Los Alamos NATIONAL LABORATORY

The mathematics and algorithmics of long-time atomistic simulations

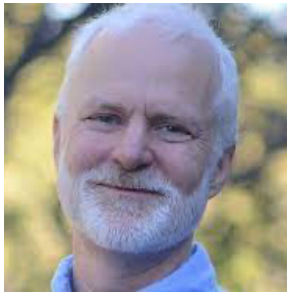
Danny Perez

Theoretical Division T-1

LA-UR-21-21287

Collaborators/Acknowledgements

- **Method development:** Arthur Voter
- **Mathematics:** Tony Lelièvre, Claude Le Bris, Mitch Luskin, David Aristoff
- **Code:** The EXAALT ECP team
- **Funding:** DOE BES, ECP; LANL LDRD
- **Computing:** LANL IC, NERSC





Los Alamos NATIONAL LABORATORY

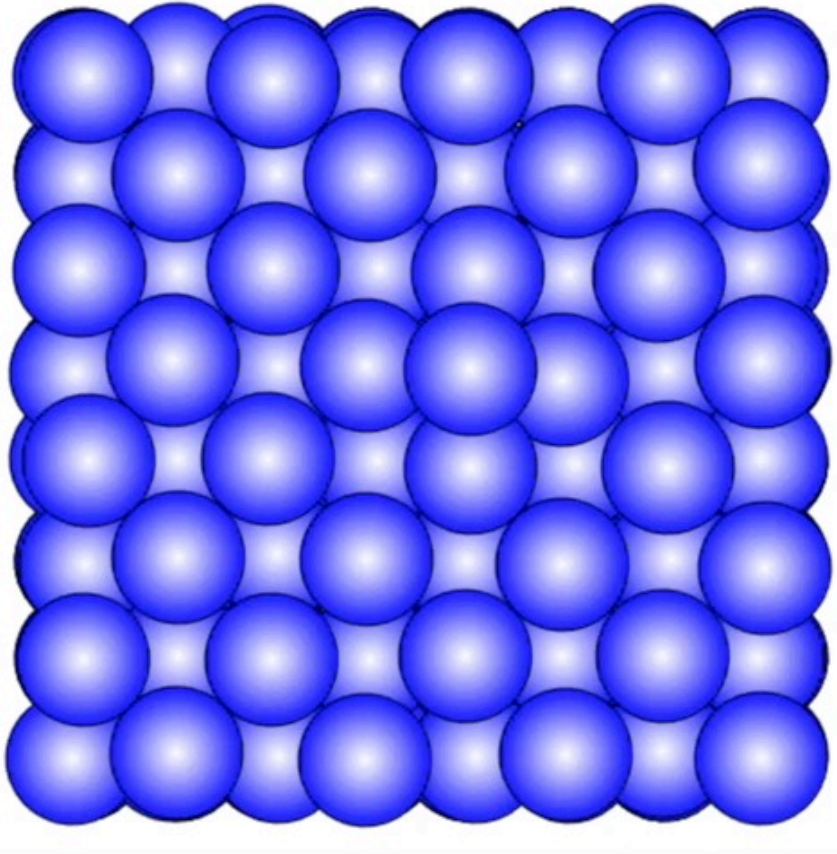
ECP EXASCALE COMPUTING PROJECT

What is Molecular Dynamics?

- **MD:** numerical solution of classical atomic equations of motion
- **This talk: Langevin dynamics**

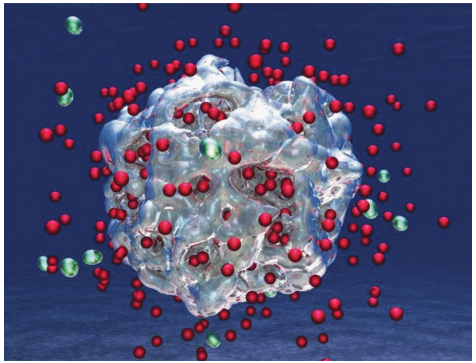
$$m\ddot{x} = -\nabla V(x) - \gamma\dot{x} + \sqrt{2k_B T\gamma}\xi(t)$$

- $V(x)$ is material-specific. Usually an empirical approximation to Schrodinger's equation
- Ubiquitous: >1M hit on Google scholar



Why Molecular Dynamics?

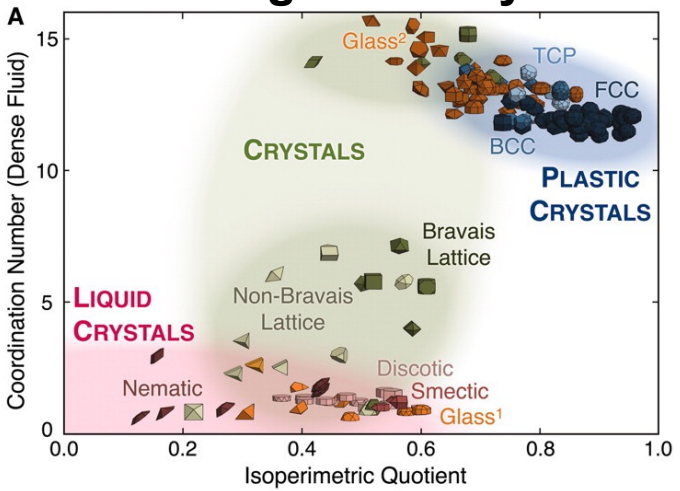
H production in Water/Al (Quantum MD)



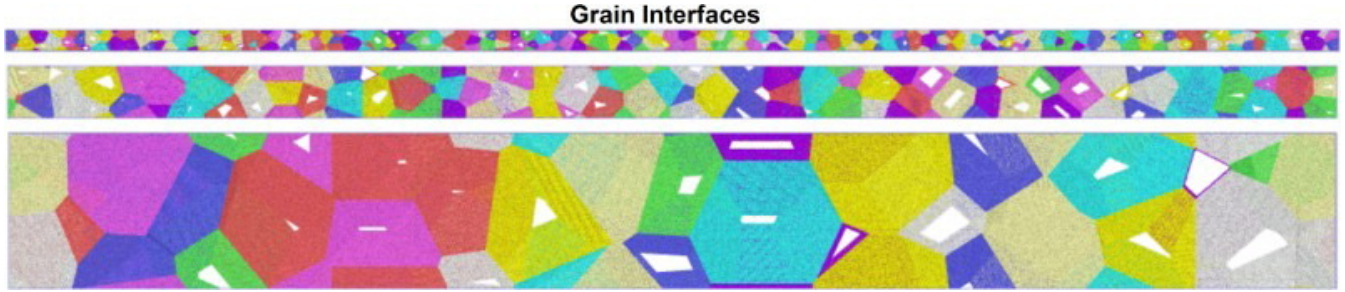
K. Shimamura et al., “Hydrogen-on-Demand Using Metallic Alloy Nanoparticles in Water,” Nano Letters, vol. 14, no. 7,2014, pp. 4090–4096

Glotzer, Sharon C., and Michael J. Solomon. "Anisotropy of building blocks and their assembly into complex structures." *Nature materials* 6.8 (2007): 557-562.

Phases of granular systems



Shock Response of coarse grained explosives



Mattox, Timothy I., et al. "Highly scalable discrete-particle simulations with novel coarse-graining: accessing the microscale." *Molecular Physics* 116.15-16 (2018): 2061-2069.

A brief history of MD

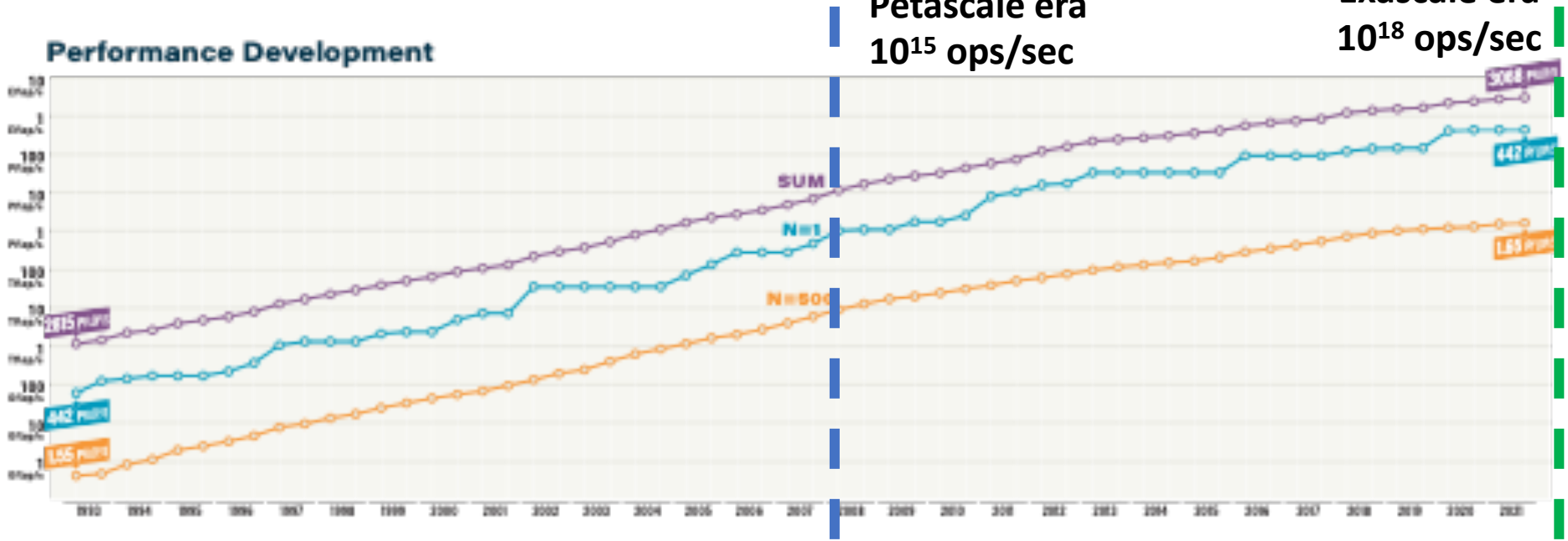
Correlations in the Motion of Atoms in Liquid Argon*

A. RAHMAN
Argonne National Laboratory, Argonne, Illinois
(Received 6 May 1964)

A system of 864 particles interacting with a Lennard-Jones potential and obeying classical equations of motion has been studied on a digital computer (CDC 3600) to simulate molecular dynamics in liquid argon at 94.4°K and a density of 1.374 g cm^{-3}. The pair-correlation function and the constant of self-diffusion are found to agree well with experiment; the latter is 15% lower than the experimental value. The spectrum of the velocity autocorrelation function shows a broad maximum in the frequency region $\omega=0.25(k_BT/\hbar)$. The shape of the Van Hove function $G_s(r,t)$ attains a maximum departure from a Gaussian at about $t=3.0\times10^{-12}$ sec and becomes a Gaussian again at about 10^{-11} sec. The Van Hove function $G_d(r,t)$ has been compared with the convolution approximation of Vineyard, showing that this approximation gives a too rapid decay of $G_d(r,t)$ with time. A delayed-convolution approximation has been suggested which gives a better fit with $G_d(r,t)$; this delayed convolution makes $G_d(r,t)$ decay as t^4 at short times and as t at long times.

864 atoms
20 ps

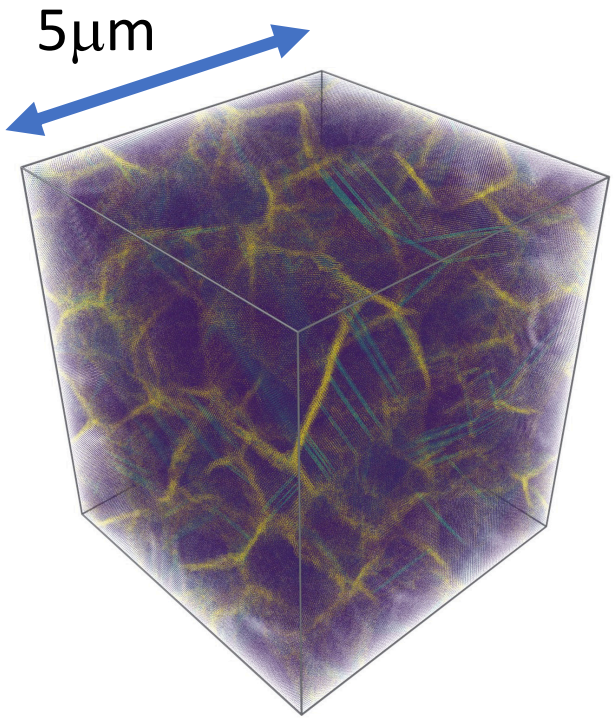
The Evolution of Supercomputing



http://top500.org

A brief history of MD

- 1959: 32 atoms (Adler et al.)
- 1964: 864 atoms (Rahman)
- …
- 1996: 100 million atoms (Beazley et al.)
- 2000: 5 billion atoms (Roth et al.)
- 2006: 320 billion atoms (Kadau et al.)
- 2008: 1 trillion atoms (Germann et al.)
- 2013: 4 trillion atoms (Eckhardt et al.)
- 2019: 20 trillion atoms (Tchipev et al.)



