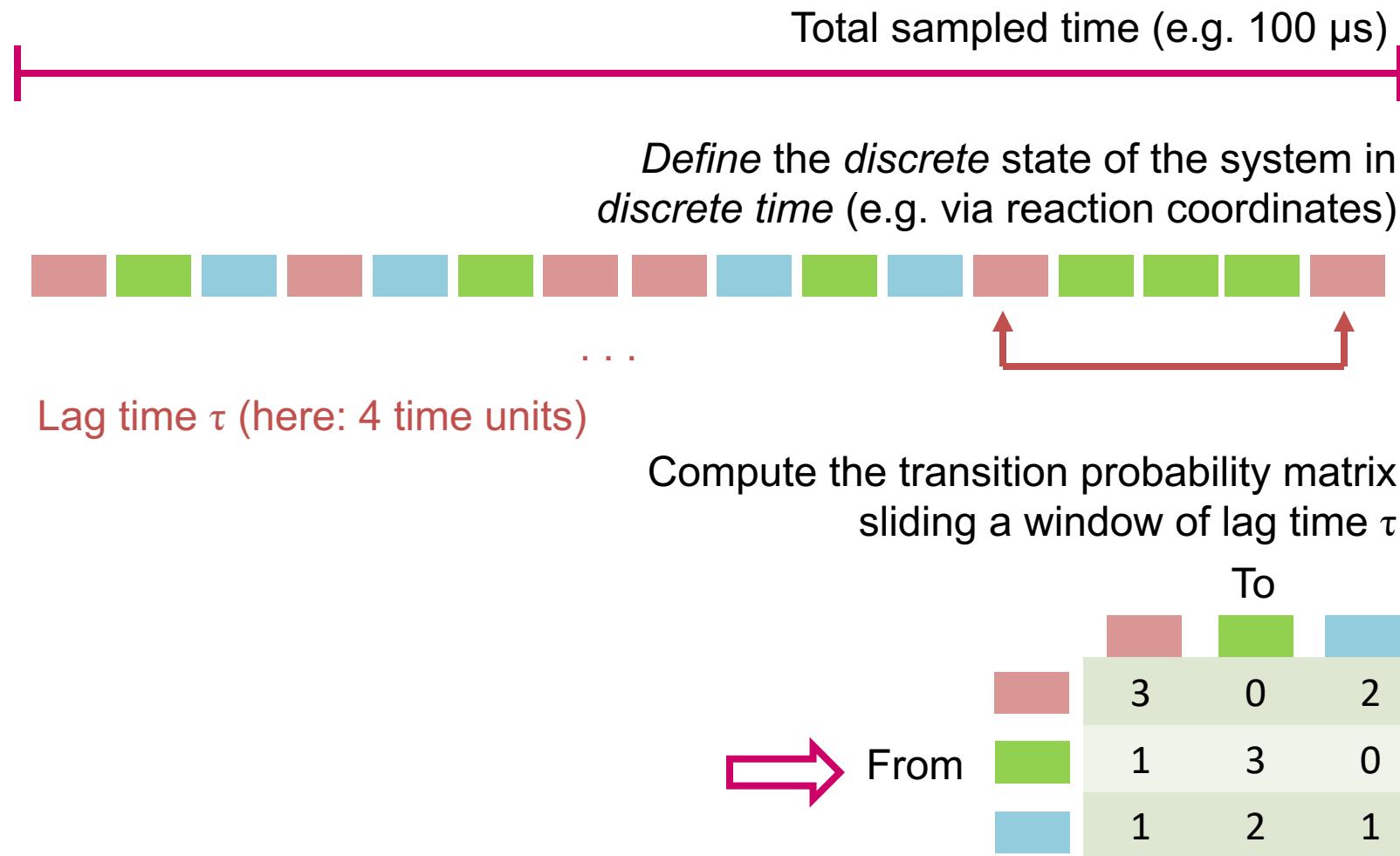
# Markov models



# Markov models



# Markov models



# Markov models

Total sampled time (e.g. 100  $\mu\text{s}$ )



Define the discrete state of the system in discrete time (e.g. via reaction coordinates)



Lag time  $\tau$  (here: 4 time units)

Transition counts

		To	
		3	0
From	Red	3	0
	Green	1	3
Blue	1	2	1

Transition probabilities

		To		$\sum_j$
		3/5	0	2/5
From	Red	3/5	0	2/5
	Green	$\frac{1}{4}$	$\frac{3}{4}$	0
Blue	$\frac{1}{4}$	$\frac{2}{4}$	$\frac{1}{4}$	1

Normalize by rows

$P_{ij}$

Repeat until the end of the trajectory.

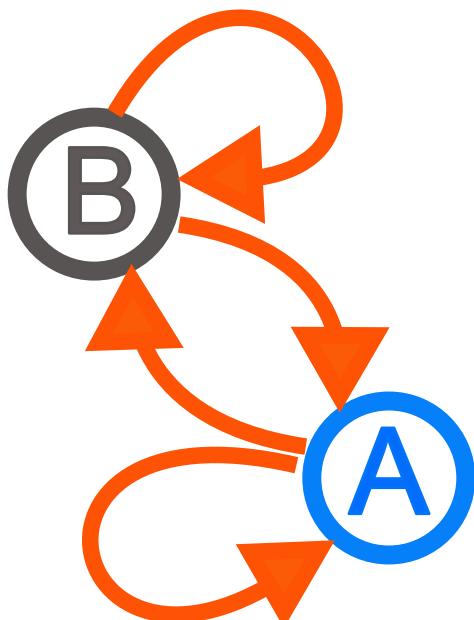
Note the Markovian assumption:  
transition probabilities **do not** depend  
on history (neither, for us, on time).

(Note how “Early” and “late” events  
are squashed in the same matrix.)

# **Discrete-time Markov Chains**



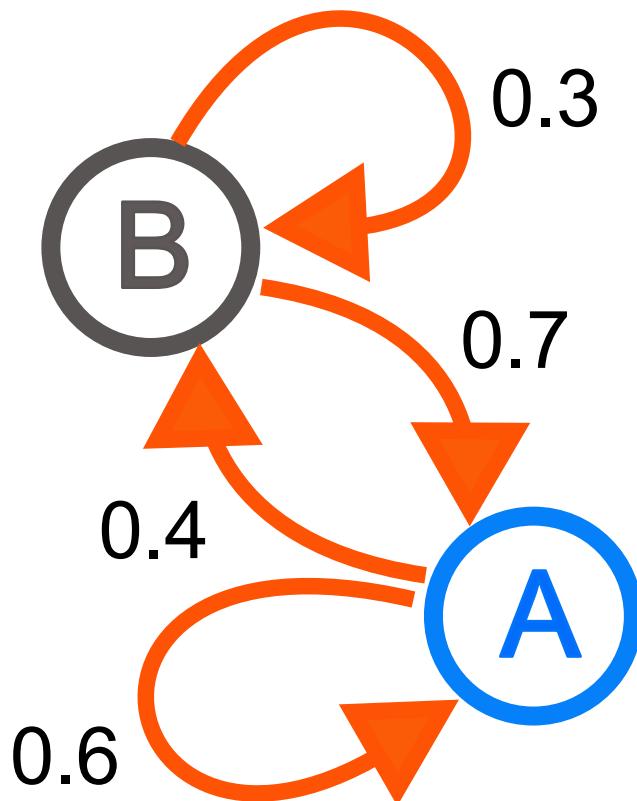
Andrei Markov  
1856-1922



# Discrete Time Markov Chains

- A **random** process.
- The system's state is a **discrete** variable.
- It undergoes transitions between states at uniformly-spaced (**discrete**) time points.
- Transition probabilities do not depend on the previous history of states (**memorylessness**).

