Short review on Monte Carlo algorithm

Hiromasa Watanabe (渡辺展正)
(Keio University, Japan)

2025/12/04
Matrix Model for Superstring/M-theory @ YITP, Kyoto U.

References

Textbooks

- M. Hanada, and S. Matsuura, *MCMC from Scratch — a practical introduction to Markov Chain Monte Carlo method*, Springer 2022.
- B. Ydri, *Matrix Models of String Theory*, IOP Publishing, 2018.
- C. Gattringer and C. B. Lang, *Quantum Chromodynamics on the Lattice — an Introductory Presentation*, Springer 2009.

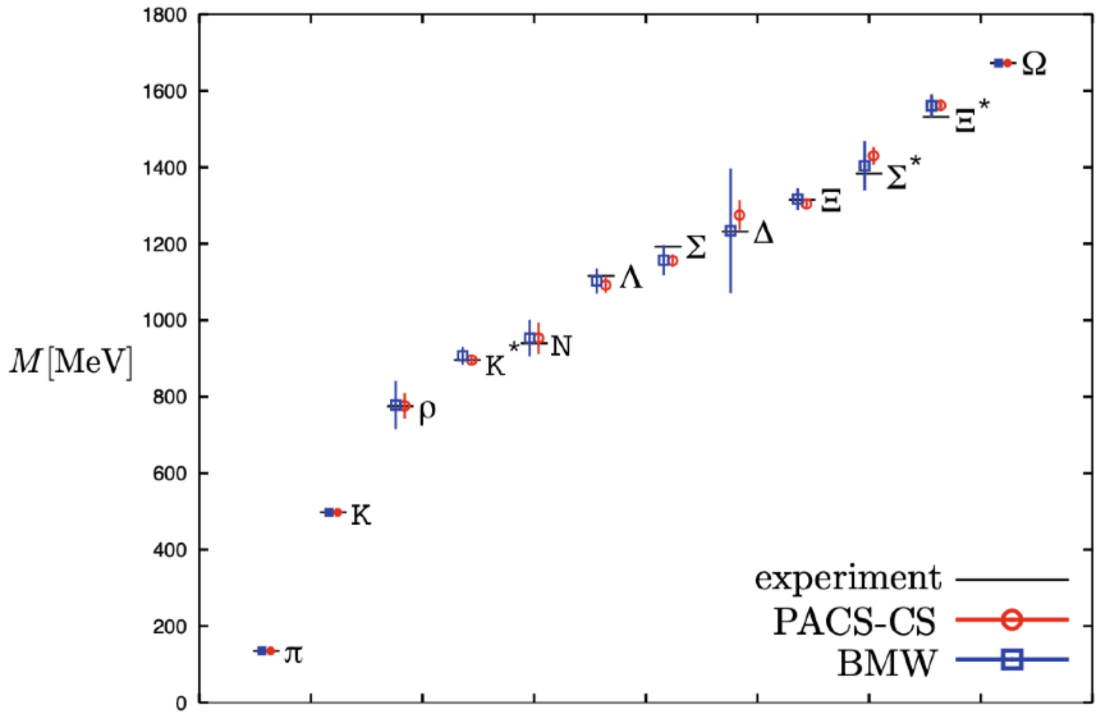
Reviews

- M. Hanada, *Markov Chain Monte Carlo for Dummies*, arXiv:1808.08490[hep-th].
- …

