MICCoM

MIDWEST INTEGRATED CENTER FOR COMPUTATIONAL MATERIALS

http://miccom-center.org

2017 Summer School

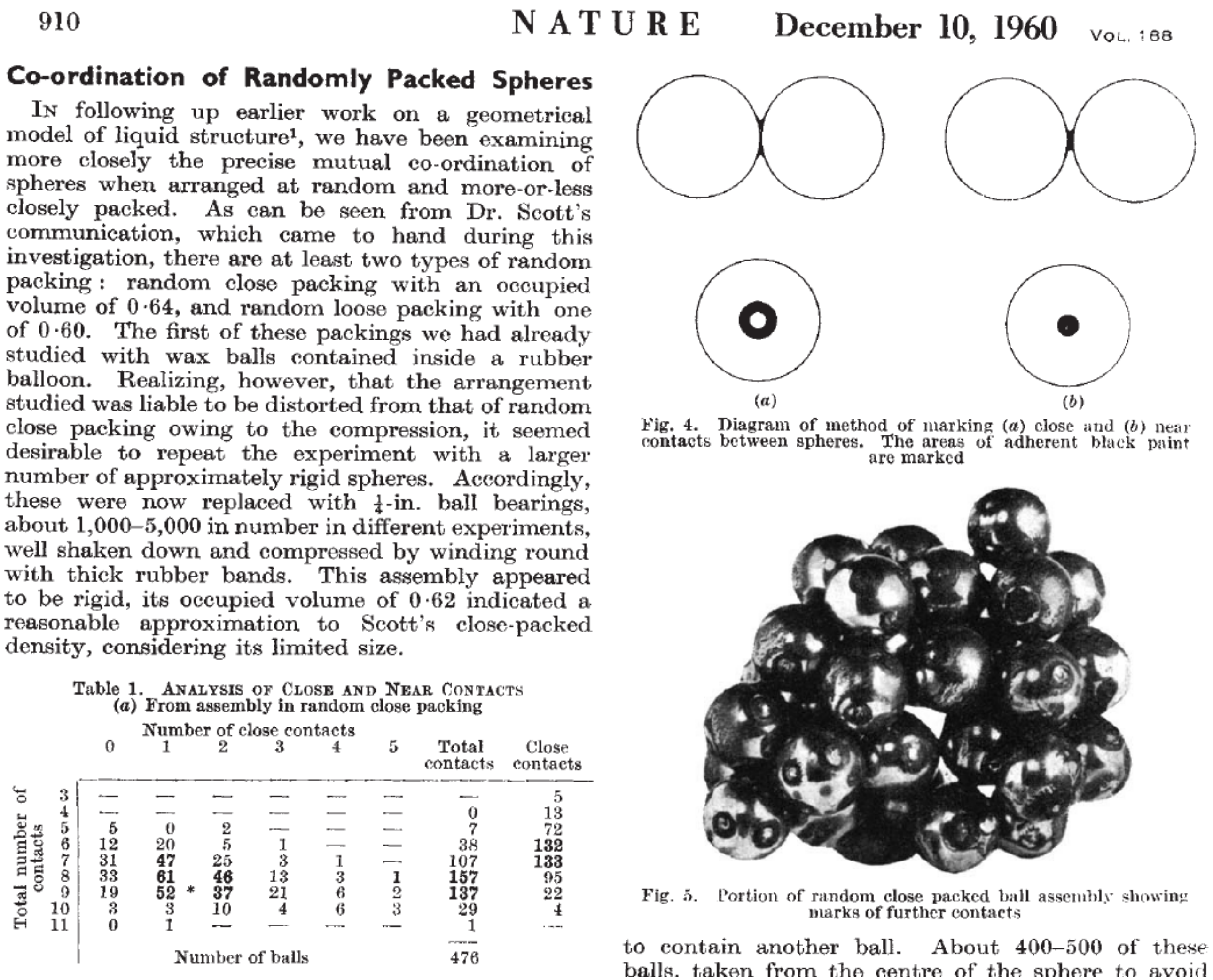
Topic: Classical Molecular Dynamics and Sampling Methods
Presenter: Jonathan K. Whitmer, University of Notre Dame

MICCoM

WHAT IS CLASSICAL MOLECULAR DYNAMICS (MD)?

- Classical MD is a way to simulate the behavior of explicit particles, including atoms, ions, and vastly more complex materials.
- Particles interact via relatively simple analytical potential energy functions.
- From the potential energy, we compute forces, torques and integrate Newton's equations
- Can treat millions of particles (especially new parallel GPU-based codes)
- Usually obtains dynamics on the order of nanosecond (full atom), microsecond (coarse-grained), or second (very coarse-grained)

“MECHANICAL” MODELS



910 NATURE December 10, 1960 Vol. 188

Co-ordination of Randomly Packed Spheres

In following up earlier work on a geometrical model of liquid structure[1], we have been examining more closely the precise mutual co-ordination of spheres when arranged at random and more-or-less closely packed. As can be seen from Dr. Scott's communication, which came to hand during this investigation, there are at least two types of random packing: random close packing with an occupied volume of 0·64, and random loose packing with one of 0·60. The first of these packings we had already studied with wax balls contained inside a rubber balloon. Realizing, however, that the arrangement studied was liable to be distorted from that of random close packing owing to the compression, it seemed desirable to repeat the experiment with a larger number of approximately rigid spheres. Accordingly, these were now replaced with ¼-in. ball bearings, about 1,000–5,000 in number in different experiments, well shaken down and compressed by winding round with thick rubber bands. This assembly appeared to be rigid, its occupied volume of 0·62 indicated a reasonable approximation to Scott's close-packed density, considering its limited size.

Table 1. Analysis of Close and Near Contacts
(*a*) From assembly in random close packing

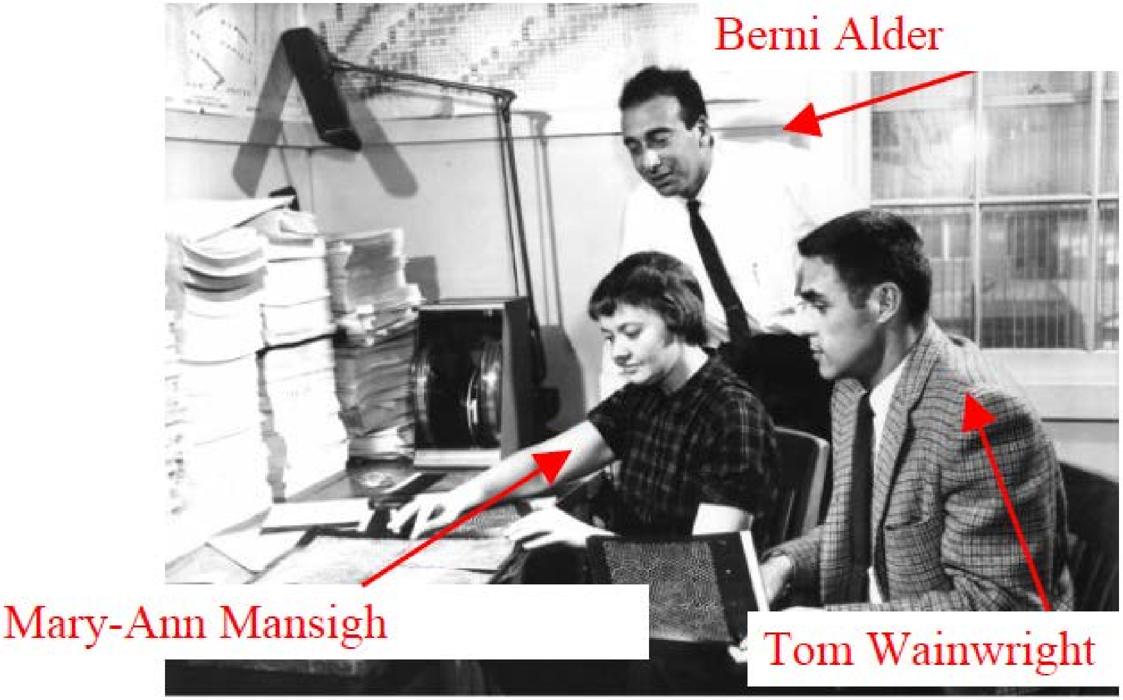
Total number of contacts	Number of close contacts 0	1	2	3	4	5	Total contacts	Close contacts
3	—	—	—	—	—	—	—	5
4	—	—	—	—	—	—	0	13
5	5	0	2	—	—	—	7	72
6	12	20	5	1	—	—	38	**132**
7	31	**47**	25	3	1	—	107	**133**
8	33	**61**	**46**	13	3	**1**	**157**	95
9	19	**52** *	**37**	21	6	2	**137**	22
10	3	3	10	4	6	3	29	4
11	0	1	—	—	—	—	1	—
Number of balls							476	