# Transition probability matrix

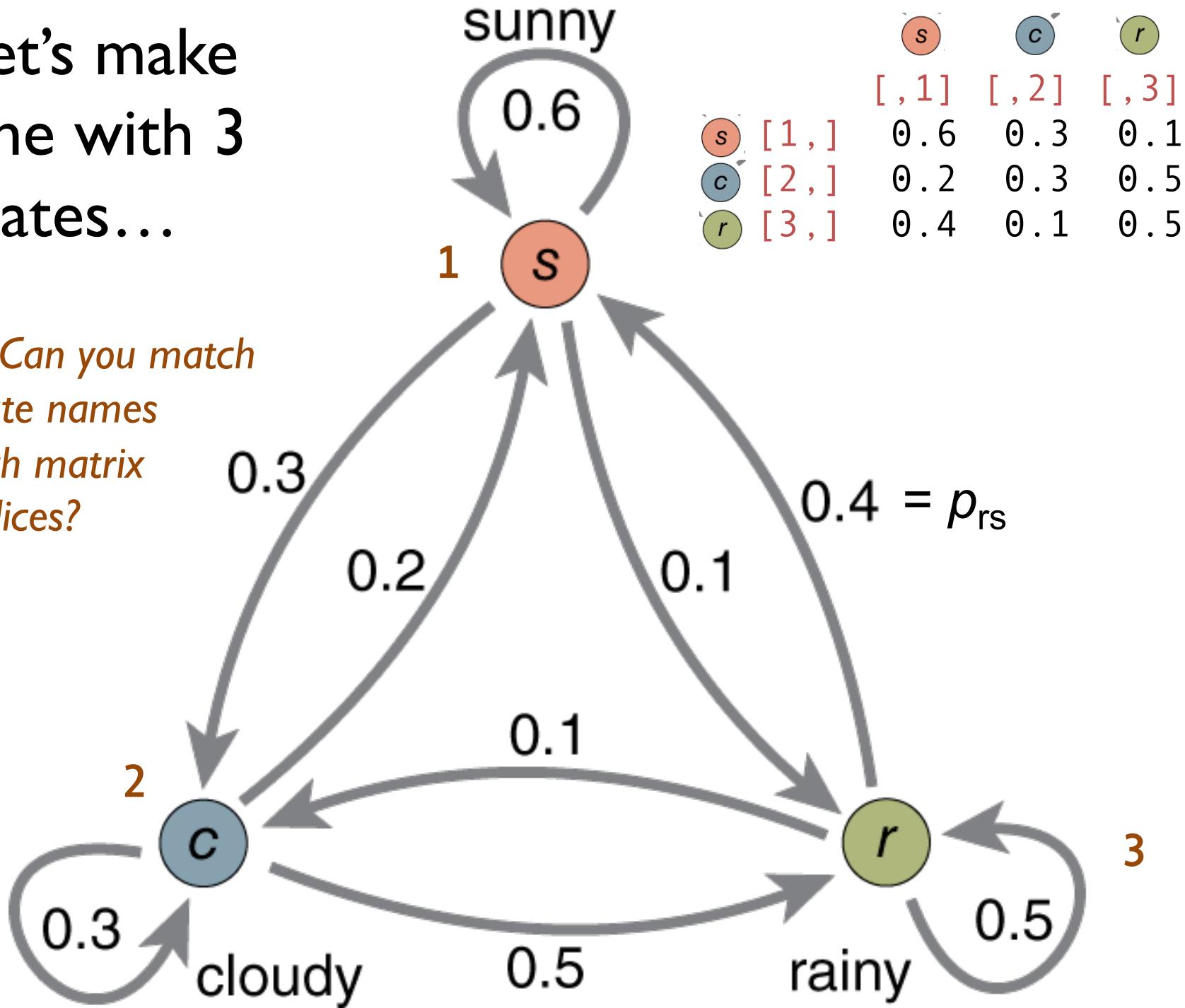


$P_{ij}$  = Probability to change state  
from  $i$  to  $j$  at each time point

$$P_{ij} = P(X_t = j | X_{t-1} = i)$$

		$j$
		A      B
$i$	A	0.6      0.4
	B	0.7      0.3

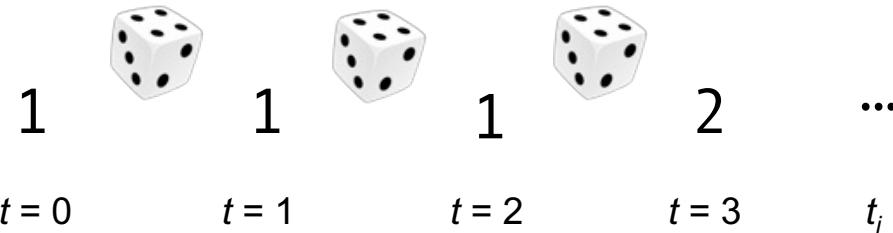
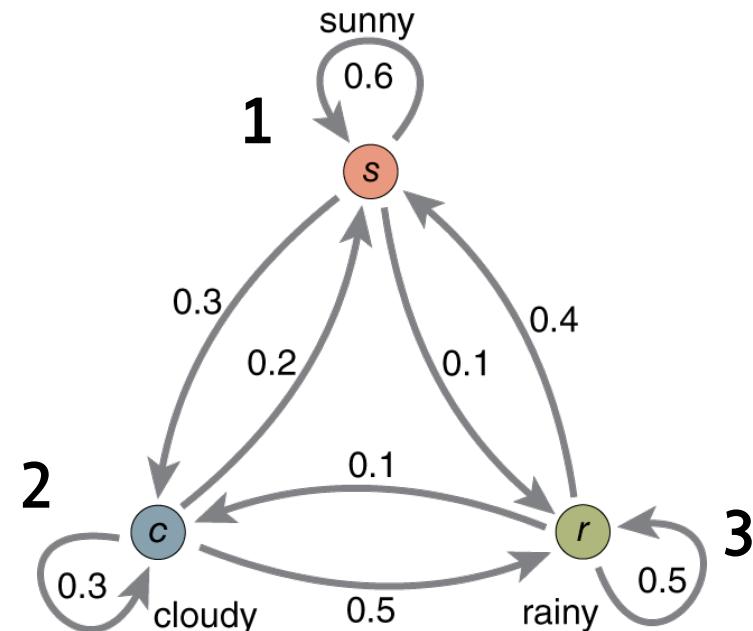
Let's make  
one with 3  
states...



# Let's generate samples...

Start e.g. from state 1  
(we'll use numbers from now on).

Play the game and follow the Markov chain for many (discrete) time steps.

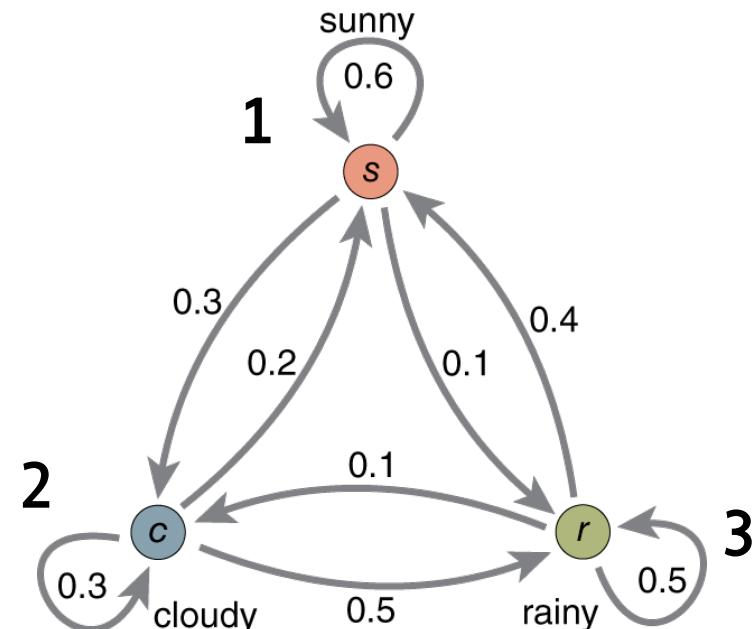


1 1 1 2 2 3 3 1 1 1 2 2 2 2 1 1 1 1 1 1 3 1 1 1 1 1 2 3 1 1 3...

# Equilibrium probabilities

Question: if we play the game a very long time, what fraction of time would we spend in each state?

- That is the *asymptotic* (equilibrium) distribution.
  - I.e., the probability of finding the system in a given state.



1 1 1 2 2 3 3 1 1 1 2 2 2 2 1 1 1 1 1 1 1 3 1 1 1 1 1 2 3 1 1 3...  
(1000 times)

1	2	3
431	220	349



$$\begin{aligned} p_\infty(1) &= 0.431 \\ p_\infty(2) &= 0.220 \\ p_\infty(3) &= 0.349 \end{aligned}$$

$$\sum_i p_\infty(i) = 1$$

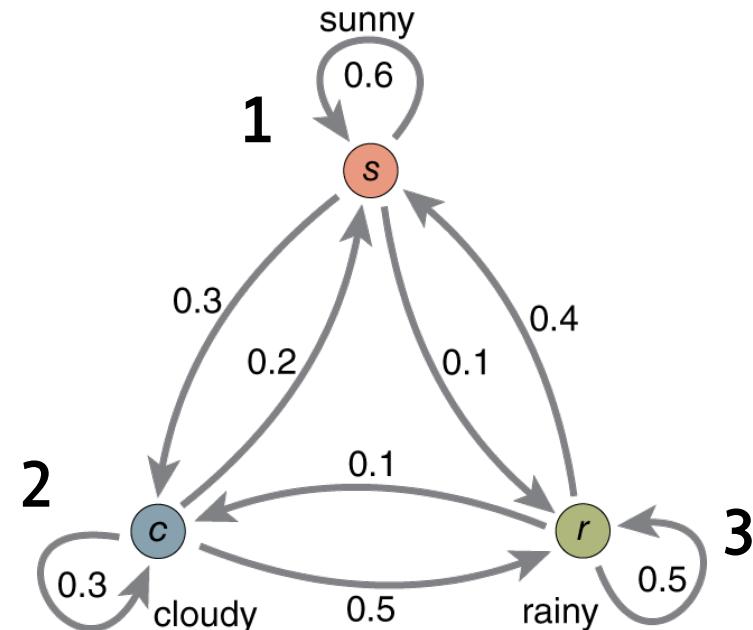
# Equilibrium probabilities

Question: if we play the game a very long time, how much time would we spend in each state?

- That is the *asymptotic* (equilibrium) distribution.
  - I.e., the probability of finding the system in a given state.
  - I.e., the free energy ( $\Delta G$ ) of that state!

1 1 1 2 2 3 3 1 1 1 2 2 2 2 1 1 1 1 1 1 1 3 1 1 1 1 1 2 3 1 1 3...  
(1000 times)

1 2 3  
431 220 349



$$\begin{aligned} p_\infty(1) &= 0.431 \\ p_\infty(2) &= 0.220 \\ p_\infty(3) &= 0.349 \end{aligned}$$

$$\sum_i p_\infty(i) = 1$$

$$G_1 = -k_B T \log(p_1) \simeq 0.50 \text{ kcal/mol}$$

$$G_2 = -k_B T \log(p_2) \simeq 0.91 \text{ kcal/mol}$$

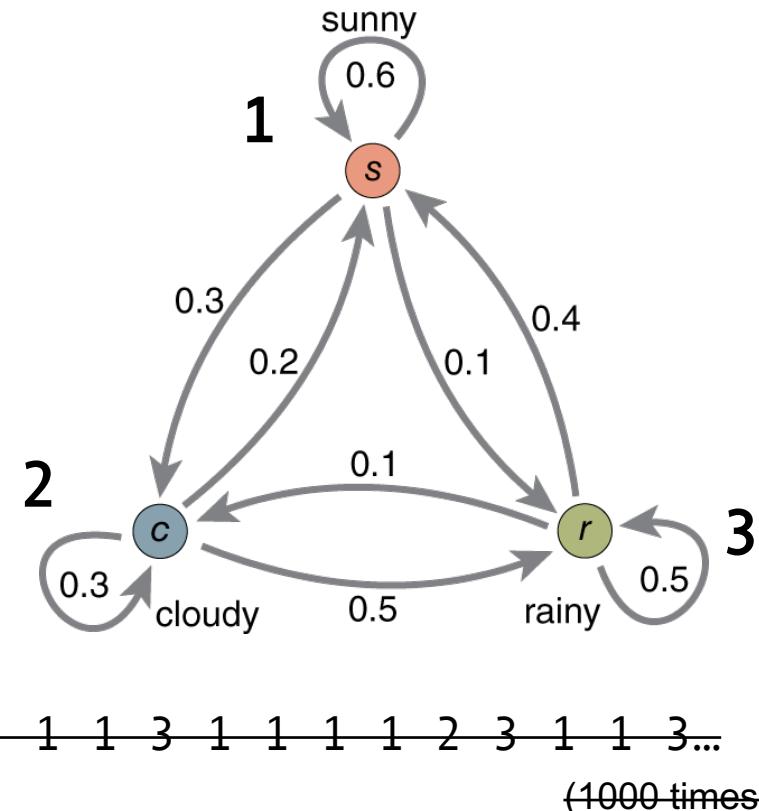
$$G_3 = -k_B T \log(p_3) \simeq 0.63 \text{ kcal/mol}$$

$$\begin{aligned}\Delta G_{11} &\doteq 0 \text{ kcal/mol} \\ \Delta G_{21} &\simeq 0.37 \text{ kcal/mol} \\ \Delta G_{31} &\simeq 0.18 \text{ kcal/mol}\end{aligned}$$

# Equilibrium probabilities

Question: if we play the game a very long time, how much time would we spend in each state?

