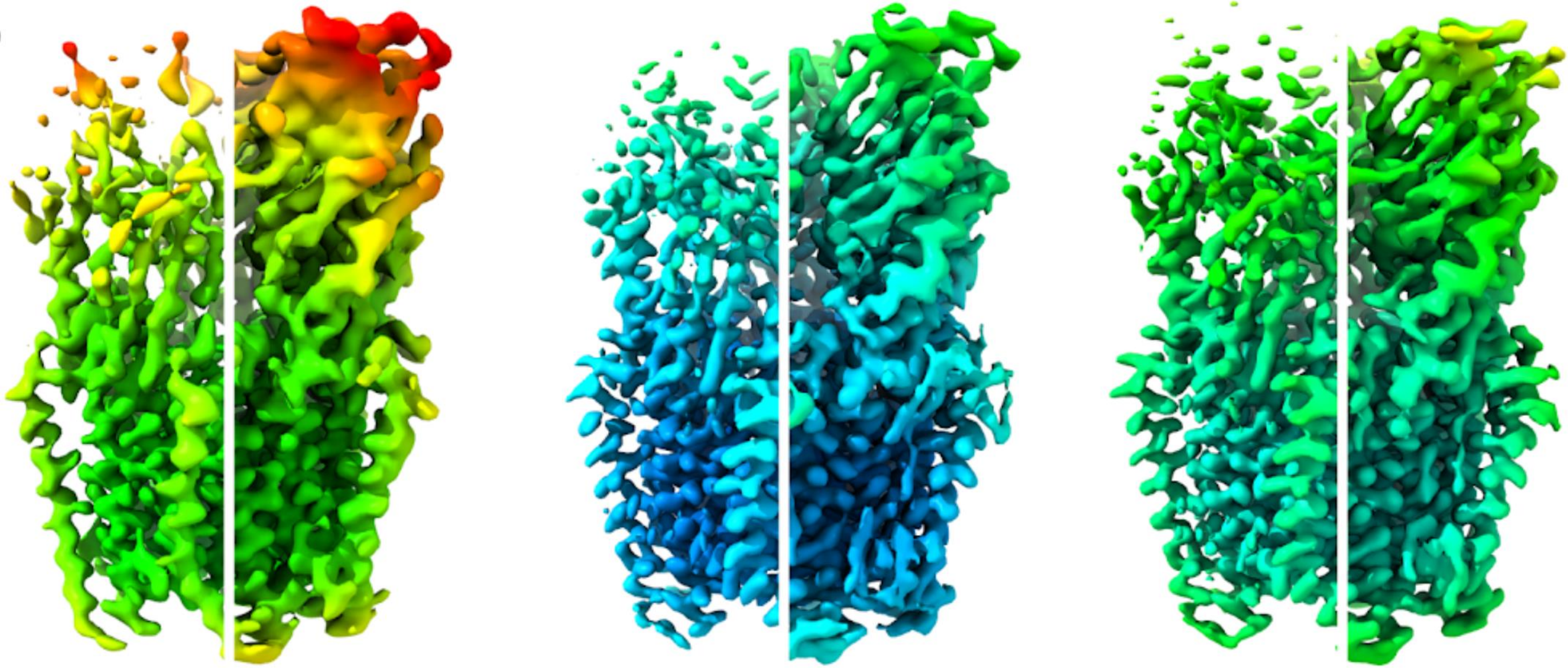


Resolving Transitions with Markov State Models – and 1000x Faster with AI

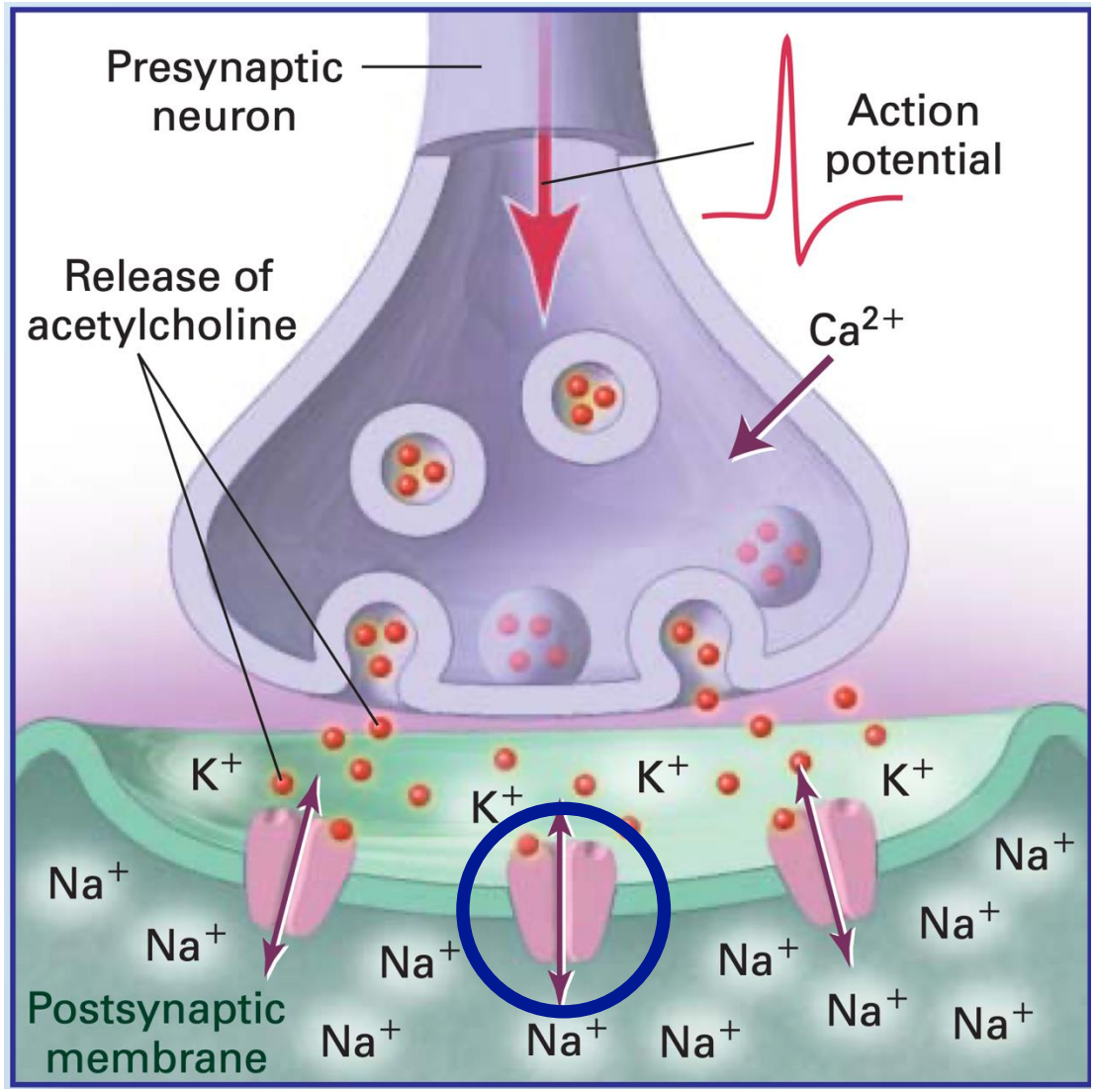
Erik Lindahl

Stockholms universitet · KTH Vetenskap och Konst · SciLifeLab

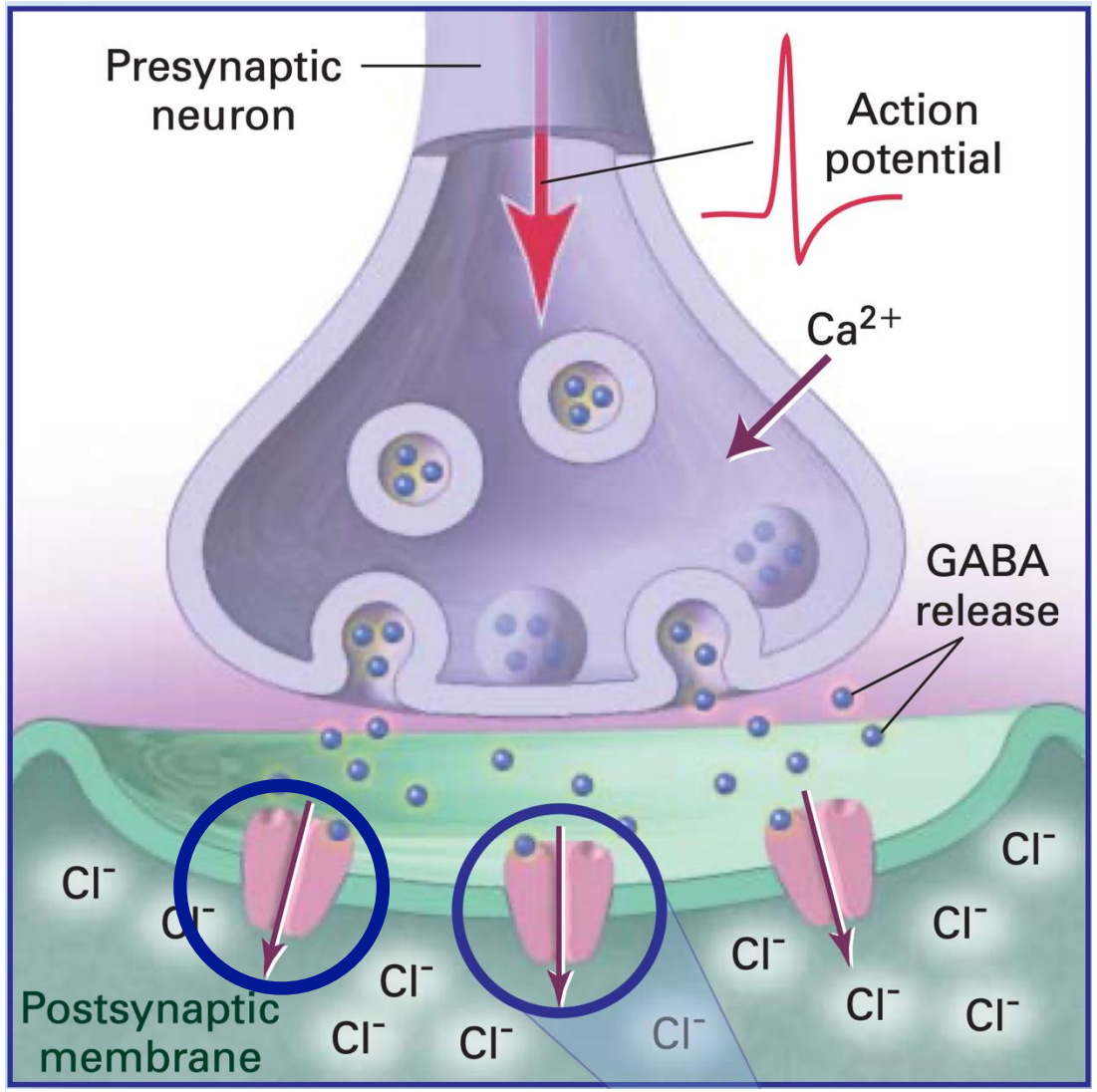
“Suffering so great as I underwent cannot be expressed in words . . . but the blank whirlwind of emotion, the horror of great darkness, and the sense of desertion by God and man, w[illegible] through my mind, and overwhelmed m[illegible] can never forget”


Morton Auditorium, Massachusetts General Hospital
October 16, 1846

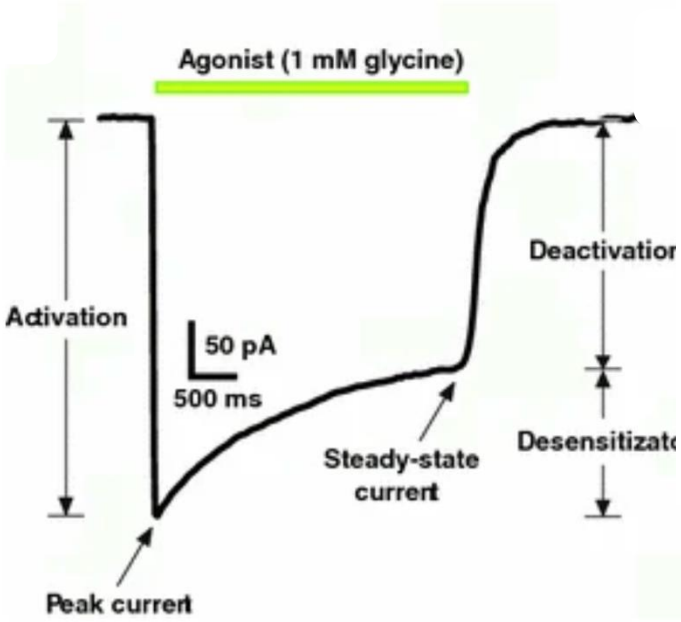
Neuroexcitation

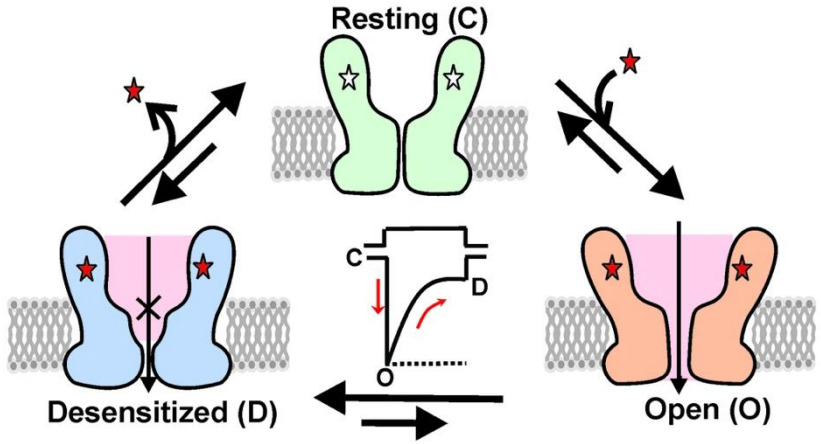


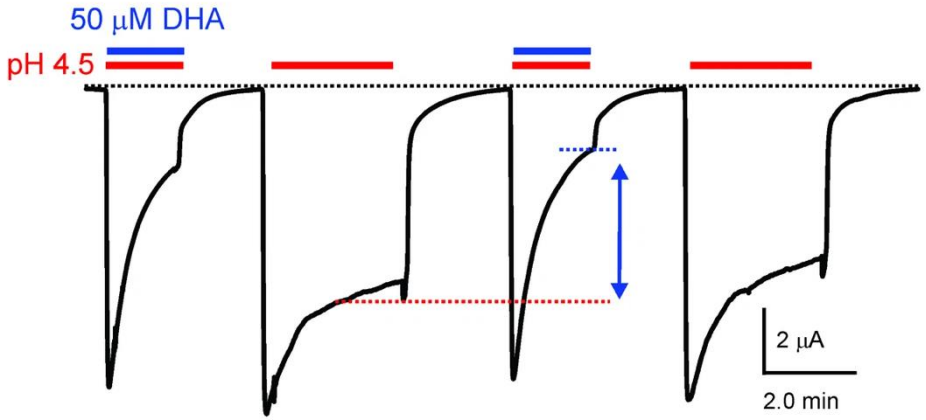
Neuroinhibition

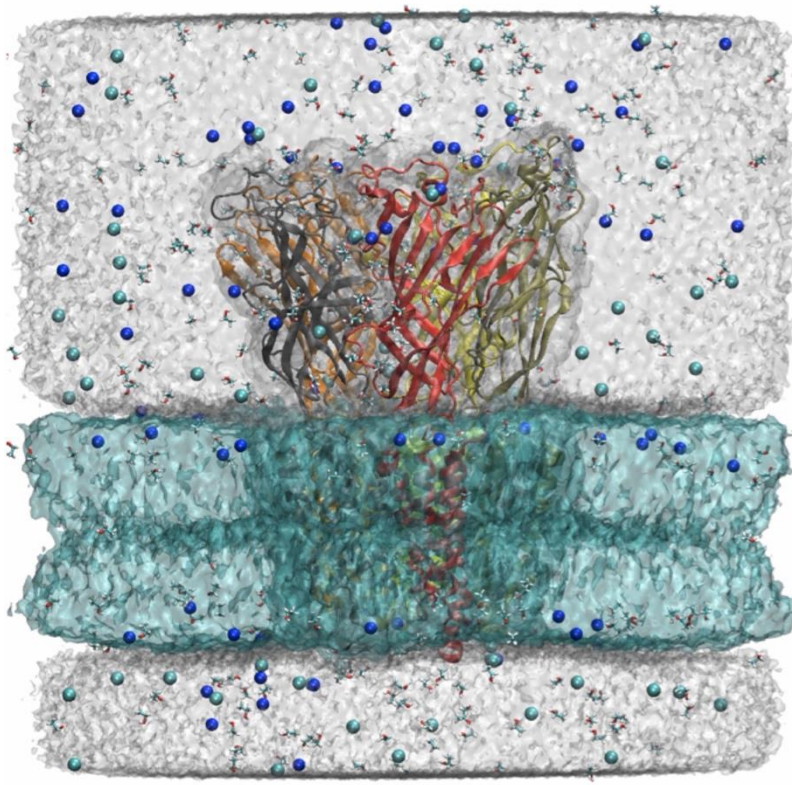


MEASURING GATING

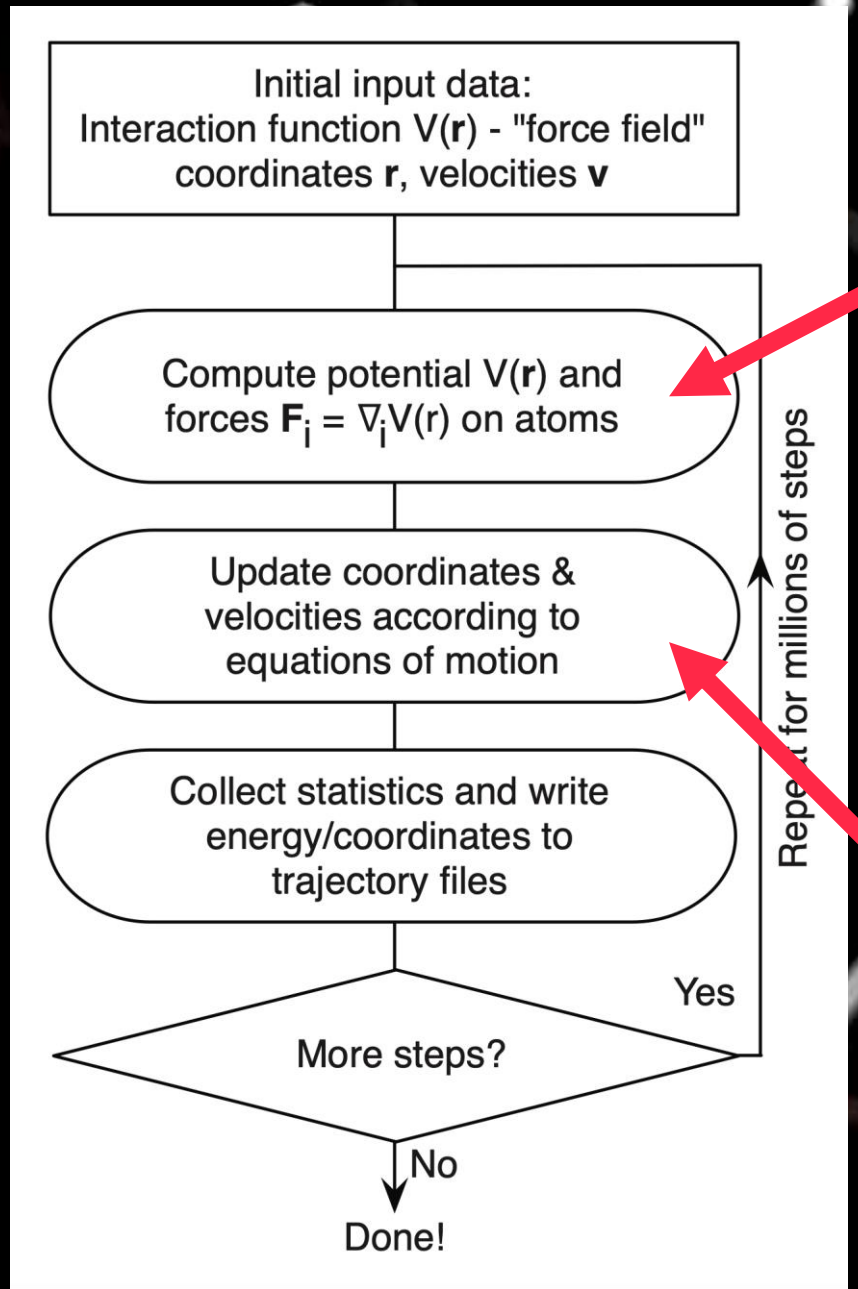








MD is intrinsically parallel, but also sequential



$$V(r) = \sum_{bonds} \frac{1}{2} k_{ij}^{b} (r_{ij} - r_{ij}^{0})^2 + \sum_{angles} \frac{1}{2} k_{ijk}^{\theta} (\theta_{ijk} - \theta_{ijk}^{0})^2 + \sum_{torsions} \left\{ \sum_{n} k_{\theta} [1 + \cos(n\phi - \phi_0)] \right\} + \sum_{impropers} k_{\xi} (\xi_{ijkl} - \xi_{ijkl}^{0}) + \sum_{i,j} \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} + \sum_{i,j} \left[\frac{C_{12}}{r_{ij}^{12}} - \frac{C_6}{r_{ij}^{6}} \right]$$

Costly, because these terms involve all pairs

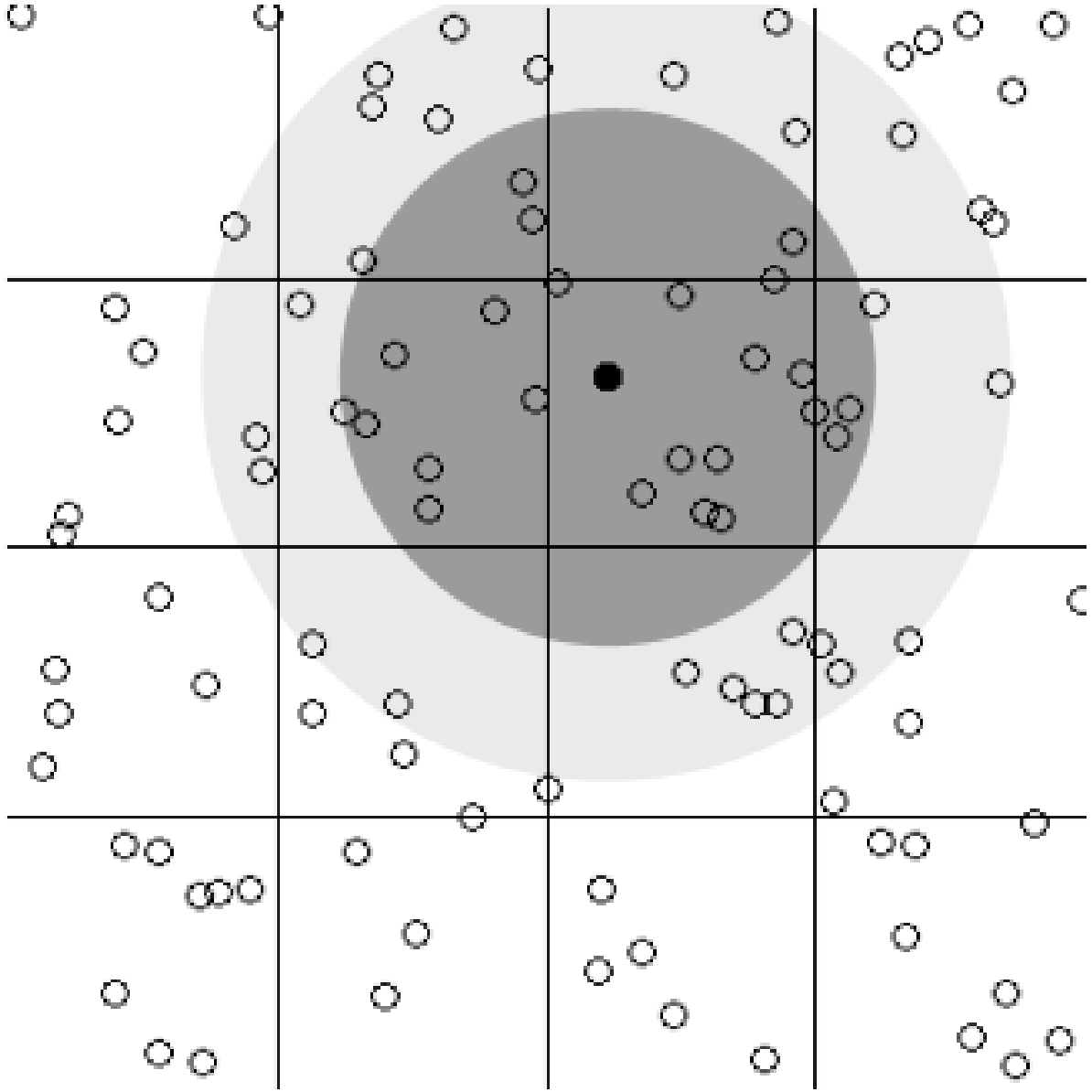
$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i \quad i = 1..N$$

$$F_i = -\frac{\partial V(r)}{\partial r_i}$$

With Δt ~ 1fs and μs to s timescales of interest, we need 10^9-10^{15} steps.

