MD Simulations – day 2
Analysis

Toni Giorgino
National Research Council of Italy
toni.giorgino@cnr.it
www.giorginolab.it

GitHub
@giorginolab

for Prof. Fuxreiter's course @ University of Padova
May 2024

https://github.com/giorginolab/MD-Tutorial-Data

Part I

Giorgino T, Mattioni D, Hassan A, Milani M, Mastrangelo E, Barbiroli A, et al. Nanobody interaction unveils structure, dynamics and proteotoxicity of the Finnish-type amyloidogenic gelsolin variant. Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease. 2019 Mar 1;1865(3):648–60.

Journal link. Preprint.

BBA - Molecular Basis of Disease 1865 (2019) 648–660

Contents lists available at ScienceDirect

BBA - Molecular Basis of Disease

journal homepage: www.elsevier.com/locate/bbadis

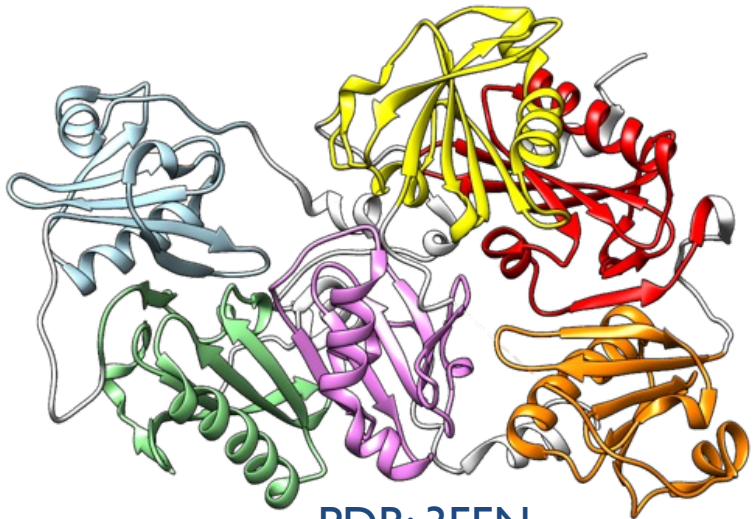
Nanobody interaction unveils structure, dynamics and proteotoxicity of the Finnish-type amyloidogenic gelsolin variant

Toni Giorgino[a,b], Davide Mattioni[a,c,1], Amal Hassan[b,1], Mario Milani[a,b], Eloise Mastrangelo[a,b], Alberto Barbiroli[d], Adriaan Verhelle[e], Jan Gettemans[f], Maria Monica Barzago[c], Luisa Diomede[c], Matteo de Rosa[a,b,*]

[a] *Istituto di Biofisica, Consiglio Nazionale delle Ricerche, Milano, Italy*
[b] *Dipartimento di Bioscienze, Università degli Studi di Milano, Milano, Italy*
[c] *Department of Molecular Biochemistry and Pharmacology, Istituto di Ricerche Farmacologiche Mario Negri IRCCS, 20156 Milan, Italy*
[d] *Dipartimento di Scienze per gli Alimenti, la Nutrizione e l'Ambiente, Università degli Studi di Milano, Milano, Italy*
[e] *Department of Molecular Medicine, Department of Molecular and Cellular Neuroscience, Dorris Neuroscience Center, The Scripps Research Institute, La Jolla, CA 92037, USA*
[f] *Nanobody Lab, Department of Biochemistry, Faculty of Medicine and Health Sciences, Ghent University, Ghent, Belgium*

Inactive

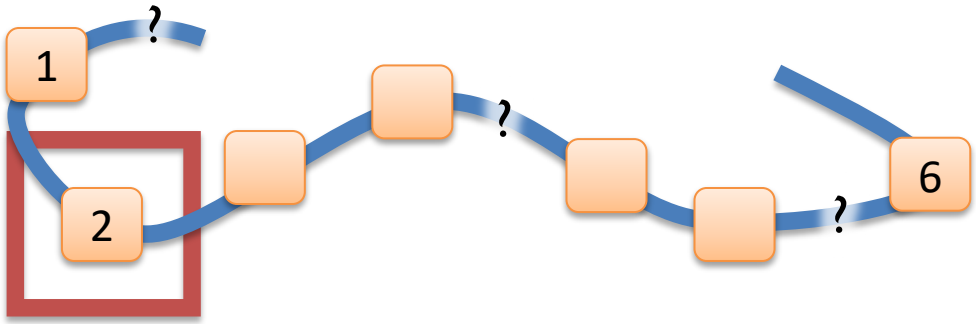
No Ca^{2+}.
Closed form.
Resolved (2008).



~ 100 Å

Active

~ mM Ca^{2+}.
Active form.
Dynamic. Elusive.



However, crystallisation of Ca^{2+}-bound **isolated domains** has been successful.

AGel amyloidosis

Also known as…

FAMILIAL Am., FINNISH TYPE
Am., MERETOJA TYPE
Am. DUE TO MUTANT GELSOLIN
& permutations

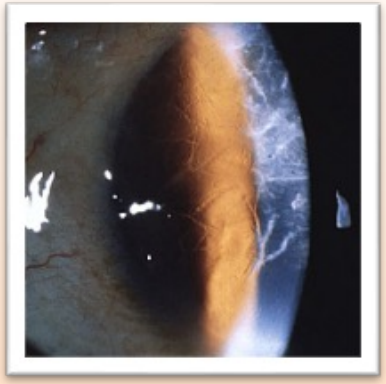
Am. V
Am. CRANIAL NEUROPATHY WITH LATTICE CORNEAL DYSTROPHY
HEREDITARY GELSOLIN Am.

"**AGel amyloidosis** is a rare, usually systemic amyloidosis characterized by a triad of ophthalmologic, neurologic and dermatologic findings due to the deposition of **gelsolin amyloid fibrils** in these tissues".

- Rare, but endemic in Finland

  "*… was 1/1,040 among 182,000 inhabitants, whereas in 1860 in the parish Valkeala the prevalence was calculated at 1/155. Today the true prevalence probably lies between these figures.*" (Meretoja 1973)

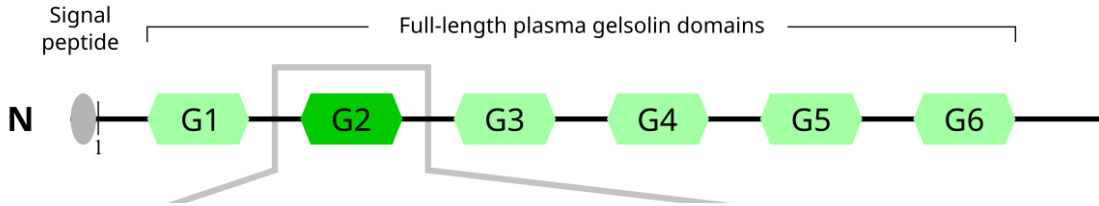
- Inherited, autosomal dominant
- Caused by *GSN* gene mutations
- No cure, just symptomatic treatments



Corneal lattice dystrophy

Cutis laxa

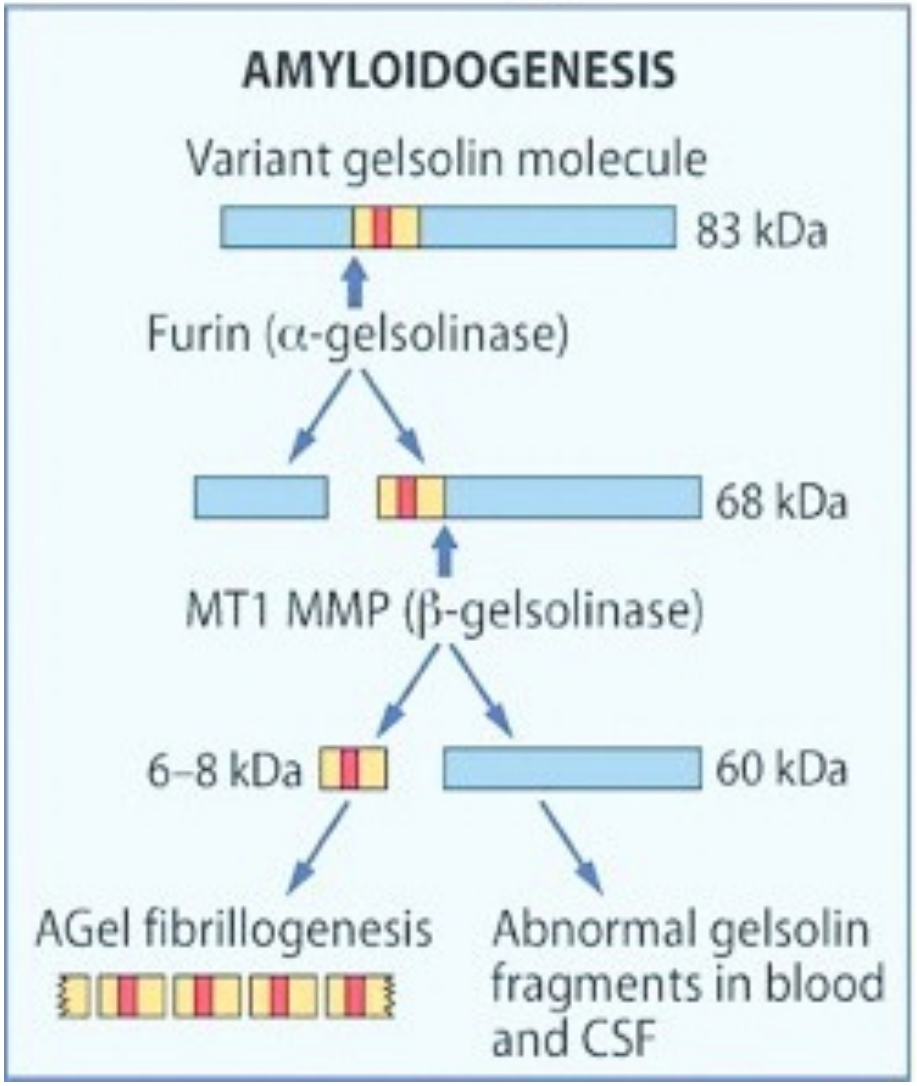
Mutations causing AGel

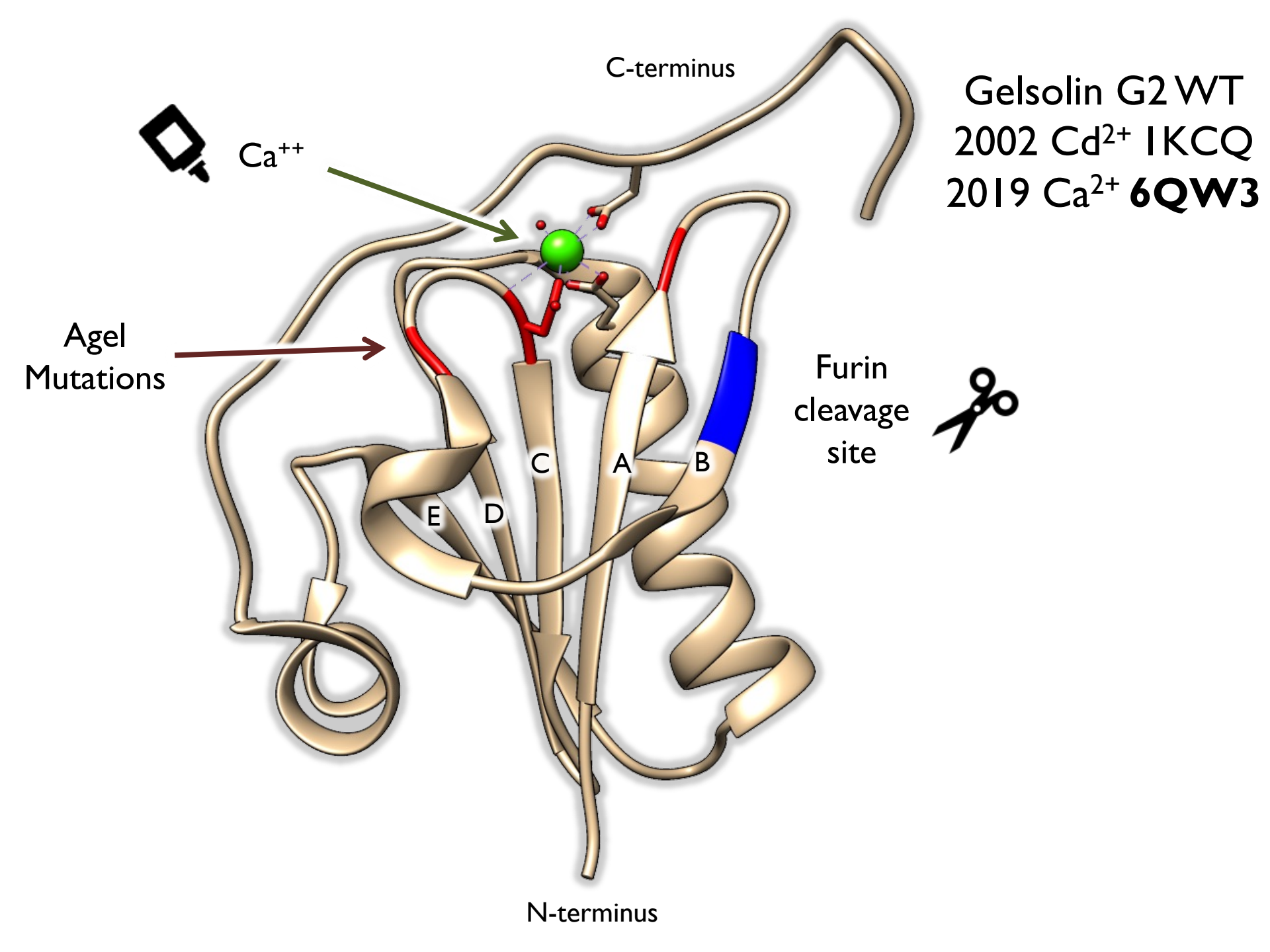


- D187N — Systemic, Finnish
- D187Y — Systemic, Danish
- G167R — Renal
- N184K — Renal

All in G2 domain

- WT plasma GSN encounters furin without consequence since WT is correctly folded.
- Mutant plasma GSN has a structural intermediate state exposing the cleavage site, and undergoes pathological furin processing.





Gelsolin G2 WT
2002 Cd^{2+} 1KCQ
2019 Ca^{2+} **6QW3**

Mutant	Structure	Year	Disease form
WT	1KCQ	2002	Sporadic
N184K	5FAF	2015	Renal (rare)
G167R	5O2Z	2017	Renal (rare)
D187N	???		Systemic, Finnish
D187Y	???		Systemic, Danish

Observation: D187X mutants **lose** Ca^{2+} coordination

Hypothesis: Is this indication of an **order-disorder** shift?

Support for the hypothesis:

- Inability to crystallize
- GSN NMR data *
- Known Ca^{2+}-regulated disorder/order transitions: sortase, a-cyc toxin, etc.
- Thermodynamic stabilities§ : WT ≠ mutant

* Kazmirski et al. Nat Struct Mol Biol. 2002 Feb;9(2):112

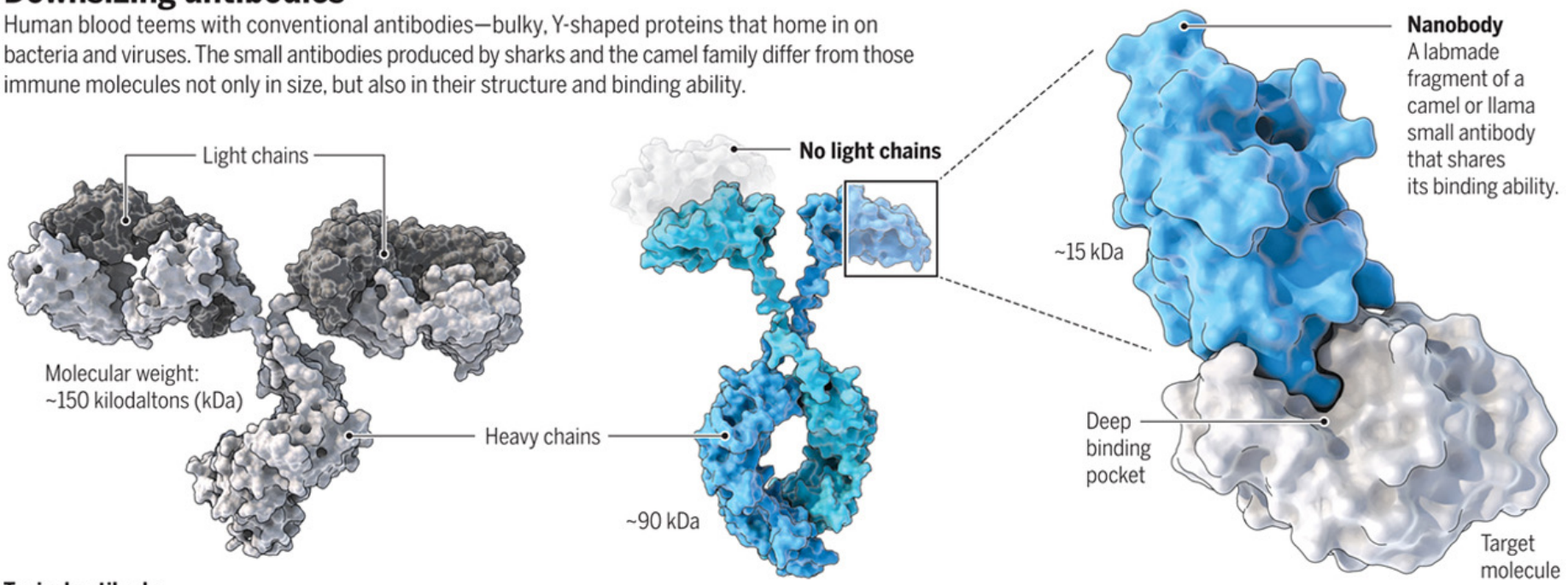
§ Thermal and chemical

Nanobodies

VHH, variable domain of hcAbs ~15 kDa

Downsizing antibodies

Human blood teems with conventional antibodies—bulky, Y-shaped proteins that home in on bacteria and viruses. The small antibodies produced by sharks and the camel family differ from those immune molecules not only in size, but also in their structure and binding ability.



Typical antibody
Two light and two heavy chains intertwine to make a protein that can identify and affix to bits of pathogens or other molecules.

Small antibody
This slimmed-down variety lacks light chains but can still bind to its targets.

Nanobody binding
Because of its binding style, a nanobody can fit into crevices on molecules.

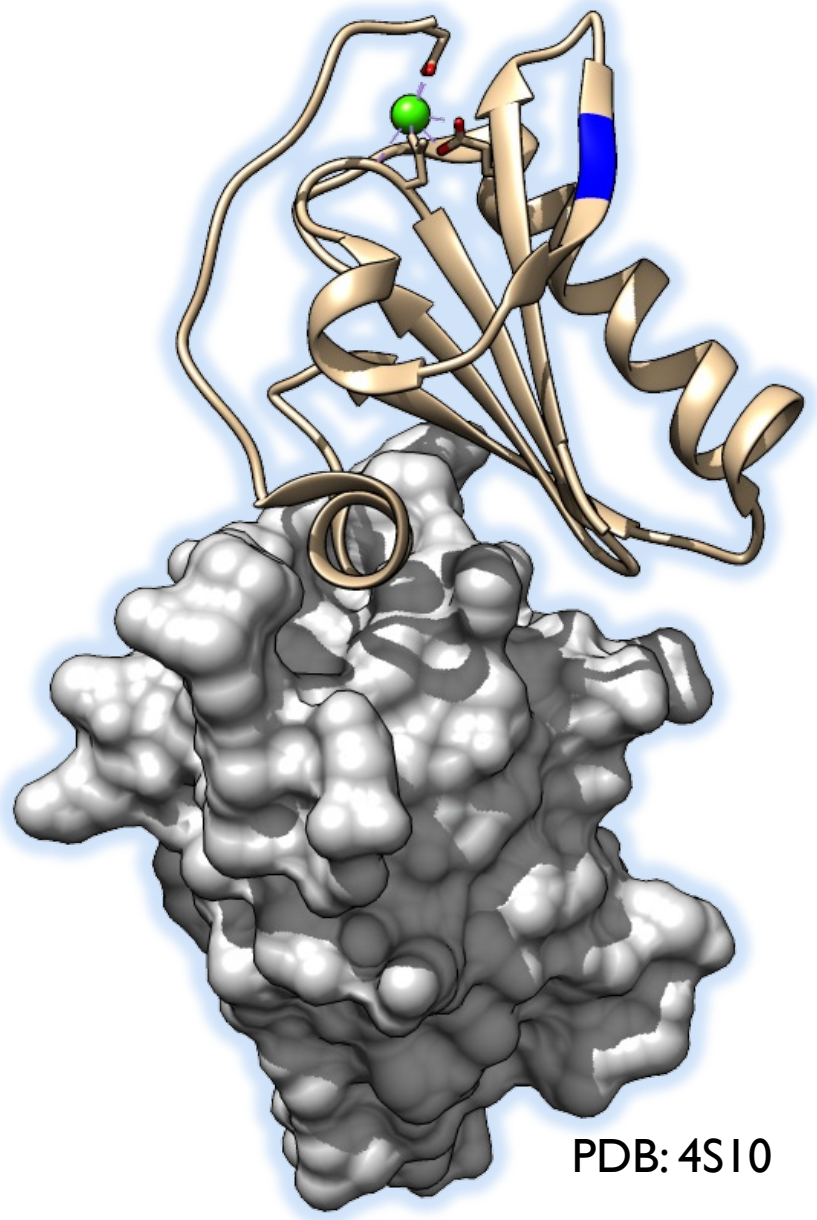
doi:10.1126/science.360.6389.594

2015: Gettemans et al. – G2 Nanobodies

- Inoculate llamas with GSN **WT** G2
- Extract nanobodies (Nb11)
- Sequence them
- Obtain WT:Nb structure

Idea

- Can the WT-raised Nb11 re-stabilize D187N enough to allow crystallization?



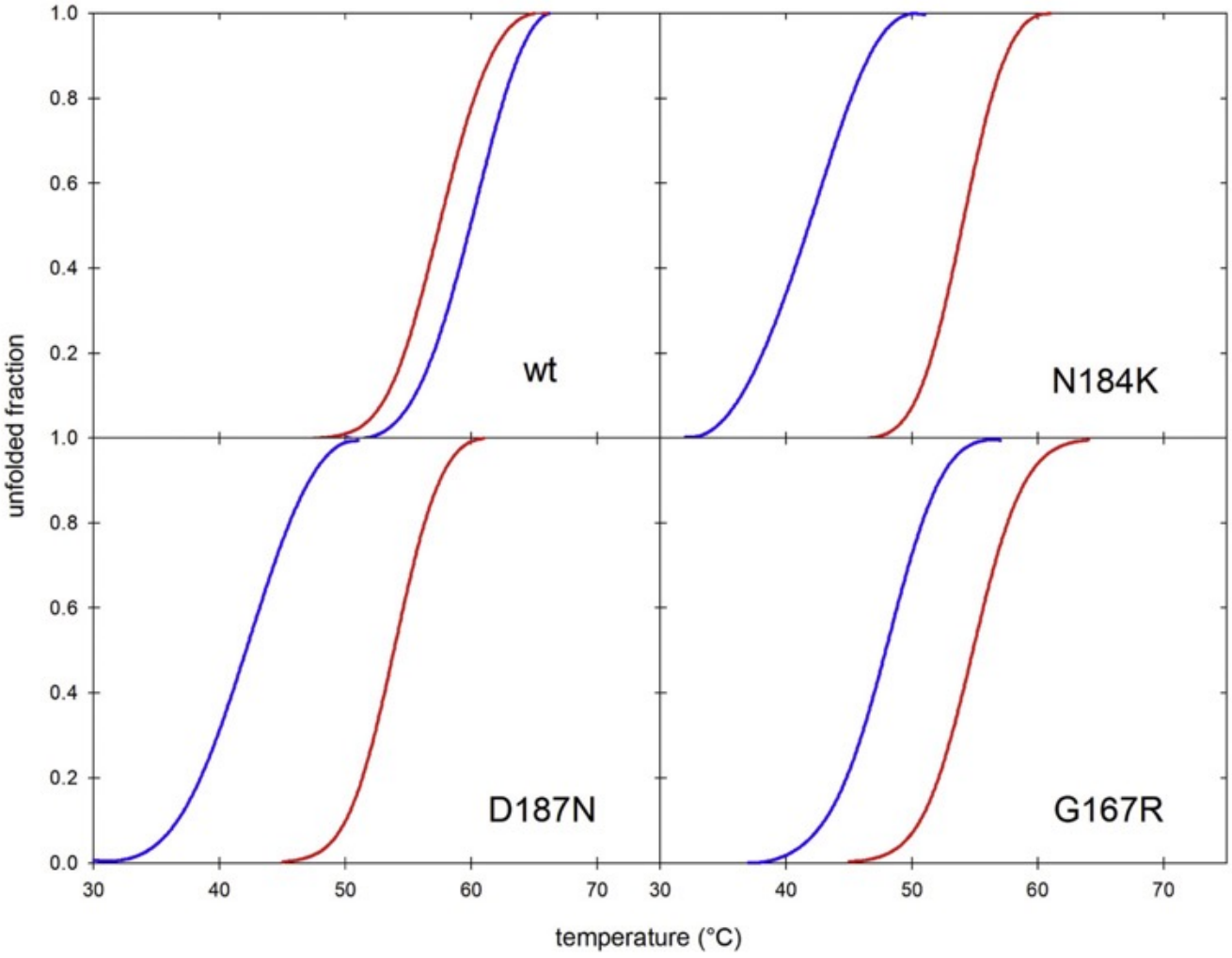
PDB: 4S10

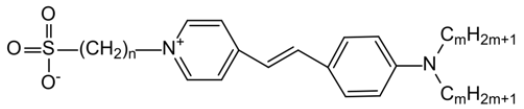
Van Overbeke et al., HMG 2015

Does Nb11 increase stability?

→ Thermofluor (differential scanning fluorimetry) experiments: measure the temperature exposing the hydrophobic core.