- That is the *asymptotic* (equilibrium) distribution.
  - I.e., the probability of finding the system in a given state.
  - I.e., the free energy ( $\Delta G$ ) of that state!



# Sample and count



# Compute eigenvectors of matrix $P$

	[,1]	[,2]	[,3]
[1, ]	0.6	0.3	0.1
[2, ]	0.2	0.3	0.5
[3, ]	0.4	0.1	0.5

Compute  
eigenvectors:  
same result  
in one shot

# Intermezzo

## Google's PageRank

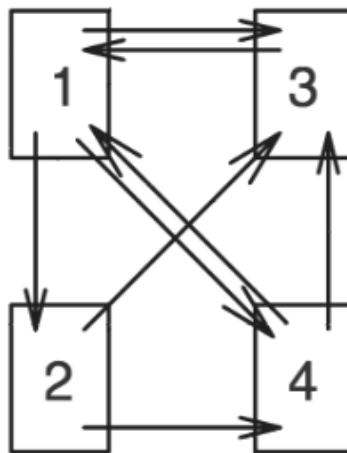
SIAM REVIEW  
Vol. 48, No. 3, pp. 569–581

© 2006 Society for Industrial and Applied Mathematics

# The \$25,000,000,000 Eigenvector: The Linear Algebra behind Google\*

Kurt Bryan<sup>†</sup>  
Tanya Leise<sup>‡</sup>

**Abstract.** Google's success derives in large part from its PageRank algorithm, which ranks the importance of web pages according to an eigenvector of a weighted link matrix. Analysis of the PageRank formula provides a wonderful applied topic for a linear algebra course. Instructors may assign this article as a project to more advanced students or spend one or two lectures presenting the material with assigned homework from the exercises. This material also complements the discussion of Markov chains in matrix algebra. Maple and *Mathematica* files supporting this material can be found at [www.math.hulman.edu/~bryan](http://www.math.hulman.edu/~bryan).

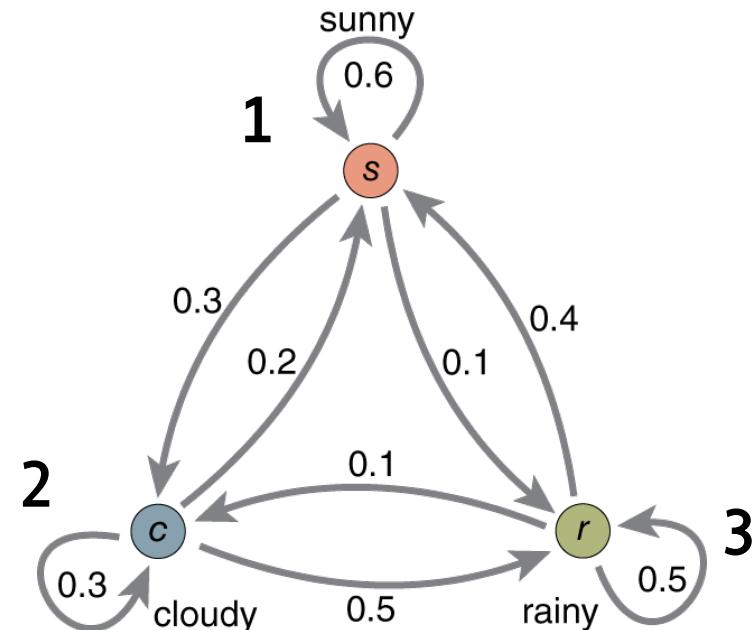


**Fig. 1** An example of a web with only four pages. An arrow from page A to page B indicates a link from page A to page B.

# Kinetics

Question: if we start from 1, how many steps on average do I wait to reach 2?

- This is the  $1 \rightarrow 2$  mean first passage time
  - I.e., the (inverse) transition rate ( $1/k_{\text{on}}$ ) between those states!



The diagram shows the sequence starting with 15 terms: 1, 1, 1, 2, 2, 3, 3, 1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 3, 1, 1, 1, 1, 2, 3, 1, 1, 3... Blue brackets above the sequence group terms together, with red numbers indicating the size of each group: a bracket of length 3 is labeled '3', a bracket of length 2 is labeled '2', and a bracket of length 11 is labeled '11'. Below the sequence, a bracket of length 2 is labeled '2' and a bracket of length 1 is labeled '1'. The text "And so on..." is at the bottom right.

**Answer:** average all the numbers in red.

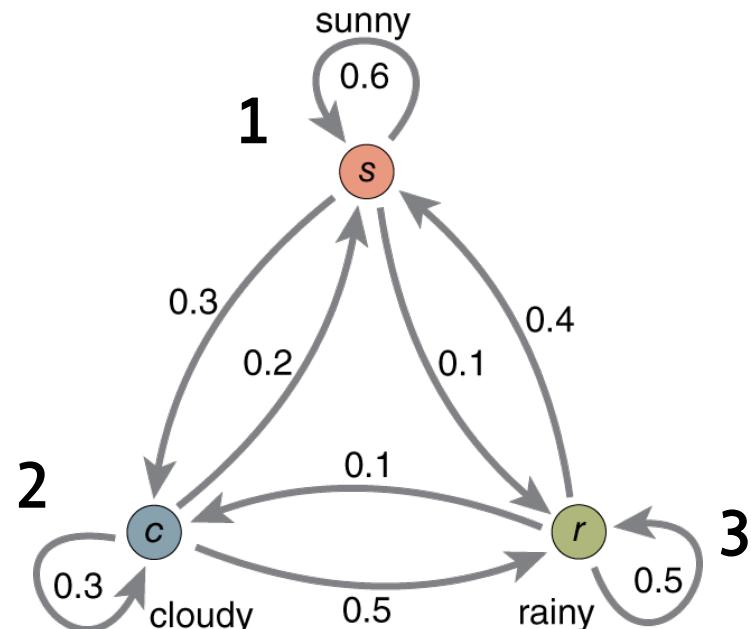
(Actually done mathematically on  $P_{ij}$ )

- Obviously one can ask the same question about any  $i \rightarrow j$
  - Remember that time is steps  $\times \tau$

# Committor

Question: if we start from 1, what is the probability to pass through 2 w.r.t. 3 first? (regardless of the path)

- This is the  $I \rightarrow \{2,3\}$  committer probability
  - States which are equally probable to fall back to the “reactants” or advance to “product” state form the *transition state*.



The diagram shows the first 15 digits of the Cantor sequence: 1, 1, 1, 2, 2, 3, 3, 1, 1, 1, 2, 2, 2, 2, 1, 1, 1, 1, 1, 3. Blue brackets above the sequence group digits into sets of 2^n. The first bracket groups the first two digits (1, 1) as a set of 2^0 = 2. The second bracket groups the next four digits (1, 1, 2, 2) as a set of 2^1 = 2. The third bracket groups the next eight digits (1, 1, 1, 2, 2, 2, 2, 2) as a set of 2^2 = 4. The fourth bracket groups the next sixteen digits (1, 1, 1, 1, 1, 1, 3) as a set of 2^3 = 8. A green bracket above the sequence groups the last three digits (1, 1, 1) as a set of 2^3 = 8. An arrow points from the text "And so on..." to the fourth bracket.

Answer: compute the fraction of end-in-2 vs end-in-3.

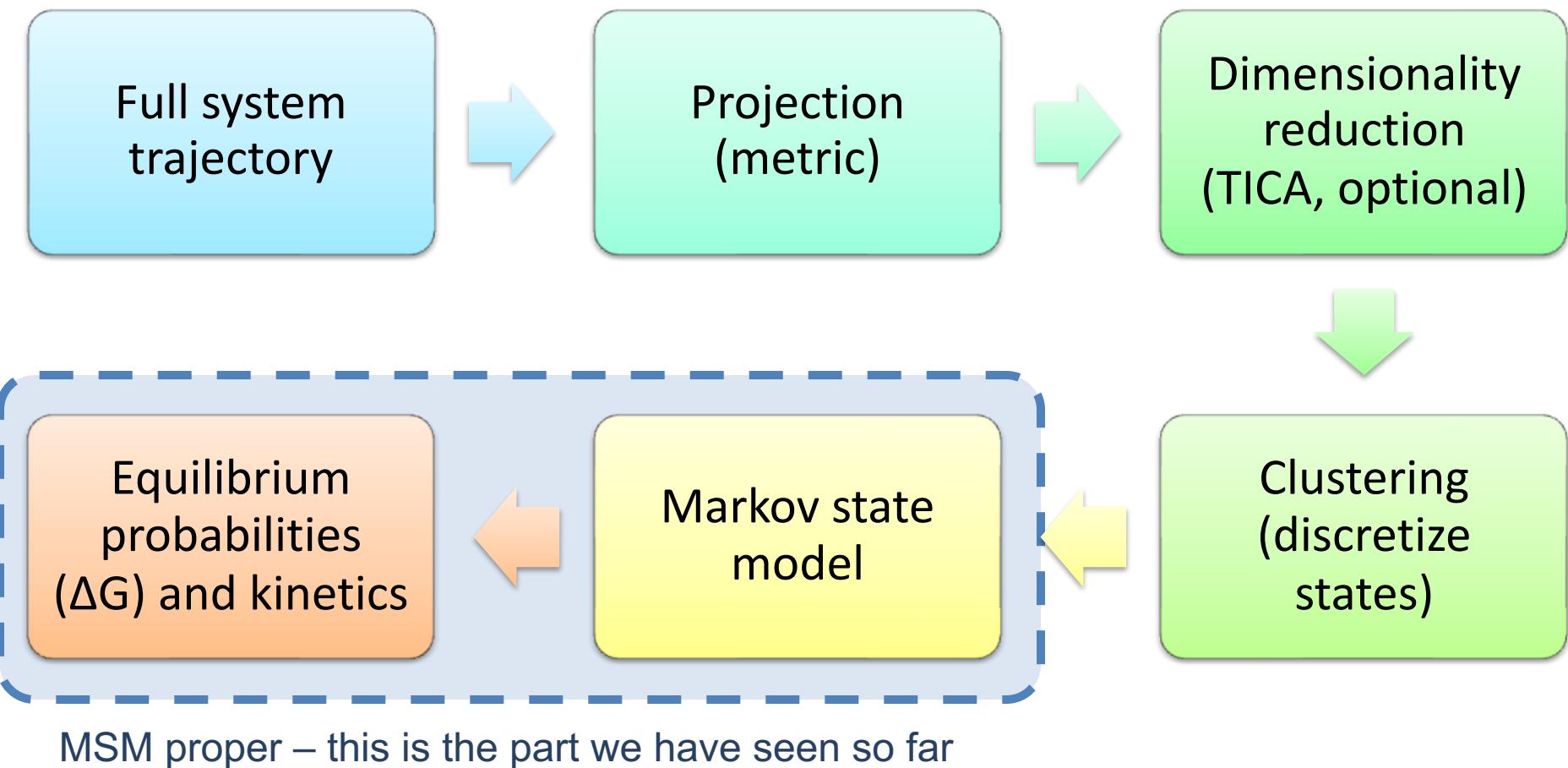
(Actually done mathematically on  $P_{ij}$ )

