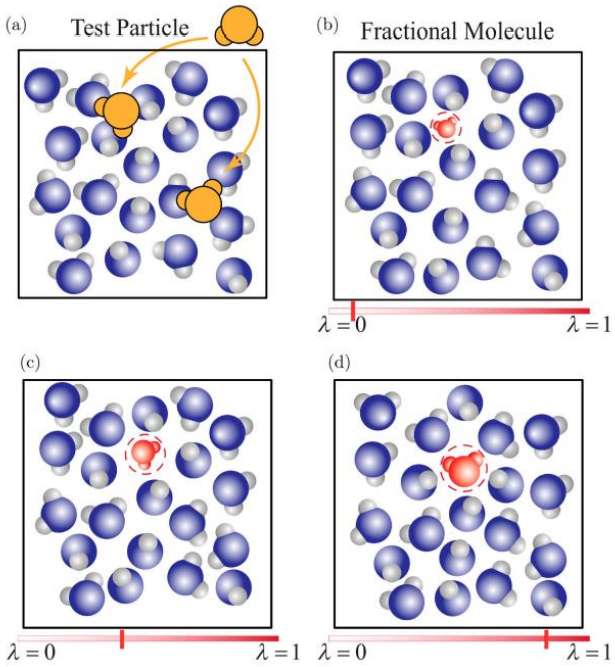
Enhanced Monte Carlo



2024 Winter Seminar

January 26th, 2024

Sangmin Lee

Mol. Sim., ***2021***, 47, 10, 804-823.

Disclaimer

For the sake of clarity, tons of formula and derivation involving MC (ex. proof that acceptance criterion satisfies detailed balance condition) will be omitted in this presentation.

In here, I firmly point out that this is not because it's unnecessary, but just for the conciseness.

One should be more careful about running MC simulation compared to MD simulation, since the former one does not follow the natural time evolution by integrating the equation of motion.

Although such feature serves room for inspiring ingenuity in MC simulation,

it also serves room for any wrong physics to be smeared in MC simulation.

One should always be careful about the validation of MC simulation from any unphysical result.

Understanding the theoretical background of MC simulation and following its formal derivation is therefore definitely required for any practitioner.

Content

Motivation: Why Enhanced MC?

MC simulation: Chain molecules

MC simulation: Condensed phases

MC simulation: Acceleration

Content

Theoretical foundation of molecular simulation

In classical mechanics, the equation of motion is integrated to generate the trajectory.

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{r}}_i}\right) - \frac{\partial \mathcal{L}}{\partial \mathbf{r}_i} = 0. \qquad \dot{q}_\alpha = \frac{\partial \mathcal{H}}{\partial p_\alpha}, \;\; \dot{p}_\alpha = -\frac{\partial \mathcal{H}}{\partial q_\alpha}.$$

Energy is conserved in classical mechanics!

By solving F=ma, we sample the **microcanonical** ensemble (constant E) for ensemble averages.

$$\langle a \rangle = \frac{\int \mathrm{dx}\; a(\mathrm{x})\delta(\mathcal{H}(\mathrm{x}) - E)}{\int \mathrm{dx}\; \delta(\mathcal{H}(\mathrm{x}) - E)} = \lim_{\mathcal{T}\to\infty} \frac{1}{\mathcal{T}} \int_0^{\mathcal{T}} \mathrm{d}t\; a(\mathrm{x}_t) \equiv \bar{a}.$$

We need to sample the x_t in microcanonical ensemble, which we call it **trajectory**.(.dcd)

In here, we introduce the time discretization parameter dt, known as the **time step.**

Starting with the initial cond x_0, x_dt, x_2dt, x_3dt are generated by applying the integrator iteratively.

$$A = \langle a \rangle = \frac{1}{M} \sum_{n=1}^{M} a(\mathrm{x}_{n\Delta t}) \equiv \bar{a}.$$

Here is the question

There are several ensembles (NPT, NVT, muVT, Gibbs, Osmotic...) in statistical mechanics.

By running either MD or MC simulation, we are trying to sample the specific ensemble and compute the ensemble averages.

$$A = \langle a \rangle = \frac{1}{M} \sum_{n=1}^{M} a(\mathrm{x}_{n\Delta t}) \equiv \bar{a}.$$

In MD, we **integrate** the equation of motion for time evolution to sample the ensemble.

In MC, we use the **random walk** move to sample the ensemble.

Note that there is no ‘dynamic’ information in MC simulation,

therefore computing transport properties (ex. Diffusion coefficient) is only available in MD simulation.

So if both ways are able to sample the ensemble, why do we have to learn MC simulation?

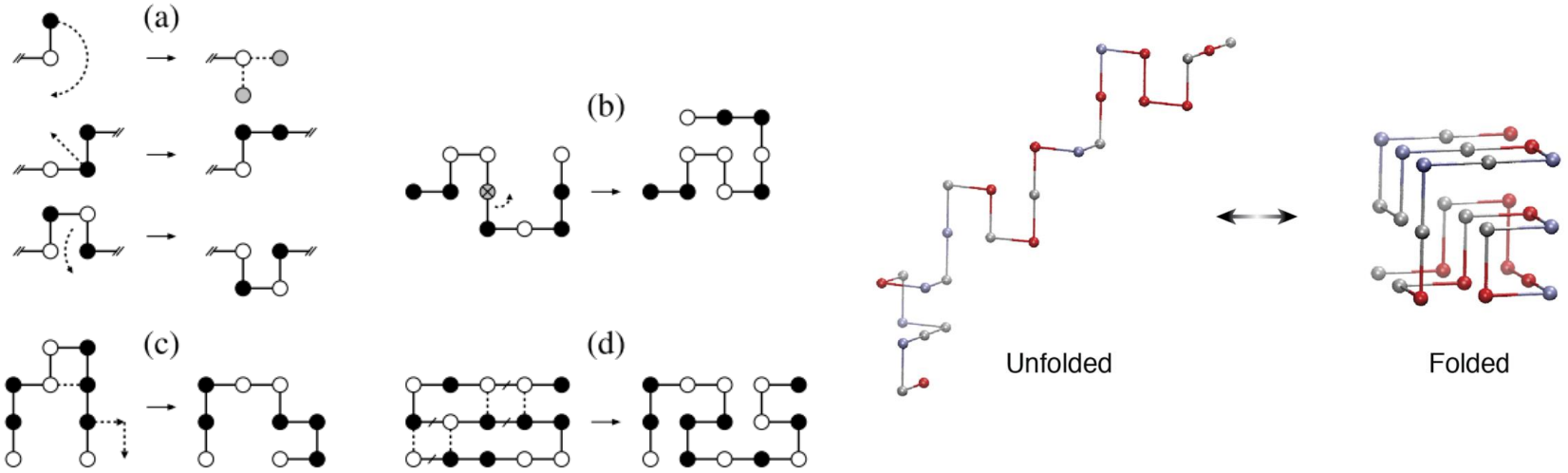
There should be some reasons for this!

The answers are...