A brief history of MD

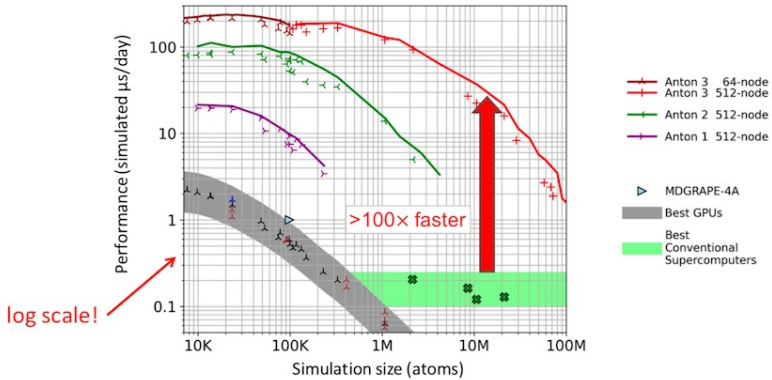
Collective Solvation and Transport at Tetrahydrofuran–Silica Interfaces for Separation of Aromatic Compounds: Insight from Molecular Dynamics Simulations

Fei Liang, Jing Ding, and Shule Liu*

Cite This: https://dx.doi.org/10.1021/acs.langmuir.0c03077 Read Online

Simulation time = 44 ns

MD performance



D E Shaw Research

20

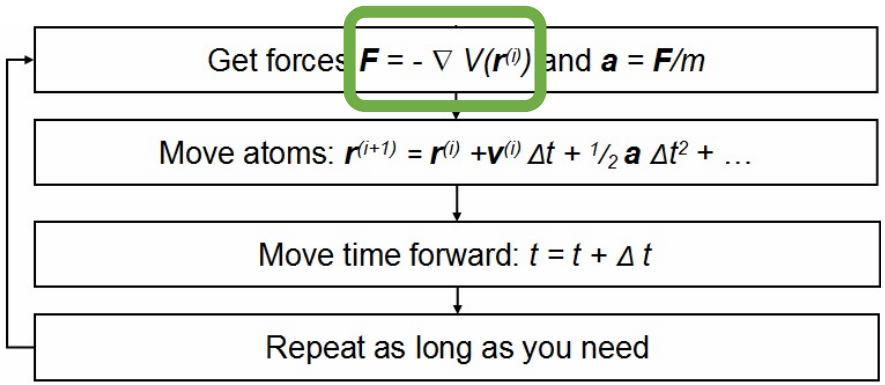
Simulation time = 100 us

The Evolution of Super-Computing

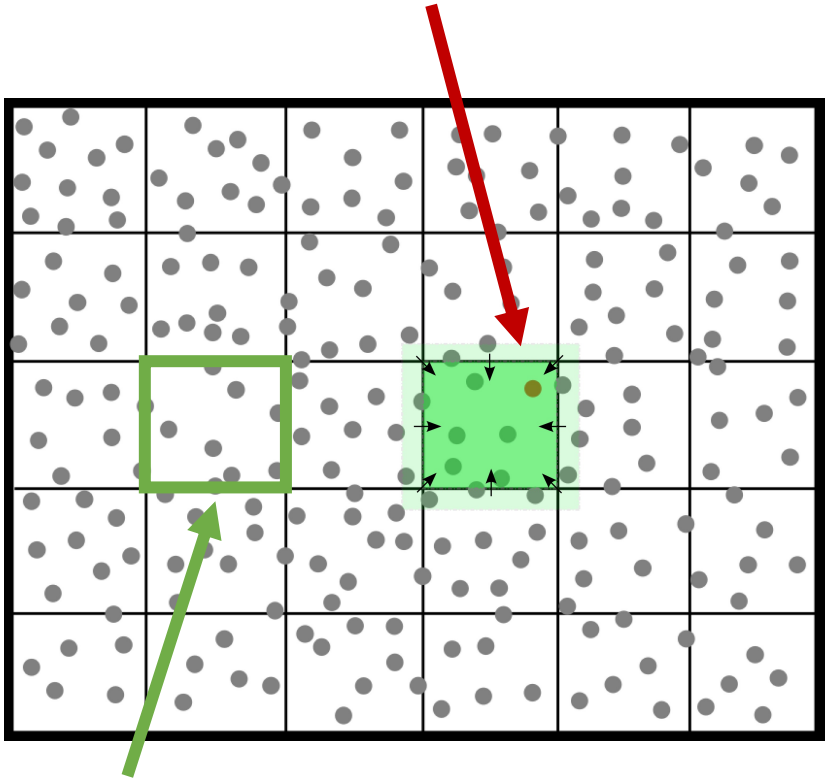
Rank	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	**Supercomputer Fugaku** - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,299,072	415,530.0	513,854.7	28,335
2	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM DOE/SC/Oak Ridge National Laboratory United States	2,414,592	148,600.0	200,794.9	10,096
3	**Sierra** - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1,572,480	94,640.0	125,712.0	7,438
4	**Sunway TaihuLight** - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway, NRCPC National Supercomputing Center in Wuxi China	10,649,600	93,014.6	125,435.9	15,371
5	**Tianhe-2A** - TH-IVB-FEP Cluster, Intel Xeon E5-2692v2 12C 2.2GHz, TH Express-2, Matrix-2000, NUDT National Super Computer Center in Guangzhou China	4,981,760	61,444.5	100,678.7	18,482