# Markovianity

- The state transition probabilities only depend on the current state. Examples:
  - Today's weather, not yesterday's
  - Where the ligand is, not how did it got there
- The property may be false at short timescales but true at longer ones: study varying  $\tau$
- It does depend on the chosen states: study varying state discretization

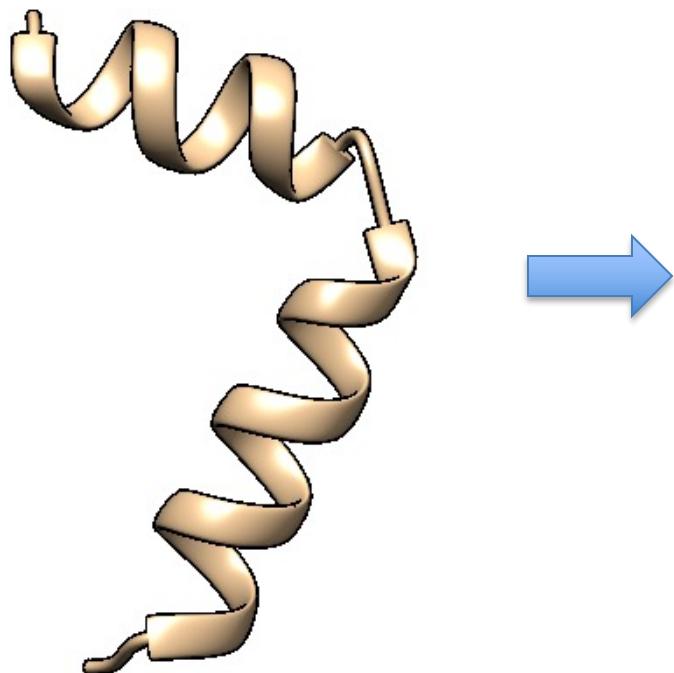
**Back to molecules:**  
**>> 1 dimension**

# MSM-based analysis overview



# Metric projection

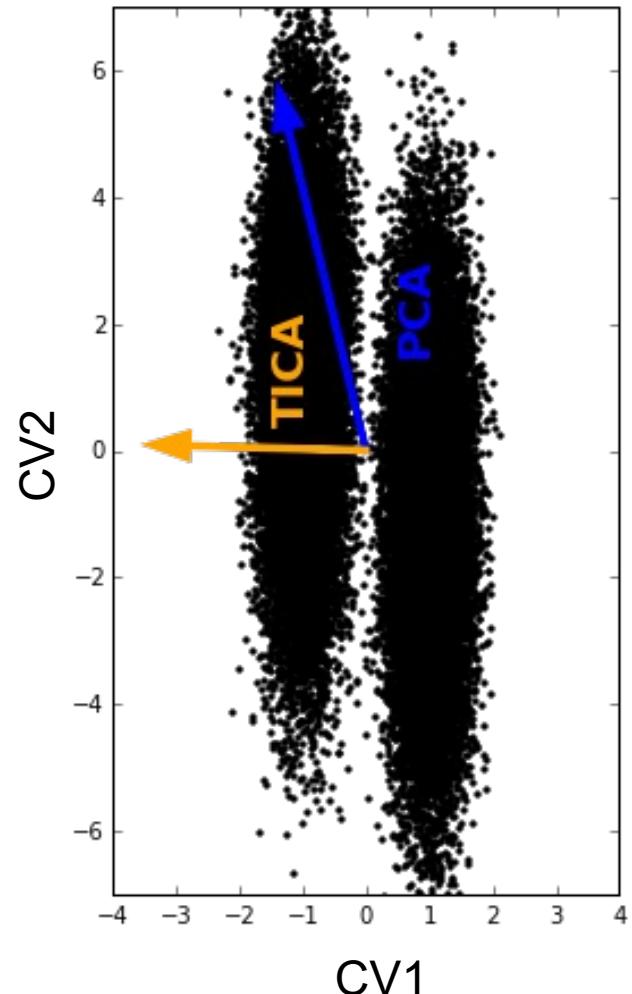
The first step is to project the system state in a lower-dimensional space (“metric”). Many choices are available, e.g.



- Manually chosen distances
- Atom coordinates
- N phi/psi Ramachandran angles
- Distance matrix
- Contact matrix
- ...

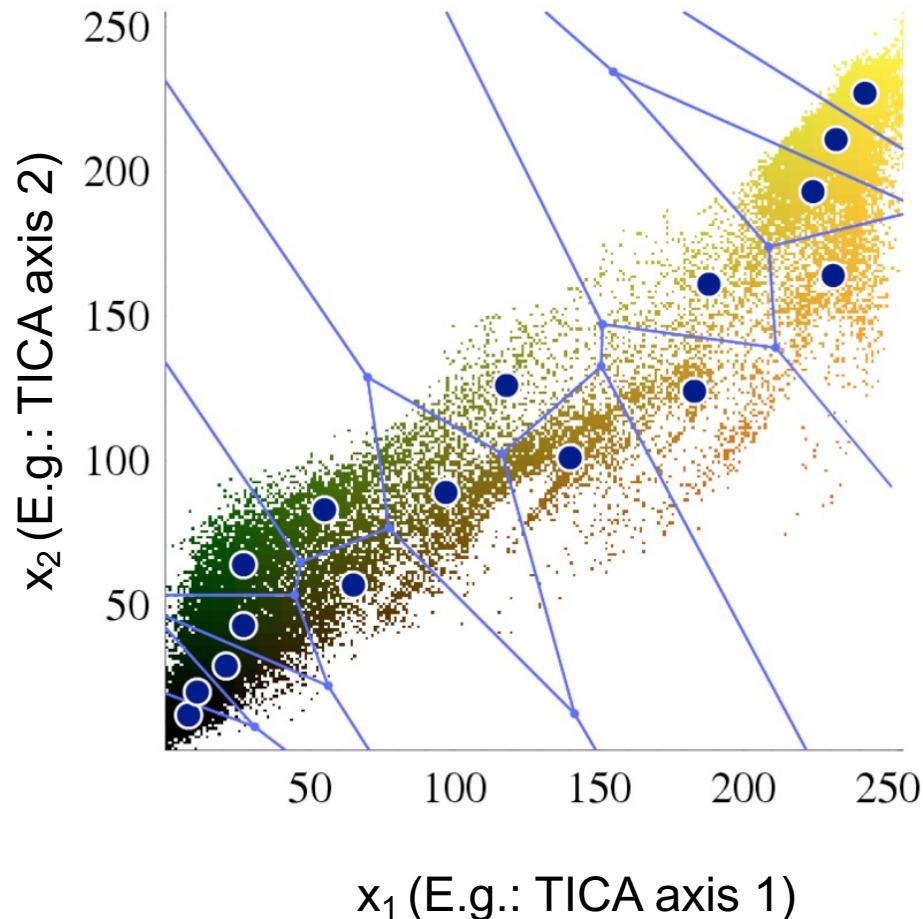
# Time-lagged independent component analysis

- The feature space dimensionality may still be too high
- Do a further low-dimensional projection on the “slow” degrees of freedom: TICA
- It is based on lagged autocorrelation
- Contrast with PCA, which fits the “most elongated” ellipsoid, **ignoring time**



