分子動力学法とその応用

Molecular Dynamics and Its Application

理学系研究科　物理学専攻　大久保　毅

Outline in the last week

- Standard Monte Carlo method
  - Importance sampling and Markov Chain Monte Carlo
    - Metropolis-Heisting sampling
    - Heat-bath sampling (Gibbs sampling)
- Application to classical spin systems
  - Local update, Global update
- Computational Science using Monte Carlo method
  - Important tips to obtain reliable results
  - Application and analysis in the case of critical phenomena

Computational science using MCMC methods

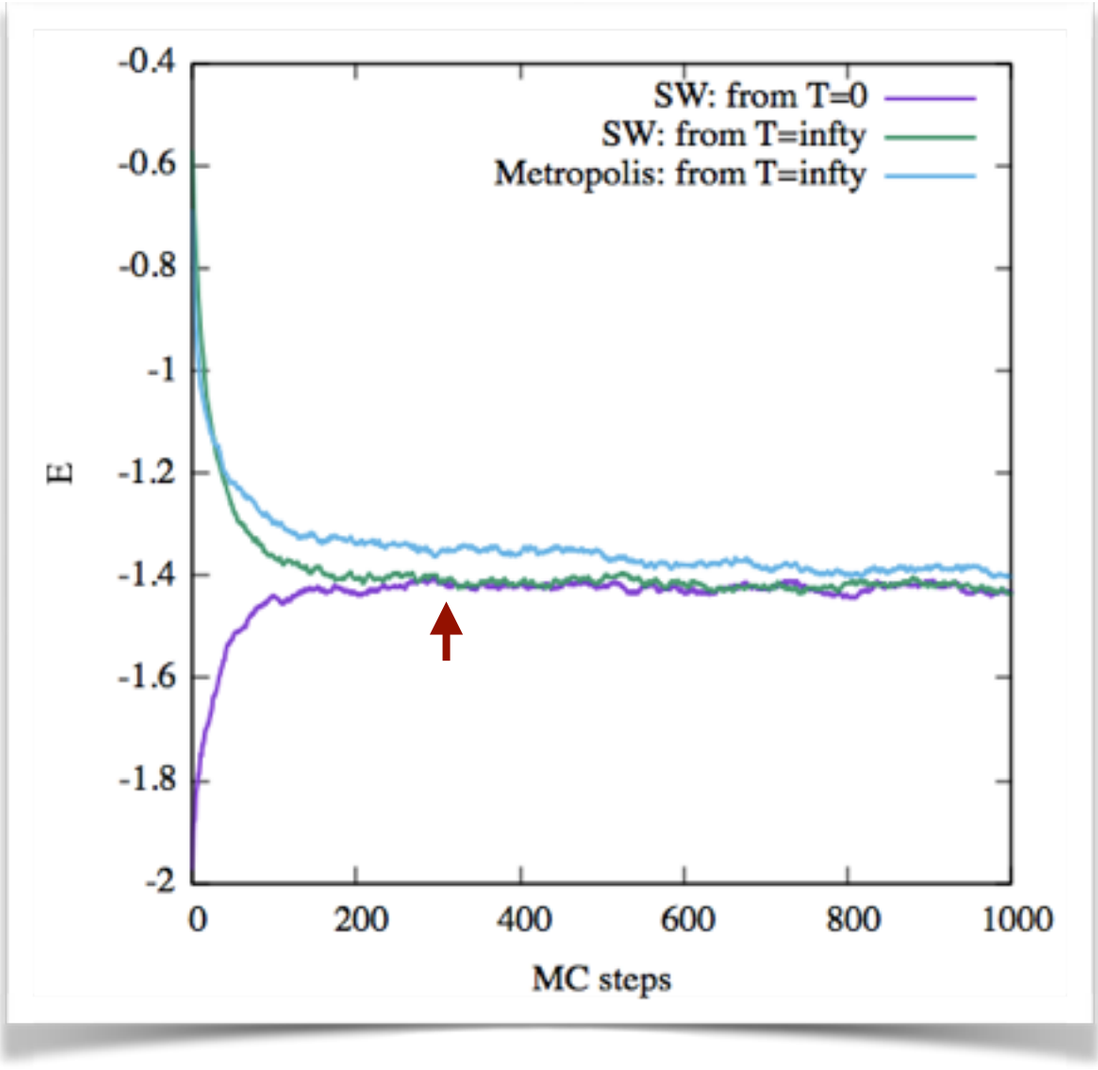
Important tips for real calculations 1

In each calculation, we have to check the convergence.

If the correlation time is very long, obtained data (expectation values) might be biased from the initial state Γ_0.

Usual procedure:

- Discard initial several MC steps
- Change MC steps and compare results
- Change Initial state
- ….



Important tips for real calculations 2

We need to estimate the statistical errors.

$$\bar{A} \equiv \frac{1}{T}\sum_{t=1}^{T} \hat{A}(\Gamma(t))$$

Standard error: $\epsilon^2 = \langle \bar{A}^2 \rangle - \langle \bar{A} \rangle^2$ $\qquad \epsilon \propto \sqrt{\dfrac{1+2\tau}{T}}$

Maximum likelihood estimation for standard error

Prepare "independent" M samples for $\bar{A}$: $\{\bar{A}_1, \bar{A}_2, \cdots, \bar{A}_M\}$

$$\sigma^2(M) = \frac{\frac{1}{M}\sum_i \bar{A}_i^2 - \left(\frac{1}{M}\sum_i \bar{A}_i\right)^2}{M-1}$$

$$\lim_{M\to\infty} \sigma^2(M) = \epsilon^2$$

Make "error bar" based on σ, and use it for data analysis.

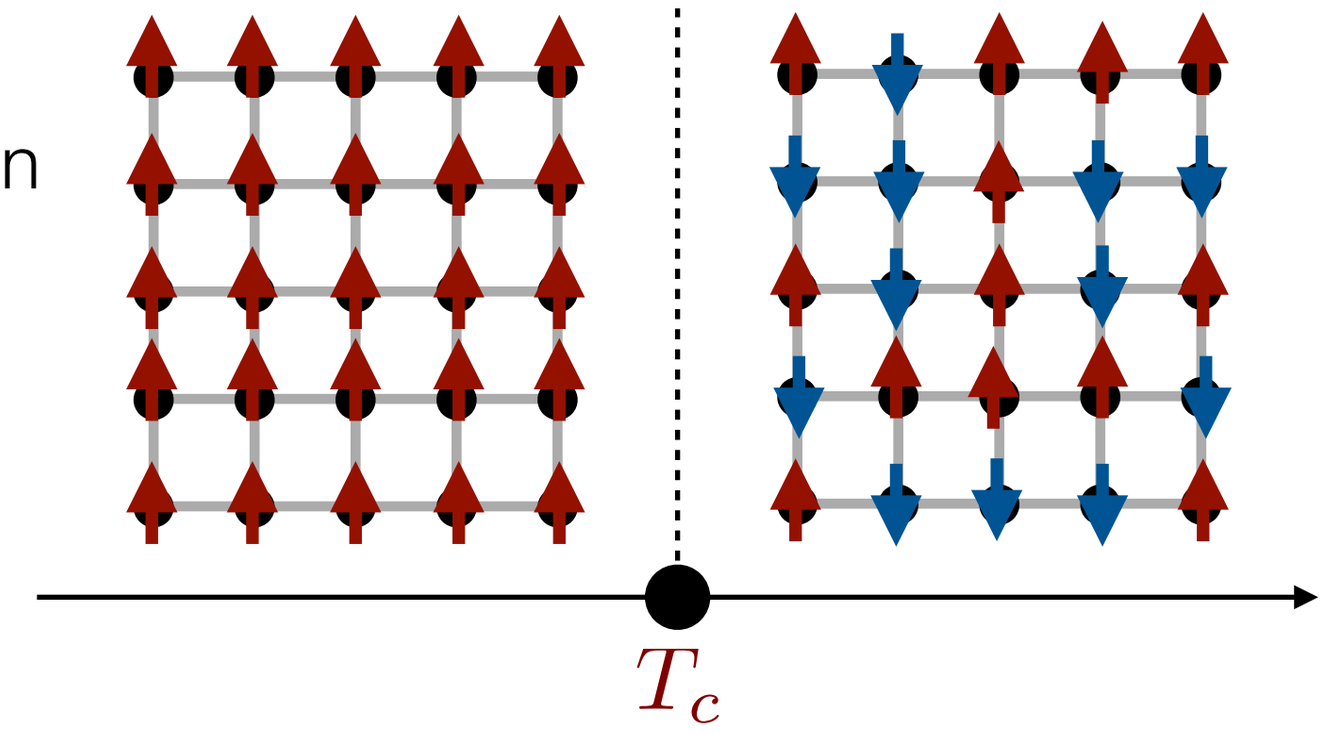
Example: Application for critical phenomena

- Square lattice Ising model

$$\mathcal{H} = -J \sum_{\langle i,j \rangle} S_i S_j$$

  - Continuous phase transition at $T=T_c$

$$T_c/J = \frac{2}{\ln(1+\sqrt{2})} = 2.26918531\ldots$$



  - $T>T_c$: Paramagnetic
  - $T<T_c$: Ferromagnetic

- Monte Carlo Simulation using ALPS (spinmc)
  - spinmc: Simulator for classical spin system by MCMC

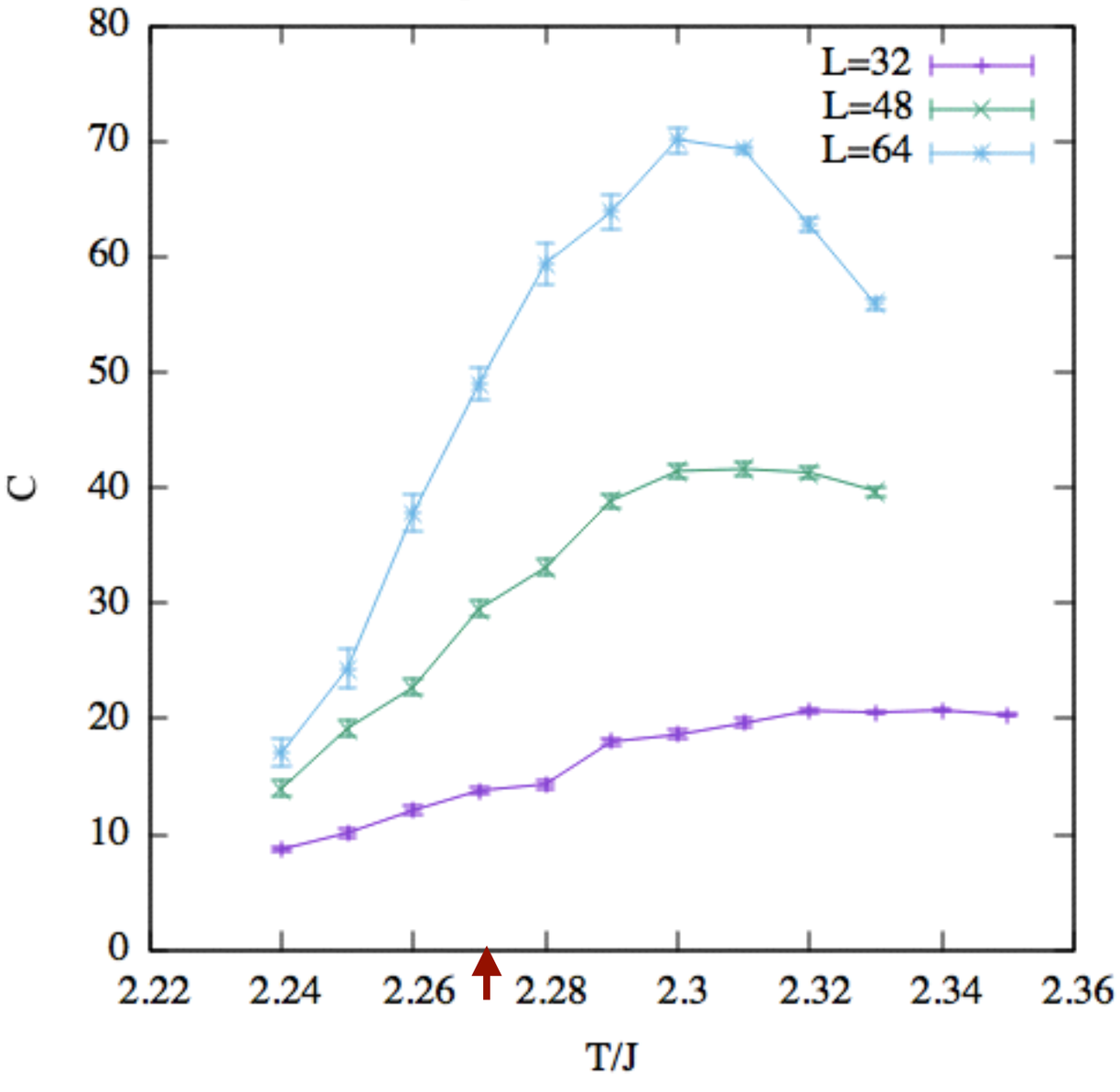
ALPS（Applications and Libraries for Physical Simulation)

- Set of libraries and applications for a variety of lattice models.
- Support for spin models, Hubbard model, Kondo lattice model, …
- A lot of solvers for models:
  - Classical/Quantum Monte Carlo, Exact Diagonalization, Density Matrix Renormalization Group (DMRG), Dynamical Mean Field Theory (DMFT), Time Evolving Block Decimation (TEBD), …
- We can select efficient solver for your problems.
- It can be applicable to the frontier research.