Parallel MD

Most cycles spent here

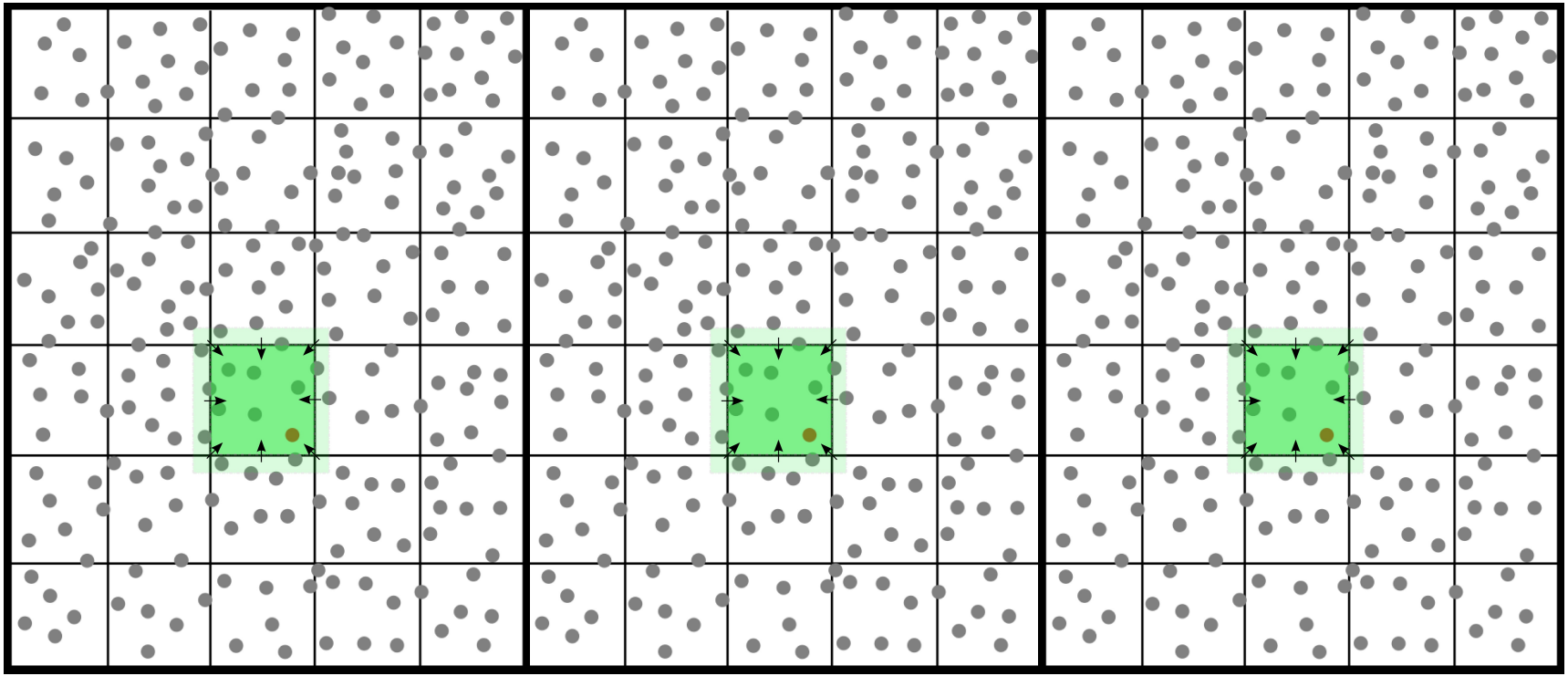


Communication required at every step

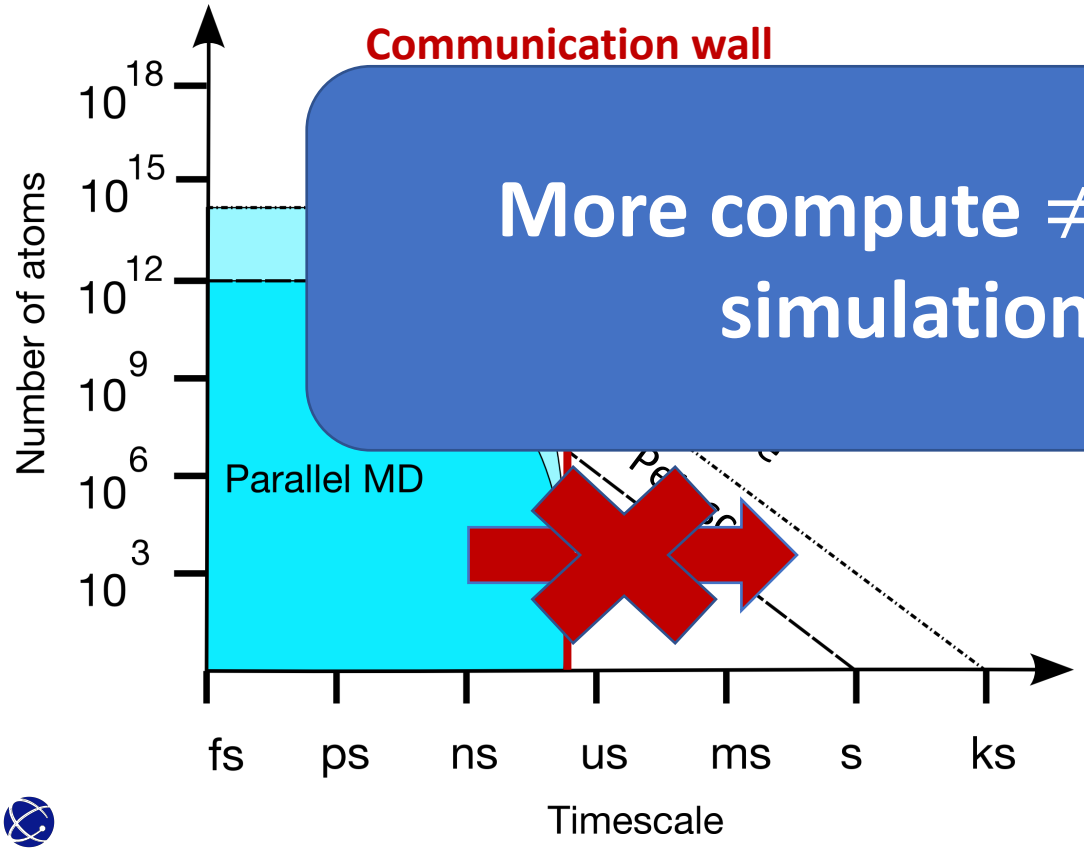


Each processor owns its domain

MD weak-scales

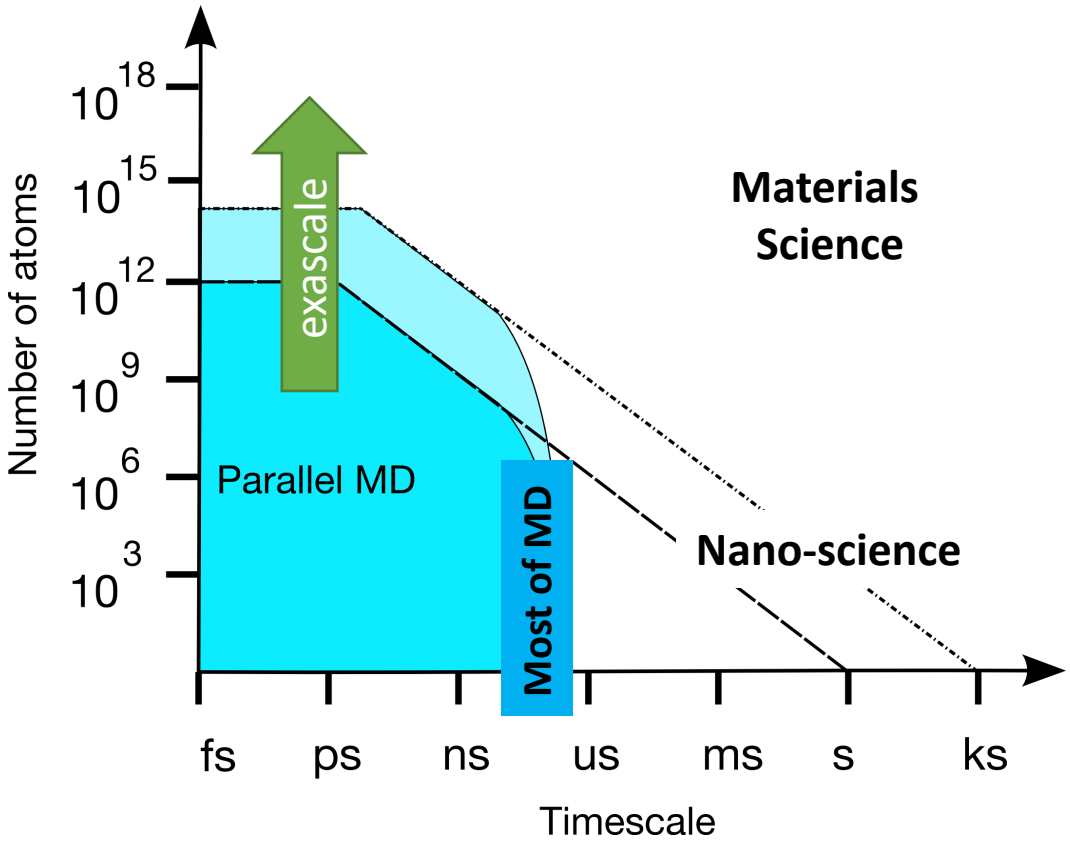


MD does not strong-scale

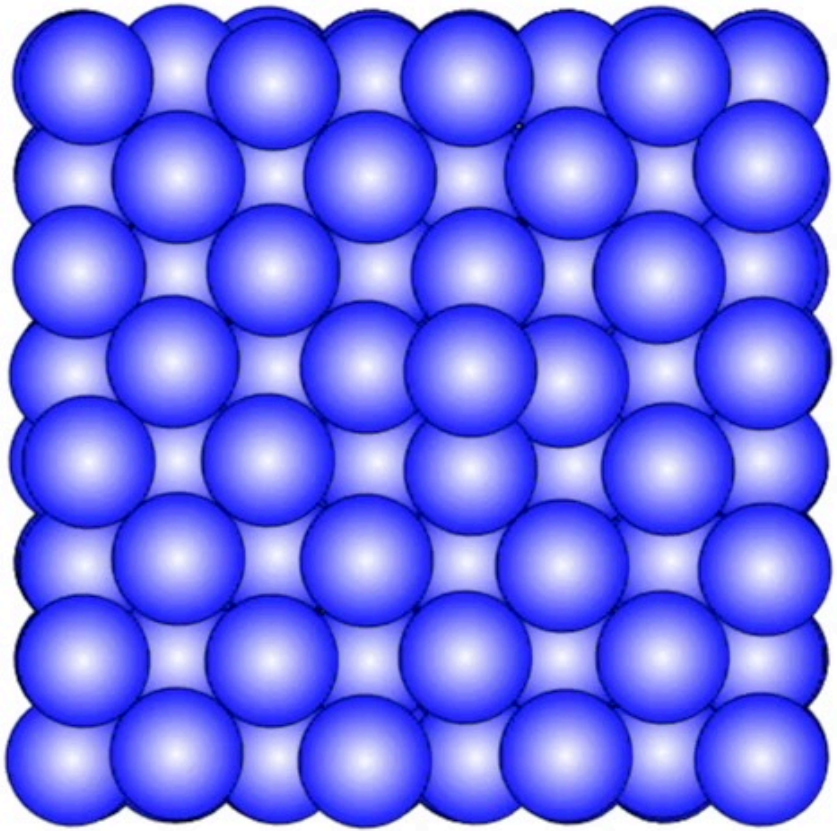


More compute ≠ longer simulations

The prospect for MD at the exascale



Metastability

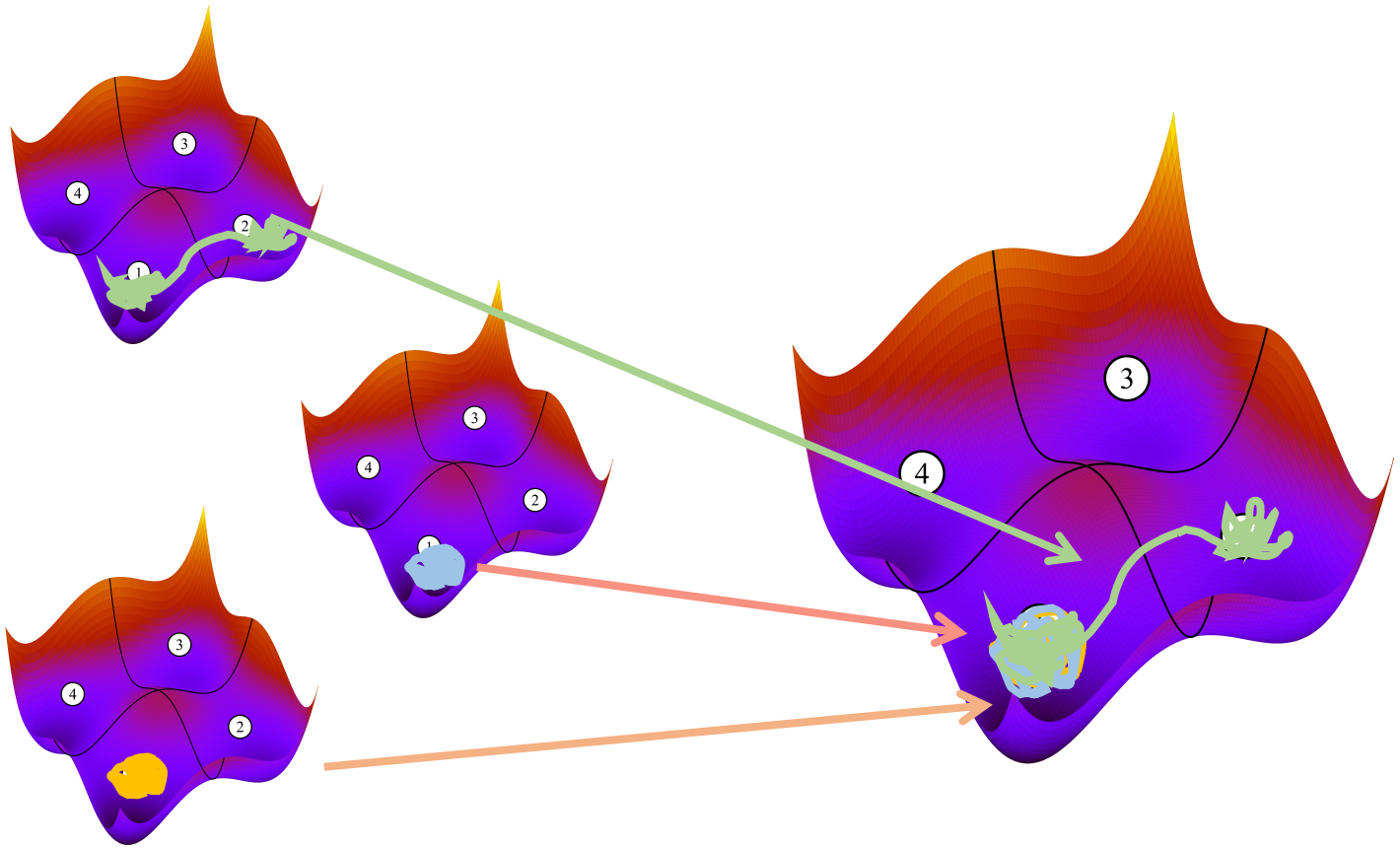


- For materials away from melting:
  - Fast vibrations/fluctuations (ps)
  - Slow conformational changes (ns-s)

- Short simulations are often **not informative** of long-time behavior

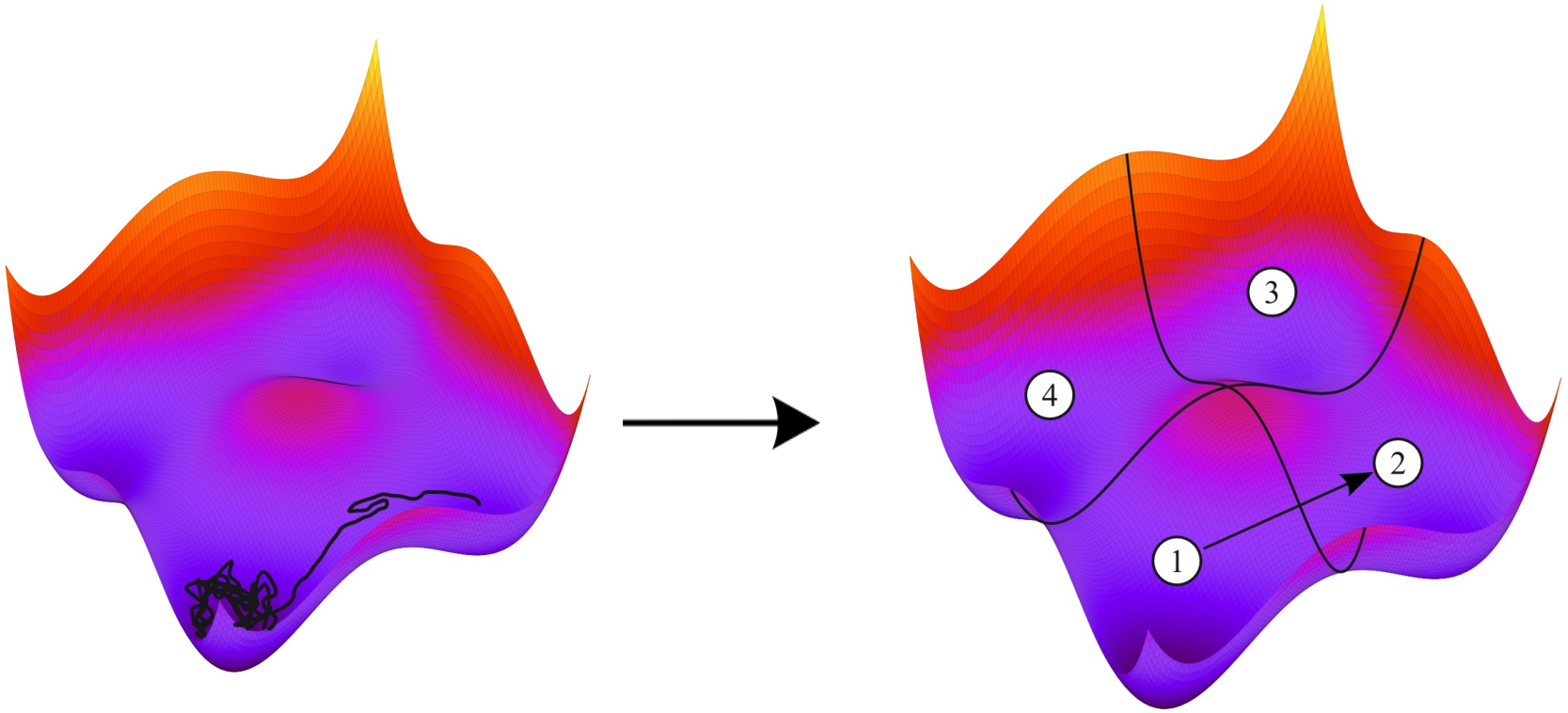
Theme of today's talk:
How can we leverage this separation of timescales to **parallelize** the dynamics **in time** instead of space

Can we parallelize over ~~space~~ time?



