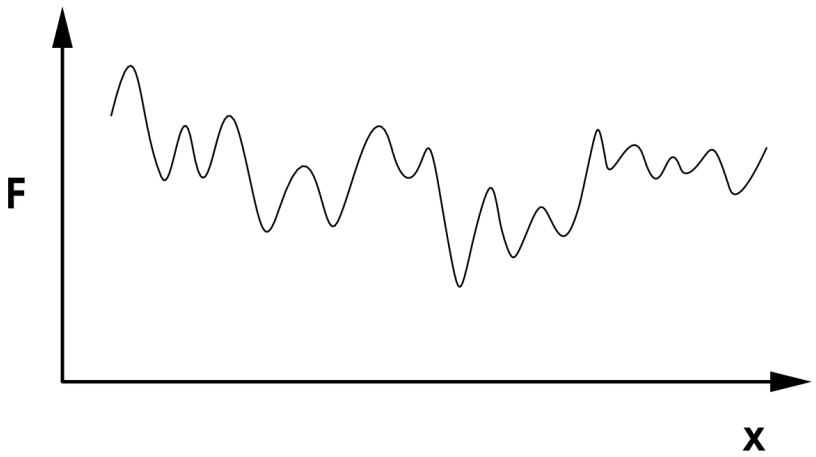
Accelerating Molecular Dynamics Simulations with Population Annealing

Henrik Christiansen[1], Martin Weigel[2], and Wolfhard Janke[1]

[1]University of Leipzig, Institute for Theoretical Physics
[2]Coventry University, Applied Mathematics Research Centre

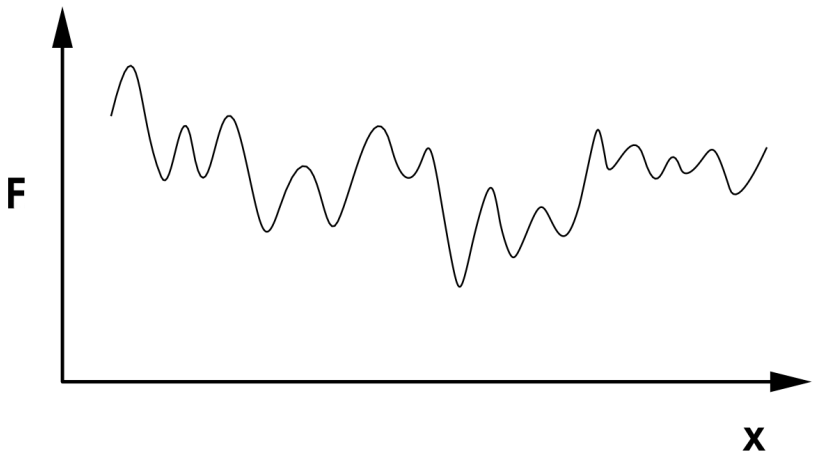
CompPhys18
19th International NTZ-Workshop on
New Developments in Computational Physics
November 30, 2018

What is the Problem?



- The folding/dynamics of biopolymers is inhibited by free-energy barriers.
- A simulation at a fixed (low) temperature (canonical ensemble) can "fall" into a local free energy minimum and get stuck there.

What is the Problem?



- The folding/dynamics of biopolymers is inhibited by free-energy barriers.
- A simulation at a fixed (low) temperature (canonical ensemble) can "fall" into a local free energy minimum and get stuck there.
- $\rightarrow$ only subset of configuration space at that temperature is sampled

$\rightarrow$ need better way to simulate

Simulation Details

- Simulations of biopolymers can be done by using Molecular Dynamics (solve newtons equation using numerical integration)
  - Canonical ensemble realized by the Langevin thermostat.
  - Interactions are modelled by an all-atom force-field (AMBER ff94).

Simulation Details

- Simulations of biopolymers can be done by using Molecular Dynamics (solve newtons equation using numerical integration)
  - Canonical ensemble realized by the Langevin thermostat.
  - Interactions are modelled by an all-atom force-field (AMBER ff94).
- Goal: Be able to sample the "full" configuration space at every temperature (also at very low temperatures) $\rightarrow$ Canonical simulations (at one temperature) are not sufficient, because they get stuck in configurational sub regions

We will compare direct canonical and parallel tempering simulations with the (for MD) novel method called population annealing.

Current State: Parallel Tempering [1] [2] + Spatial Decomposition

Simulate the same system simultaneously at a number of temperatures.

[1] U.H.E. Hansmann, Chem. Phys. Lett. 281, 140 (1997)
[2] Y. Sugita, Y. Okamoto, Chem. Phys. Let. 314, 141 (1999)

Current State: Parallel Tempering [1] [2] + Spatial Decomposition

Simulate the same system simultaneously at a number of temperatures.
$\rightarrow$ Allow exchanges of configurations between temperatures.

[1] U.H.E. Hansmann, Chem. Phys. Lett. 281, 140 (1997)
[2] Y. Sugita, Y. Okamoto, Chem. Phys. Let. 314, 141 (1999)

Current State: Parallel Tempering [1] [2] + Spatial Decomposition

Simulate the same system simultaneously at a number of temperatures.
$\rightarrow$ Allow exchanges of configurations between temperatures.