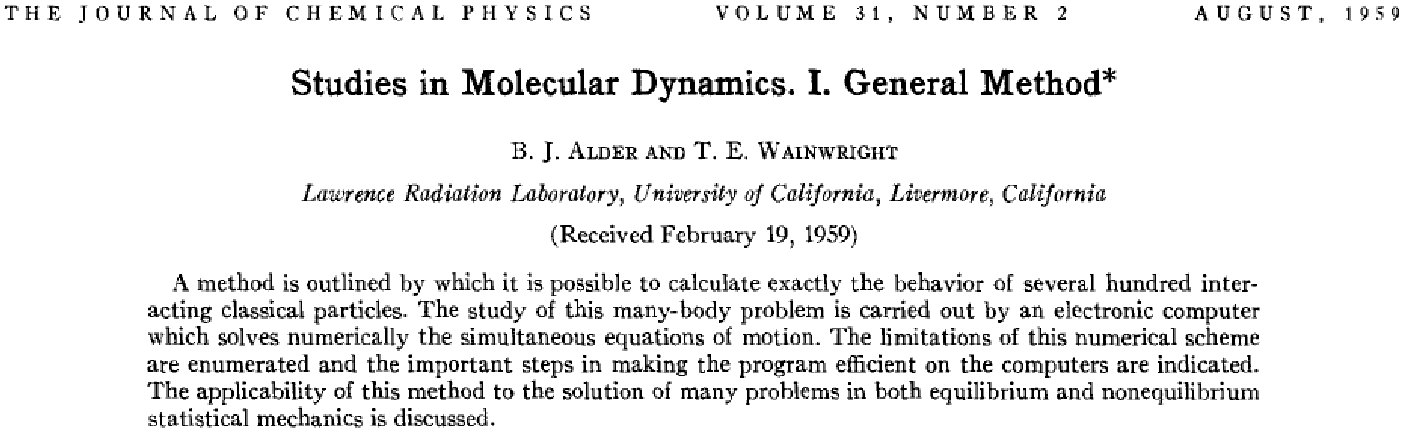
Fig. 4. Diagram of method of marking (*a*) close and (*b*) near contacts between spheres. The areas of adherent black paint are marked

Fig. 5. Portion of random close packed ball assembly showing marks of further contacts

to contain another ball. About 400–500 of these balls, taken from the centre of the sphere to avoid

COMPUTERS MAKE THIS EASIER!





THE JOURNAL OF CHEMICAL PHYSICS VOLUME 31, NUMBER 2 AUGUST, 1959

Studies in Molecular Dynamics. I. General Method*

B. J. ALDER AND T. E. WAINWRIGHT
Lawrence Radiation Laboratory, University of California, Livermore, California

(Received February 19, 1959)

A method is outlined by which it is possible to calculate exactly the behavior of several hundred interacting classical particles. The study of this many-body problem is carried out by an electronic computer which solves numerically the simultaneous equations of motion. The limitations of this numerical scheme are enumerated and the important steps in making the program efficient on the computers are indicated. The applicability of this method to the solution of many problems in both equilibrium and nonequilibrium statistical mechanics is discussed.

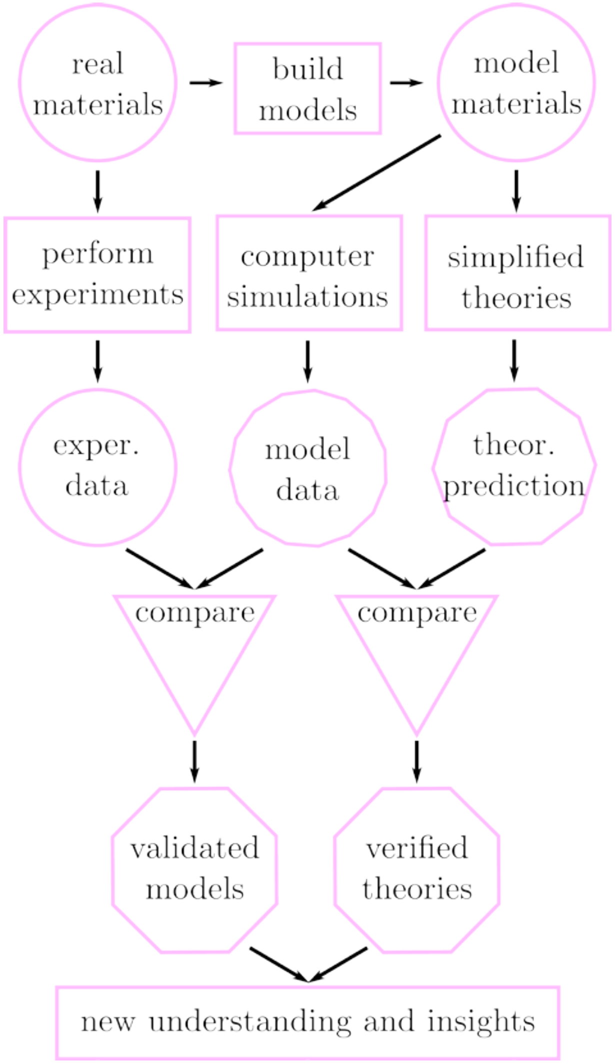
WHY DO WE WANT TO CONDUCT MOLECULAR SIMULATIONS?

1. To *predict* properties of (new) materials
2. To *understand* phenomena at a molecular level
3. To model *known* phenomena?

Why would we model the melting point of water?

Do this to *test* models and methods.
[Verification and Validation!]

CONNECTION TO EXPERIMENTS



Adapted from Allen and Tildesley, “Computer Simulation of Liquids”

PHASE SPACE

State vector or *system point* in a constant-volume system is defined as

$$\mathbf{\Xi} := (\mathbf{X}; \mathbf{P}) = (\mathbf{x}_1, \ldots, \mathbf{x}_N; \mathbf{p}_1, \ldots, \mathbf{p}_N)$$

"$6N$"-dimensional set Ω from which $\mathbf{\Xi}$ takes a value is called the *phase space* of the system

A point $\mathbf{\Xi}$ in Ω is a *microstate* of the system.

If $\mathbf{\Xi}$ is known at any time, it is *completely determined* for all other times (past and present) through the classical equations of motion.

MD ALGORITHMS

- An MD simulation generates a connected trajectory of points in phase space.
- The motion is given by *Hamilton's* equations (which are essentially Newton's equations)

$$\begin{aligned} H(\mathbf{x};\mathbf{p}) &= U(\mathbf{x}) + K(\mathbf{p}) \\ \dot{\mathbf{x}}_i &= \frac{\partial H}{\partial \mathbf{p}_i} = \frac{\mathbf{p}_i}{m_i} \\ \dot{\mathbf{p}}_i &= -\frac{\partial H}{\partial \mathbf{x}_i} = -\mathbf{F}_i \end{aligned}$$

MICROSTATES, MACROSTATES AND ENSEMBLES

Systems with different *microstates* may exist at the same *thermodynamic state*.

Example: Many different sets of $(\boldsymbol{x}; \boldsymbol{p})$ can be at the same temperature and volume.

A thermodynamic state is a *collection* of many different microstates.

- Identical in composition and at the same “macrostate”
- Gibbs called this collection an “ensemble”; NVE is natural for MD
- “macrostate” has a definition in terms of statistical labels as well, we will come back to this for “advanced sampling”

OBSERVATION: COMPUTING AVERAGES IN MOLECULAR SIMULATIONS

To compute the average of a quantity *f*, you can:

1. Observe a single trajectory $\boldsymbol{\Xi}(\mathrm{t})$ as it passes through phase space Ω, and compute $\langle f \rangle_t = \tau^{-1} \int f(t)dt$. This is how *molecular dynamics* (MD) works.
2. Take snapshots of the constant energy hypersurface $\Gamma = \{\boldsymbol{\Xi} \in \Omega \,|H(\boldsymbol{\Xi}) = E\}$ at various times and compute $\langle f \rangle_{NVE} = \frac{1}{N}\sum_i f(\boldsymbol{\Xi}_i)$. This is the principle behind *Monte Carlo* (MC).

Ergodic hypothesis: both methods are equivalent.

For swift sampling, we'll make use of both!

NVE? SURE, THE UNIVERSE IS CONSTANT-ENERGY…

…but most systems of interest are not studied this way.

Why not?