[Perez, Cubuk, Waterland, Kaxiras, Voter, JCTC 12, 18 (2016)]

State-to-state dynamics

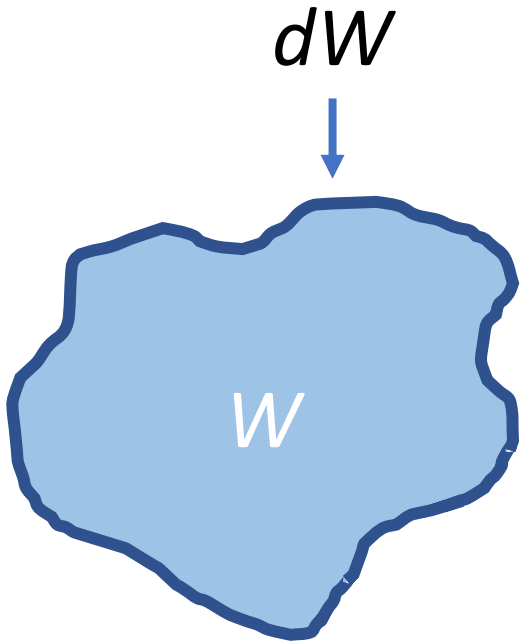


Goal is to generate a single *statistically correct* state-to-state trajectory

State-to-state dynamics

Need to capture **transition statistics**:

- Distribution of first-escape times from *W*
- Distribution of first-escape points on *dW*



Key Concept: Quasi-stationary Distribution (QSD)

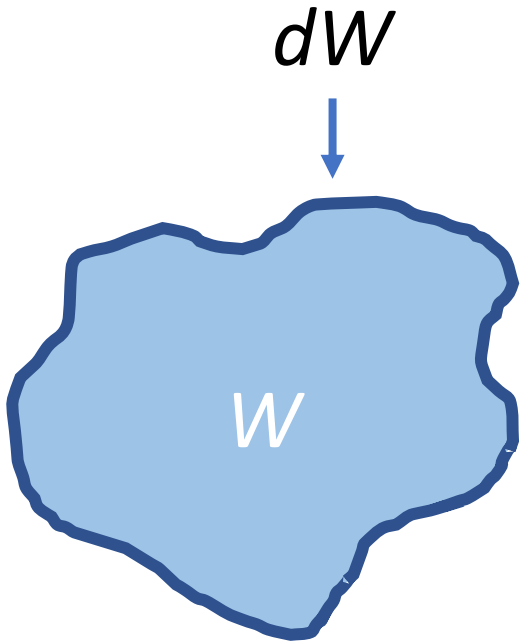
$$\nu(A) = \frac{\int_W \mathbb{P}(X_t^x \in A,\, t < T_W^x)\, d\nu}{\int_W \mathbb{P}(t < T_W^x)\, d\nu}.$$

If X_0 is distributed according to *QSD*, then, conditionally on not having left W up to time t, X_t is still distributed according to *QSD*

QSD for Langevin dynamics

In the following:

- Overdamped Langevin dynamics
- Absorbing boundary conditions on *dW*
- Generator has eigenvalues $0 > -\lambda_1 > -\lambda_2 \geq -\lambda_3$ ...
- **QSD is eigenfunction $u_1(X)$ of generator corresponding to λ_1**



Most of the following also applies to other dynamics, if:

- QSD exists
- QSD is unique
- Convergence to the QSD is fast

QSD for Langevin dynamics

$$\frac{\partial \rho}{\partial t} = L\rho \text{ on } W$$
$$\rho = 0 \text{ on } \partial W$$

With $L = -\nabla V \cdot \nabla + \beta^{-1}\Delta$

Then:

$$\rho(X,t) = \sum_k e^{-\lambda_k t}\, c_k^0 u_k(X)$$

For $t > (\lambda_2 - \lambda_1)^{-1}$ and conditional on not having escaped,

$$\hat{\rho}(X,t) \cong u_1(X) + O(e^{-(\lambda_2-\lambda_1)t})$$

Properties of the QSD

Rate of memory loss

- The QSD of W is **unique**
- Convergence to the QSD is **exponential** with rate $(\lambda_2 - \lambda_1)$

From the QSD:

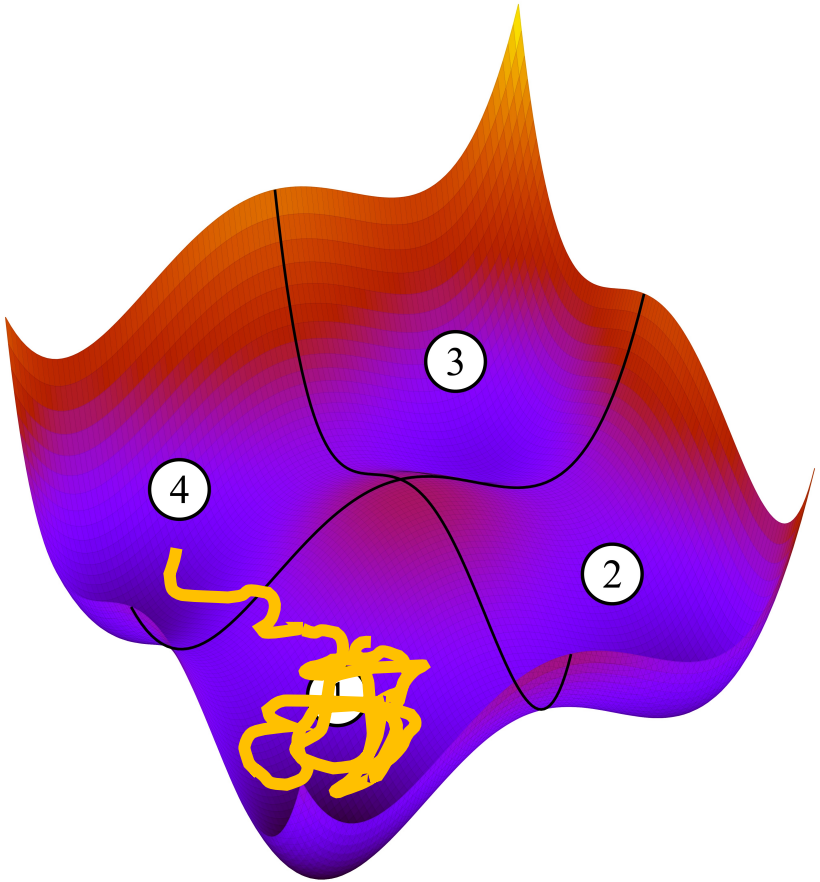
- First escape time is random and exponentially distributed with rate λ_1
- First escape point is random and uncorrelated with escape time

This is true for any state definition!

Does not depend on history before reaching the QSD

Overdamped Langevin: [Le Bris, Lelievre, Luskin, and DP, MCMA 18, 119 (2012)]

Langevin: [Lelievre, Ramil, Reygner, arXiv:2101.11999]



After only a short time in the state, the next escape time/location distribution is a complex function of the entry point

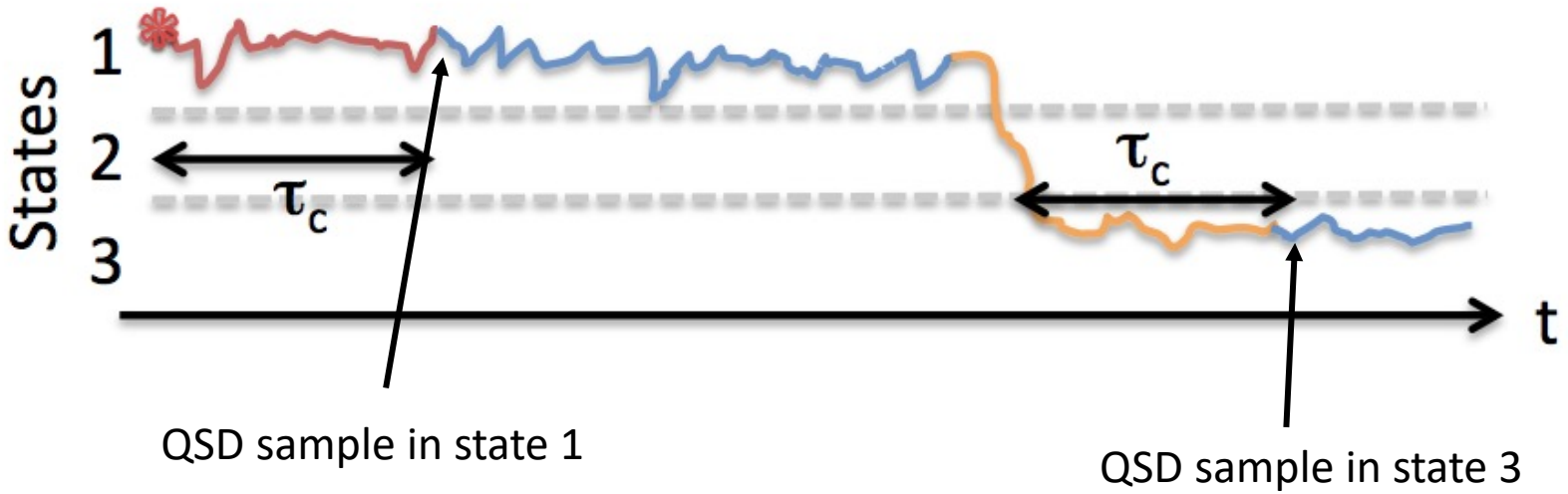
After spending $t_c > (\lambda_2 - \lambda_1)^{-1}$ in W, the next escape from W becomes *Markovian

All trajectories that spent $t_c > (\lambda_1 - \lambda_2)^{-1}$ in W are statistically equivalent with respect to how and when they will leave W*

* Up to an exponentially small error in t_c

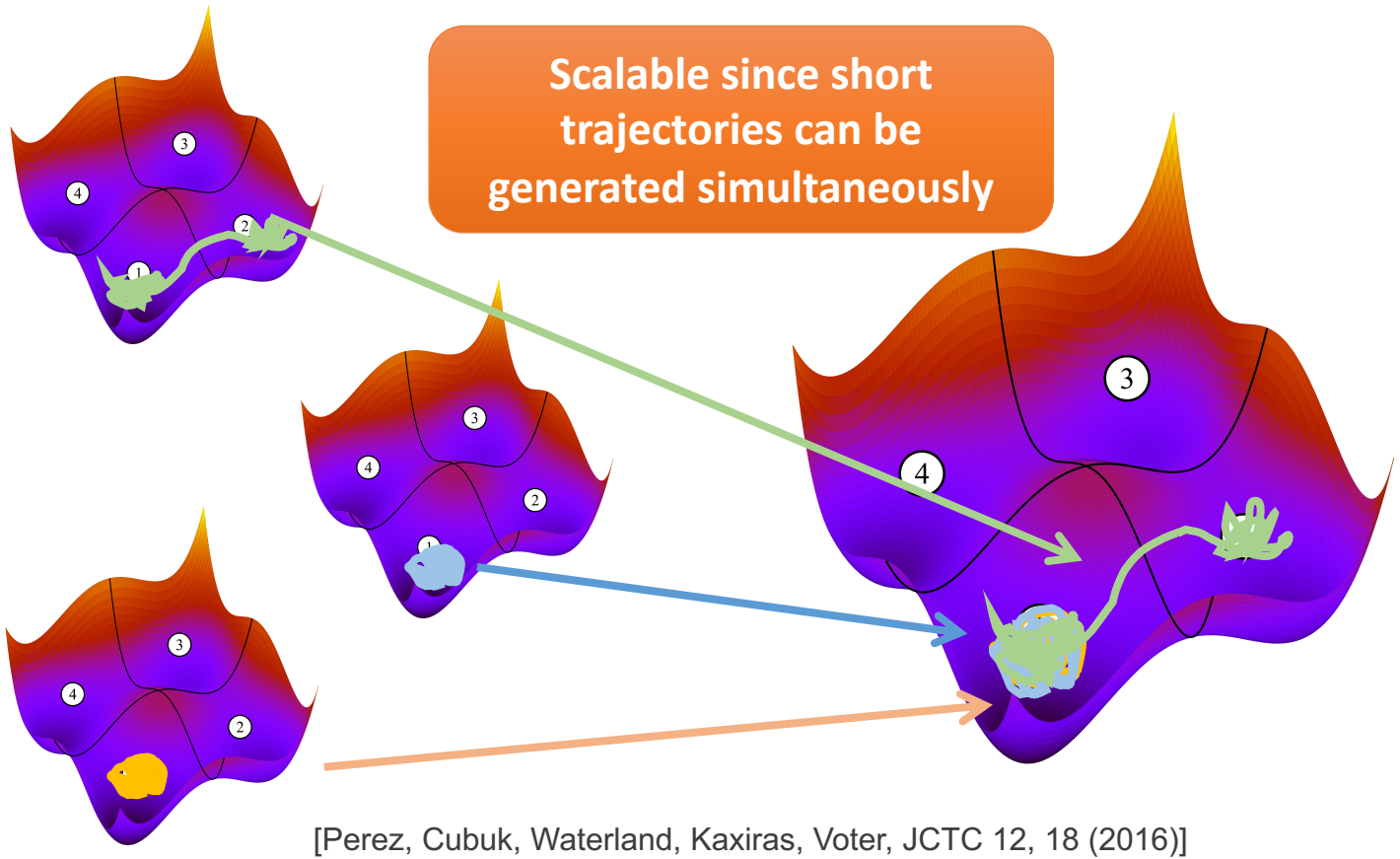
Trajectory building block

Define a **segment** as a trajectory that spent at least t_c in the same state before its beginning and before its end.