1. There are some technical advantages to compute several physical quantities in a more straightforward way! (Pressure via virial expression, Chemical potential)

2. Sometimes, we are interested in models that do not have 'natural dynamics'

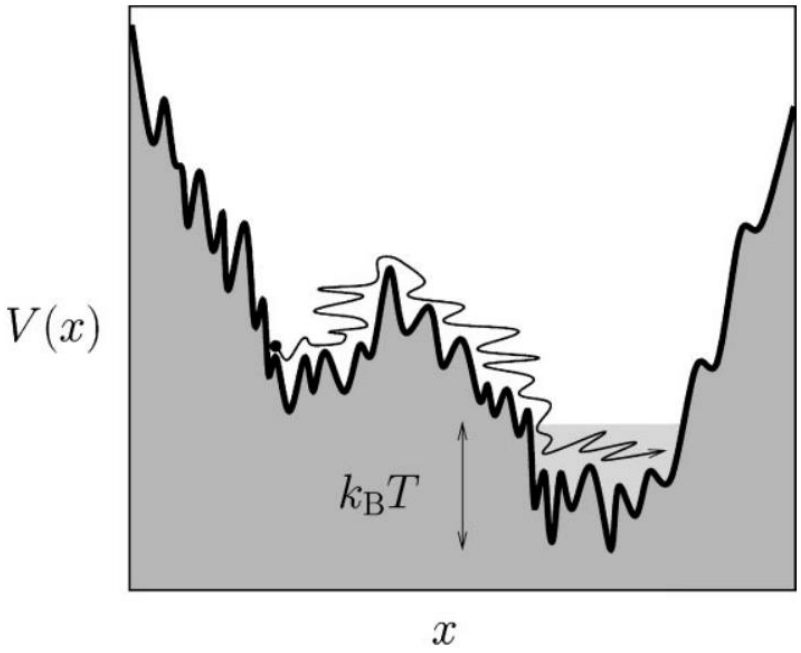
ex. Lattice model



J. Stat. Phys., ***2011,*** 144, 638-651.
J. Chem. Phys.. ***2018***, 149, 234107.

The answers are...

3. Sampling issue



In MD simulation, we integrate the equation of motion.

This means that it's difficult to surmount the energy barrier where its height is above kT due to the equipartition theorem.

If the dynamics of system is inherently slow, we have a sampling issue.

Ex. Polymer movement, First order phase transition

"opus liberabit vos" (The true will set you free)

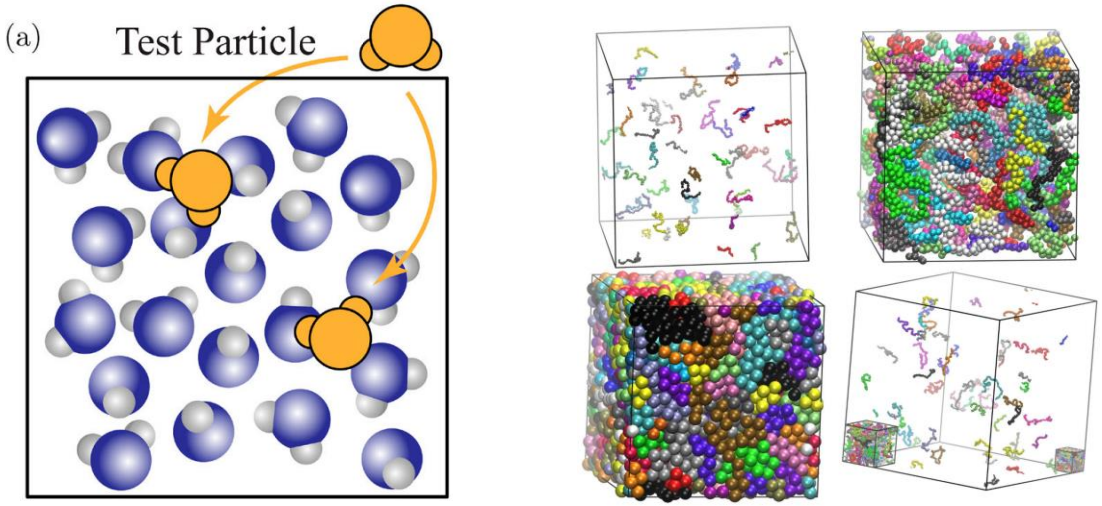
If our true is not dynamics, but just the sampling, then we may envisage a more clever MC move that can efficiently sample the ensemble without sticking to the nature dynamics.

Note that designing efficient moves requires striking a balance between rapid traversal of phase space and ensuring reasonable acceptance probabilities.

Annu. Rev. Phys. Chem. ***2002***, 53, 291-318.

So, is this really true?

The inherent issue of MC simulation in condensed phase



Particle insertion: the energetic penalty by overlapping with already existing molecules is too high

$$\mathrm{acc}(\mathrm{N} \rightarrow \mathrm{N}+1) = \min\left[1, \frac{V}{\Lambda^3(\mathrm{N}+1)} \exp\{\beta[\mu - \mathcal{U}(\mathrm{N}+1) + \mathcal{U}(\mathrm{N})]\}\right]$$

Particle deletion: the energetic penalty by spontaneously creating the cavity is too high

$$\mathrm{acc}(\mathrm{N} \rightarrow \mathrm{N}-1) = \min\left[1, \frac{\Lambda^3 \mathrm{N}}{V} \exp\{-\beta[\mu + \mathcal{U}(\mathrm{N}-1) - \mathcal{U}(\mathrm{N})]\}\right]$$

Therefore, the sampling of MC simulation is inherently **hampered** by the low acceptance probabilities in **condensed phase**.

This is the reason why early MC researches are **limited** on simulating the gas molecule adsorption or liquid-vapor phase equilibria only.

Mol. Sim., ***2021***, 47, 10, 804-823.
Int. J. Mol. Sci. ***2021***, 22, 12464.

We need ‘Enhanced’ Monte Carlo

Content

Motivation: Why Enhanced MC?

MC simulation: Chain molecules

MC simulation: Condensed phases

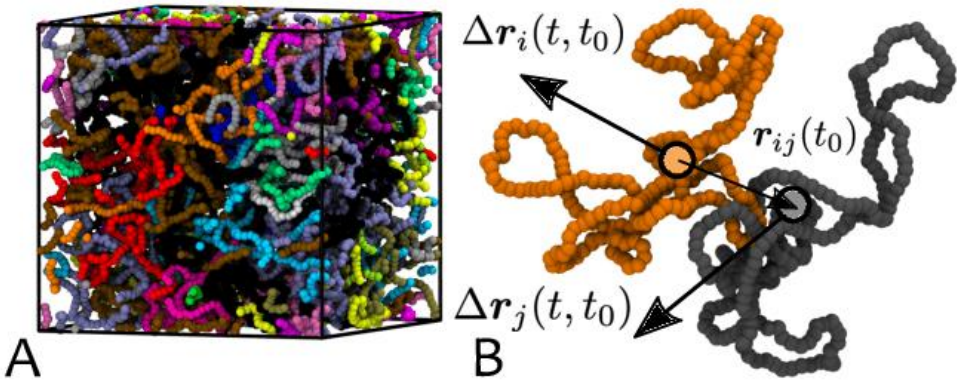
MC simulation: Acceleration

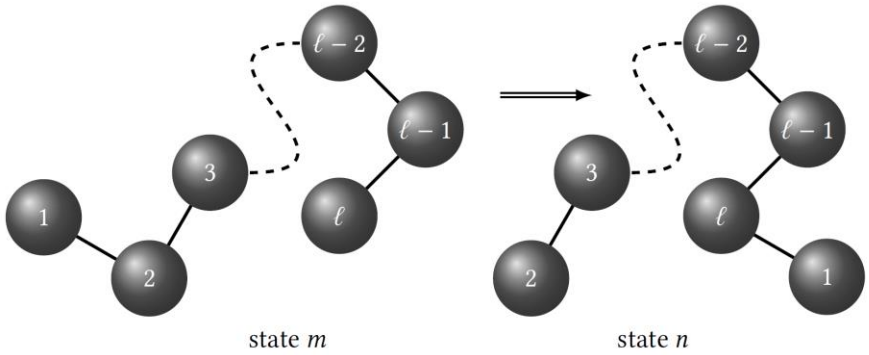
Polymer configurational sampling – Reptation MC

Sampling the single chain polymer configuration in polymer solution

In MD, the natural dynamics of single chain is restricted due to the topological constraints.