- The Universe is isolated, but most systems have thermal contact with others, through which energy may be exchanged.
- Systems may exchange volume or particle number with surrounding environment in order to reach equilibrium (constant P, $\boldsymbol{\mu}$)

NVE is the natural ensemble for molecular dynamics, however; so what are we to do?

ENSEMBLES

ENSEMBLES

Assumptions: The “universe” is *fixed volume* (sorry cosmologists!) and *total* energy (sum of system and universe) is constant.

PROBABILITY

- Suppose we have a known state with energy E_A in system A, and energy E_{tot} overall.

$$E_{\mathrm{univ}} = E_{\mathrm{tot}} - E_A$$

- Setting the state of A forces the surroundings to adopt a known energy, which can occur in multiple ways.

- Denote $W_i(E)$ to be the “density of states in system i; the number of ways the energy can be partitioned. Each of these states is equally likely when the total energy is fixed.

PROBABILITY

- What is the probability of system A having energy E_i?

$$P(E_A = E_i) = \frac{\text{sum of microstates with } E_A{=}E_i}{\text{sum of all microstates}}$$

$$= W_A(E_i)\frac{W_{\text{univ}}(E_{\text{tot}}-E_i)}{\sum_j W_A(E_j)W_{\text{univ}}(E_{\text{tot}}-E_j)}$$

$$\log(P(E_A = E_i)) = \log\left(\frac{W_A(E_i)W_{\text{univ}}(E_{\text{tot}}-E_i)}{\sum_j W_A(E_j)W_{\text{univ}}(E_{\text{tot}}-E_j)}\right)$$

$$\log W_{\text{univ}}(E_{\text{tot}}-E_i) \approx \log W_{\text{univ}} - E_i \underbrace{\frac{\partial \log W_{\text{univ}}(E_{\text{tot}})}{\partial E_{\text{tot}}}}_{\boxed{\beta = 1/k_B T}} + \mathcal{O}(1/E_{\text{tot}})$$

$$\boxed{W_{\text{univ}}(E_{\text{tot}}-E_i) \approx W_{\text{univ}}(E_{\text{tot}})e^{-\beta E_i}}$$

$$\boxed{P^A(E_i) := P(E_A = E_i) = \frac{W_A(E_i)e^{-\beta E_i}}{\sum_j W_A(E_i)e^{-\beta E_j}}}$$

“Boltzmann distribution”

Probability of finding the system in microstate *i* is

$$P_i = \frac{e^{-\beta E_i}}{\sum_j e^{-\beta E_j}} = \frac{e^{-\beta E_i}}{Z}$$

and defines the “partition function” Z.

Average of *any* property A is

$$\langle A \rangle = \sum_i P_i A_i = \sum_i \frac{A_i e^{-\beta E_i}}{Z}$$

In particular energy…

$$\langle E \rangle = \sum_i P_i E_i = \sum_i \frac{E_i e^{-\beta E_i}}{Z}$$

Re-write this last expression

$$\langle E \rangle_{NVT} = -\frac{\partial \log\left(\sum_i e^{-\beta E_i}\right)}{\partial \beta}$$

$$\langle E \rangle_{NVT} = -\frac{\partial \log Z}{\partial \beta}$$

Classical thermo…

$$E = U = \frac{\partial A/T}{\partial(1/T)}$$

Helmholtz energy related to log of partition function

$$A = -k_B T \log Z$$

This is a general formula for any ensemble; free energy is related to the logarithm of an ensemble partition function.

CAN Z BE COMPUTED?

Given a partition function, all thermodynamics of the ensemble are specified. Can we directly compute partition function?

Assume a 2-state system (“up” or “down”). Number of configurations for N “particles”?

2^N

So for N=100, must make 2^{100} evaluations ~ 1×10^{30}. This is impossible! Thus, we resort to averaging in classical MD and MC.

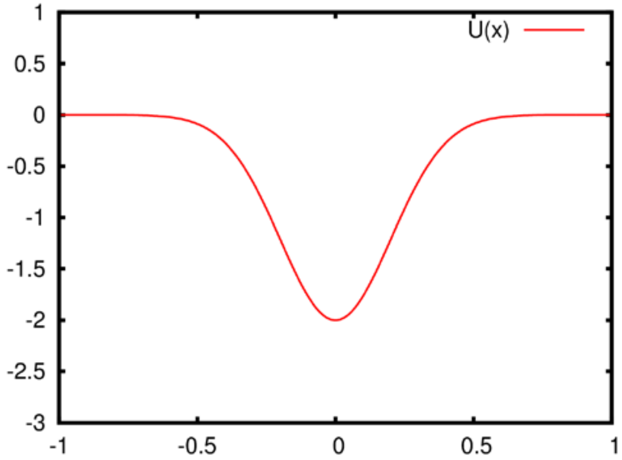
FUNDAMENTALS OF MD

- Consider a particle within a one dimensional Gaussian well at constant temperature. The potential energy is

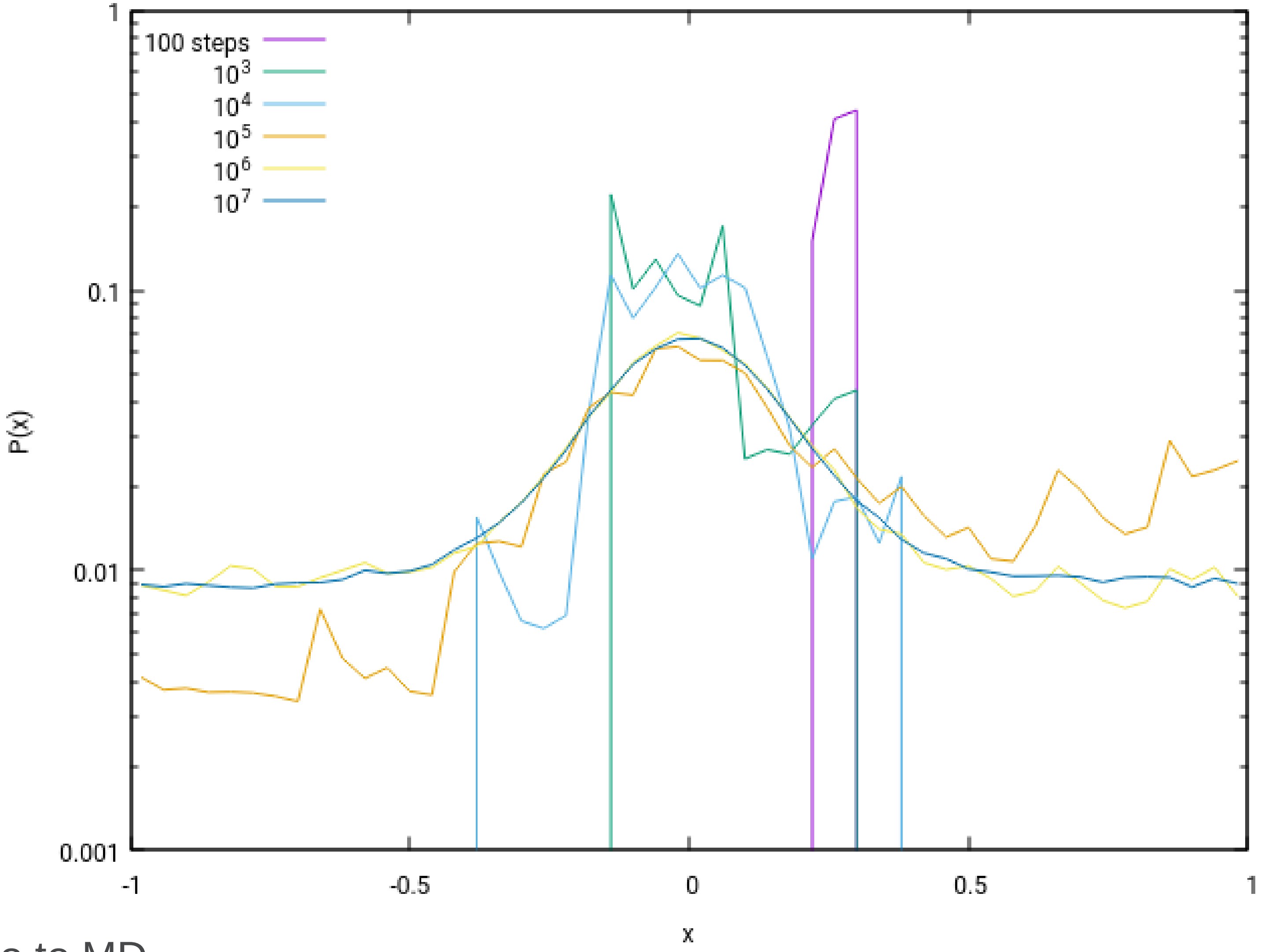
$$U(x) = U_0 e^{-\frac{x^2}{2\sigma^2}}$$

on the domain $x \in [-1,1]$ with $U_0 = -2$ and $\sigma = 0.2$:

  - How do I compute $\langle x \rangle$?
  - What is $\langle v \rangle$?
  - What should $P(x)$ look like?