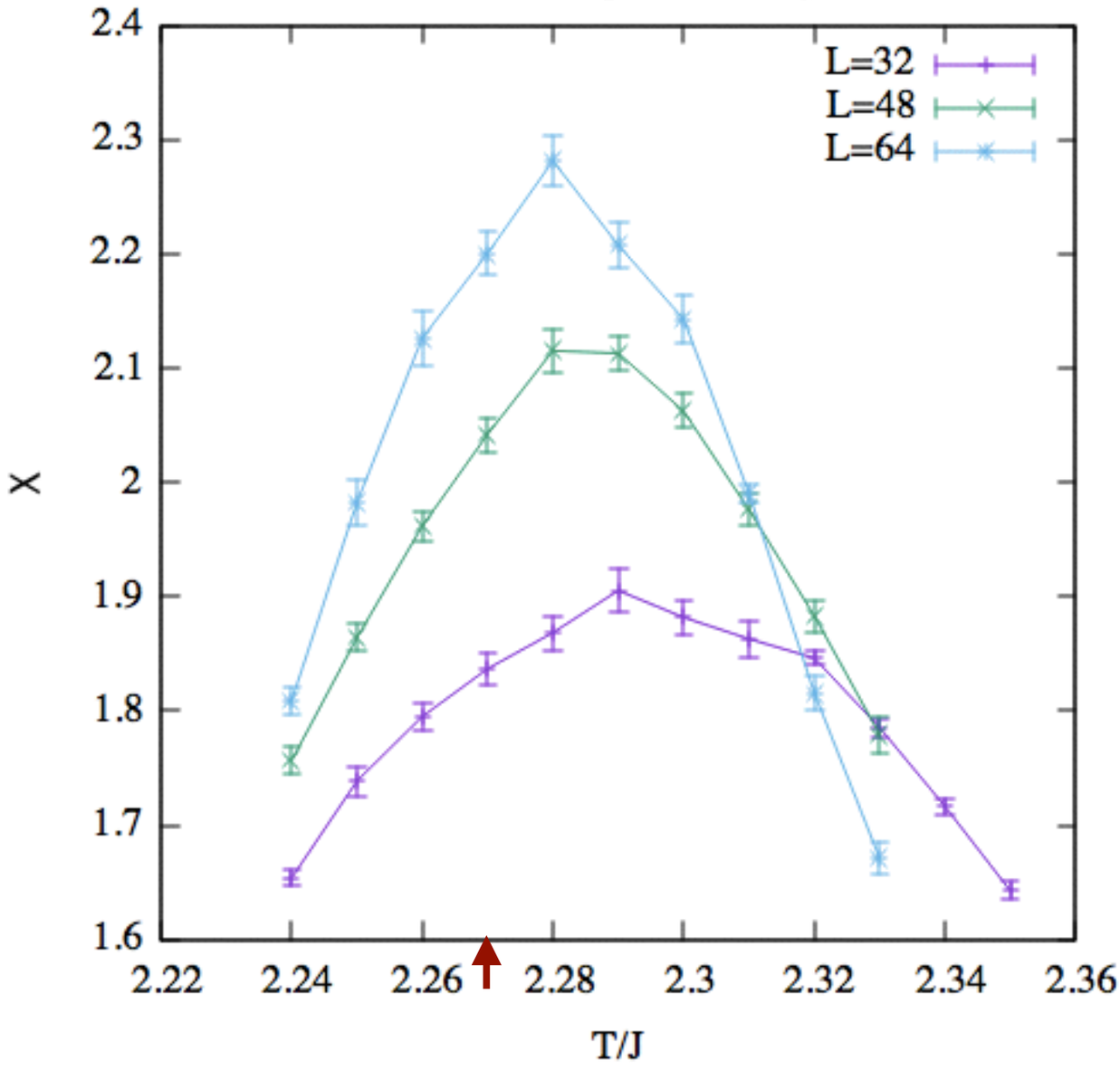
ALPS Wiki http://alps.comp-phys.org/mediawiki/index.php/Main_Page

Calculated data (ALPS tutorial 7b)

Specific heat



Susceptibility



$$T_c/J \simeq 2.269$$

Data analysis: Finite size scaling (outline)

Near the critical point (transition temperature):

The singular part of the free energy density satisfies finite size scaling

$$f_s(t, h, L) = L^{-d} f_s(tL^{y_t}, hL^{y_h})$$

$$t = T - T_c$$

y_t, y_h :scaling exponent $\Longleftrightarrow$ $y_t = 1/\nu, \quad y_h = \gamma/(2\nu)$

By taking derivatives, we see

$$M^2 = \frac{\partial^2 f}{\partial h^2} = L^{2y_h - d} g(tL^{y_t}, 0) \quad \text{(we set } h=0\text{)}$$

Physical quantity obeys common scaling function independent of L.

$\Rightarrow$ At the critical point, $M^2 \sim L^{-x_{M2}} \quad (x_{M2} \equiv d - 2y_h)$

x: scaling dimension

If x =0, it has no size dependence at the critical point!

Binder ratio

Binder ratio

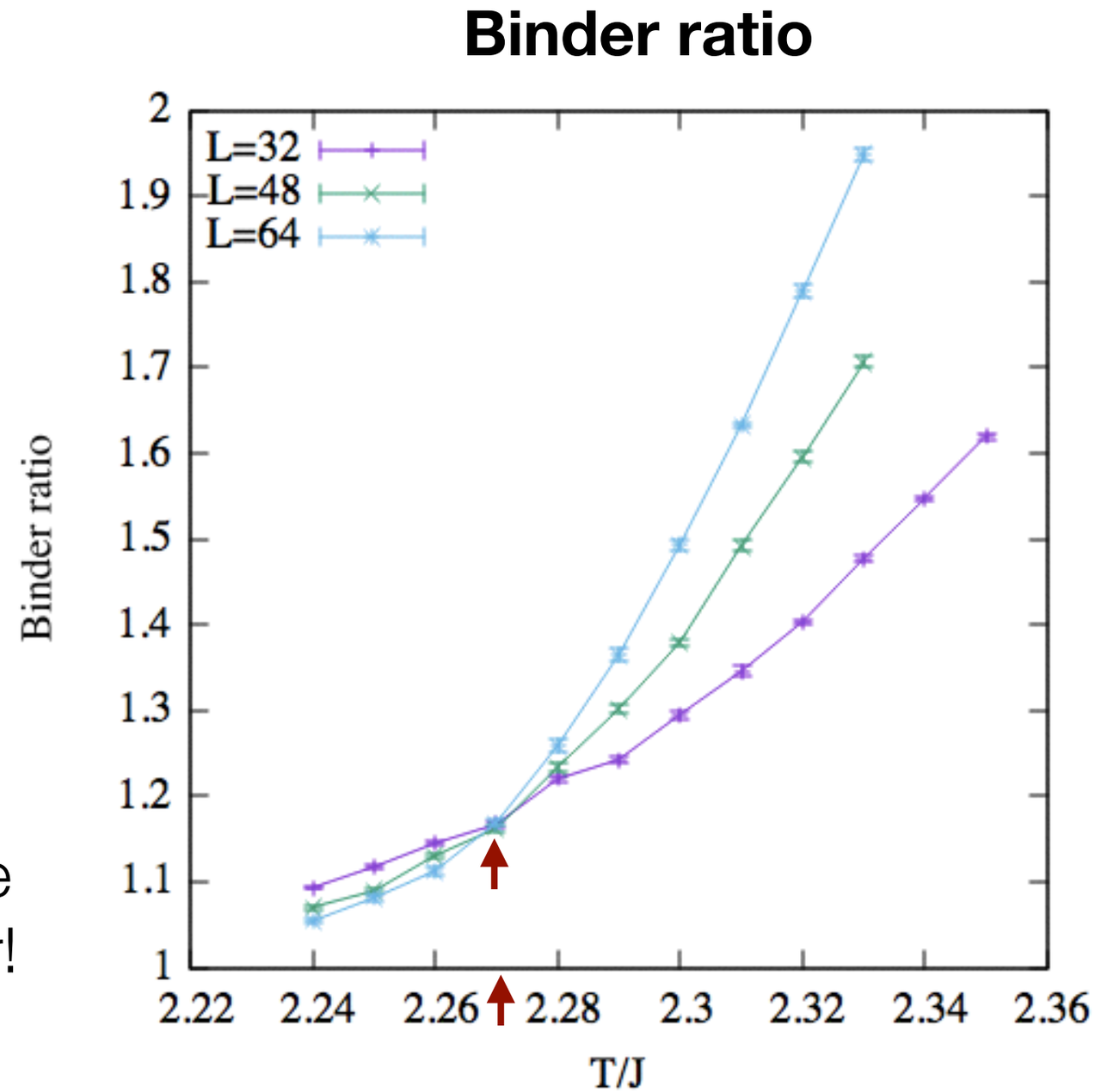
$$b = \frac{\langle M^4 \rangle}{\langle M^2 \rangle^2}$$

$$b = 3 \quad (T \to \infty)$$

$$b = 1 \quad (T \to 0)$$

The scaling dimension of b is exactly zero.