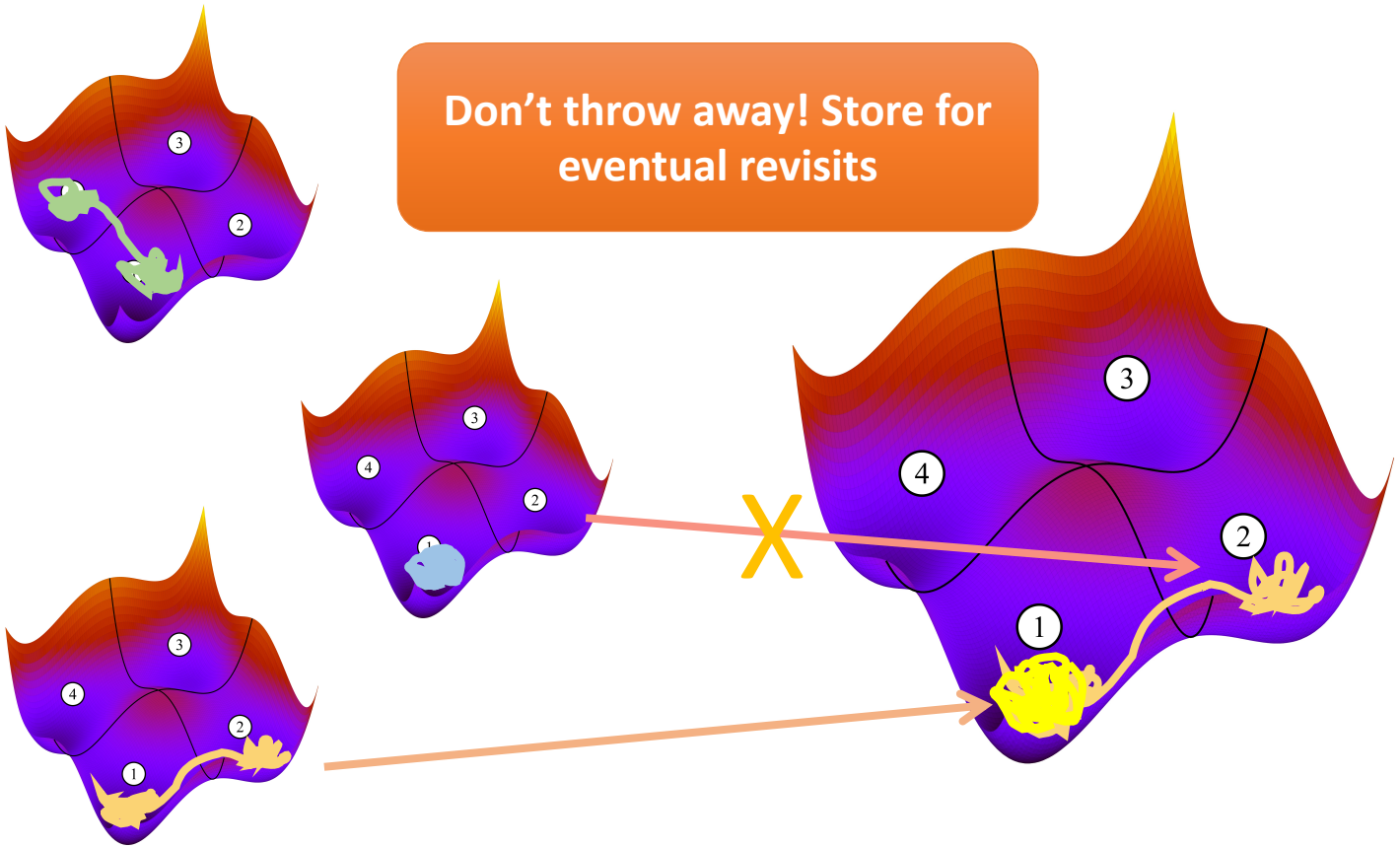
* Up to an exponentially small error in t_c

Parallel Trajectory Splicing (ParSplice)

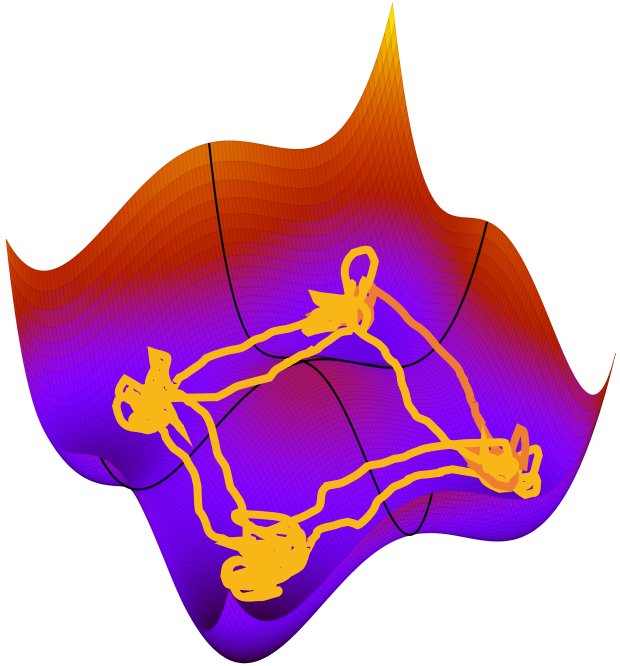


[Perez, Cubuk, Waterland, Kaxiras, Voter, JCTC 12, 18 (2016)]
[Aristoff, SIAM/ASA Journal on Uncertainty Quantification 7, no. 2 (2019): 685-719]

Bookkeeping

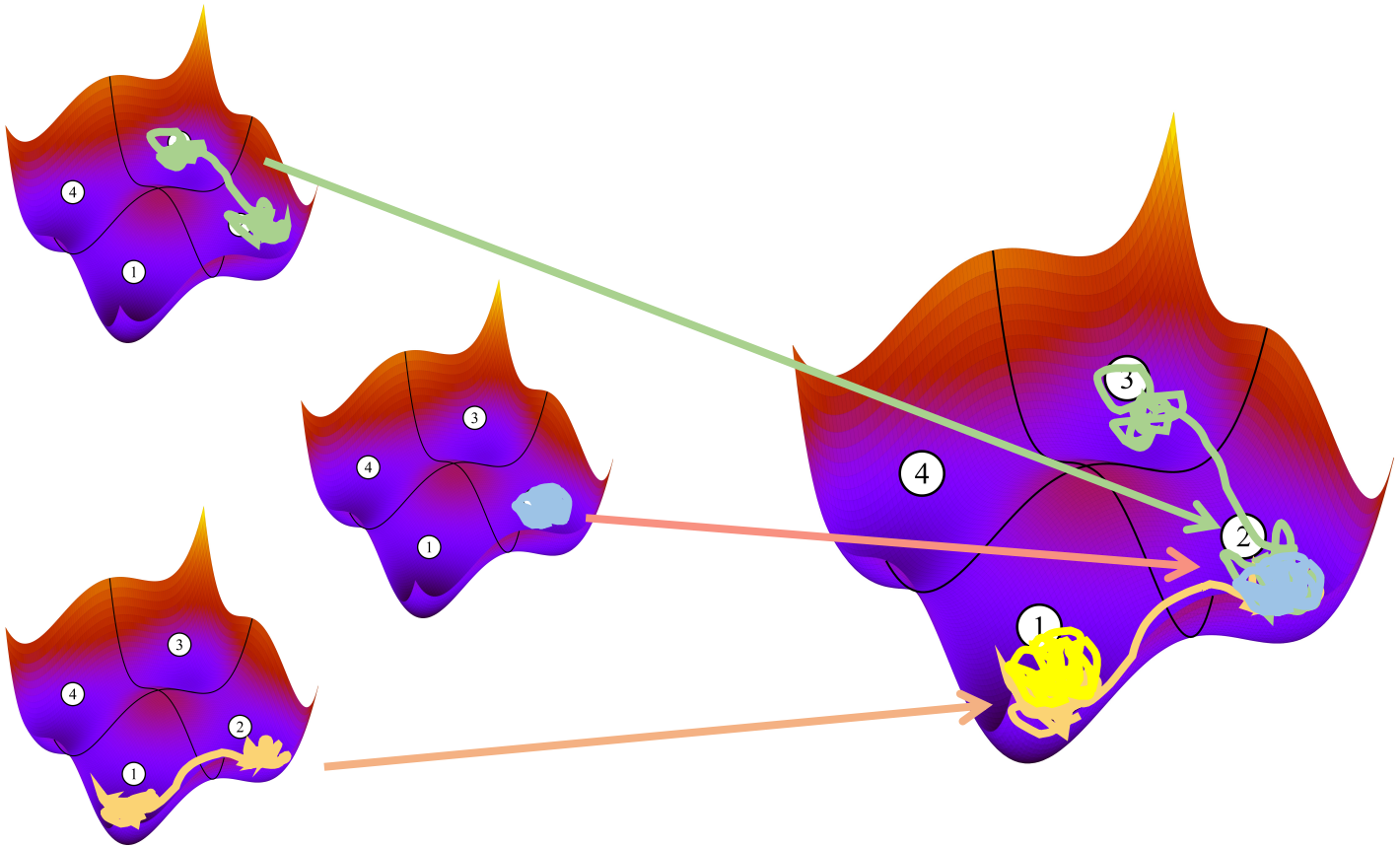


Super-basins

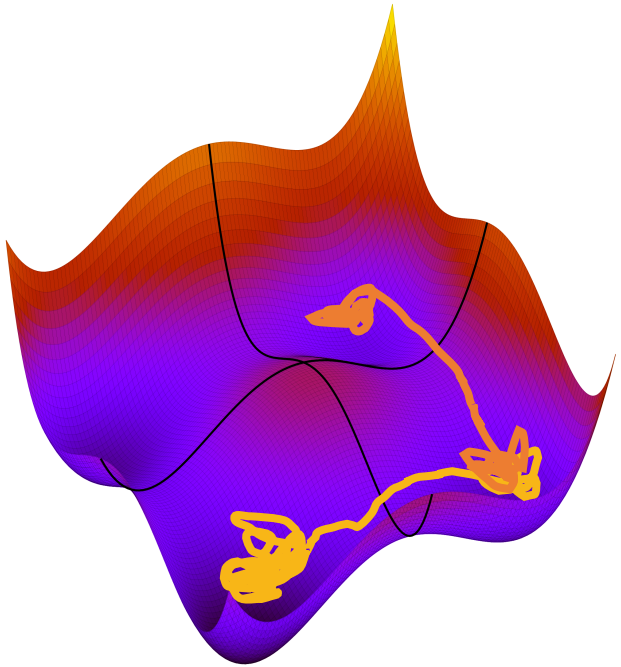


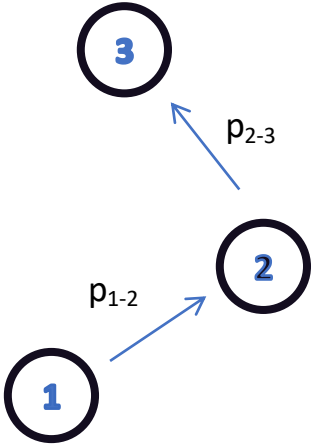
Revisits are extremely common!

Speculation



Statistical oracle

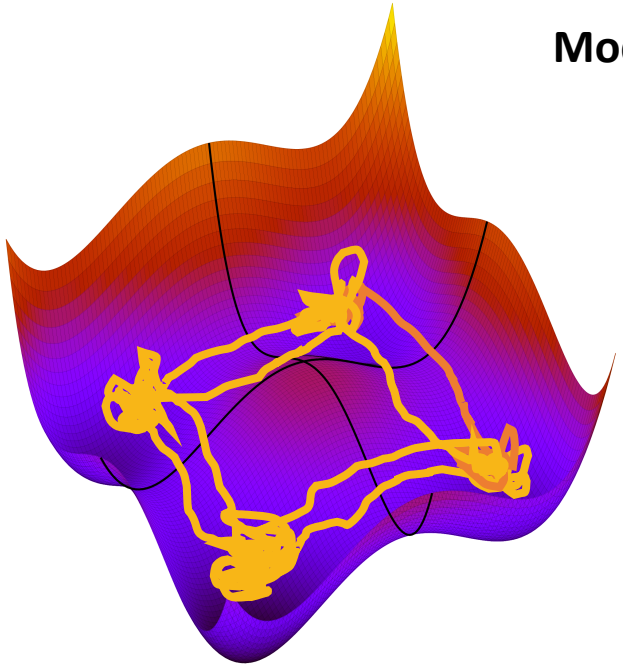


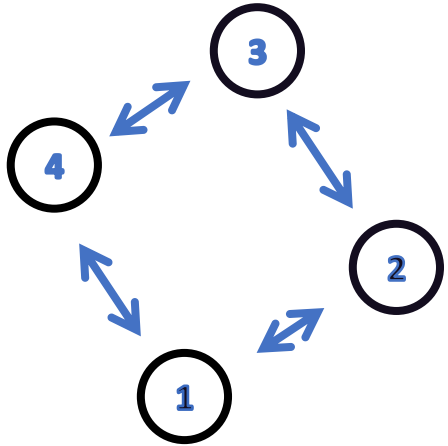


Statistical oracle

We use this model to speculate where the trajectory will be in the future

Model quality affects efficiency, but *not* accuracy





See A. Garmon, DP, MSMSE 28, 065015 (2020) for more detail on model construction
See A. Garmon, V. Ramakrishnaiah, DP, arXiv:2010.11792, for use of model for resource allocation

Maximum Parallel Speedup

Given an infinitely large computer, the (wall-clock) speedup vs MD is:

With perfect oracle:

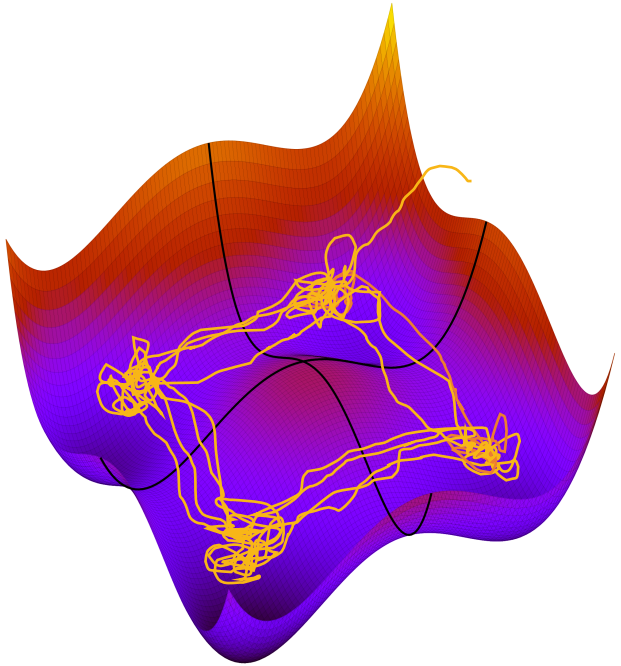
$\sim$ infinity

If trapped in a super-state:

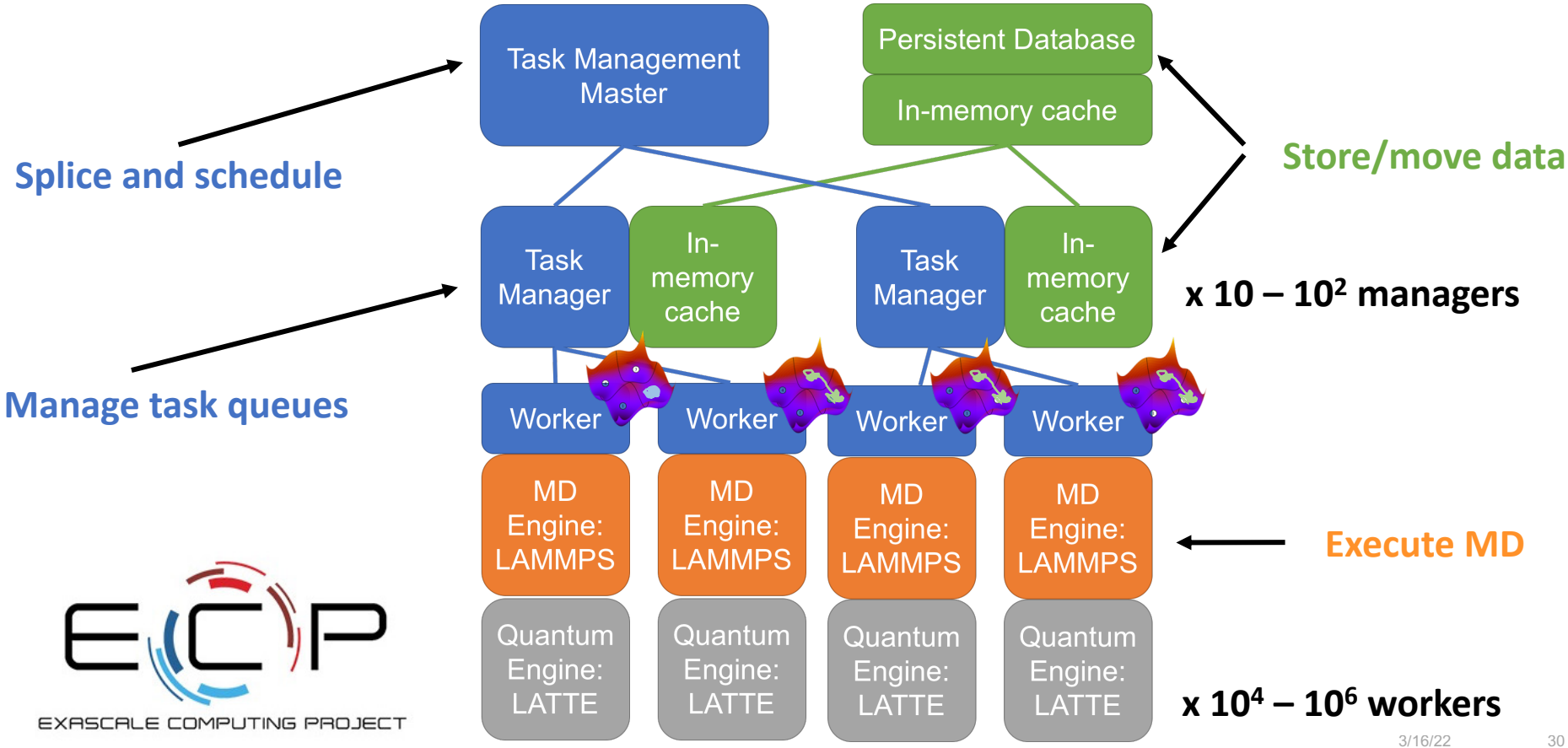
$\sim \tau^{ss}_{esc} / \tau_c >> \tau^{i}_{esc} / \tau_c$

If trajectory never revisits states:

$\sim \tau^{i}_{esc} / \tau_c$



Implementation in the EXAALT code



EXASCALE COMPUTING PROJECT

Shape fluctuations in nanoclusters

- Properties of nanoclusters are sensitive to shapes and sizes
- Some small nanoparticles don't have well defined shapes; continuously transform between different conformations
- This affects their physical/chemical properties
- How do these shape changes occur?

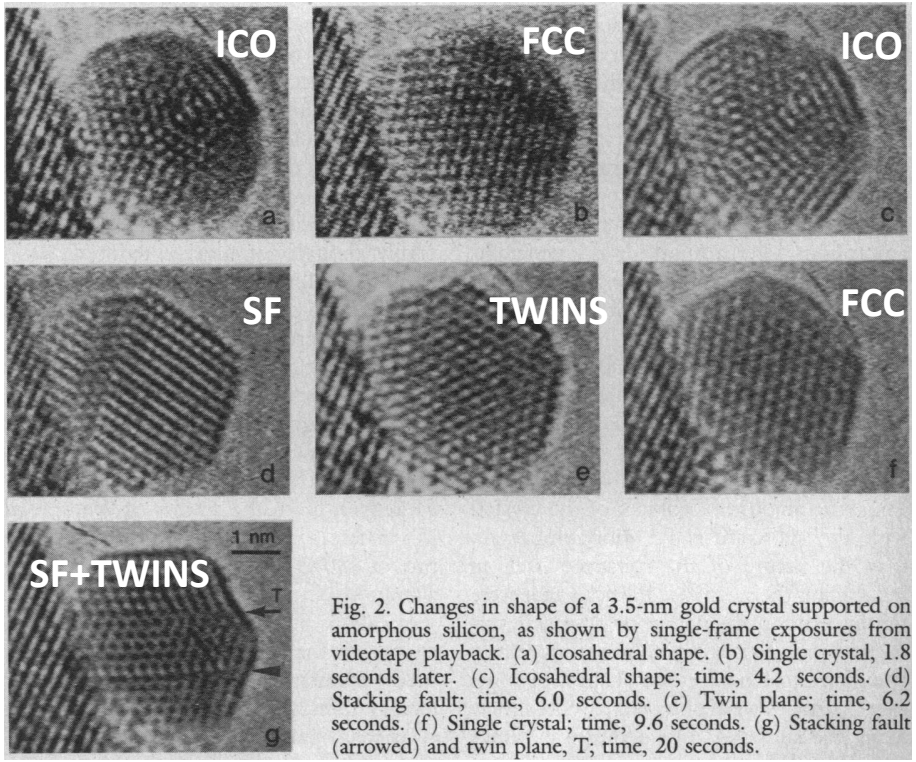


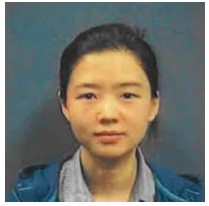
Fig. 2. Changes in shape of a 3.5-nm gold crystal supported on amorphous silicon, as shown by single-frame exposures from videotape playback. (a) Icosahedral shape. (b) Single crystal, 1.8 seconds later. (c) Icosahedral shape; time, 4.2 seconds. (d) Stacking fault; time, 6.0 seconds. (e) Twin plane; time, 6.2 seconds. (f) Single crystal; time, 9.6 seconds. (g) Stacking fault (arrowed) and twin plane, T; time, 20 seconds.

Smith *et al.*, Science 233, 872 (1986)

Shape Fluctuations in Nanoparticles

- Metallic nanoparticles (150-300 atoms)
- Between 3,600 and 36,000 cores

- **Long simulations:** up to 4 ms
- **Many transitions:** up to ~100M per run
- **Many states:** up to ~1M per run



Rao Huang
(Xiamen U.)

Huang, Lo, Wen, Voter, Perez, JCP 147, 152717 (2017)
Perez, Huang, Voter, JMR 33, 813 (2018)
Huang, Wen, Voter, Perez, Phys. Rev. Mat. 2, 126002 (2018)

Element	Number of Atoms	T (K)	Trajectory Length (ps)	Number of Transitions	Number of States	Description
Pt	146	900	70,257,528	162,965	6,246	fcc ⇒ deca ⇒ ico
	170	800	672,396,434	1,937,031	147,377	fcc ⇔ 5-fold caps ⇒ ico
		900	20,373,095	240,306	117,680	
	190	800	1,350,168,728	6,630,131	303,572	----------
		900	348,662,895	688,027	93,346	fcc ⇒ ico
	231	900	1,986,709,692	4,395,285	252,153	----------
		1000	92,171,602	955,401	42,383	
		1100	24,608,419	914,005	110,290	
Cu	146	550	301,832,137	3,942,180	237,293	fcc ⇒ ico
	170	500	4,156,073,707	6,160,286	240,594	----------
		550	23,712,165	656,202	241,491	fcc ⇔ 5-fold caps ⇒ ico
		600	21,690,608	1,039,065	144,713	fcc ⇒ deca ⇒ ico deca⇒ fcc ⇒ ico
	190	500	489,113,720	93,863,998	368,356	----------
		600	91,701,072	9,863,950	847,016	
	231	500	438,302,547	49,409	12,817	
		550	66,578,597	4,623,717	262,785	
		600	85,056,822	184,737	169,217	
		700	832,190	237,840	89,356	
Au	146	600	237,233,817	22,910,983	119,489	fcc ⇒ ico
	190	600	521,506,615	10,198,278	85,875	fcc ⇔ 5-fold caps
	231	800	774,813,889	795,678	159,743	fcc ⇒ 5-fold caps ⇒helical
Ag	146	500	122,897,307	2,558,937	71,357	----------
		550	21,613,546	1,988,646	136,297	fcc ⇔ off-centered 5-fold axis
	170	500	841,036,559	1,529,663	258,281	----------
		600	128,965,726	3,961,585	616,430	
	190	400	1,651,496,973	2,416,400	60,802	
		500	109,165,848	1,414,790	154,083	
		600	30,620,753	1,091,307	147,863	
	231	500	20,445,451	946,623	92,818	

Benchmark results: An Easy Case

Rare events

T=300K, LANL Grizzly, 4h runs

N_{cores}	Trajectory length (ps)	Generated segment time (ps)	#Transitions	#States	$\langle t_{trans}/N_{tc} \rangle$	<R>	Simulation rate (μs/hour)
9,000	556,093,988	556,539,980	4,614	28	13.[illegible]9	166	139
18,000	1,315,941,923	1,346,516,503	24,610	64	2.[illegible]7	384	333
27,000	2,209,432,238	2,214,868,608	13,479	47	4.[illegible]5	294	552
36,000	2,291,027,808	2,318,254,470	50,258	60	1.26	909	592

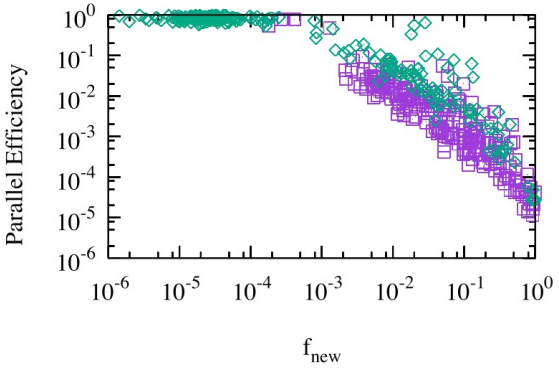
99% of generated segments were spliced

Peak simulation rate: 10 μs/min, 10 ms/day

Benchmark results: Hard Cases

~4x from speculation

Most of performance from revisits



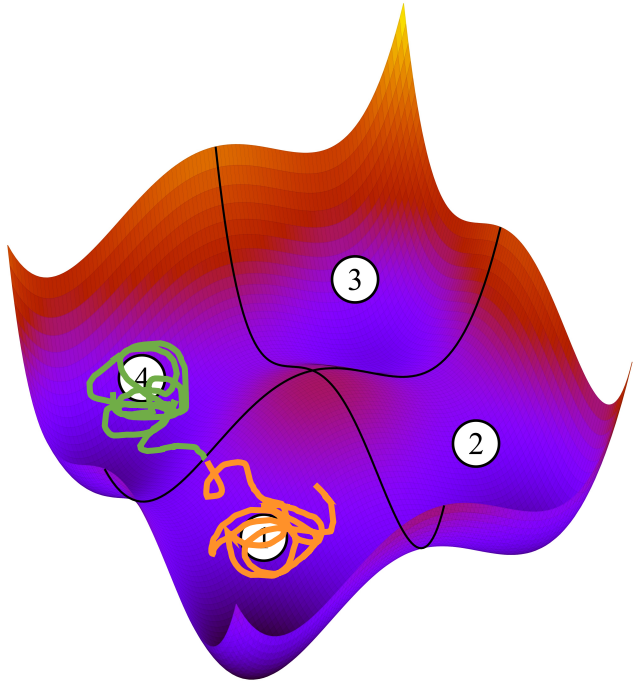
	Trajectory length (ps)	Generated segment time (ps)	#Transitions	#States	$\langle t_{trans}/M\ t_c \rangle$	<R>	Simulation rate (μs/hour)
	340,699,441	341,011,427	2,227	15	21.25	149	85
400	267,608,621	305,631,041	21,629,711	4,785	0.0017	545	69
600	194,178,176	269,536,048	90,096,511	24,161	0.00029	3846	48
700	70,212,784	228,559,866	33,780,937	250,867	0.00028	135	17
800	1,754,393	220,695,348	738,292	36,613	0.00033	20	0.45
900	169,943	10,673,302	64,208	11,577	0.00030	6	0.043

75% of generated segments were spliced
2700x speedup over MD

Very fast events: need only a few segments to escape

Discretization of continuous dynamics

- The ParSplice formalism maps complex continuous dynamics into a simple, **arbitrarily accurate,** discrete framework
- Can it inform the development of accurate discrete state models?