Introduction

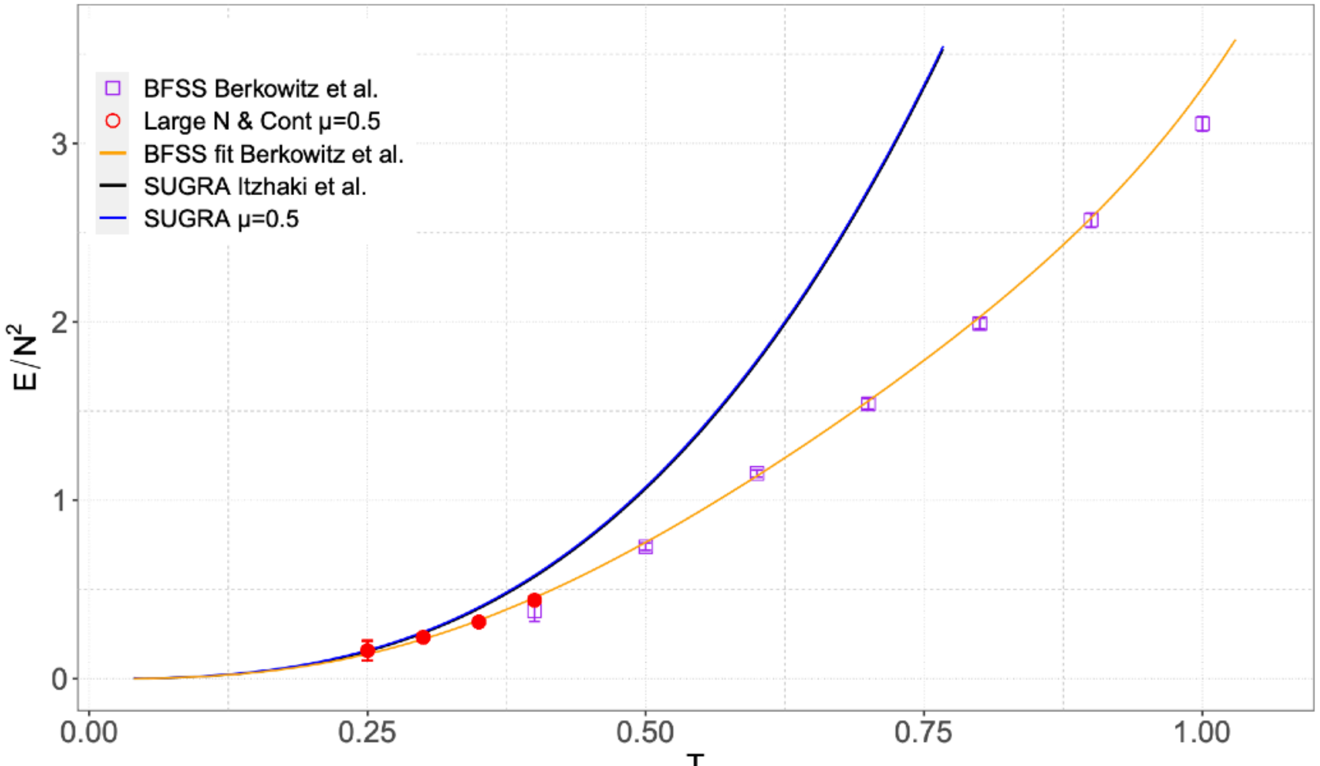
- A success of Monte Carlo in QCD w/ lattice regularization

  ← unveiling a nonperturbative aspect of QFT



[Textbook by Hanada, Matsuura,'22]

- The technology is applicable to matrix models!



from [Pateloudis et al. (MCSMC), '23]

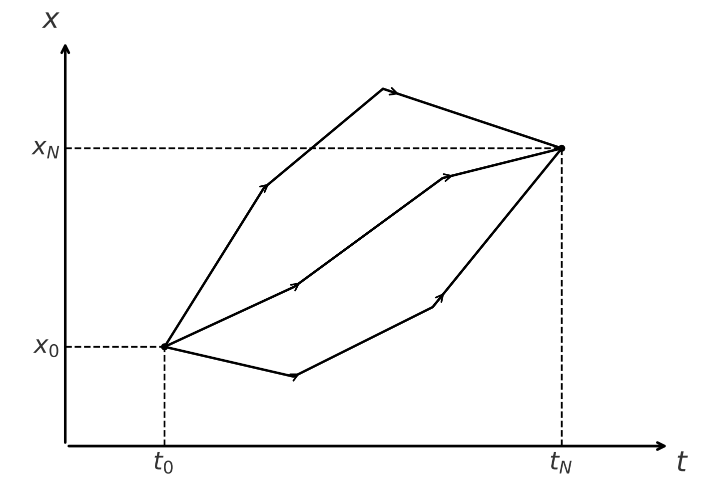
Path-integral formalism

Monte Carlo algorithm is based on the path-integral formalism:

Ex) 1d QM

$$\langle x_{\rm F}, t_{\rm F} | x_{\rm I}, t_{\rm I} \rangle = \int {\rm d}x_{N-1} \cdots {\rm d}x_1 \, \langle x_{\rm F} | {\rm e}^{-{\rm i}H\Delta t} | x_{N-1} \rangle \cdots \langle x_1 | {\rm e}^{-{\rm i}H\Delta t} | x_{\rm I} \rangle$$

$$= \int_{x(t_{\rm I})=x_{\rm I}}^{x(t_{\rm F})=x_{\rm F}} \mathscr{D}x \, {\rm e}^{{\rm i}S[x]} \xrightarrow{\tau={\rm i}t} \int_{x(\tau_{\rm I})=x_{\rm I}}^{x(\tau_{\rm F})=x_{\rm F}} \mathscr{D}x \, {\rm e}^{-S_{\rm E}[x]}$$



$$Z(\beta) = {\rm tr}\left({\rm e}^{-\beta H}\right) = \int {\rm d}x \, \langle x | {\rm e}^{-\beta H} | x \rangle = \int_{x:\text{periodic}} \mathscr{D}x \, {\rm e}^{-S_{\rm E}[x]} \qquad \langle O \rangle = \frac{1}{Z} \int \mathscr{D}x \, O(x) \, {\rm e}^{-S_{\rm E}[x]}$$

Similarly,

$$Z = \int \mathscr{D}\Phi \, {\rm e}^{-S_{\rm E}[\Phi]}, \quad \langle O \rangle = \frac{1}{Z} \int \mathscr{D}\Phi \, O(\Phi) \, {\rm e}^{-S_{\rm E}[\Phi]}$$

- For IKKT type (0+0 dim): $\Phi = \left\{X_{I,ij}\right\}$, matrix integral
- For BFSS type (0+1 dim): $\Phi = \left\{X_{I,ij}(t_n)\right\}$, matrix quantum mechanics

Curse of dimensionality

Let's estimate the cost of the multivariable integral by direct computation.

$$Z = \int \mathscr{D}\Phi\, \mathrm{e}^{-S_{\mathrm{E}}[\Phi]}, \quad \langle O \rangle = \frac{1}{Z}\int \mathscr{D}\Phi\, O(\Phi)\, \mathrm{e}^{-S_{\mathrm{E}}[\Phi]}$$

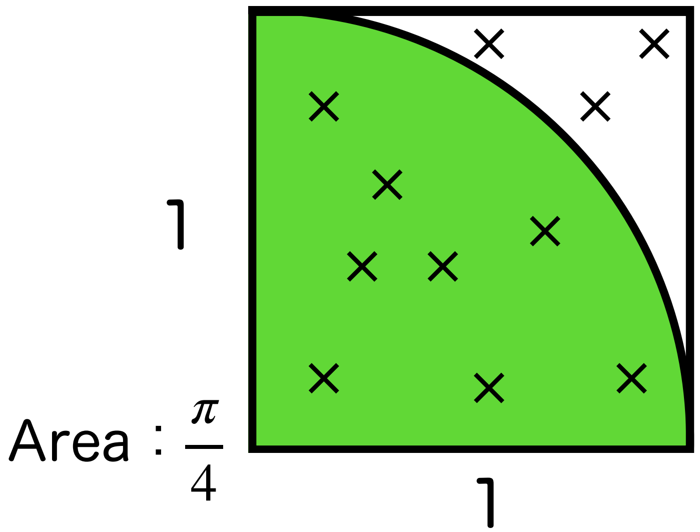
e.g.) Bosonic BFSS model (9 scalars) w/ $N = 64,\ L = 32$ (L=#(temporal sites))