At Tc, the size dependence disappears in leading order!



$$T_c/J \simeq 2.269$$

Finite size scaling

Binder ratio

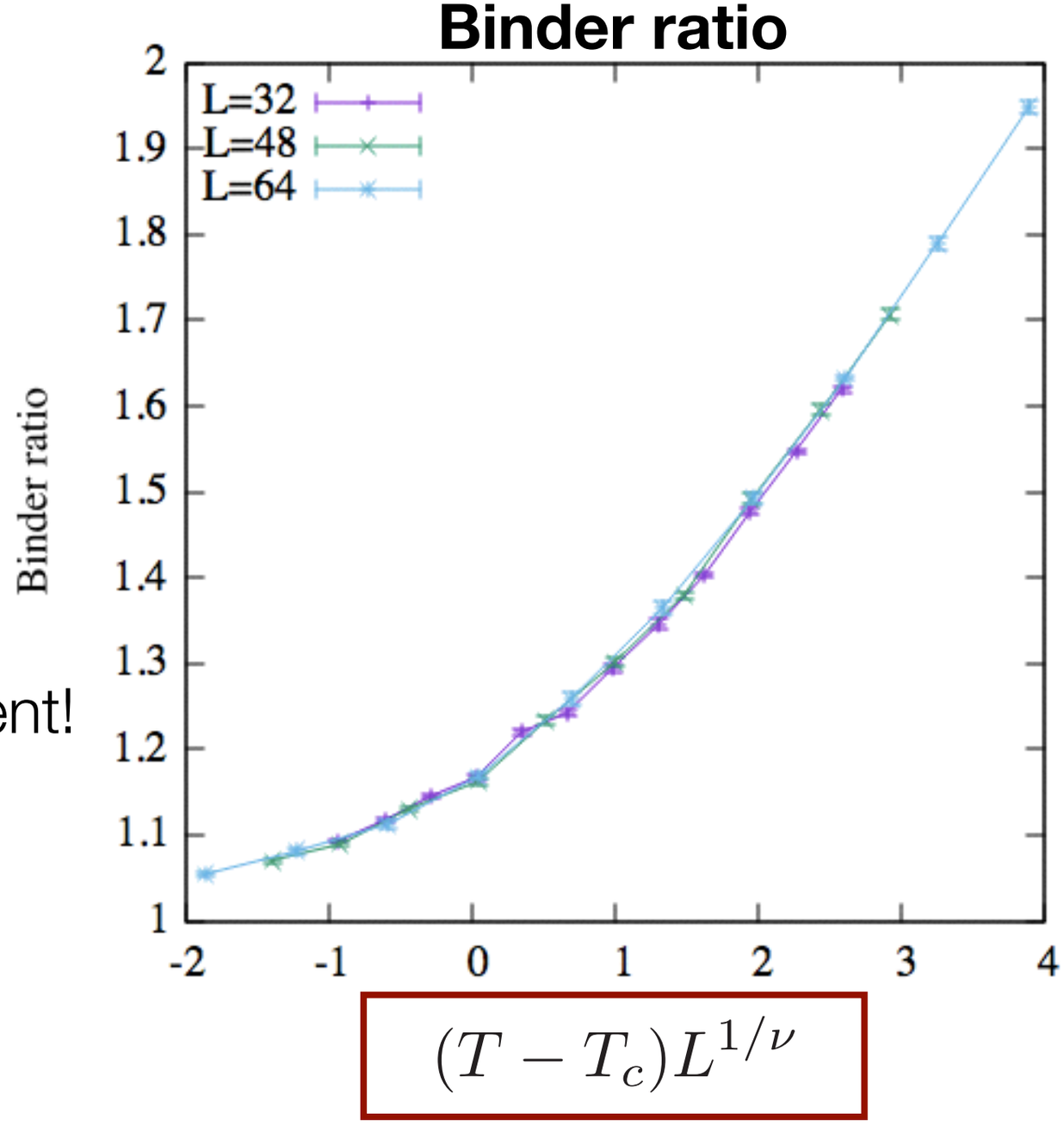
$$b = \frac{\langle M^4 \rangle}{\langle M^2 \rangle^2}$$

finite size scaling around Tc

$$b = f((T - T_c)L^{1/\nu})$$

We can determine critical exponent!

$$\nu = 1$$



Exercise: autocorrelation of MCMC (not a report)

See correlation time or autocorrelation function of
Ising model calculated by Monte Carlo simulation.

- Around Tc, how does the correlation time behave by varying the temperature?
- At Tc, how about the size (L) dependence?
- Does the correlation time depend on the algorithms?

To perform these exercise, you can use,

- Your own code
- ALPS (it is not straight forward to see the correlation time...)
- My sample codes for jupyter notebook (or python2.7).
  - In order to run the sample code you need
    - numpy, and numba
    - numba is used for speed up. In case you do not want to install numba, please use Ising_wo_jit.py .

usage: *jupyter notebook* → select Ising.ipynb

python Ising.py

Contents

- Basics of MD simulation
  - Newton equation, purpose of MD simulation
  - Examples of discrete equations and their stabilities
- NVE ensemble: standard MD simulation
  - Symplectic integral
- Control temperature and pressures
  - Velocity scaling and Nosé-Hoover method
  - Andersen method for pressure

Target: Newtonian mechanics

N-particle system:

$$m_i \frac{d^2 \boldsymbol{r}_i}{dt^2} = \boldsymbol{F}_i(\{\boldsymbol{r}_i\}) \qquad i = 1, 2, \cdots, N$$

e.g.

$$\boldsymbol{F}_i(\{\boldsymbol{r}_i\}) \equiv \sum_{j \neq i} F(|\boldsymbol{r}_i - \boldsymbol{r}_j|)\hat{\boldsymbol{r}}_{ij}$$

Unit vector

$$\hat{\boldsymbol{r}}_{ij} = \frac{\boldsymbol{r}_j - \boldsymbol{r}_i}{|\boldsymbol{r}_j - \boldsymbol{r}_i|}$$

Molecular Dynamics (MD) simulation:

Solve the newton equation numerically.

Standard flow of MD simulation

1. Determined the model
   - Potential energies, Constraints (*e.g.* polymers)
   - Periodic boundary, Open boundary, …
2. Prepare initial conditions $\{\boldsymbol{r}_i(t=0), \boldsymbol{v}_i(t=0)\}$
3. Calculate forces acting to all particles $\{\boldsymbol{F}_i(\{\boldsymbol{r}_i(t)\})\}$
4. Change positions and velocities by a discrete method $\{\boldsymbol{r}_i(t+\Delta t), \boldsymbol{v}_i(t+\Delta t)\}$
5. Calculate physical quantities and control them if we need $T(\{\boldsymbol{r}_i(t), \boldsymbol{v}_i(t)\})$, $P(\{\boldsymbol{r}_i(t), \boldsymbol{v}_i(t)\})$, …
   - Constant temperature, Constant pressure, …
6. Analyze trajectories

Periodic boundary condition

A particle interacts with all other particles in “image cells”.

Short-range interaction

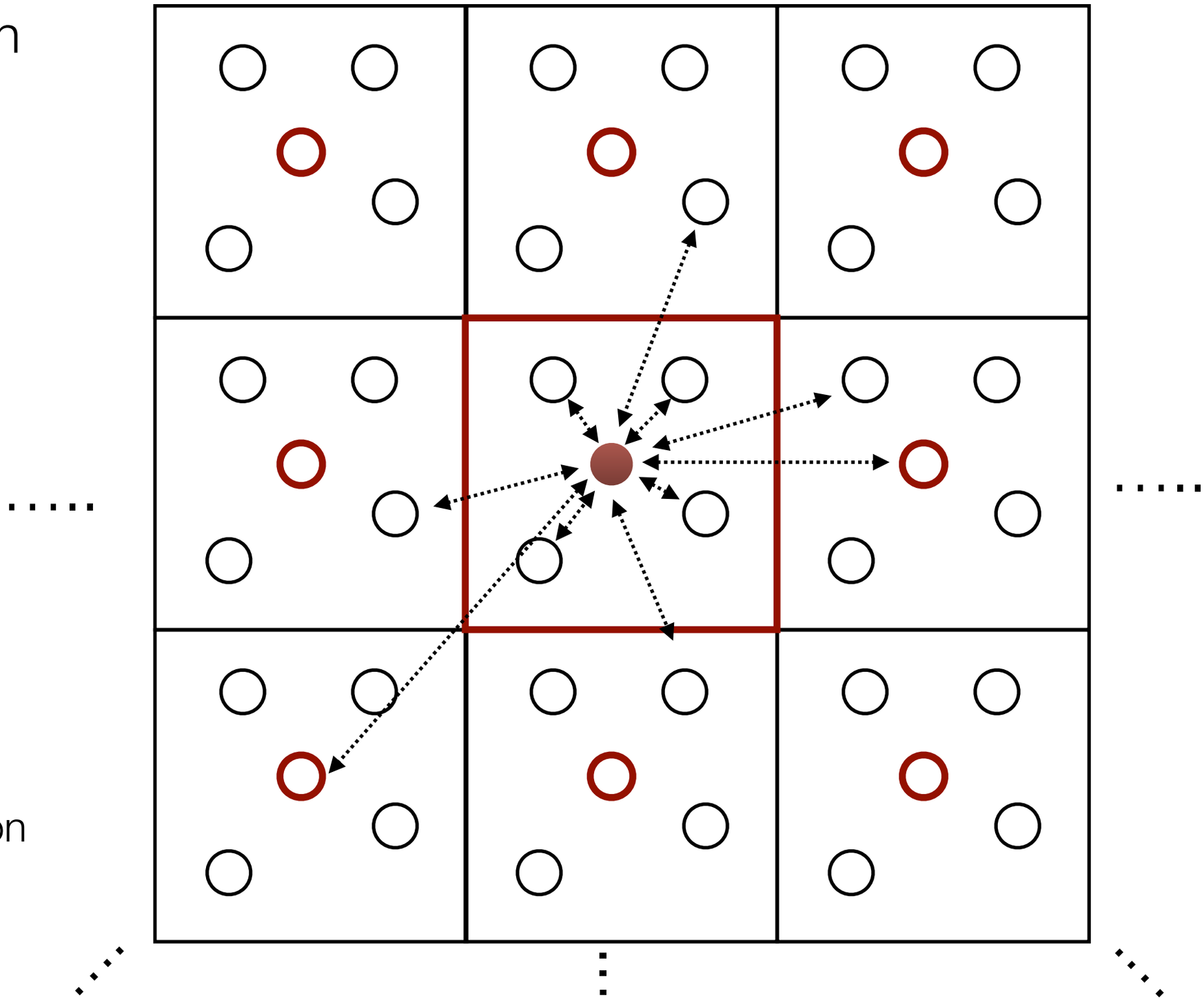
e.g. LJ potential

→ Introduce cut-off

Long-range interaction

e.g. Coulomb potential

→
- Ewald sum
- Multipole expansion



Purpose of MD simulation: Equilibrium properties

By MD simulation, we can calculate equilibrium properties.

Usual newton dynamics give us a NVE ensemble.

$$\langle \hat{O} \rangle_{NVE} = \lim_{\tau \to \infty} \frac{1}{\tau} \int_0^\tau dt \hat{O}(\Gamma(t))$$

By using temperature or pressure controls, we can also obtain other ensemble averages.

$$\langle \hat{O} \rangle_{NVT} = \lim_{\tau \to \infty} \frac{1}{\tau} \int_0^\tau dt \hat{O}(\Gamma_{NVT}(t))$$

$$\langle \hat{O} \rangle_{NPT} = \lim_{\tau \to \infty} \frac{1}{\tau} \int_0^\tau dt \hat{O}(\Gamma_{NPT}(t))$$

Modified dynamics!

Note: For large N limit, difference among ensembles is negligible.

We can use any ensembles for simulation.

Purpose of MD simulation: Equilibrium dynamics

By MD simulation, we can also calculate equilibrium dynamics

$$\langle \hat{A}\hat{B}(\Delta t)\rangle_{NVE} = \lim_{\tau\to\infty}\frac{1}{\tau}\int_0^\tau dt \hat{A}(\Gamma(t))\hat{B}(\Gamma(t+\Delta t))$$

$$\langle \hat{A}\hat{B}(\Delta t)\rangle_{NPT} = \lim_{\tau\to\infty}\frac{1}{\tau}\int_0^\tau dt \hat{A}(\Gamma_{NPT}(t))\hat{B}(\Gamma_{NPT}(t+\Delta t))$$

$$\langle \hat{A}\hat{B}(\Delta t)\rangle_{NVT} = \lim_{\tau\to\infty}\frac{1}{\tau}\int_0^\tau dt \hat{A}(\Gamma_{NVT}(t))\hat{B}(\Gamma_{NVT}(t+\Delta t))$$

Modified dynamics!

Note:

In this case, as far as I know, there is no proof that the modified dynamics for different ensembles give us same results in large N limit.

Probably, it is better to use NVE ensemble, after proper initialization using NPT of NVT dynamics.

Purpose of MD simulation: Non-Equilibrium

We can also calculate non-equilibrium properties using MD.

- Applying external fields
- Observing relaxation from initial conditions

e.g. linear response coefficients

- We can calculation the coefficient from equilibrium simulation by using Kubo formula.
- It can be obtained by MD simulation applying the external field.
  - Usually the non-equilibrium calculation gives us smaller error.