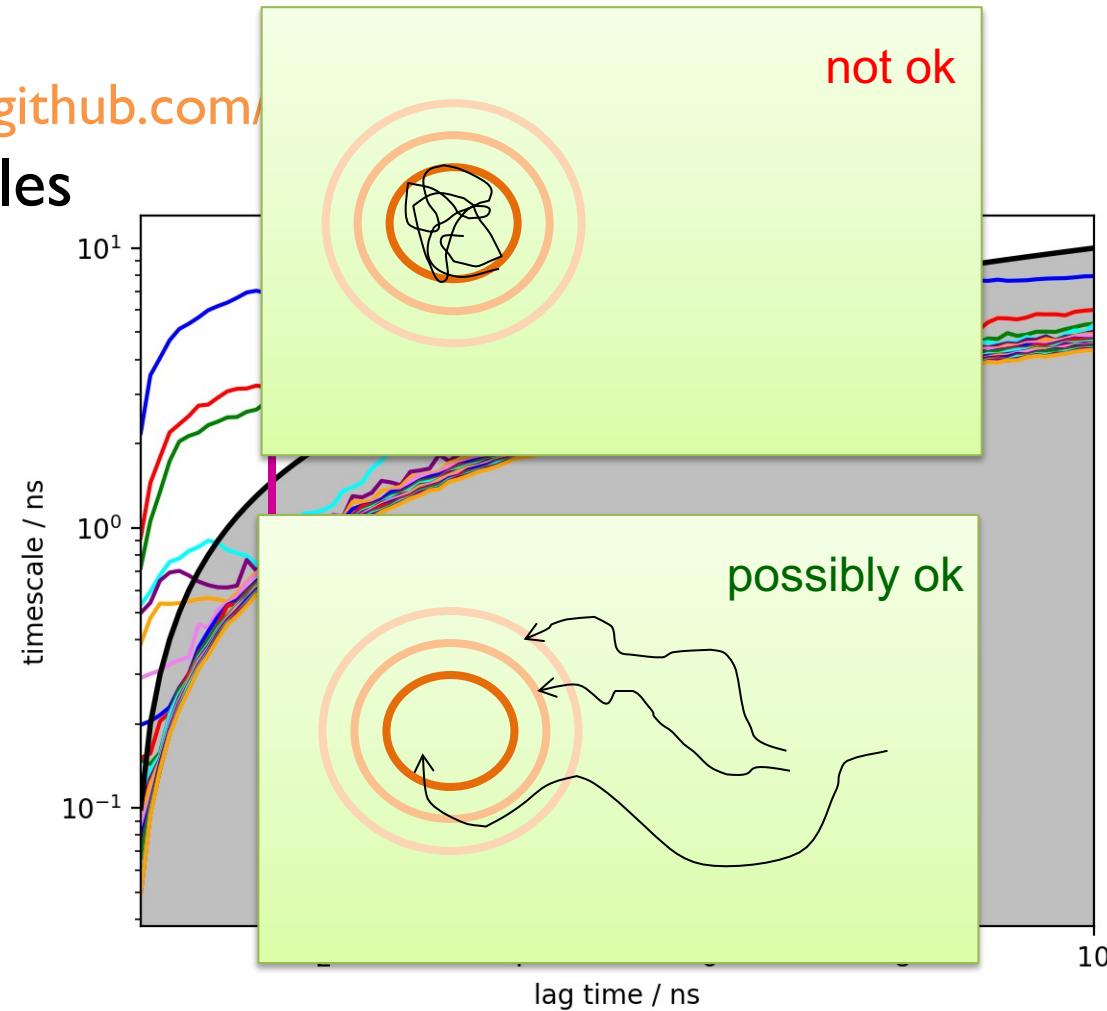
# Clustering (1<sup>st</sup> level)

- Now move to a discrete state space
- Reduction from the low-dimensional space is done by *clustering*
  - Usually called “microstates”
- Several algorithms are implemented in MSM packages (e.g. grid; k-means; etc. We won’t discuss them)



# Even when merging multiple trajectories, making efficient use of sampling does not dispense us from sampling the phase space

- Ace-Ala<sub>3</sub>-Nme
- Part of examples at: [github.com/omnisimulations/ace-alanine](https://github.com/omnisimulations/ace-alanine)
- 6 Ramachandran angles
- The system is trapped
- How to «shoot» trajectories:
  - «bathtub» not ok
  - «shower» maybe
  - adaptive spawning
  - or your favourite string-like method



# Examples from the literature

# Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations

Ignasi Buch, Toni Giorgino, and Gianni De Fabritiis<sup>1</sup>

Computational Biochemistry and Biophysics Laboratory, Universitat Pompeu Fabra, Barcelona Biomedical Research Park, C/Doctor Aiguader 88, 08003 Barcelona, Spain

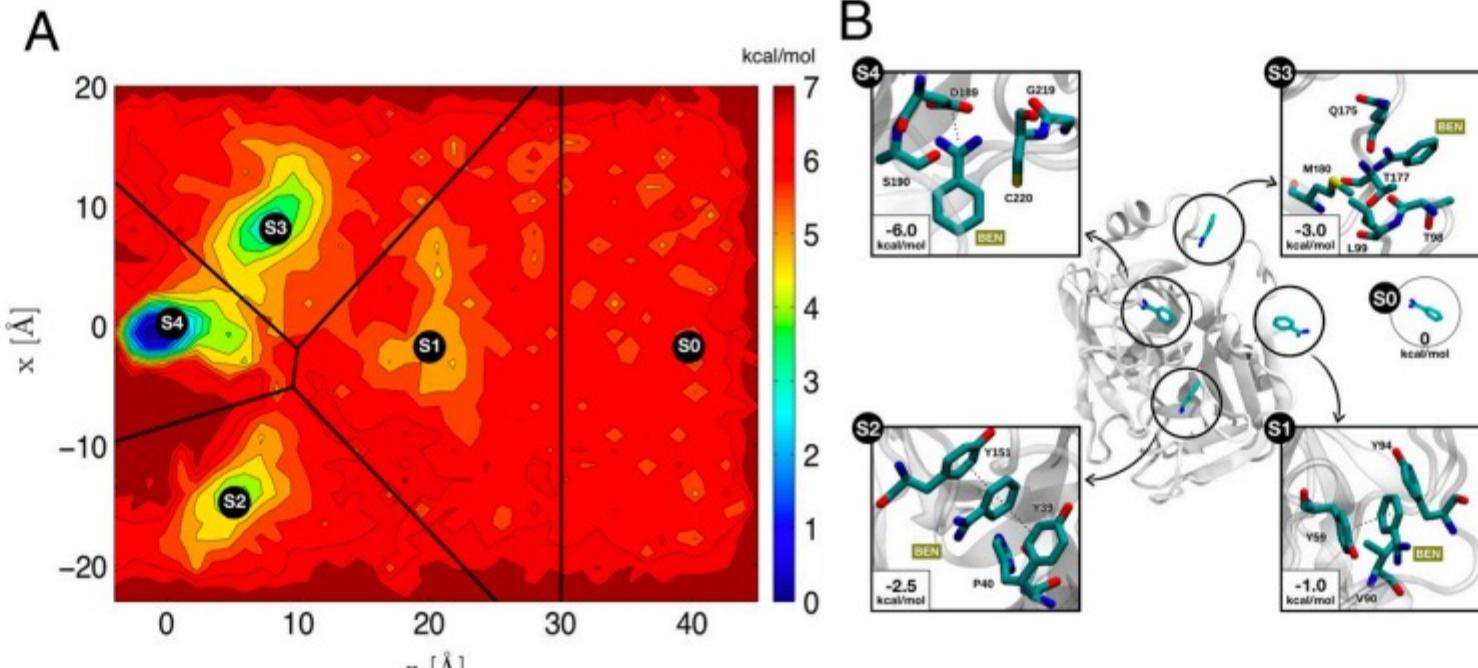
Edited by Arieh Warshel, University of Southern California, Los Angeles, CA, and approved May 11, 2011 (received for review March 4, 2011)

The understanding of protein–ligand binding is of critical importance for biomedical research, yet the process itself has been very difficult to study because of its intrinsically dynamic character. Here, we have been able to quantitatively reconstruct the complete binding process of the enzyme-inhibitor complex trypsin-benzamidine by performing 495 molecular dynamics simulations of the

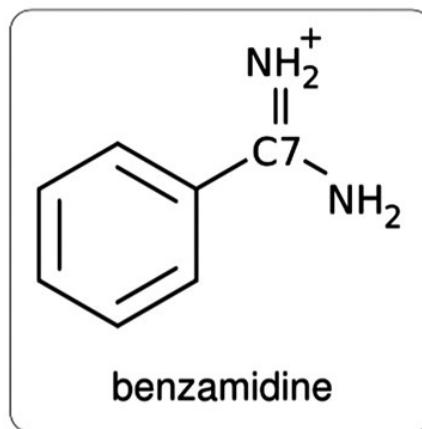
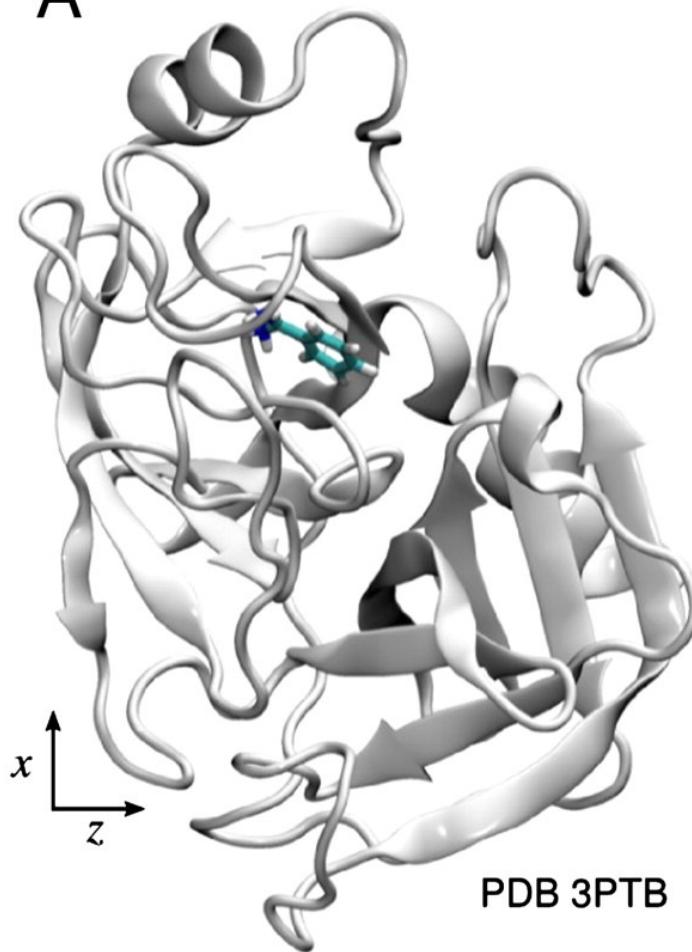
reproduce with atomic resolution the crystallographic mode of binding, but we also provide the kinetically and energetically meaningful transition states of the process.