- The path integral is (#dof)-dimensional, (#dof) $= 64 \times 64 \times 32 \times 9 \simeq 1.2 \times 10^6$.
- Integral can be approximated by a discrete sum.
  Even approximating each dof by 8 pts., totally $8^{1.2\times10^6} \simeq 10^{1.1\times10^6}$ pts.
- Using HPC (w/ GHz CPU), it takes $\sim 10^{1.1\times10^6-9} \simeq 10^{1.1\times10^6}$ sec. for the sum!!
  c.f.) Age of the universe : 13.8 billion yrs. ~ 4.4×10^{17} sec.

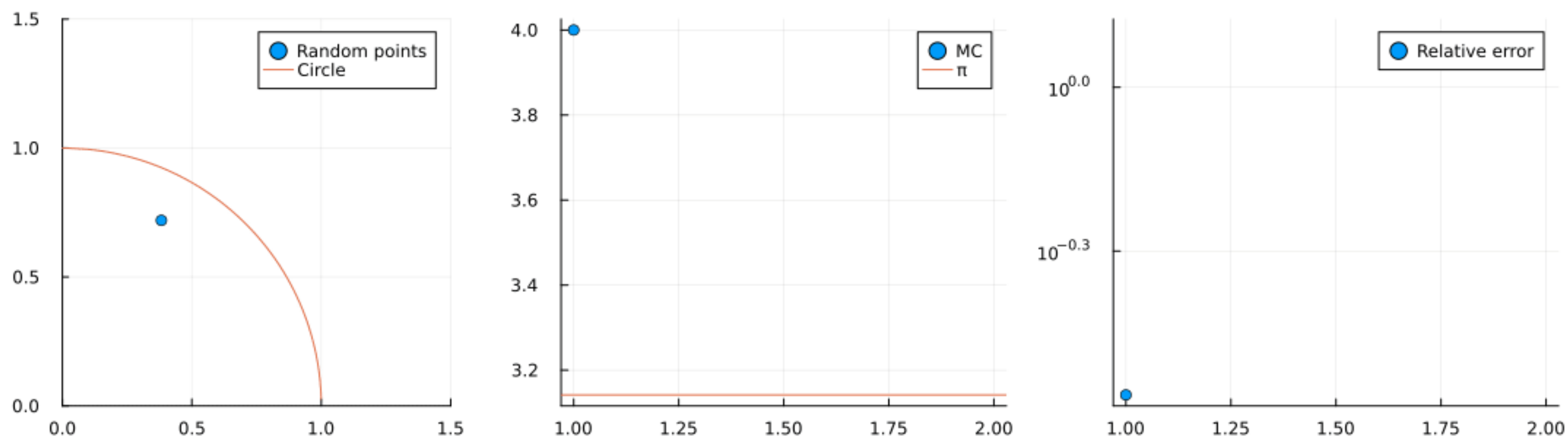
→ A more clever way: Monte Carlo

A Naive example: evaluating π

Idea of Monte Carlo: Computation of integrals by “rolling dice”



- x, y : random numbers b/w 0 and 1
- (pt. in $x^2 + y^2 \leq 1$)/(#trial) ≈ (Area of green region)
- More accurate with many trials



A Naive example: evaluating π

$\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\}$: sampling data ($x = x_\mu$) generated by the prob. density $P(x)$

$$\langle f(x) \rangle = \int [\mathrm{d}x]\, f(x) P(x) = \lim_{N_{\text{sample}} \to \infty} \frac{1}{N_{\text{sample}}} \sum_{n=1}^{N_{\text{sample}}} f(\{x^{(n)}\}) \approx \frac{1}{N_{\text{sample}}} \sum_{n} f(\{x^{(n)}\})$$

e.g.) π estimation

$$I = \int_0^1 \mathrm{d}x\, f(x) = \int_0^1 \mathrm{d}x\, \frac{f(x)}{P(x)} P(x) = \left\langle \frac{f(x)}{P(x)} \right\rangle$$

$$f(x) = 4\sqrt{1 - x^2}$$

$$P(x) = 1, \quad x \in [0,1]$$

Too primitive, but this can overcome the curse of dimensionality.