Mutants are destabilized
Nb11-binding re-stabilizes them





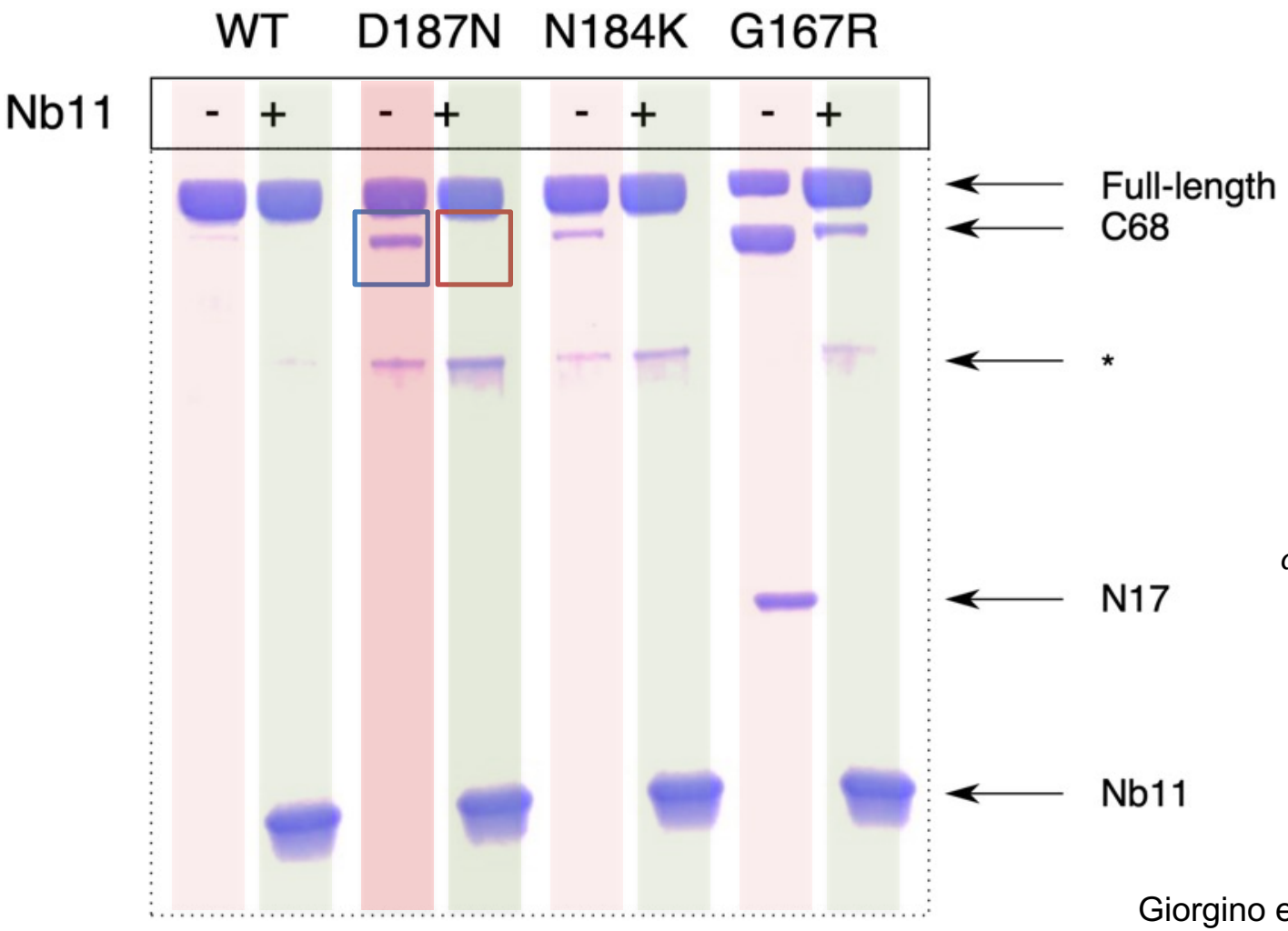
SYPRO Orange binds nonspecifically to hydrophobic surfaces, and water strongly quenches its fluorescence. When the protein unfolds, the exposed hydrophobic surfaces bind the dye, resulting in an increase in fluorescence by excluding water.

Giorgino et al., BBA MBD, 2019

Does Nb11 protect from proteolysis?

→ Furin proteolysis experiments

Mutants are cleaved
Nb11-binding protects them



C68 and N17 (kDa) are the two fragments produced by furin activity.

Giorgino et al., BBA MBD, 2019

6H1F: Gelsolin G2+nanobody

Structure Summary | 3D View | Annotations | Experiment | Sequence | Genome | Versions

Biological Assembly 1

3D View: Structure | 1D-3D View | Electron Density | Validation Report | Ligand Interaction

Global Symmetry: Asymmetric - C1
Global Stoichiometry: Hetero 2-mer - A1B1

Find Similar Assemblies

Biological assembly 1 assigned by authors and generated by PISA (software)

Biological Assembly Evidence: gel filtration

Macromolecule Content

- Total Structure Weight: 28.49 kDa
- Atom Count: 1,896
- Modelled Residue Count: 229
- Deposited Residue Count: 259
- Unique protein chains: 2

Display Files | Download Files

6H1F

Structure of the nanobody-stabilized gelsolin D187N variant (second domain)

PDB DOI: 10.2210/pdb6H1F/pdb

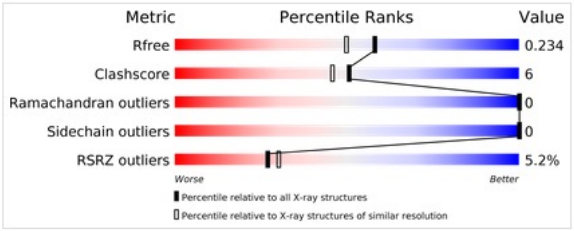
Classification: STRUCTURAL PROTEIN
Organism(s): Lama glama, Homo sapiens
Expression System: Escherichia coli
Mutation(s): Yes

Deposited: 2018-07-11 **Released:** 2019-01-23
Deposition Author(s): Hassan, A., Milani, M., Mastrangelo, E., de Rosa, M.
Funding Organization(s): Amyloidosis Foundation

Experimental Data Snapshot

Method: X-RAY DIFFRACTION
Resolution: 1.90 Å
R-Value Free: 0.233
R-Value Work: 0.199
R-Value Observed: 0.202

wwPDB Validation | 3D Report | Full Report



This is version 1.0 of the entry. See complete history.

Literature | Download Primary Citation

Nanobody interaction unveils structure, dynamics and proteotoxicity of the Finnish-type amyloidogenic gelsolin variant.

Giorgino, T., Mattioni, D., Hassan, A., Milani, M., Mastrangelo, E., Barbiroli, A., Verhelle, A., Gettemans, J., Barzago, M.M., Diomede, L., de Rosa, M.
(2019) Biochim Biophys Acta Mol Basis Dis **1865**: 648-660

PubMed: 30625383 Search on PubMed
DOI: 10.1016/j.bbadis.2019.01.010
Primary Citation of Related Structures:

3D Protein Feature View: 6H1F

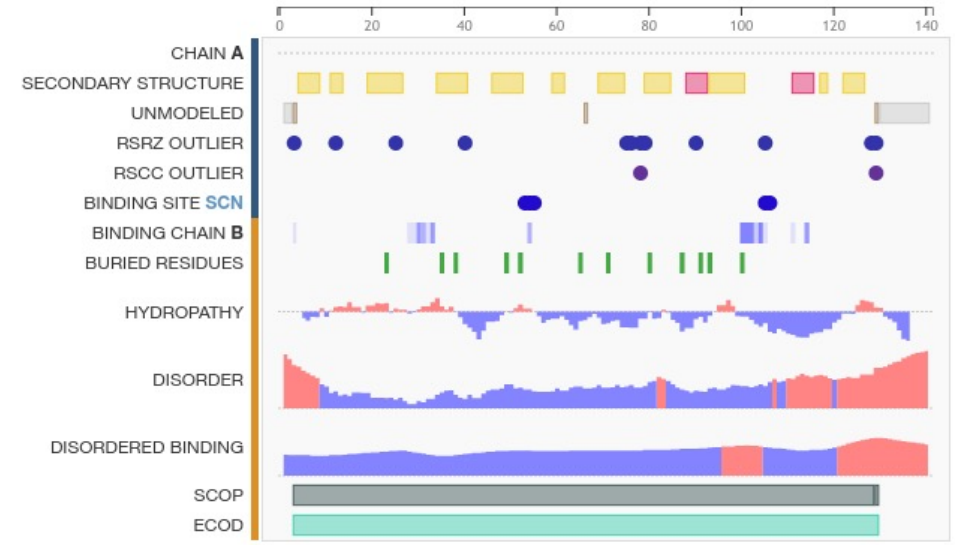
Help Back

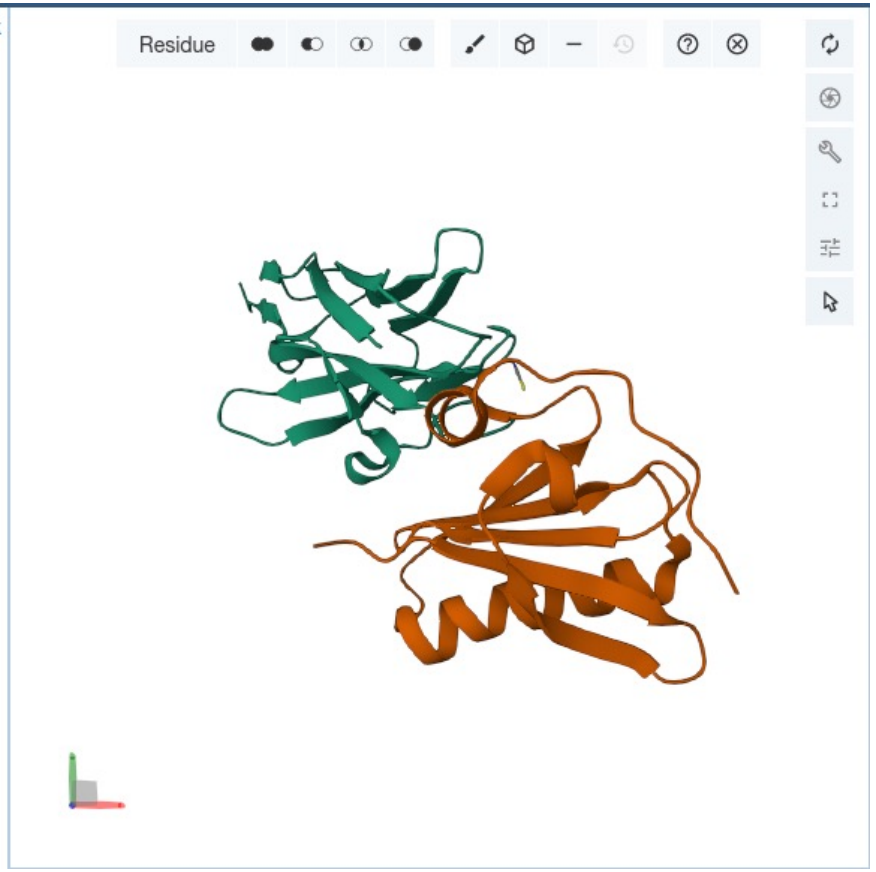
Structure of the nanobody-stabilized gelsolin D187N variant (second domain)

Chain

A

Gelsolin nanobody - Lama glama



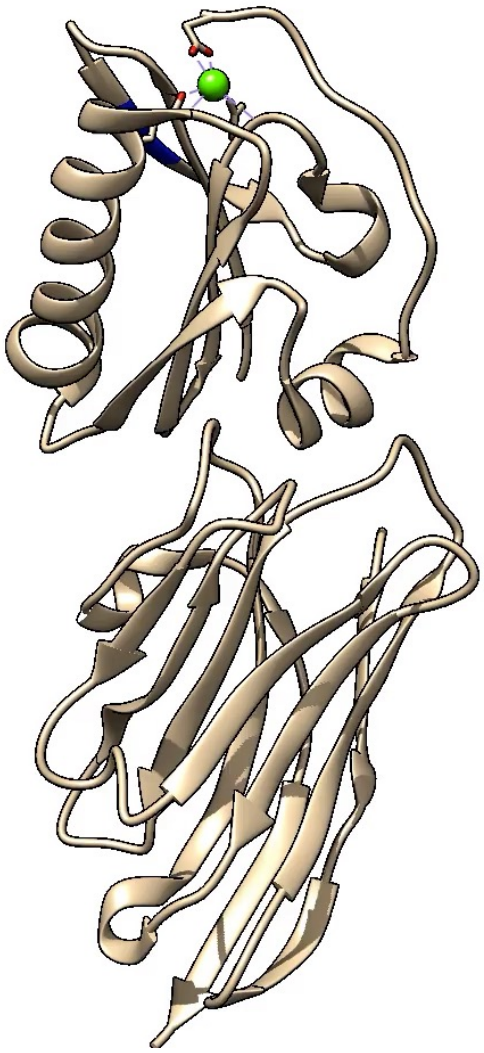


Three puzzles!

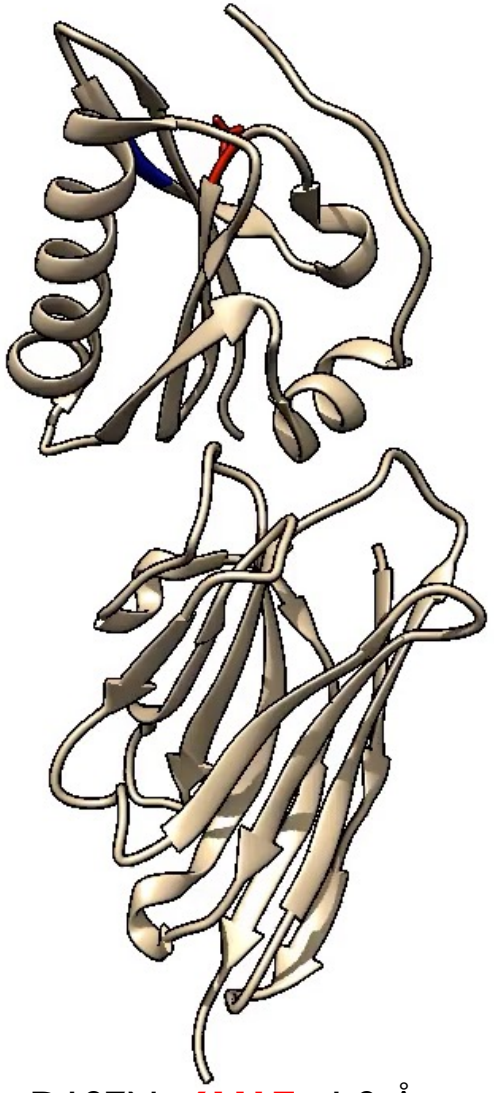
1. WT and D187N are **virtually identical***: same structure, different function
2. Nb11 binds far from the furin cleavage site…
3. …and far from the Ca^{2+} ion

* Except Ca^{2+} binding

WT:Nb11 complex compared to D187N:Nb11.



WT: 4S10, 2.6 Å



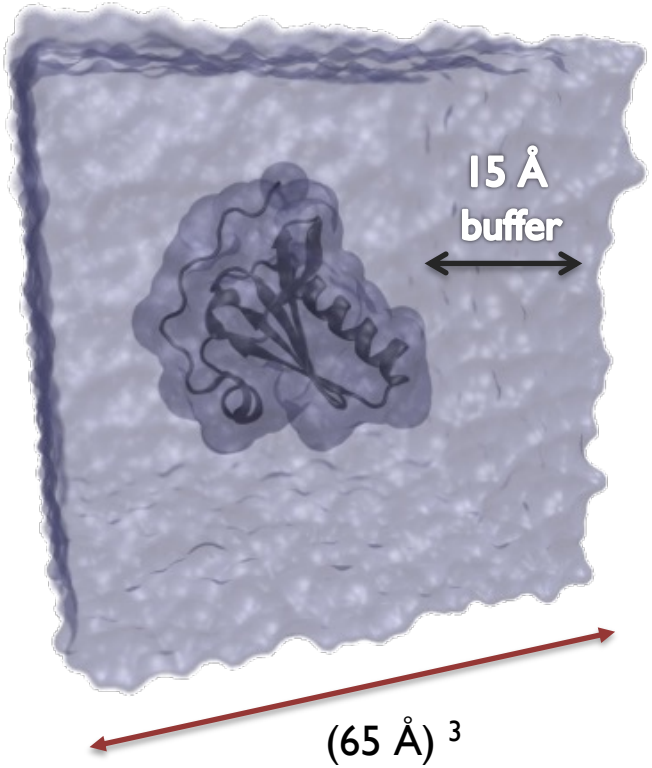
D187N: **6H1F**, 1.9 Å

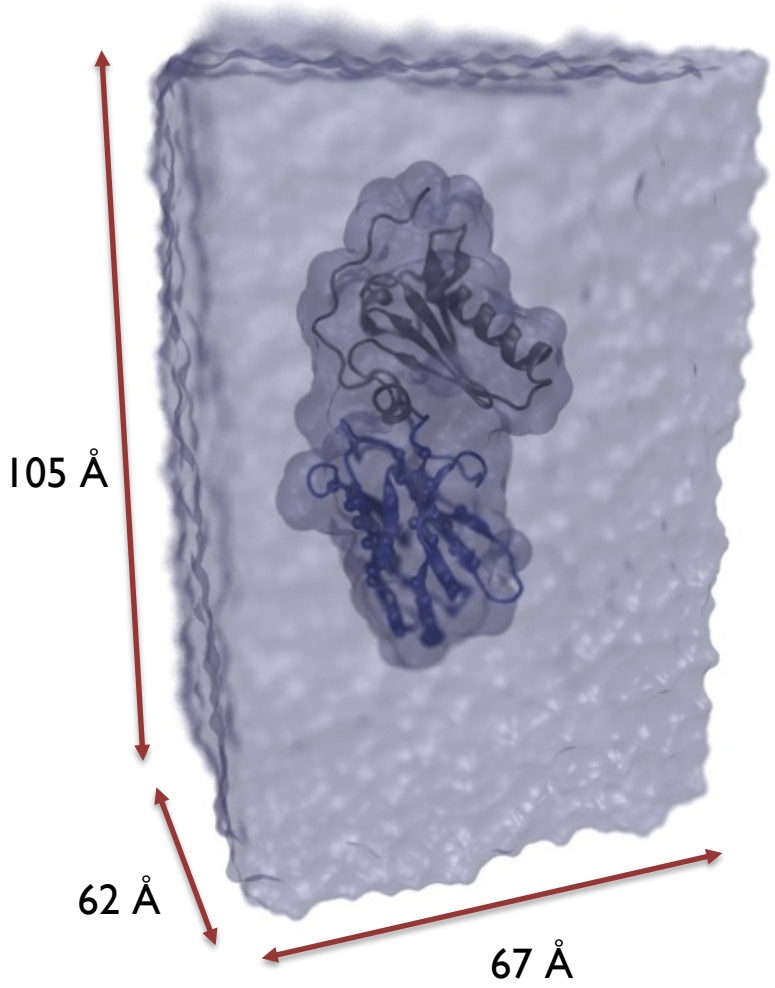
GSN ± Nb11 MD simulations

- Unbiased sampling @300 °K
- 100 mM NaCl
- Harmonic restraints:
  SS Nb1 @ 0.03 kcal/mol/Å^2

CHARMM36

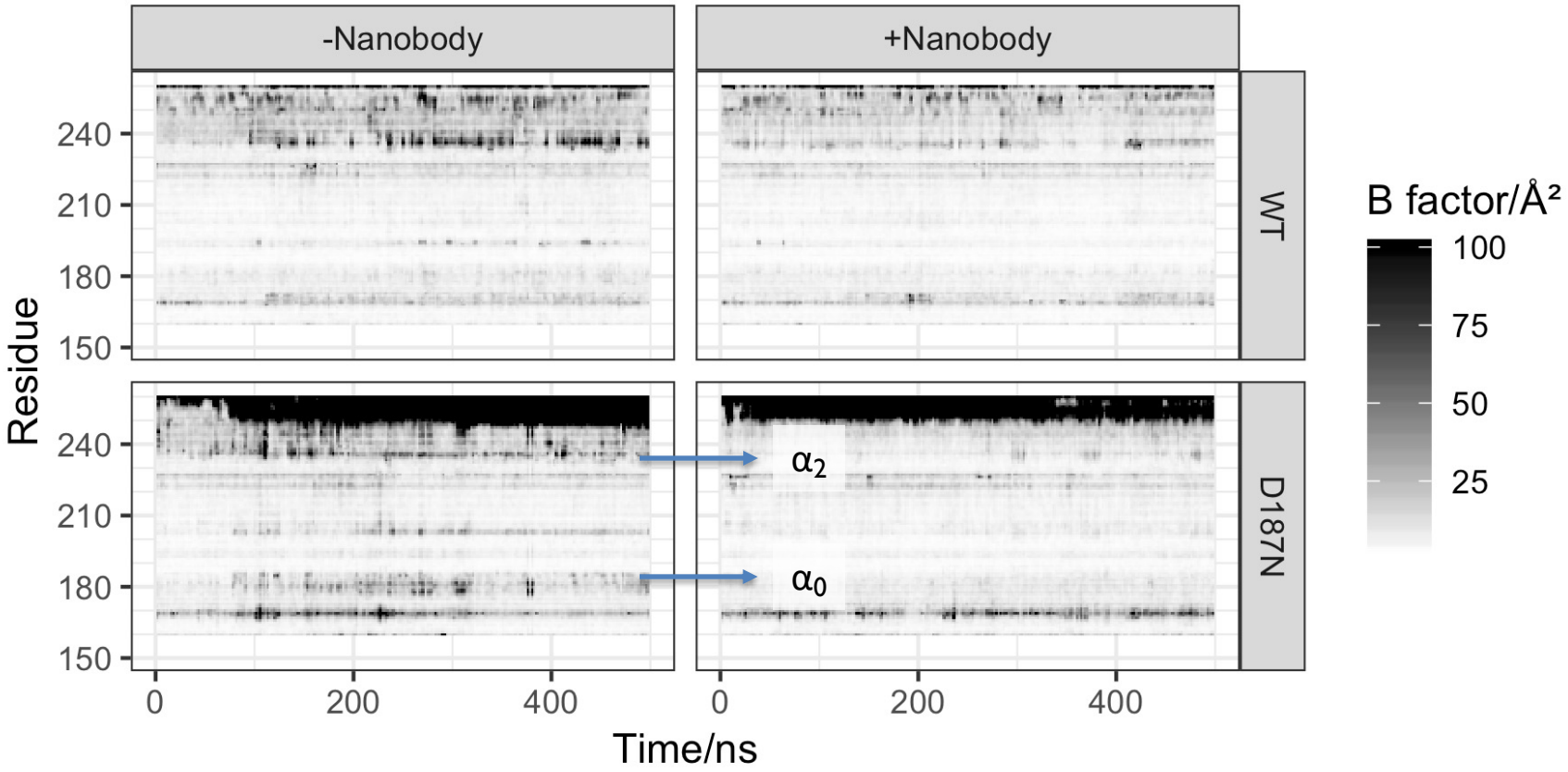
~3 μs tot. ~25k/43k atoms





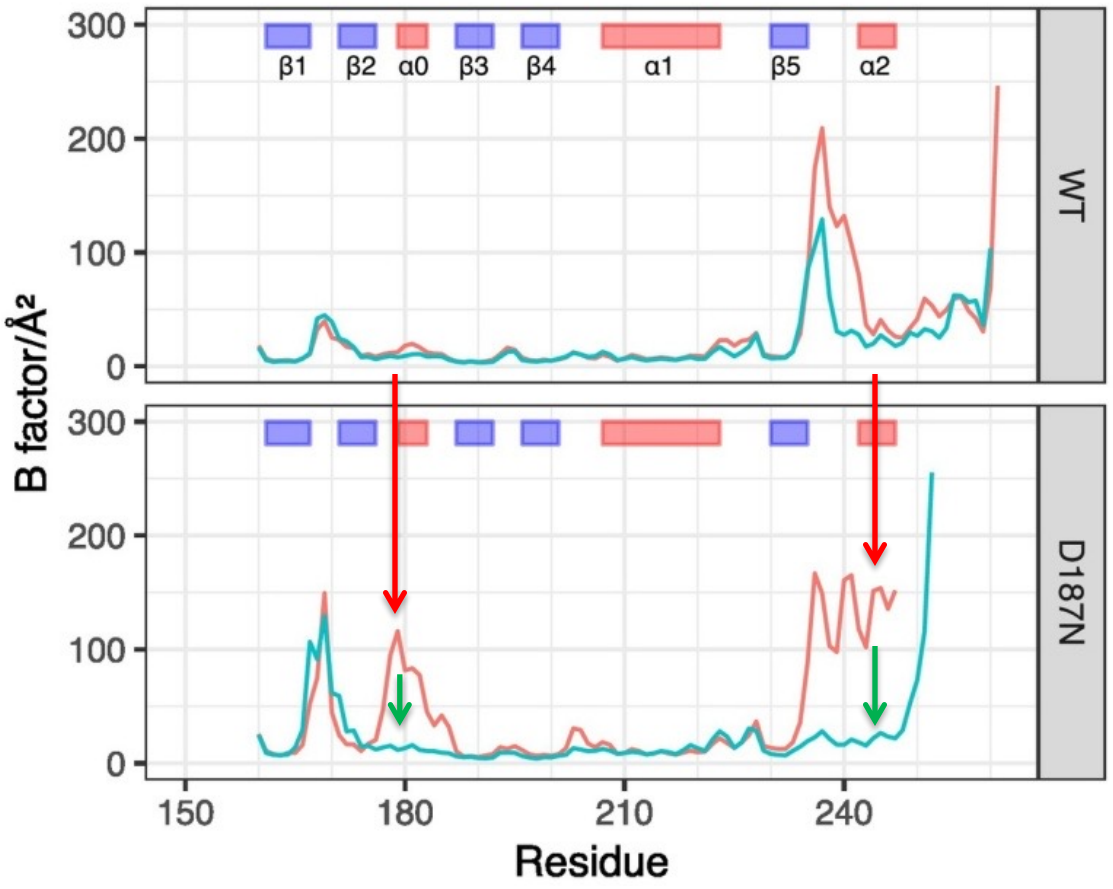
MD results

Sample	Nb11	Ca^{2+}	Simulated time (ns)	C-terminal disorder onset
WT_{G2}	−	+	800	Not observed
WT_{G2}	+	+	750	Not observed
$D187N_{G2}$	−	−	748	After 83 ns
$D187N_{G2}$	+	−	512	After 40 ns



A matter of dynamics?