The floating-point performance generation of HPC scientists

```
for(k=nj0; (k<nj1); k++)
{

    /* Get j neighbor index, and coordinate index */
    jnr             = jjnr[k];
    j3              = 3*jnr;

    /* load j atom coordinates */
    jx1             = pos[j3+0];
    jy1             = pos[j3+1];
    jz1             = pos[j3+2];

    /* Calculate distance */
    dx11            = ix1 - jx1;
    dy11            = iy1 - jy1;
    dz11            = iz1 - jz1;
    rsq11           = dx11*dx11+dy11*dy11+dz11*dz11;

    /* Calculate 1/r and 1/r2 */
    rinv11          = 1.0/sqrt(rsq11);

    /* Load parameters for j atom */
    qq              = iq*charge[jnr];
    tj              = nti+2*type[jnr];
    c6              = vdwparam[tj];
    c12             = vdwparam[tj+1];
    rinvsq          = rinv11*rinv11;

    /* Coulomb interaction */
    vcoul           = qq*rinv11;
    vctot           = vctot+vcoul;

    /* Lennard-Jones interaction */
    rinvsix         = rinvsq*rinvsq*rinvsq;
    Vvdw6           = c6*rinvsix;
    Vvdw12          = c12*rinvsix*rinvsix;
    Vvdwtot         = Vvdwtot+Vvdw12-Vvdw6;
    fscal           = (vcoul+12.0*Vvdw12-6.0*Vvdw6)*rinvsq;

    /* Calculate temporary vectorial force */
    tx              = fscal*dx11;
    ty              = fscal*dy11;
    tz              = fscal*dz11;

    /* Increment i atom force */
    fix1            = fix1 + tx;
    fiy1            = fiy1 + ty;
    fiz1            = fiz1 + tz;

    /* Decrement j atom force */
    faction[j3+0]   = faction[j3+0] - tx;
    faction[j3+1]   = faction[j3+1] - ty;
    faction[j3+2]   = faction[j3+2] - tz;

    /* Inner loop uses 38 flops/iteration */
}
```




ATI
RADEON

STREAM COMPUTING ON GRAPHICS HARDWARE

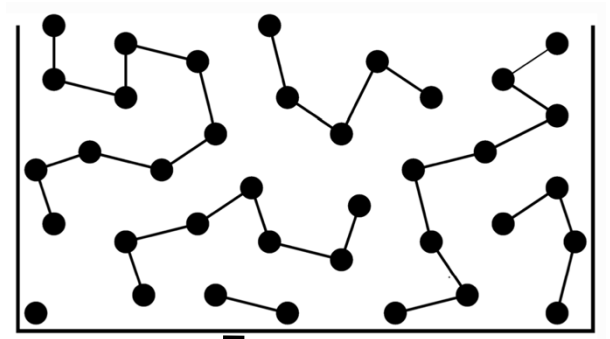
A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Ian Buck

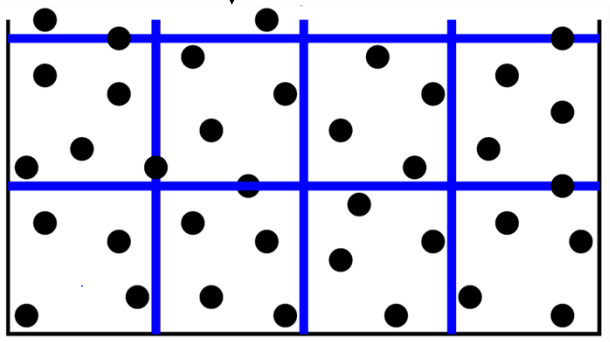
September 2006

Ian Buck, vice president and general manager of accelerated computing at NVIDIA, delivers a special address at SC25

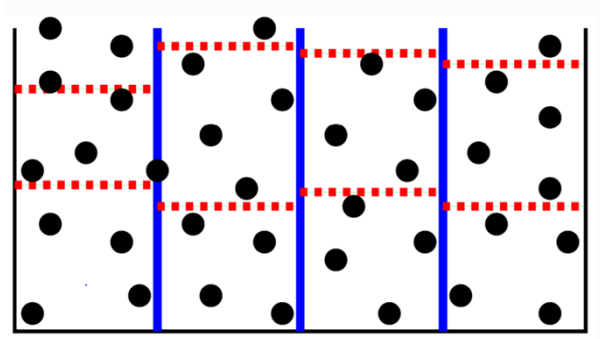
Don't optimize your **code** for GPUs - optimize the ***algorithm***



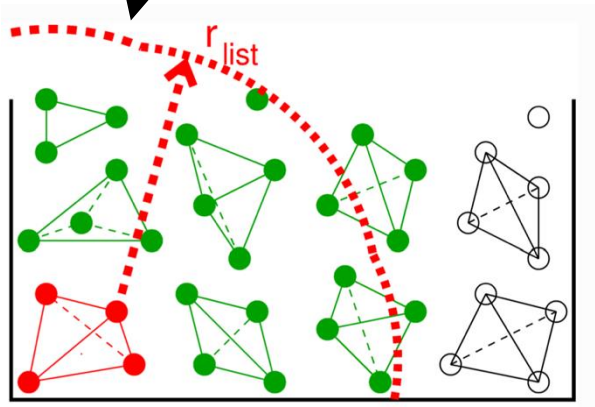
x,y,z
gridding



x,y grid
z sort
z bin

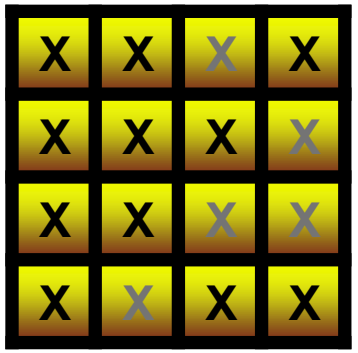


Cluster pairlist



i=3:	5	6	9	12	15	17	18	25	32	…	
i=4:	7	8	9	11	12	15	17	25	32	43	54	…
…	8	9	10	11	12	13	19	20	…		

Organize
as tiles with
all-vs-all
interactions:



Tile interaction algorithms:
Load N atoms, compute N^2 forces

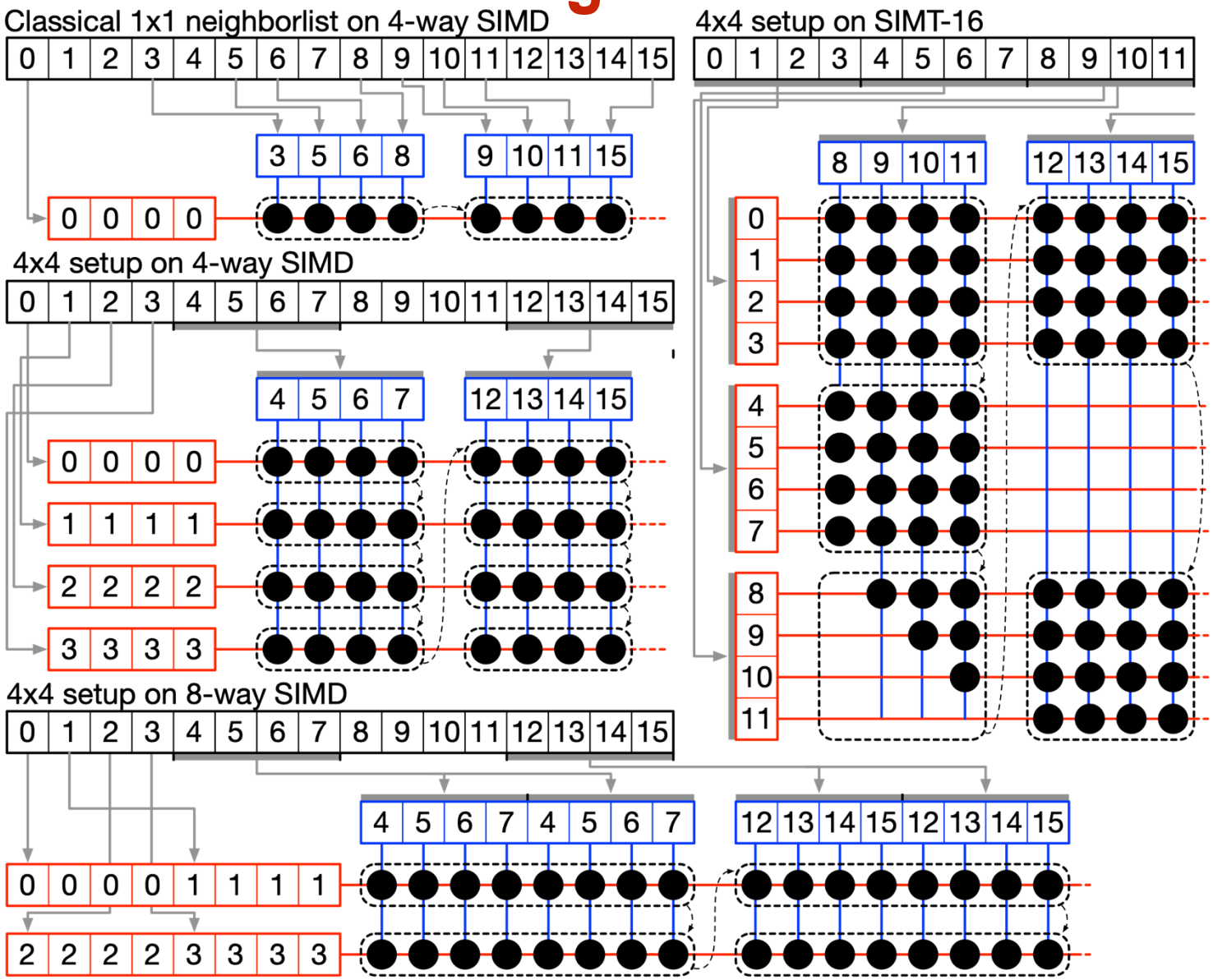
The Link-cell algorithm: Load 1 atom, calculate 1 interaction
Verlet, Phys Rev 159, 98-103 (1967)

TO FIRST APPROXIMATION, GPUS PROVIDE AN INFINITE AMOUNT OF FLOPS

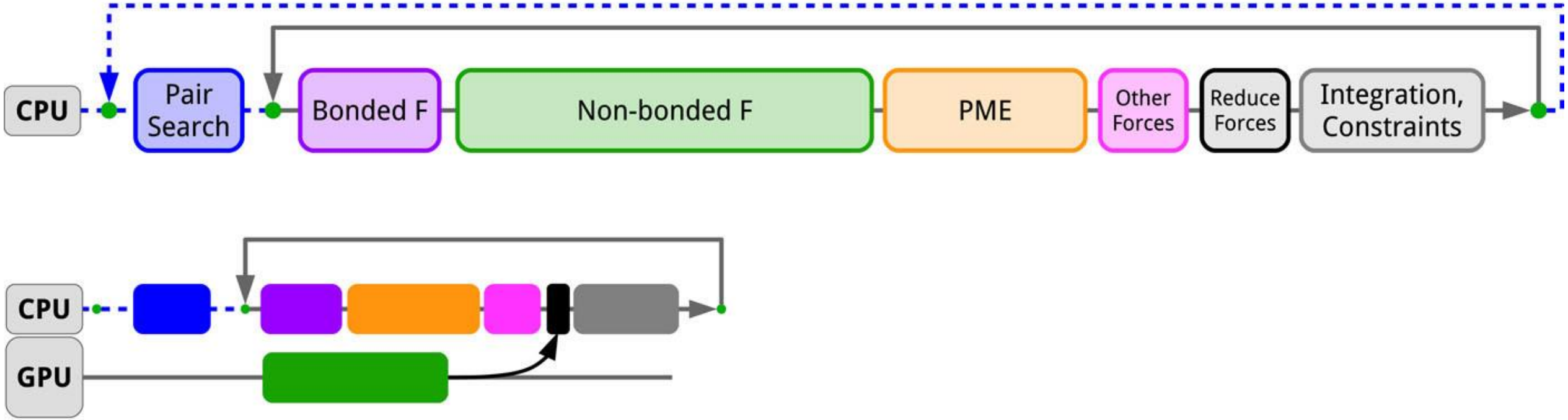
OUR FIRST JOB IS TO HANDLE MEMORY BANDWIDTH

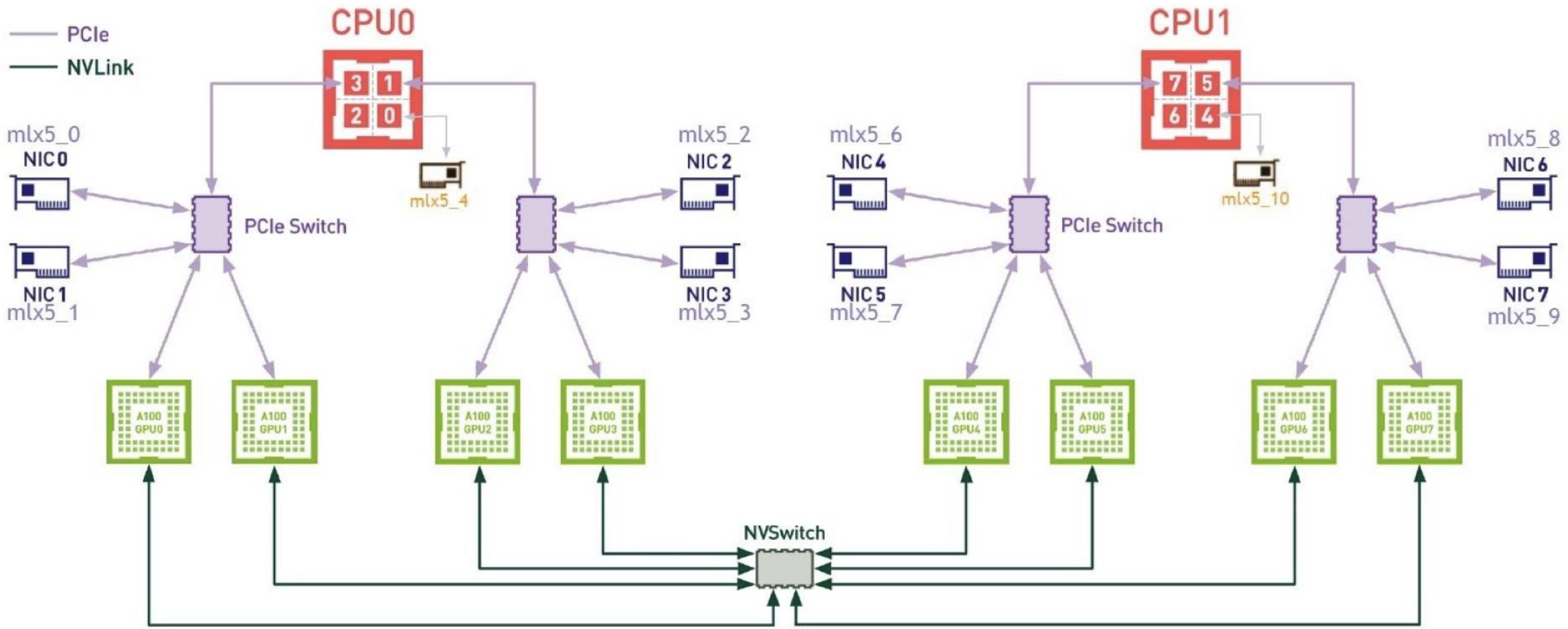
We change the algorithms to improve compute/memory ratio

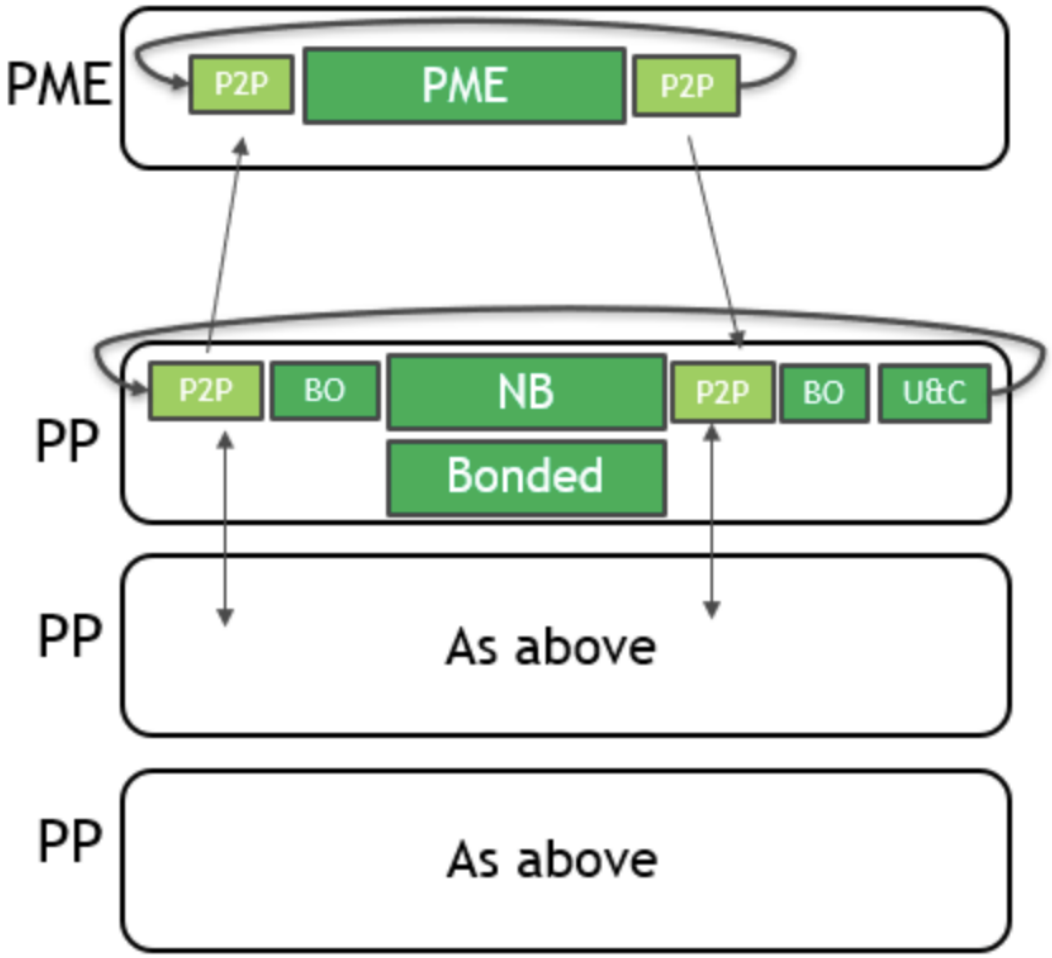
This is great for modern CPUs too



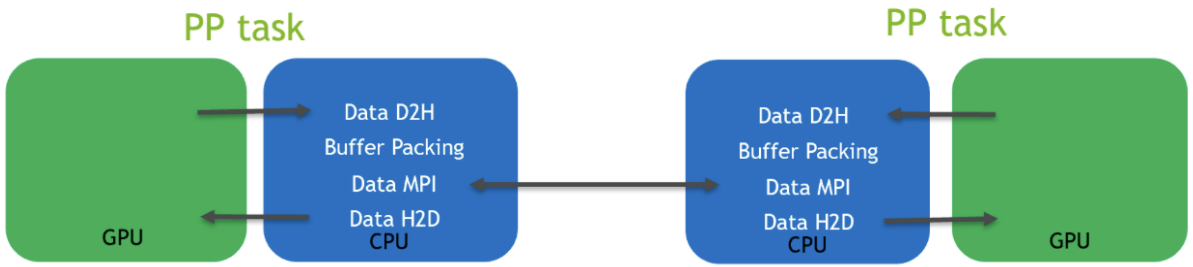
- CUDA
- OpenCL
- Intel MIC
- x86 SSE2
- x86 SSE4.1
- x86 AVX
- x86 AVX-128-FMA
- x86 AVX2
- x86 AVX2_128
- x86 AVX-512F
- x86 AVX-512ER
- Arm Neon
- Arm64 Asimd
- IBM QPX
- IBM VMX
- IBM VSX
- Fujitsu HPC-ACE
- ARM SVE & SVE2
- SYCL
- HIP


CPU
Pair Search
Bonded F
Non-bonded F
PME
Other Forces
Reduce Forces
Integration, Constraints
CPU
GPU

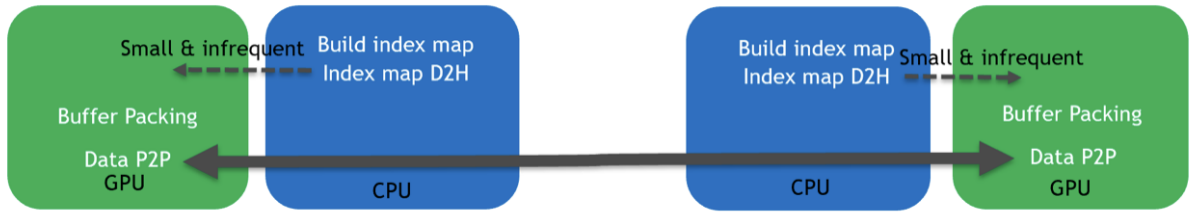

PCIe
NVLink
CPU0
CPU1
3 1
2 0
7 5
6 4
mlx5_0
NIC 0
mlx5_1
NIC 1
PCIe Switch
mlx5_4
mlx5_2
NIC 2
NIC 3
mlx5_3
mlx5_6
NIC 4
NIC 5
mlx5_7
PCIe Switch
mlx5_10
mlx5_8
NIC 6
NIC 7
mlx5_9
A100 GPU0
A100 GPU1
A100 GPU2
A100 GPU3
A100 GPU4
A100 GPU5
A100 GPU6
A100 GPU7
NVSwitch



MPI staged through CPUs

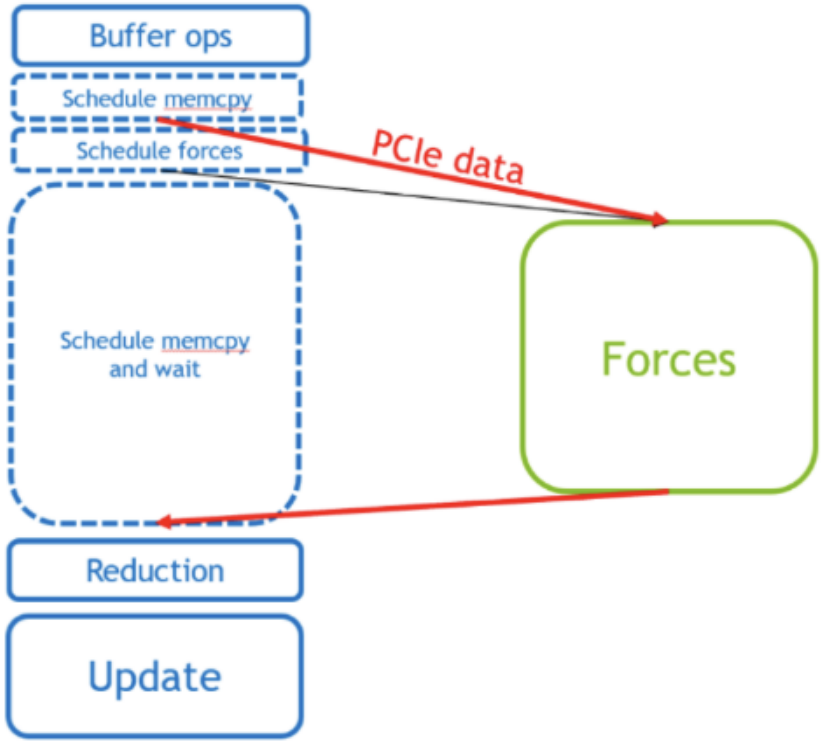


GPU-direct halo communication

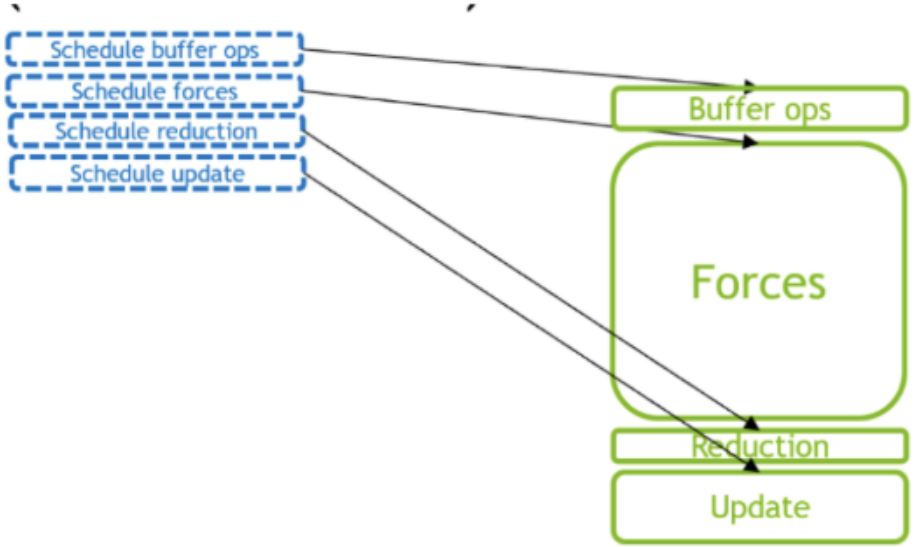


From CPU to offloading to GPU-resident CPU-scheduled kernels

CPU-only (with multi-core threading)