1. Set up one simulation at each temperature of the temperature set
2. In each simulation independently perform a MD simulation of θ steps
3. Switch temperatures of two simulations with probability
   $p = \min\left(1, e^{(E_i - E_j)\left(\frac{1}{kT_i} - \frac{1}{kT_j}\right)}\right)$
4. Go to 2, until enough statistics in accumulated. Observables $\mathcal{O}$ are calculated as time averages at each temperature

[1] U.H.E. Hansmann, Chem. Phys. Lett. 281, 140 (1997)
[2] Y. Sugita, Y. Okamoto, Chem. Phys. Let. 314, 141 (1999)

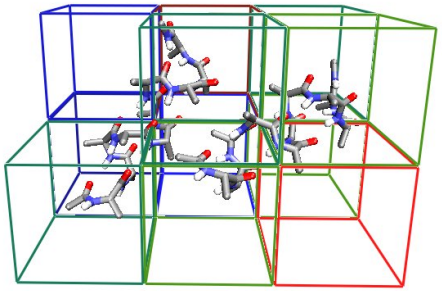
Current State: Parallel Tempering [1] [2] + Spatial Decomposition

Simulate the same system simultaneously at a number of temperatures.
$\rightarrow$ Allow exchanges of configurations between temperatures.

1. Set up one simulation at each temperature of the temperature set
2. In each simulation independently perform a MD simulation of θ steps
3. Switch temperatures of two simulations with probability
   $p = \min\left(1, e^{(E_i - E_j)\left(\frac{1}{kT_i} - \frac{1}{kT_j}\right)}\right)$
4. Go to 2, until enough statistics in accumulated. Observables $\mathcal{O}$ are calculated as time averages at each temperature

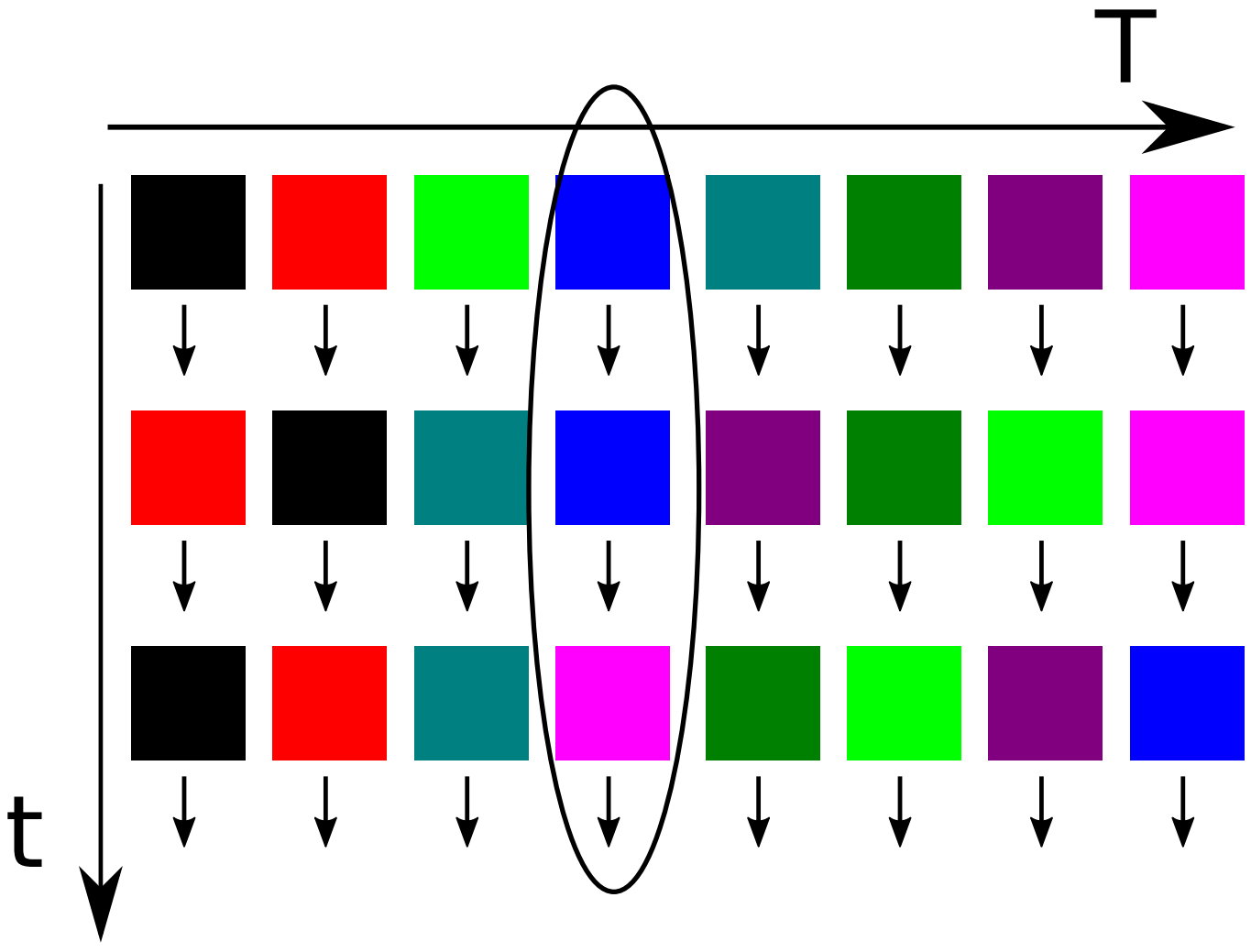
Spatial Decomposition (suited for GPUs):

- subdivide configuration space into several "independent" parts - treat neighbouring region separately