In MC, displacing the entire chain or single chain insertion/deletion would be definitely impossible.





"In a real fluid a chain is likely to move in a slithering fashion: the head of the chain moves to a new position and the rest of the chain follows like a snake or lizard." –de Gennes

In Reptation MC, for initial coordinates,

$$\{\mathbf{r}_{i,1}^m, \mathbf{r}_{i,2}^m, \ldots, \mathbf{r}_{i,\ell}^m\}.$$

A new position is selected for the head of chain i

$$\mathbf{r}_{i,1}^n = \mathbf{r}_{i,\ell}^m + \delta\mathbf{r}.$$

The new trial configuration is purposed

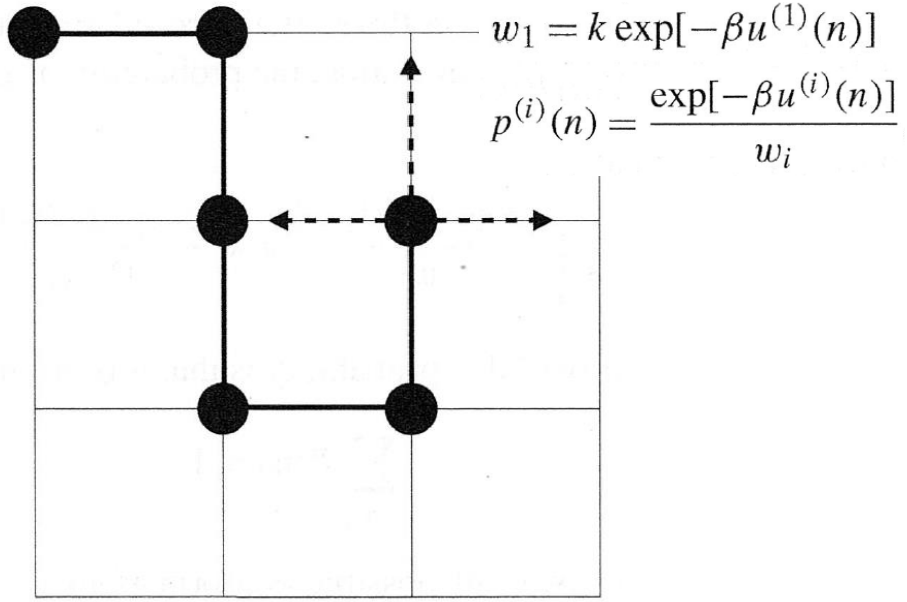
$$\{\mathbf{r}_{i,2}^n, \mathbf{r}_{i,3}^n, \ldots, \mathbf{r}_{i,\ell}^n, \mathbf{r}_{i,1}^n\}$$

ACS Macro Lett., ***2021***, 10, 1, 129-134.
M. P. Allen, D. J. Tildesley, *Computer Simulation of Liquids 2thed.*, ***2017***

Rosenbluth sampling

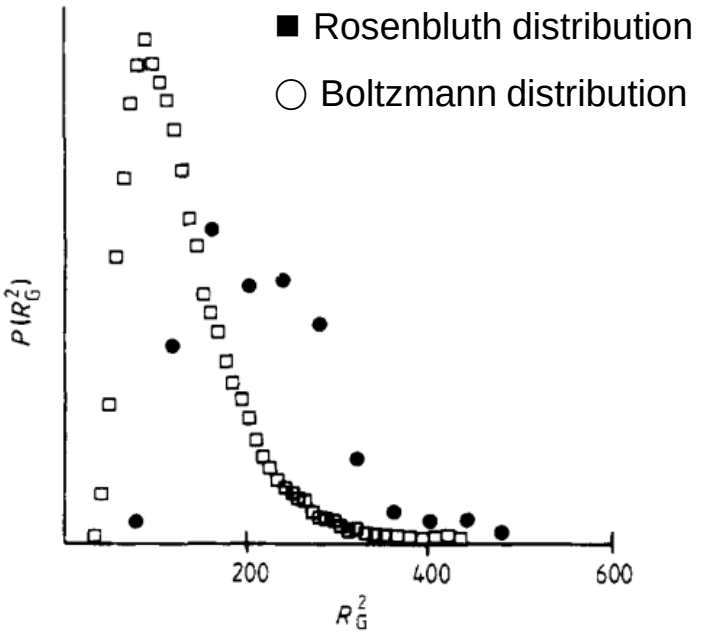
Acceptance probability for inserting/deleting single monomer is still very low!

How about biasing the chain generation by favoring the direction with the largest Boltzmann factor?



$$w_1 = k \exp[-\beta u^{(1)}(n)]$$

$$p^{(i)}(n) = \frac{\exp[-\beta u^{(i)}(n)]}{w_i}$$



Note that the distribution generated with the Rosenbluth procedure is not the Boltzmann distribution, but so called the "**Rosenbluth distribution**". (We need to recover it by reweighting)

D. Frenkel, B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications 3rd ed.,* ***2023***

Configurational-Bias Monte Carlo (CBMC)

We can **bias** the MC trial moves to enhance the MC sampling.

$$\alpha(o \to n) = f[\mathcal{U}(n)] \qquad \frac{\mathrm{acc}(o \to n)}{\mathrm{acc}(n \to o)} = \frac{f[\mathcal{U}(o)]}{f[\mathcal{U}(n)]} \exp\{-\beta[\mathcal{U}(n) - \mathcal{U}(o)]\}.$$
$$\alpha(n \to o) = f[\mathcal{U}(o)].$$

Where f[u] is the probability to generate trial configurations. The acceptance rule is :

$$\mathrm{acc}(o \to n) = \min\left(1, \frac{f[\mathcal{U}(o)]}{f[\mathcal{U}(n)]} \exp\{-\beta[\mathcal{U}(n) - \mathcal{U}(o)]\}\right)$$

Introducing an arbitrary biasing function f[u], we can generate a Boltzmann distribution with modified acceptance rule: **Configurational-Bias Monte Carlo (CBMC)**