Free ligand binding has been used in the past to describe computational experiments in which, typically, a ligand is placed at a certain distance from the target protein and first by diffusion and

in the pro-

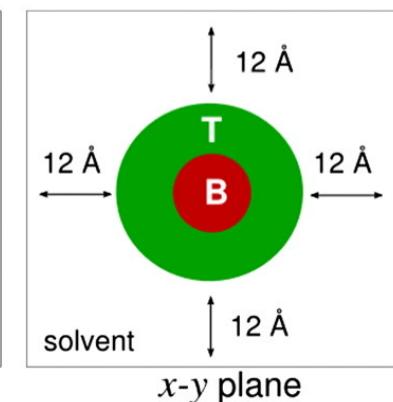
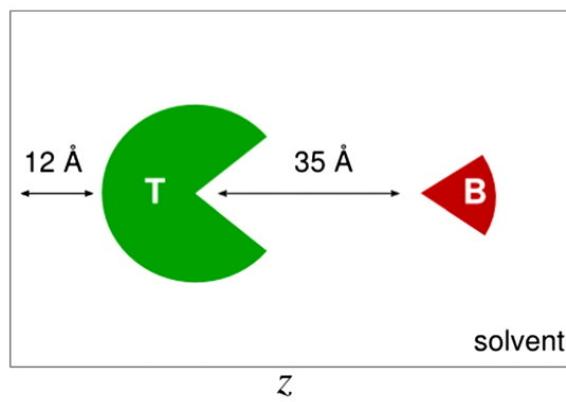


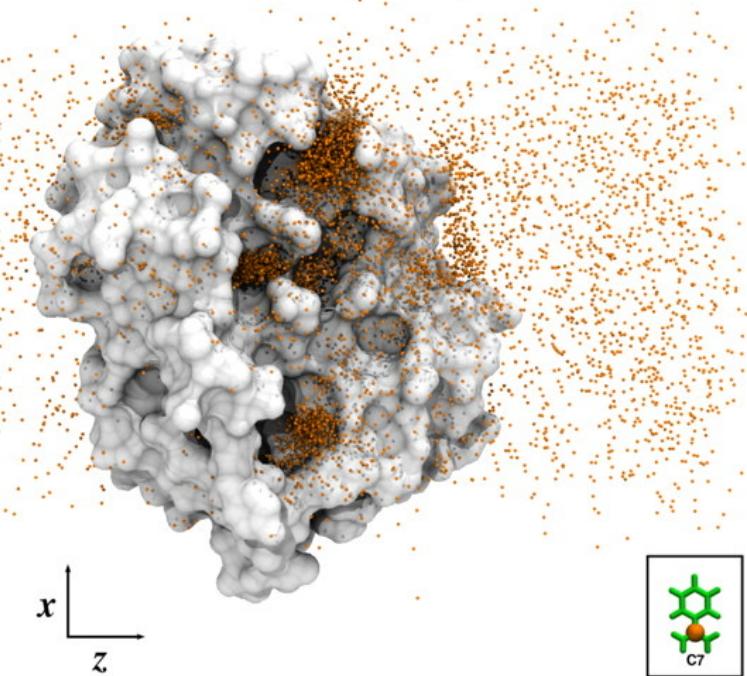
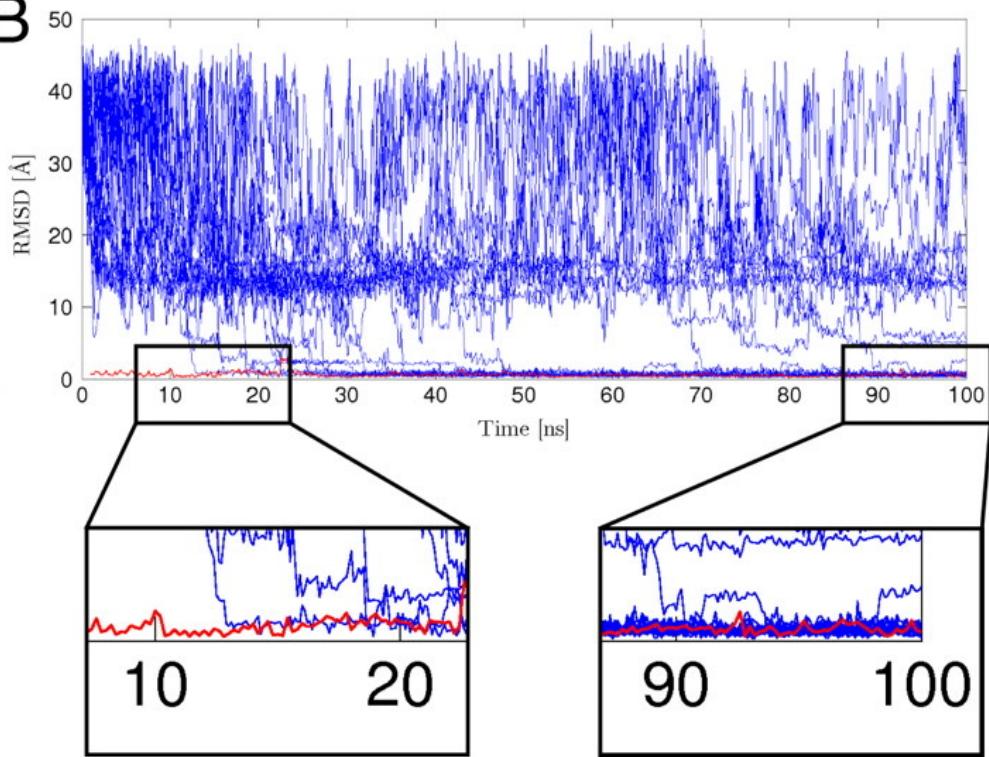
A



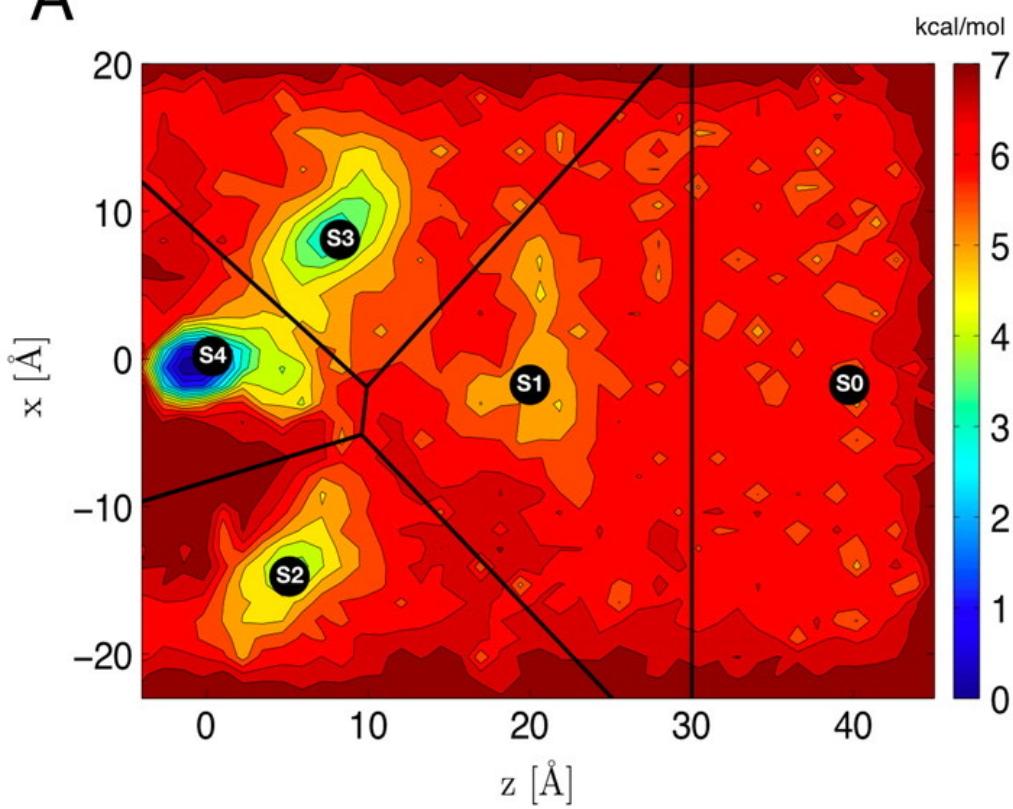
- Rigid ligand: no internal degrees of freedom
- Small ligand: no orientation d.o.f.
- States defined as small cubes according just to the position in 3D space