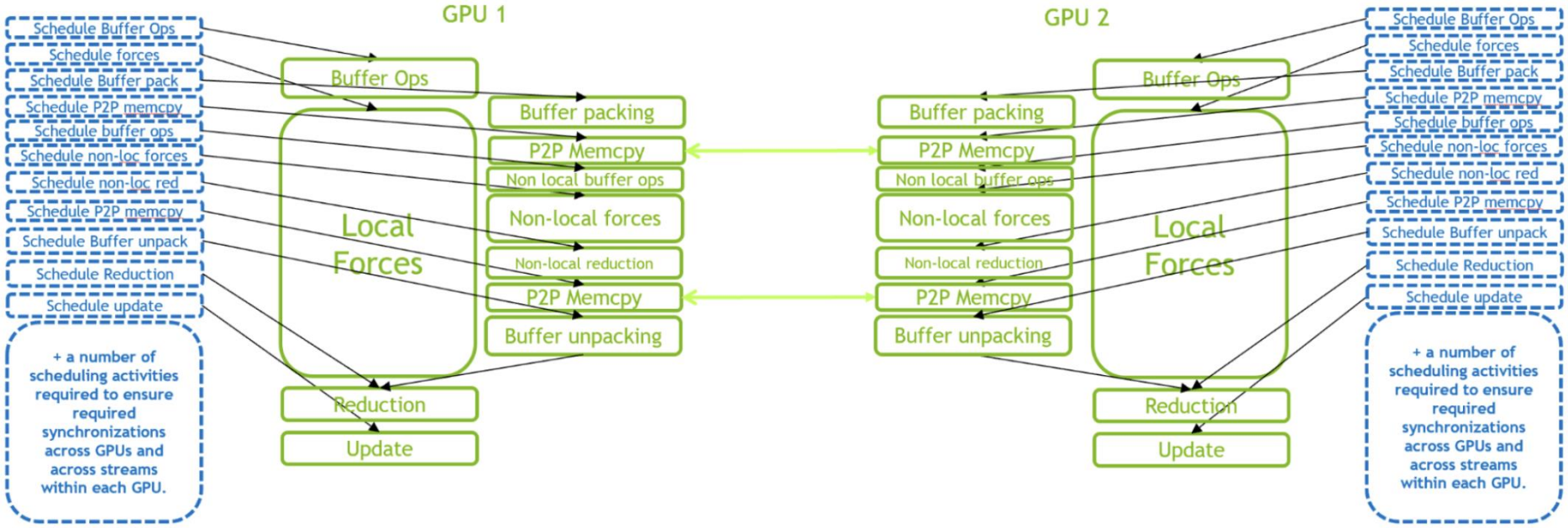


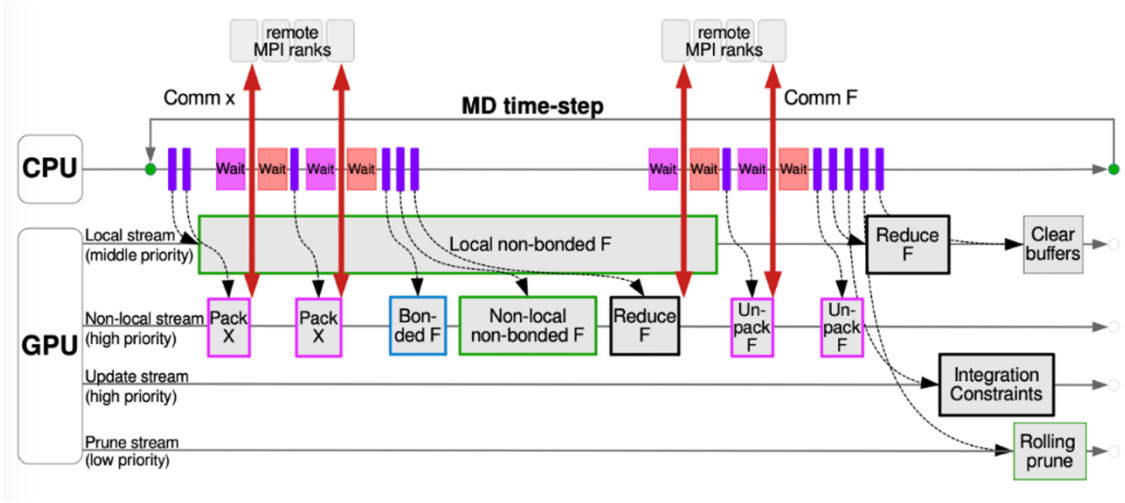
Pre-v2020: GPU acceleration of forces only.

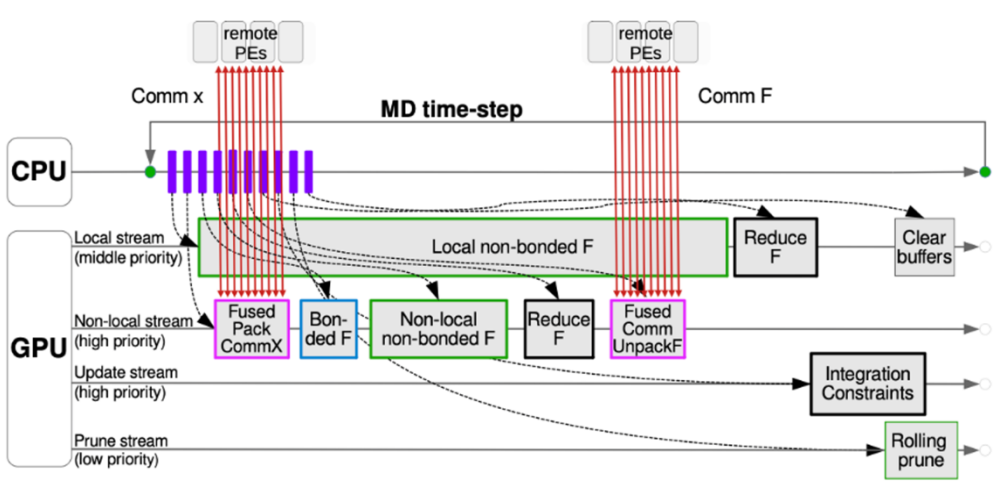


GPU-resident step with asynchronous task scheduling (from v2020)

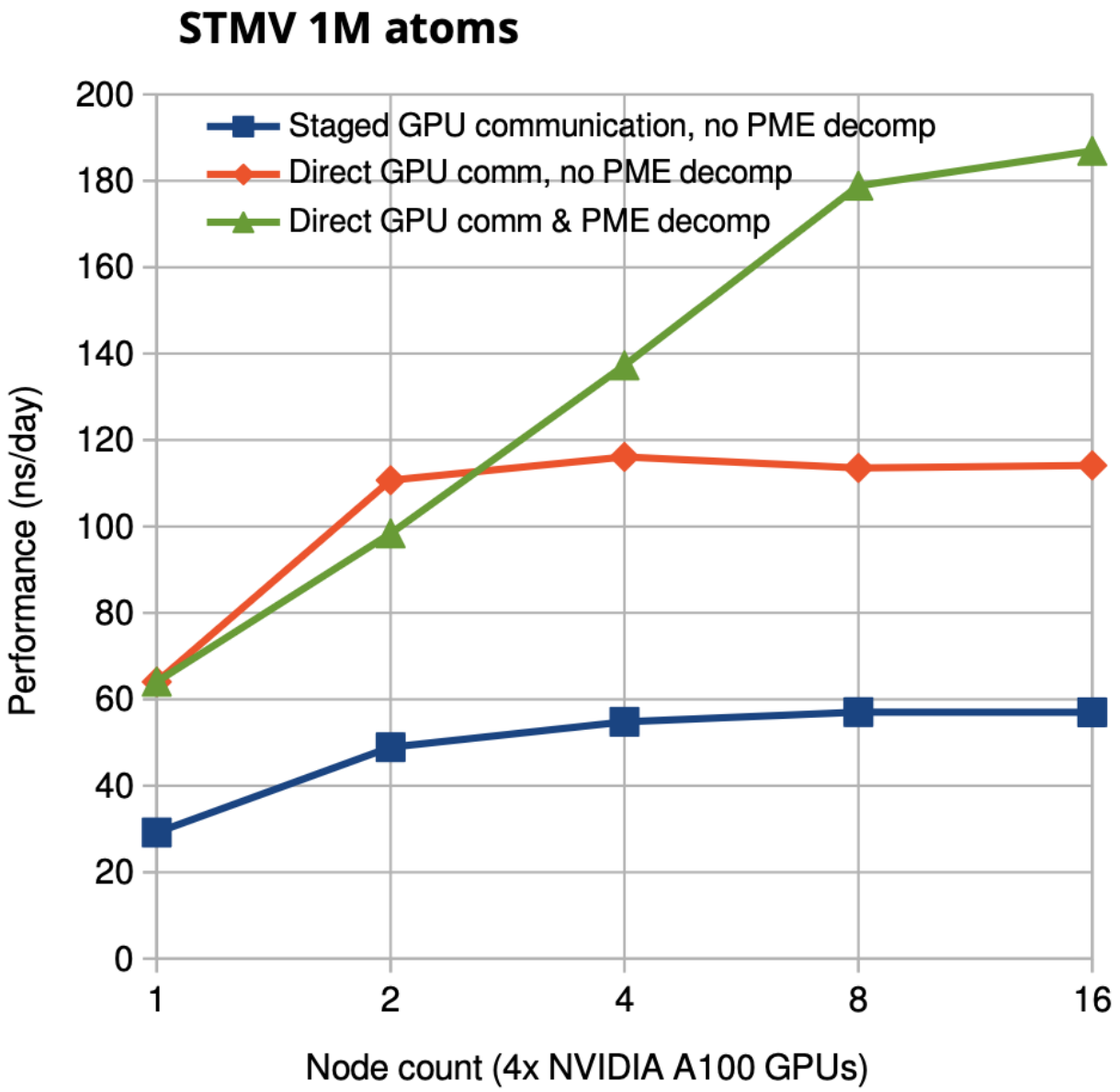
Multiple GPUs & Multiple nodes

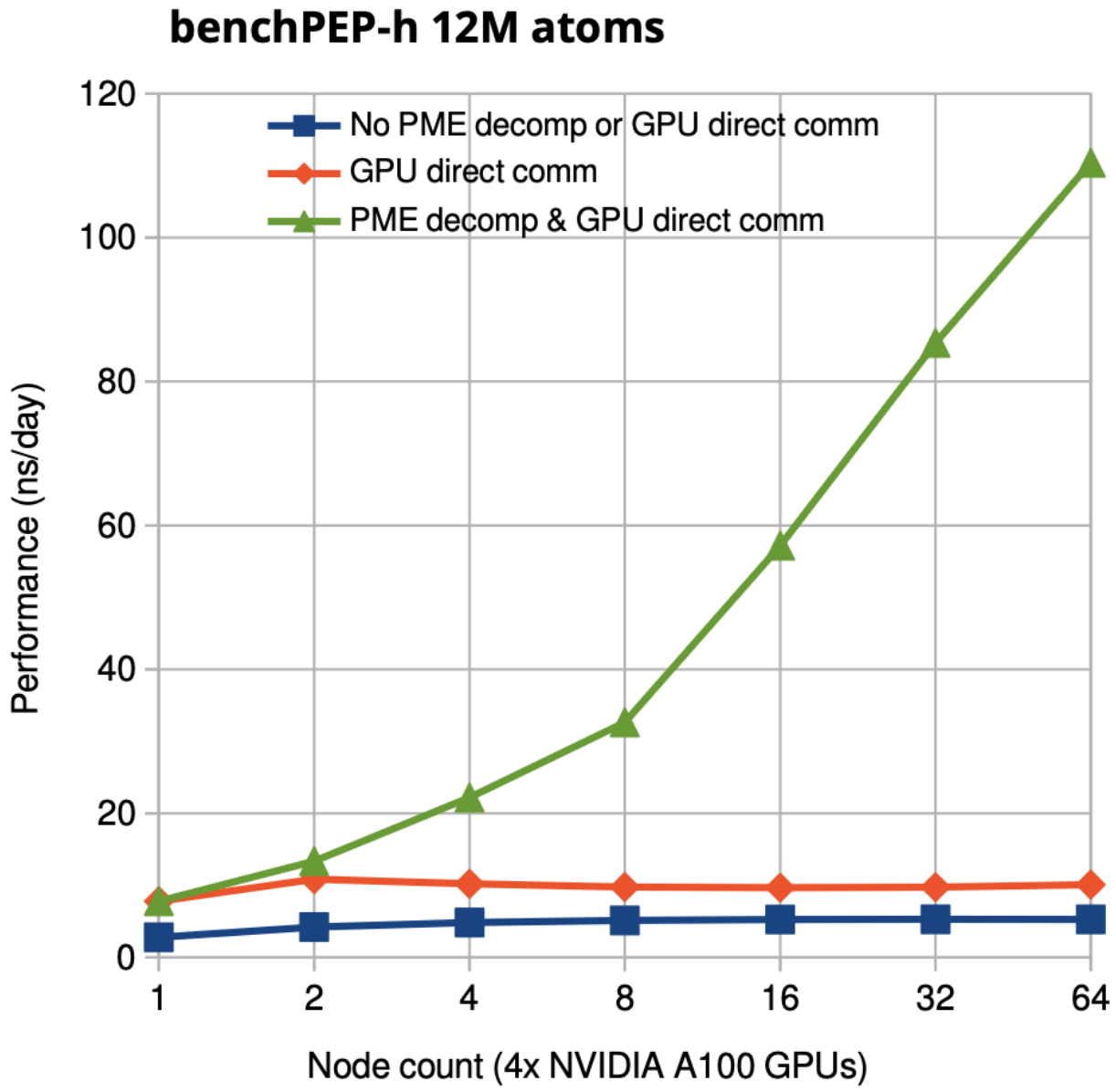




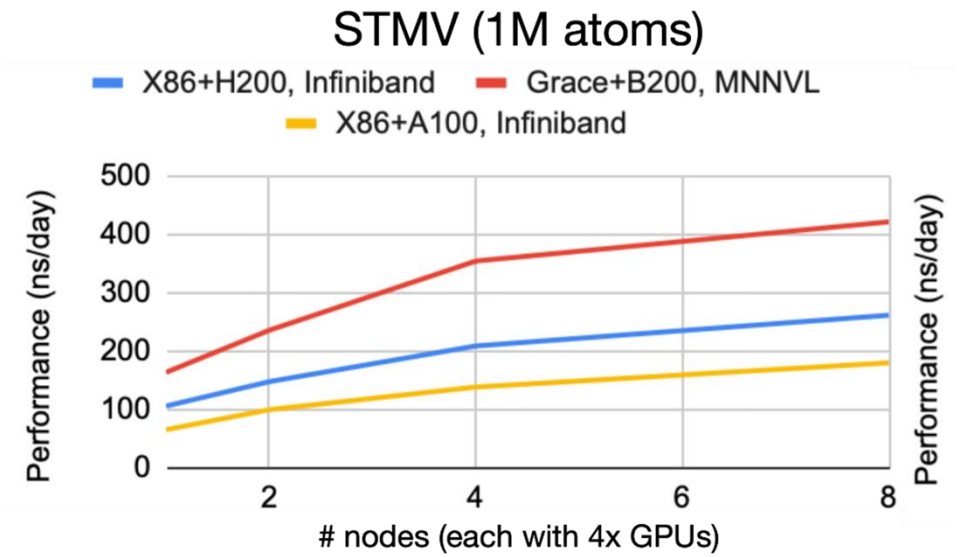


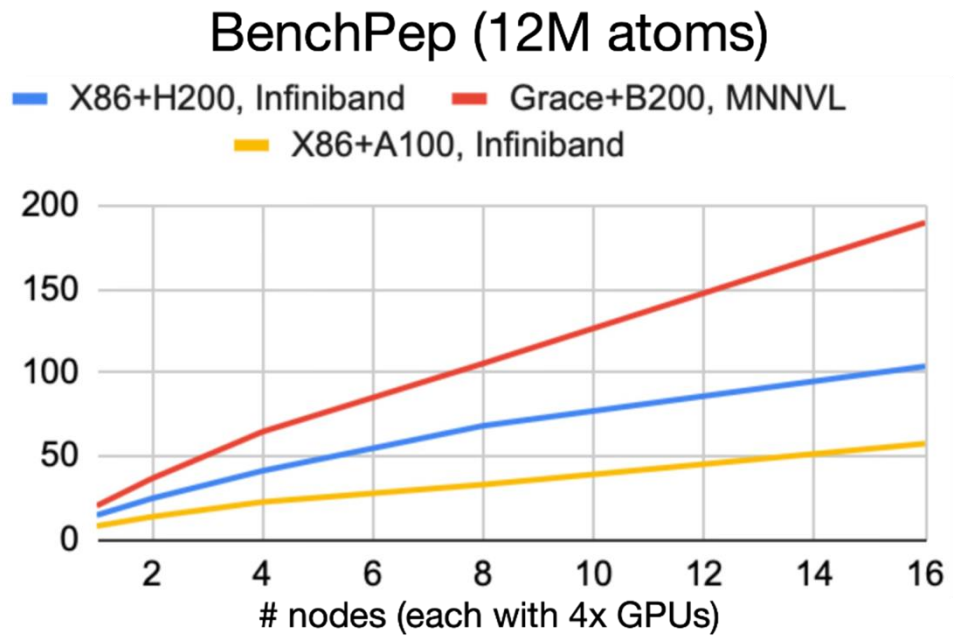
Efficient strong scaling down to ~25,000 atoms per GPU





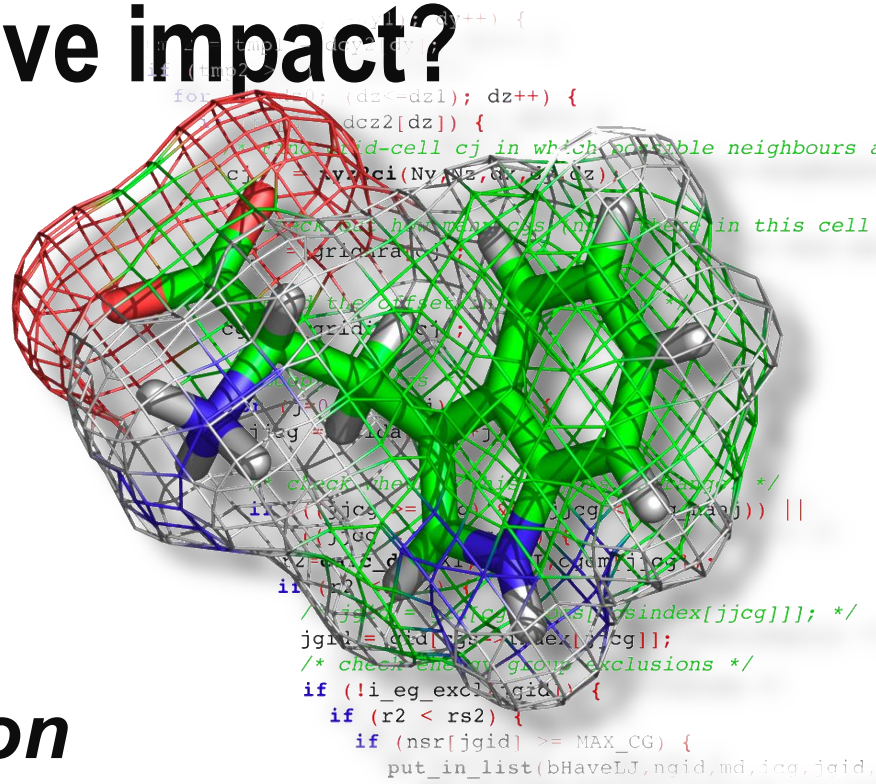
Tightly connected expensive nodes are not only good for AI!

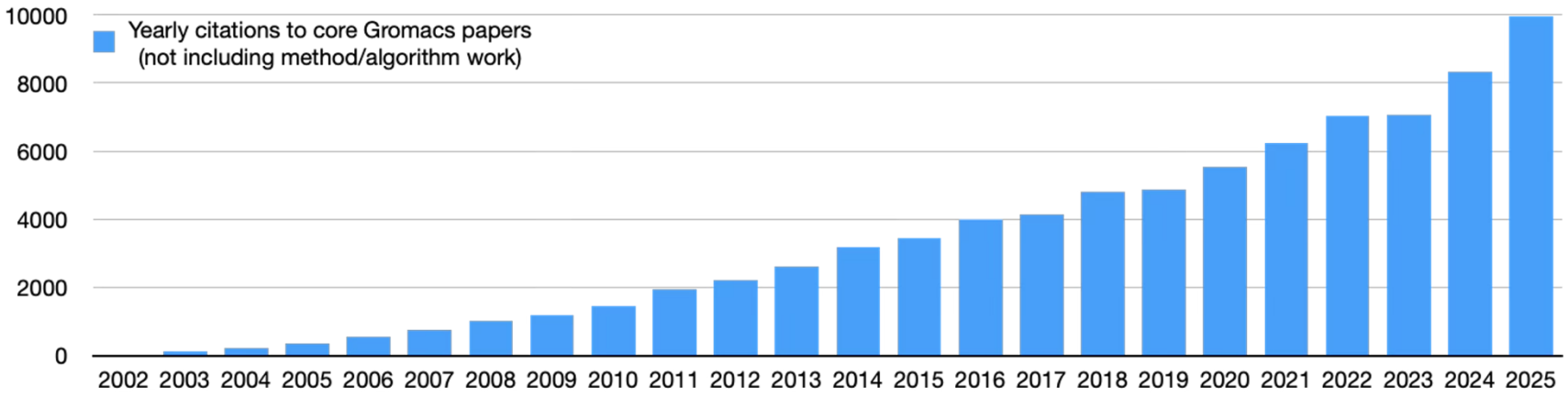




In retrospect: How does software achieve impact?