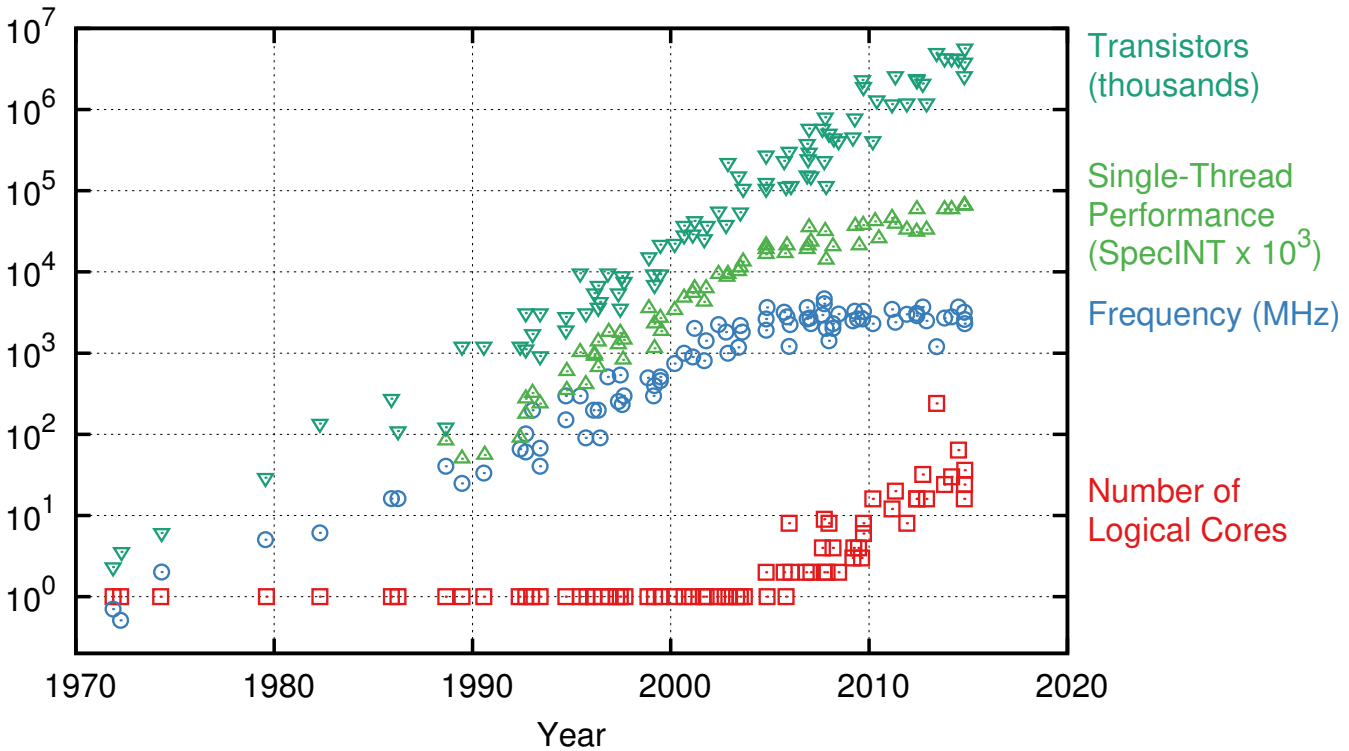
[1] U.H.E. Hansmann, Chem. Phys. Lett. 281, 140 (1997)
[2] Y. Sugita, Y. Okamoto, Chem. Phys. Let. 314, 141 (1999)

Schematics of the Parallel Tempering Algorithm



Effective use of up to $\approx$ hundreds of CPU-cores

Why is that not enough?



Top 3 supercomputers of Top500 supercomputer-list in November 2018:

- Summit with 2,397,824 CPU-cores
- Sierra with 1,572,480 CPU-cores
- Sunway TaihuLight with 10,649,600 CPU-cores

Simulated Annealing [3]

1. Set up an equilibrium ensemble of R independent copies of the system at some high temperature T_0
2.
3. each of the replicas is now updated by θ molecular dynamics steps at temperature T_i
4.
5. goto step 2 until T_i is the target temperature

$\rightarrow$ very well known method to find the ground-state of systems, however no thermodynamic properties can be measured

[3] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, Science 220, 671 (1983).

Population Annealing [3] [4]