B

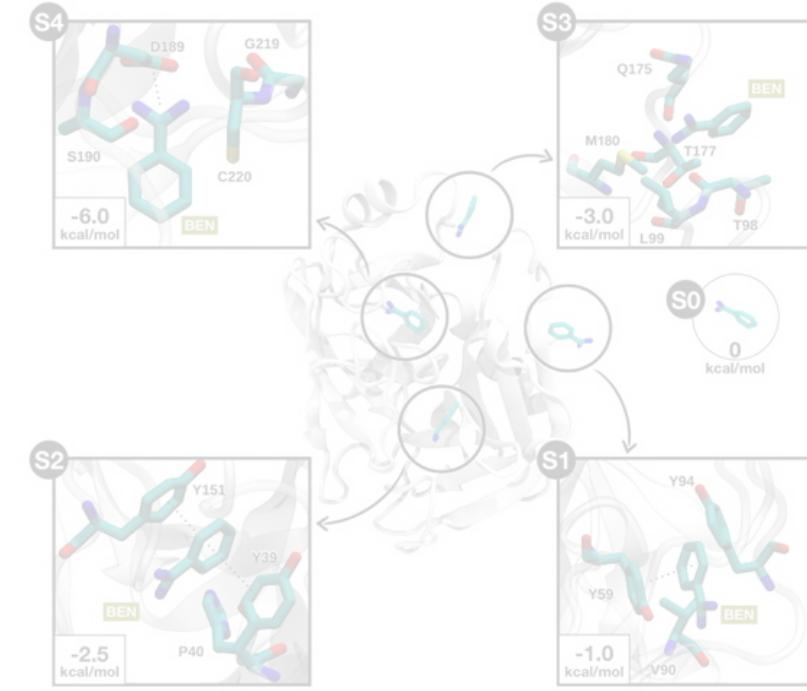


**A****B**

A



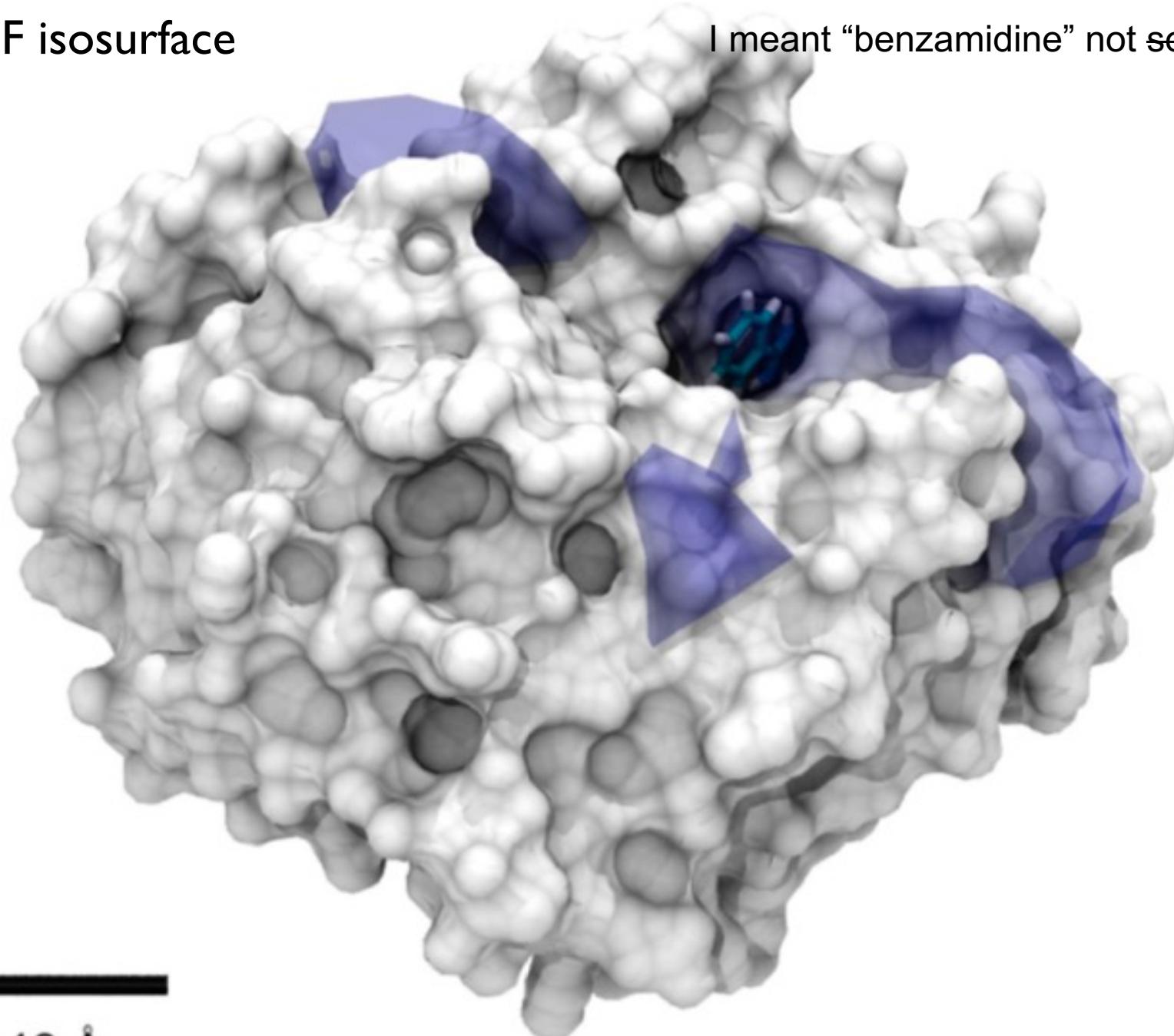
B



Identification of metastable states. (A) PMF in the xz plane. Five different metastable states can be identified from the different free-energy minima (S0 to S4). The relative free energy between the unbound state S0 and the bound state S4 is -6 kcal/mol. The most probable transition to the bound state S4 may be from S3 from the fact that the barrier between the two states is just 1.5 kcal/mol. (B) Structural characterization of metastable states. In states S1 and S2, benzamidine is stabilized by  $\pi$ - $\pi$  stacking interactions with Y151 and Y39 side chains. In S3, a hydrogen bond may be formed between NH<sub>2</sub> groups of benzamidine (only heavy atoms shown for clarity) and Q175 side chain, or by a cation- $\pi$  interaction between the Q175 side chain again, and benzamidine's benzene ring.