DEMONSTRATION OF ERGODICITY

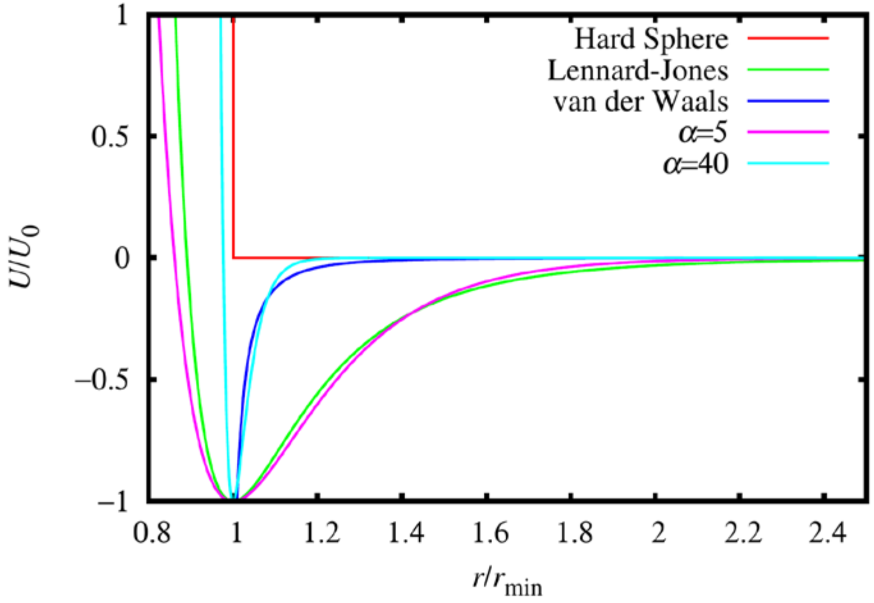


PAIR INTERACTIONS

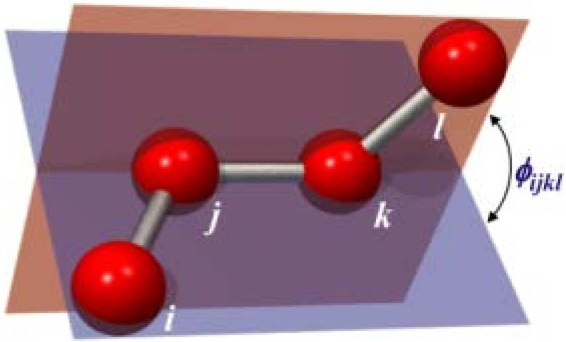
- Most of the interactions we think of normally are pair interactions; these are easiest to consider in thinking about extra-molecular systems (interior to the molecule, other forces contribute).

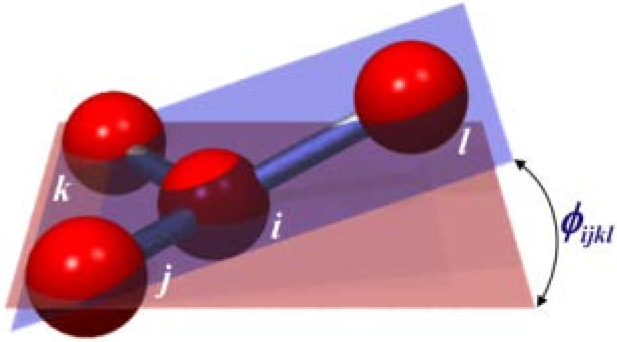
$$U(\mathbf{x}) = \sum_{\text{pairs } i,j} U_{\text{pair}}(\mathbf{r}_{ij})$$

$$\mathbf{F}_i = -\nabla_{\mathbf{x}_i} U(\mathbf{x})$$



MOLECULES



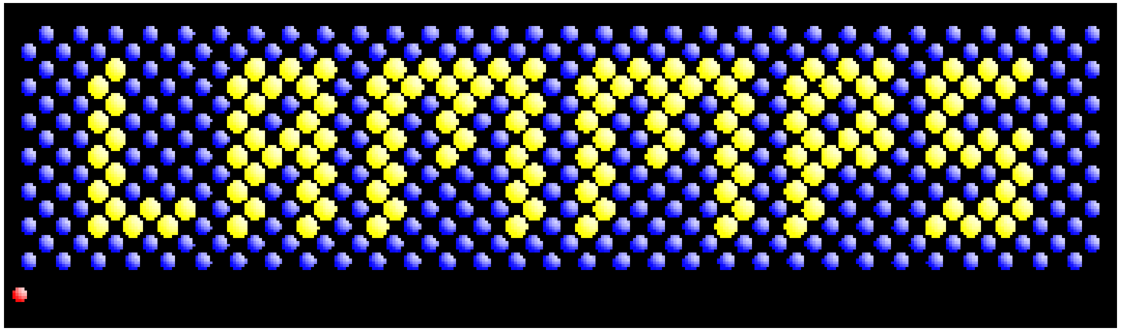


$$U(\mathbf{x}) = \sum_{\text{pairs } i,j} U^{ij}_{\text{pair}}(\mathbf{r}_{ij}) + \sum_{\text{bonded } i,j} U^{ij}_{\text{bond}}(\mathbf{r}_{ij}) + \sum_{\text{angles}} U^{ijk}_{\text{angle}}(\mathbf{r}_{ij}, \mathbf{r}_{jk}) + \sum_{\text{dihedrals}} U^{ijkl}_{\text{dihedral}}(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k, \mathbf{x}_l)$$

http://cbio.bmt.tue.nl/pumma/index.php/Theory/Potentials

BACK TO THE EQUATIONS OF MOTION

- What are the minimal requirements for an integrator?
- To faithfully integrate Newton’s equations, we must have an integrator which is time reversible and conserves energy.
- NVE Molecular dynamics should run in forward or reverse.



http://lammps.sandia.gov

INTEGRATION

- Start with a Taylor approximation of the positions and velocities

$$x(t+\delta t) = x(t) + v(t)\delta t + \frac{1}{2}a(t)\delta t^2 + \frac{1}{6}j(t)\delta t^3 + \cdots$$
$$v(t+\delta t) = v(t) + a(t)\delta t + \frac{1}{2}j(t)\delta t^2 + \cdots$$

- Simplest method is to truncate each series at the first order. This is Euler's method.

$$x(t+\delta t) = x(t) + v(t)\delta t$$
$$v(t+\delta t) = v(t) + a(t)\delta t$$

PERFORMANCE: EULER INTEGRATOR

- The harmonic oscillator is exactly solvable. Given an initial condition $x(0), v(0)$ the solutions should be sine waves. For $m = k = 1$, starting from rest:

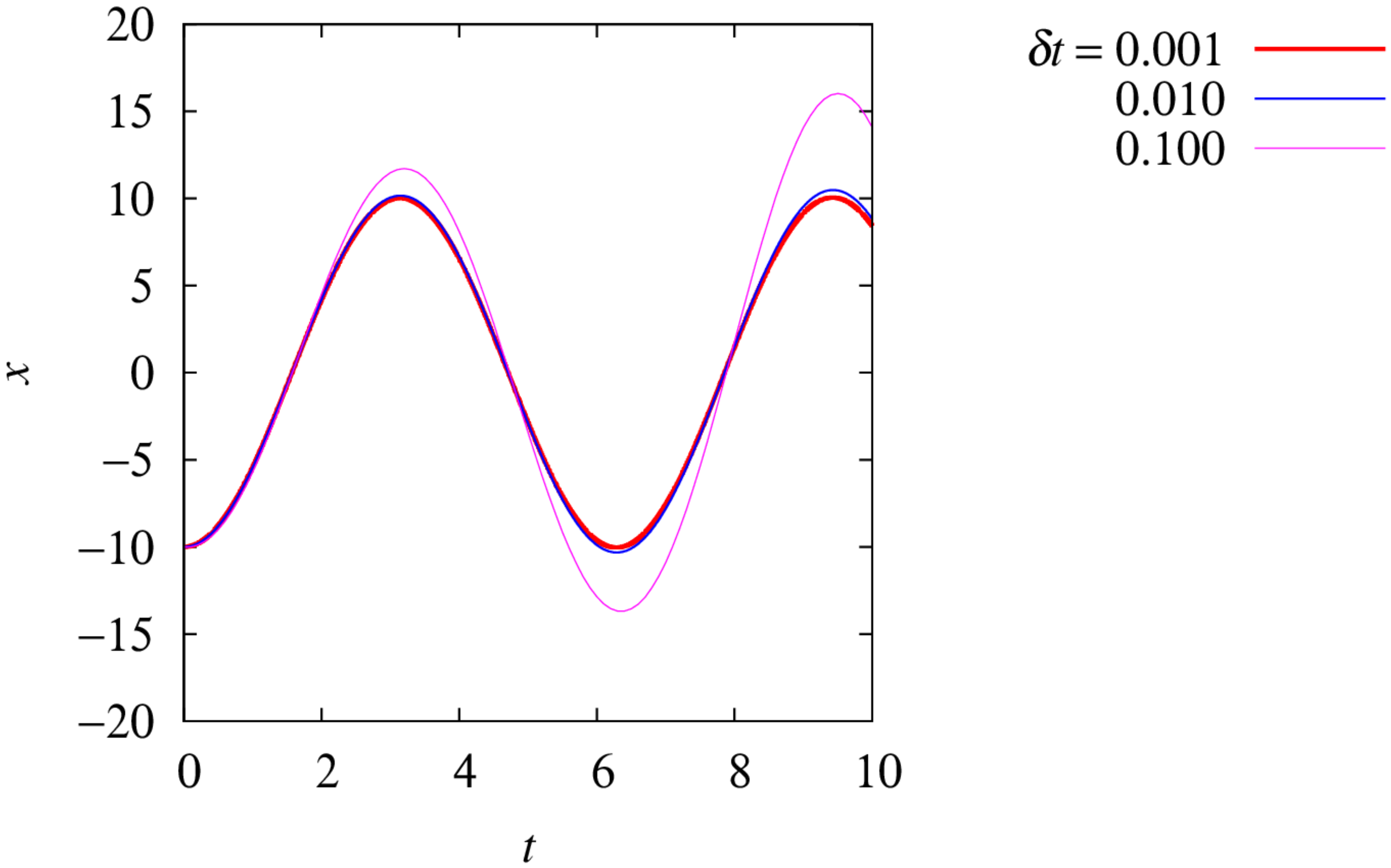
$$x(t) = x(0)\cos(t)$$
$$v(t) = x(0)\sin(t)$$

- The trajectory of $v(t)$ and $x(t)$ should trace a circle.
- Since $\cos^2 t + \sin^2 t = 1$ the energy

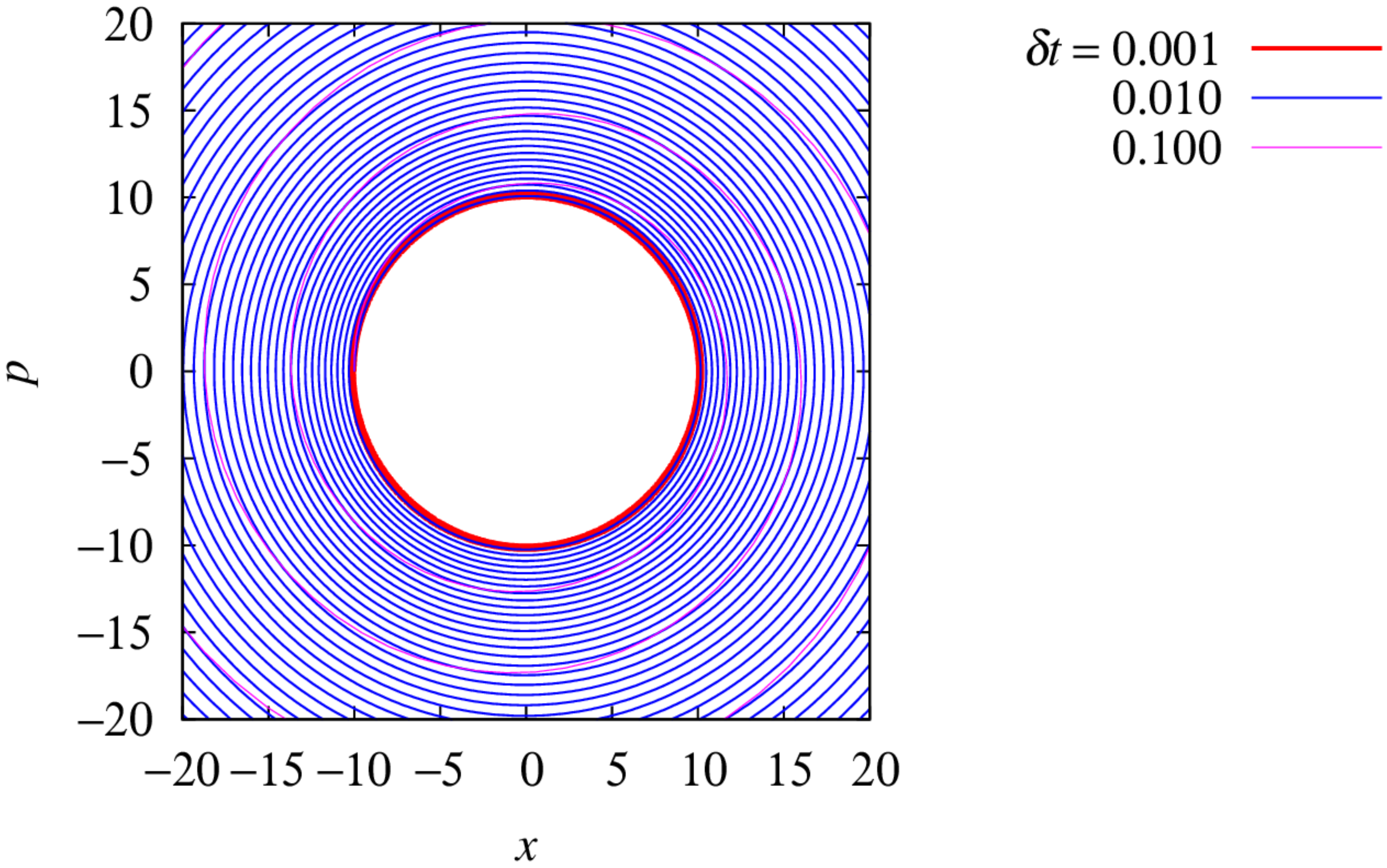
$$E = \frac{1}{2}v^2 + \frac{1}{2}x^2$$

is conserved.