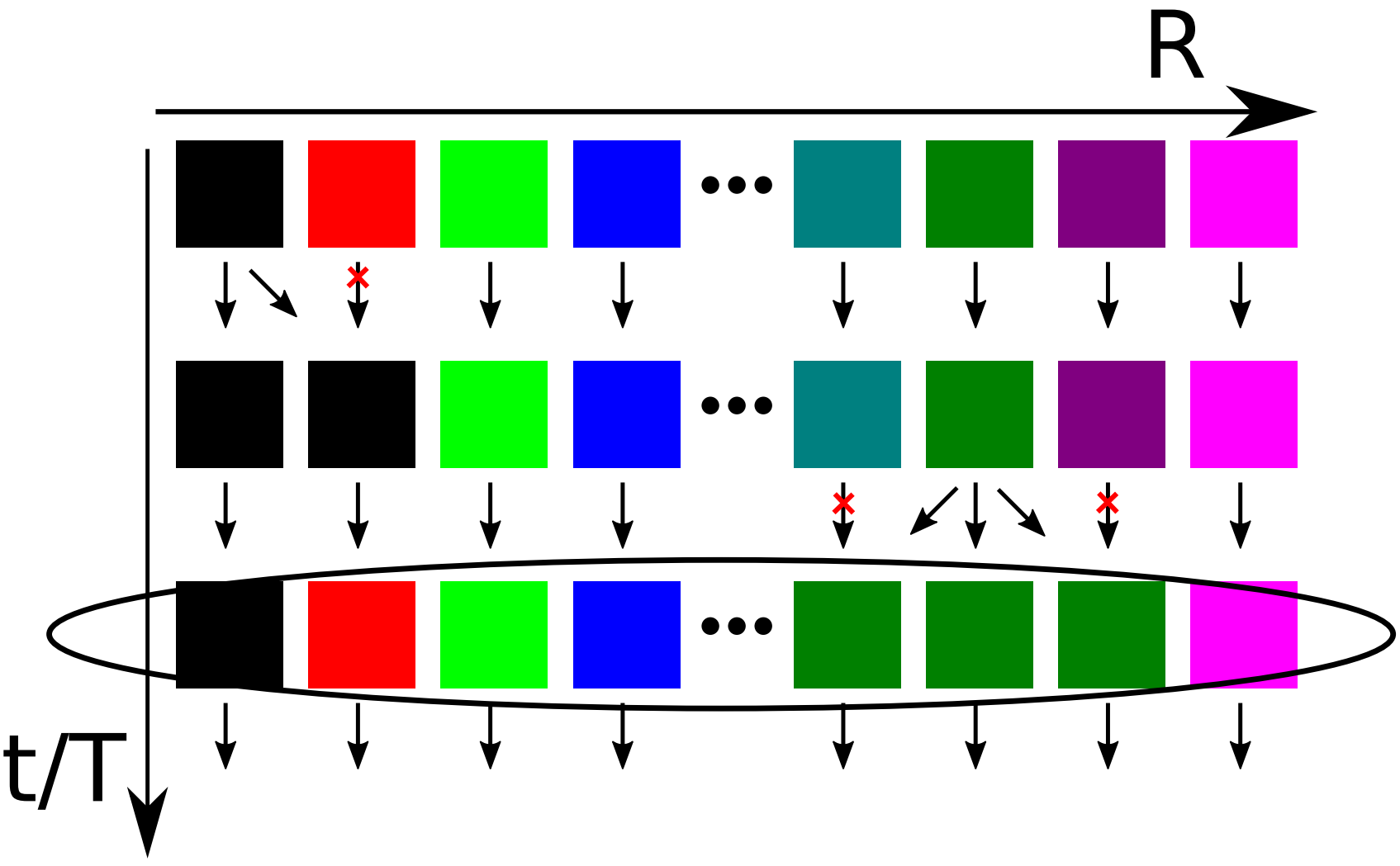
1. Set up an equilibrium ensemble of R independent copies of the system at some high temperature T_0
2. Resample the ensemble of systems to a temperature $T_i < T_{i-1}$
   - to reweight them use their relative Boltzmann weight $e^{-(1/k_B T_i - 1/k_B T_{i-1})E_j} / Q$, with the normalization $Q = \sum_{j=1}^{R} e^{-(1/k_B T_i - 1/k_B T_{i-1})E_j}$
3. each of the replicas is now updated by θ molecular dynamics steps at temperature T_i
4. calculate observables $\mathcal{O}$ at temperature T_i as population averages
5. goto step 2 until T_i is the target temperature

$\rightarrow$ The resampling method implies, that the thermostat has to be stochastic: Use of the Langevin-thermostat.

Of course one can also still make use of spatial decomposition!

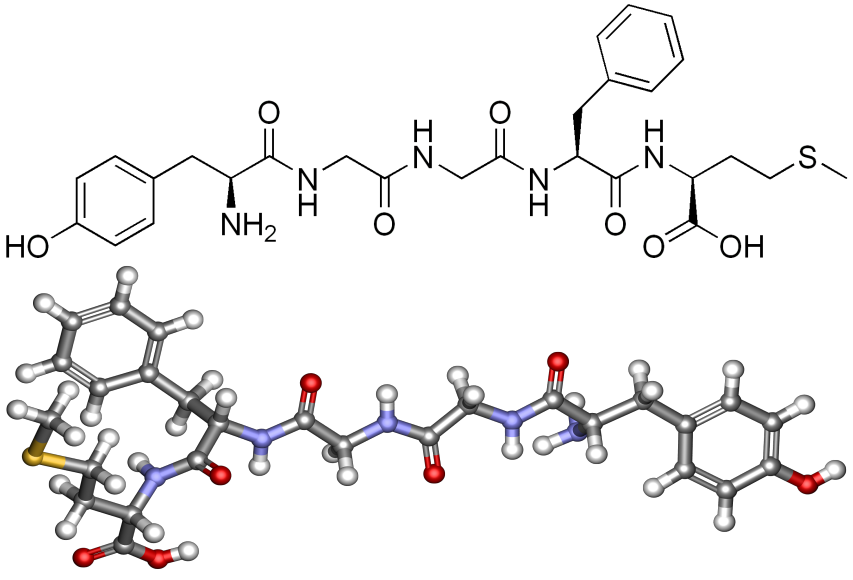
[3] Y. Iba, Trans. Jpn. Soc. Artif. Intell. 16, 279 (2001).
[4] S. Kirkpatrick, C.D. Gelatt and M.P. Vecchi, Science 220, 671 (1983).

Schematics of Population Annealing Algorithm



Effective use of "arbitrary" number of CPU-cores

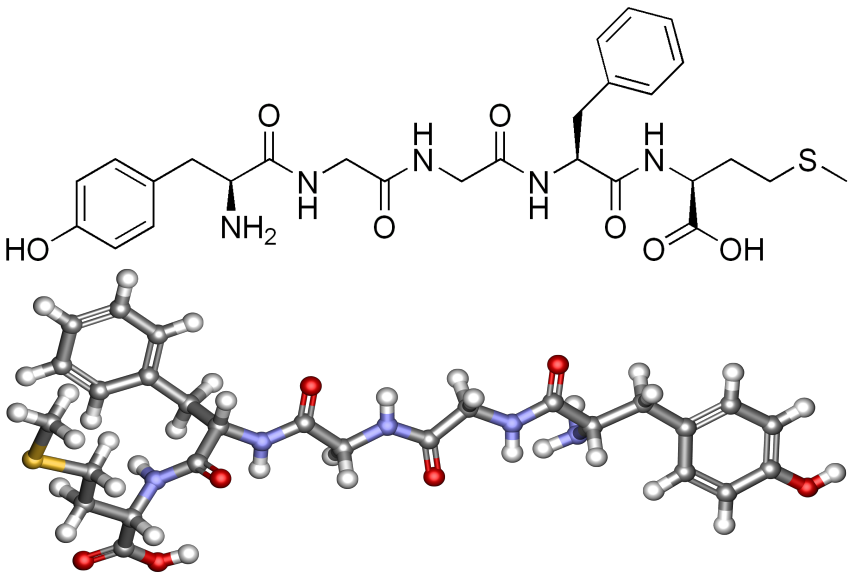
Testcase: Met-enkephalin



- penta-peptide with amino-acid sequence Tyr–Gly–Gly–Phe–Met
- "naturally occurring, endogenous opioid peptide that has opioid effects of a relatively short duration"
- "[...] has the ability to inhibit tumor growth and metastasis" [5]

[5] P. Chen and P. Bonaldo, in International Review of Cell and Molecular Biology, International Review of Cell and Molecular Biology, Vol. 301, edited by K. W. Jeon (Academic Press, 2013) pp. 1 – 35.