For path integral,

$$\langle O \rangle = \frac{1}{Z} \int [\mathrm{d}x]\, O(x)\, \mathrm{e}^{-S[x]}, \quad P(x) \sim \frac{\mathrm{e}^{-S[x]}}{Z}$$

Sampling w/ $P(x)$ efficiently based on importance sampling → Markov-Chain MC

Markov-Chain Monte Carlo (MCMC)

Markov Chain: $\Phi^{(n)} \xrightarrow{P} \Phi^{(1)} \xrightarrow{P} \cdots \xrightarrow{P} \Phi^{(N_{\text{sample}})}$

- $\Phi^{(n)}$ depends only on the sample one step before $\Phi^{(n-1)}$.

Properties of MC

Irreducibility: any two states can be connected by a finite step.

Aperiodicity: any state has a periodicity of one.

- periodicity : gcd of possible MC steps coming back to itself.

Sufficient condition: detailed balance

$P(\Phi)T(\Phi \to \Phi') = P(\Phi')T(\Phi' \to \Phi)$ $\quad$ $T(\Phi \to \Phi')$: transition probability

- implying the probability distribution is in “equilibrium.”

Metropolis(-Hastings) algorithm

A well-established MCMC implementation

For one variable x, $$P(x) = \frac{\mathrm{e}^{-S(x)}}{Z}$$

1. Set an initial value $x^{(0)}$
2. In a certain way, generate a candidate x' from the config. (sample) $x^{(p)}$

   e.g.) $$x' = x^{(p)} + \epsilon\left(y - \frac{1}{2}\right)$$ ϵ : const、$y \in [0,1]$ uniform random

3. Metropolis test: determine if x' is accepted/rejected.

   Preparing $r \in [0,1]$,

   $$\min\left(1, W(x', x^{(p)})\right), \qquad W(x', x^{(p)}) = \mathrm{e}^{S(x^{(p)})-S(x')} = \frac{\mathrm{e}^{-S(x')}}{\mathrm{e}^{-S(x^{(p)})}}$$

   $x^{(p+1)} = x'$ if $r < \min(1, W(x', x^{(p)}))$, otherwise $x^{(p+1)} = x^{(p)}$.

4. Repeating step 2 & 3 generates $x^{(0)} \to x^{(1)} \to x^{(2)} \to \cdots$
5. Switching to data analysis.

Proof of satisfying the detailed balance

Let us consider a generalized case.

A probability generating x' from $x^{(p)}$: $Q(x^{(p)}, x') \neq Q(x', x^{(p)})$

$$W = \frac{P(x')Q(x', x^{(p)})}{P(x^{(p)})Q(x^{(p)}, x')} \qquad \text{— } (\star)$$

Transition probability when rejected: $x^{(p+1)} = x^{(p)}$ and trivially satisfied.

Transition probability when accepted:

- for $W \geq 1$: $T(x^{(p)} \to x') = Q(x^{(p)}, x')$
- for $W < 1$: $T(x^{(p)} \to x') = WQ(x^{(p)}, x')$

$$\to P(x^{(p)})T(x^{(p)} \to x') = \min(1, W)P(x^{(p)})Q(x^{(p)}, x') \overset{(\star)}{=} \min(1, 1/W)P(x')Q(x', x^{(p)})$$

Same computation leads $P(x')T(x' \to x^{(p)}) = \min(1, 1/W)P(x')Q(x', x^{(p)})$.