$B = (8\pi^2/3)\,\mathrm{RMSF}^2$

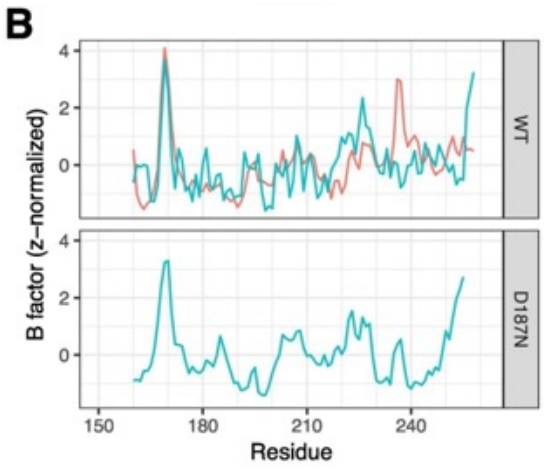


WT: Nb11 has ~ no effect on fluctuation profile (control)

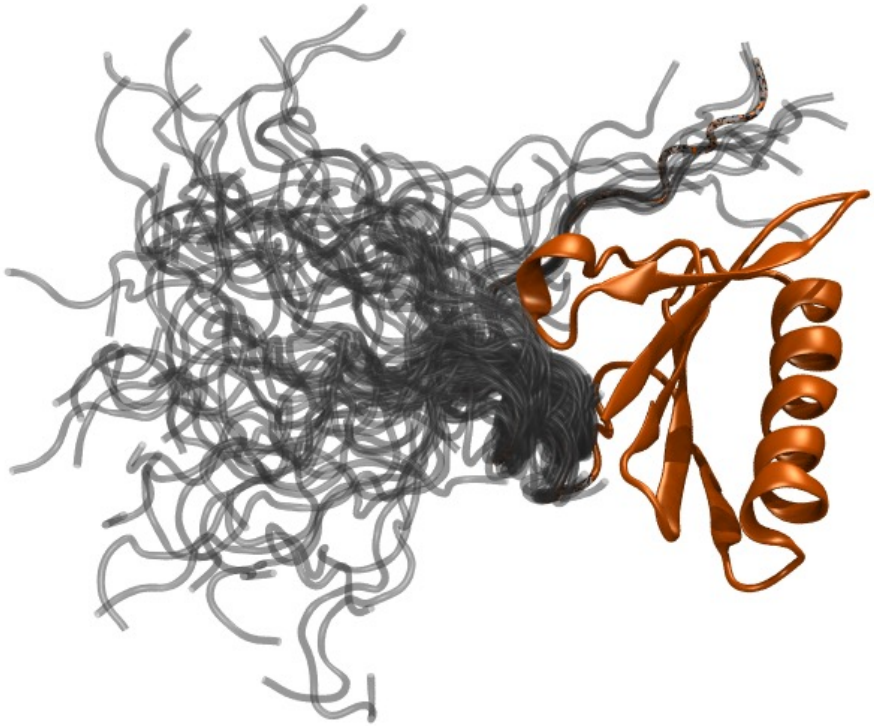
Mutant: destabilized α_0, α_2

Nb11: complete stability recovery

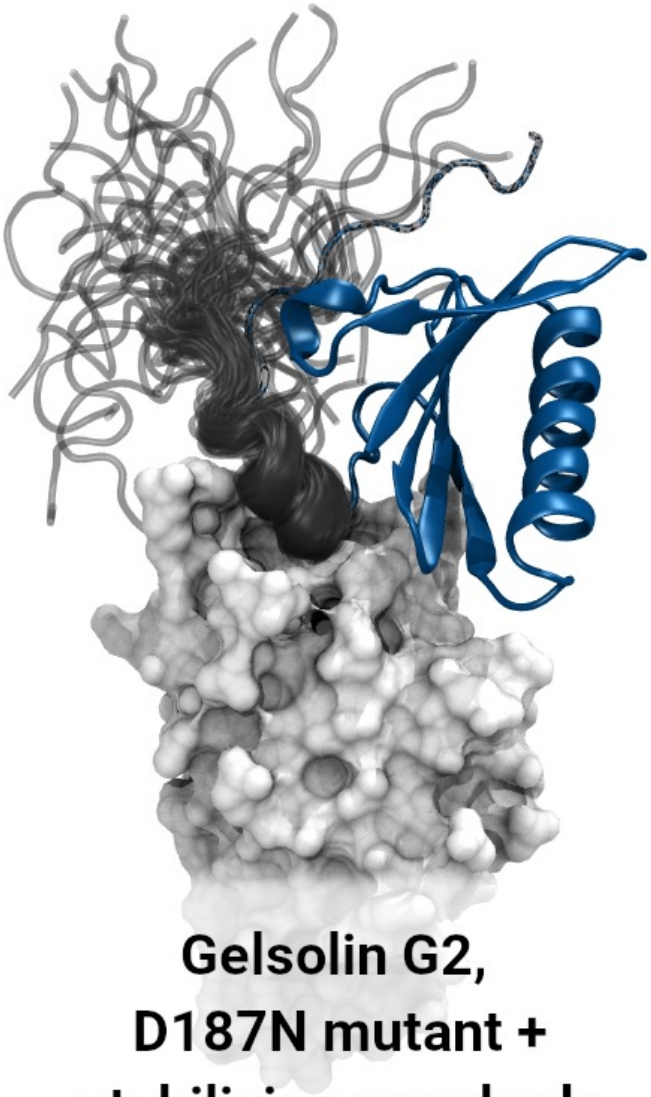
MD vs exp. B-factors →



Reduction of disorder at C-terminus



Gelsolin G2,
D187N mutant



Gelsolin G2,
D187N mutant +
stabilizing nanobody

doi:10.1016/j.bbadis.2019.01.010

C

D187N

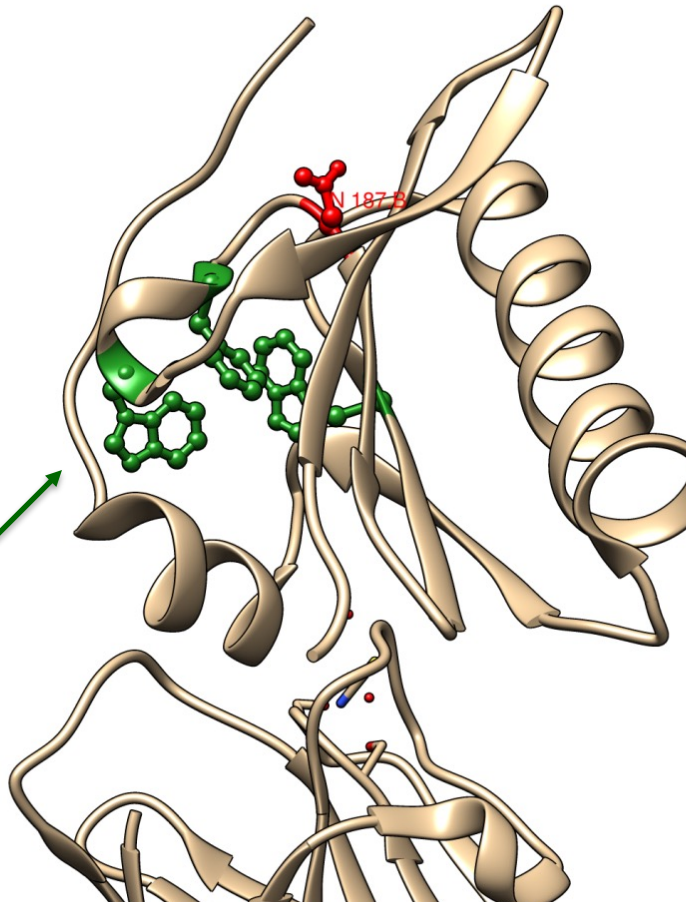
D187N
+Nb11

Initial configuration

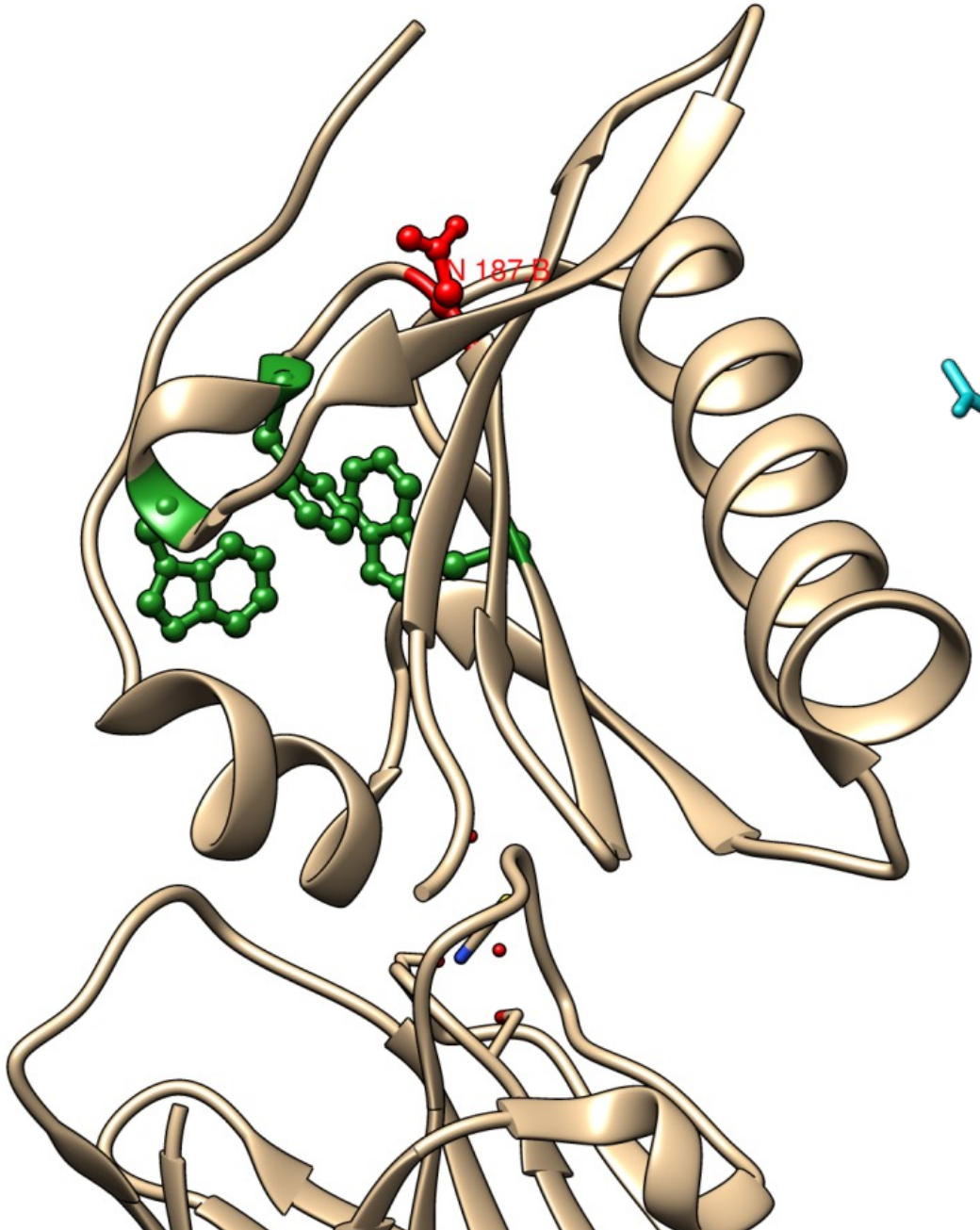
Reduction of C-terminus entropy

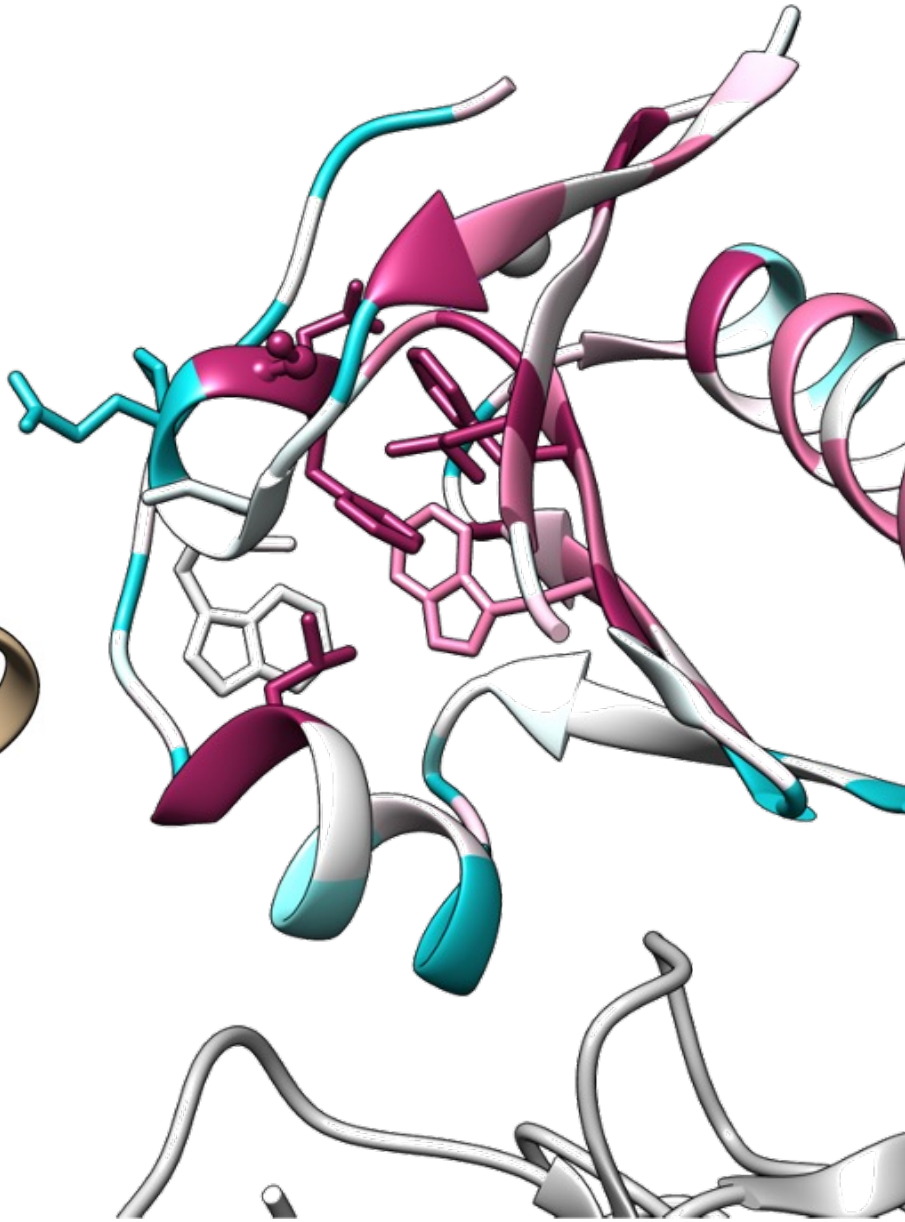
Stabilization of α_0 and α_2

Preservation of the $W_{180}F_{183}W_{200}$ core



W180, F183, W200 form a **conserved** hydrophobic core centered ~ at F183 (α0)



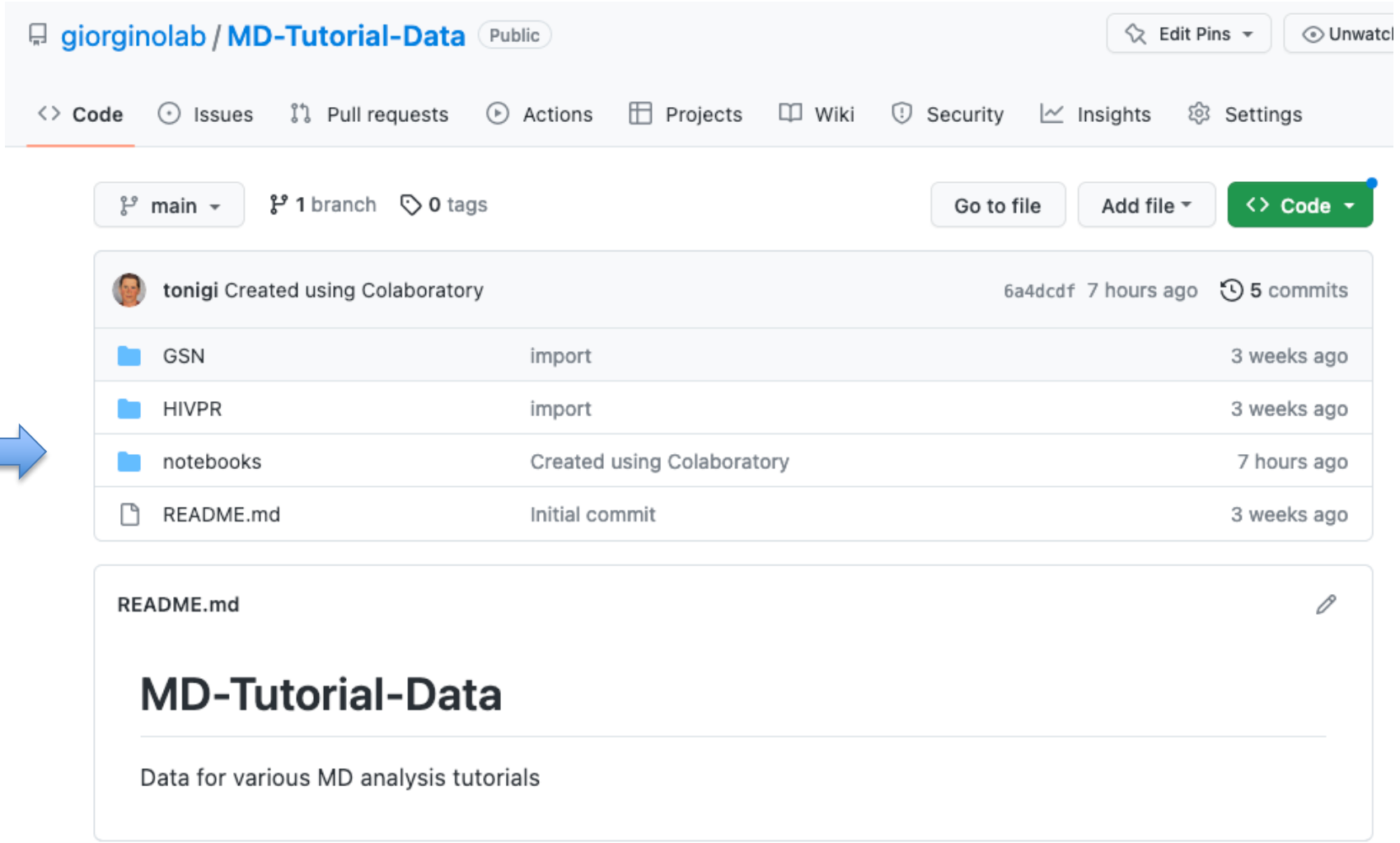


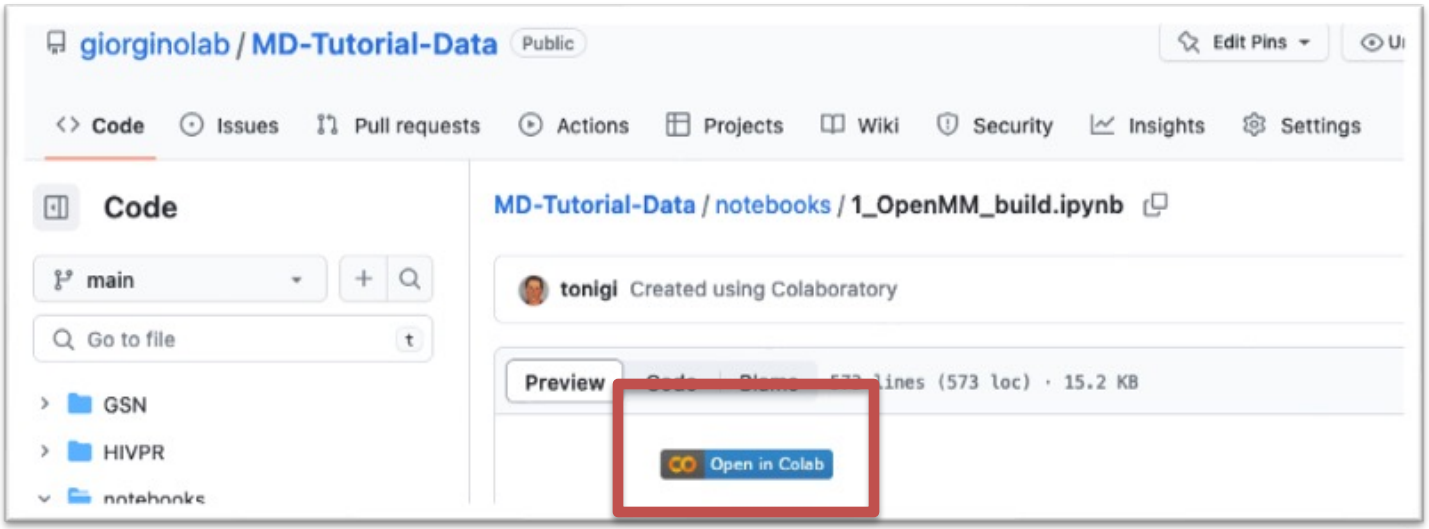
In practice

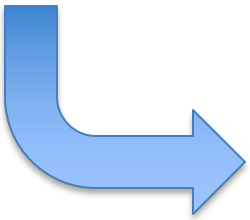
Using OpenMM on Google Colab

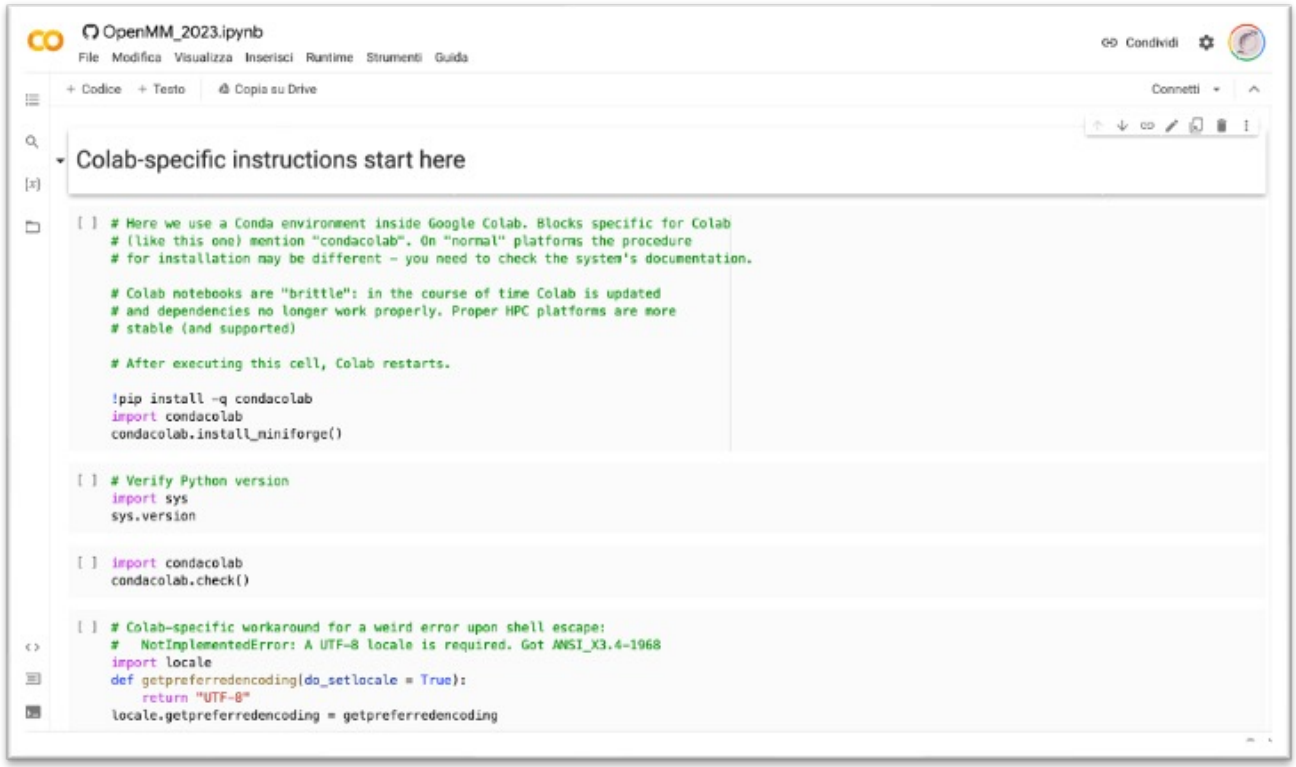
- We'll test OpenMM on Google Colab to run molecular dynamics simulations without the need for installing any software on your local machine.
- **Google Colab** is a free Jupyter environment that allows you to run Python code in the cloud. GPUs runtimes are available.
- To use OpenMM on Google Colab or locally, open the provided notebook (read the comments)

https://github.com/giorginolab/MD-Tutorial-Data




giorginolab / MD-Tutorial-Data
Public
Edit Pins
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Code
main
Go to file
t
GSN
HIVPR
notebooks
MD-Tutorial-Data / notebooks / 1_OpenMM_build.ipynb
tonigi Created using Colaboratory
Preview
lines (573 loc) · 15.2 KB
Open in Colab




OpenMM_2023.ipynb
File Modifica Visualizza Inserisci Runtime Strumenti Guida
Condividi
+ Codice + Testo
Copia su Drive
Connetti
Colab-specific instructions start here
Here we use a Conda environment inside Google Colab. Blocks specific for Colab
(like this one) mention "condacolab". On "normal" platforms the procedure
for installation may be different - you need to check the system's documentation.
Colab notebooks are "brittle": in the course of time Colab is updated
and dependencies no longer work properly. Proper HPC platforms are more
stable (and supported)
After executing this cell, Colab restarts.
!pip install -q condacolab
import condacolab
condacolab.install_miniforge()
# Verify Python version
import sys
sys.version
import condacolab
condacolab.check()
# Colab-specific workaround for a weird error upon shell escape:
# NotImplementedError: A UTF-8 locale is required. Got ANSI_X3.4-1968
import locale
def getpreferredencoding(do_setlocale = True):
return "UTF-8"
locale.getpreferredencoding = getpreferredencoding

…when done…

Visualize

- After you have done the simulation, load the minimized PDB and output.dcd in PyMOL
- What about PBCs? Fix with: `pbc_unwrap` …

Questions

- How many atoms?
- How many residues?
- Disulfide bridges?
- How many trajectory frames?
- Simulation length in *actual* time?