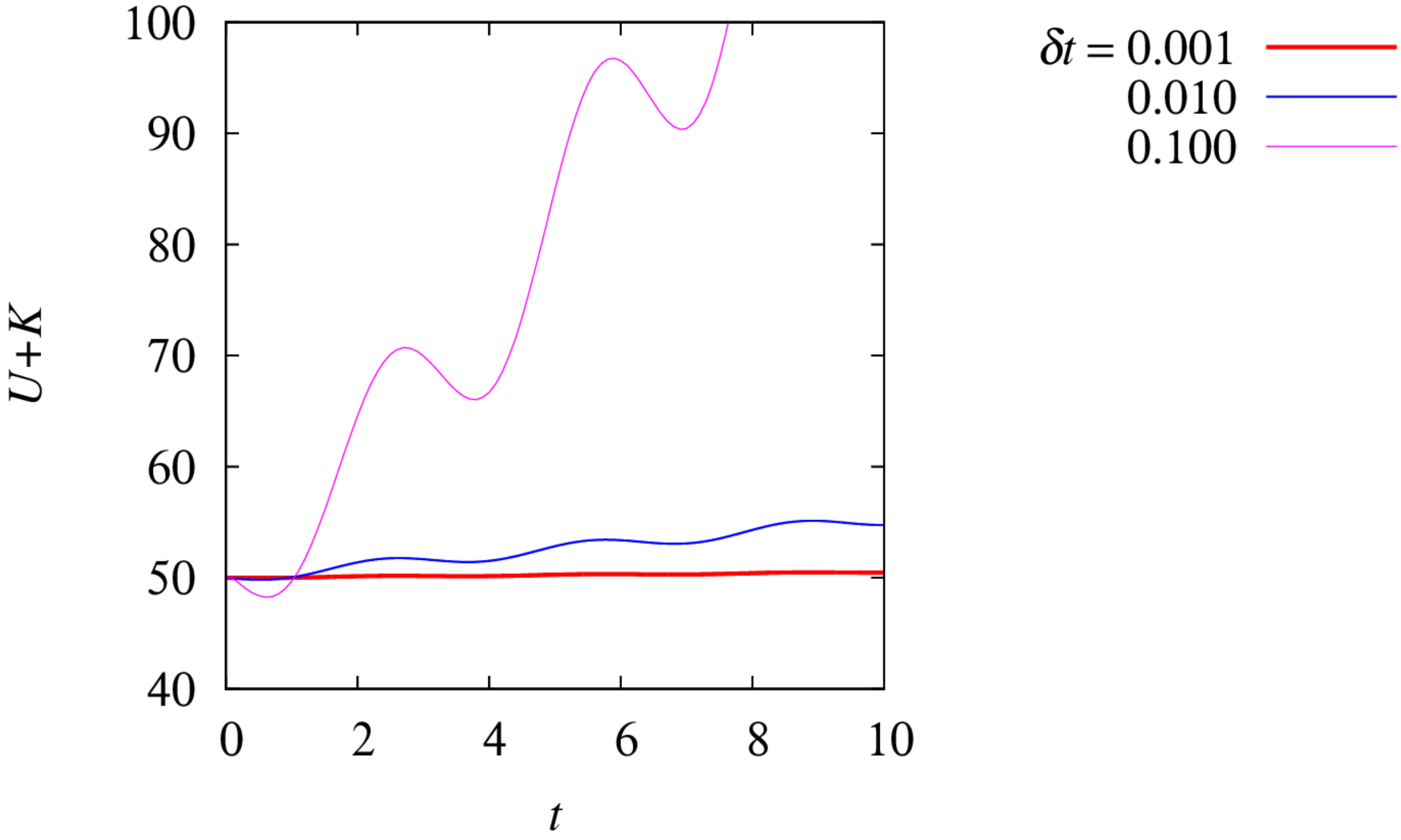
PERFORMANCE: EULER INTEGRATOR



PERFORMANCE: EULER INTEGRATOR



PERFORMANCE: EULER INTEGRATOR



INTEGRATION II

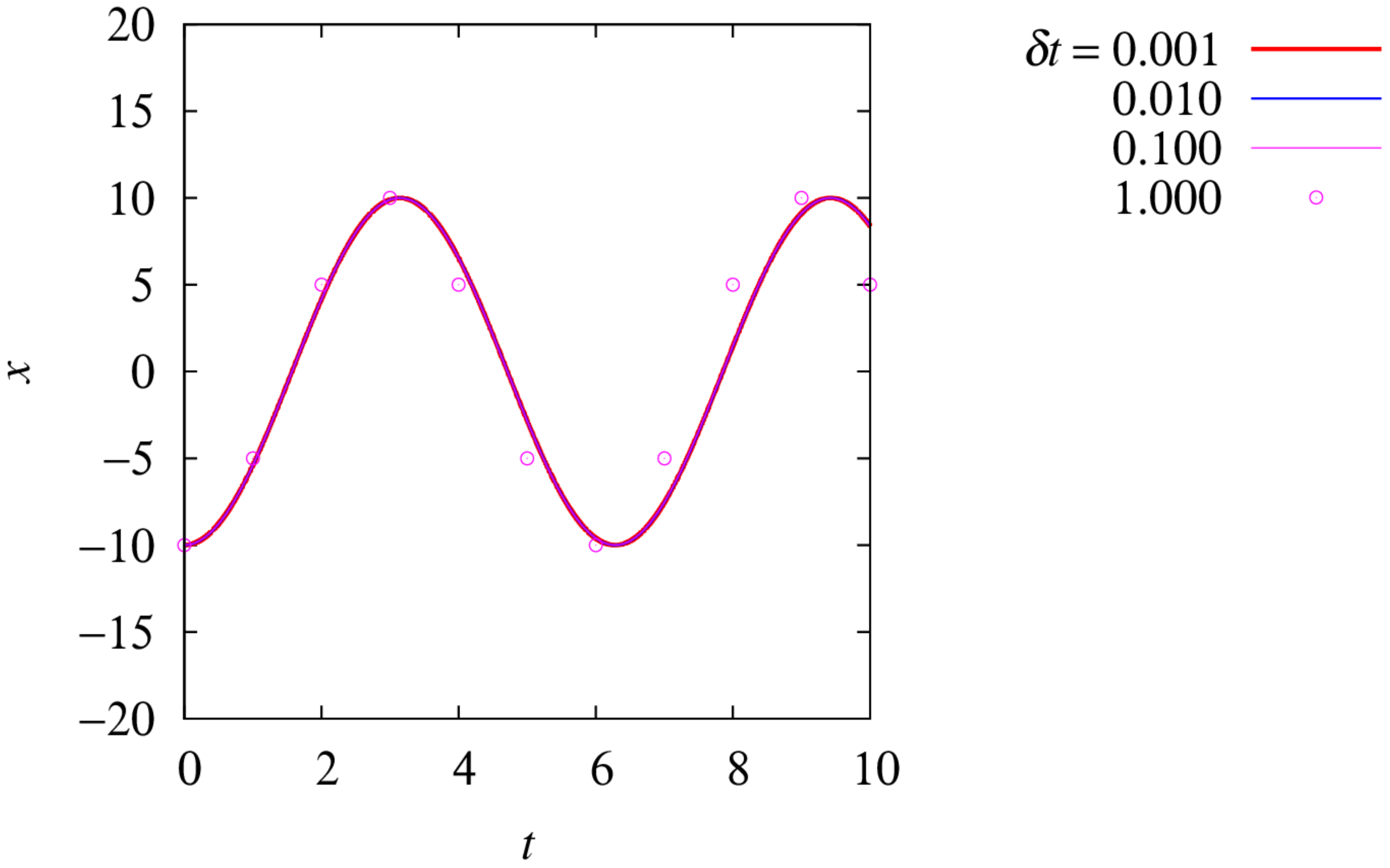
- One can improve on this by utilizing the *Verlet* integrator. Begin by writing the Euler propagation in forward and reverse directions

$$r(t+\delta t) = r(t) + v(t)\delta t + \frac{1}{2}a(t)\delta t^2 + \frac{1}{6}j(t)\delta t^3 + \cdots$$
$$r(t-\delta t) = r(t) - v(t)\delta t + \frac{1}{2}a(t)\delta t^2 - \frac{1}{6}j(t)\delta t^3 + \cdots$$

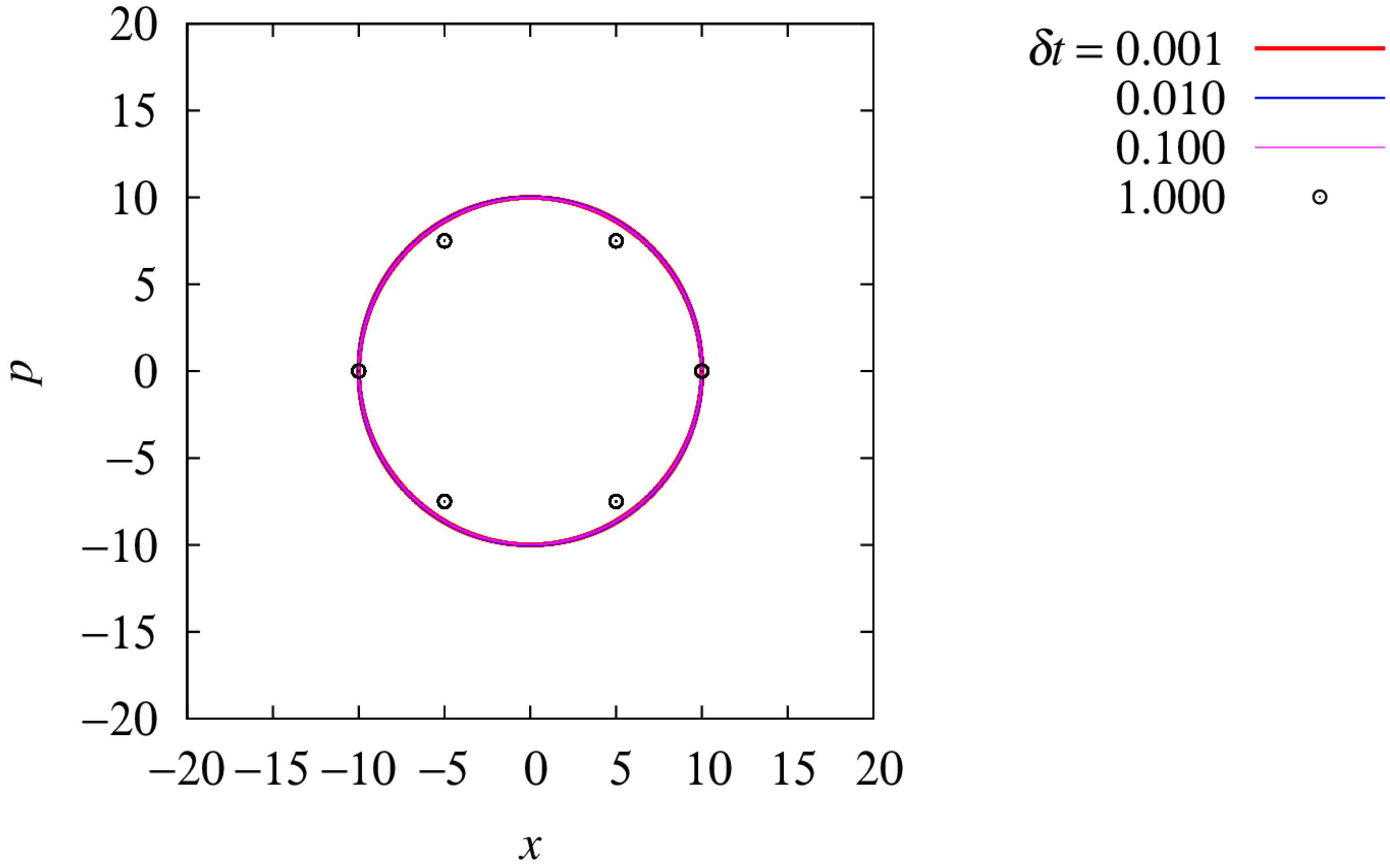
- Note that summing these together eliminates terms of δt^3 from the expression. Errors in the integration of positions. Since we know acceleration explicitly from the force field, will be order δt^4.