- Usability, quality, performance, portability
- You must build a broad *external* user base
- Support your users: blood, sweat & tears
- Just as for research, you first do the work, invest time & effort, show promising early impact on others - and then get funding.
- "***Life's most persistent and urgent question is, 'What are you doing for others?'***"

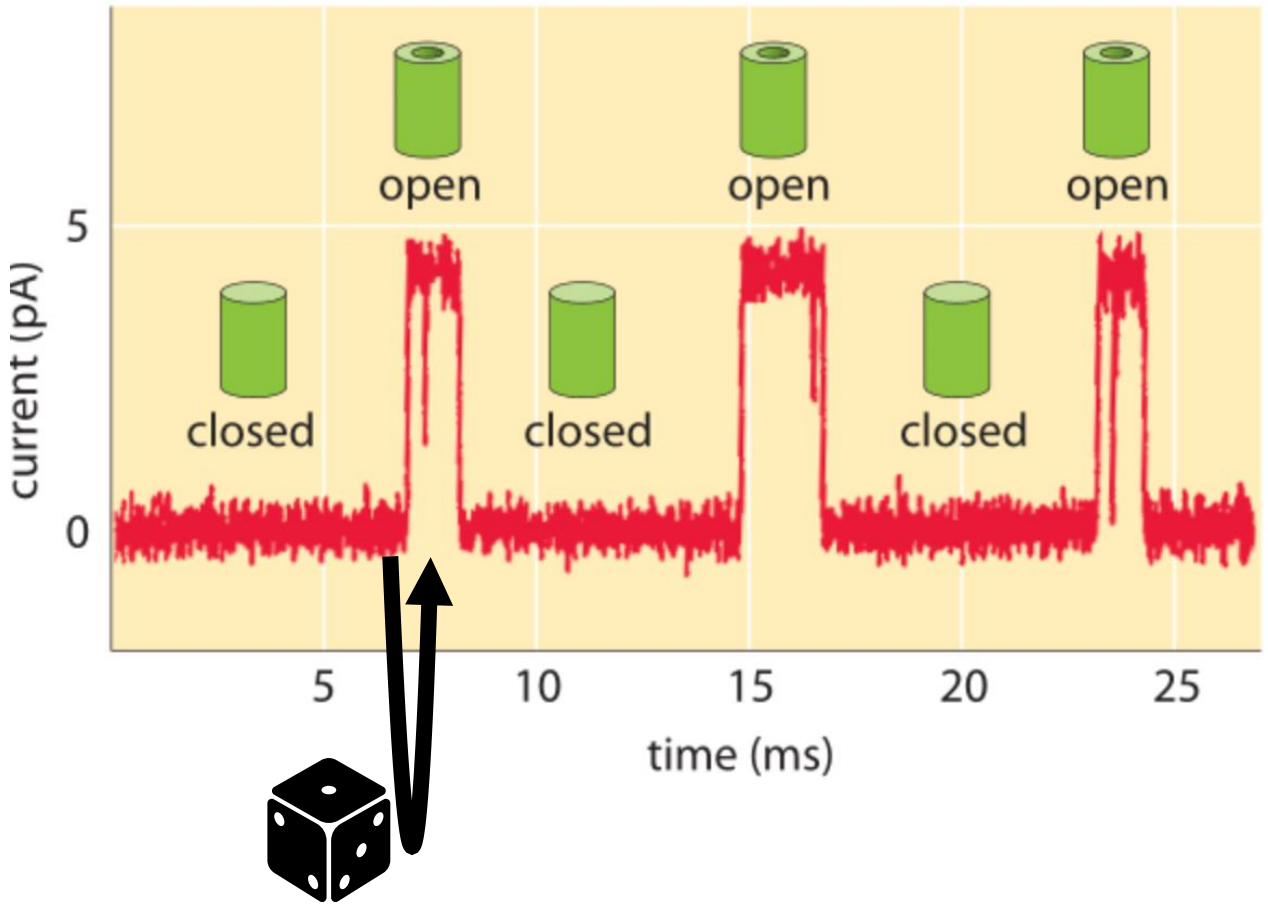




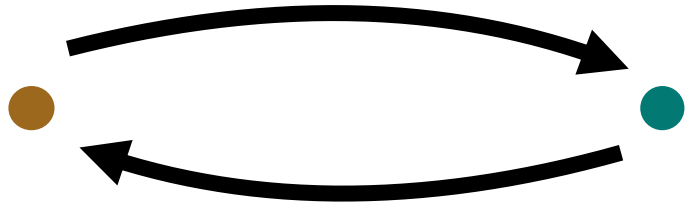
26,060 FPS

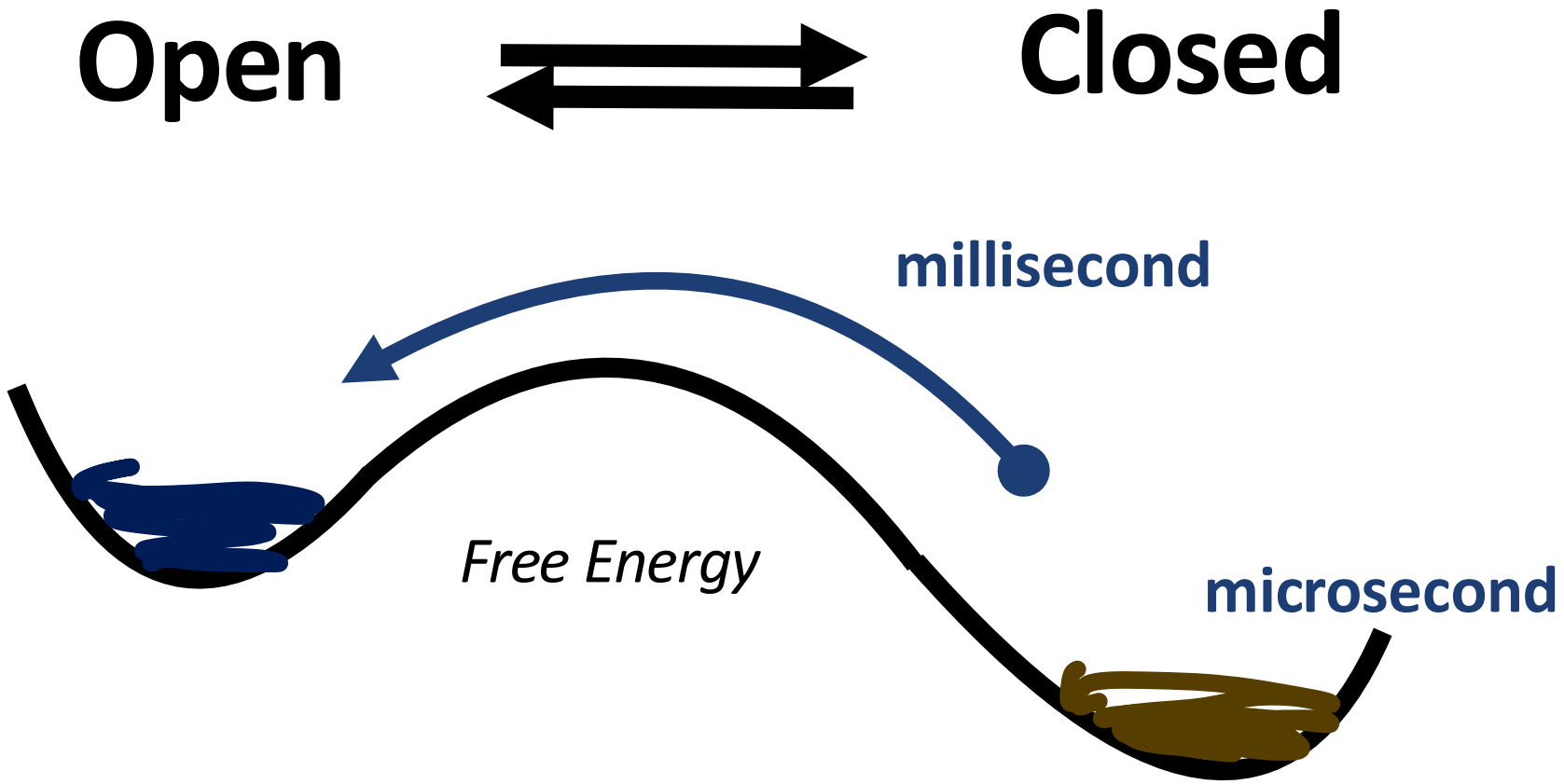
MODELING CONFORMATIONAL CHANGES WITH (MANY) SIMULATIONS

QUANTIFYING SLOW GATING IN SIMULATIONS: MARKOV STATE MODELS

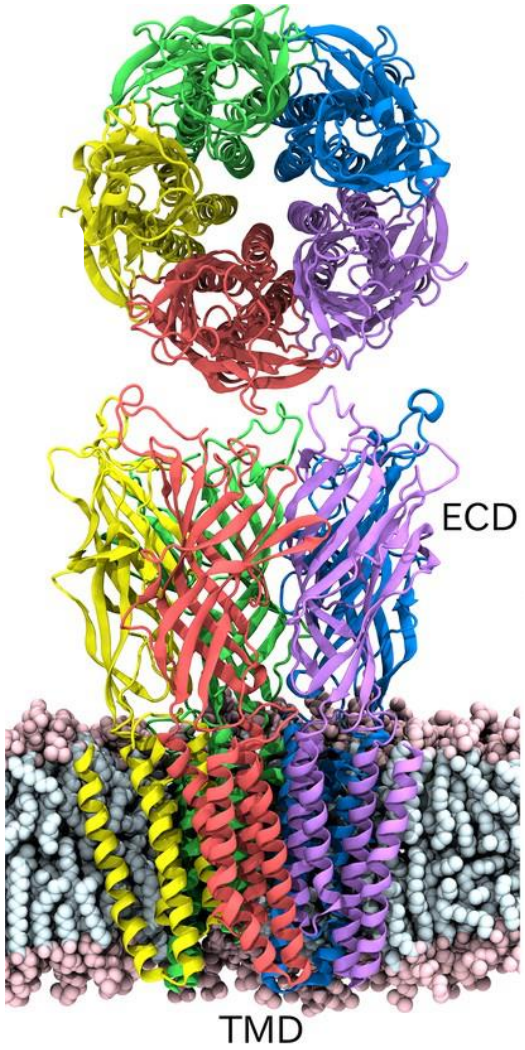


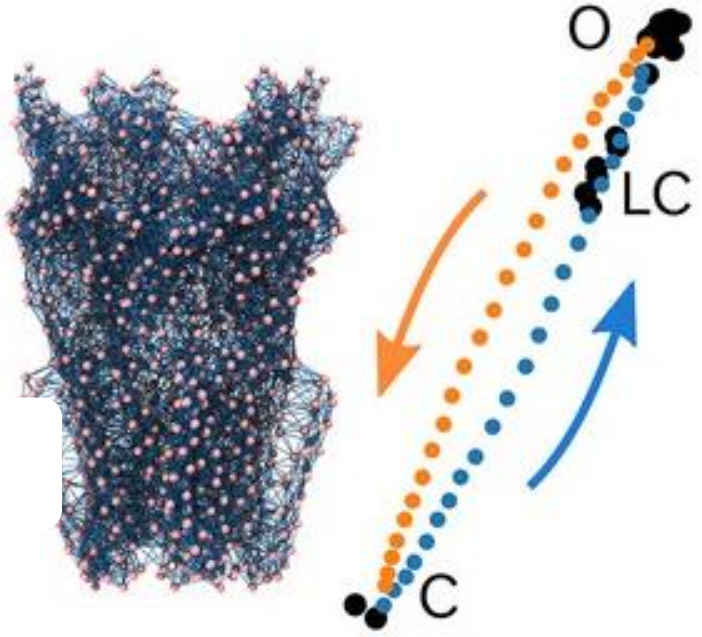
Kinetic model



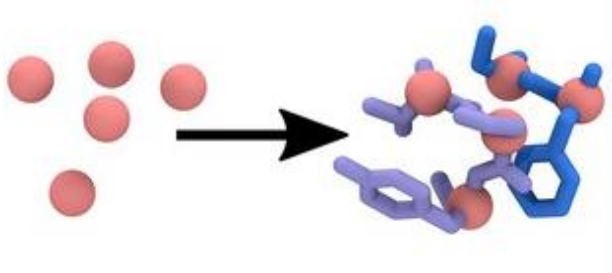

Open
Closed
millisecond
Free Energy
microsecond

QUANTIFYING THE GATING PROCESS WITH MARKOV STATE MODELS

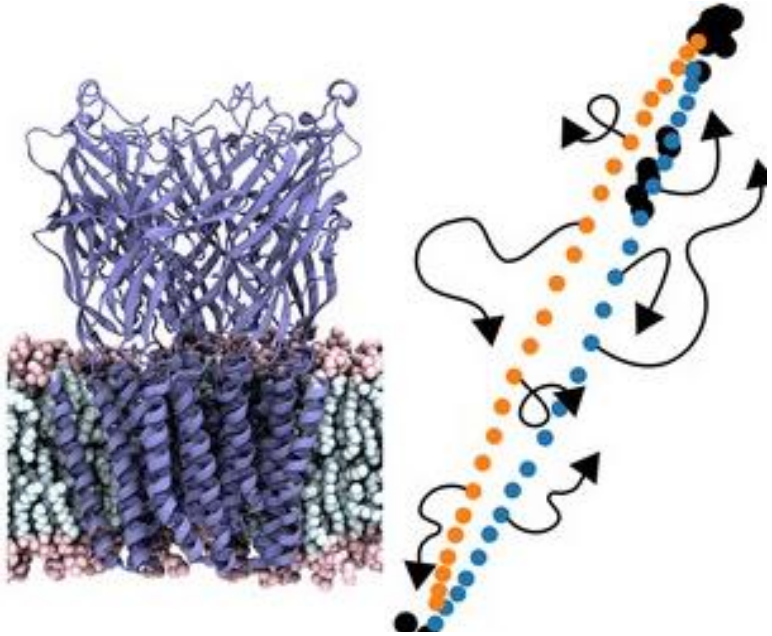


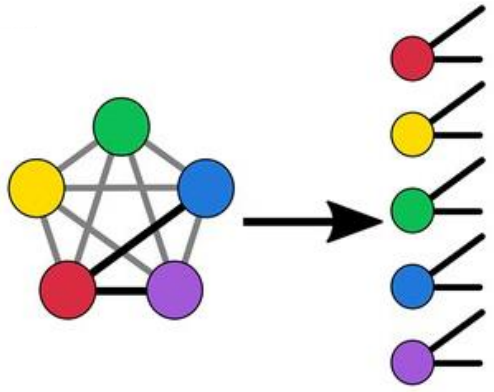


https://ebdims.biophysics.se/



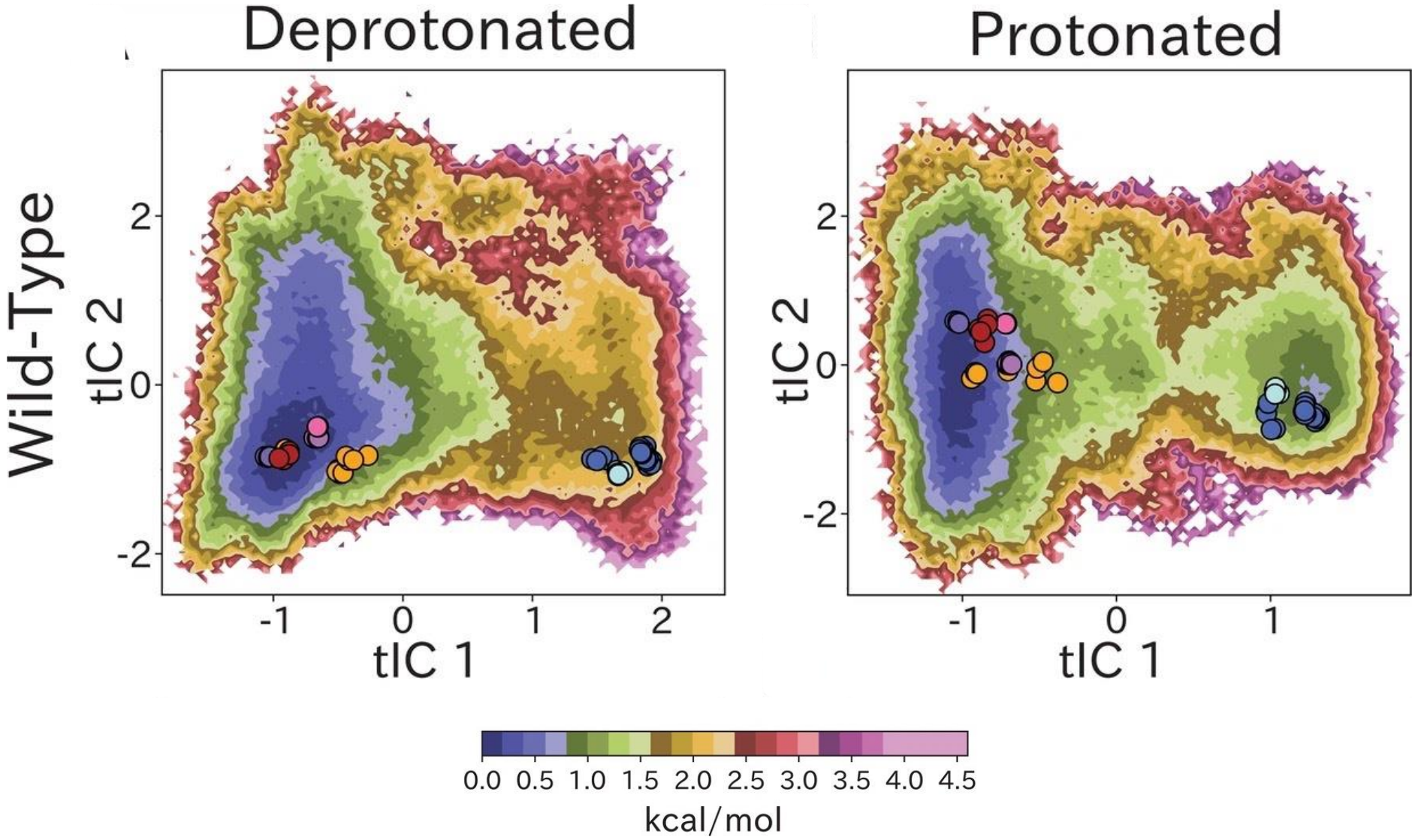
Laura Orellana, Nature Comm. 7, 12575 (2016)



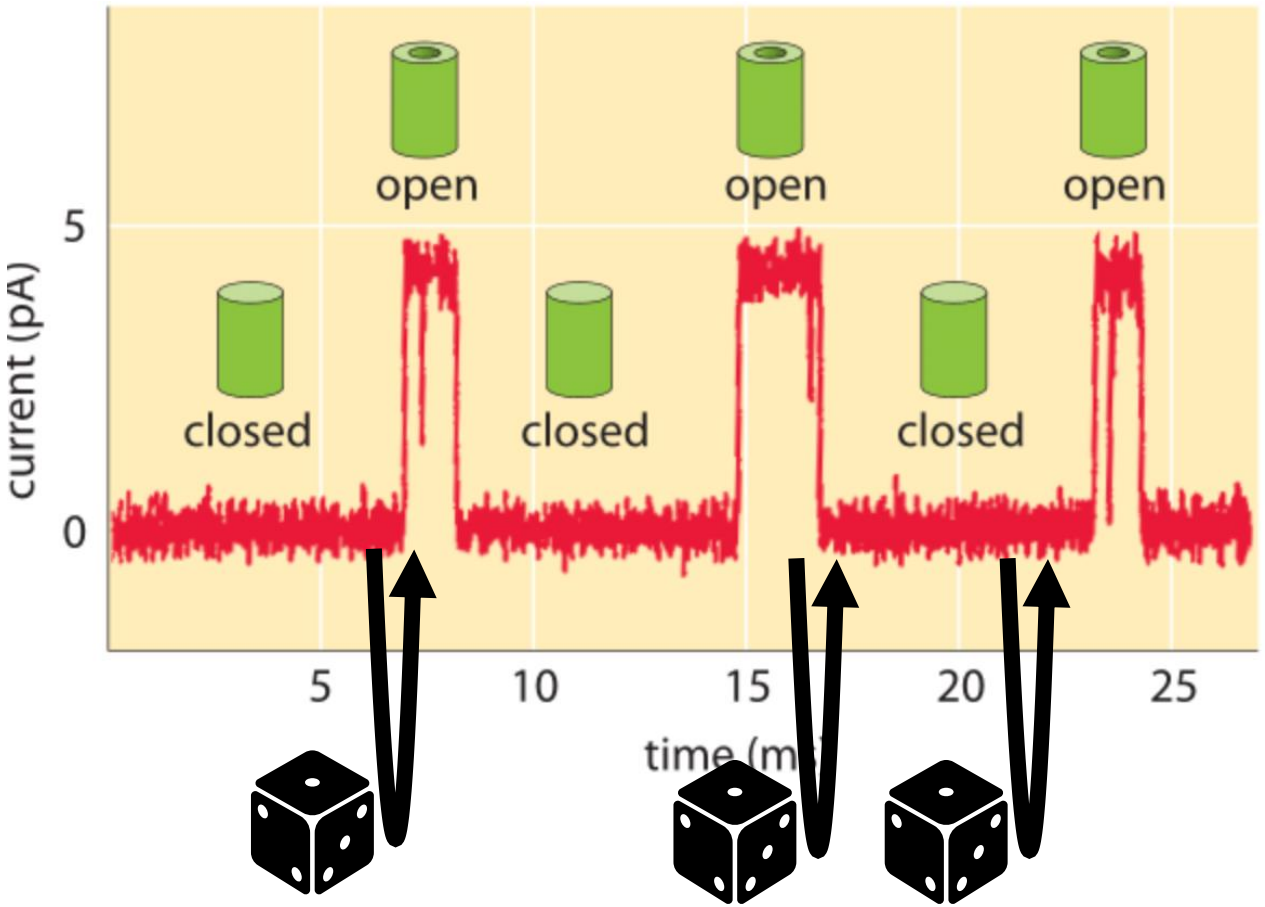


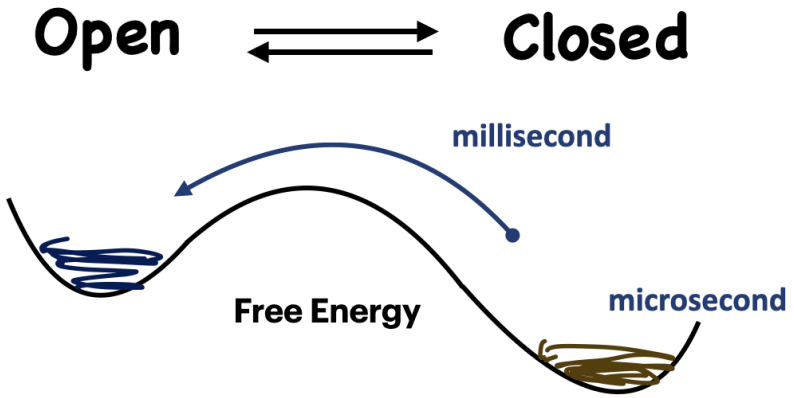
Cathrine Bergh, eLife 10:e68369 (2021)