Numerical integration: Basics

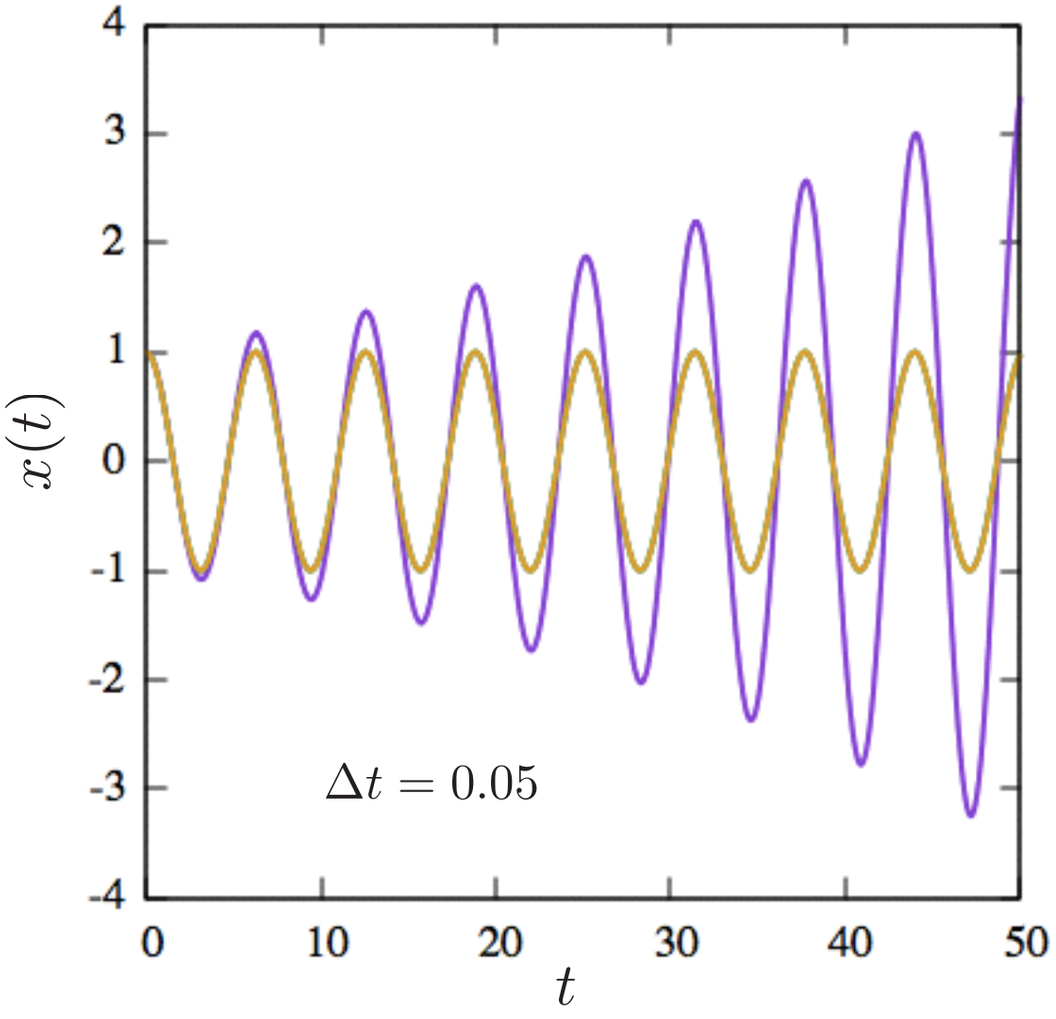
Example: 1d harmonic oscillator

$$\mathcal{H}(x) = \frac{1}{2}x^2 + \frac{1}{2}v^2 \quad \Rightarrow \quad \frac{dv}{dt} = -x, \quad \frac{dx}{dt} = v$$

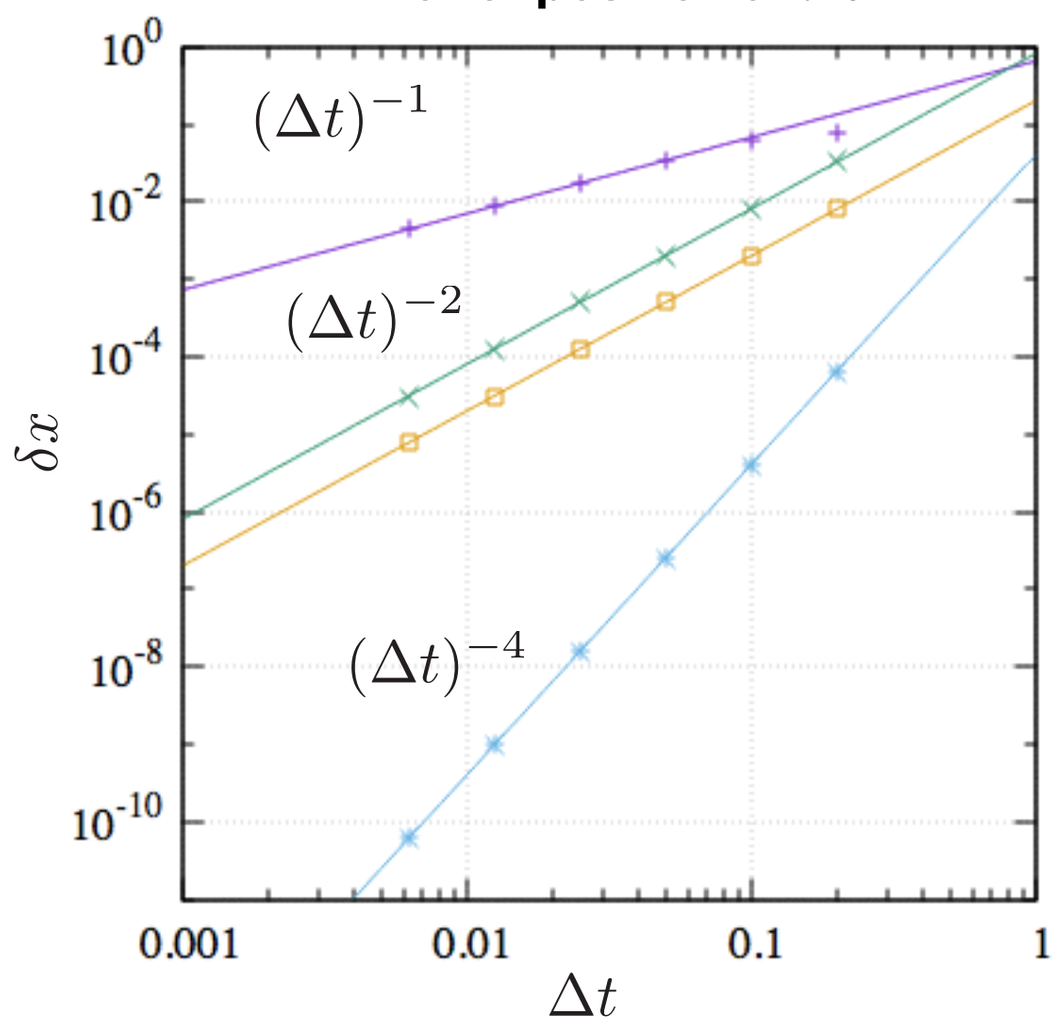
Several explicit methods

Euler
Improved Euler
4th Runge-Kutta
Verlet

Position



Error of position at t=5



Numerical integration: accuracy and cost

Important points for molecular dynamics simulation

- Error
- Stability
- Number of force calculations

Main part of cpu cost

e.g.

$$\boldsymbol{F}_i(\{\boldsymbol{r}_i\}) \equiv \sum_{j \neq i} F(|\boldsymbol{r}_i - \boldsymbol{r}_j|)\hat{\boldsymbol{r}}_{ij}$$

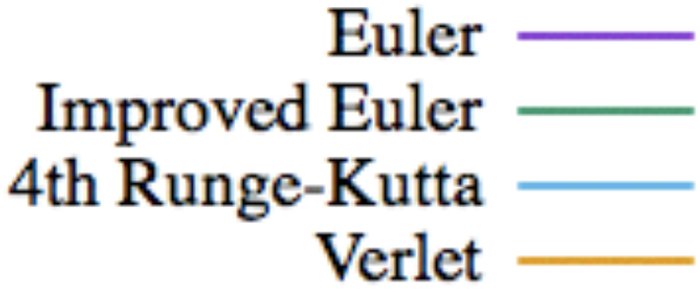
	order of error	#of force calculation	initial condition
Euler	Δt	1	$r(0)$, $v(0)$
Improved Euler	$(\Delta t)^2$	2	$r(0)$, $v(0)$
4th Runge-Kutta	$(\Delta t)^4$	4	$r(0)$, $v(0)$
Verlet	$(\Delta t)^2$	1	$r(0)$, $r(\Delta t)$ (velocity Verlet:$r(0)$, $v(0)$)
Predictor-Corrector	$(\Delta t)^5$	2 (or 1)	$r(0)$, $r'(0)$, $r''(0)$, $r'''(0)$, $r''''(0)$, $r'''''(0)$

Numerical integration: instability (energy drift)

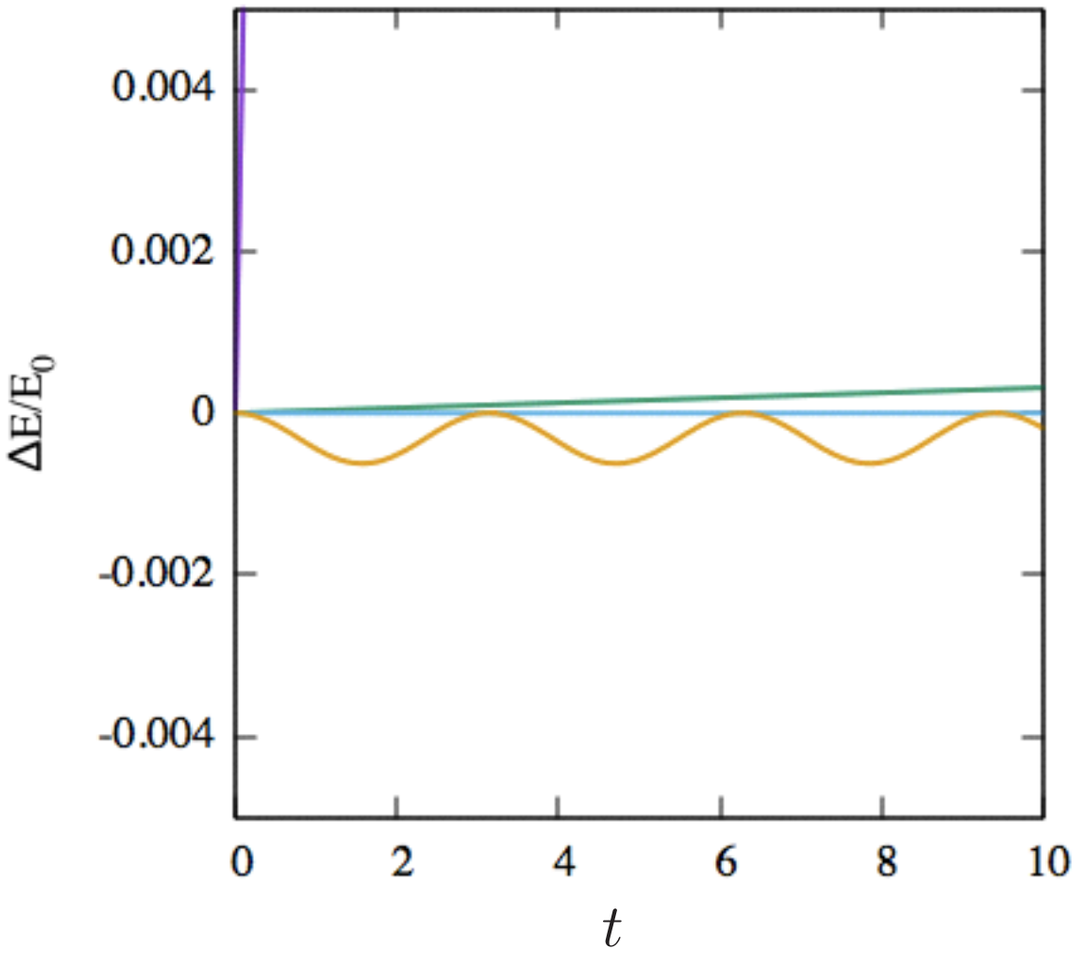
Example: 1d harmonic oscillator

Usual methods shows a drift of energy!
(Predictor-Corrector also shows large energy drift)