$$r(t+\delta t) - r(t-\delta t) = 2r(t) + a(t)\delta t^2$$

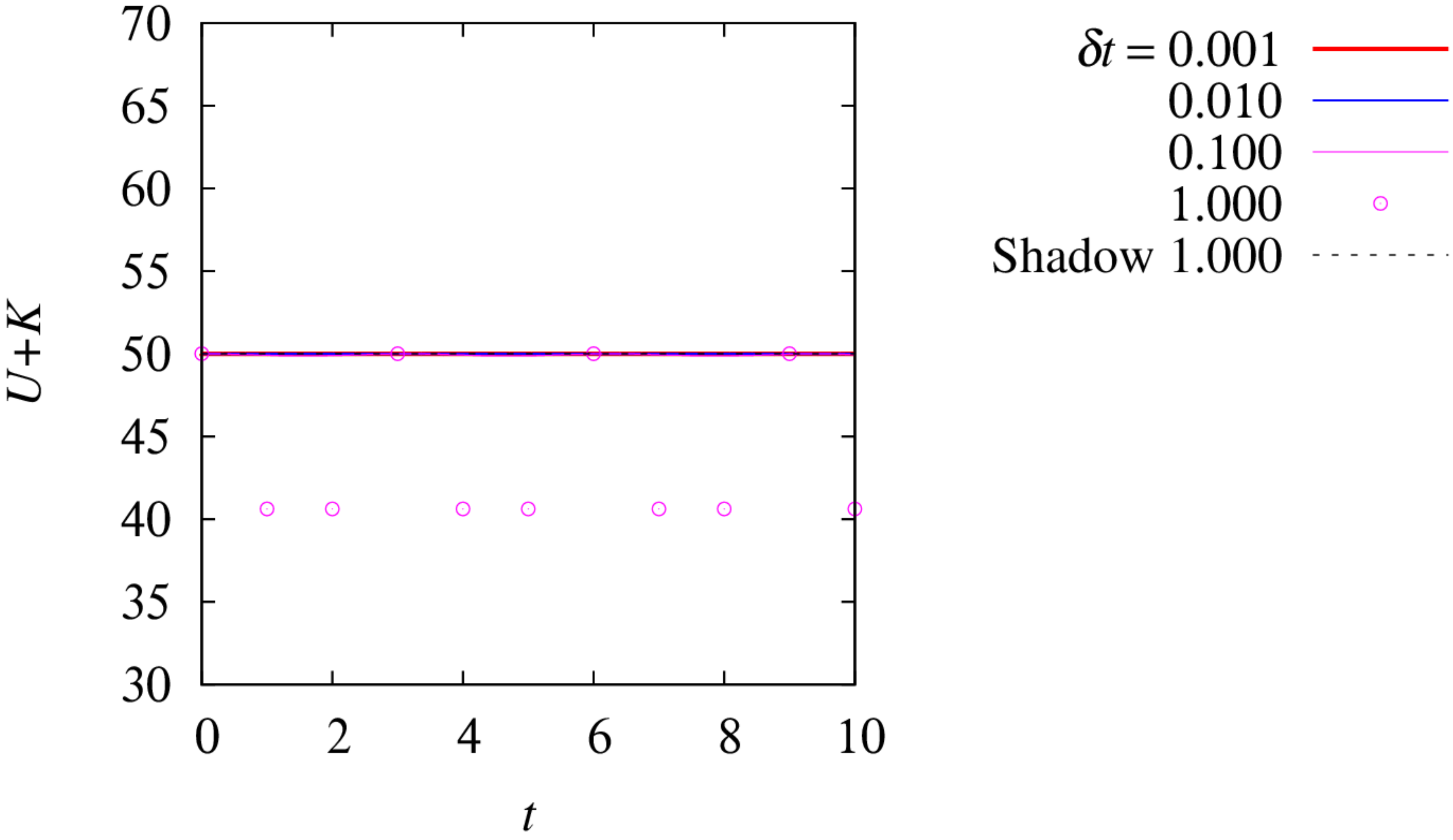
PERFORMANCE: VERLET INTEGRATOR



PERFORMANCE: VERLET INTEGRATOR



PERFORMANCE: VERLET INTEGRATOR



VARIATIONS

- A more common form of the Verlet algorithm utilizes a “half-step” velocity. This is the integrator used in LAMMPS.

$$\vec{x}(t+\delta t) = \vec{x}(t) + \vec{v}(t)\delta t + \frac{1}{2}\vec{a}(t)\delta t^2$$

$$\vec{v}\left(t+\frac{\delta t}{2}\right) = \vec{v}(t) + \frac{1}{2}\vec{a}(t)\delta t$$

$$\vec{a}(t+\delta t) = \vec{f}(t+\delta t) = -\nabla U\big(\vec{x}(t+\delta t)\big)$$

$$\vec{v}(t+\delta t) = \vec{v}\left(t+\frac{\delta t}{2}\right) + \frac{1}{2}\vec{a}(t+\delta t)$$