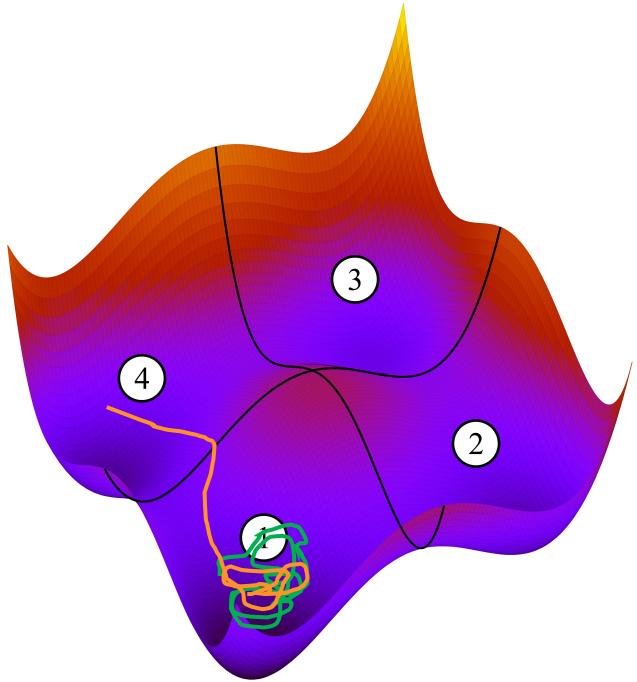


- Usual mapping is based on domains in configurations space
- Discrete model becomes a CTMC in the limit $(\lambda_2 - \lambda_1) \rightarrow \infty$ for all states
- No clear picture away from this limit

[T. Lelièvre, Handbook of Materials Modeling: Methods: Theory and Modeling, 773]

Markov Renewal Process representation

- ParSplice inspired mapping:
  - The “color” of a trajectory is the color of the last state it spent t_c in
- The color encodes the last domain the trajectory reached the QSD in.
- What is the appropriate representation of the color-to-color dynamics?



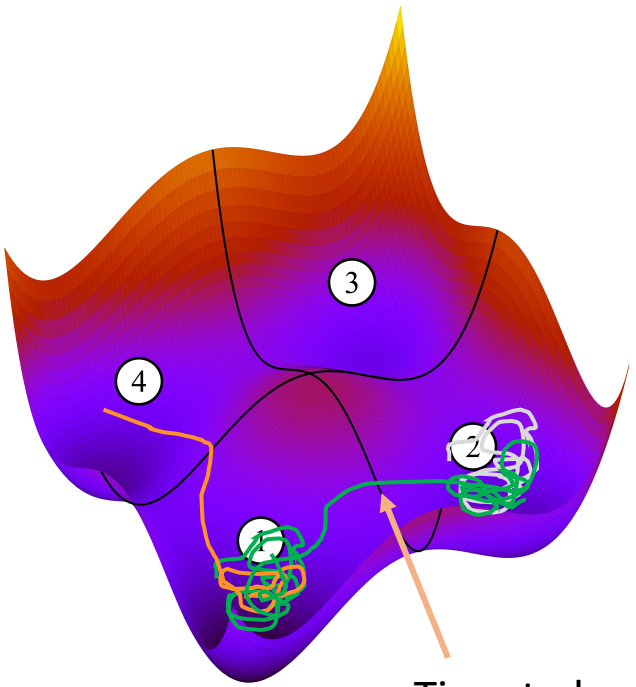
Markov Renewal Process representation

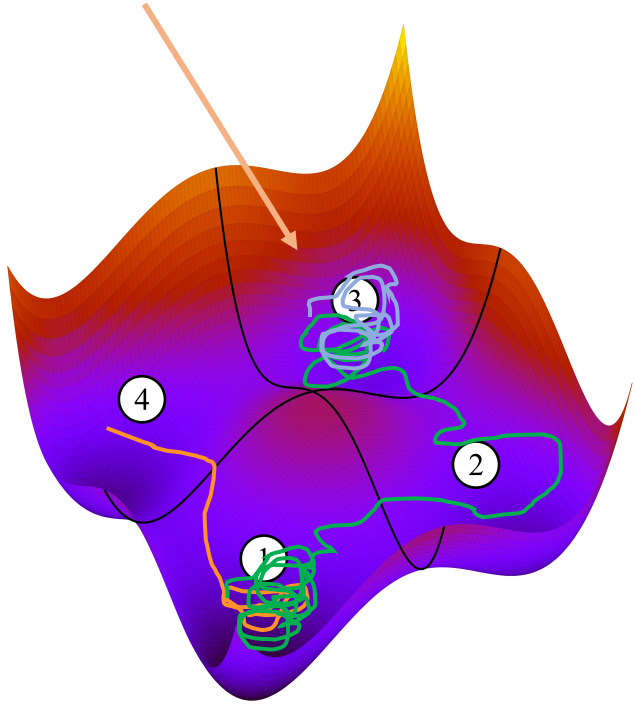
- Color changes when trajectory reaches QSD in a new state

- From the properties of the QSD:
  - Probability of next color can only depend on current color
  - Distribution of time to next color change cannot depend on previous colors
  - Distribution of time to next color change cannot depend on previous change times
  - Distribution of time to next color change can depend on next color

Agarwal, Gnanakaran, Hengartner, Voter, Perez, arXiv:2008.11623

Time to settle in new state and change color can depend on new color





Time to leave the state independent of past and future color

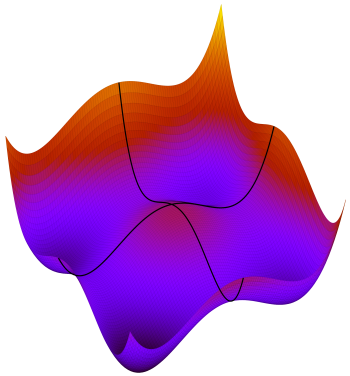
Markov Renewal Process

Color-to-color dynamics is described by a Markov Renewal Process*

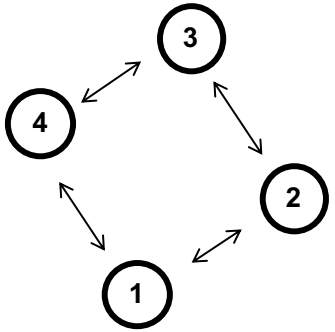
$$P(c_{n+1}, t_{n+1} < T | \, history \,) = p_{c_{n+1},c_n} \, F_{c_{n+1},c_n}(T - t_n)$$