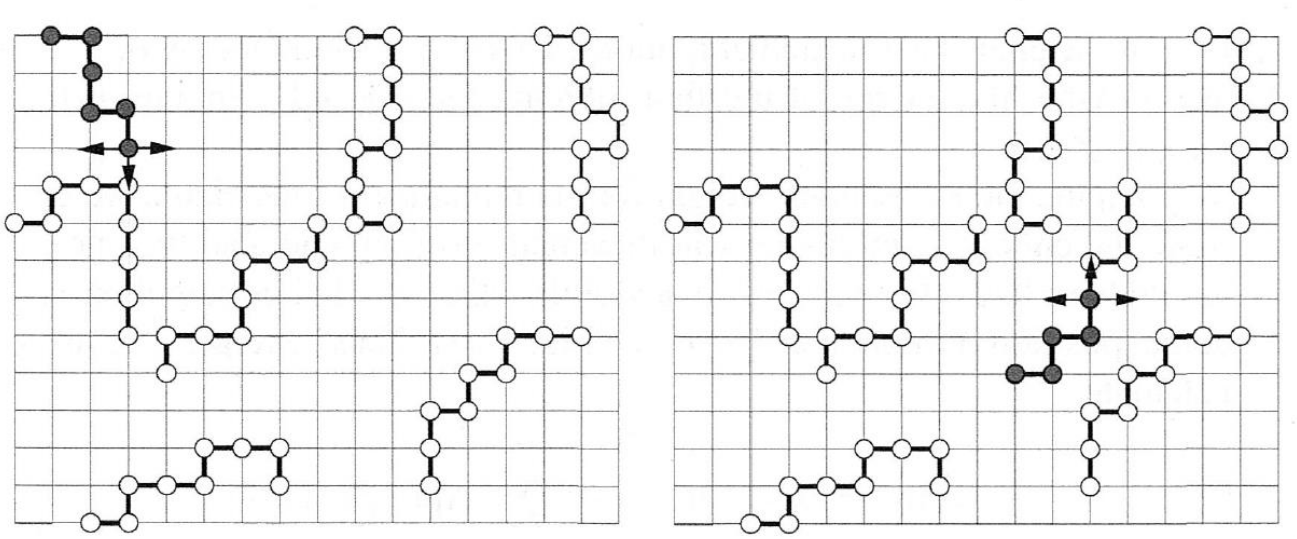
In Rosenbluth scheme, this biasing function is actually the Rosenbluth weight W(n)

D. Frenkel, B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications 3rd ed.,* ***2023***

Configurational-Bias Monte Carlo (CBMC) + Rosenbluth scheme

CBMC + Rosenbluth algorithm

1. Generate a trial conformation using the Rosenbluth scheme to grow the entire molecule
2. Compute its Rosenbluth weight W(n)
3. Retrace the old conformation and determine its Rosenbluth weight W(o)
4. Accept the trial move with a probability $\mathrm{acc}(o \to n) = \min[1, W(n)/W(o)]$



Now, we can generate the true representative configurations following Boltzmann distribution via Rosenbluth scheme!

D. Frenkel, B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications 3rd ed.,* ***2023***

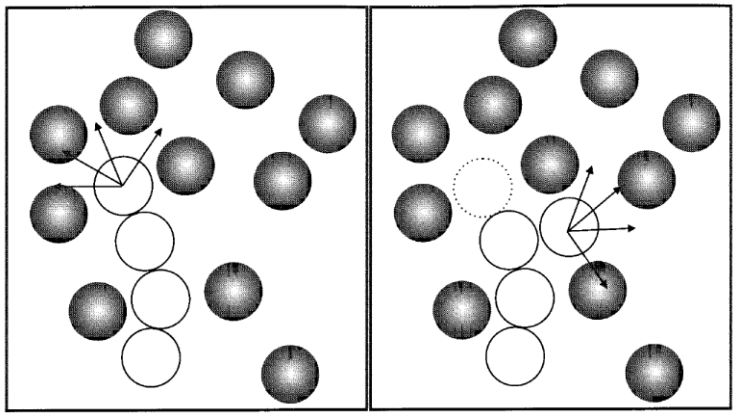
Recoil Growth (RG) scheme

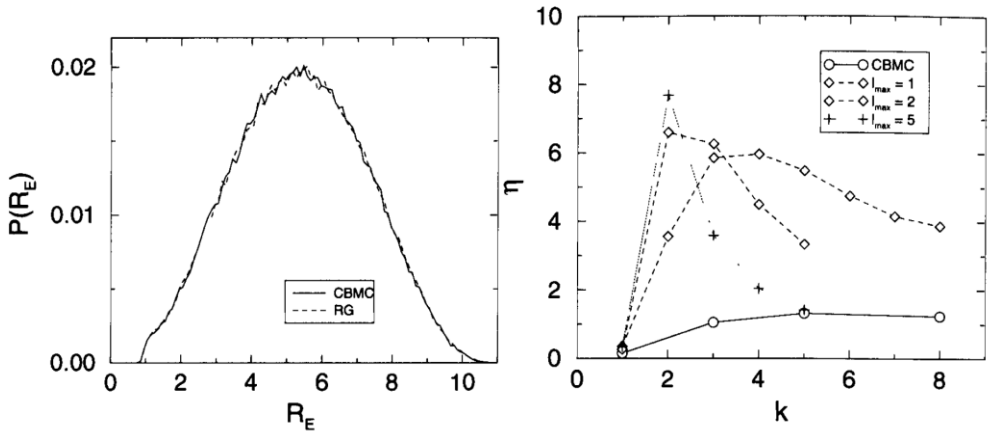
In the configurational-bias scheme, one has to "grow" an entire chain molecule and calculate its Rosenbluth weight, before a trial move can be accepted or rejected.

However, if one of the first segments during this growth was placed at an unfavorable position ("dead alley"), then no matter how often we try, the insertion will not be accepted.

(The main reason why CBMC method becomes inefficient in condensed phases)

In **recoil growth (RG) algorithm**, it uses a chap test to check if a given branch will die within a specific number of steps (l_{max})

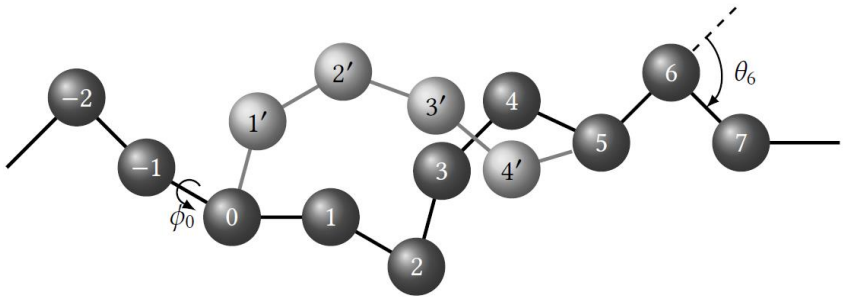




Concerted rotation MC

CBMC method only works from the end of the polymer by threading segments

How about suggesting trial moves that rearrange the conformation of interior segments?