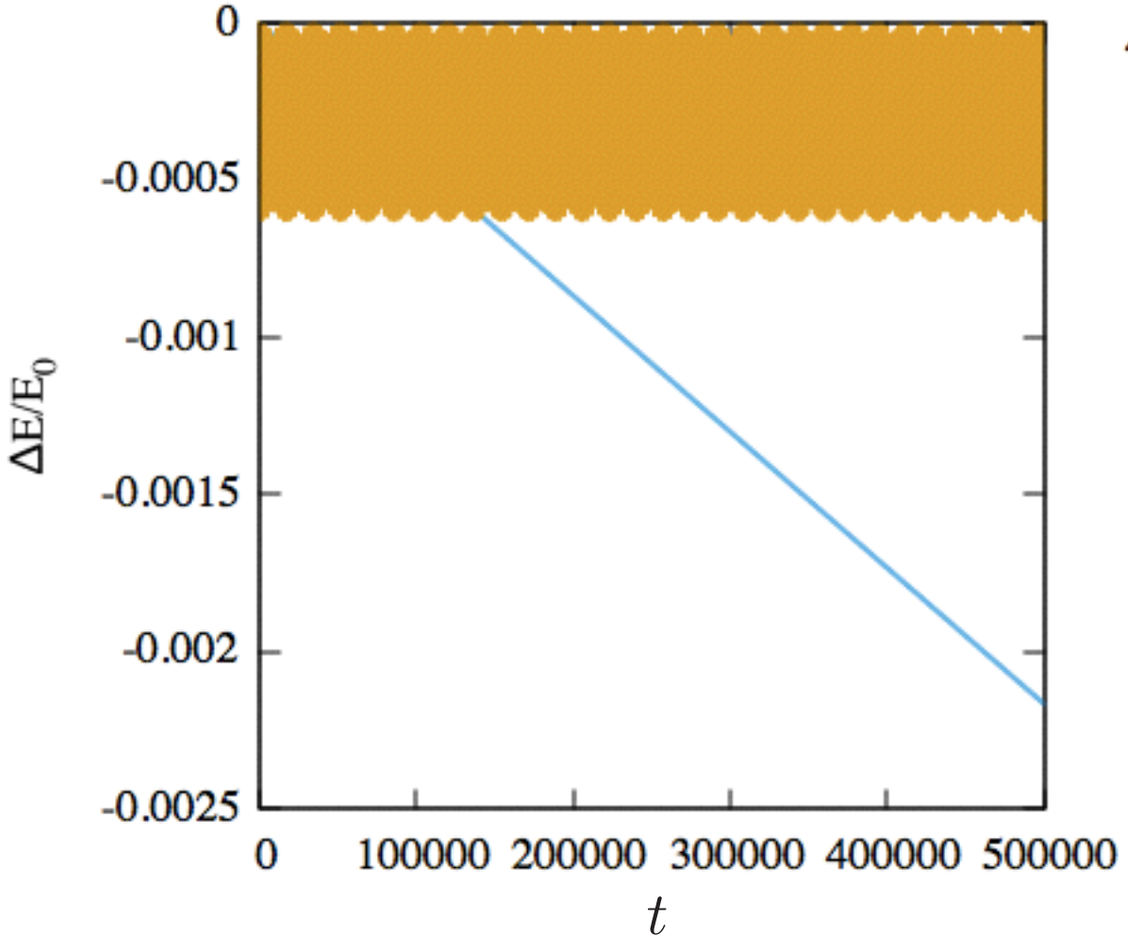
Verlet shows a very small energy drift



Energy error (short time)



Energy error (long time)



Better methods for molecular dynamics simulation

Verlet method:

- Error
  - $(\Delta t)^2 \rightarrow$not bad
- Stability
  - It seems to so stable!
- Number of force calculations
  - Only 1 force calculation for 1 step

Verlet method:

$$\boldsymbol{r}_i(t+\Delta t) = 2\boldsymbol{r}_i(t) - \boldsymbol{r}_i(t-\Delta t) + \frac{(\Delta t)^2}{m_i}\boldsymbol{F}_i(\{\boldsymbol{r}_i(t)\})$$

$$\boldsymbol{v}_i(t) = \frac{\boldsymbol{r}_i(t+\Delta t) - \boldsymbol{r}_i(t-\Delta t)}{2\Delta t}$$

$$m_i \frac{d^2\boldsymbol{r}_i}{dt^2} = \boldsymbol{F}_i(\{\boldsymbol{r}_i\})$$

Velocity Verlet method:

$$\boldsymbol{r}_i(t+\Delta t) = \boldsymbol{r}_i(t) + \Delta t\boldsymbol{v}_i(t) + \frac{(\Delta t)^2}{2m_i}\boldsymbol{F}_i(\{\boldsymbol{r}_i(t)\})$$

$$\boldsymbol{v}_i(t+\Delta t) = \boldsymbol{v}_i(t) + \Delta t\frac{\boldsymbol{F}_i(\{\boldsymbol{r}_i(t)\}) + \boldsymbol{F}_i(\{\boldsymbol{r}_i(t+\Delta t)\})}{2m_i}$$

Leap-frog method:

$$\boldsymbol{r}_i(t+\Delta t) = \boldsymbol{r}_i(t) + \boldsymbol{v}_i\left(t+\frac{\Delta t}{2}\right)\Delta t$$

$$\boldsymbol{v}_i\left(t+\frac{\Delta t}{2}\right) = \boldsymbol{v}_i\left(t-\frac{\Delta t}{2}\right) + \Delta t\frac{\boldsymbol{F}_i(\{\boldsymbol{r}_i(t)\})}{m_i}$$

These methods are basically equivalent.
They are based on the second-order symplectic integration scheme.

NVE ensemble: symplectic integrator

Hamilton mechanics

Hamilton mechanics

$$\mathcal{H}(\{q_i\},\{p_i\}) \Rightarrow \begin{aligned} \frac{dq_i}{dt} &= \frac{\partial \mathcal{H}}{\partial p_i} \\ \frac{dp_i}{dt} &= -\frac{\partial \mathcal{H}}{\partial q_i} \end{aligned}$$

Any quantities: $A(t) = A[\{q_i(t)\},\{p_i(t)\}]$

$$\frac{dA}{dt} = \{A, \mathcal{H}\}$$

Poisson bracket:

$$\{u, v\} = \sum_i \left(\frac{\partial u}{\partial q_i}\frac{\partial v}{\partial p_i} - \frac{\partial u}{\partial p_i}\frac{\partial u}{\partial p_i} \right)$$

Liouville operator: $i\mathcal{L} = \{\quad, \mathcal{H}\}$

$$\frac{dA}{dt} = i\mathcal{L}A \Rightarrow A(t) = e^{it\mathcal{L}}A(0)$$

Unitary operator

Liouville's theorem

Distribution function: $\rho(\{q_i\}, \{p_i\}; t)$

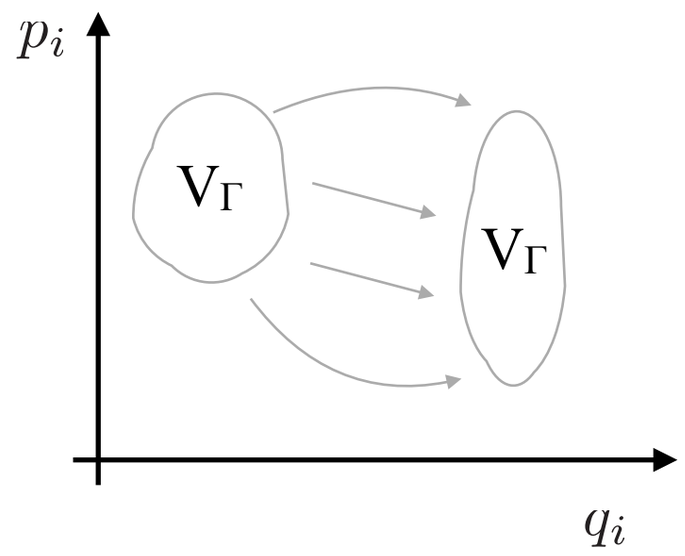
Liouville equation

$$\frac{\partial \rho}{\partial t} = \{\mathcal{H}, \rho\} = -i\mathcal{L}\rho$$

$$\frac{d\rho}{dt} = \frac{\partial \rho}{\partial t} + \sum_i \left(\frac{\partial \rho}{\partial q_i}\frac{\partial \mathcal{H}}{\partial p_i} - \frac{\partial \rho}{\partial p_i}\frac{\partial \mathcal{H}}{\partial q_i} \right) = \frac{\partial \rho}{\partial t} + i\mathcal{L}\rho = 0$$

Liouville's theorem:

Along Hamiltonian mechanics,
the volume in phase space is conserved.



Canonical transformation and symplectic condition

$$\boldsymbol{\Gamma} = (\{q_i\}, \{p_i\}) = (\boldsymbol{q}, \boldsymbol{p})$$

(*2n*-dim. vector)

$$\frac{dq_i}{dt} = \frac{\partial \mathcal{H}}{\partial p_i} \quad \frac{dp_i}{dt} = -\frac{\partial \mathcal{H}}{\partial q_i}$$

$$\rightarrow \quad \frac{d\boldsymbol{\Gamma}}{dt} = J\frac{\partial \mathcal{H}}{\partial \boldsymbol{\Gamma}}$$

$$J = \begin{pmatrix} 0 & \mathbf{1} \\ -\mathbf{1} & 0 \end{pmatrix} \quad : 2n \times 2n \text{ matrix}$$

($n=3N$ in 3d system)

Canonical transformation: $\boldsymbol{\Gamma} \rightarrow \boldsymbol{\Gamma}' = (\boldsymbol{q}'(\boldsymbol{q},\boldsymbol{p}), \boldsymbol{p}'(\boldsymbol{q},\boldsymbol{p}))$

$$\boldsymbol{\Gamma}' = S\boldsymbol{\Gamma} \quad , \quad S_{ij} = \frac{\partial \Gamma'_i}{\partial \Gamma_j} \quad : 2n \times 2n \text{ matrix}$$

$$\rightarrow \quad \frac{d\boldsymbol{\Gamma}'}{dt} = S\frac{d\boldsymbol{\Gamma}}{dt} = SJ\frac{\partial \mathcal{H}}{\partial \boldsymbol{\Gamma}}$$

$$= SJS^T \frac{\partial \mathcal{H}}{\partial \boldsymbol{\Gamma}'} = J\frac{\partial \mathcal{H}}{\partial \boldsymbol{\Gamma}'}$$

$$\frac{\partial}{\partial \Gamma_i} = \sum_j \frac{\partial \Gamma'_j}{\partial \Gamma_i}\frac{\partial}{\partial \Gamma'_j} = \sum_j (S^T)_{ij}\frac{\partial}{\partial \Gamma'_j}$$

Definition of the canonical transformation

Canonical transformation satisfy the symplectic condition: $SJS^T = J$

Infinitesimal time evolution

$\Gamma(\boldsymbol{q}(t), \boldsymbol{p}(t)) \to \Gamma'(\boldsymbol{q}(t+\Delta t), \boldsymbol{p}(t+\Delta t))$

$$\boldsymbol{q}' = \boldsymbol{q}(t+\Delta t) = \boldsymbol{q}(t) + \Delta t \frac{\partial \mathcal{H}}{\partial \boldsymbol{p}(t)}$$

$$\boldsymbol{p}' = \boldsymbol{p}(t+\Delta t) = \boldsymbol{p}(t) - \Delta t \frac{\partial \mathcal{H}}{\partial \boldsymbol{q}(t)}$$

This is a canonical transformation, when Δt is infinitesimal.

→ Exact Hamiltonian dynamics satisfies the symplectic condition.
(The symplectic condition contains Liouville's theorem.)

⇔ For finite Δt (Euler method), it breaks the symplectic condition, and the volume of phase space is **not** conserved.
(Main reason for the energy drift.)

If we can construct discrete approximations satisfying the symplectic condition, we obtain more stable methods!

Symplectic integrator

Symplectic integrator:

Discrete approximation of Hamilton dynamics
satisfying the symplectic condition.

$$e^{it\mathcal{L}} \simeq \ldots$$

→ Because the volume of phase space is conserved,
the energy does not drift along this dynamics.
(if there is no numerical error.)