Testcase: Met-enkephalin

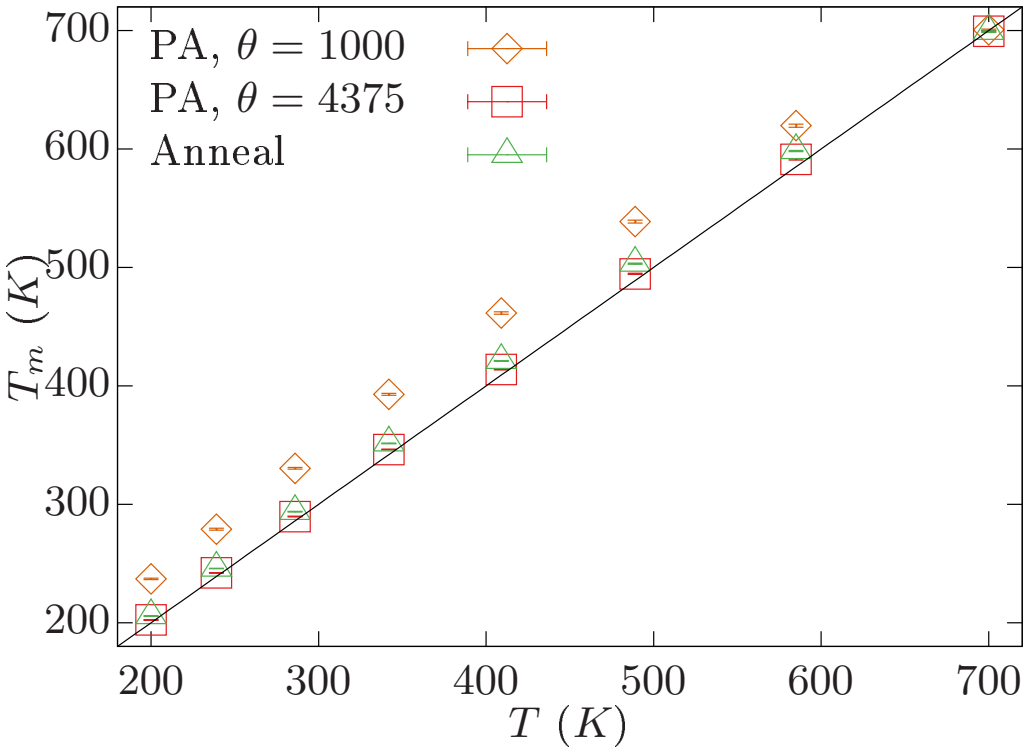


- penta-peptide with amino-acid sequence Tyr–Gly–Gly–Phe–Met
- "naturally occurring, endogenous opioid peptide that has opioid effects of a relatively short duration"
- "[...] has the ability to inhibit tumor growth and metastasis" [5]
- very common test protein for novel simulation methods

[5] P. Chen and P. Bonaldo, in International Review of Cell and Molecular Biology, International Review of Cell and Molecular Biology, Vol. 301, edited by K. W. Jeon (Academic Press, 2013) pp. 1 – 35.

Thermalization

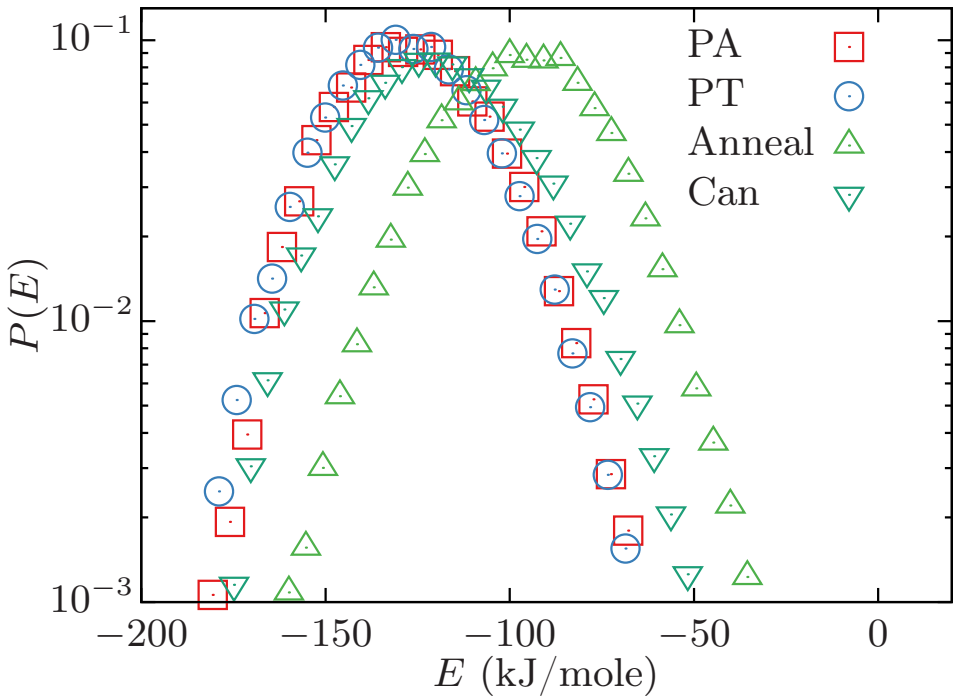
Thermostat temperature T vs. measured temperature T_m



"Anneal" here refers to a simulated annealing simulation.