Hybrid Monte Carlo (HMC)

Basic idea:

$$\langle \mathcal{O}(x) \rangle = \frac{1}{Z} \int \mathrm{d}x \ \mathcal{O}(x)\, \mathrm{e}^{-S(x)} \cdot \frac{\int \mathrm{d}p\, \mathrm{e}^{-\frac{p^2}{2}}}{\int \mathrm{d}p\, \mathrm{e}^{-\frac{p^2}{2}}} = \frac{1}{Z_{\mathrm{H}}} \int \mathrm{d}x \mathrm{d}p \ \mathcal{O}(x)\, \mathrm{e}^{-H(x,p)} \qquad H(x,p) = \frac{1}{2}p^2 + S(x)$$

Update w/ a distribution $\quad P(x,p) = \dfrac{\mathrm{e}^{-H(x,p)}}{Z_{\mathrm{H}}}$

- Molecular Dynamics (MD) phase

  Solve the Hamilton eq. w/ a fictitious time τ and $\left\{ x(\tau=0) = x^{(n)},\, p(\tau=0) = p^{(n)} \right\}$

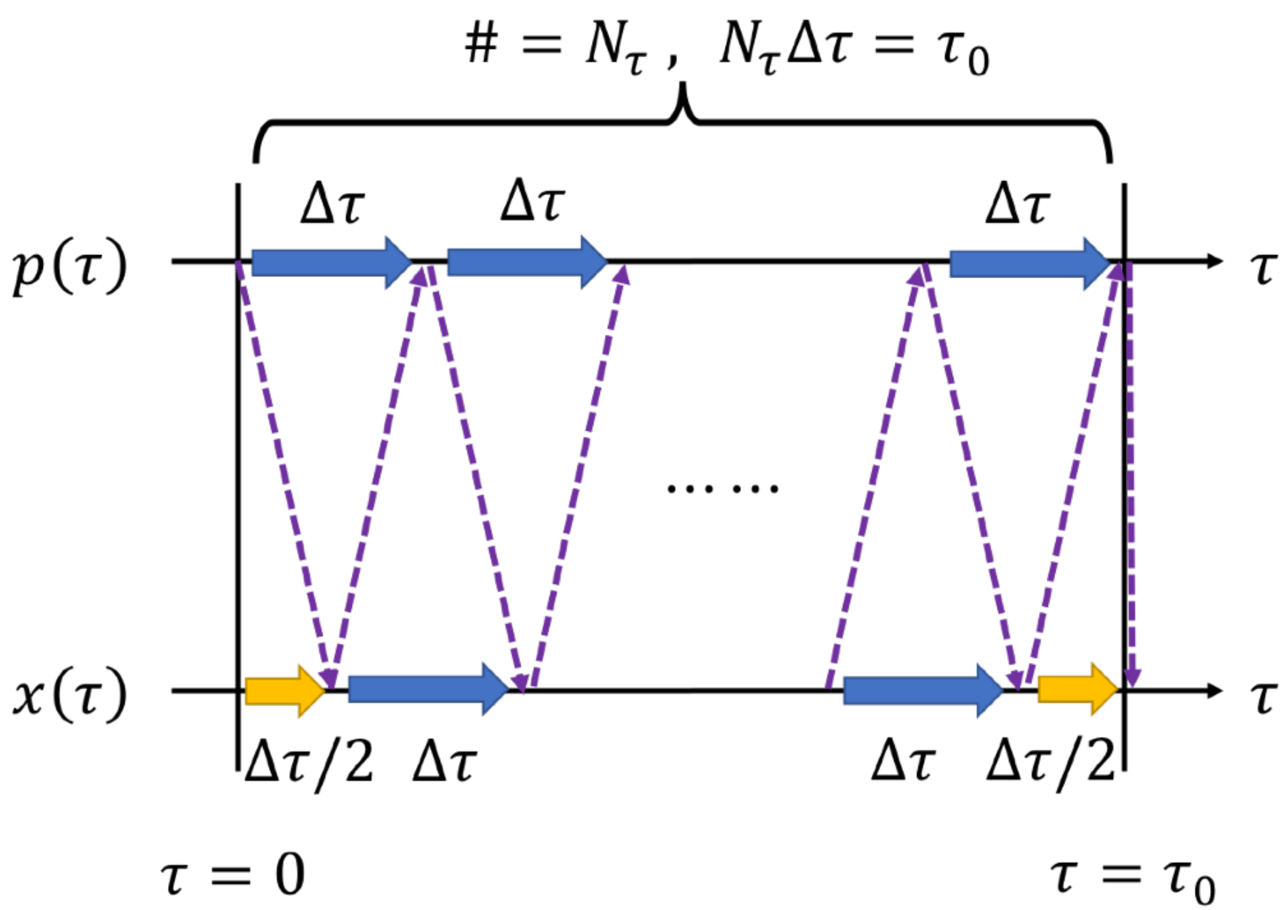
$$\frac{\mathrm{d}x(\tau)}{\mathrm{d}\tau} = \frac{\partial H}{\partial p} = p, \quad \frac{\mathrm{d}p(\tau)}{\mathrm{d}\tau} = -\frac{\partial H}{\partial x} = -\frac{\partial S(x)}{\partial x}$$