MSM FREE ENERGY LANDSCAPES INDICATE LOW OPEN PROBABILITIES

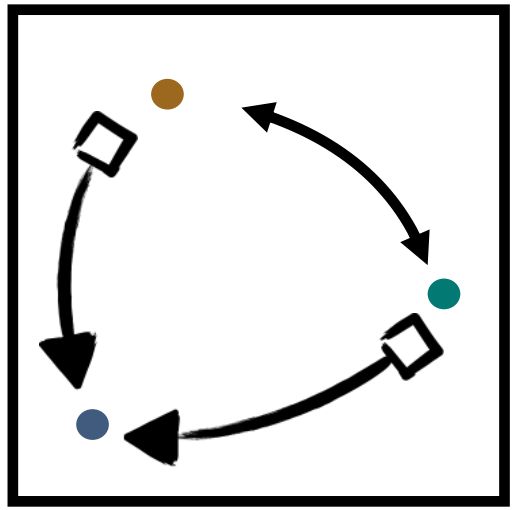


MODELING A MORE REALISTIC CONFORMATIONAL CYCLE

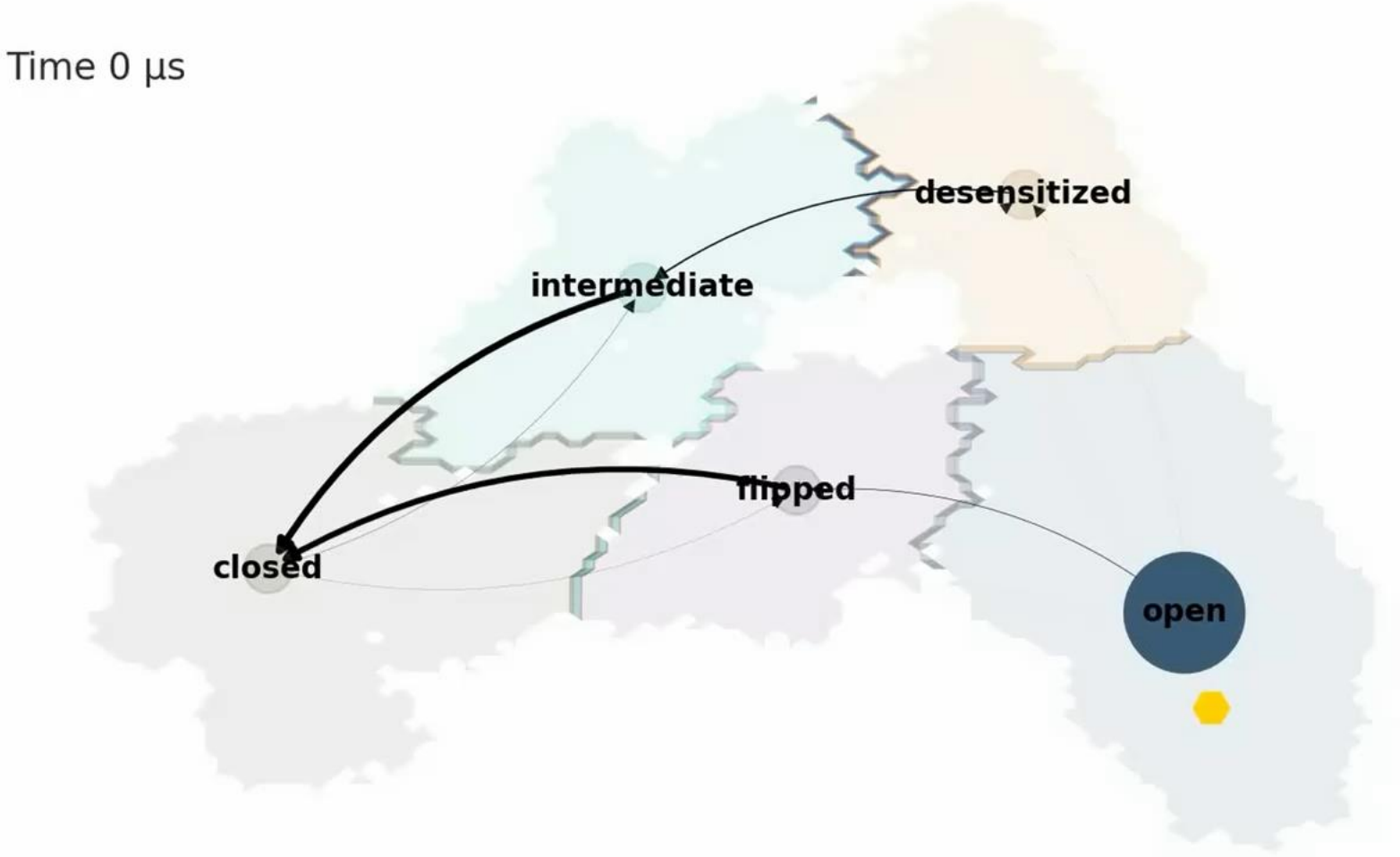


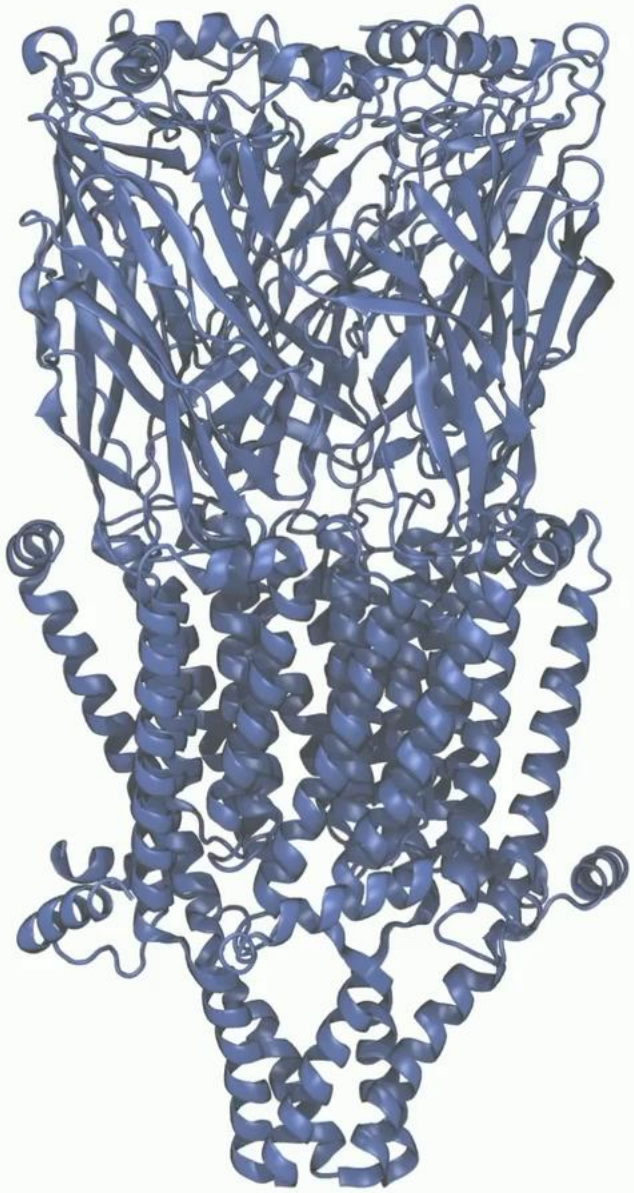


Kinetic model
including desensitization



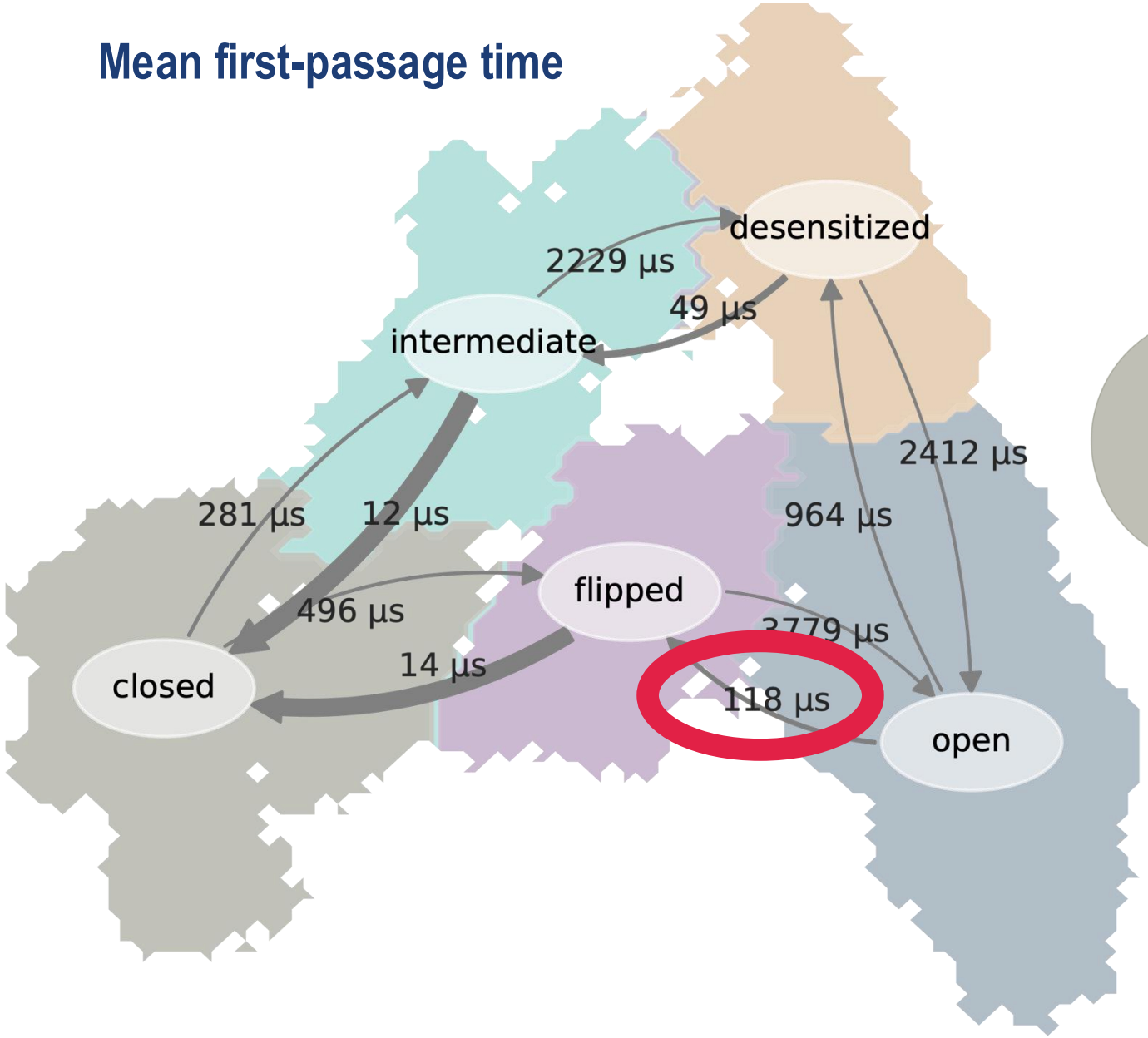
VISUALIZING 1MS OF NACHR GATING FROM MSM SAMPLING

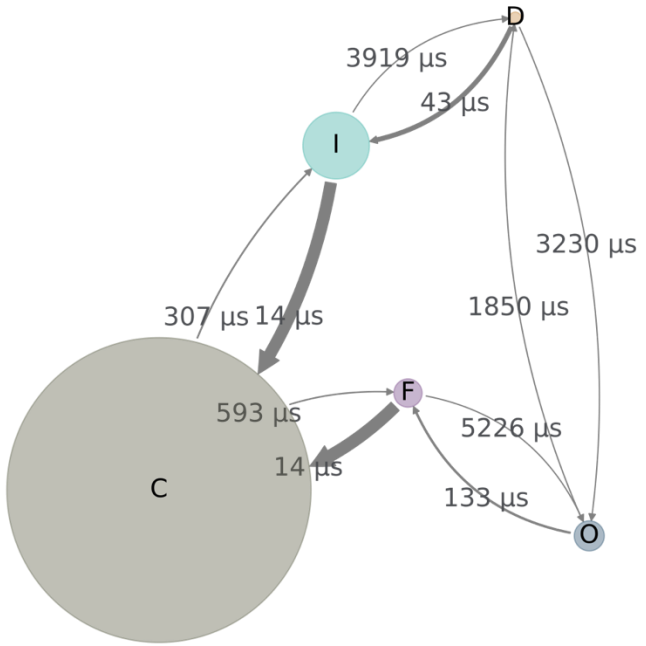


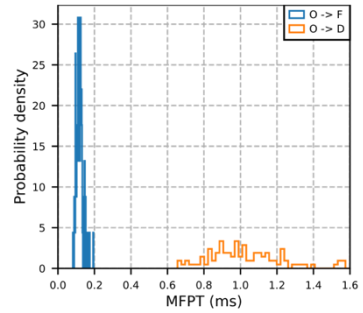


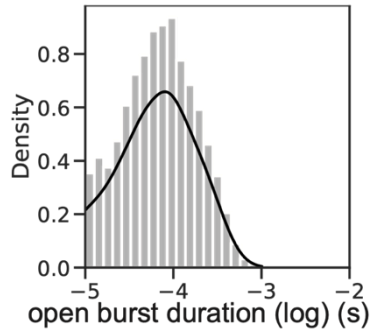
TIMESCALES OF nAChR GATING & DESENSITIZATION FROM MSMs

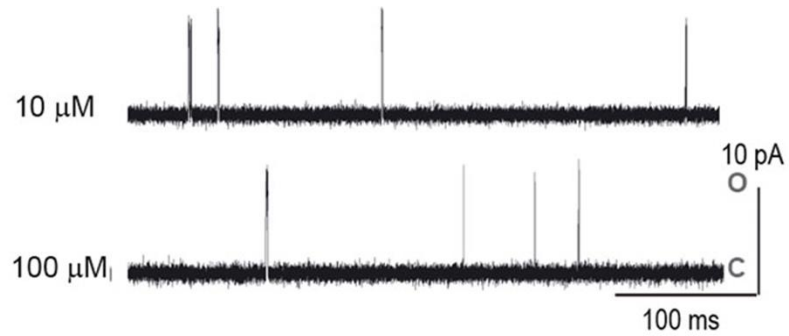
Mean first-passage time









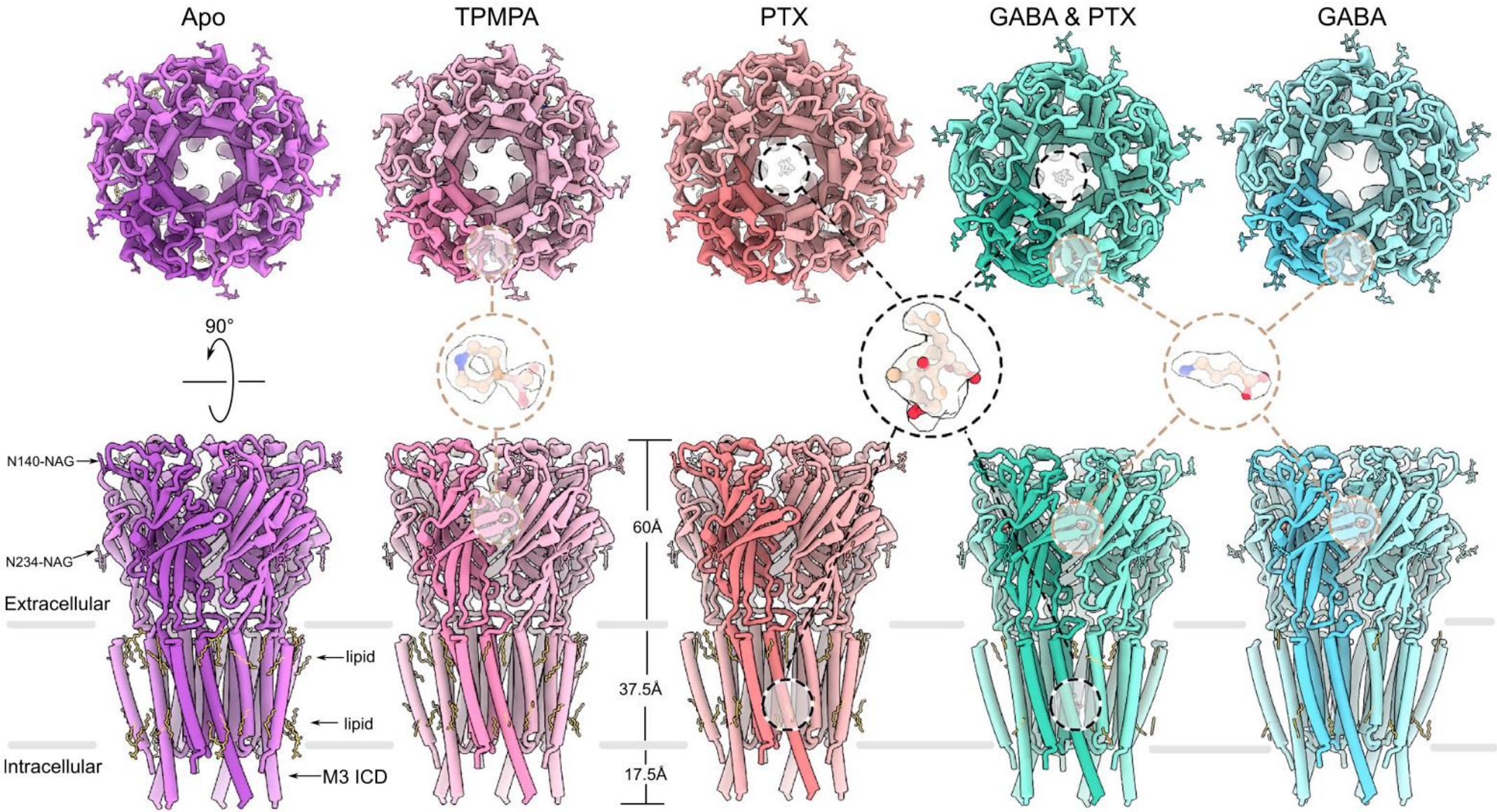


E-phys: "Opening exists as 100 µs isolated events"