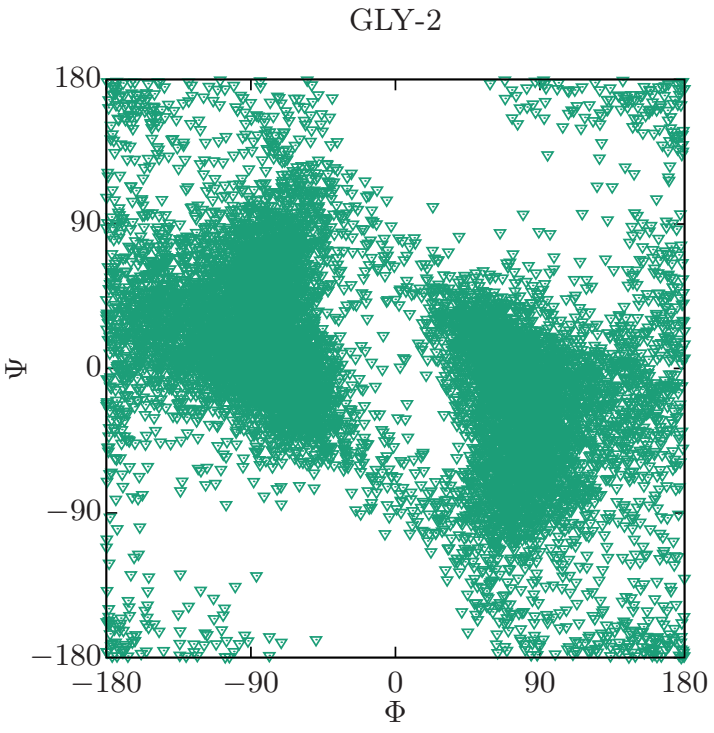
Energy Histogram - $T = 200K$

Energy histogram of parallel tempering and population annealing compatible to each other.

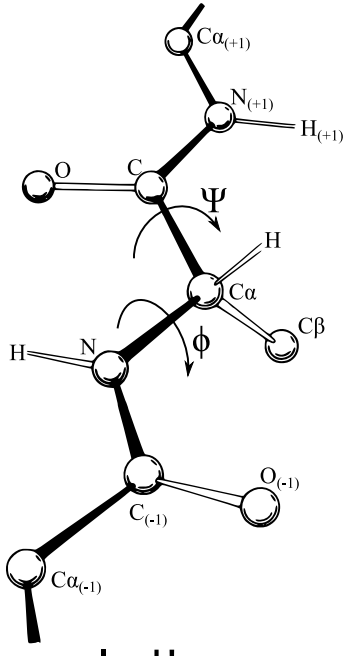


- Annealing scheme does lead to a energy histogram which is shifted toward higher energies.
- The canonical simulation shows a smaller, but still noticeable, energy shift towards higher energies.