If the Hamiltonian can be decomposed, for example

$$\mathcal{H} = K(\{p_i\}) + V(\{q_i\})$$
$$i\mathcal{L} = i\mathcal{L}_K + i\mathcal{L}_V$$

→ There is a systematic derivation of
symplectic integrators.

Decomposition of exponential operator

Symplectic integrator:

$$e^{it\mathcal{L}} = \prod_{k=1}^{n} \left[e^{ia_k t\mathcal{L}_K} e^{ib_k t\mathcal{L}_V} \right] + O(t^{n+1})$$

$$\sum_{k=1}^{n} a_k = \sum_{k=1}^{n} b_k = 1$$

$$\mathcal{H} = K(\{p_i\}) + V(\{q_i\})$$

$$i\mathcal{L} = i\mathcal{L}_K + i\mathcal{L}_V$$

Note: $e^{ia_k t\mathcal{L}_K}, e^{ia_k t\mathcal{L}_V}$ satisfy the symplectic condition

n=1 $e^{it\mathcal{L}} \simeq e^{it\mathcal{L}_K} e^{it\mathcal{L}_V}$

$$q(t+\Delta t) = q(t) + p(t)\delta t$$
$$p(t+\Delta t) = p(t) + F(q(t+\Delta))\delta t$$

Euler like equation
(but this is more stable!)

n=2 $e^{it\mathcal{L}} \simeq e^{i\frac{t}{2}\mathcal{L}_K} e^{it\mathcal{L}_V} e^{i\frac{t}{2}\mathcal{L}_K}$

$$p(t+\frac{\Delta t}{2}) = p(t) + F(q(t))\frac{\Delta t}{2}$$
$$q(t+\Delta t) = q(t) + p(t+\frac{\Delta t}{2})\Delta t$$
$$p(t+\Delta t) = p(t+\frac{\Delta t}{2}) + F(q(t+\Delta))\frac{\Delta t}{2}$$

Exactly equal to
(Velocity) Verlet method

Control temperature and pressure

Temperature control: velocity scaling

The most simplest method for temperature setting: **Velocity Scaling**

(L. V. Woodcock, Chem. Phys. Lett. **10** ,257 (1971).)

Total kinetic energy: $K = \sum_i \frac{\boldsymbol{p}_i^2}{2m_i}$

Under the canonical (NVT) ensemble

$$\langle K \rangle = \frac{3}{2} N k_B T$$ (Equipartition of energy in 3d)

➡ Define effective temperature of a snapshot:

$$T_{\text{eff}} \equiv \frac{2K}{3Nk_B}$$

Rescale velocities every time step as

$$\boldsymbol{p}_i' = \boldsymbol{p}_i \sqrt{\frac{T}{T_{\text{eff}}}} \quad \Rightarrow \quad K' = \frac{3}{2} N k_B T$$

Results of the velocity scaling

Total kinetic energy is artificially fixed to

$$K = \frac{3}{2} N k_B T$$

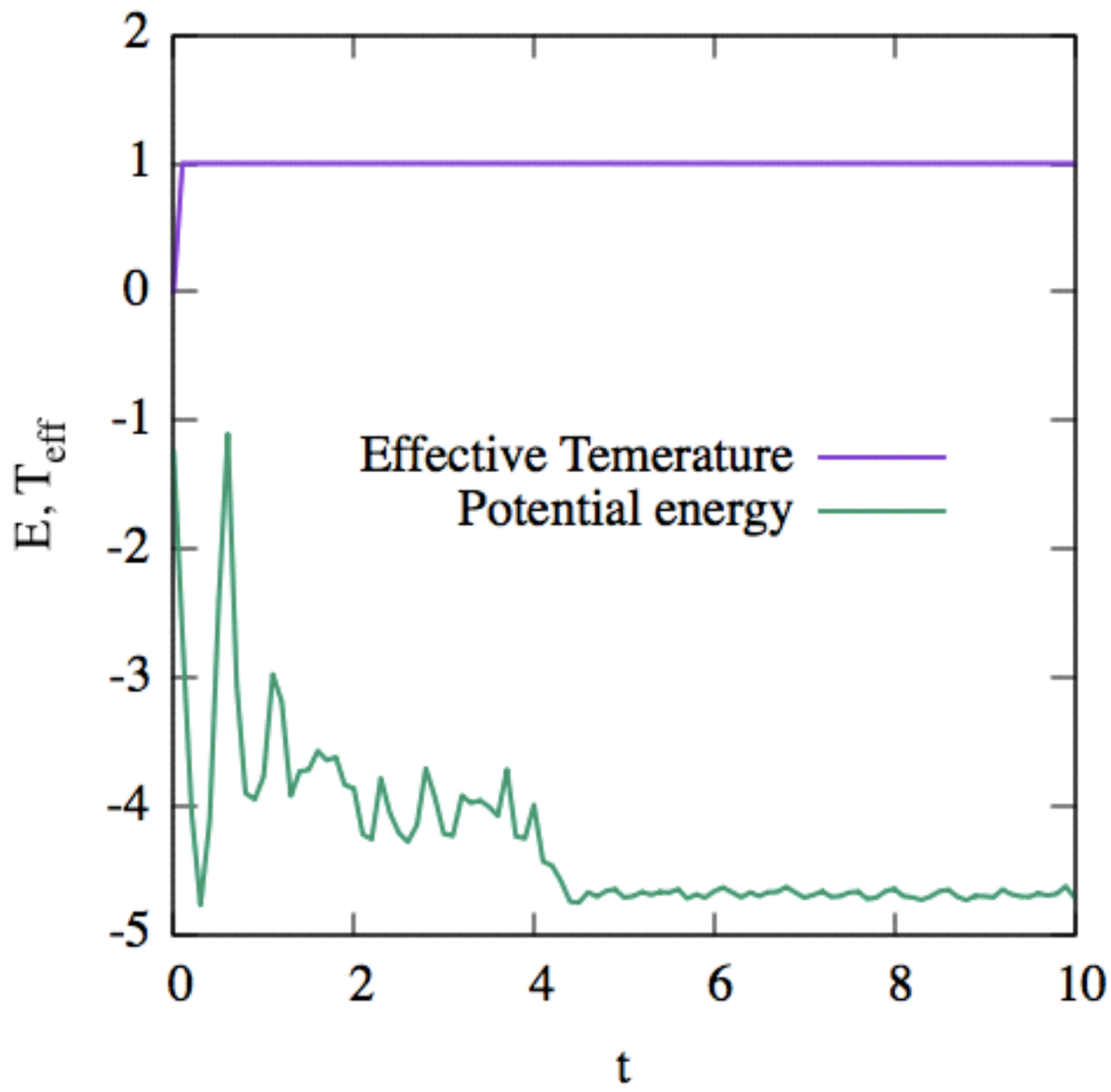
Under velocity scaling dynamics,
the trajectories do not
necessarily obey
the canonical ensemble.

However,

- We can use it for an initialization for NVE ensemble.
- Position fluctuation could be effectively similar to that of NVT ensemble.

MD of LJ system with velocity scaling

($\Delta t = 0.01$, N=1000, ρ=0.8, T=1)



Temperature control: Langevin dynamics

Langevin dynamics

$$\frac{d\boldsymbol{p}_i}{dt} = \boldsymbol{F}_i(\{\boldsymbol{q}_i\}) \underline{- \gamma \boldsymbol{p}_i} + \underline{\boldsymbol{R}_i}$$

Dissipation

Random force

(Gaussian white noise)

$$\langle \boldsymbol{R}_i(t) \rangle = \boldsymbol{0}$$

$$\langle \boldsymbol{R}_i(0) \boldsymbol{R}_j(t) \rangle = 2D_i \delta_{ij} \delta(t)$$

Long-time average of Langevin dynamics becomes
the canonical ensemble with temperature T,
if random forces satisfy the relation

$$D_i = \frac{k_B T}{m_i \gamma}$$

Einstein relation

Fluctuation-dissipation theorem

Temperature control: Nosé thermostad

Nose thermostad S. Nosé, Mol. Phys., **52**, 255 (1984). S. Nosé, J. Chem. Phys., **81**, 511 (1984).

Extended Hamiltonian System with a "heat bath"

$$\mathcal{H}_N = \sum_i \frac{(\boldsymbol{p}_i')^2}{2m_i s^2} + V(\{\boldsymbol{q}_i\}) + \frac{P_s^2}{2Q} + gk_BT\ln s$$

Original Hamiltonian with scaled momentum

$$\mathcal{H}\left(\left\{\frac{\boldsymbol{p}_i'}{s}\right\}, \{\boldsymbol{q}_i\}\right)$$

Heat-bath

s: scale factor for time

$$t' = st$$
$$\boldsymbol{p}_i' = s\boldsymbol{p}_i$$

Canonical equation

$$\frac{d\boldsymbol{p}_i'}{dt'} = -\frac{\partial V}{\partial \boldsymbol{q}_i} = \boldsymbol{F}_i(\{\boldsymbol{q}_i\}) \qquad \frac{dP_s}{dt'} = \frac{1}{s}\left(\sum_i \frac{(\boldsymbol{p}_i')^2}{m_i s^2} - gk_BT\right)$$

$$\frac{d\boldsymbol{q}_i}{dt'} = \frac{\boldsymbol{p}_i'}{m_i s^2} \qquad \frac{ds}{dt'} = \frac{P_s}{Q}$$

Temperature control: Nosé-Hoover method

Nosé-Hoover dynamics

Real-time dynamics with $\zeta = \dfrac{ds}{dt'}$ (W. G. Hoover, Phys. Rev. A, **31**, 1695 (1985).)

$$\boldsymbol{p}_i = \frac{\boldsymbol{p}_i'}{s} \qquad t = \frac{t'}{s}$$

$$\frac{d\boldsymbol{q}_i}{dt} = \frac{\boldsymbol{p}_i}{m_i}$$

$$\frac{d\boldsymbol{p}_i}{dt} = \boldsymbol{F}_i(\{\boldsymbol{q}_i\}) - \zeta \boldsymbol{p}_i$$

$$\frac{d\zeta}{dt} = \frac{gk_B}{Q}\left[\frac{1}{gk_B}\sum_i \frac{\boldsymbol{p}_i^2}{2m_i} - T\right] = \frac{1}{\tau^2}\left[T_{\mathrm{eff}} - T\right]$$

$g = 3N$ (# of DOF)

$\tau^2 = \dfrac{Q}{gk_B}$ (characteristic time scale)

New degree of freedom represents viscosity: ζ

➡ It changes the sign depending on the difference between the effective temperature and the aimed temperature.