  and the candidate $\{x', p'\}$ is checked by Metropolis test.

- Satisfying the detailed balance condition (c.f. next slide)
- Discretization effect for τ can be absorbed.
- Applicable to a variety of models.

Leapfrog method

Discretization of MD phase respecting a time reversibility.



$$x_1 = x_0 + p_0 \cdot \frac{\Delta\tau}{2}$$

$$p_1 = p_0 - \frac{\partial H(p_0, x_1)}{\partial x}\Delta\tau$$

$$x_{\tau+1} = x_\tau + p_\tau \Delta\tau$$

$$p_{\tau+1} = p_\tau - \frac{\partial H(p_\tau, x_{\tau+1})}{\partial x}\Delta\tau$$

$$x_{N_\tau} = x_{N_\tau-1} + p_{N_\tau-1}\Delta\tau$$

$$p_{N_\tau} = p_{N_\tau-1} - \frac{\partial H(p_{N_\tau-1}, x_{N_\tau})}{\partial x}\Delta\tau$$

$$x_{N_\tau+1/2} = x_{N_\tau} + p_{N_\tau} \cdot \frac{\Delta\tau}{2}$$

Box Müller method

"Conjugate momentum" is sampled from the Gaussian distribution.

Box Müller method: Gaussian random num. from uniform random num.

Suppose $x, y \in [0,1]$ and

$$r = \sqrt{-2\ln x} \qquad \theta = 2\pi y \quad \Rightarrow \quad \xi = r\cos\theta \qquad \eta = r\sin\theta$$

$\rightarrow$ ξ, η are Gaussian random numbers.

$\because$) Computing the Jacobian,

$$\mathrm{d}x\mathrm{d}y = \frac{1}{2\pi}\mathrm{e}^{-\frac{r^2}{2}} r\mathrm{d}r\mathrm{d}\theta = \frac{1}{2\pi}\mathrm{e}^{-\frac{1}{2}(\xi^2+\eta^2)}\mathrm{d}\xi\mathrm{d}\eta = P(\xi)\mathrm{d}\xi P(\eta)\mathrm{d}\eta$$

- A good example to generate a nontrivial target distrib. from a simple base one
- Similar idea can be found in ML technique (e.g. Normalizing flow)

Notes on HMC

- To compute the force terms for matrices,

$$\frac{\mathrm{d}P_{ij}}{\mathrm{d}\tau} = -\frac{\partial S(X)}{\partial X_{ji}} \quad \rightarrow \quad P_{ij} \leftrightarrow X_{ji}$$

- Hamiltonian must be conserved w/o discretization.
  → Monitoring the magnitude of breaking ΔH is useful for debugging.

- For 0+1d models (Gaussian MM, BFSS, ⋯)
    - Lattice discretization along time direction is also needed.
    - A gauge field is updated by MC, in addition to adjoint scalars X.
    - Since gauge dof does not propagate, we can fix all of them.
      → Vandermonde determinant appears.