High Temperature Simulations

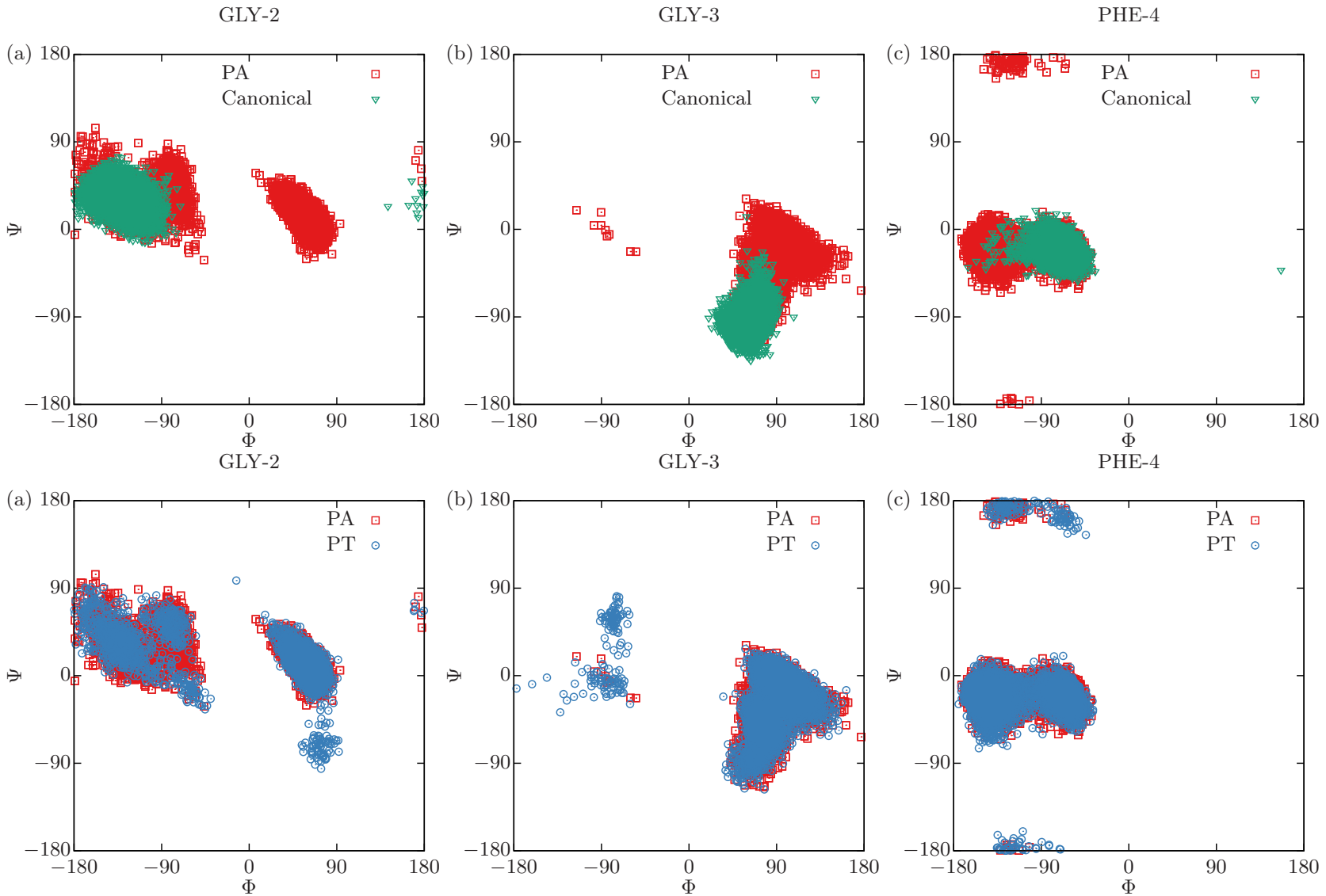


Torsion angles ϕ and ψ along the backbone of the polymer indicate visited configurations.

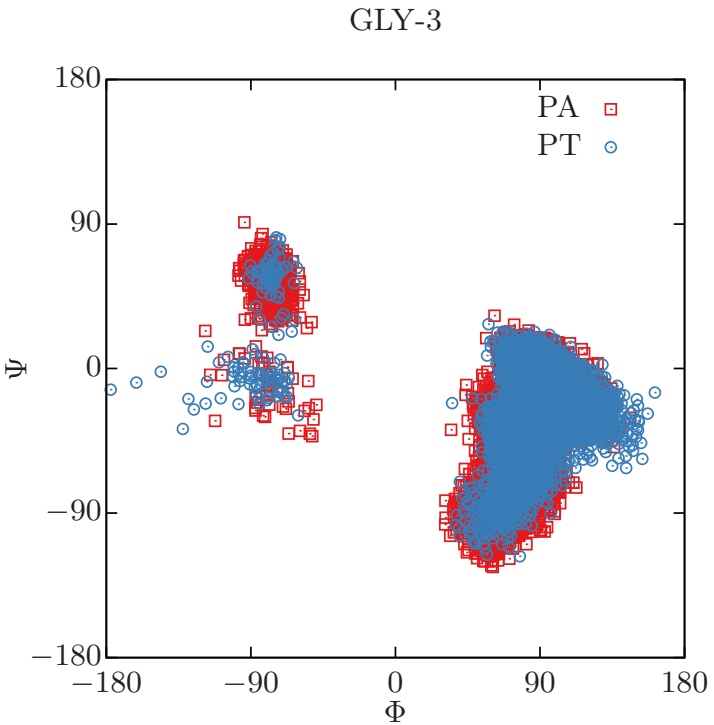


Available configurations fully sampled!

Population Annealing vs. Canonical vs. Parallel Tempering at $T = 200K$



Doubling of Replicas

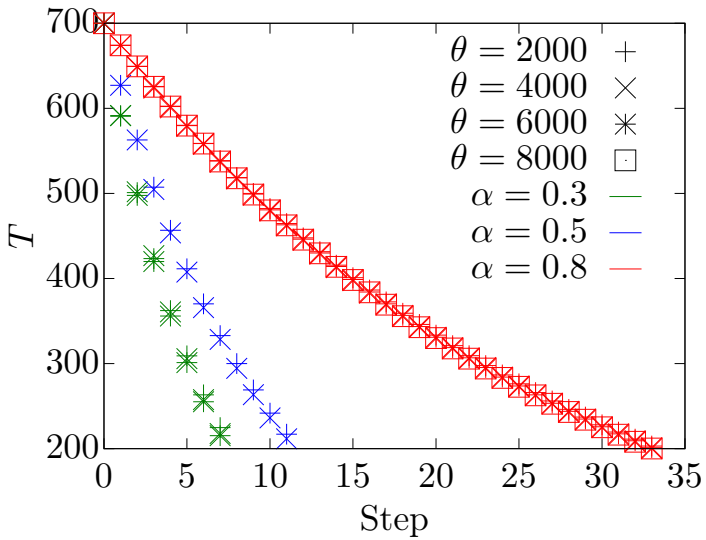


- Doubling the number of replicas in population annealing leads to a much better sampling than with parallel tempering (using the old resources)
- Increasing the number of replicas is easily doable in population annealing simulations, without increasing the wall-clock time!

Constant Energy Overlap Simulations

$$\alpha(T_{i-1}, T_i) = \frac{1}{R}\sum_{j=1}^{R} \min\left(1, \frac{\exp[-(1/k_B T_i - 1/k_B T_{i-1})E_j]}{Q(T_{i-1}, T_i)}\right).$$

with $Q = \sum_{j=1}^{R} \exp(-(1/k_B T_i - 1/k_B T_{i-1})E_j)$.



1. Set up an equilibrium ensemble.
2. Tune the next temperature T_i in such a way, that $\alpha(T_{i-1}, T_i) = \alpha^\star$.
3. Resample the ensemble of systems to the estimated temperature $T_i < T_{i-1}$.
4. Update each copy with θ simulation steps.
5. Goto step (ii) until T_i reaches or falls below the target temperature T_N.

Constant Energy Overlap Simulations

What are good parameters?