More questions

- Does density change? Should it?
- What is the box size? Is it appropriate?
- Relaxation time?
- Plot the log file

Markov-state modeling of biomolecular systems

Toni Giorgino
National Research Council of Italy
toni.giorgino@cnr.it
www.giorginolab.it

GitHub
@giorginolab

Consiglio Nazionale delle Ricerche

https://github.com/giorginolab/Markov-Tutorial-Data

Introduction

- The aim of this class is to provide a practical *overview* of Markov state models in computational structural biology
- General intuition: a Markov model *propagates forward in time the distribution of probabilities of a set of discrete states*

Introduction

- 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

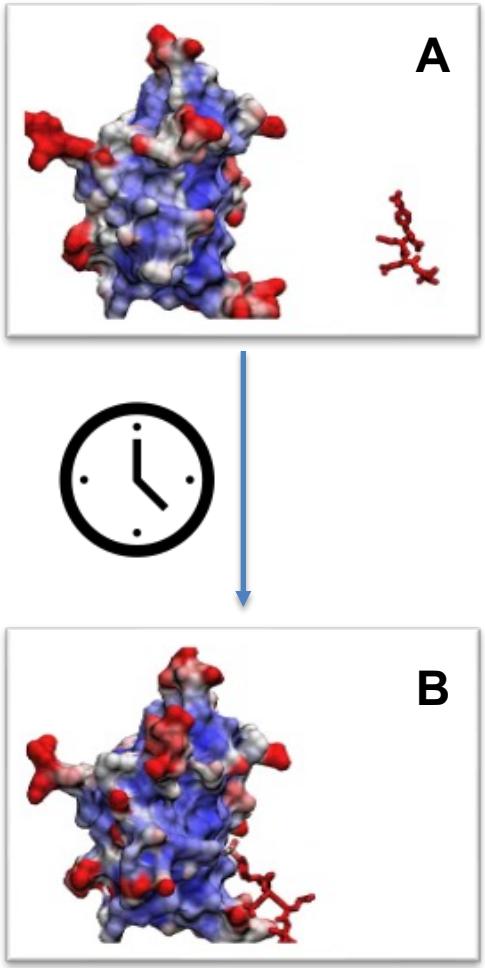
- Can we use an *ensemble* of simulations to estimate dynamic behaviours which happen on timescales longer than each of the observed trajectories?
- In other words, can we leverage several "short-sighted" (non-equilibrium) observations to *extrapolate* long-time behaviour?

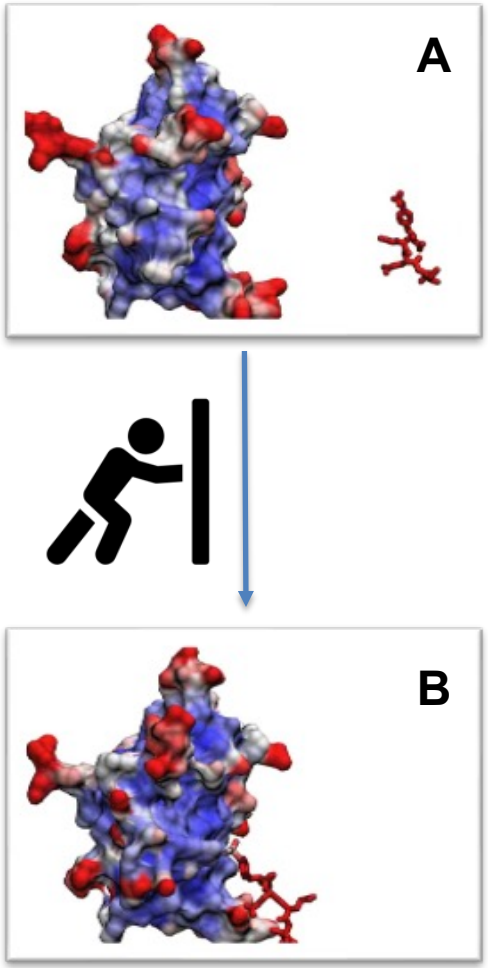
The general idea

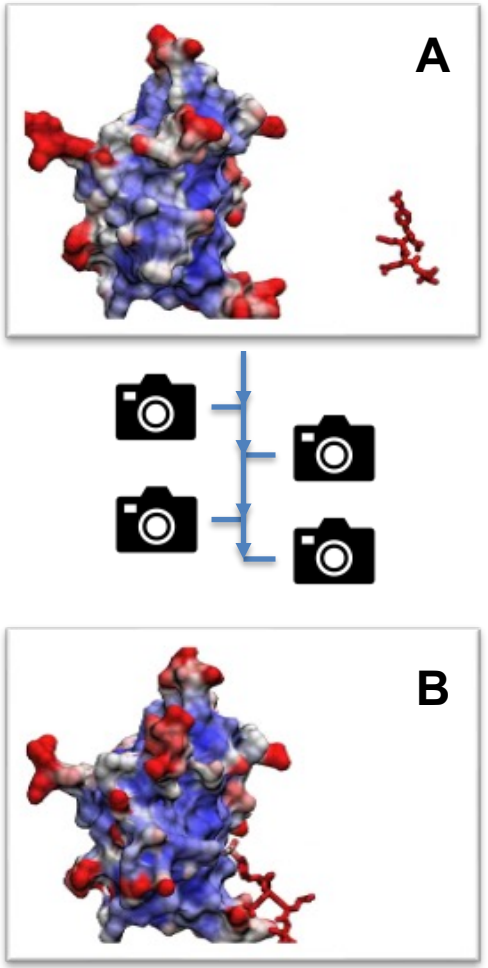
Unbiased sampling

Biased sampling

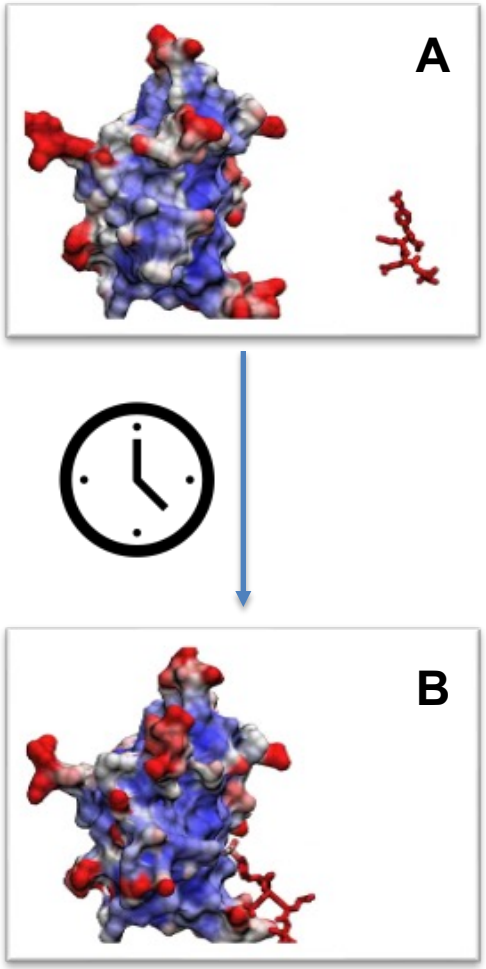
Markov-state modeling







Unbiased sampling



System evolves in time*.

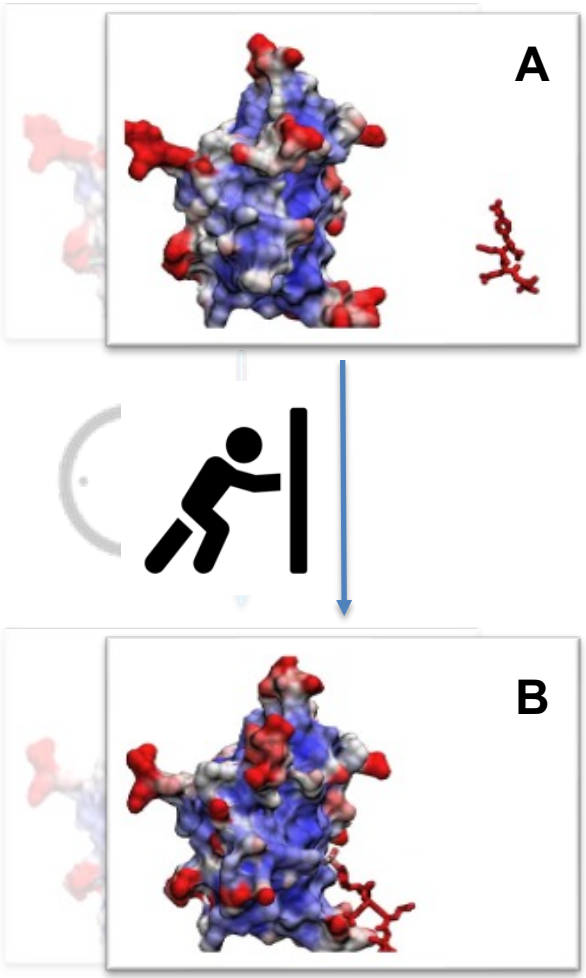
We do not "guide" the system with biases in any particular direction.

Pro: Kinetics are "true"

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

* Under the influence of internal end external forces designed to approximate "real" ones.

Biased sampling



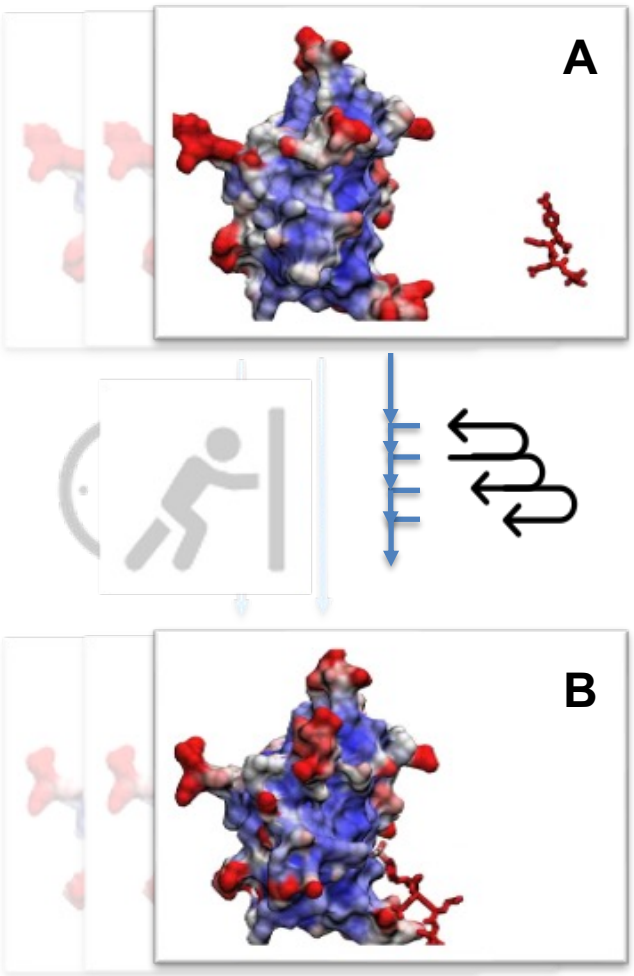
Various strategies* exist to "push" the system *towards* desired states

Pros: Sampling is accelerated..
Can recover ΔG.

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

* Examples: metadynamics, steered MD, …

Markov-state modeling



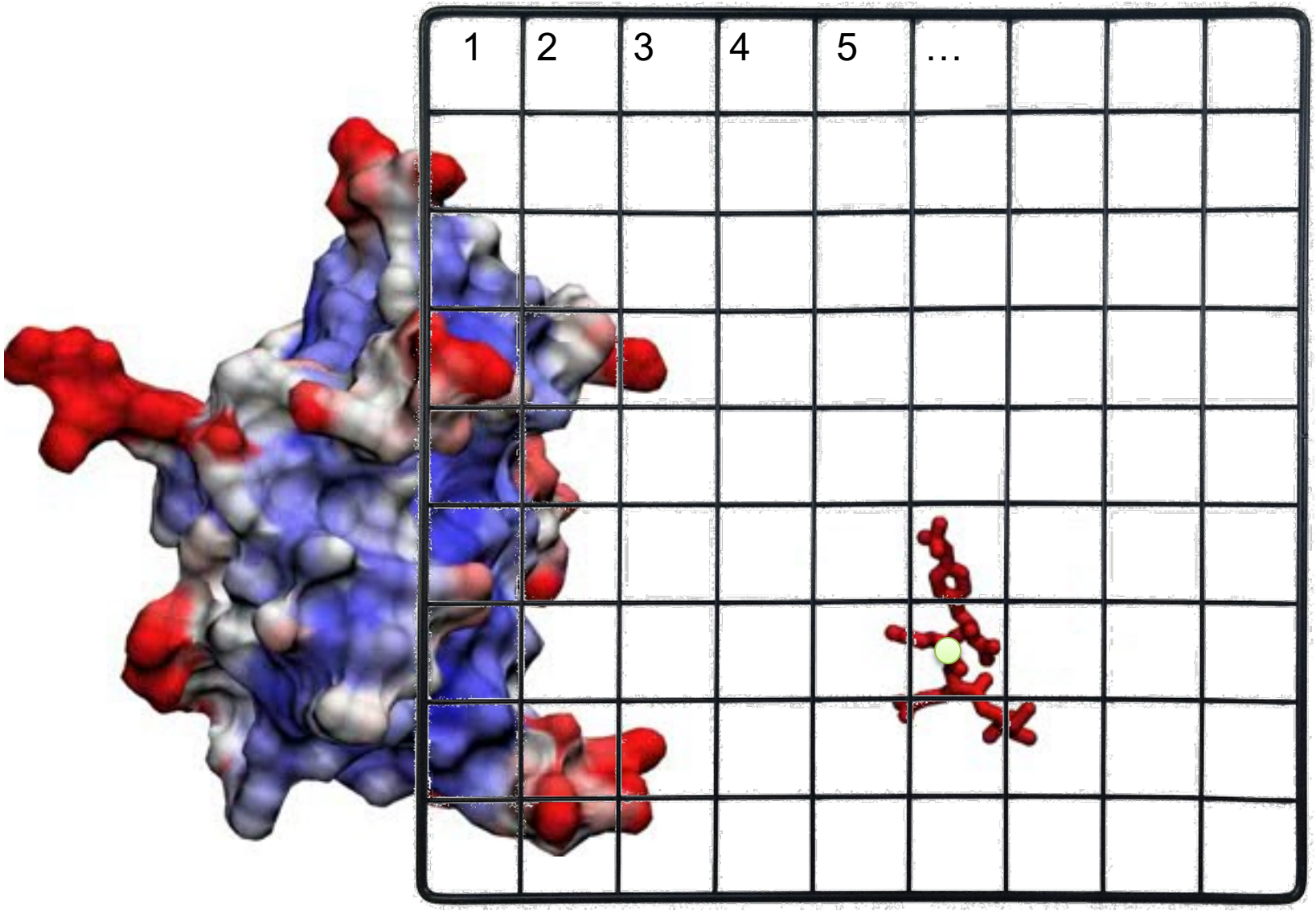
Decompose a system in multiple *states*. Use shorter trajectories to compose a statistical* picture.

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

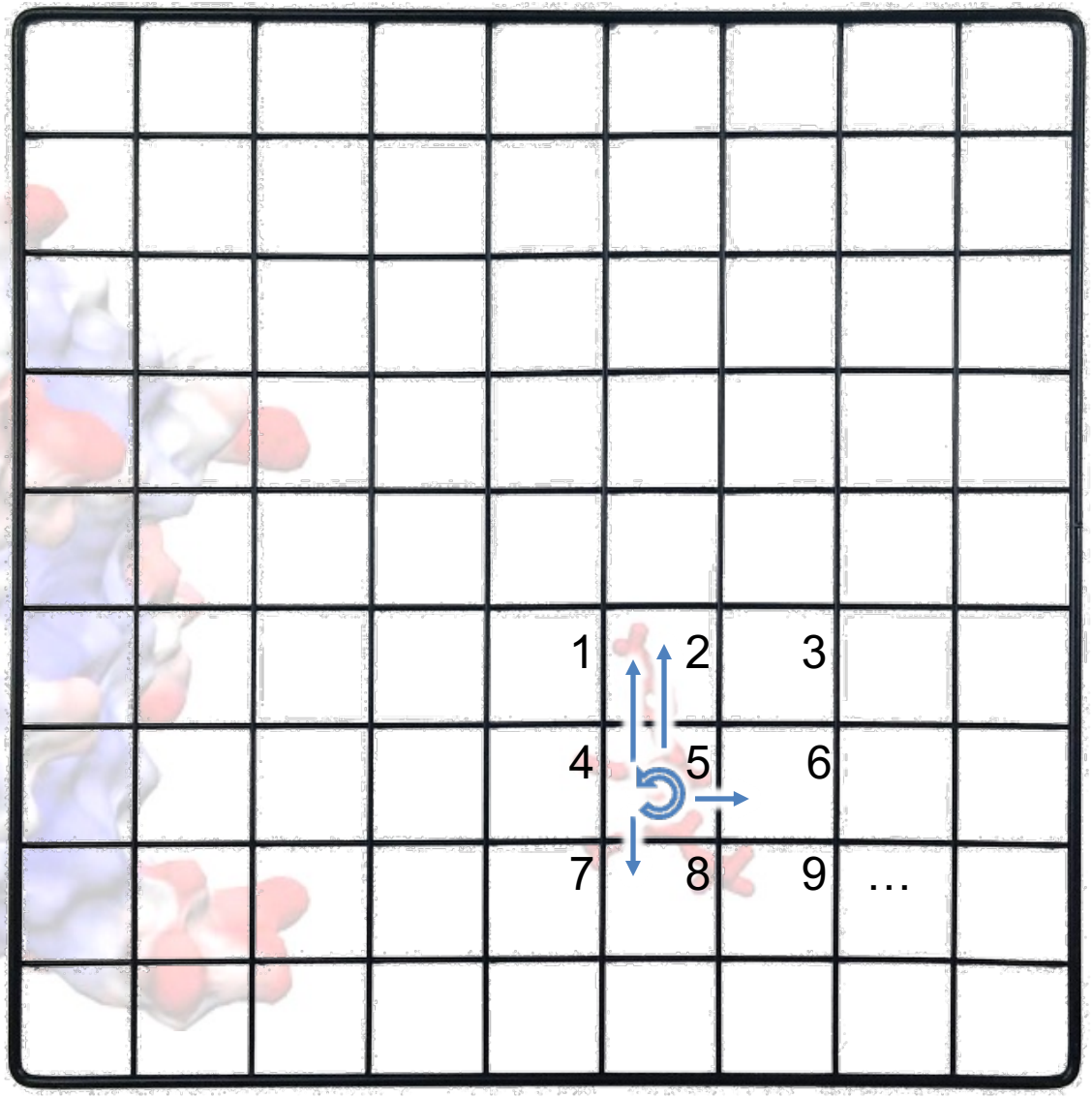
Cons: Non obvious (but software helps).
Sampling still needed for convergence.

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

Decomposing the state space


1
2
3
4
5
…

Count the state transitions made within a time interval (τ)