Bouzat, 2008, 2018

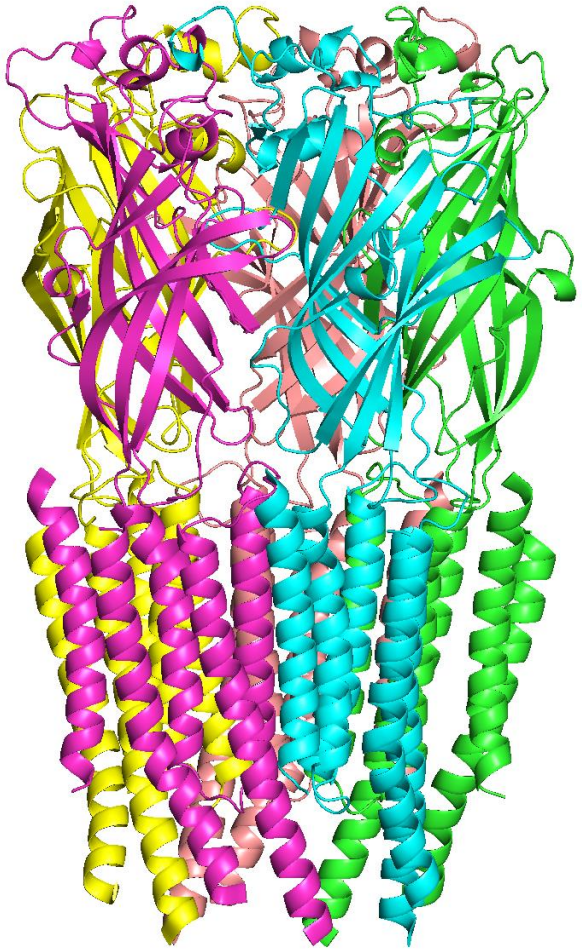
REVISITING THE SAME CHALLENGE WITH AI

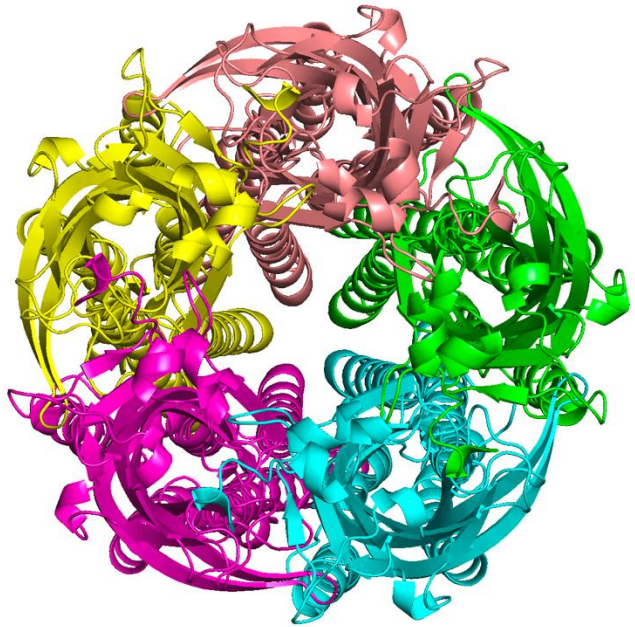
THE GABA$_A$-ρ1 RECEPTOR

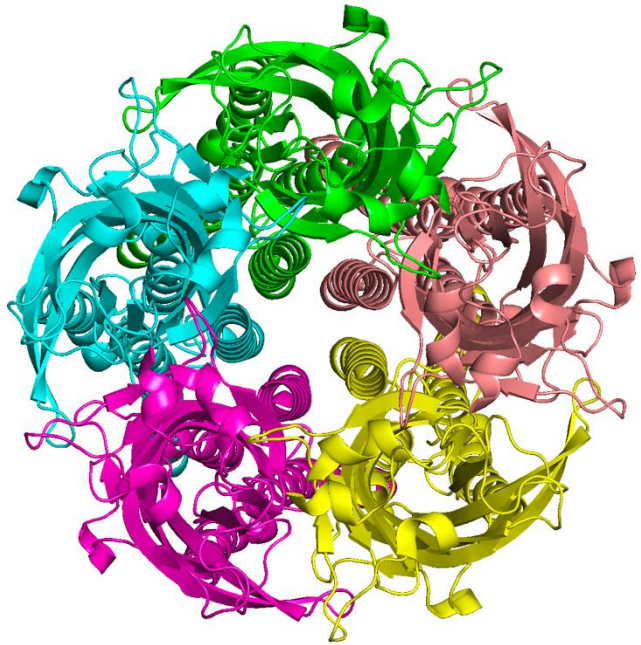


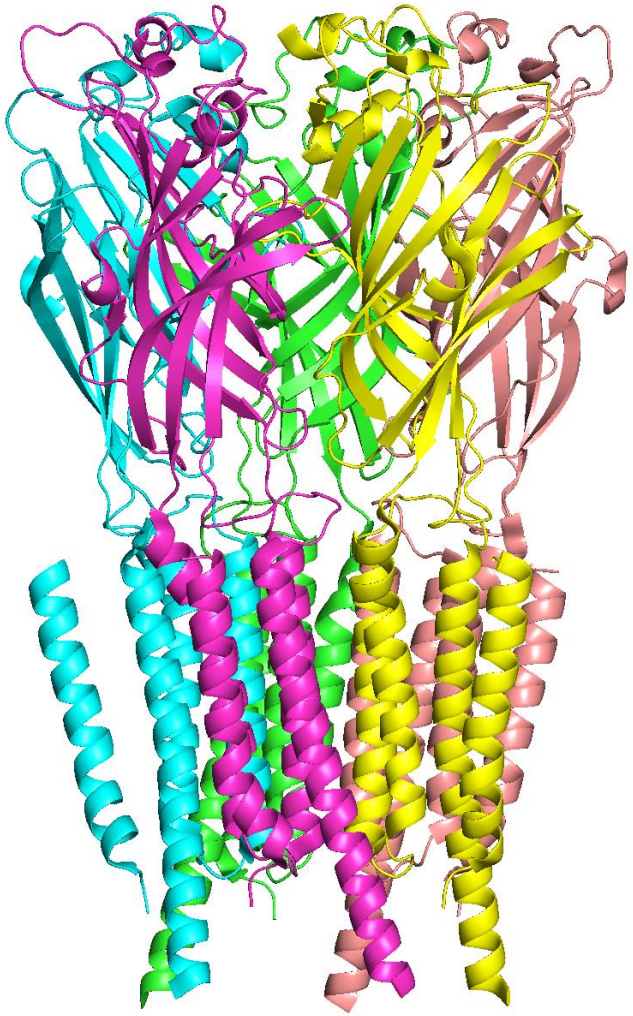
ALPHAFOLD 3

CRYO-EM

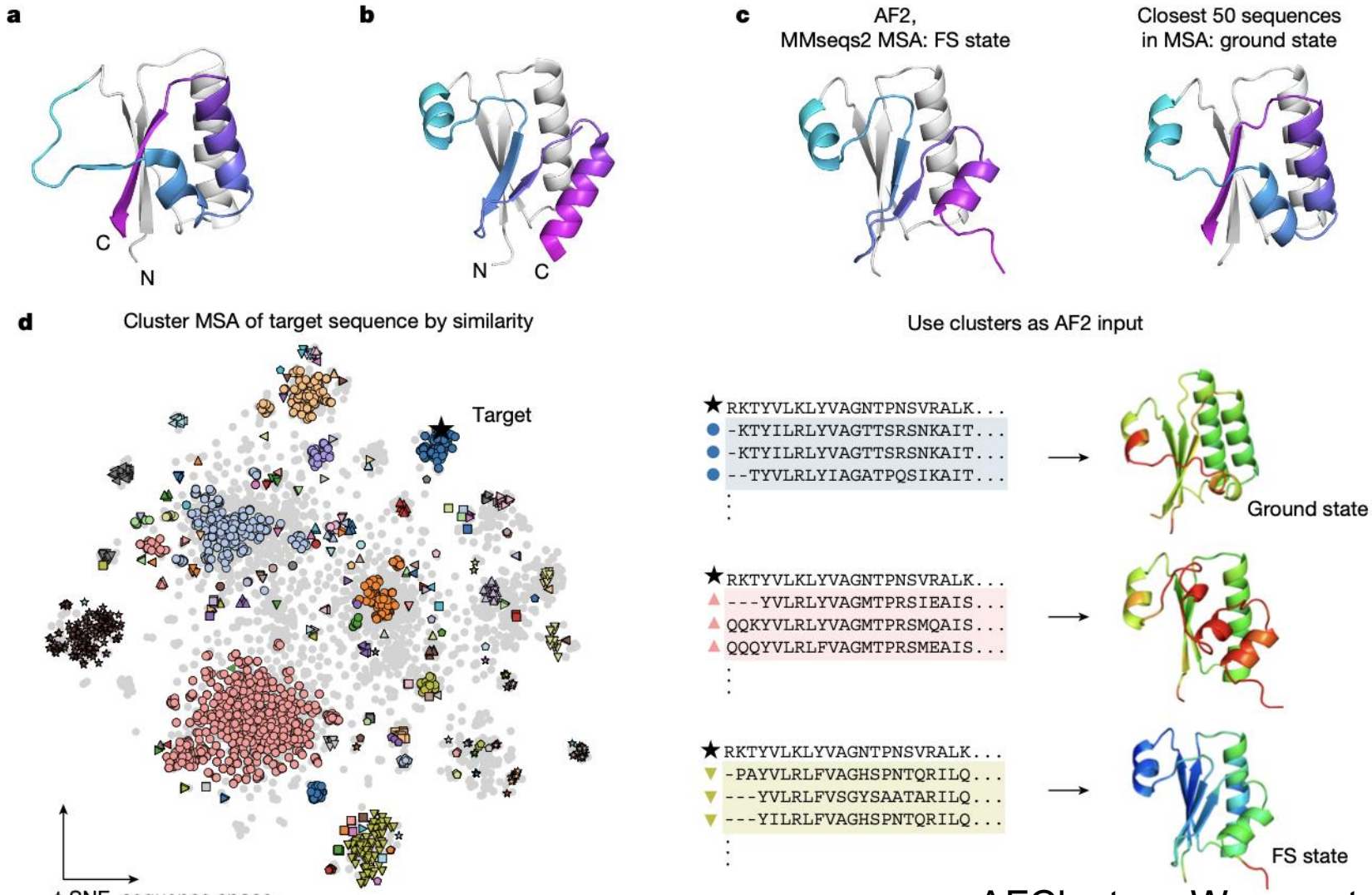






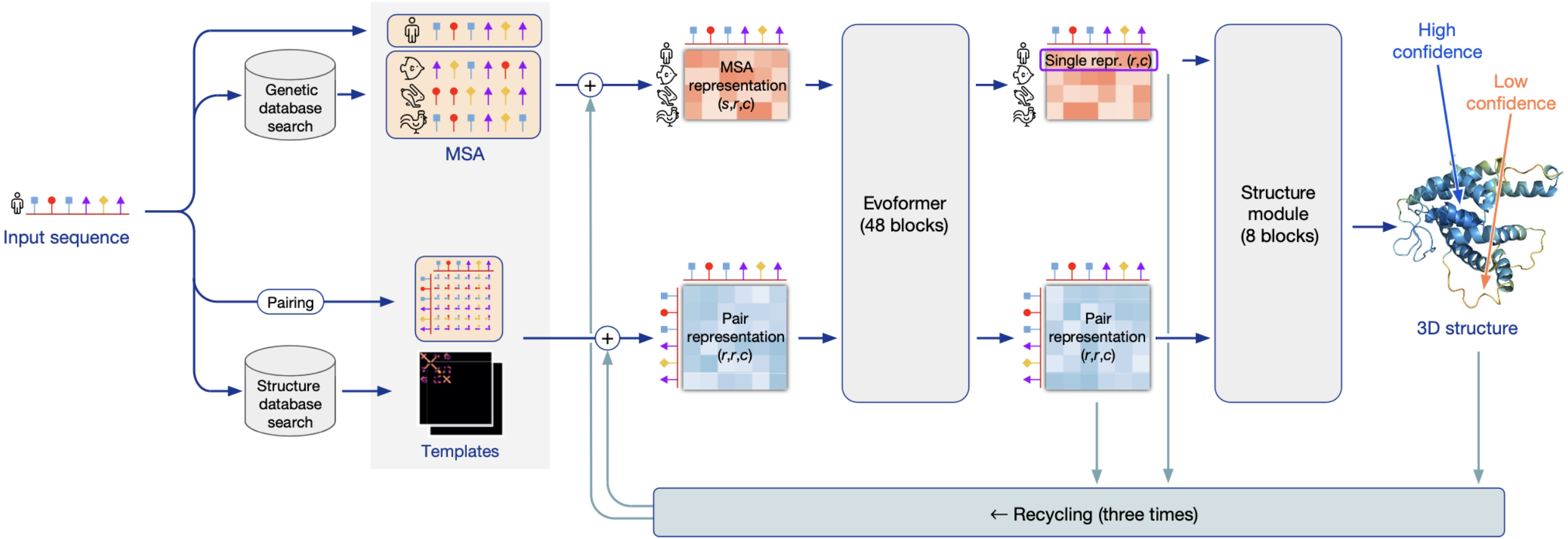


Can structure prediction methods resolve multiple different states?

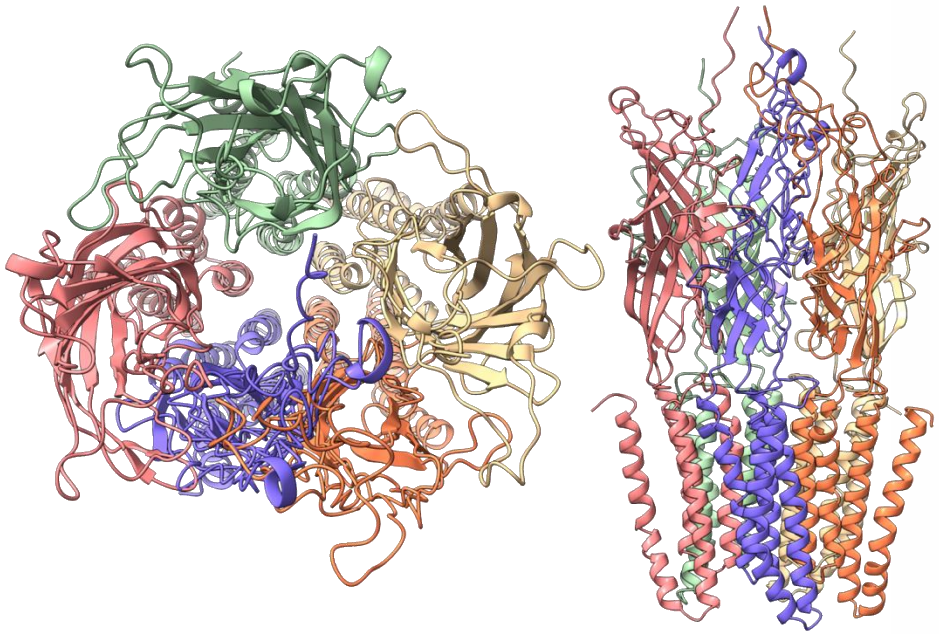


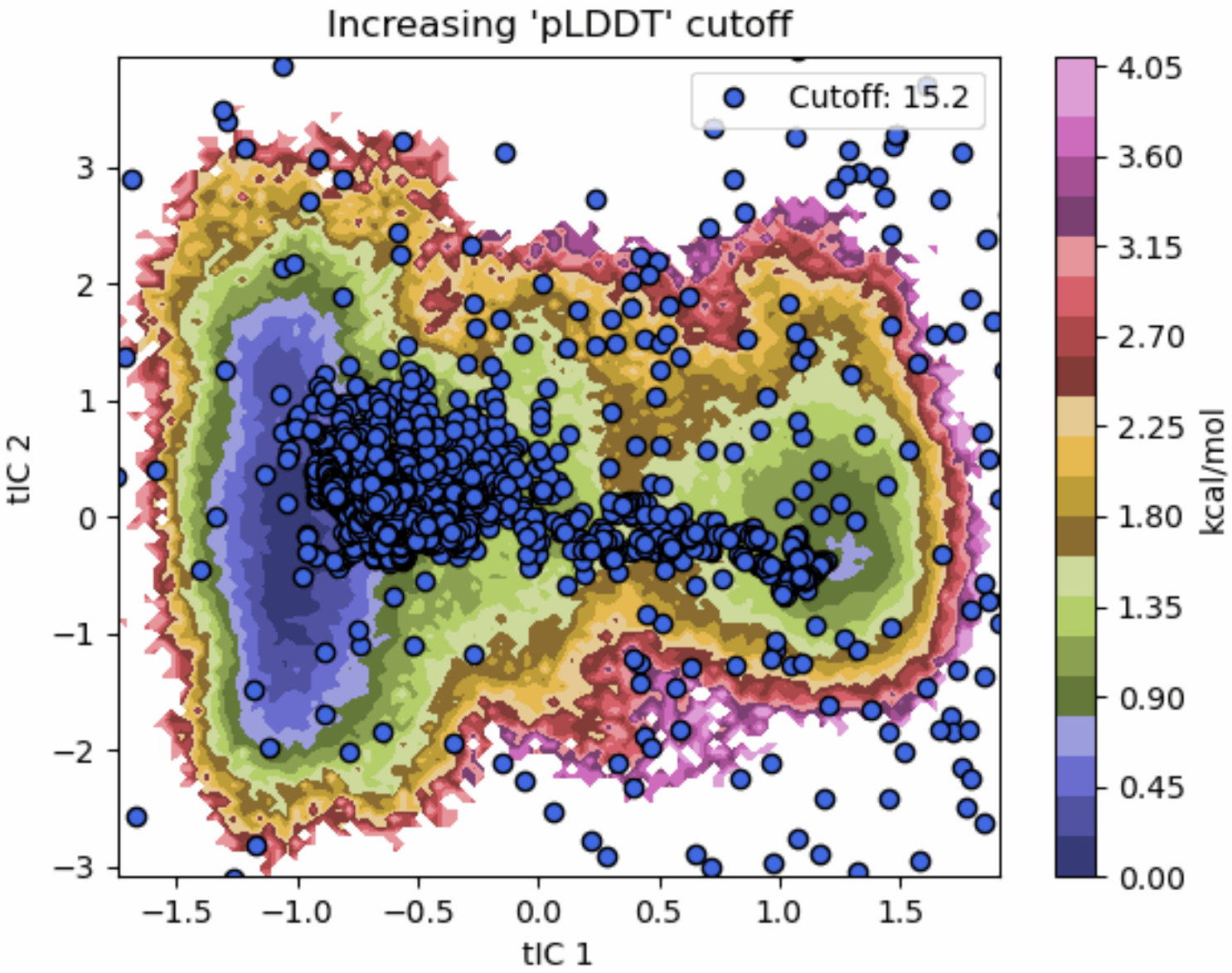
AFCluster - Wayment-Steele et al. (2023)

The MSA stage is critical to what AlphaFold generates



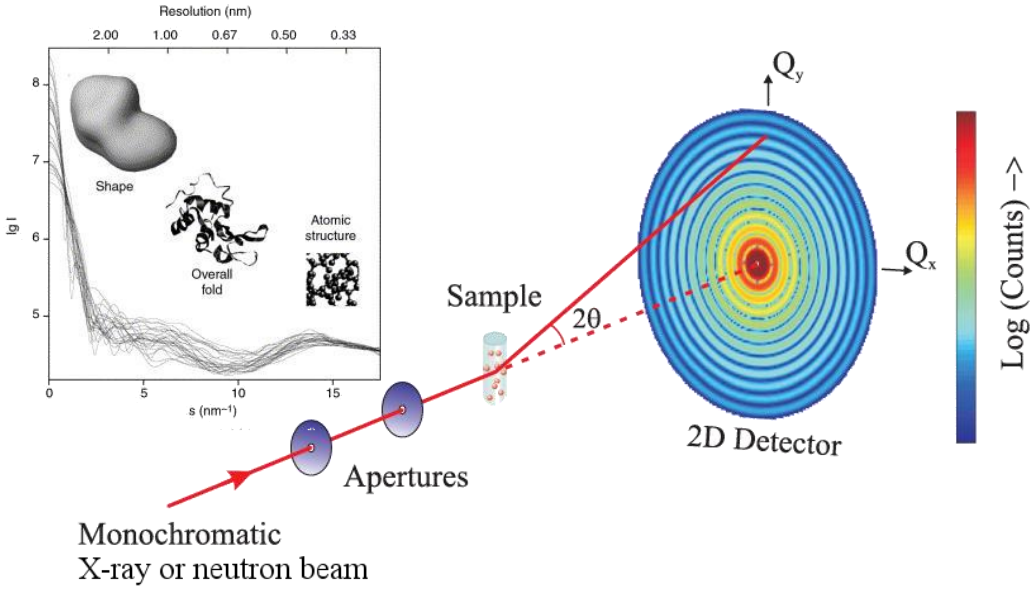
STOCHASTIC MSA SUBSAMPLING & FILTERING

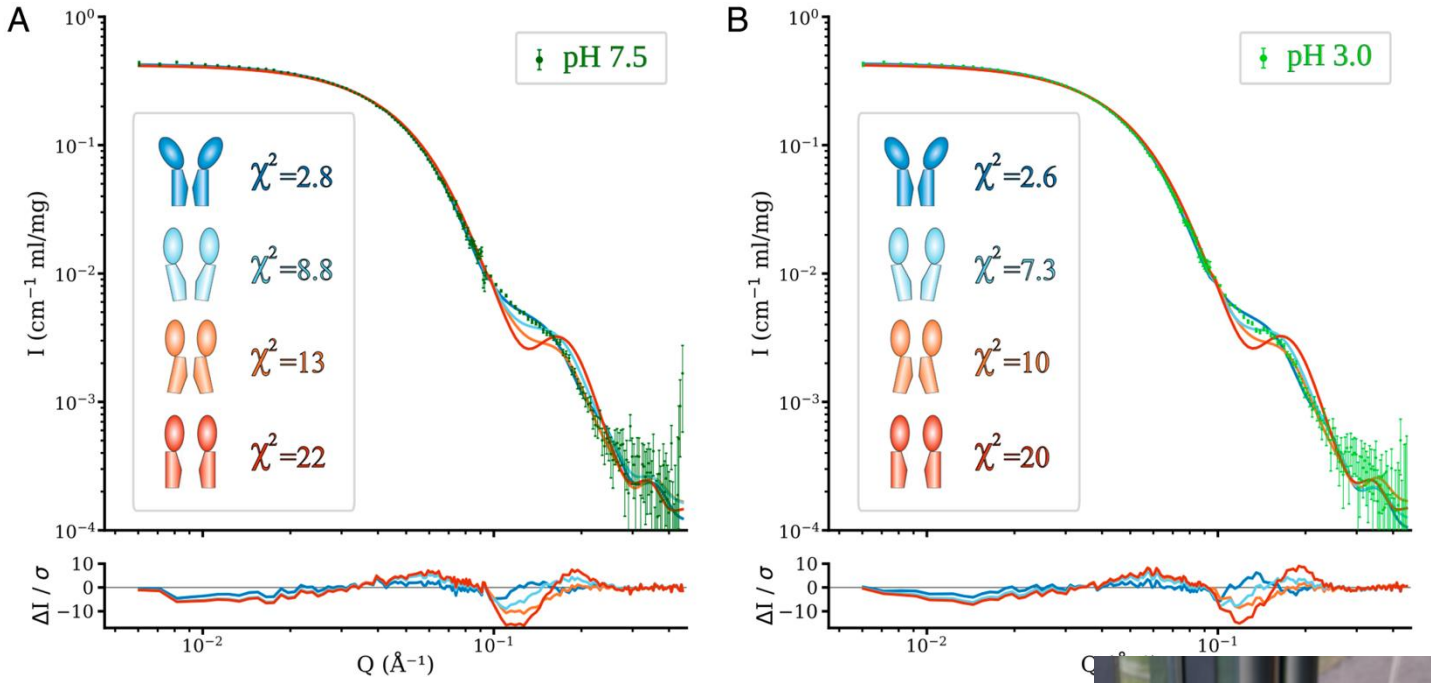




Samuel Eriksson Lidbrink, Nandan Haloi

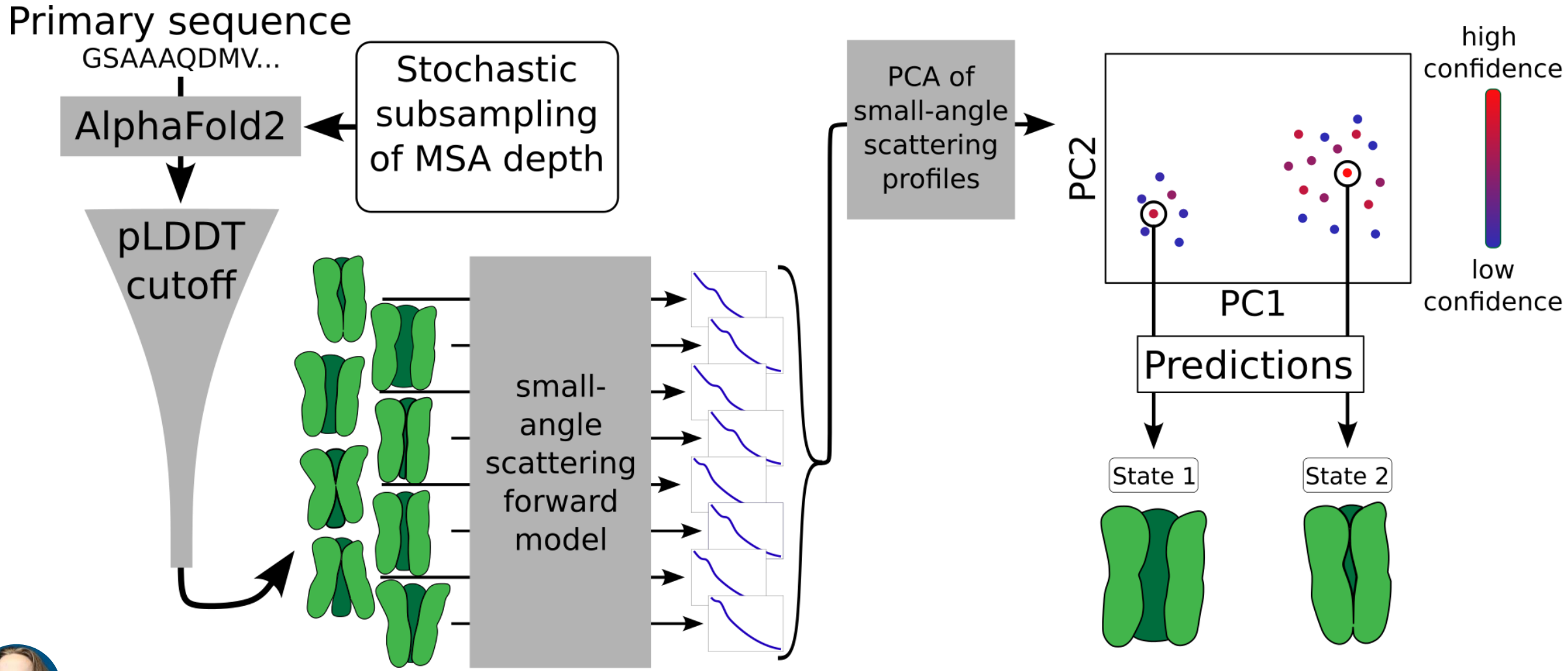
Use low-resolution data (SANS) to discriminate resting/active state





Marie Lycksell

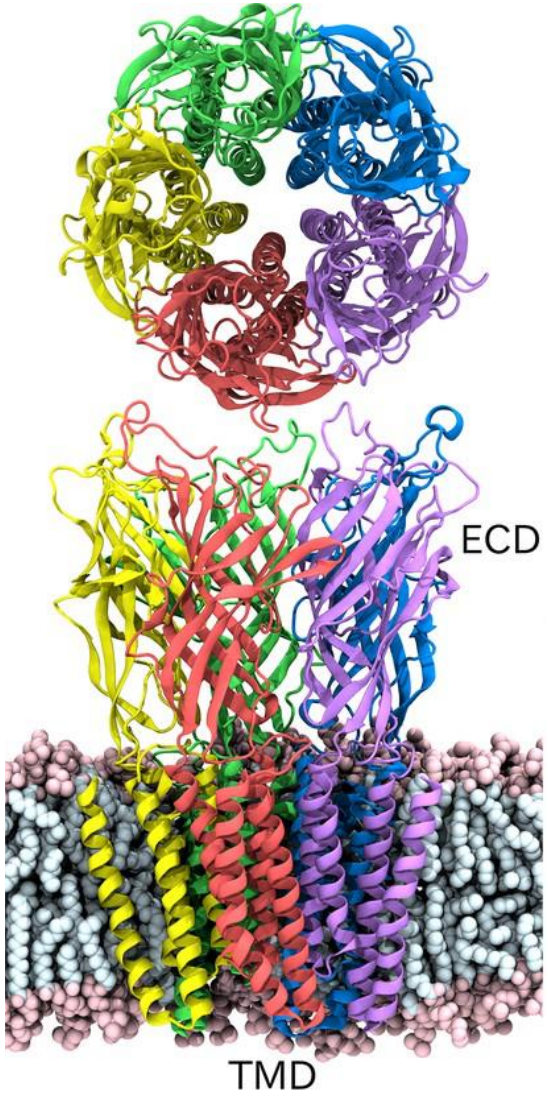
AlphaFold generates candidates while SANS discriminates

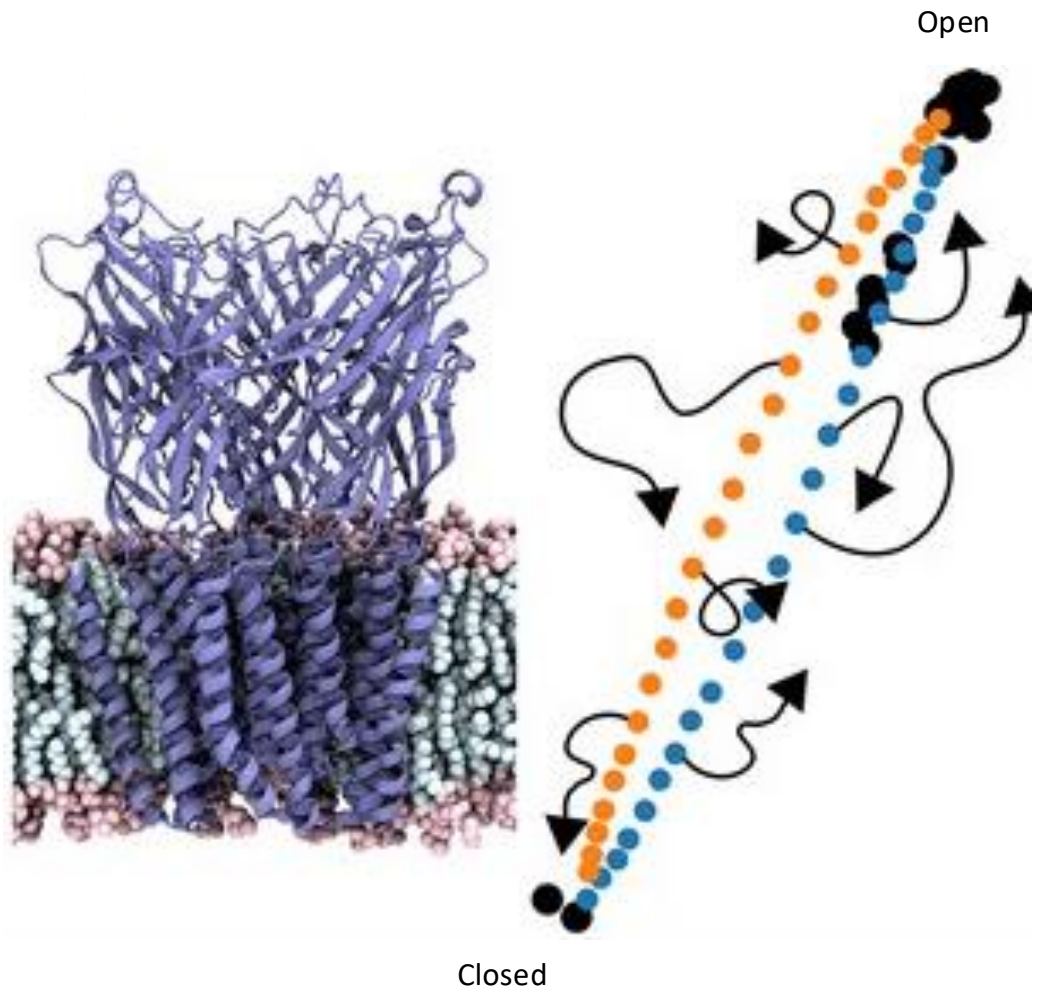


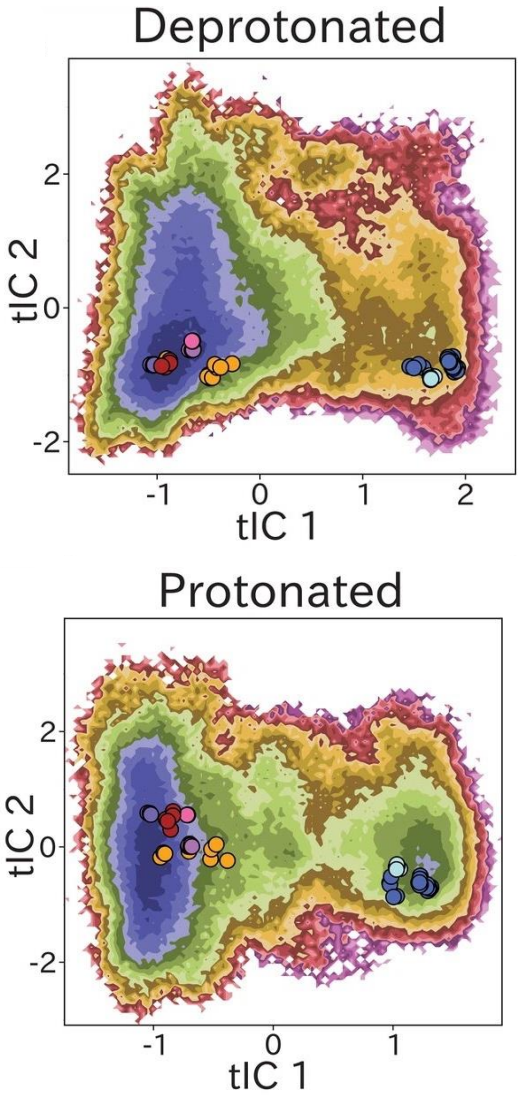
Samuel Eriksson Lidbrink

PLoS Comp Biol 21: e1013187

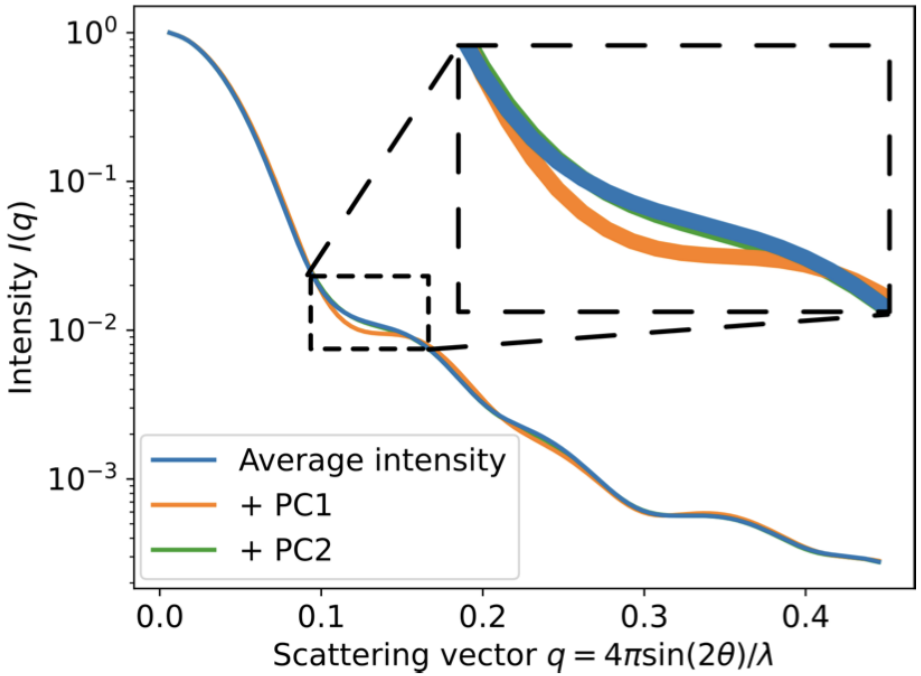
Assess against full free energy landscapes from MSM simulations