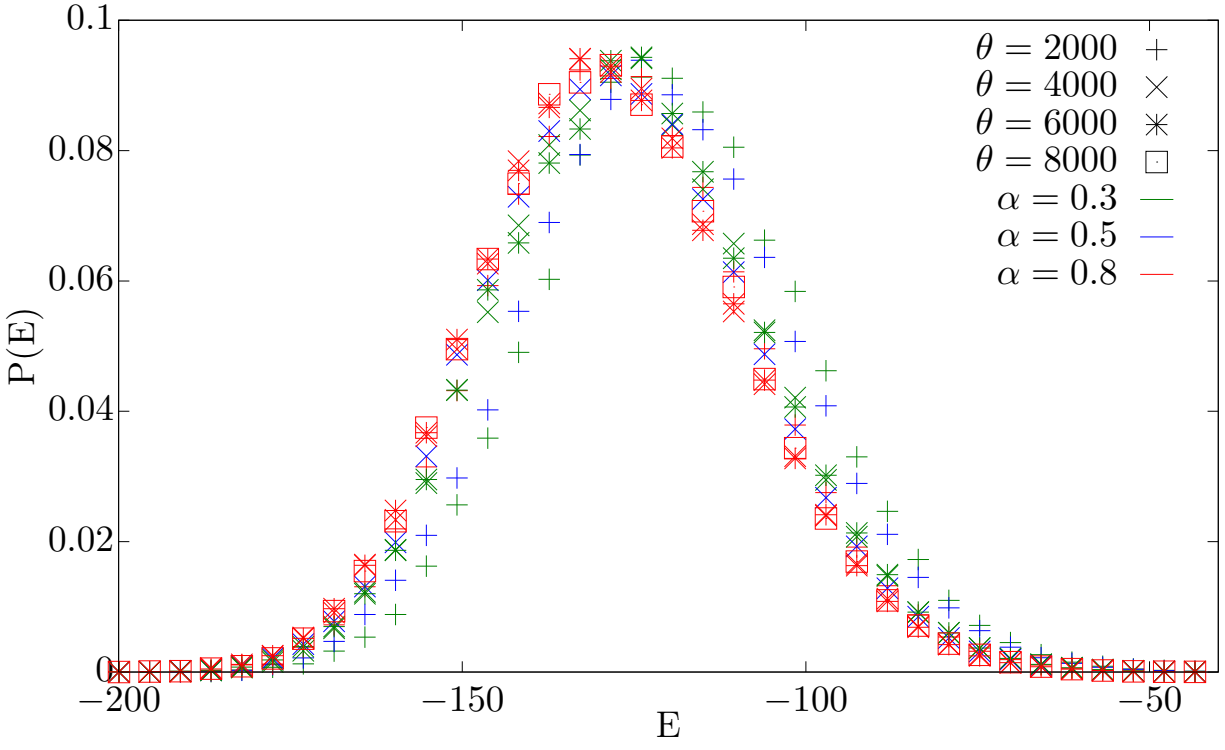
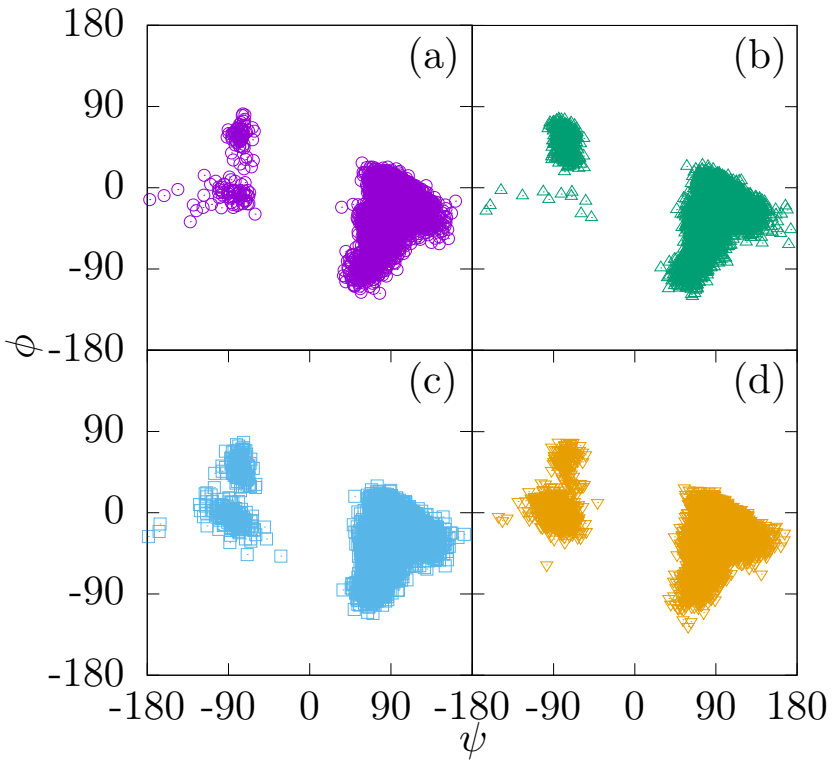


Figure: Energy histogram at $T = 200$ K for different overlaps and MD steps θ.

Constant Energy Overlap Simulations



Computational effort (normalized to (a)):

- (b) 1.525
- (c) 1.875
- (d) 1.875

Figure: Ramachandran plots for GLY3. (a) The PT simulation with runtime of 200 ns as discussed previously. PAMD simulation with adaptive temperature stepping, with $R = 10\,000$ and (b) $\alpha^\star = 0.5$, $\theta = 4000$ and (c) $\alpha^\star = 0.8$, $\theta = 2000$. (d) PAMD simulation using the given temperature set and $\theta = 4375$ with $R = 20\,000$ replicas.

Conclusion

- Using the same computational effort, both methods show comparable performance.
- The big advantage of population annealing is the massive potential for parallelism.
- Constant energy-overlap simulations make it significantly easier than in PT to find an appropriate temperature set.

Preprint: arxiv.org/abs/1806.06016