for *any state definition*

* Up to an exponentially small error in t_c



QSD-to-QSD factorization



Continuous Trajectory

QSD-to-QSD factorization

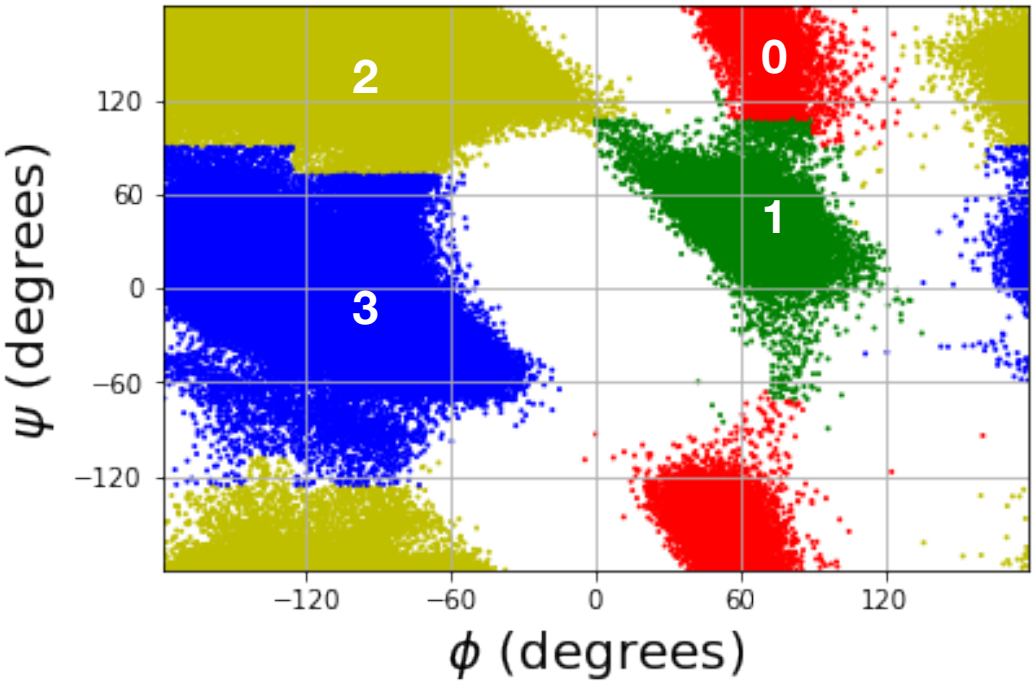
Markov renewal process

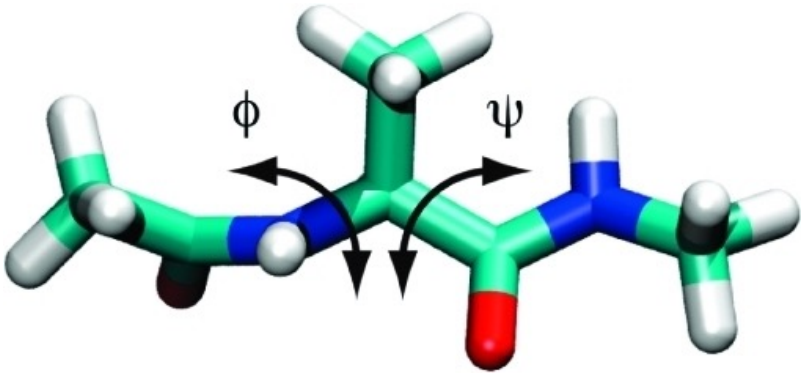
Fokker-Planck equation

QSD-to-QSD factorization

Renewal equations

Alanine dipeptide



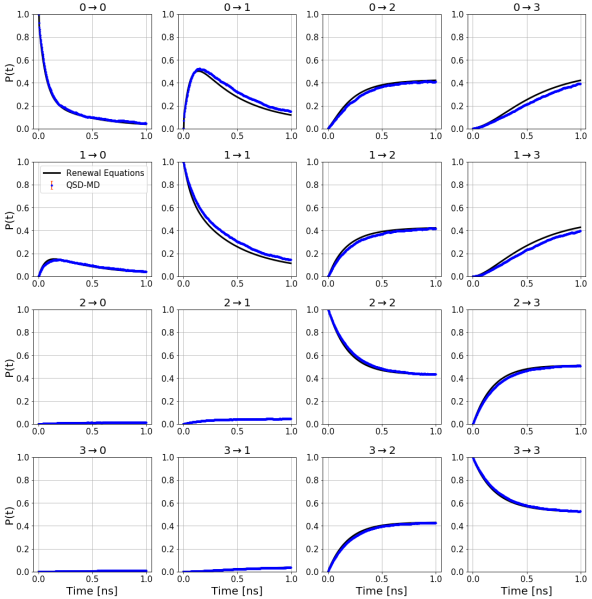


Carefully defined domains using PCCA

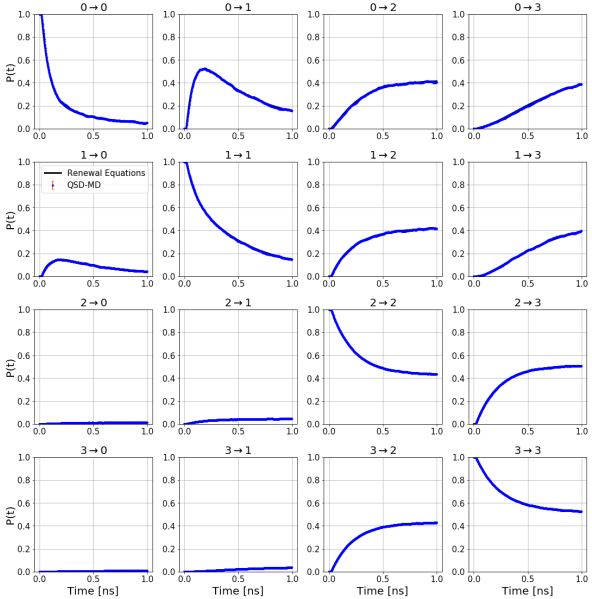
Alanine dipeptide

Direct MD

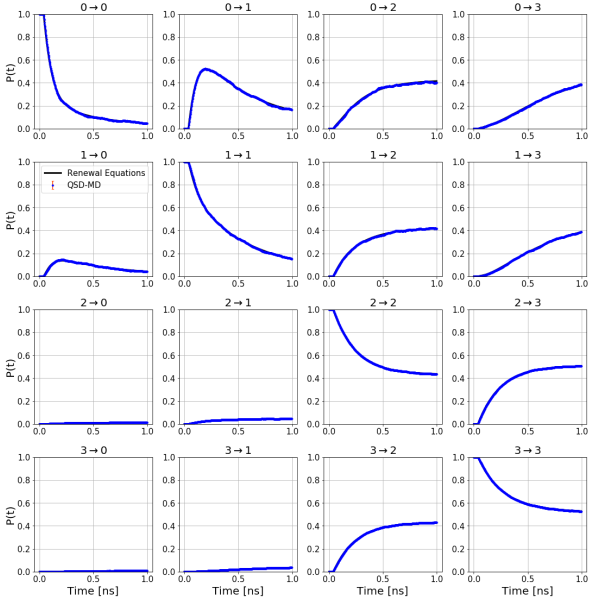
Renewal equations



t_c = 2 ps

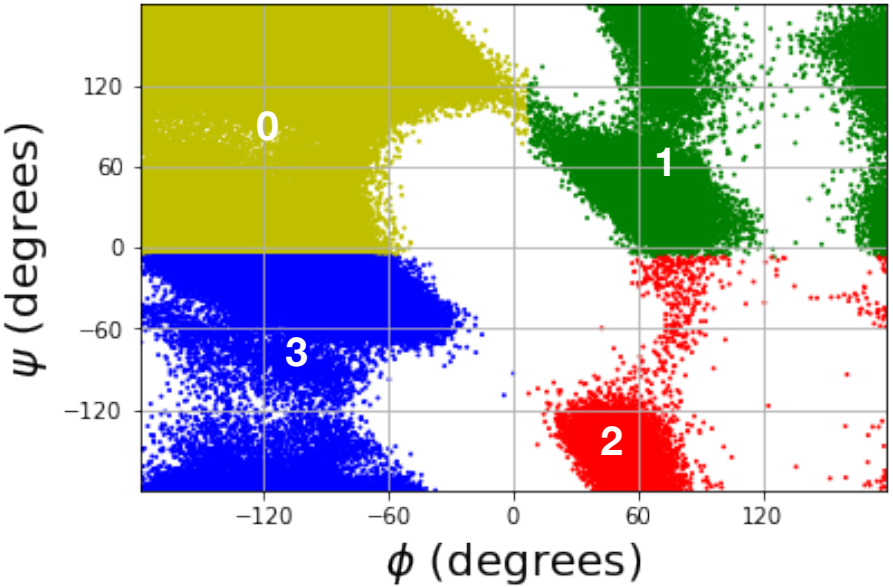


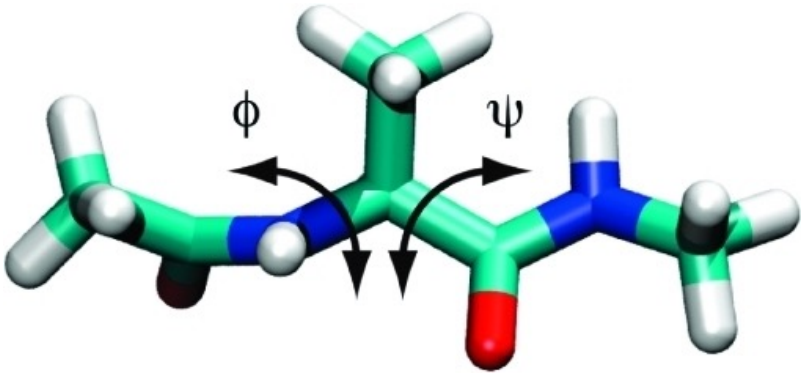
t_c = 20 ps



t_c = 40 ps

Alanine dipeptide



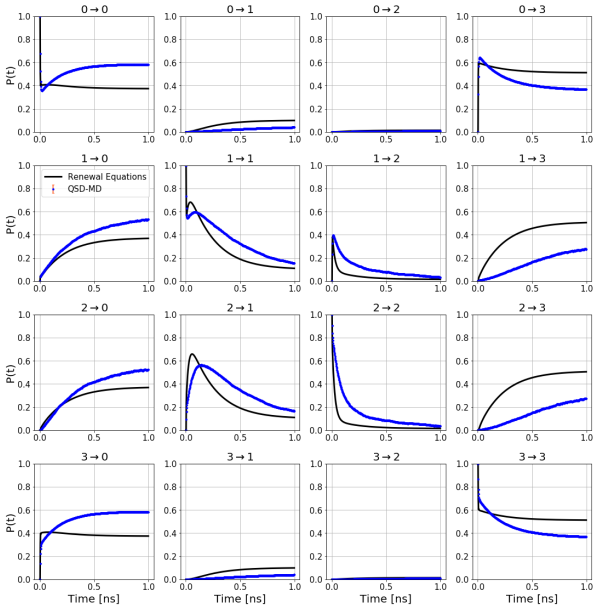


Intentionally poorly defined states

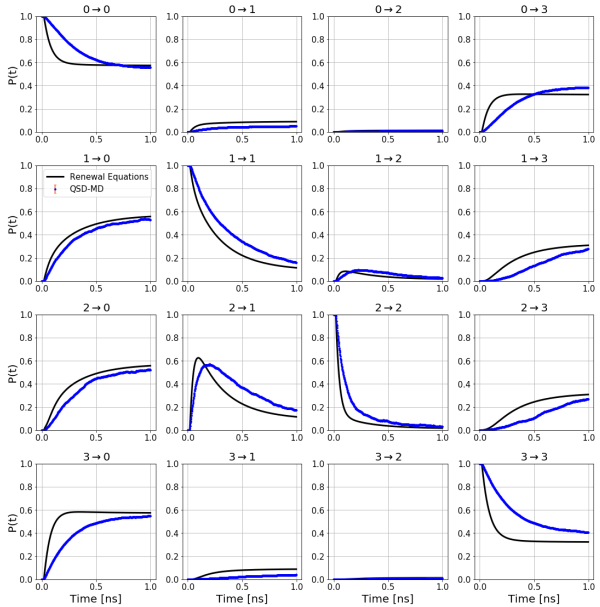
Alanine dipeptide

Direct MD

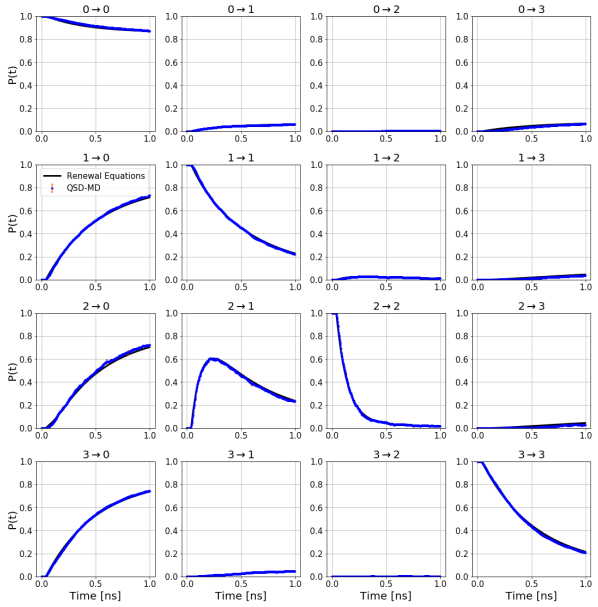
Renewal equations



t_c = 2 ps



t_c = 20 ps

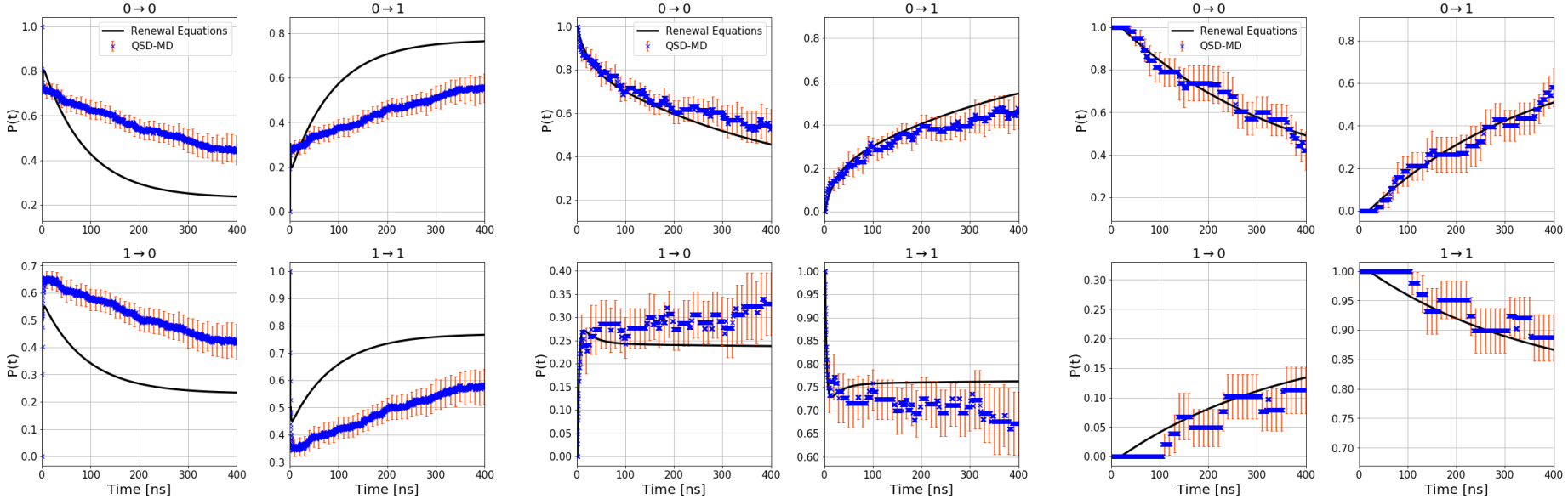


t_c = 40 ps

Villin headpiece

Direct MD

Renewal equations



t_c = 2 ps

t_c = 2 ns

t_c = 20 ns

Markov Renewal Process

- Not the only discretization scheme (CTMC, Hidden Markov Model, …)
- To our knowledge, simplest scheme that provides arbitrary accuracy *for any state definition*
- **Caveat:**
  - not very informative if dynamics are not metastable. Leads to very long jumps.
- Next step: provide efficient numerical schemes to parameterize the MRP (ongoing work with D. Aristoff)

Agarwal, Gnanakaran, Hengartner, Voter, DP, arXiv:2008.11623

Conclusion

- MD is extremely powerful, but has a severe timescale limitations that cannot be cured by brute-force alone, even with exascale computing

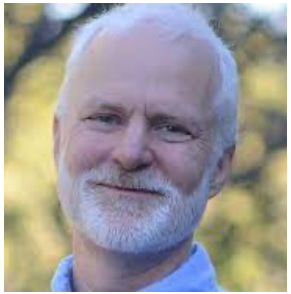
- By leveraging insights from the theory of QSD, one can design rigorous parallel-in-time techniques that dramatically extend simulation times *into the milliseconds* at basically no cost in accuracy

- This enables more direct comparison with experiments and help fill-in the blanks in experimental measurements

- Progress in applied math, computer science, and domain science, was essential to address this problem.

Collaborators/Acknowledgements

- **Method development:** Arthur Voter
- **Mathematics:** Tony Lelièvre, Claude Le Bris, Mitch Luskin, David Aristoff
- **Code:** The EXAALT ECP team
- **Funding:** DOE BES, ECP; LANL LDRD
- **Computing:** LANL IC, NERSC





Los Alamos NATIONAL LABORATORY

ECP EXASCALE COMPUTING PROJECT