$T_{56}=1$

$T_{52}=2$

$T_{58}=1$

$T_{55}=1$

Total: $N_5=5$

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

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

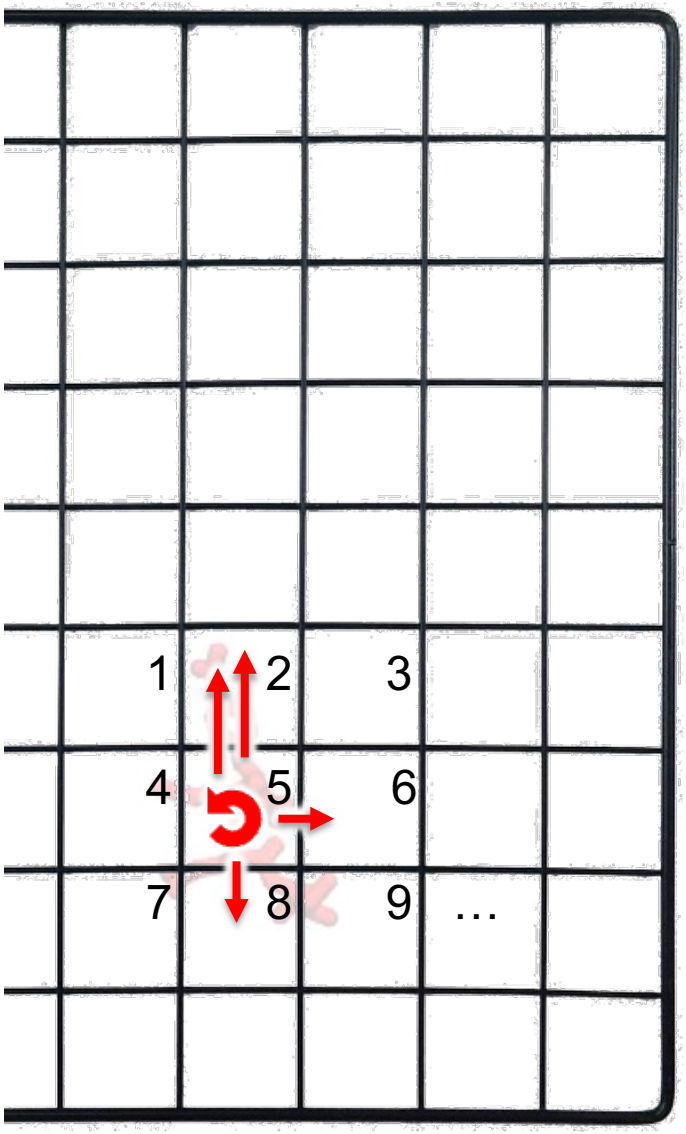
$T_{58}=1$ $p_{58} = 1/5 = 0.2$

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

$N_5=5$ $p_{5*} = \Sigma_i p_{5i} = 1$

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

Row 5 of the
transition probability matrix P

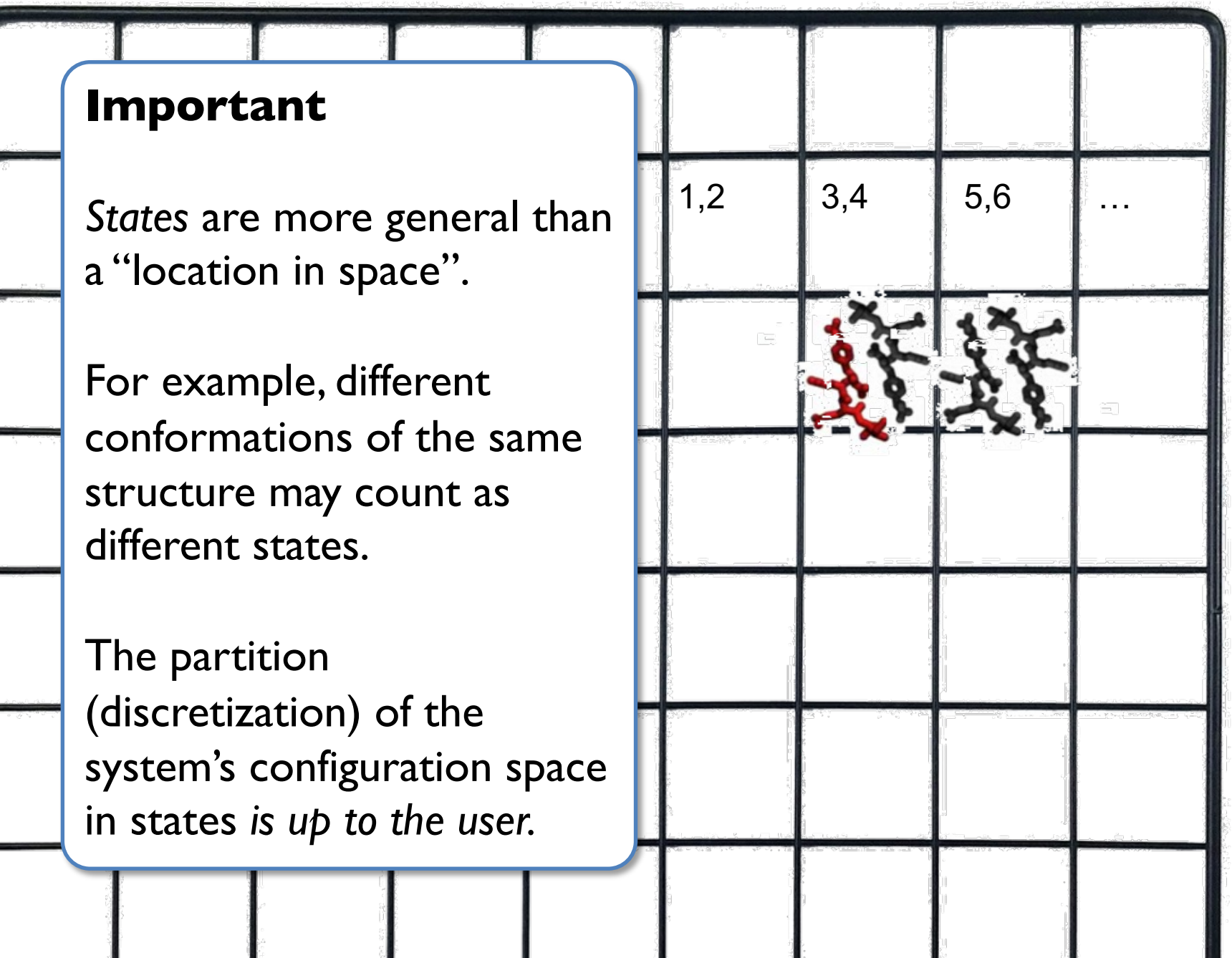


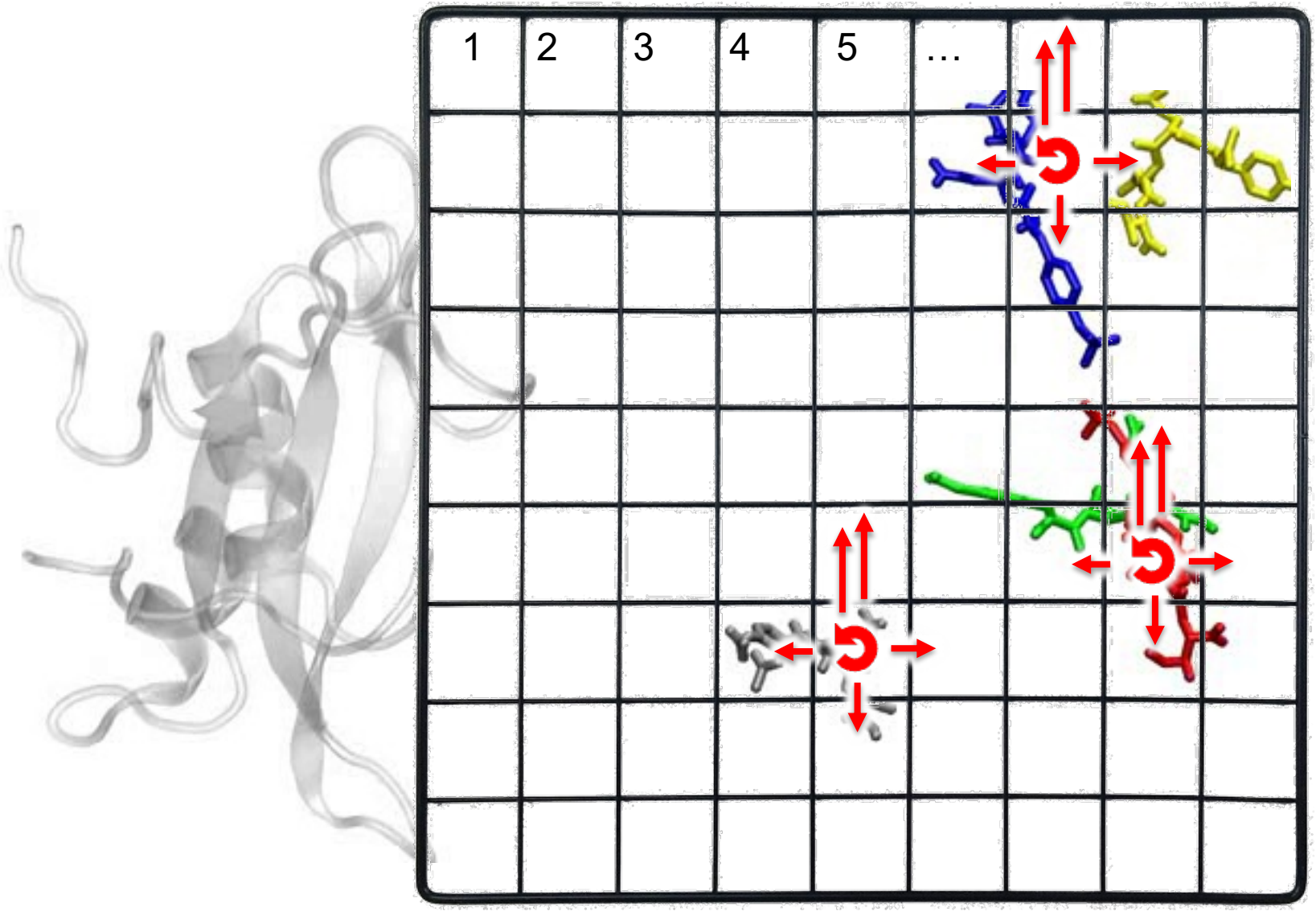
Important

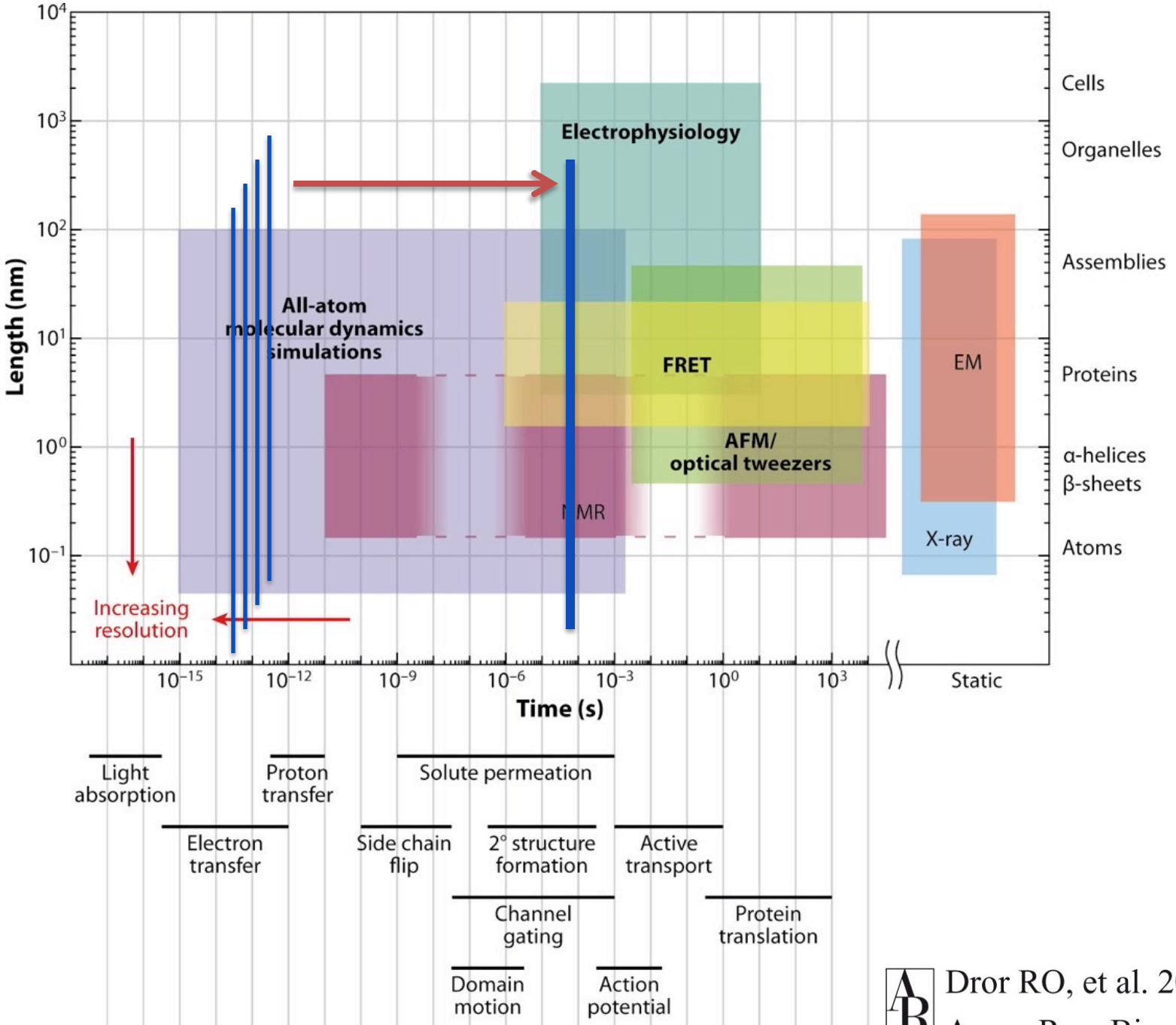
States are more general than a "location in space".

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.*




1
2
3
4
5
…



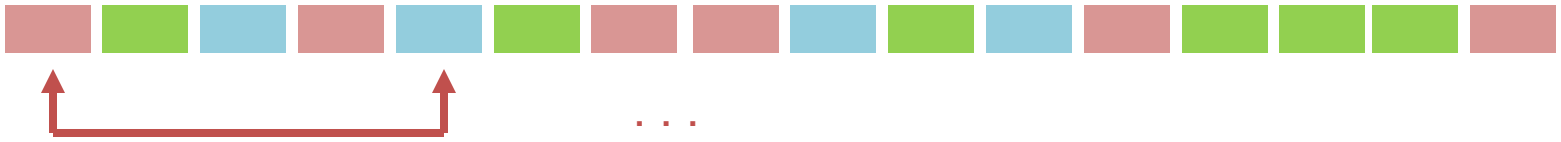
Dror RO, et al. 2012.
Annu. Rev. Biophys. 41:429–52

Learning transition matrices from trajectories

Markov models

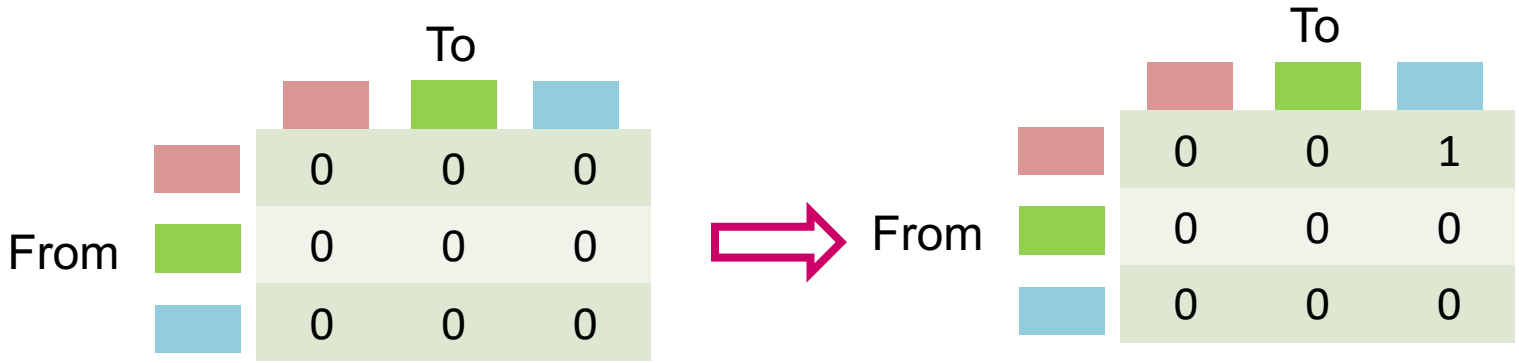
Total sampled time (e.g. 100 μs)

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



Lag time τ (here: 4 time units)

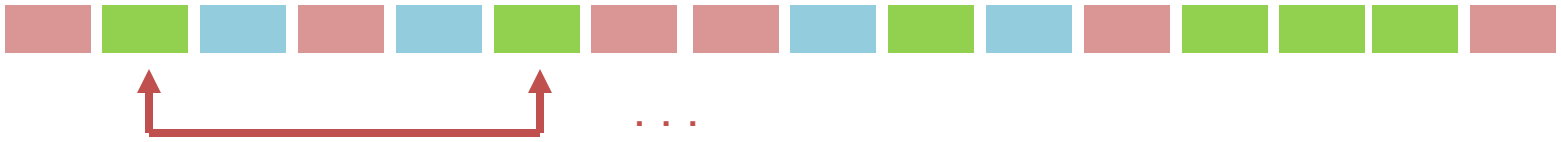
Compute the transition probability matrix sliding a window of lag time τ



Markov models

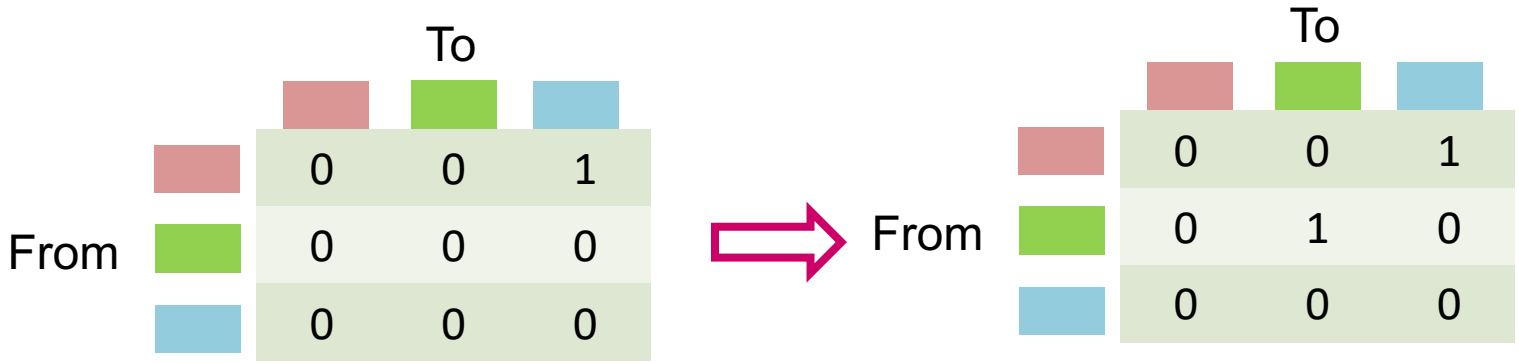
Total sampled time (e.g. 100 μs)

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

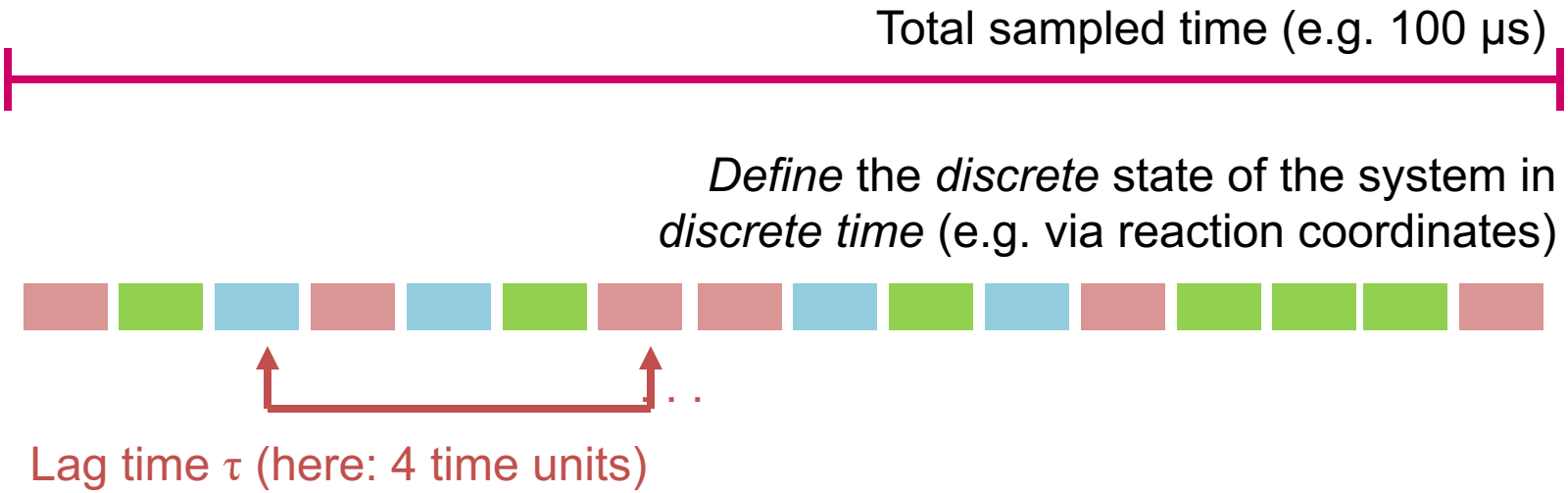


Lag time τ (here: 4 time units)

Compute the transition probability matrix sliding a window of lag time τ



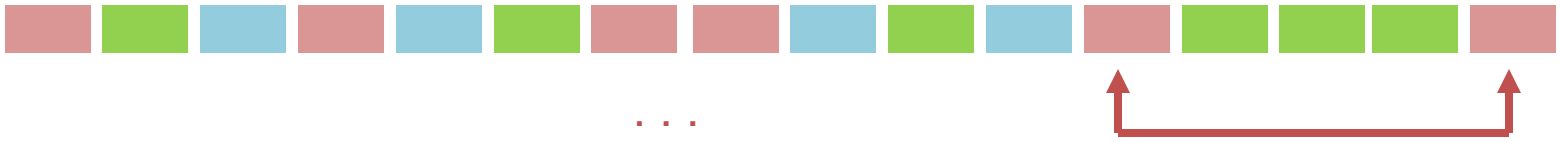
Markov models



Markov models

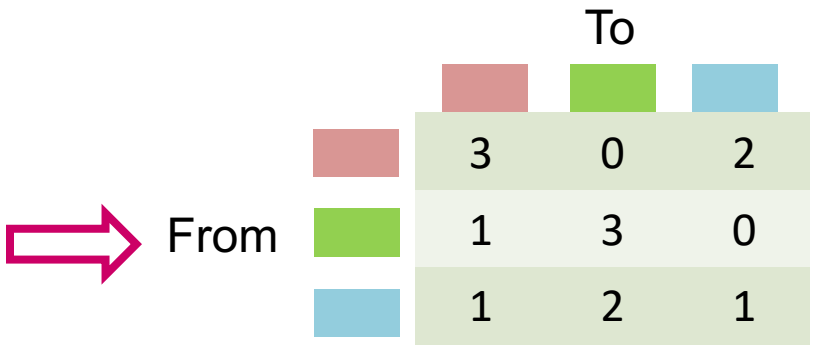
Total sampled time (e.g. 100 μs)

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

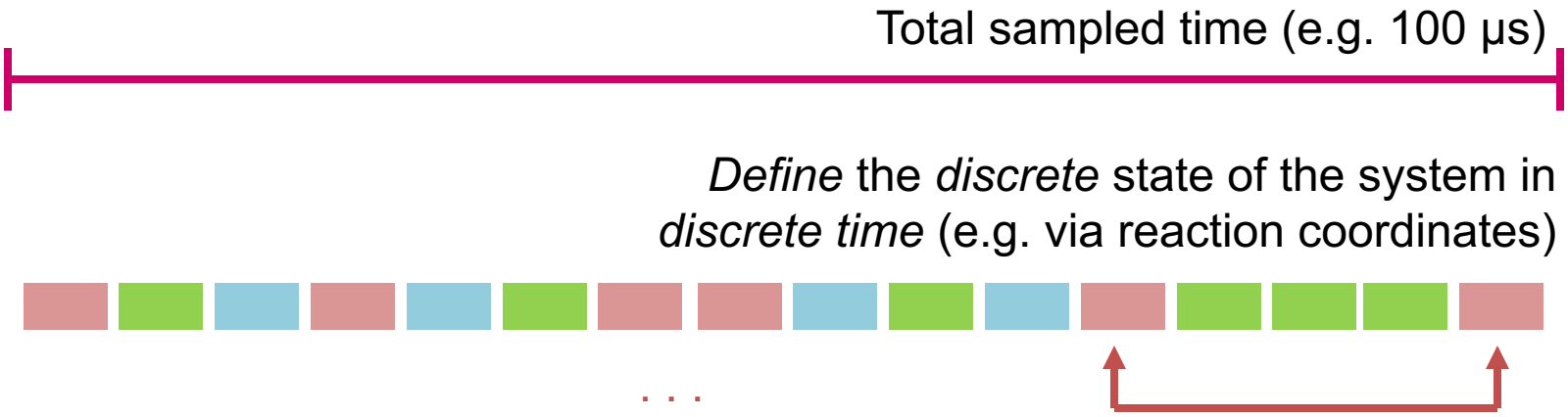


Lag time τ (here: 4 time units)

Compute the transition probability matrix sliding a window of lag time τ



Markov models



Transition counts

From \ To			
	3	0	2
	1	3	0
	1	2	1

Normalize by rows

Transition probabilities

From \ To				Σ_j
	3/5	0	2/5	1
	¼	¾	0	1
	¼	2/4	¼	1

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).

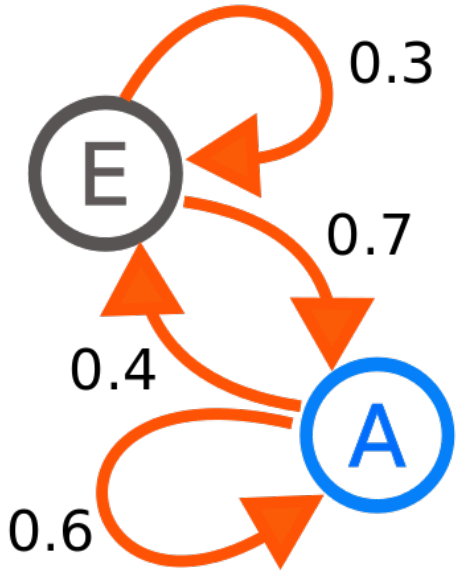
(Note how "Early" and "late" events are squashed in the same matrix.)

Discrete-time Markov Chains

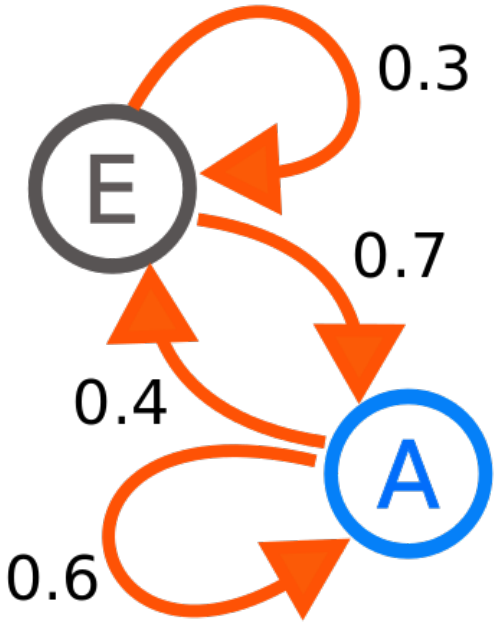
Andrei Markov
1856-1922

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



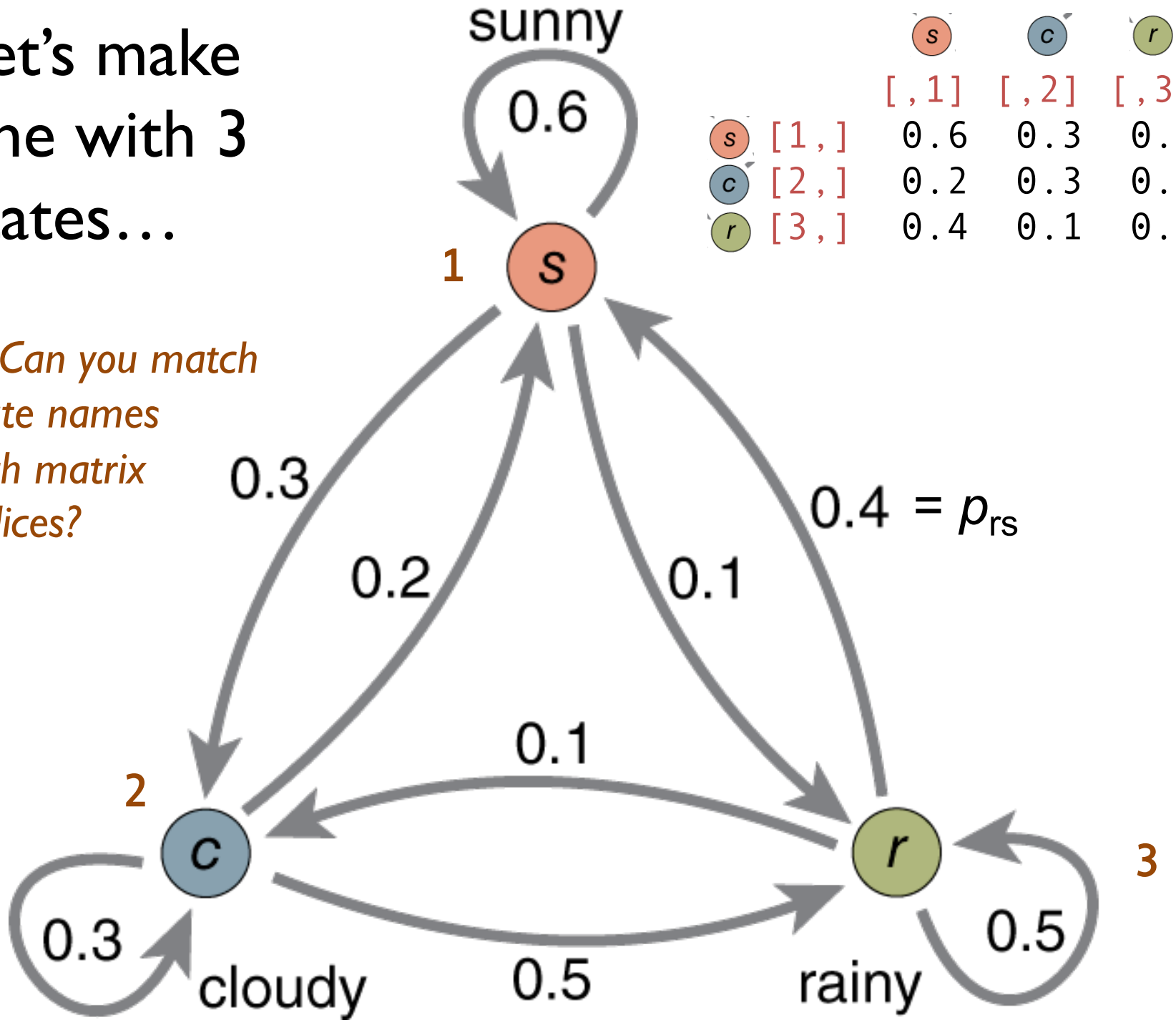
P_{ij} = *Probability to change state from i to j at each time point*

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

i \ j	**A**	**E**
A	0.6	0.4
E	0.7	0.3

Let's make one with 3 states…

Q: Can you match state names with matrix indices?

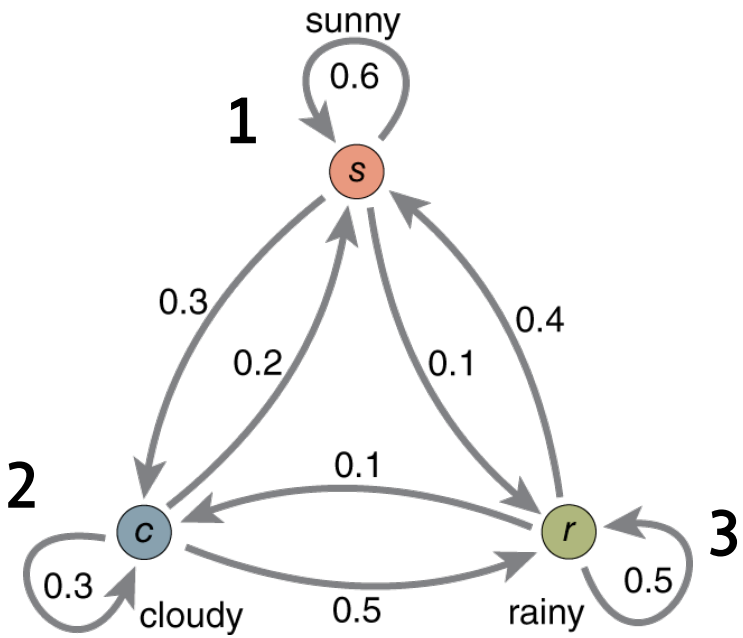


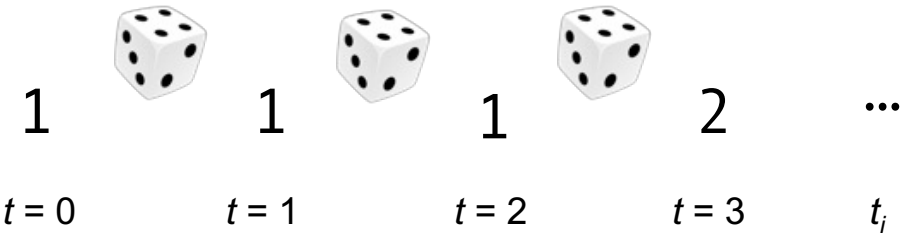
		s	c	r
		[,1]	[,2]	[,3]
s	[1,]	0.6	0.3	0.1
c	[2,]	0.2	0.3	0.5
r	[3,]	0.4	0.1	0.5

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.