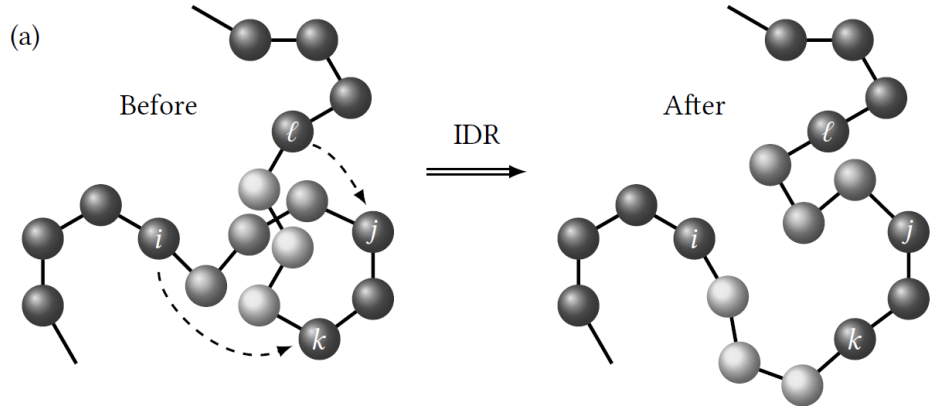


Propose trial move with the **concerted rotation** configuration by randomly selecting the trial state torsional angle $\{\phi_0^n, \phi_1^n, \ldots, \phi_6^n\}$

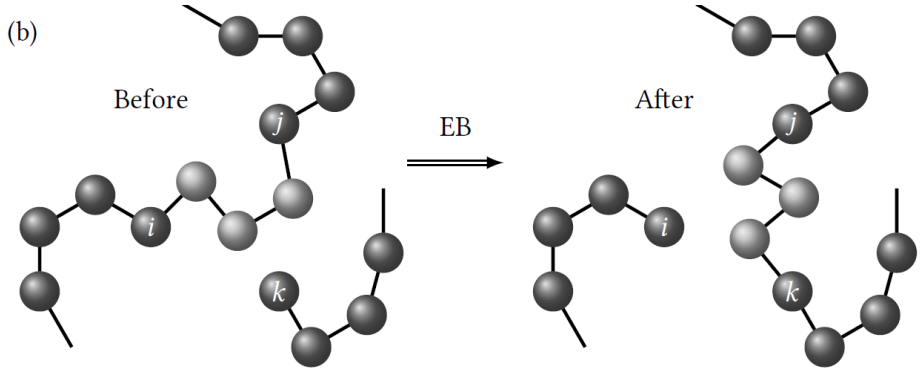
Accept the trial move with a probability give by

$$\min\left[1, \frac{N_m J_n}{N_n J_m} \exp(-\beta \delta \mathcal{V}_{nm})\right]$$

Intramolecular Double Rebridging (IDR) move exchanges the triplets



End Bridging (EB) move extracts the triplet and attach it to another



Content

Motivation: Why Enhanced MC?

MC simulation: Chain molecules

MC simulation: Condensed phases

MC simulation: Acceleration

Cavity-Biased GCMC

In the convention MC, all the molecules are moved with equal probability, in directions chosen at random

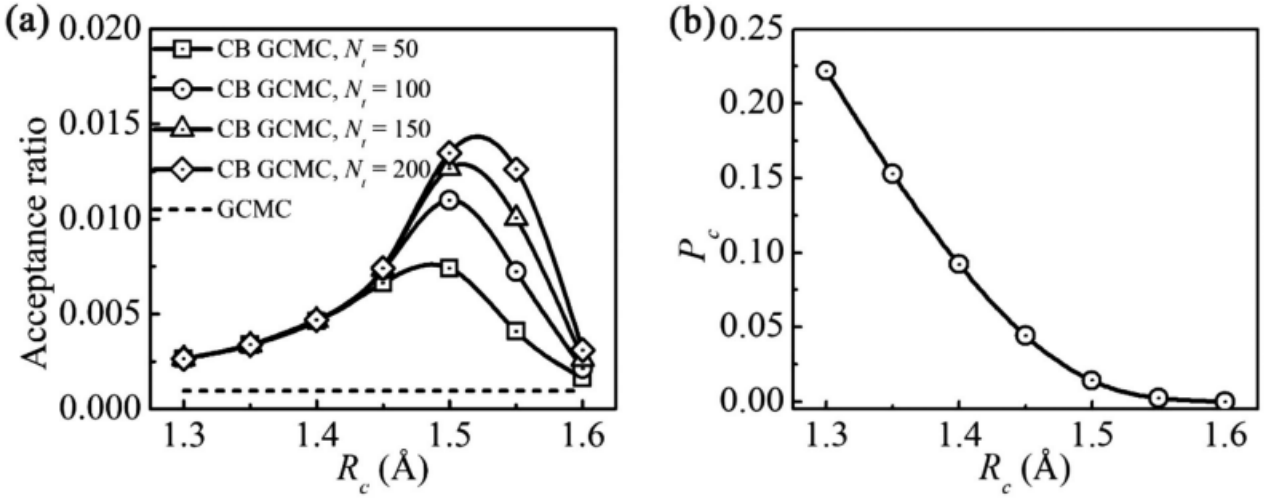
For running MC simulation in condensed phase, spontaneously inserting/deleting particles are difficult.

In the cavity-biased MC,

1. First generate randomly distributed test points
2. then, compute the probability of the positions of sites {r} that inside cavities as P_c
3. Insertion is then attempted at a randomly selected position in {r}. (If no site available, random insertion)

$$acc^{CB}(N \to N+1) = \min\left(1, \frac{Vp_c^{N+1}}{\Lambda^3(N+1)} \exp\{-\beta[\mu - U(N+1) + U(N)]\}\right)$$

$$acc^{CB}(N \to N-1) = \min\left(1, \frac{\Lambda^3 N}{Vp_c^{N}} \exp\{-\beta[\mu + U(N-1) - U(N)]\}\right).$$

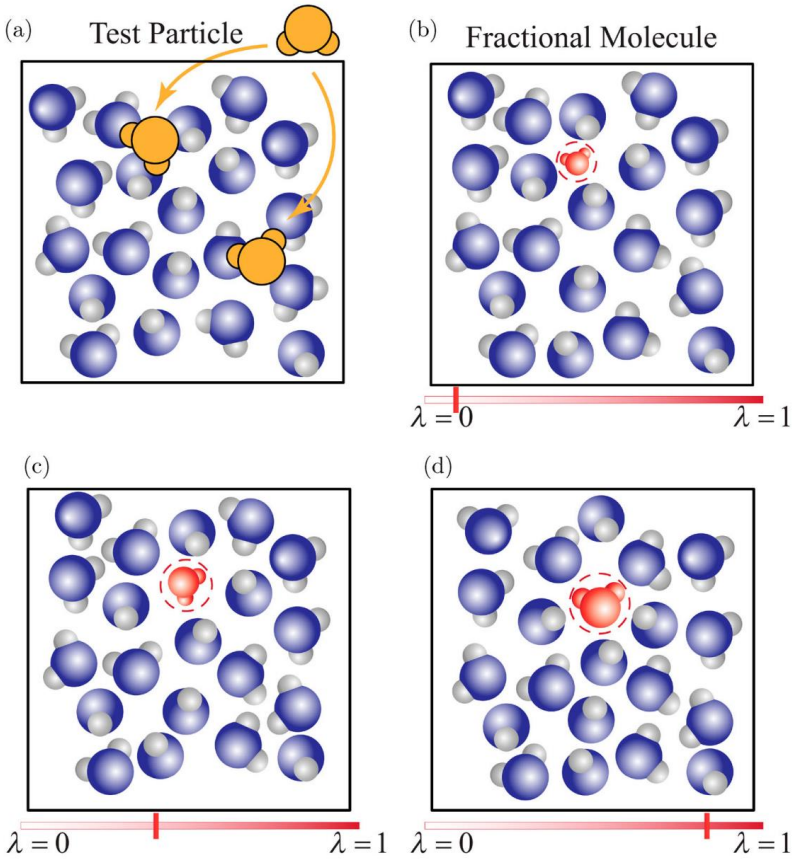


Mol. Sim., **2020,** 46, 10, 736-742.