(It also accelerates the velocity if $T_{\mathrm{eff}} < T$)

* This dynamics is not symplectic. ➡ Symplectic version: Nosé-Poincare method

S. D. Bond, *et.al.* J. Comp. Phys. **151**, 114 (1999)

Nosé-Hoover dynamics becomes NVT ensemble

Short proof:

(Based on Hisashi Okumura's review paper,
"分子動力学シミュレーションにおける温度・圧力制御")

$$\mathcal{H}_N = \mathcal{H}\left(\left\{\frac{\boldsymbol{p}'_i}{s}\right\}, \{\boldsymbol{q}_i\}\right) + \frac{P_s^2}{2Q} + gk_BT\ln s$$

MD on (q, p',t') dynamics yields NVE ensemble of H_N

$$\lim_{\tau'\to\infty}\frac{1}{\tau'}\int_0^{\tau'} dt' O(\{\frac{\boldsymbol{p}'_i}{s}\},\{\boldsymbol{q}_i\}) = \frac{\int d\boldsymbol{p}'_i d\boldsymbol{q}_i dP_s ds O(\{\frac{\boldsymbol{p}'_i}{s}\},\{\boldsymbol{q}_i\})\delta(E-\mathcal{H}_N)}{\int d\boldsymbol{p}'_i d\boldsymbol{q}_i dP_s ds \delta(E-\mathcal{H}_N)}$$

$$= \frac{\int d\boldsymbol{p}_i d\boldsymbol{q}_i dP_s ds s^{3N} O(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\})\delta(E-\mathcal{H}-\frac{P^2}{2Q}-gk_BT\ln s)}{\int d\boldsymbol{p}_i d\boldsymbol{q}_i dP_s ds s^{3N}\delta(E-\mathcal{H}-\frac{P^2}{2Q}-gk_BT\ln s)}$$

from $$\int ds s^{3N}\delta(E-\mathcal{H}-\frac{P^2}{2Q}-gk_BT\ln s) = \frac{1}{gk_BT}e^{-\frac{3N+1}{gk_BT}(\mathcal{H}+\frac{P^2}{2Q}-E)}$$

$$= \frac{\int d\boldsymbol{p}_i d\boldsymbol{q}_i O(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\}) e^{-\frac{3N+1}{gk_BT}\mathcal{H}}}{\int d\boldsymbol{p}_i d\boldsymbol{q}_i e^{-\frac{3N+1}{gk_BT}\mathcal{H}}}$$

$$\delta(f(x)) = \frac{\delta(x-x_0)}{|f'(x_0)|} \qquad (f(x_0)=0)$$

Canonical ensemble if $g = 3N+1$

Nosé-Hoover dynamics becomes NVT ensemble 2

Time average on t

$$\lim_{\tau\to\infty}\frac{1}{\tau}\int_0^{\tau} dtO(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\}) = \lim_{\tau\to\infty}\frac{\tau'}{\tau}\frac{1}{\tau'}\int_0^{\tau'} dt'\frac{O(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\})}{s}$$

from $\boxed{\tau = \int_0^{\tau'}\frac{1}{s}dt'}$

$$= \frac{\lim_{\tau'\to\infty}\frac{1}{\tau'}\int_0^{\tau'} dt'\frac{O(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\})}{s}}{\lim_{\tau'\to\infty}\frac{1}{\tau'}\int_0^{\tau'} dt'\frac{1}{s}}$$

$$= \frac{\int d\boldsymbol{p}_i d\boldsymbol{q}_i O(\{\boldsymbol{p}_i\},\{\boldsymbol{q}_i\})e^{-\frac{3N}{gk_BT}\mathcal{H}}}{\int d\boldsymbol{p}_i d\boldsymbol{q}_i e^{-\frac{3N}{gk_BT}\mathcal{H}}}$$

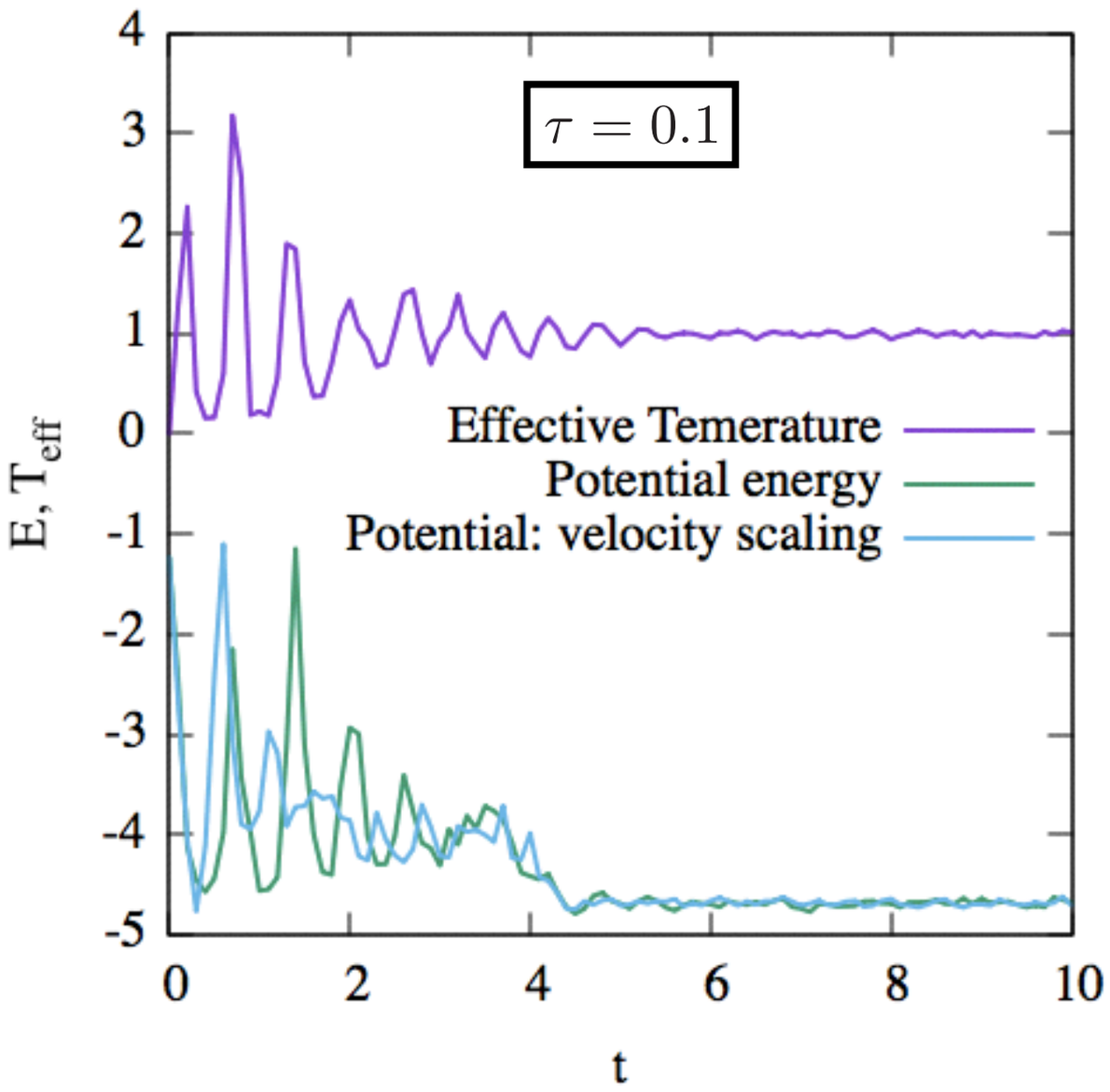
Canonical ensemble if $g = 3N$

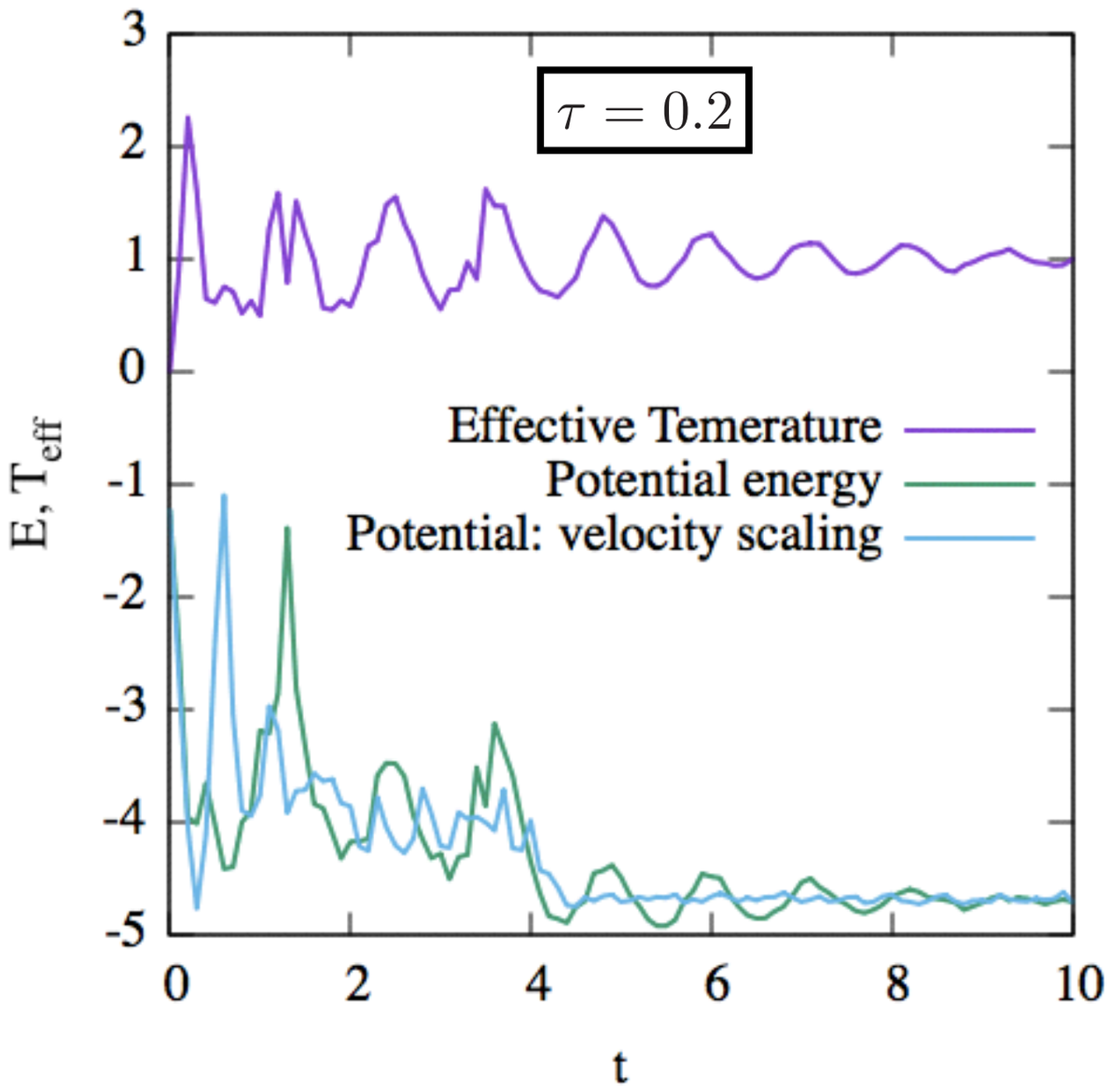
Results of the Nose-Hoover dynamics

- Temperature behaves like damped oscillation.
  - Period is related to τ (or Q)
- Potential energy converges almost same value with that of velocity scaling.

MD of LJ system

($\Delta t = 0.01$, N=1000, ρ=0.8, T=1)





Pressure control: Andersen method

H. C. Andersen, J. Chem. Phys. **72** (1980) 2384.

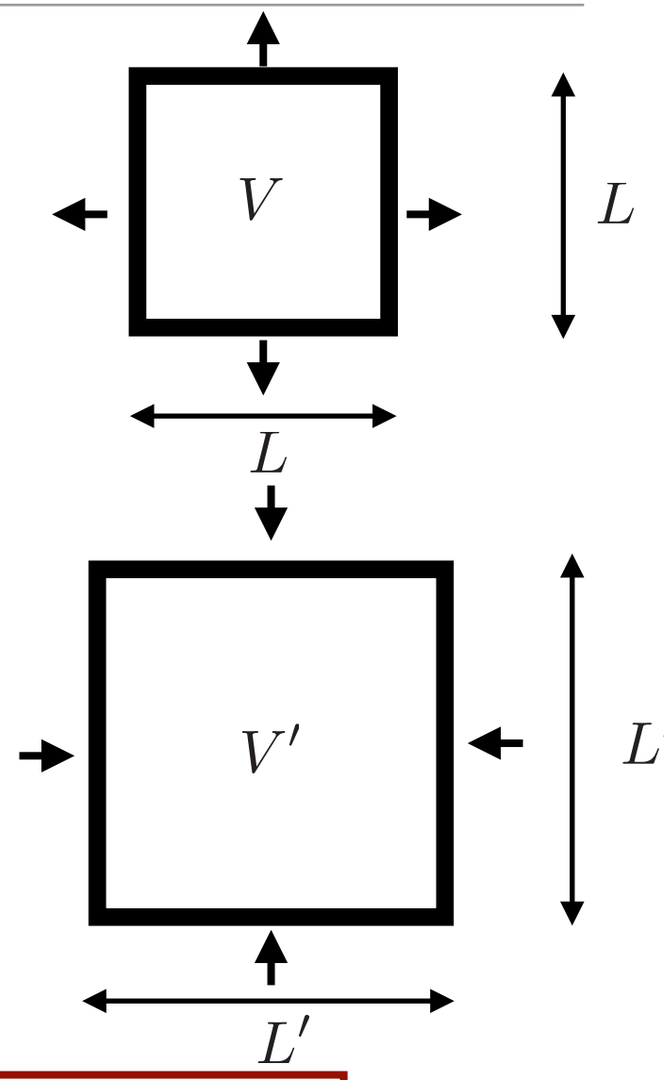
Extended Hamiltonian System with a "piston"

$$\mathcal{H}_A = \sum_i \frac{\tilde{\boldsymbol{p}}_i^2}{2m_i V^{\frac{2}{3}}} + V_p(\{V^{\frac{1}{3}}\tilde{\boldsymbol{q}}_i\}) + \frac{P_V^2}{2M} + PV$$