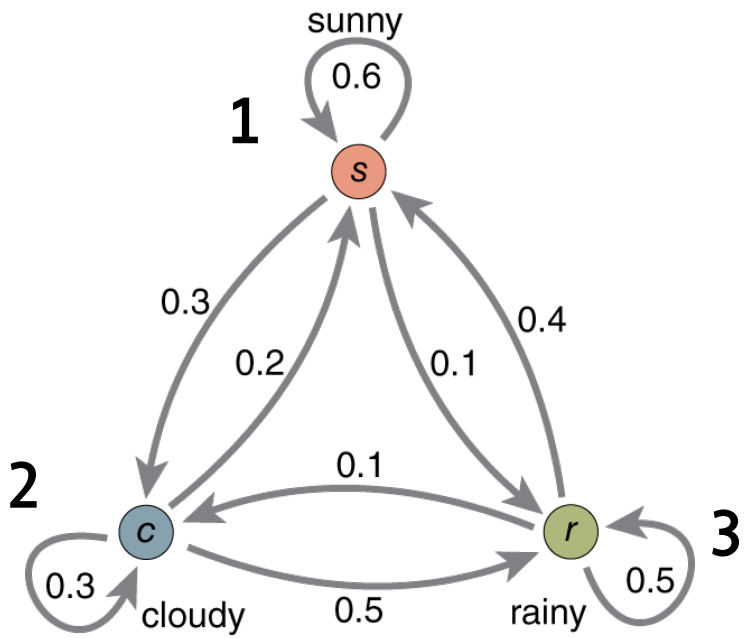
1 1 1 2 2 3 3 1 1 1 2 2 2 2 1 1 1 1 1 1 3 1 1 1 1 2 3 1 1 3…

https://github.com/giorginolab/Markov-Tutorial-UniMi-2021

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

(1000 times)

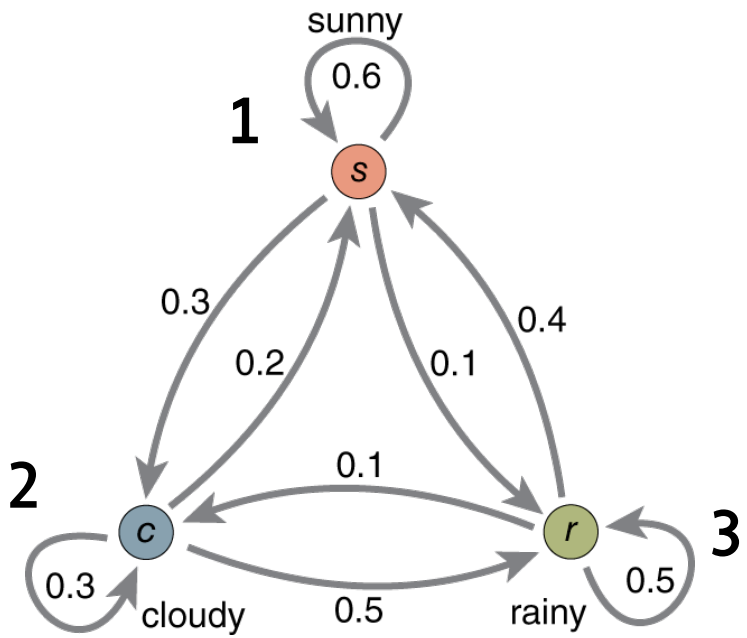
```
1    2    3
431 220 349
```

$p_\infty(1) = 0.431$
$p_\infty(2) = 0.220$
$p_\infty(3) = 0.349$

$\Sigma_i\, p_\infty(i) = 1$

https://github.com/giorginolab/Markov-Tutorial-UniMi-2021

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 (ΔG) of that state!

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 2 3 1 1 3…

(1000 times)

```
1    2    3
431 220 349
```

$p_\infty(1) = 0.431$
$p_\infty(2) = 0.220$
$p_\infty(3) = 0.349$

$\Sigma_i\, p_\infty(i) = 1$

$G_1 = -k_B T \log(p_1) \simeq 0.50$ kcal/mol
$G_2 = -k_B T \log(p_2) \simeq 0.91$ kcal/mol
$G_3 = -k_B T \log(p_3) \simeq 0.63$ kcal/mol

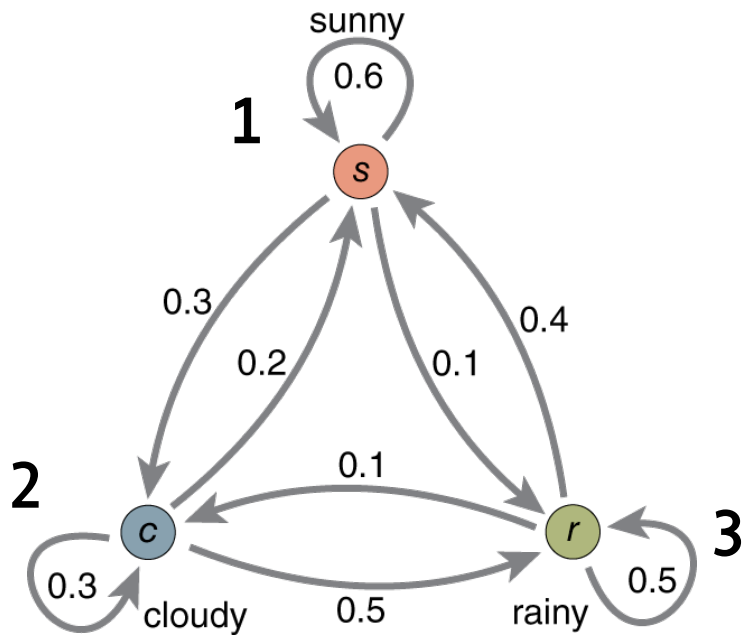
$\Delta G_{11} \doteq 0$ kcal/mol
$\Delta G_{21} \simeq 0.37$ kcal/mol
$\Delta G_{31} \simeq 0.18$ kcal/mol

https://github.com/giorginolab/Markov-Tutorial-UniMi-2021

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 (ΔG) of that state!



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 2 3 1 1 3...

(1000 times)

Sample and count

Compute eigenvectors of matrix *P*

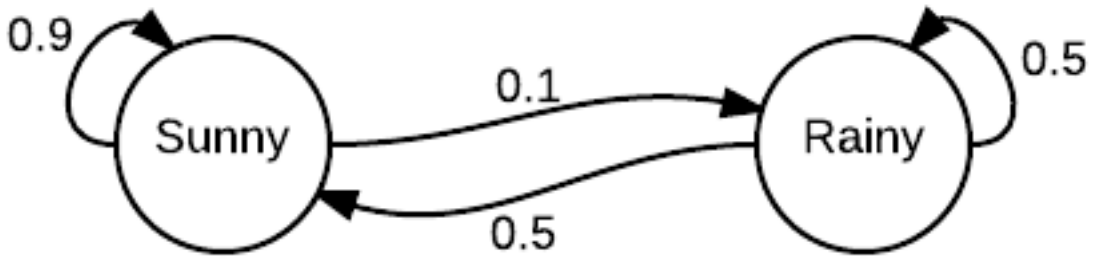
```
     [,1] [,2] [,3]
[1,]  0.6  0.3  0.1
[2,]  0.2  0.3  0.5
[3,]  0.4  0.1  0.5
```

Compute eigenvectors: same result in one shot

https://github.com/giorginolab/Markov-Tutorial-UniMi-2021

A slightly more mathematical point of view

First example



$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

Assume a deterministic initial condition:

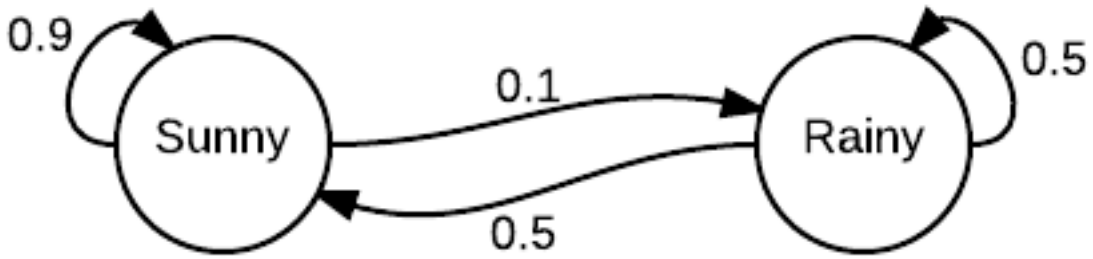
X_0 = Sunny with certainty; *i.e.*,

p(Sunny | t=0)=1 and
p(Rainy | t=0)=0 *i.e.*

$s_0 = [1, 0]$

…now, what is s_1?

First example



$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

What is s_1?

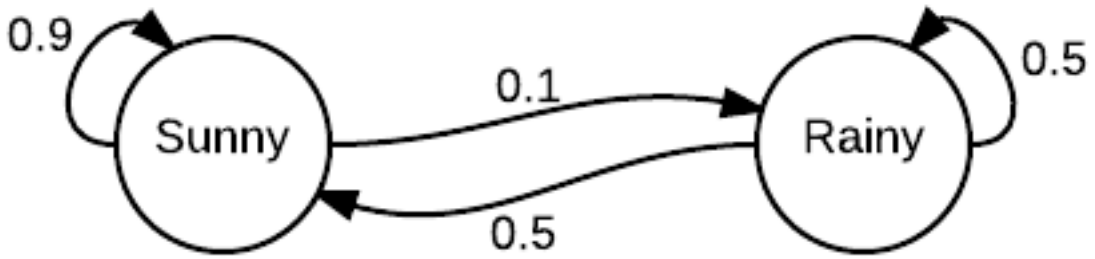
s_1 { p(Sunny | t=1) = 0.9
p(Rain | t=1) = 0.1

In matrix form…

$$\mathbf{s}_1 = \mathbf{s}_0 P$$

…now, what is s_2?

First example



$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$

What is s_2?

s_2 {
p(Sunny | t=2) = 0.9 p(Sunny | t=1) + 0.5 p(Rainy | t=1) = 0.86
p(Rainy | t=2) = 0.1 p(Sunny | t=1) + 0.5 p(Rainy | t=1) = 0.14

In matrix form…

$$\mathbf{s}_2 = \mathbf{s}_1 P$$

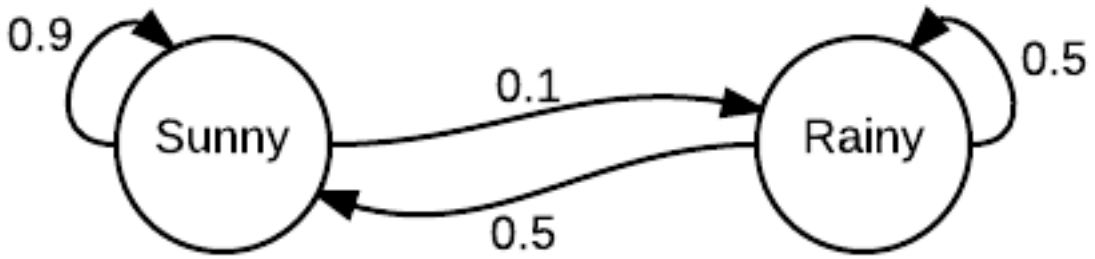
And in general…

$$\mathbf{s}_{t+1} = \mathbf{s}_t P$$

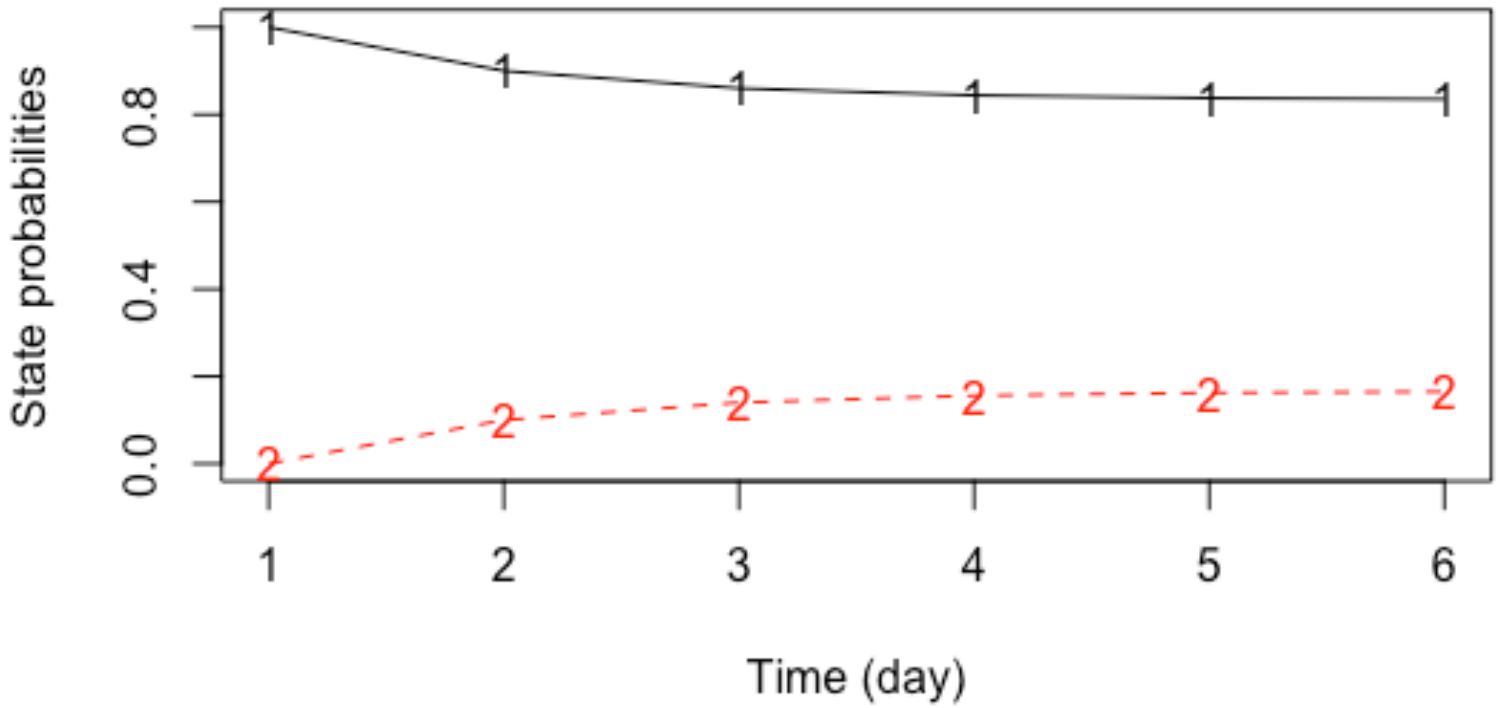
Meaning…

$$\mathbf{s}_t = \mathbf{s}_0 P^t$$

Let's do a numerical test…

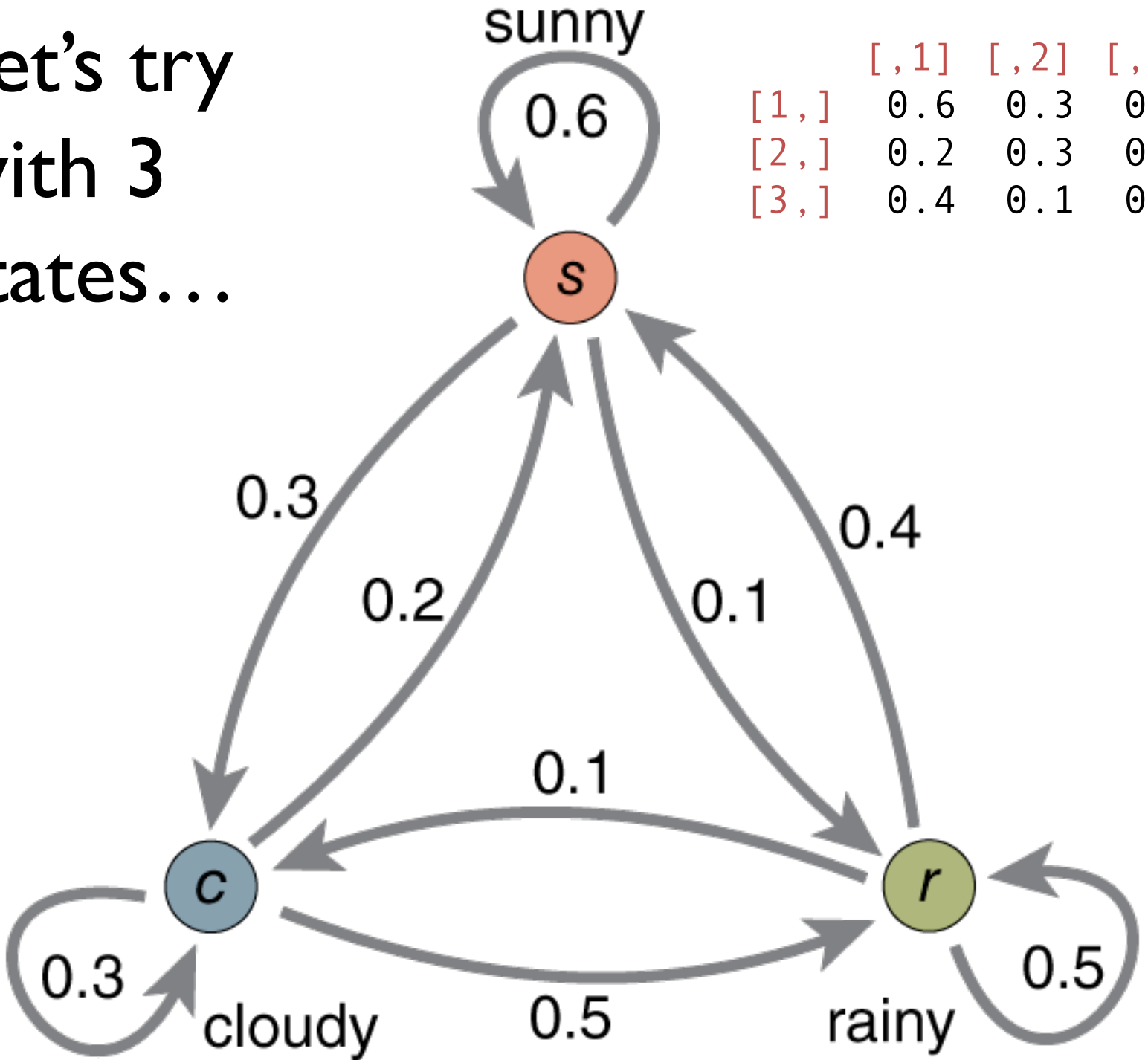


$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}$$



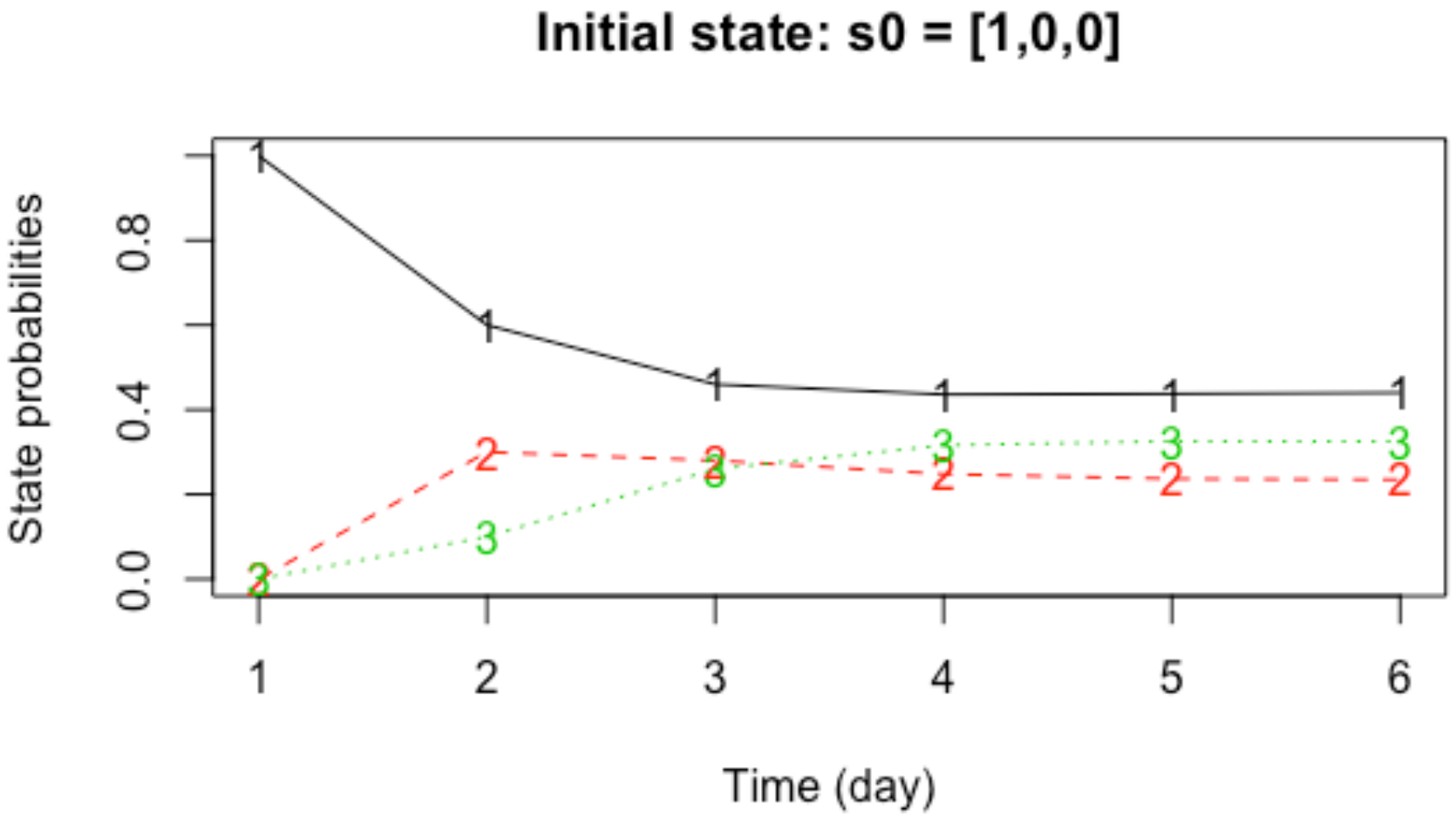
Let's try with 3 states…

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



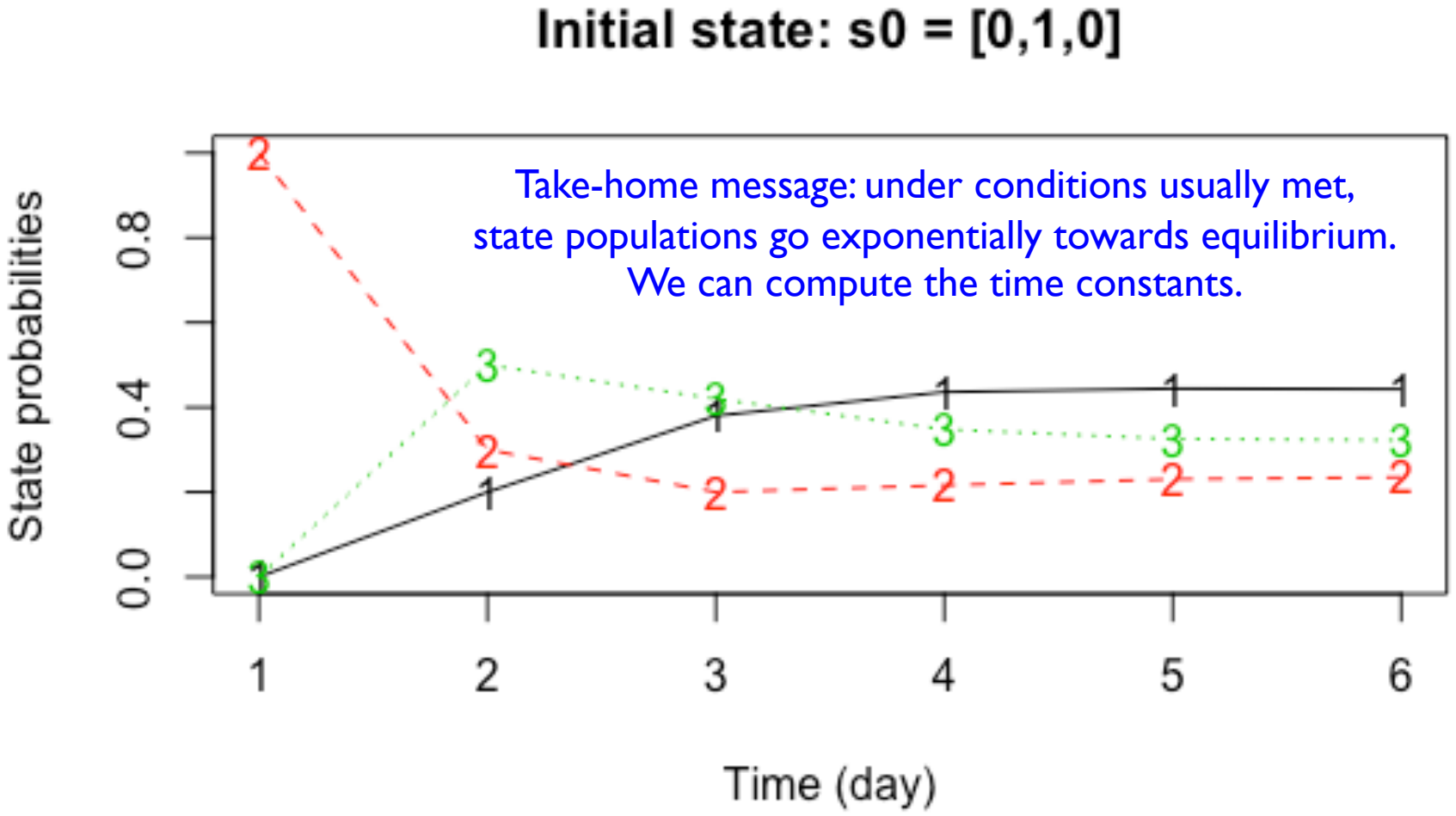
Starting from *Sunny*…


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



Starting from *Cloudy*…


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



Important quantities we can compute

- Stationary distribution (→ eq. probabilities)
- Relaxation times
- Mean-first passage times (→ kinetic rates)
- And others we won't discuss:
  - Committor probabilities
  - Fluxes
  - …

Intermezzo

Google's PageRank

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

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

Kurt Bryan†
Tanya Leise‡

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.rose-hulman.edu/~bryan.