Continuous Fractional Component Monte Carlo (CFCMC)

Q: Why energy penalty for particle insertion/deletion is so high?

A: Majorly from nonbonded interaction (LJ/elec)



Then, let's scale down the nonbonded interaction between test particle and existing particle by λ.

$$u_{LJ}(r,\lambda_{LJ}) = \lambda_{LJ}^{a} 4\epsilon \left[\left(\frac{1}{\alpha(1-\lambda_{LJ})^{b} + (r/\sigma)^{c}} \right)^{12/c} - \left(\frac{1}{\alpha(1-\lambda_{LJ})^{b} + (r/\sigma)^{c}} \right)^{6/c} \right]$$

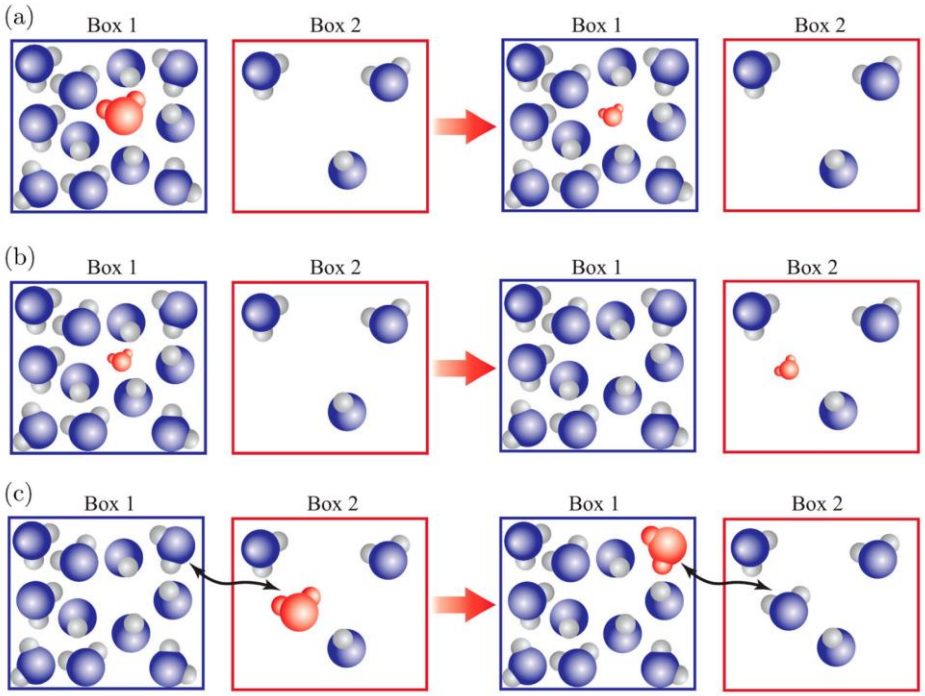
DO NOT ADD/DELETE PARTICLE BY SINGLE STEP.

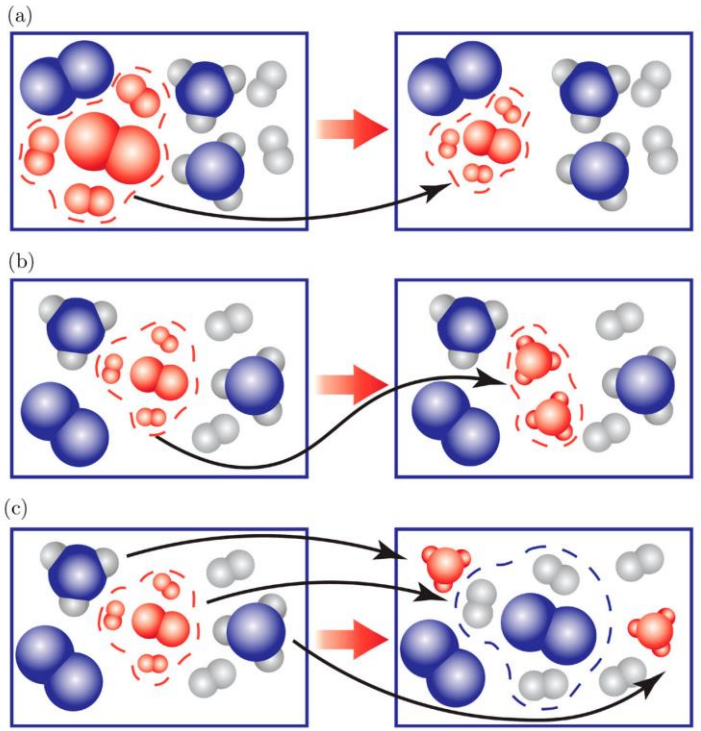
Random walks in λ-space via MC simulation.
("Expanded ensemble")

Give some 'space' for test particle to grow or fade in the system by **multiple MC steps.**

Mol. Sim., ***2021***, 47, 10, 804-823.

CFCMC with Gibbs ensemble and reaction ensemble





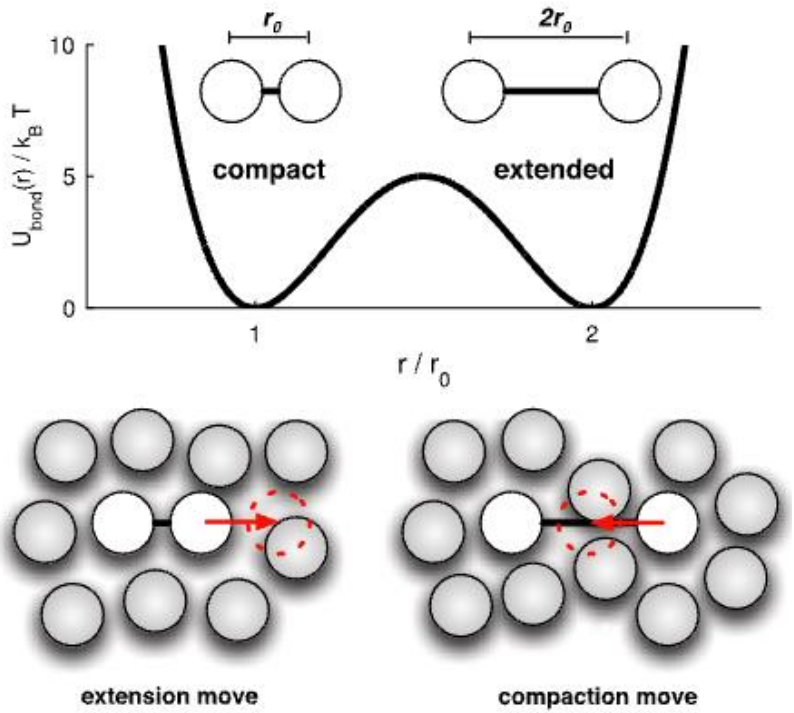
Mol. Sim., ***2021***, 47, 10, 804-823.