Original Hamiltonian with scaled coordinate and momentum

Piston

$$\tilde{\boldsymbol{q}}_i = V^{-\frac{1}{3}}\boldsymbol{q}_i$$
$$\tilde{\boldsymbol{p}}_i = V^{\frac{1}{3}}\boldsymbol{p}_i$$



Canonical equation

$$\frac{d\tilde{\boldsymbol{q}}_i}{dt} = \frac{\tilde{\boldsymbol{p}}_i}{m_i V^{\frac{2}{3}}} \qquad \frac{dV}{dt} = \frac{P_V}{M}$$

$$\frac{d\tilde{\boldsymbol{p}}_i}{dt} = V^{\frac{1}{3}}\boldsymbol{F}_i(\{V^{\frac{1}{3}}\tilde{\boldsymbol{q}}_i\}) \qquad \frac{dP_V}{dt} = \frac{1}{3V}\sum_i \left[\frac{\tilde{\boldsymbol{p}}_i{}^2}{m_i V^{\frac{2}{3}}} + \boldsymbol{F}_i \cdot (V^{\frac{1}{3}}\tilde{\boldsymbol{q}}_i)\right] - P$$

Pressure control: Andersen method

H. C. Andersen, J. Chem. Phys. **72** (1980) 2384.

In original coordinates

$$\frac{d\boldsymbol{q}_i}{dt} = \frac{\boldsymbol{p}_i}{m_i} + \frac{\dot{V}}{3V}\boldsymbol{q}_i \qquad \frac{dV}{dt} = \frac{P_V}{M}$$

$$\frac{d\boldsymbol{p}_i}{dt} = \boldsymbol{F}_i - \frac{\dot{V}}{3V}\boldsymbol{p}_i \qquad \frac{dP_V}{dt} = \underline{\frac{1}{3V}\sum_i \left[\frac{\boldsymbol{p}_i^2}{m_i} + \boldsymbol{F}_i \cdot \boldsymbol{q}_i\right]} - P$$

P_{eff} :virial theorem

$$= P_{\mathrm{eff}} - P$$

New degree of freedom controls the pressure like a piston.

P_{v} changes the sign depending on the difference between the effective pressure and the aimed pressure.

Andersen method gives us “approximate” NPH ensemble.

H = Enthalpy

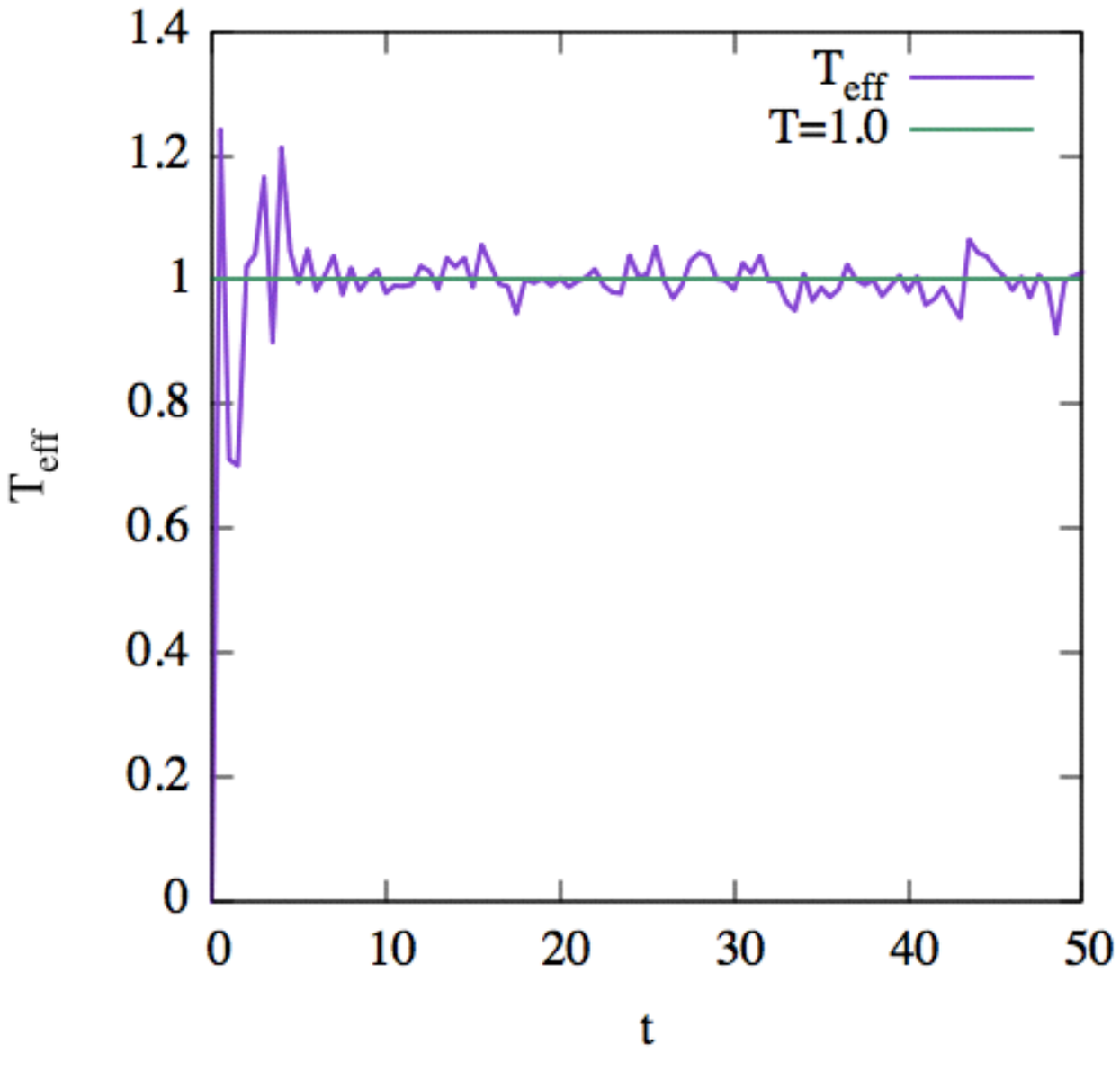
NPT ensemble

MD of LJ system

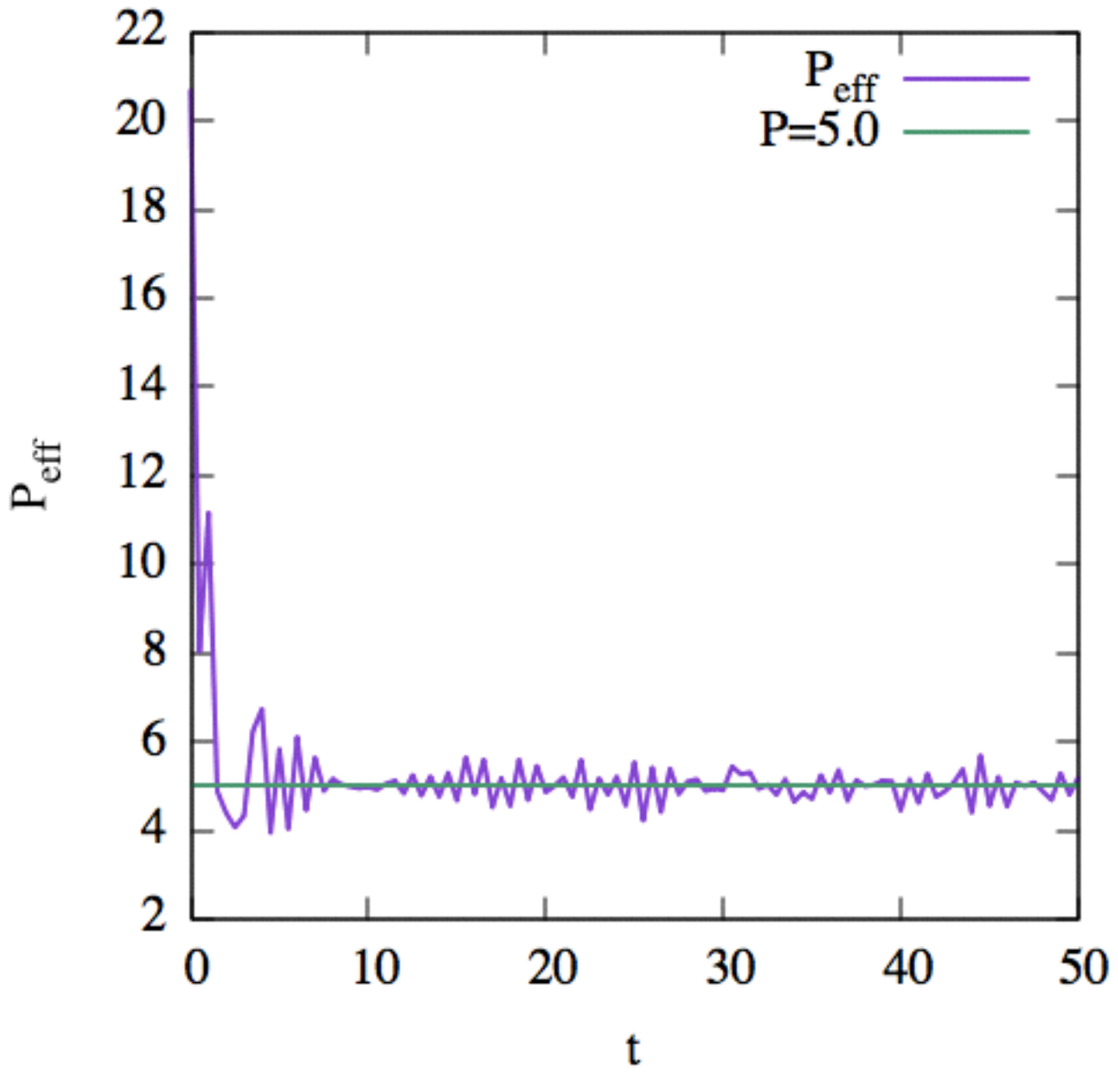
($\Delta t = 0.005$, N=1000, T=1, P=5)

By combining temperature and pressure controls,
we can obtain NPT ensemble. *e.g.* Nosé-Andersen method

Temperature



Pressure



Exercise: MD simulation of LJ particles(not a report)

Let's try MD simulation of LJ particles with NVE, NVT, and NPT ensembles.

- In NVE simulation (e.g. by Verlet method),
  see the conservation of the total energy.
- By using, velocity scaling or Nose-Hoover thermostat,
  try to control temperature.
- By combining temperature control and pressure control
  try to simulate NPT ensemble.

To perform these exercise, you can use,

- Your own code
- LAMMPS
  - http://lammps.sandia.gov
- MDACP (for NVE simulation.)
  - http://mdacp.sourceforge.net/index.html
- My sample codes for jupyter notebook (or python2.7).
  - In order to run the sample code you need
    - numpy, and numba (numba is used for speed up)

Next week

Classical

第１回：　物理学における多体問題
第２回：　多体問題における困難
第３回：　古典統計力学模型と数値計算
第４回：　古典モンテカルロ法とその応用
第５回：　分子動力学法とその応用
第６回：　拡張アンサンブル法によるモンテカルロ計算
(Extended ensemble method for Monte Carlo Methods)

Quantum

第７回：　量子統計力学模型と数値計算
第８回：　量子モンテカルロ法
第９回：　量子モンテカルロ法の応用
第１０回：　量子多体問題と巨大な疎行列の線形代数
第１１回：　クリロフ部分空間法の量子多体問題への応用
第１２回：　巨大な疎行列と量子統計力学
第１３回：　多体問題の並列計算アルゴリズム