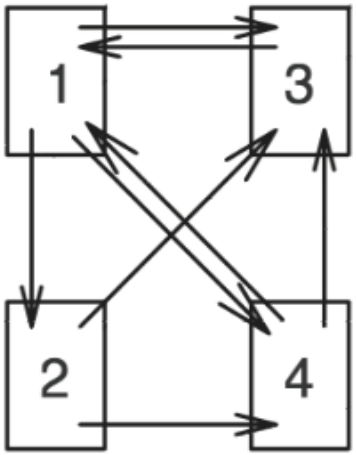
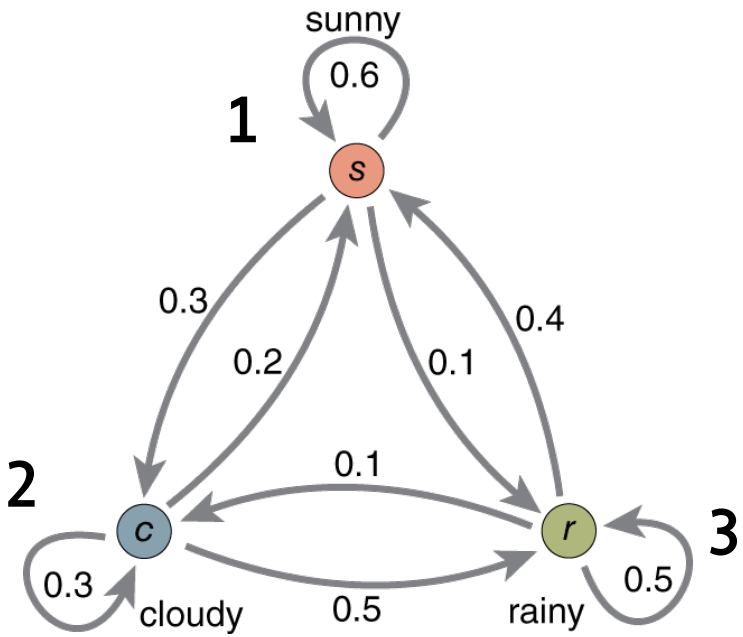


Fig. 1 *An example of a web with only four pages. An arrow from page* A *to page* B *indicates a link from page* A *to page* B.

Kinetics

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

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



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 2 3 1 1 3…

3 2 1 3 2 1 11

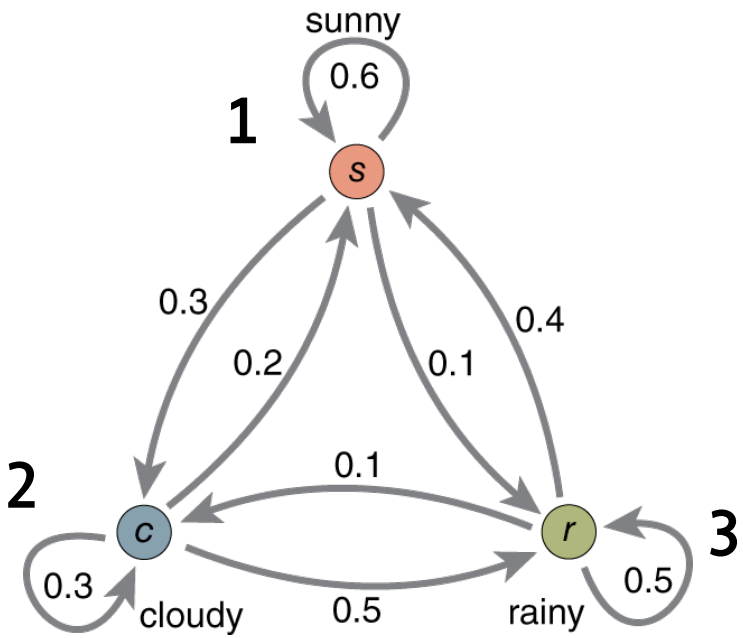
And so on…

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 × τ

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 1 → {2,3} *committor probability*
- States which are equally probable to fall back to the "reactants" or advance to "product" state form the *transition state*.

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 2 3 1 1 3…

2
2
2
2
2
2
3
3

And so on…

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

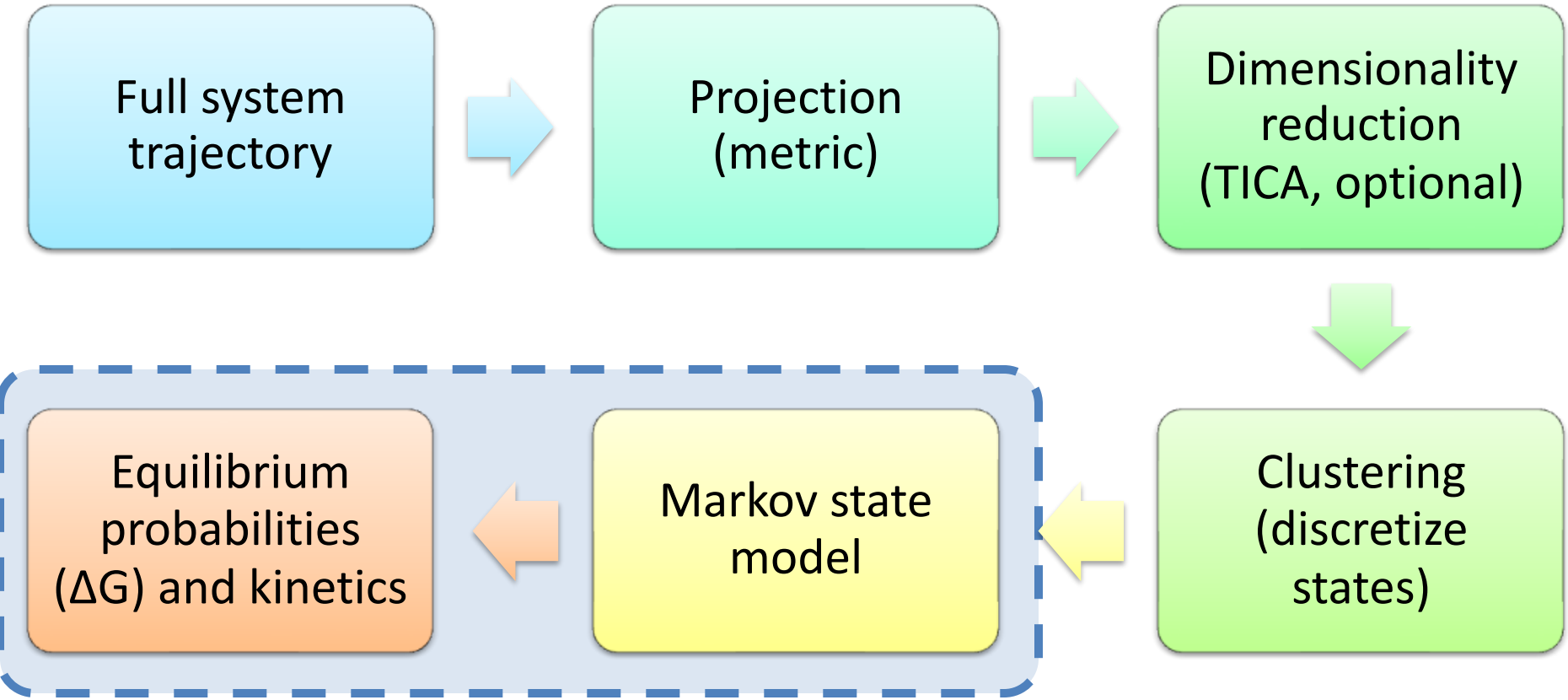
(Actually done mathematically on P_{ij})

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 τ
- It does depend on the chosen states: study varying state discretization

Back to molecules:
>> 1 dimension

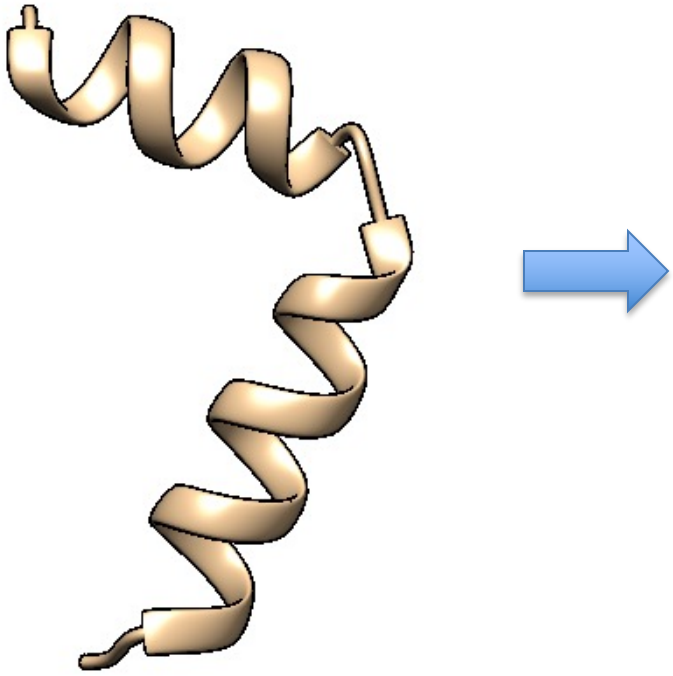
MSM-based analysis overview



MSM proper – this is the part we have seen so far

Metric projection

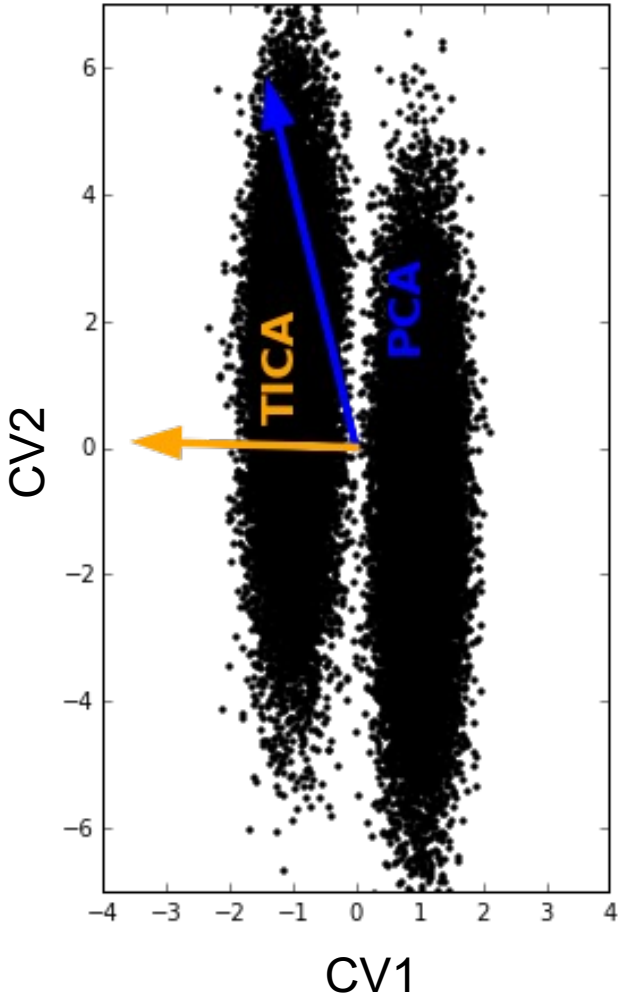
The first step is to project the system state in a lower-dimensional space ("metric"). Many choices are available, e.g.



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

Time-lagged independent component analysis

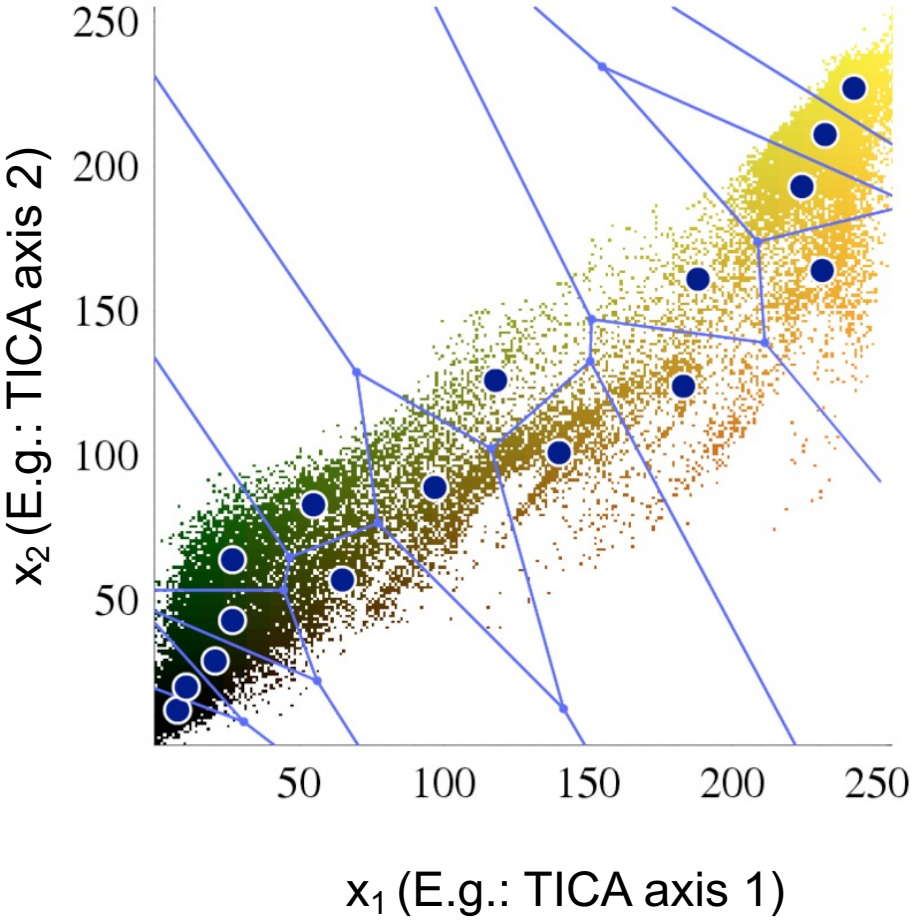
- The feature space dimensionality may still be too high
- Do a further low-dimensional projection on the "slow" degrees of freedom: TICA
- It is based on lagged autocorrelation
- Contrast with PCA, which fits the "most elongated" ellipsoid, **ignoring time**



J. Chem. Phys. 139, 015102 (2013).

Clustering (1st level)

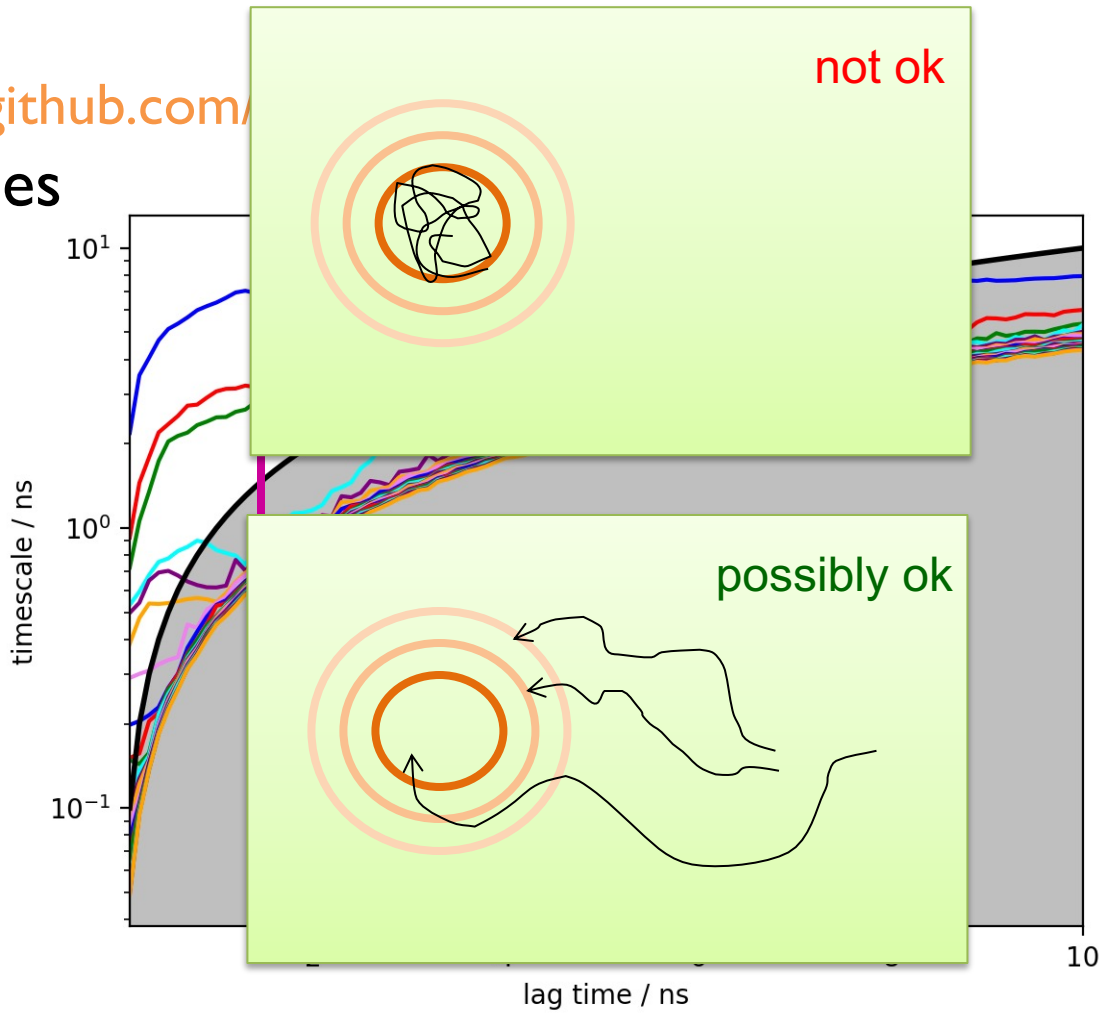
- Now move to a discrete state space
- Reduction from the low-dimensional space is done by *clustering*
  - Usually called "microstates"
- Several algorithms are implemented in MSM packages (e.g. grid; *k*-means; etc. We won't discuss them)



https://en.wikipedia.org/wiki/K-means_clustering

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

- Ace-Ala3-Nme
- Part of examples at: github.com/
- 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[1]

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

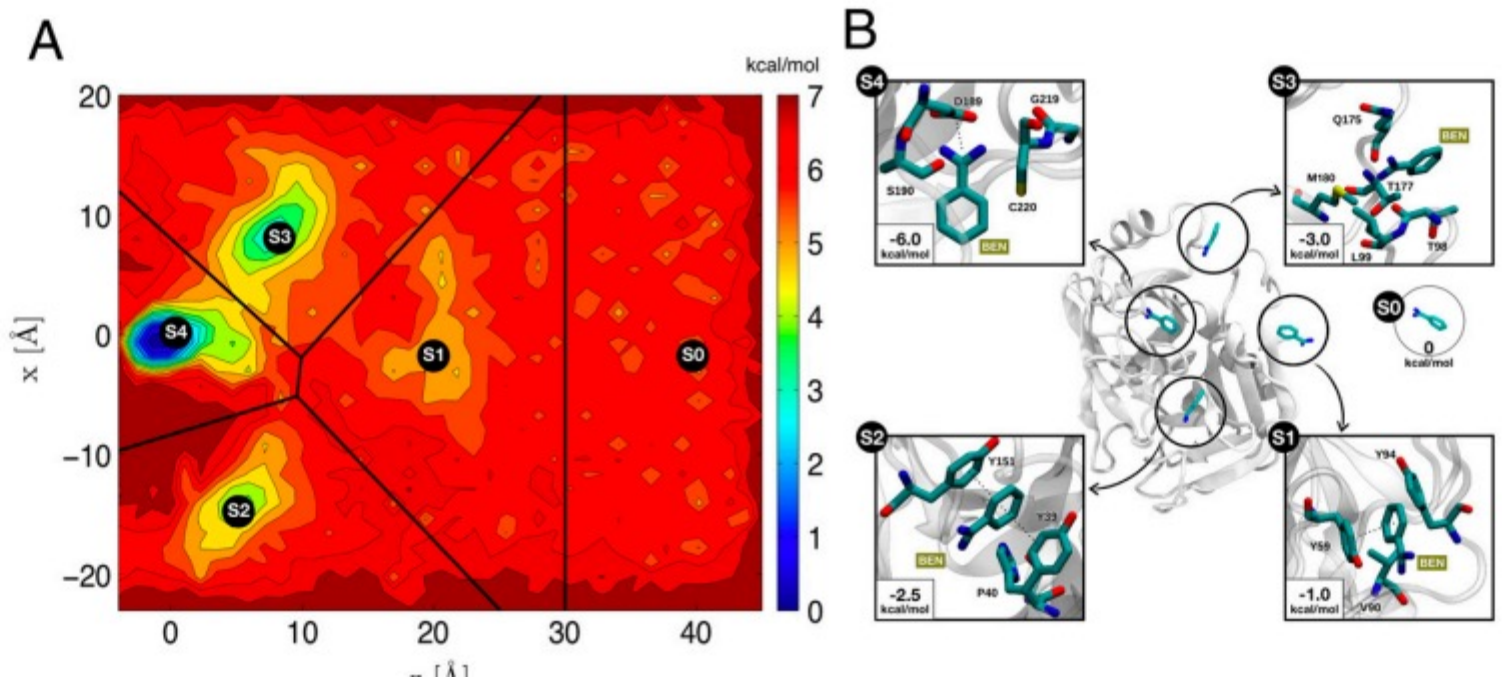
Edited by Arieh Warshel, University of Southern California, Los Angeles, CA, and approved May 11, 2011 (received for review March 4, 2011)

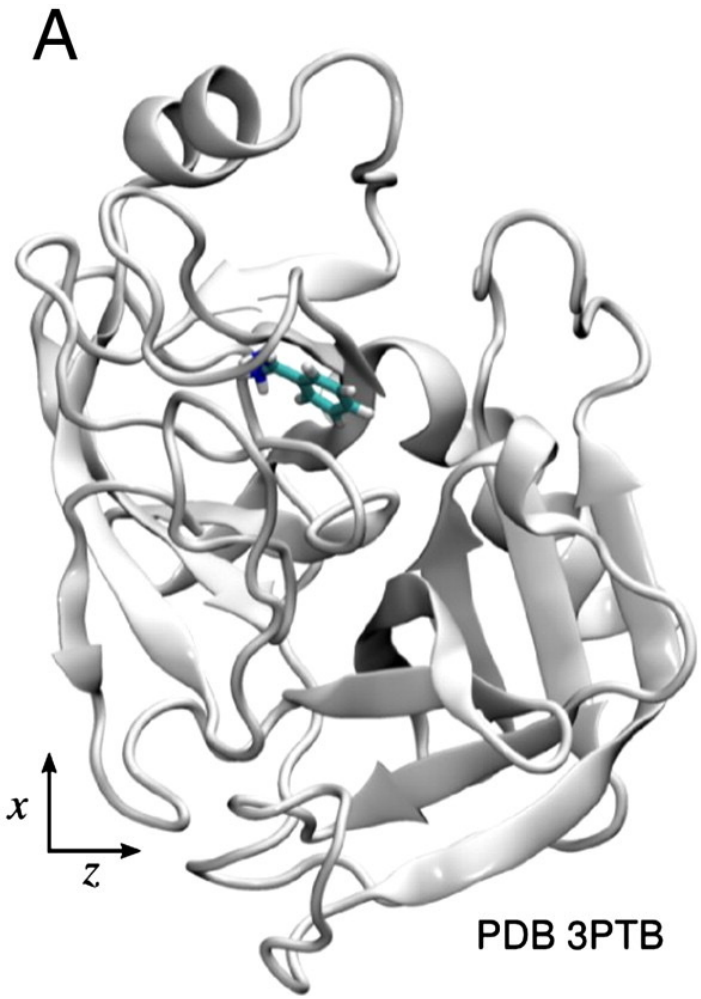
The understanding of protein–ligand binding is of critical importance for biomedical research, yet the process itself has been very difficult to study because of its intrinsically dynamic character. Here, we have been able to quantitatively reconstruct the complete binding process of the enzyme-inhibitor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

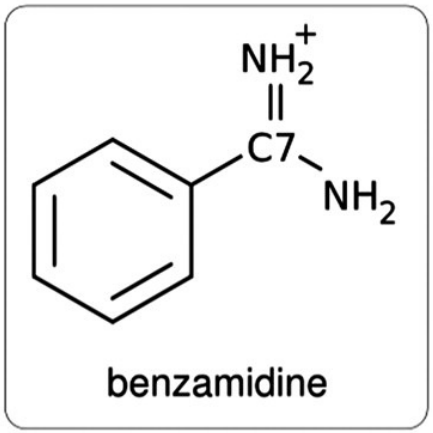
reproduce with atomic resolution the crystallographic mode of binding, but we also provide the kinetically and energetically meaningful transition states of the process.