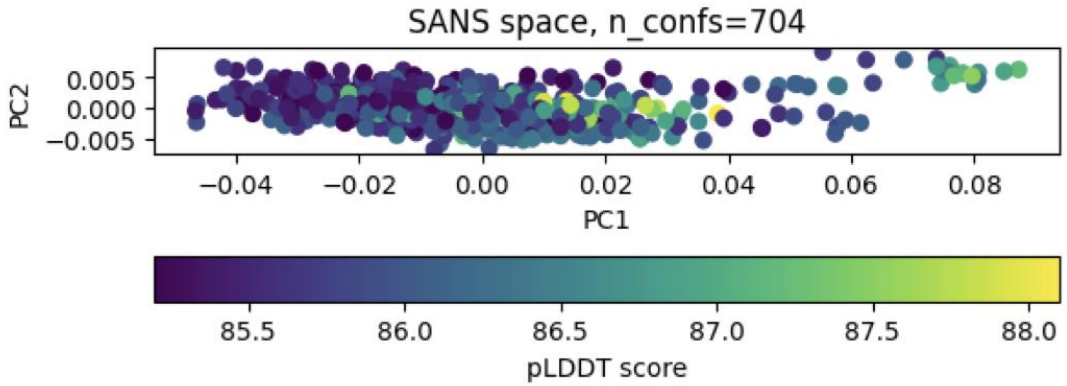


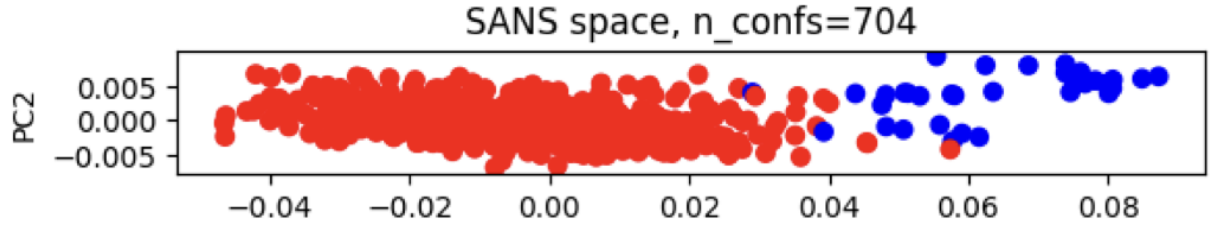


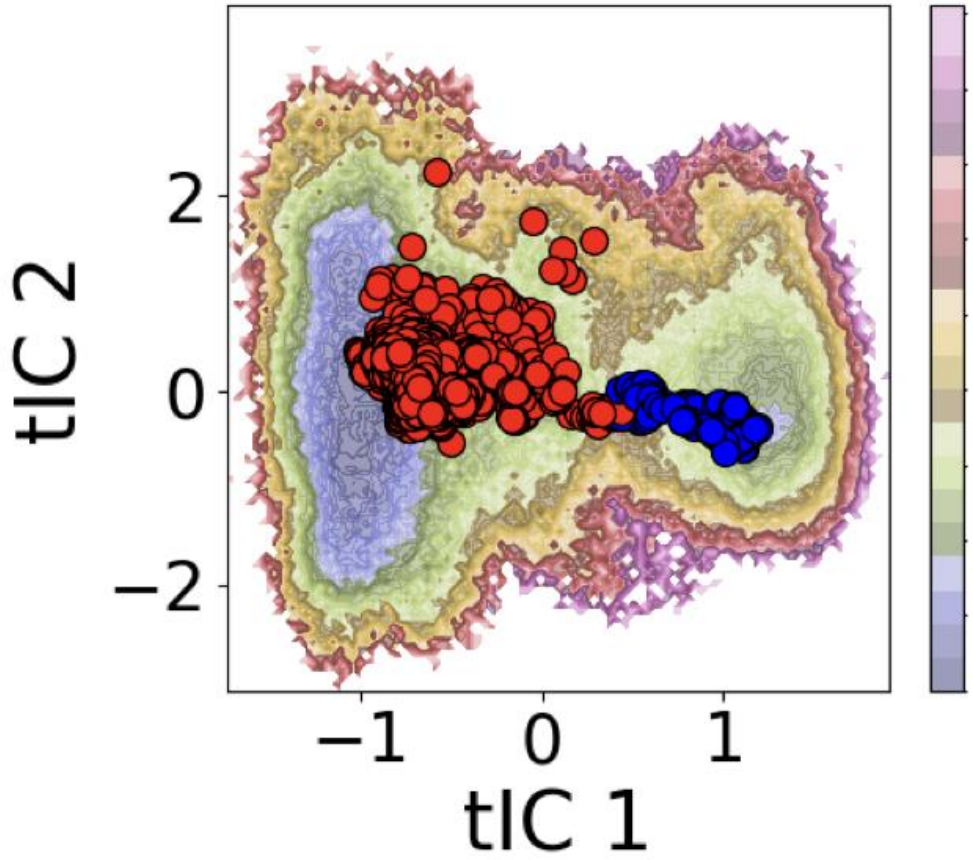


Bergh et al., eLife 10:e68369 (2021)



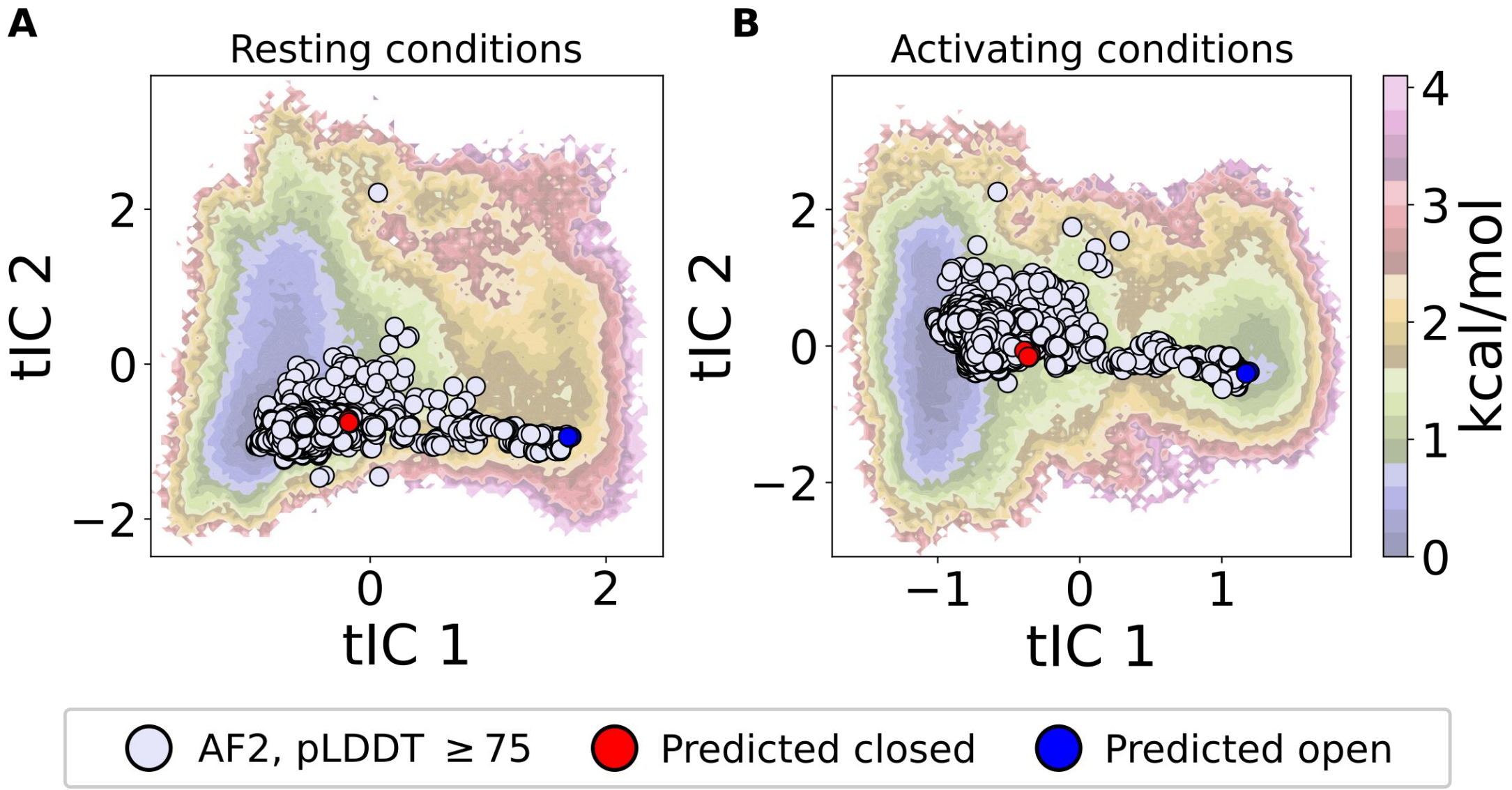




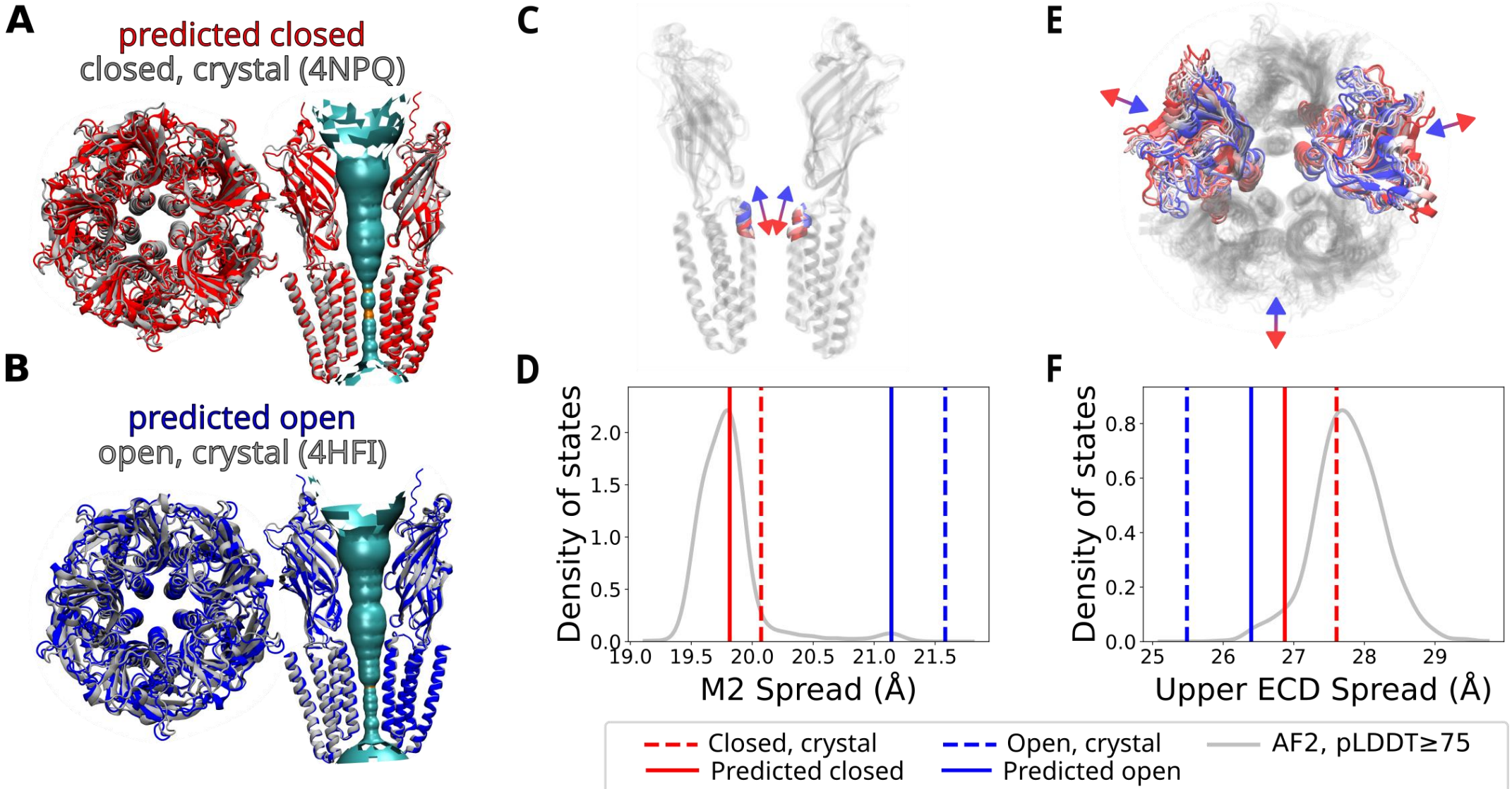


Samuel Eriksson Lidbrink, Nandan Haloi

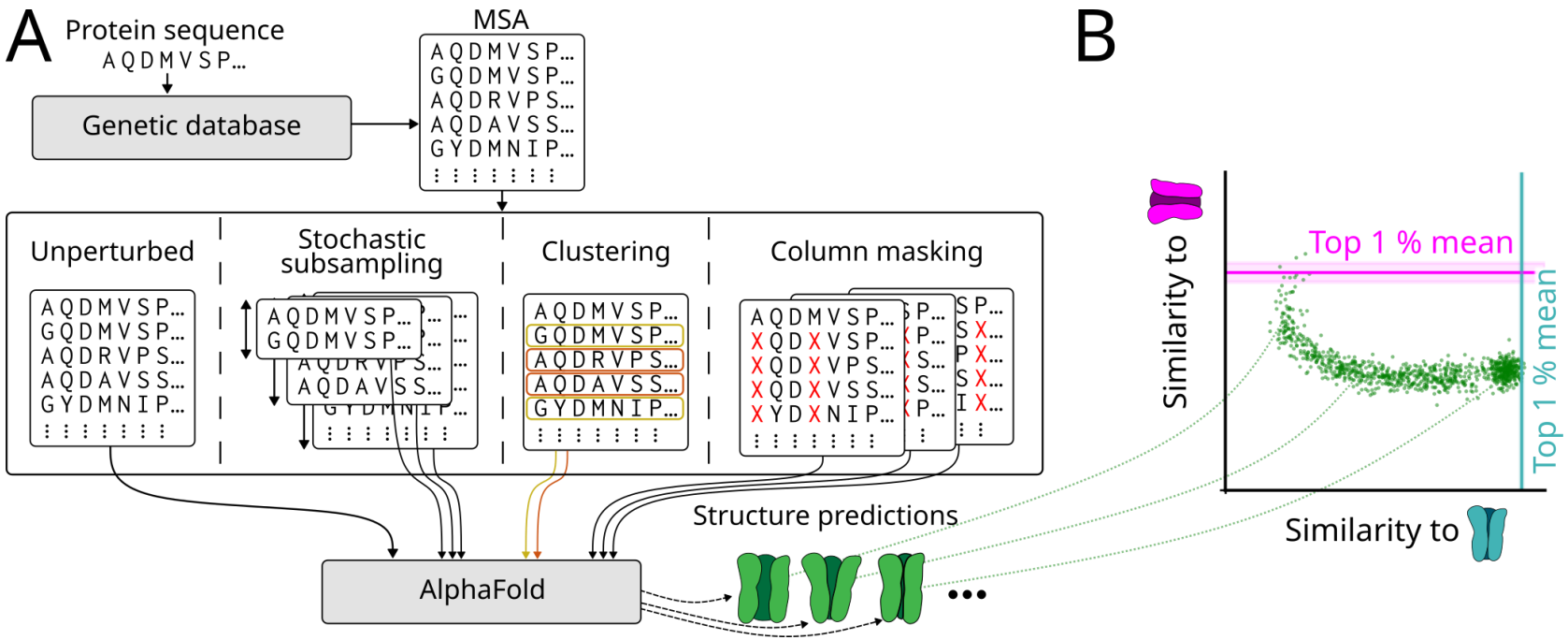
Structure ensembles reproduce minima and paths in energy landscapes



Selected predictions are close to X-ray structures of closed/open states

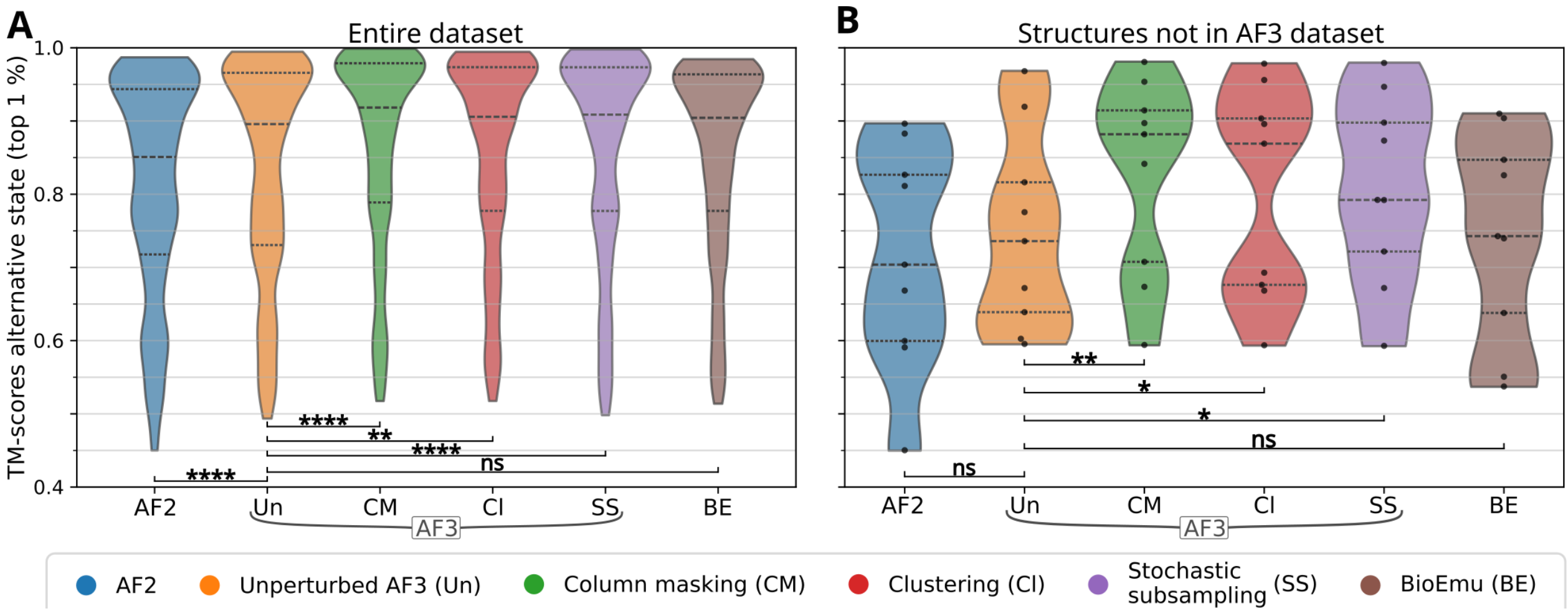


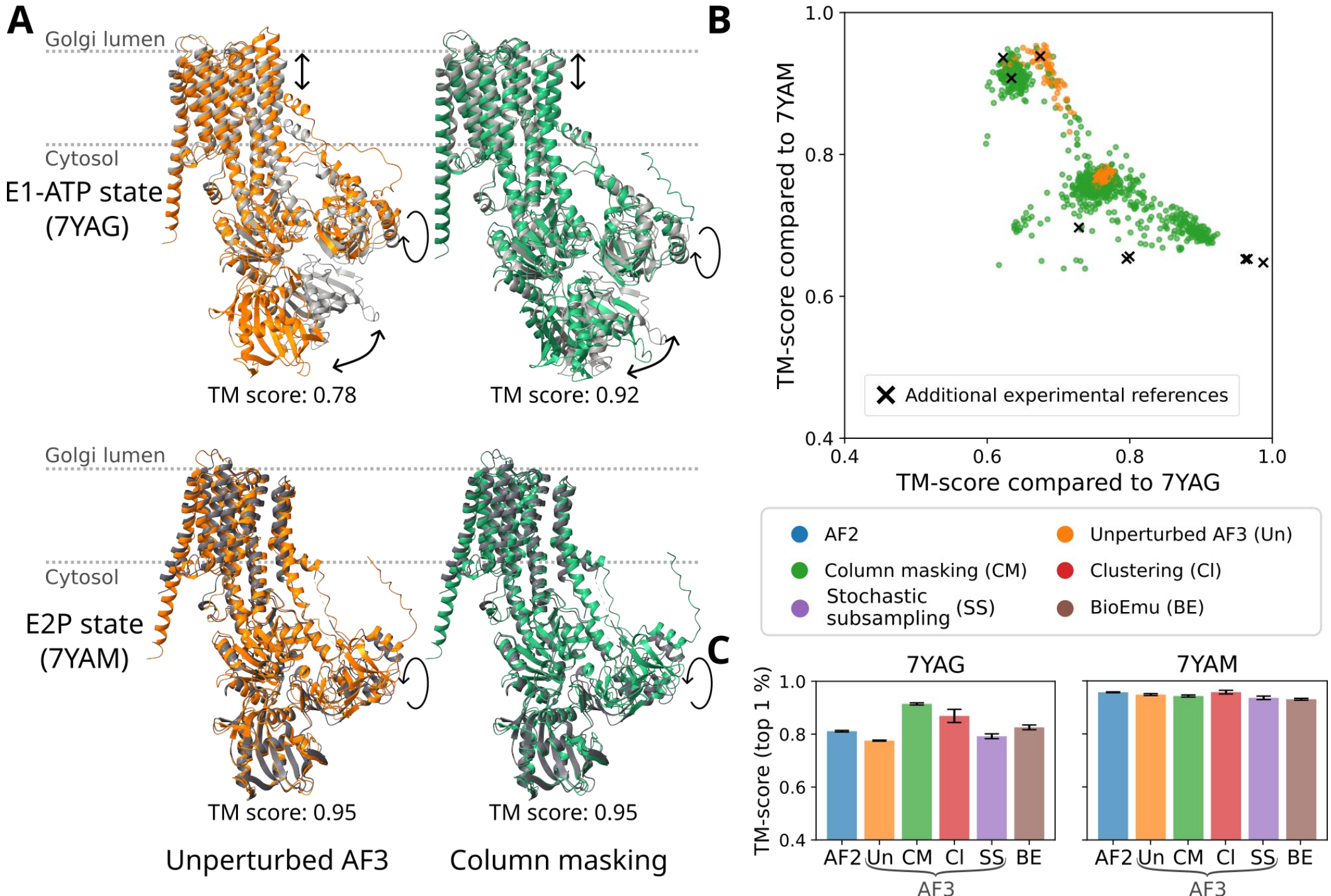
CAN AI METHODS PREDICT CONFORMATIONAL LANDSCAPES IN GENERAL?