Example: One-matrix model

For a hermitian $N \times N$ matrix X,

$$Z = \int [\mathrm{d}X]\, \mathrm{e}^{-V(x)}, \quad V(X) = N \sum_{n=1}^{\infty} c_n \operatorname{tr} X^n \qquad \mathrm{d}X = \prod_{i=1}^{N} \mathrm{d}X_{ii} \prod_{i<j}^{N} \mathrm{d}(\mathrm{Re}\, X)_{ij}\, \mathrm{d}(\mathrm{Im}\, X)_{ij}$$

Analytic solution for $c_2 = 1/2,\ c_{n\neq 2} = 0$:

$$\frac{1}{N}\langle \operatorname{tr} X^2 \rangle = \frac{1}{Z}\int [\mathrm{d}X]\ \operatorname{tr} X^2\, \mathrm{e}^{-V(X)} = 1$$

$$\rho(x) = \frac{1}{2\pi}\sqrt{4 - x^2}$$ at large N

e.g. [Eynard, Kimura, Ribault, '18]
or textbooks of RMT

[Exec. Environment]

Julia & Jupyter Notebook on Google Colaboratory

Hurdles for MC of matrix models

In addition to the case w/ large #(dof),

- Practical difficulty w/ fermions coming from fermion determinant
- Studying the following systems
    - supersymmetry (periodic. b.c. for fermions)
    - near critical points or continuum limit, flat directions
    - real-time evolution (Lorentzian)

Reasons:

- Auto correlation among field configurations
- Supersymmetry on the lattice
- Sign problem

Sign Problem

Sign problem disrupts performing path integral numerically

(e.g., real-time evolution, SUSY model, QCD at finite density, ⋯)

Monte Carlo simulation: generating samples w/ physical weight

$$\langle O(\Phi)\rangle = \frac{1}{Z}\int \mathscr{D}\Phi\, O(\Phi)\, \mathrm{e}^{-S[\Phi]} \approx \frac{1}{N_{\text{sample}}}\sum_n O(\Phi^{(n)}), \qquad \Phi^{(n)} \sim \frac{\mathrm{e}^{-S}}{Z} =: P(\Phi)$$

works when action is real and positive.

A naive refinement for complex actions: reweighting method

$$\langle O(\Phi)\rangle = \frac{\int \mathscr{D}\Phi\, \left(O(\Phi)\,\mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right)\mathrm{e}^{-\mathrm{Re}S[\Phi]}}{\int \mathscr{D}\Phi\, \left(\mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right)\mathrm{e}^{-\mathrm{Re}S[\Phi]}} = \frac{\left\langle O(\Phi)\,\mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right\rangle_{\mathrm{Re}S}}{\left\langle \mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right\rangle_{\mathrm{Re}S}}$$

$\mathrm{Im}S[\Phi] = 0 \;\Rightarrow\; \left\langle \mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right\rangle_{\mathrm{Re}S} = 1,$

$\mathrm{Im}S[\Phi]$ is highly oscillatory $\Rightarrow \left\langle \mathrm{e}^{-\mathrm{i}\mathrm{Im}S[\Phi]}\right\rangle_{\mathrm{Re}S} \approx 0$

Overcoming the sign problem

- Just reweighting
- Approximation by dropping the complex phase (:quenching)
- Analytic continuation from real variables
- Complexifying variables c.f. [Talk by Jun Nishimura & poster by Ashutosh Tripathi]
  - Complex Langevin equation c.f. [Review by K. Nagata, '21]
    - An extension of Parisi-Wu Stochastic quantization
  - (Generalized) Lefschetz thimble method
    - Deforming the integration contour by solving the holomorphic gradient flow eq.
  - Path optimization
    - Deforming the integration contour.
    - Combined w/ machine learning techniques

Other approaches than MC

Many applications to matrix models are investigated:

c.f. [Talks by Zechuan Zheng and Enrico Rinaldi]

- Bootstrap method
- Tensor network
  - Hamiltonian formalism (DMRG, PEPS, ⋯)
  - Lagrangian formalism (TRG, TNR, ⋯)
- Quantum computing
  - Anticipating Hamiltonian formalism
- Machine learning

⋯

They are playing a complementary role of MC!

Summary

Enjoy your research life
w/ Monte Carlo!