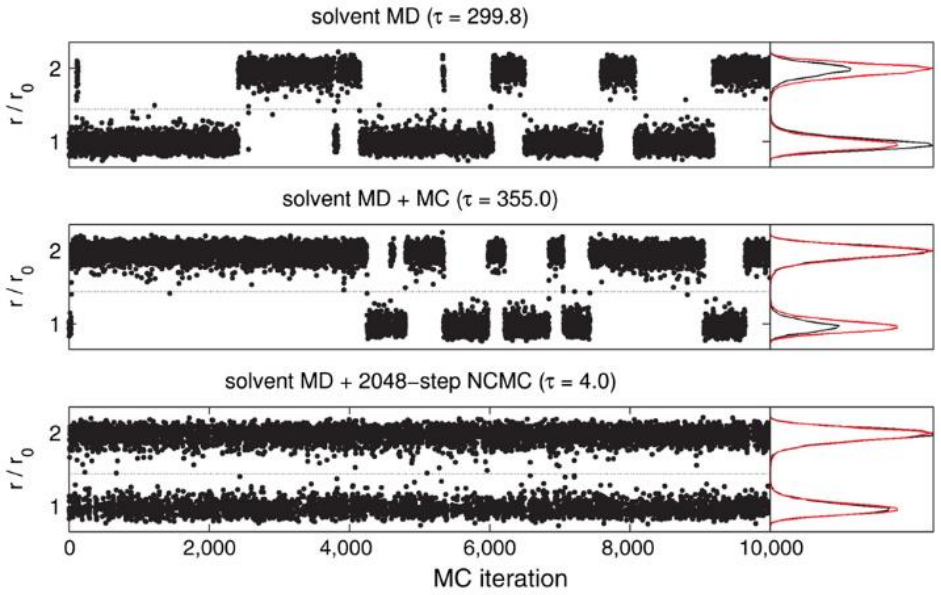
Nonequilibrium Candidate Monte Carlo (NCMC)

How about ensuring some times for solvent to relax and reorganize to avoid overlapping with solute?

MC move -> propagation (solvent relaxation) -> MC move -> propagation (solvent relaxation) -> ….

Note that propagation from noneq candidate requires noneq work, which will be incorporated in acceptance criteria. (This is implemented in OpenMMtools as GHMC Integrator)

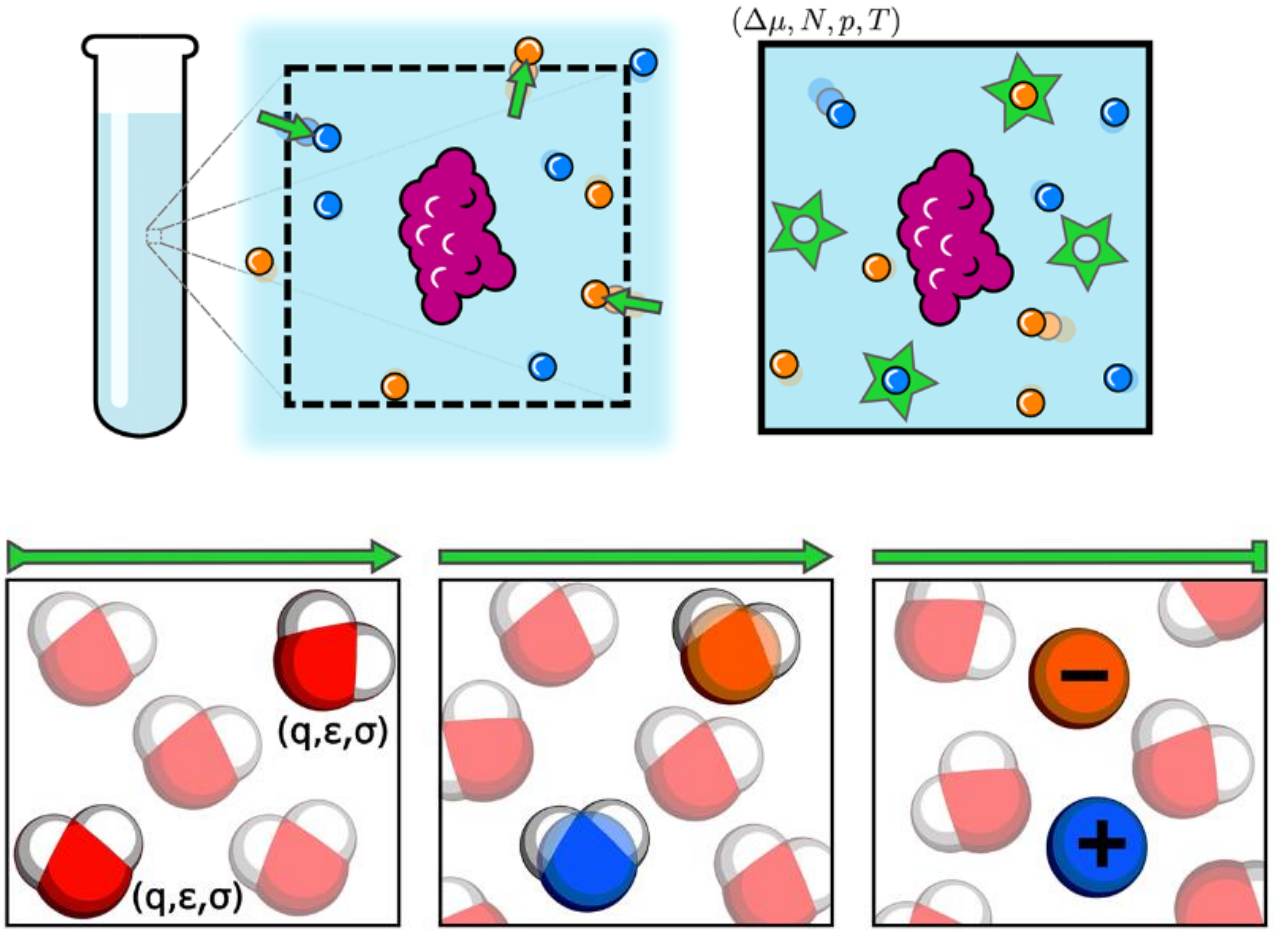




PNAS, **2011,** 108, 45, 1009-1018.