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

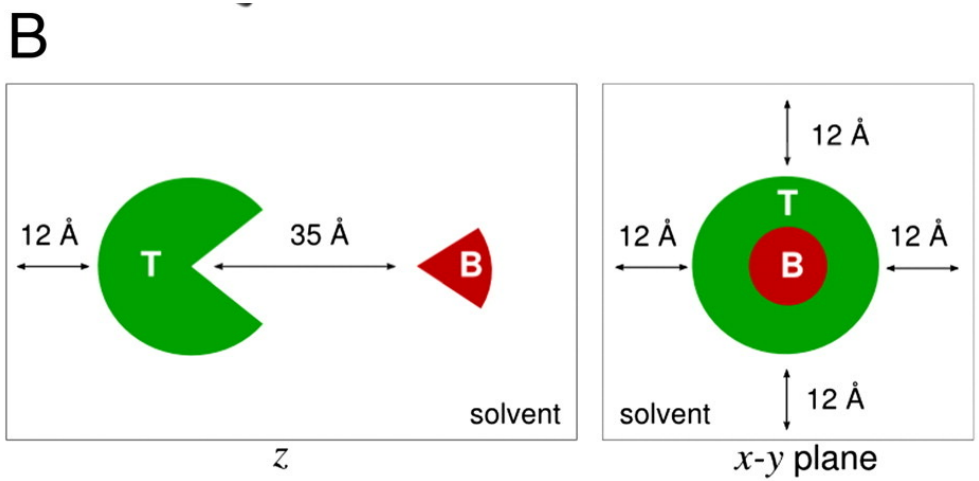
n the pro-



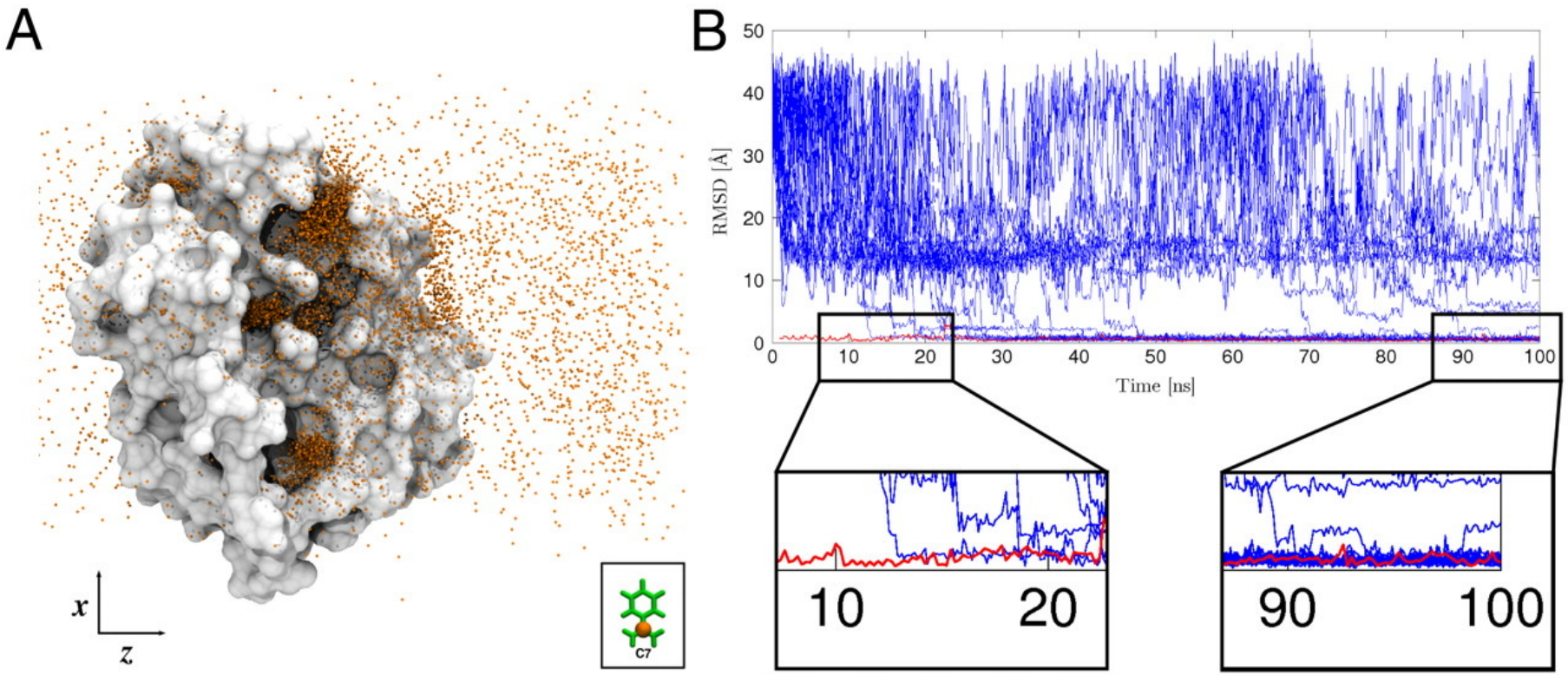


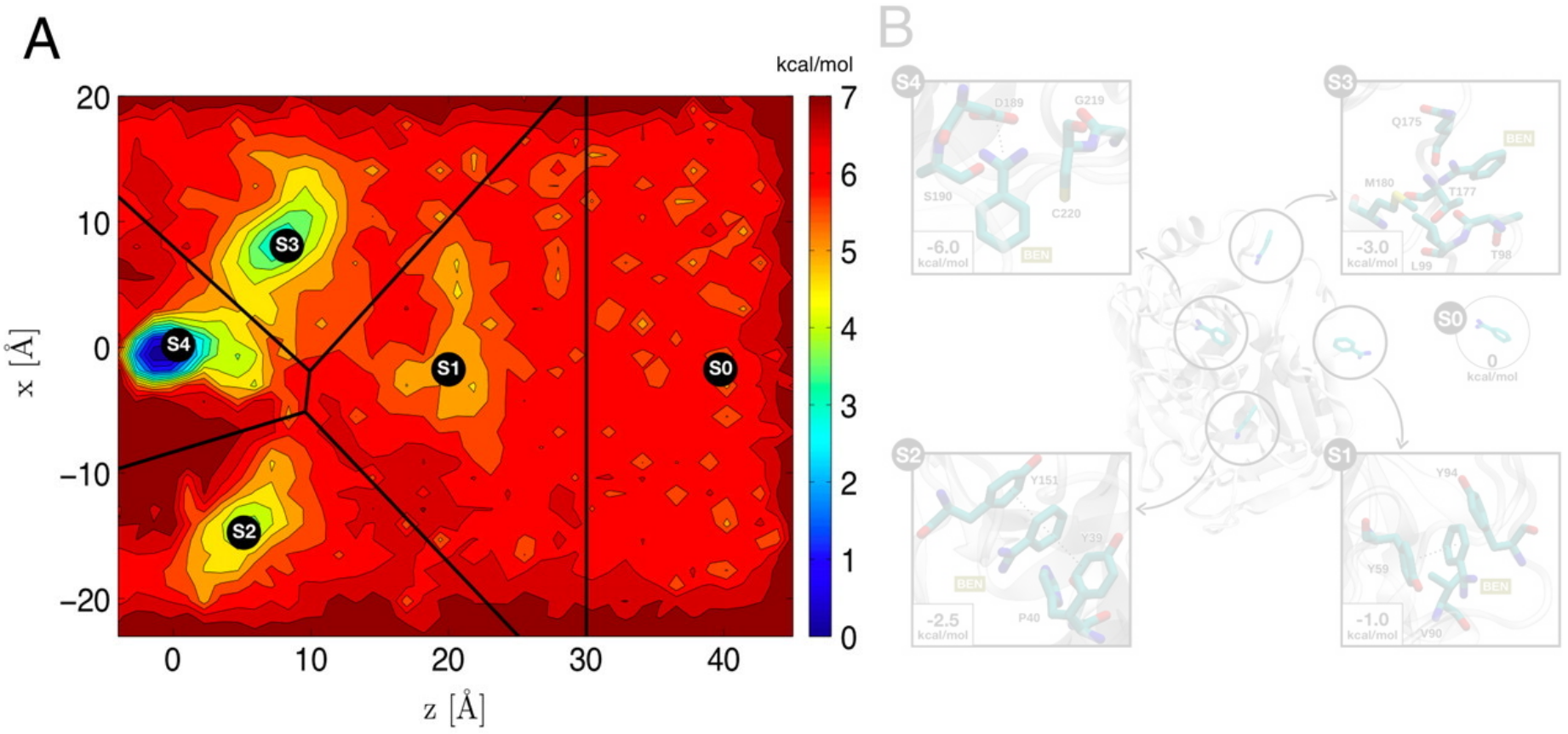


- Rigid ligand: no internal degrees of freedom
- Small ligand: no orientation d.o.f.
- States defined as small cubes according just to the position in 3D space



Ignasi Buch et al. PNAS 2011;108:10184-10189


A
x
z
C7
B
RMSD [Å]
Time [ns]
0
10
20
30
40
50
60
70
80
90
100



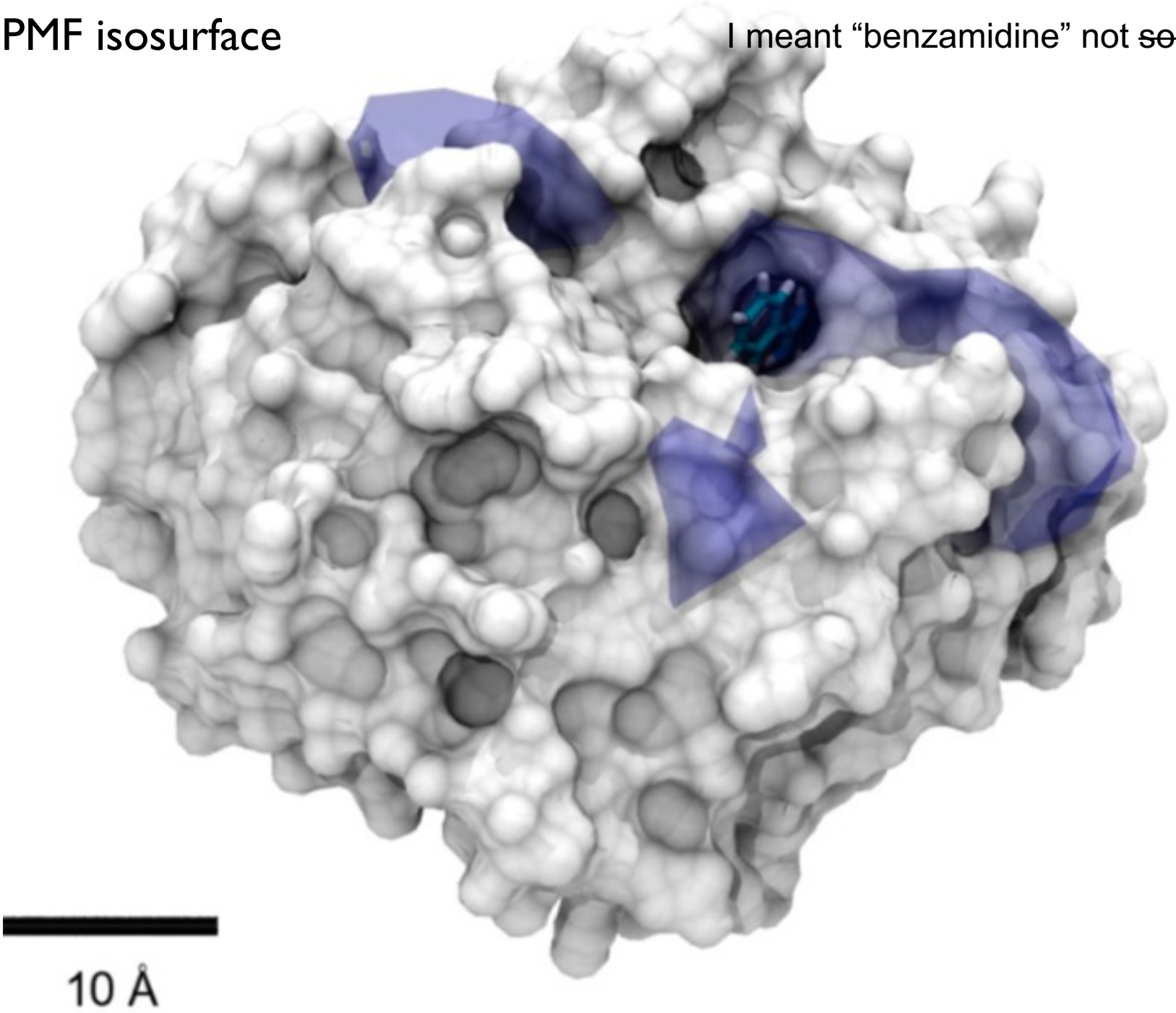
Identification of metastable states. (A) PMF in the xz plane. Five different metastable states can be identified from the different free-energy minima (S0 to S4). The relative free energy between the unbound state S0 and the bound state S4 is -6 kcal/mol. The most probable transition to the bound state 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 π-π stacking interactions with Y151 and Y39 side chains. In S3, a hydrogen bond may be formed between NH2 groups of benzamidine (only heavy atoms shown for clarity) and Q175 side chain, or by a cation-π interaction between the Q175 side chain again, and benzamidine's benzene ring.

Ignasi Buch et al. PNAS 2011;108:10184-10189

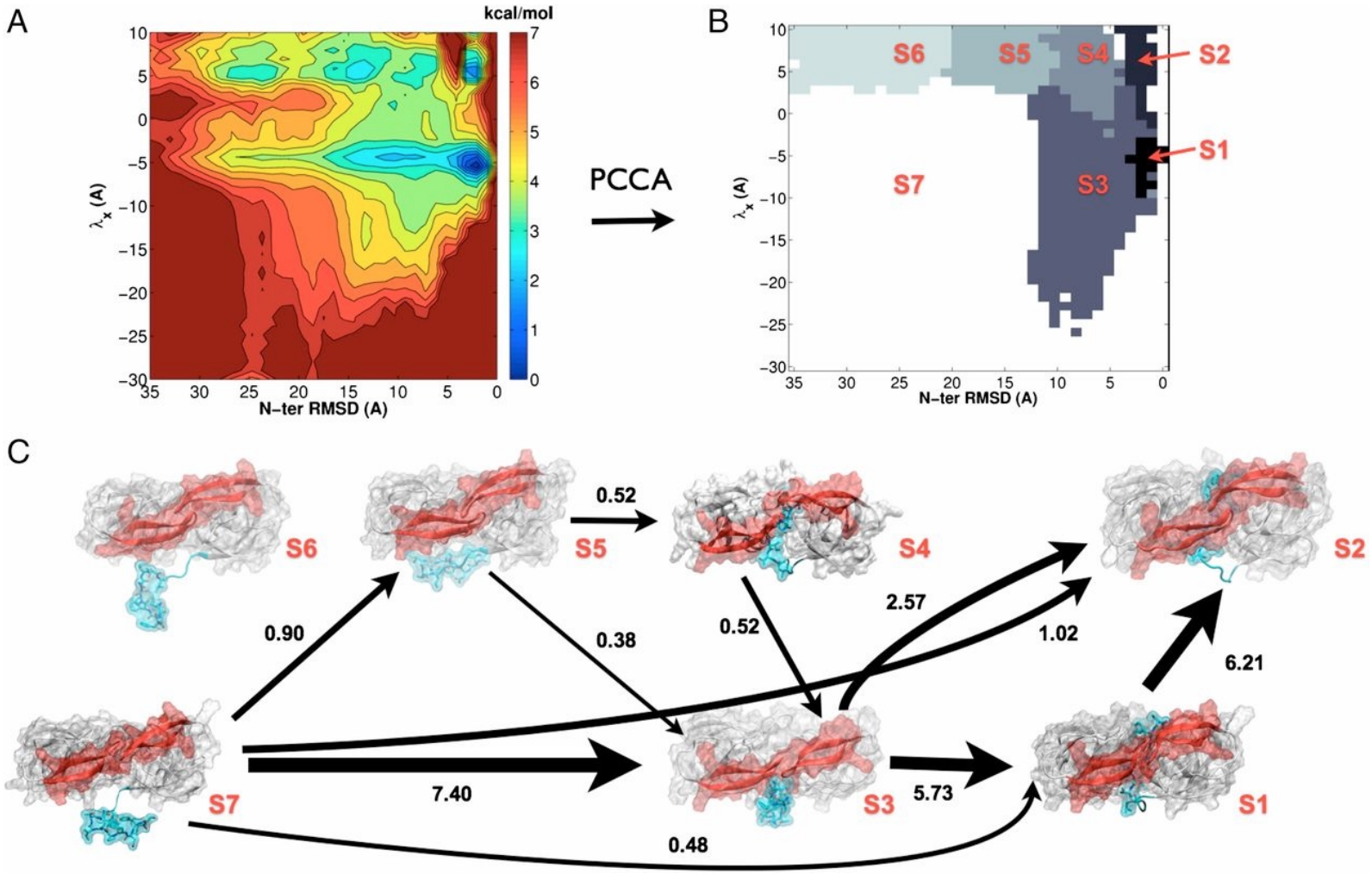
PNAS

3D PMF isosurface

I meant “benzamidine” not ~~solvent~~

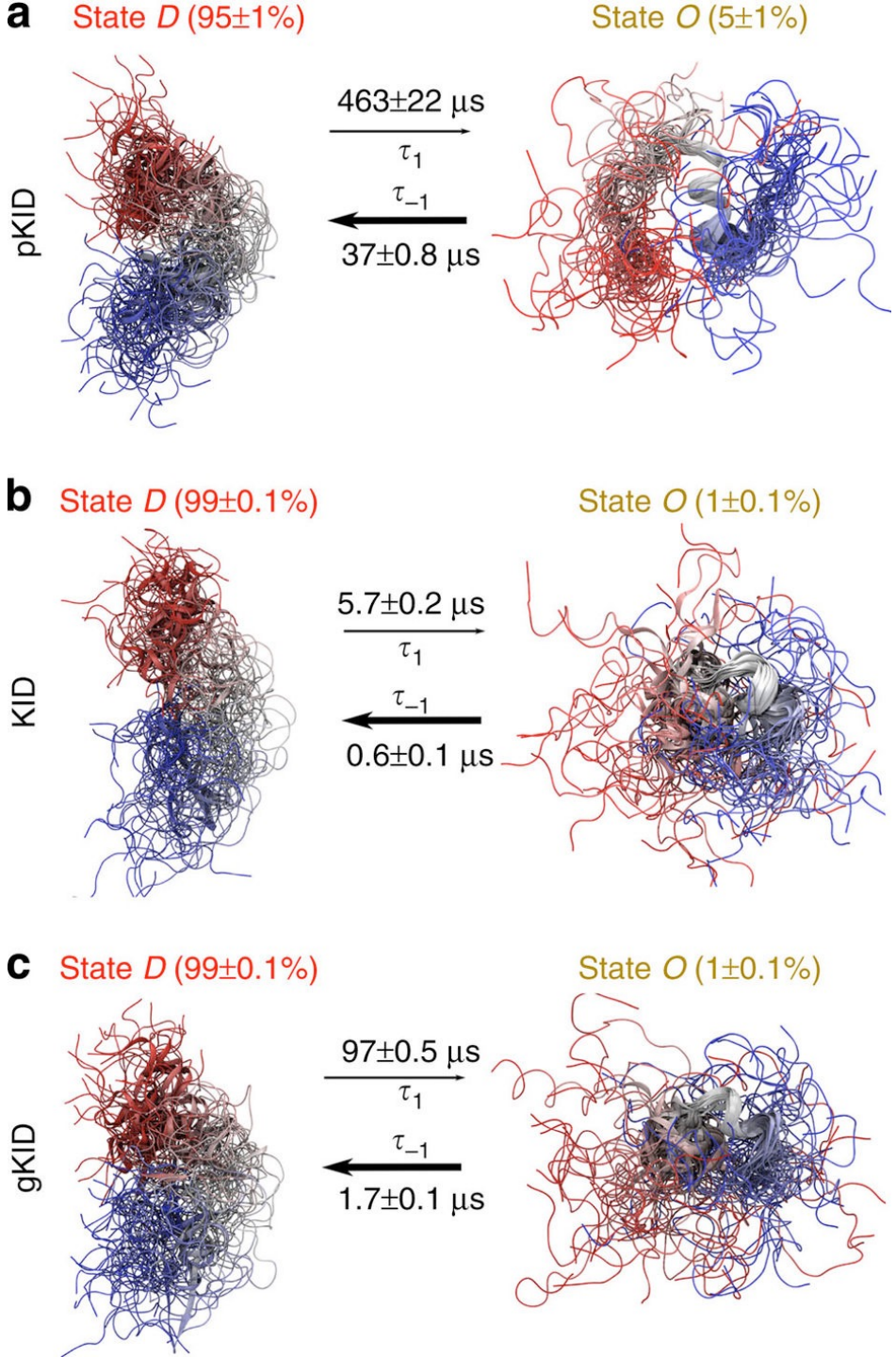


Kinetic characterization of the critical step in HIV-1 protease maturation



www.pnas.org/content/109/50/20449.full

Kinetic modulation of a disordered protein domain by phosphorylation



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

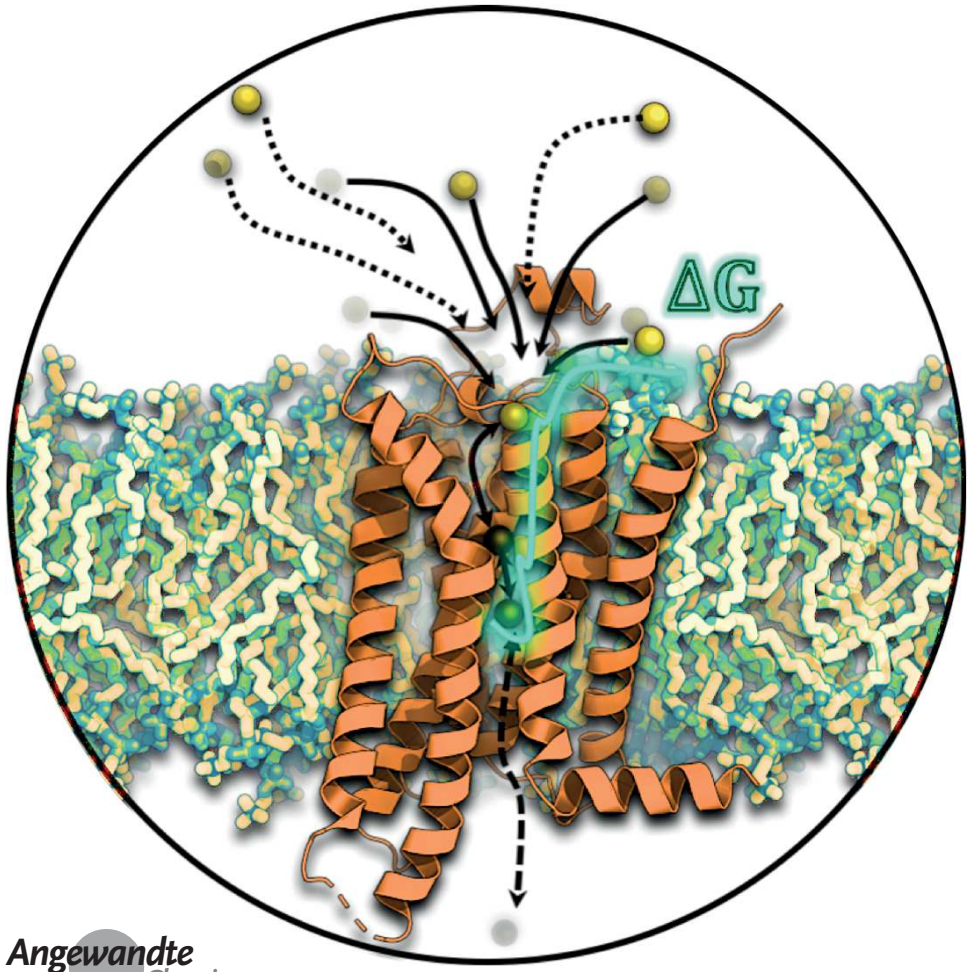
doi:10.1038/ncomms6272

Allosteric Modulators

International Edition: DOI: 10.1002/anie.201708889
German Edition: DOI: 10.1002/ange.201708889

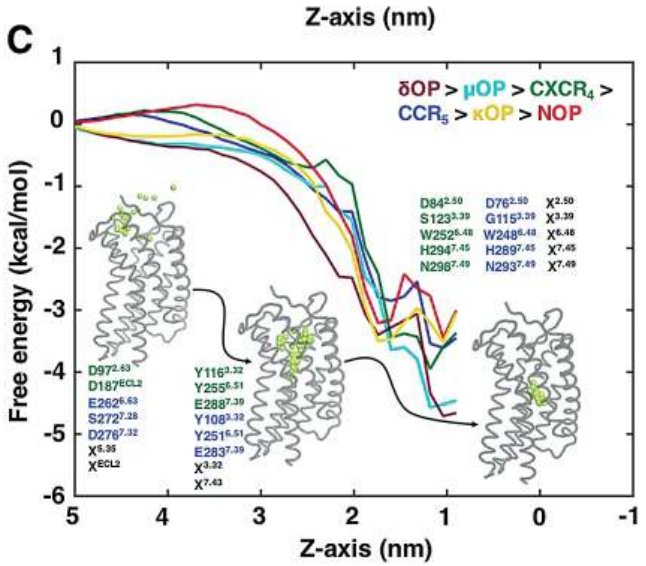
Universality of the Sodium Ion Binding Mechanism in Class A G-Protein-Coupled Receptors

*Balaji Selvam, Zahra Shamsi, and Diwakar Shukla**



plays a key role in identifying drug candidates.[7] Several studies have shown that GPCRs are allosterically modulated by endogenous Na^+ ions.[8–11] Selent and co-workers explored the binding mechanism of Na^+ to the D_2 receptor by molecular dynamics simulations.[12] Na^+ binds in the middle of the transmembrane (TM) region to a conserved $Asp^{2.50}$ residue (Ballesteros–Weinstein numbering),[13] as elucidated by high-resolution crystal structures of the GPCRs.[14–16] More recent MD studies on μ-, κ-, and δ-opioid (OP) receptors also support Na^+ ion binding to $Asp^{2.50}$.[17,18] Na^+ 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.[19–21]

elusive. Herein, we performed hundreds-of-microsecond long simulations of 18 GPCRs to elucidate their Na^+ binding mechanism (see the Supporting Information, Tables S1 and S2). We constructed Markov state models (MSMs) to estimate the free energy profiles for Na^+ binding to each GPCR. Analysis of the Na^+ binding kinetics revealed key residues that act as major barriers for Na^+ entry to the intracellular site. We also predicted the average mean first passage time (MFPT) for Na^+ 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 achieavable) 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 ←
- HTMD – www.htmd.org
  - Also analysis + system build + adaptive …
  - Can aggregate large-scale datasets
- MSMBuilder - msmbuilder.org

End