A
Protein sequence
A Q D M V S P...
Genetic database
MSA
A Q D M V S P...
G Q D M V S P...
A Q D R V P S...
A Q D A V S S...
G Y D M N I P...
Unperturbed
Stochastic subsampling
Clustering
Column masking
AlphaFold
Structure predictions

B
Similarity to
Top 1 % mean
Top 1 % mean
Similarity to


AAAS
Science
PROTEIN
SHAKE
Generative deep learning powers emulation of protein ensembles p. 700
14 AUGUST 2025
Can synthetic organs and AI replace animal testing? p. 676
Plant volatiles enhance pathogen defense through soil feedback pp. 680 & 698
Diamond is not the only carbon allotrope in rings p. 708


A
Entire dataset
TM-scores alternative state (top 1 %)
1.0
0.8
0.6
0.4
****
****
**
****
ns
AF2
Un
CM
Cl
SS
BE
AF3
B
Structures not in AF3 dataset
ns
**
*
*
ns
AF2
Un
CM
Cl
SS
BE
AF3
AF2
Unperturbed AF3 (Un)
Column masking (CM)
Clustering (Cl)
Stochastic subsampling (SS)
BioEmu (BE)


A
Golgi lumen
Cytosol
E1-ATP state
(7YAG)
TM score: 0.78
TM score: 0.92
Golgi lumen
Cytosol
E2P state
(7YAM)
TM score: 0.95
TM score: 0.95
Unperturbed AF3
Column masking
B
TM-score compared to 7YAM
TM-score compared to 7YAG
Additional experimental references
AF2
Unperturbed AF3 (Un)
Column masking (CM)
Clustering (Cl)
Stochastic subsampling (SS)
BioEmu (BE)
C
7YAG
7YAM
TM-score (top 1 %)
AF2 Un CM Cl SS BE
AF3

USING AI FOR YOUR SIMULATIONS: NEURAL NETWORK POTENTIALS

The Efficiency-Accuracy Tradeoff

Molecular Mechanics (MM)

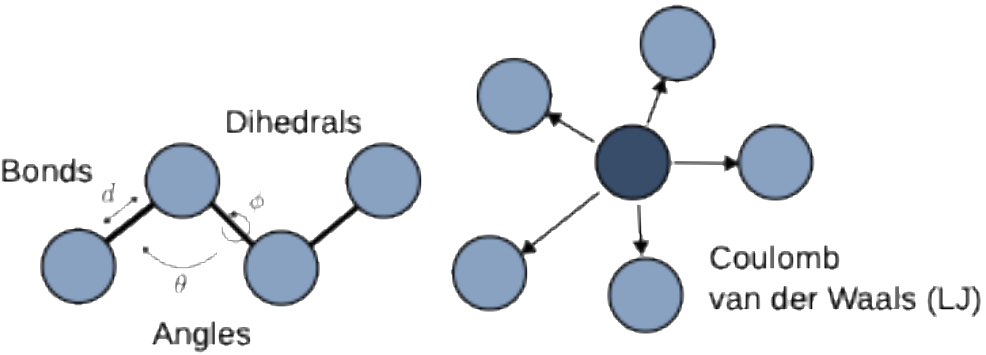
+ computationally efficient
- rigid topology
- non-standard systems need careful validation

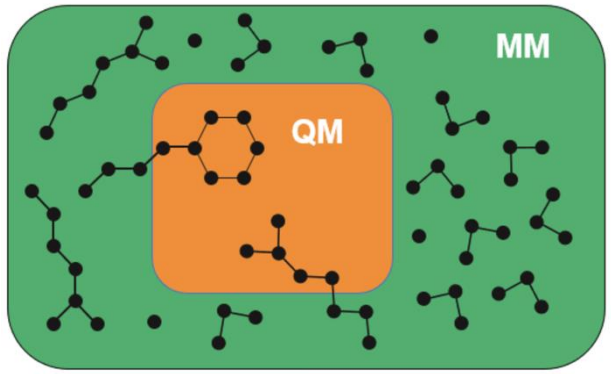
Hybrid QM/MM

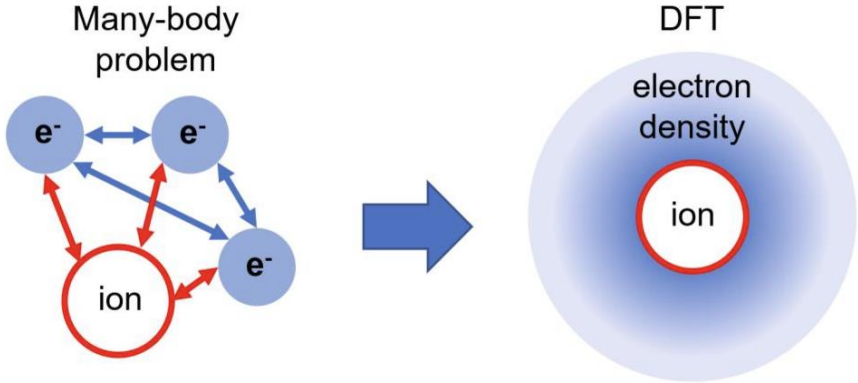
+ best of both worlds
- ... still too expensive!

Quantum Mechanics (QM)

+ first principles, QM-accurate
+ flexible
+ can handle bond breaking
- Computationally very expensive







$$\mathbf{U}(\mathbf{r}^{\mathbf{N}}) = \mathbf{U}_{\mathbf{bonded}} + \mathbf{U}_{\mathbf{Lennard-Jones}} + \mathbf{U}_{\mathbf{Coulomb}}$$

$$\mathbf{H} = \mathbf{H}_{\mathbf{QM}} + \mathbf{H}_{\mathbf{MM}} + \mathbf{H}_{\mathbf{QM-MM}}$$

$$\mathbf{i}\hbar\frac{\mathbf{d}}{\mathbf{dt}}|\Psi(\mathbf{t})\rangle = \hat{\mathbf{H}}|\Psi(\mathbf{t})\rangle$$

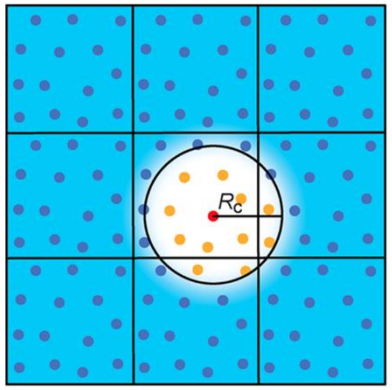
Figure credit:
GROMACS Website / Dmitry Morozov
Sergei Posysaev

Can we do better?

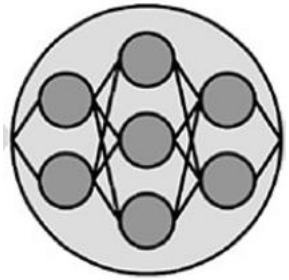
Neural network potentials in GROMACS 2026

Quantum accuracy at (almost) force-field performance

Local chemical environments
Cartesian coordinates $\mathbf{r_i}$
Nuclear charges $\mathbf{Z_i}$



NNP
Parameters $\boldsymbol{\theta}$

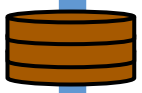


Potential Energy, Forces
(via backprop.)

$$\mathbf{E}(\mathbf{r}^N), \mathbf{F}(\mathbf{r}^N)$$

MD Simulation

Update $\mathbf{r}^N, \mathbf{v}^N$



Electronic structure data
DFT/CCSD(T)

Minimize loss function

$$\mathcal{L} = \frac{1}{N}\sum_i \left(E_i - \hat{E}_i\right)^2$$

Popular NNP architectures:

- **ANI-1x/2x [1,2]**
- **SchNet [3]**
- **MACE [4]**
- **EMLE [5]**

Figures adapted from
Behler, Chem. Rev. 121 (2021)
Unke et al., Chem. Rev. 121 (2021)

The nnpot interface in GROMACS 2026

simulation parameters

md.mdp

```
[...]
nnpot-active       = yes
nnpot-model-file   = model.pt
nnpot-input-group  = ligand
nnpot-model-input1 = atom-positions
nnpot-model-input2 = atom-numbers
nnpot-model-input3 = box
nnpot-model-input4 = pbc
```

starting configuration — conf.gro

molecular topology — topol.top

index groups — index.ndx

wrap_model.py

```
import torch
from NNPArch import PretrainedModel

class GmxModel(torch.nn.module):
    def __init__(self):
        self.model = PretrainedModel()
        self.length_conversion = 10.0   # A->nm
        self.energy_conversion = 2625.5 # Hartree->kJ/mol

    def forward(self, positions, numbers, box, pbc):
        positions = positions * self.length_conversion
        cell      = cell * self.length_conversion
        energy    = self.model(
            positions, numbers, box, pbc)
        )
        return energy * self.energy_conversion

model = GmxModel()
torch.script(model).save('model.pt')
```

NNP model prep: PyTorch

model.pt ← TorchScript serialization

FAST. FLEXIBLE. FREE. GROMACS

gmx grompp

- topology preprocessing: removing bonds, angles, dihedrals, opt. charges in the ML region
- determination of link atoms
- checking model compatibility

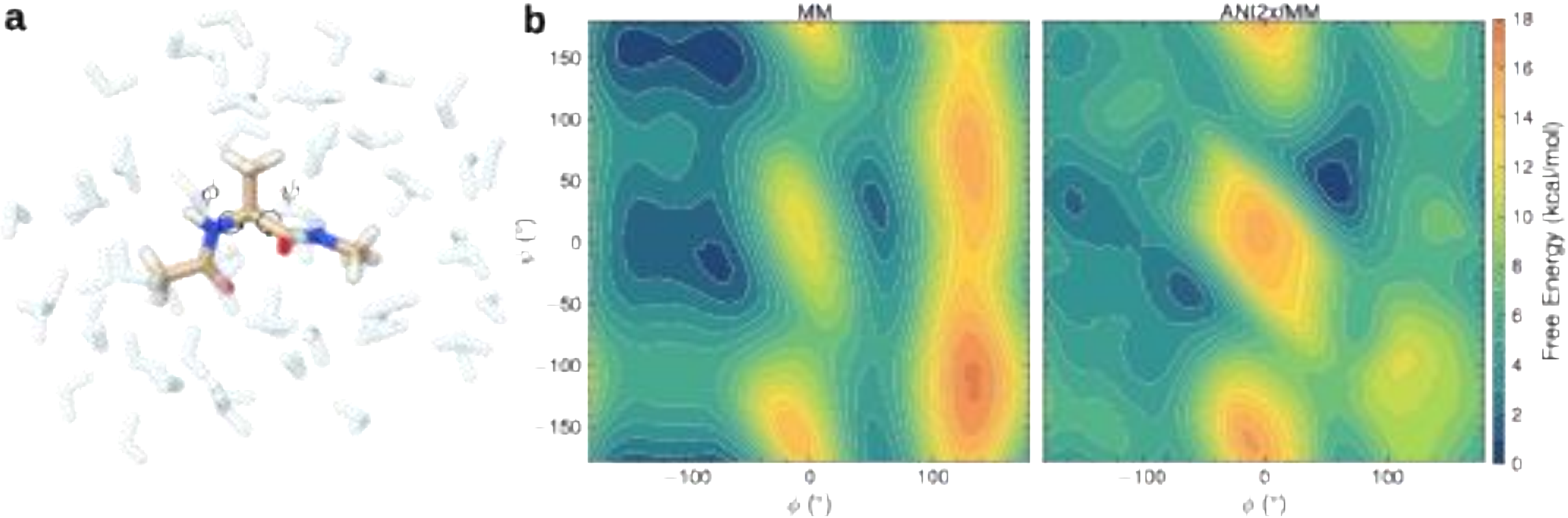
md.tpr →

gmx mdrun

- loading model and opt. torch extensions at runtime
- during MD loop: gather necessary inputs, e.g. positions, atomic numbers, MM region positions & charges, GROMACS pair list
- collect result of model inference, accumulate energy and force buffers

Actual MD simulation

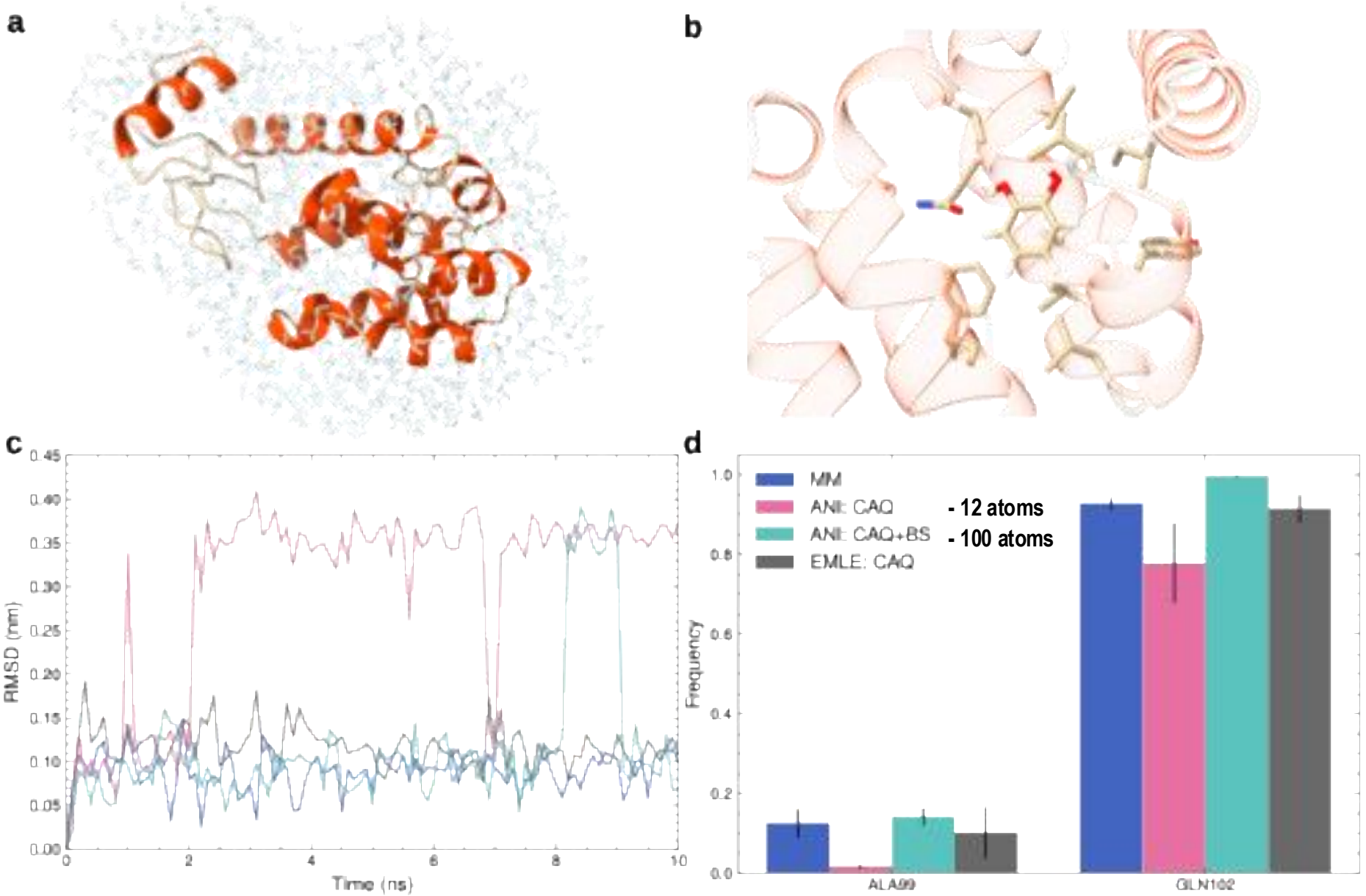
ML/MM Enhanced Sampling with GROMACS AWH



[Unpublished data]

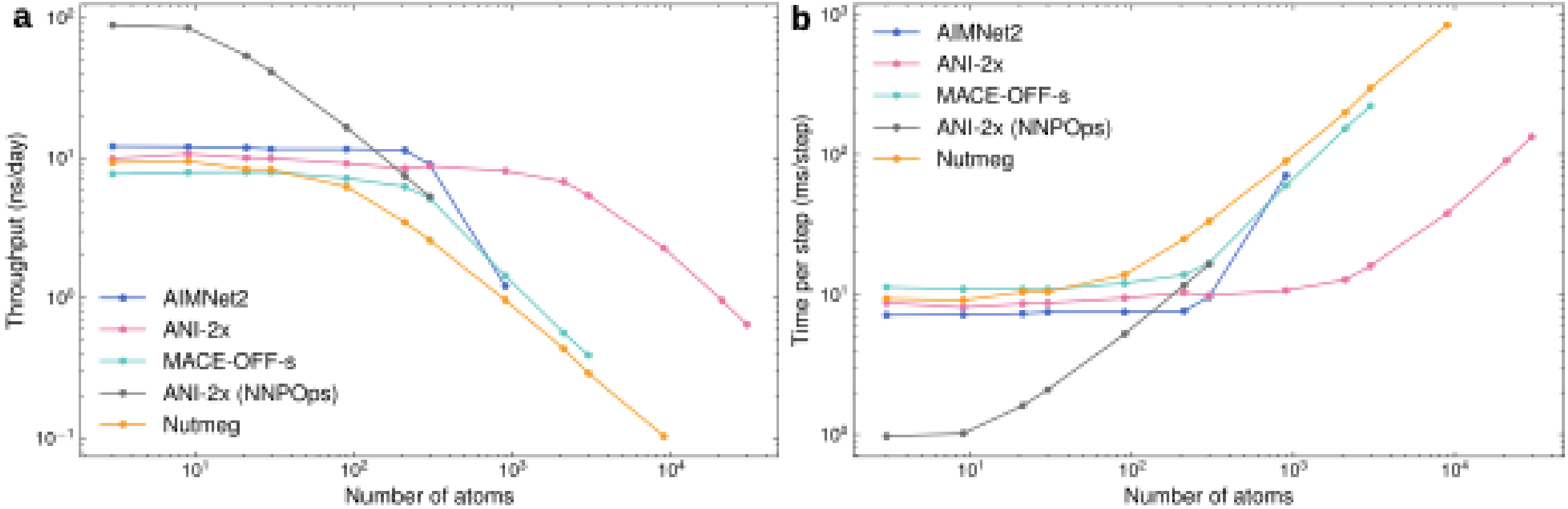
Protein-Ligand Systems: ML region size matters

Catechol bound to
T4 Lysozyme
L99A/M102Q



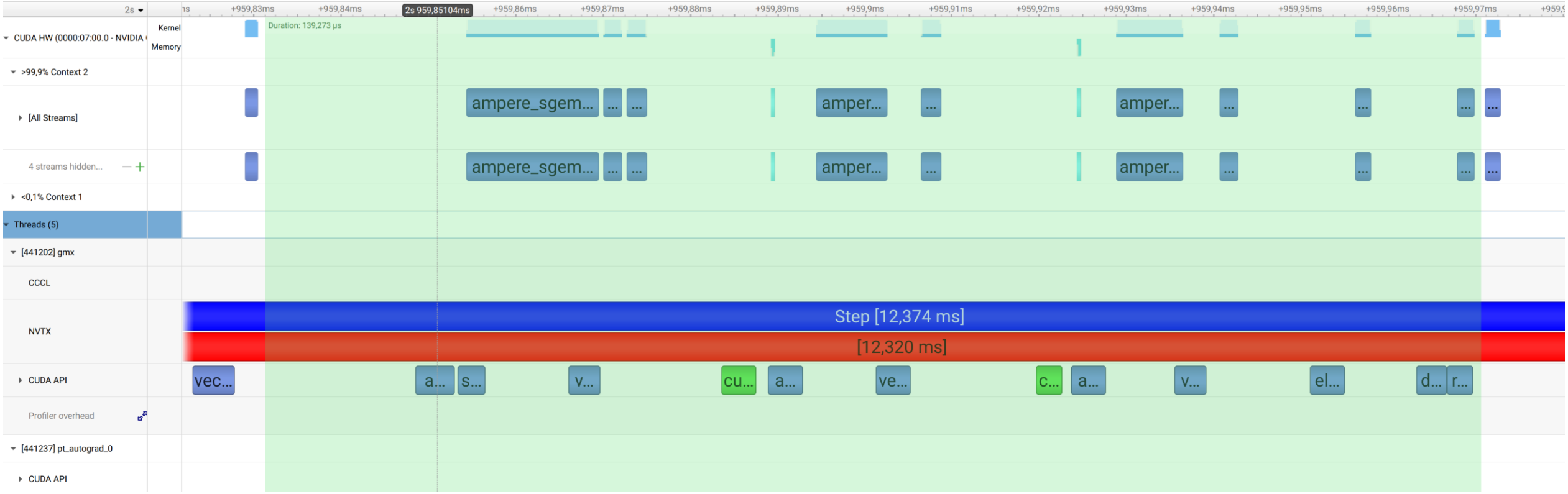
Performance Scaling

NVIDIA RTX3070
1 fs time step
32-bit floating point



Smaller systems are dominated by libtorch latency

Libtorch is often very inefficient – room to improve



AI is not a threat to HPC - but our best friend!

- You don’t need to train models yourself - just as you don’t need to write all code
- Generative AI is amazing as a hypothesis/model generator
- It is less amazing as final arbiter of physical truth - combine with hard data
- AlphaFold/Chai-1 and friends are great at generating many models,
  but the ensembles are *not* anywhere near a correct Boltzmann distribution
- Beware of the hype - models are usually not as good in the real world
- But … the speed of improvement is insane
- Being creative in using data often beats developing a really advanced AI model
- The vast majority of HPC users employ codes from others – use AI from others too!

Acknowledgments

GROMACS: Szilárd Páll, Berk Hess, Andrey Alekseenko, Artem Zhmurov, Umair Sadiq, Lukas Müllender, Vedran Miletic, Cathrine Bergh, Yang Zhang, Petter Johansson and numerous other contributors
Vendor co-design: Alan Gray, Mahesh Doijade (Nvidia); Mark Abraham, Roland Shulz (Intel); Paul Bauer (AMD)
Molecular simulation: Magnus Lundborg, Tatiana Shugaeva, Sebastian Wingbermühle, Alessandra Villa, Farzaneh Jalalypour, Nandan Haloi, Anton Jansen, Samuel Eriksson Lidbrink, Rebecca Howard

Szilárd Páll
KTH

Andrey Alekseenko
KTH

Berk Hess
KTH

Take-home messages:

- SW is becoming like HW: No silver bullets remaining
- Constant fight: balance latency with GPU speedup
- MNNVL can be fantastic for traditional HPC apps
- GROMACS-2026 can do neural network potential simulations
- But the really big wins come from replacing the entire iterative approach with AI
- AI has a ton of important applications, but suffers from hype and lack of critical testing