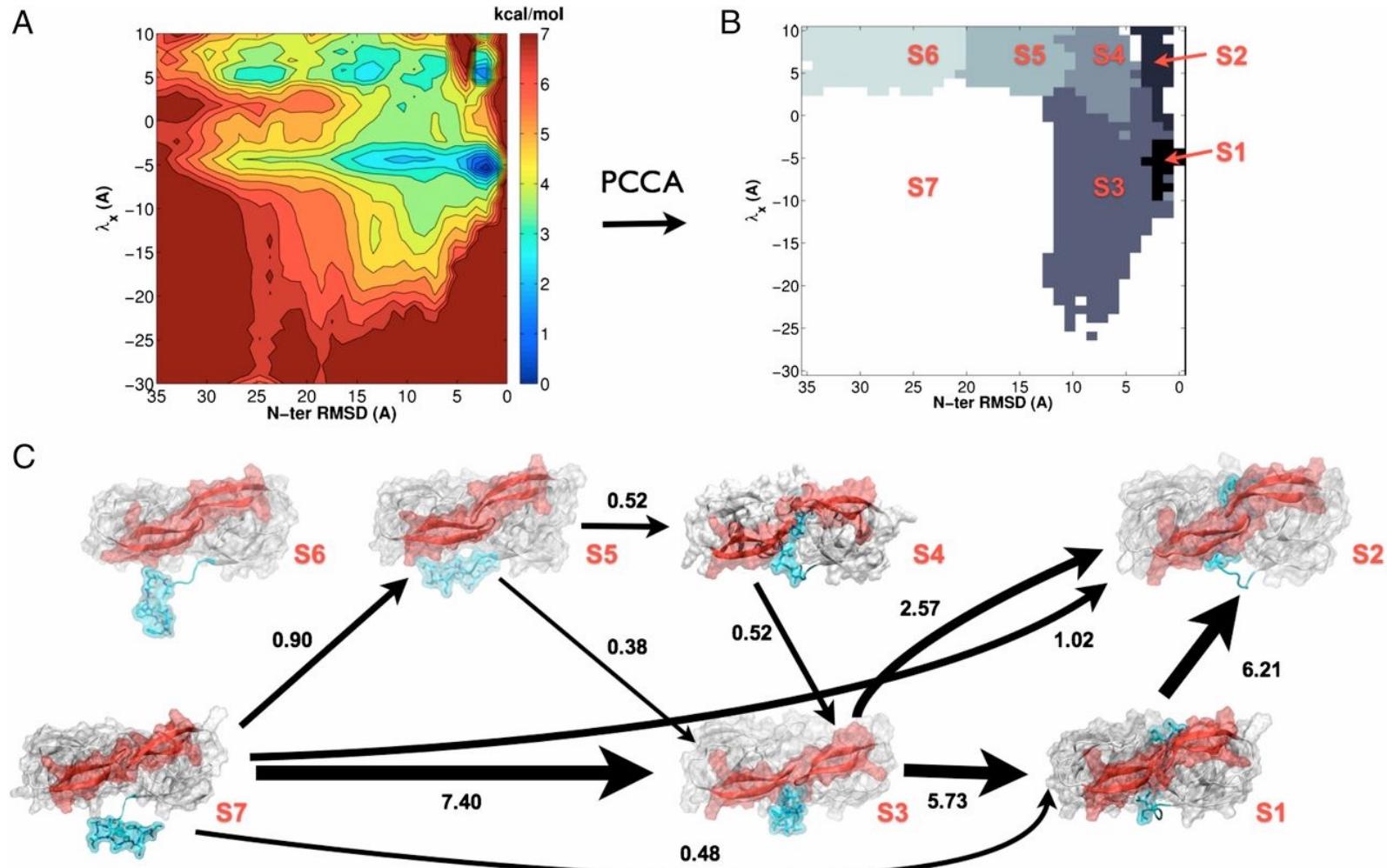
3D PMF isosurface

I meant “benzamidine” not solvent



10 Å

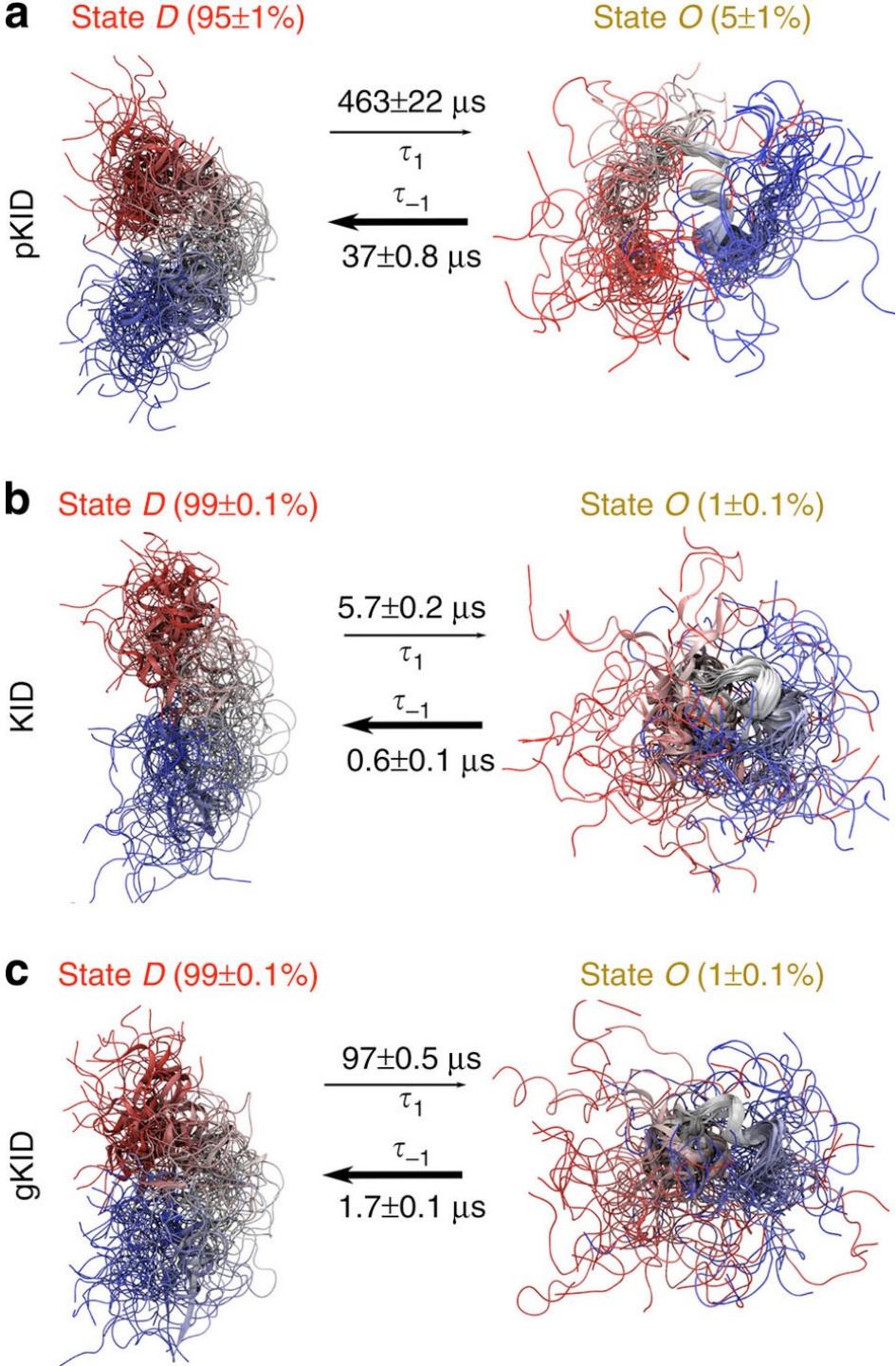
# Kinetic characterization of the critical step in HIV-1 protease maturation



# Kinetic modulation of a disordered protein domain by phosphorylation

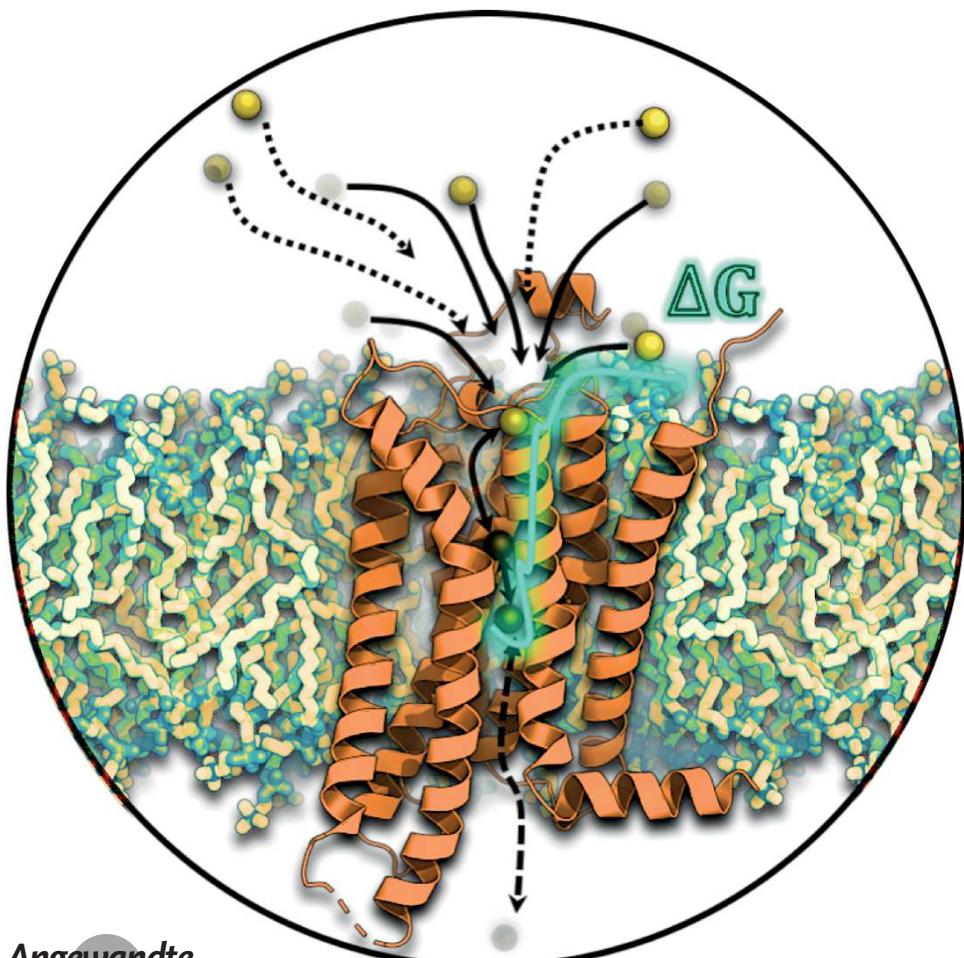
N. Stanley, S. Esteban and G. De Fabritiis, Nat. Commun. 5, 5272 (2014)

doi:10.1038/ncomms6272



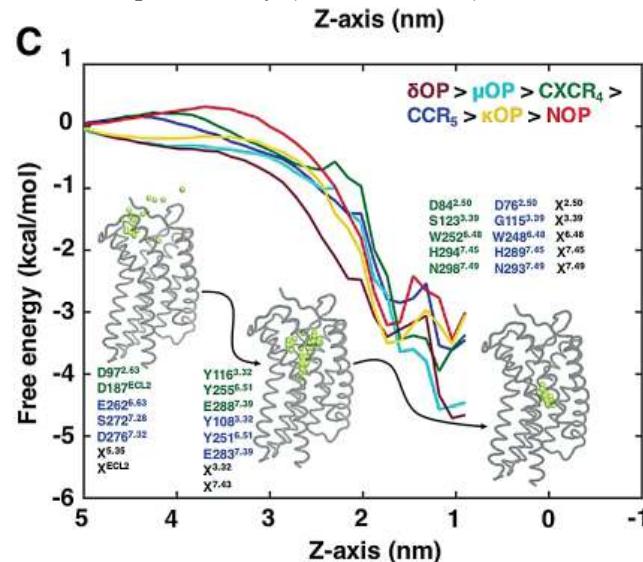
# Universality of the Sodium Ion Binding Mechanism in Class A G-Protein-Coupled Receptors

Balaji Selvam, Zahra Shamsi, and Diwakar Shukla\*



plays a key role in identifying drug candidates.<sup>[7]</sup> Several studies have shown that GPCRs are allosterically modulated by endogenous Na<sup>+</sup> ions.<sup>[8–11]</sup> Selent and co-workers explored the binding mechanism of Na<sup>+</sup> to the D<sub>2</sub> receptor by molecular dynamics simulations.<sup>[12]</sup> Na<sup>+</sup> binds in the middle of the transmembrane (TM) region to a conserved Asp<sup>2,50</sup> residue (Ballesteros–Weinstein numbering),<sup>[13]</sup> as elucidated by high-resolution crystal structures of the GPCRs.<sup>[14–16]</sup> More recent MD studies on μ-, κ-, and δ-opioid (OP) receptors also support Na<sup>+</sup> ion binding to Asp<sup>2,50</sup>.<sup>[17,18]</sup> Na<sup>+</sup> binding and a change in the protonation states of titratable residues through the water network have a significant effect on the stabilization of the conformational states of GPCRs.<sup>[19–21]</sup>

elusive. Herein, we performed hundreds-of-microsecond long simulations of 18 GPCRs to elucidate their Na<sup>+</sup> binding mechanism (see the Supporting Information, Tables S1 and S2). We constructed Markov state models (MSMs) to estimate the free energy profiles for Na<sup>+</sup> binding to each GPCR. Analysis of the Na<sup>+</sup> binding kinetics revealed key residues that act as major barriers for Na<sup>+</sup> entry to the intracellular site. We also predicted the average mean first passage time (MFPT) for Na<sup>+</sup> binding and unbinding events by transition path theory (TPT; Table S3).



# **Conclusions and Resources**

# Warning

- Still very active field
- Suggested further steps: worked out real-world examples distributed with software packages\*
- For serious work, many more details are in...
  - the theory of Markov state models
  - the discretization, projection, and estimation of models from trajectories

\* see the last slides

# Conclusions

- MSM methods may be an attractive formalism for medium-sized problems
- Make efficient use of *unbiased* sampling
- Still require *huge* (but achievable) amounts of sampling/simulation for biologically interesting systems
- Strong mathematical foundation, with good software available

# Software + Tutorials

- All are Python-based
  - They include clear walkthroughs (highly recommended) with datasets
- DeepTime-ML – [deeptime-ml.github.io](https://deeptime-ml.github.io) ←
- HTMD – [www.htmd.org](https://www.htmd.org)
  - Also analysis + system build + adaptive ...
  - Can aggregate large-scale datasets
- MSMBuilder - [msmbuilder.org](https://msmbuilder.org)

# Literature

- Buch et al.,  
10.1073/pnas.1103547108
  - Rigid ligand + protein association, simplest case
- Swinney, PMID:19152211
  - Importance of kinetics in drug design
- Chodera and Noe,  
10.1016/j.sbi.2014.04.002
  - Excellent overview (1)
- Voelz et al., 10.1021/ja9090353
  - Full reconstruction of a millisecond folding
- Selvam et al., 10.1002/anie.201708889
  - Reconstruction of Na<sup>+</sup> binding to GPCRs
- Pande et al.,  
10.1016/j.ymeth.2010.06.002
  - Excellent overview (2)
- Paul et al., 10.1038/s41467-017-01163-6
  - Peptide-peptide association
- Doerr et al., 10.1021/ct400919u
  - Adaptive sampling
- Bryan et al., “The \$25 B eigenvector”
  - The original PageRank algorithm; e.g.  
[jdc.math.uwo.ca/M1600b/l/pagerank-1600.pdf](http://jdc.math.uwo.ca/M1600b/l/pagerank-1600.pdf)

**End**