Semigrand canonical Monte Carlo (SGMC) for osmostat

Combination of NCMC and CFCMC to sample the osmotic ensemble

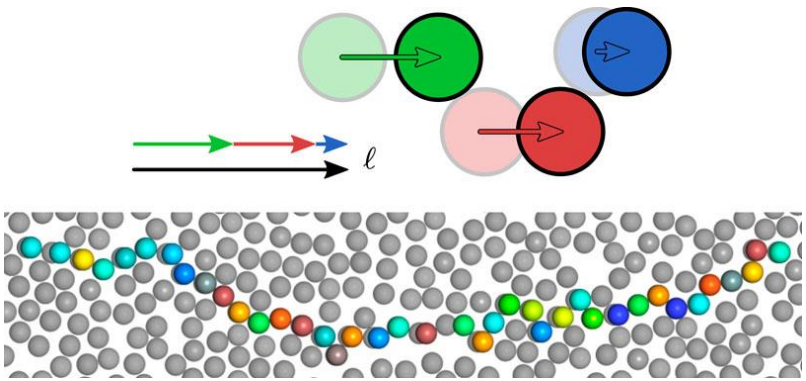


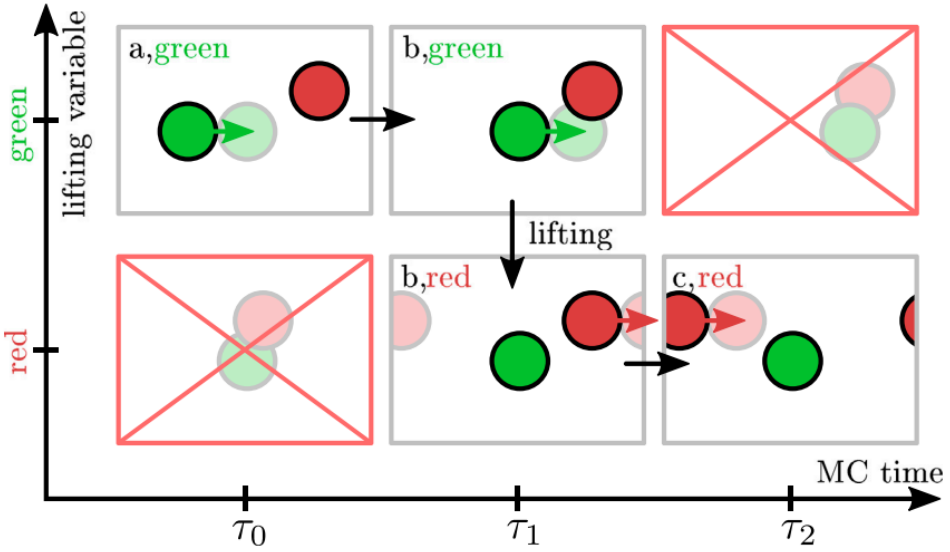
J. Phys. Chem. B, ***2018***, 122, 21, 5466-5486.

Content

Event-Chain Monte Carlo (ECMC)

Event-chain Monte Carlo(ECMC) attempts a trial move in the positive x-direction only, and if there is an overlap, “passing the baton”. (Lifting move)





Q. Why this lifting is better than conventional Metropolis MC algorithms?

A. In Metropolis algorithm, density heterogeneities decay by diffusion, which is slow in high density regions.

In ECMC algorithm, a particle that hits a high-density region can lead to a sound-like propagation through that region

Remark. This MC algorithm satisfies balance condition **but not for detailed balance condition**, since particle only moves in a positive direction. Such irreversible MC algorithms are one important strategy to enhance the sampling.

Front in Phys., ***2021***, 9, 635886.

Enhanced sampling by mapping

If we set the mapping from a set of reference coordinates to the real coordinates, then the ratio of the old and new Jacobians ends up in the acceptance rule.

System B with a harder-to-sample potential energy function $U_B(q')$ and Boltzmann distribution $\rho_B(q')$
System A with a easy-to-sample potential energy function $U_A(q)$ and Boltzmann distribution $\rho_A(q)$

The probability density generated by sampling A and transforming from q to q' is

$$\rho(q') \sim \frac{\exp(-\beta \mathcal{U}_A)}{|J|_T} \qquad |J|_T \equiv \left| \frac{\partial q'}{\partial q} \right|$$

If we construct a transformation from q to q' as T such that $|J|_T = \rho_A(q)/\rho_B(q')$

It's possible that our sampling of A automatically yields all points in B with the correct Boltzmann weight

This implies that if we chose a suitable coordinate transformation, then we may improve MC sampling!

But constructing such a perfect transformation is not feasible in general.

Normalizing flow: Boltzmann generator

In ML, there is a class of machine-learned invertible mapping known as the “normalizing flows”

We train a neural network to find a transformation that maps the **normal latent space distribution** into the actual molecular coordinates.

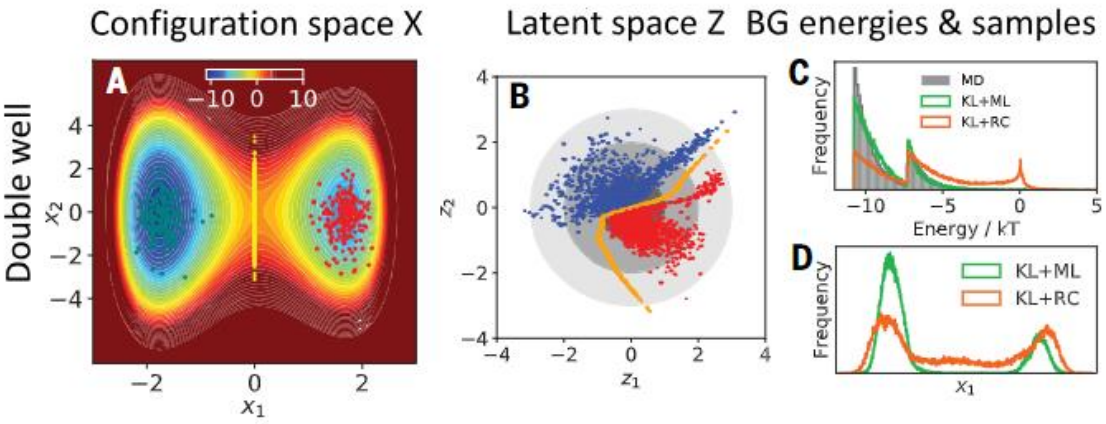
1. Sample batch $\{\mathbf{x}_1, ..., \mathbf{x}_B\}$ from X.
2. Update Boltzmann generator parameters θ by training on batch.
3. For each $\mathbf{x}$ in batch, propose a Metropolis Monte Carlo step in latent space with step size s:

$$\mathbf{z}' = T_{xz}(\mathbf{x}) + s\mathcal{N}(0, \mathbf{I}).$$

4. Accept or reject proposal with probability $\min\{1, \exp(-\Delta E)\}$ using:

$$\Delta E = u\Big(T_{zx}(\mathbf{z}')\Big) - u(\mathbf{x}) - \log R_{zx}(\mathbf{z}'; \theta) + \log R_{xz}(\mathbf{x}; \theta)$$

5. For the accepted samples, replace $\mathbf{x}$ by $\mathbf{x}' = T_{zx}(\mathbf{z}')$.



This latent space MC is still “work in progress”.

For higher-dimensional problems (liquids), the mapping starting from a **normal distribution** tends to run into trouble.

Open question: How can we create mappings from a reference state that is more similar to the target state work better?

*Science, **2019**, 365, eaaw1147.*

Takeaways

■ MC simulation serves room for designing efficient moves to enhance the sampling.

■ One issue is the low acceptance probabilities for MC particle insertion/deletion move for chain molecules or in condensed phases.

■ For chain molecules, MC sampling is enhanced by CBMC (+Rosenbluth sampling) and concerted rotation MC move.

■ In condensed phase, MC moves with high acceptance probabilities can be achieved by cavity-biased MC, CFCMC, NCMC, or SGMC.

■ Tailoring MC scheme (early rejection/event-chain) or constructing invertible mappings are the strategies to enhance the MC simulation.

Q&A