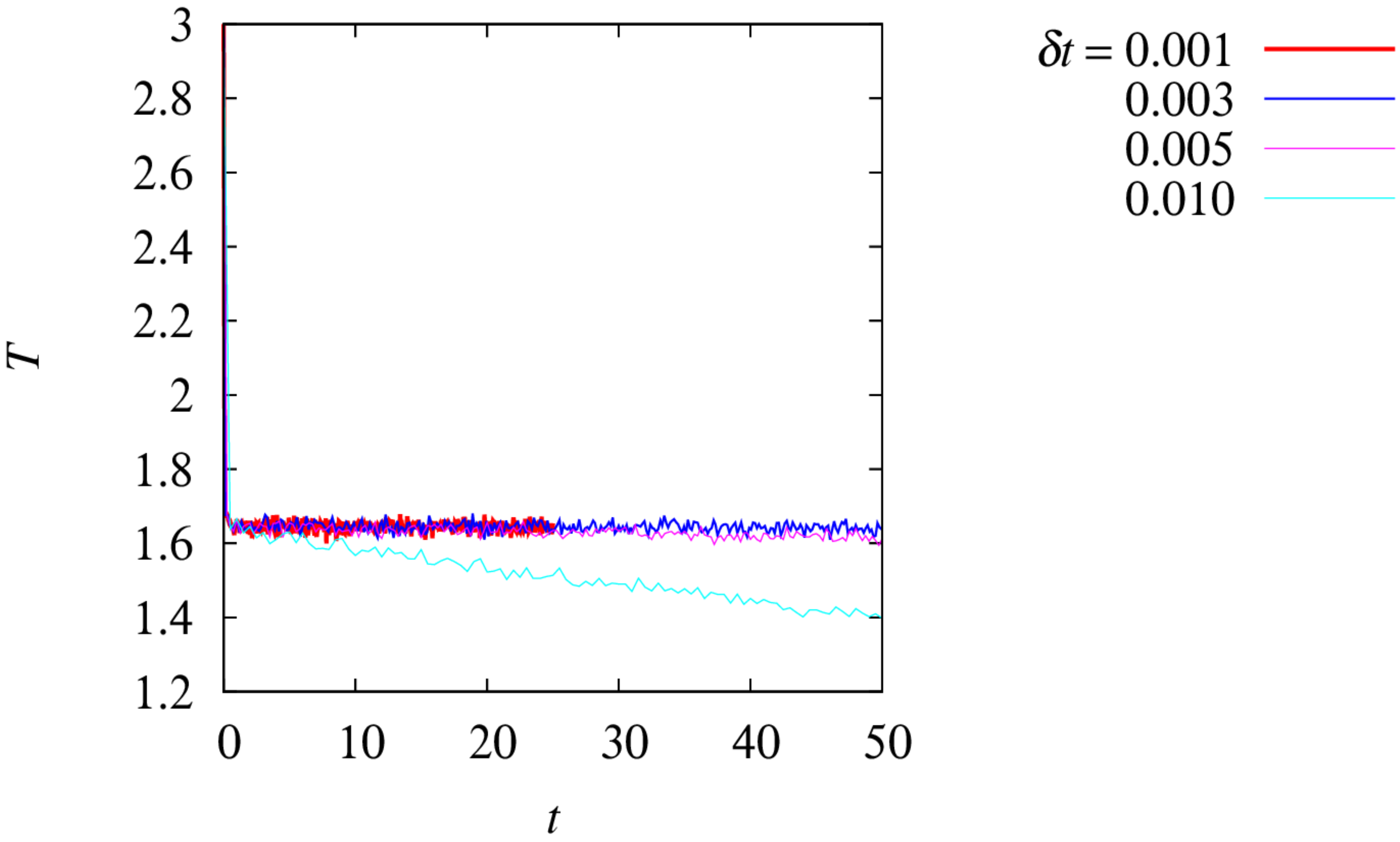
CASE STUDY: CHOOSING A TIMESTEP

Lennard-Jones Melt
Units are standard:

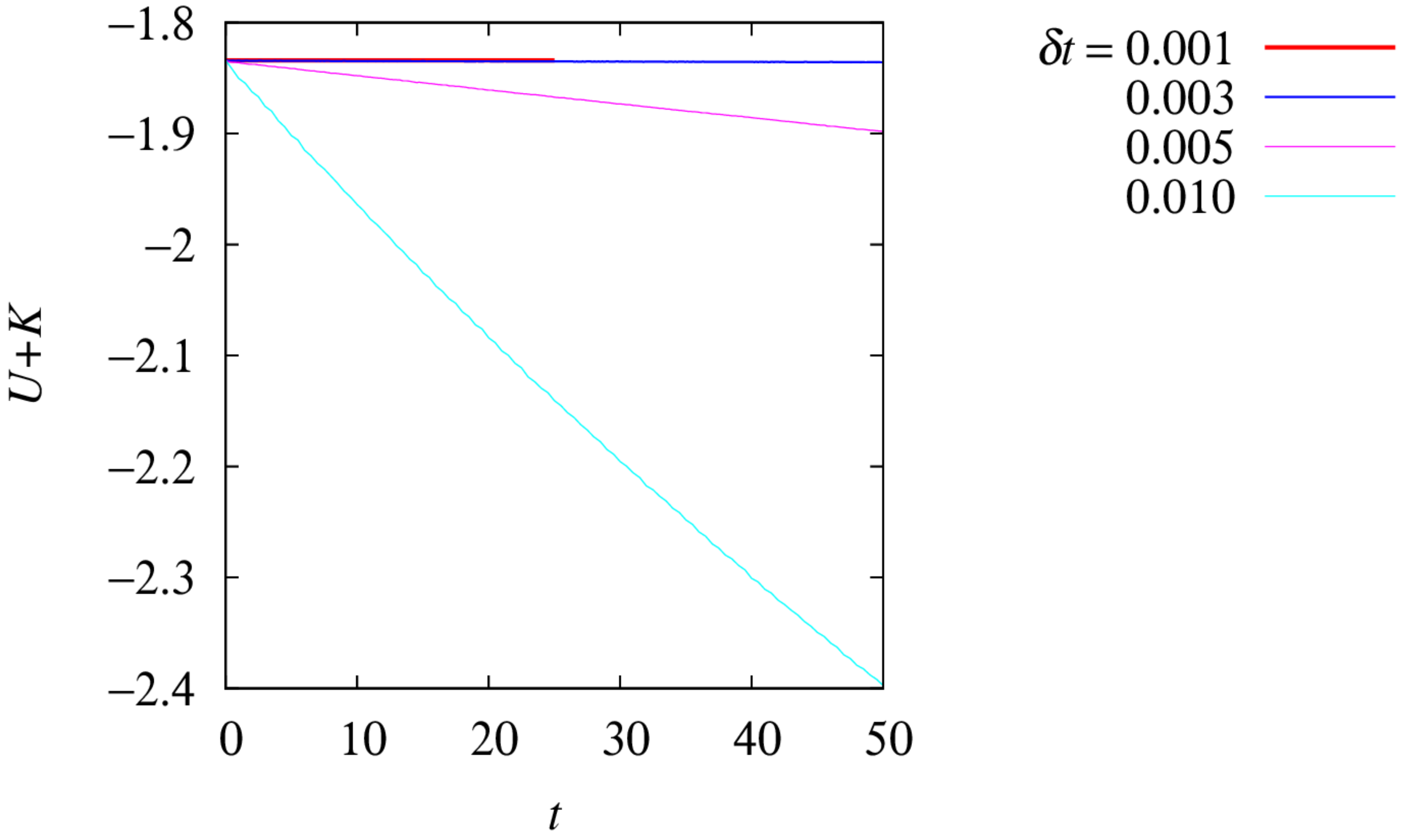
Energy: ϵ
Length: σ
Mass: m
Time: $\sigma\sqrt{m}/\sqrt{\epsilon}$

```
# 3d Lennard-Jones melt in periodic box

# set name for this simulation
variable NAME index melt # name style value

# set log file name
log ${NAME}.log

# configure and initialize system
units lj
atom_style atomic
pair_style lj/cut 2.5 # style args(=cutoff)
boundary p p p # x y z (p = periodic boundary conditions)

lattice fcc 0.8442 # style scale(=reduced_density)
region MYBOX block 0 10 0 10 0 10 # ID style args(=xlo,xhi,ylo,yhi,zlo,zhi)
create_box 1 MYBOX # number_of_atom_types region_ID

create_atoms 1 box # atom_type style args

mass 1 1.0 # atom_type mass
velocity all create 3.0 87287 # group_ID style args(=temp,seed)

# set forcefield parameters
pair_coeff 1 1 1.0 1.0 2.5 # atom_i atom_j args(=epsilon,sigma,cutoff)
pair_modify shift yes

# configure neighbor lists
neighbor 0.3 bin
neigh_modify every 20 delay 0 check no

# configure integrator
fix NVE all nve # ID group_ID style
timestep TS # The default timestep size is 0.005 for LJ units.

# configure output of atom-specific trajectory data (e.g. coordinates)
dump DUMP all atom 50 ${NAME}.lammpstrj # dump_ID group_ID style interval file args
#dump_modify DUMP image yes # dump_ID keyword value

# configure output of system property data
thermo_style custom step temp press pe ke etotal
thermo 50 # output_interval

# run simulation
run 25000 # number_of_steps
write_restart ${NAME}.restart.*
```

TEMPERATURE



TOTAL ENERGY



THERMOSTAT ALGORITHMS

$$T = \frac{2}{N_{\mathrm{df}} k_{\mathrm{B}}} K$$

- Simplest thermostat is rescaling, but this does not correspond to proper NVT ensemble fluctuations.
- Berendsen:

$$\ddot{\mathbf{x}}_i(t) = m_i^{-1}\mathbf{F}_i(t) - \frac{\Delta t}{2\tau_B}\left[\frac{\frac{N_{\mathrm{df}}-1}{N_{\mathrm{df}}}T_0}{T'(t)} - 1\right]\dot{\mathbf{x}}(t)$$

- Langevin/Stochastic:

$$\ddot{\mathbf{x}}_i(t) = m_i^{-1}\mathbf{F}_i(t) - \gamma_i(t)\dot{\mathbf{x}}(t) + m_i^{-1}\tilde{\mathbf{F}}_i(t)$$

$$0 = m_i^{-1}\mathbf{F}_i(t) - \gamma_i(t)\dot{\mathbf{x}}(t) + m_i^{-1}\tilde{\mathbf{F}}_i(t)$$

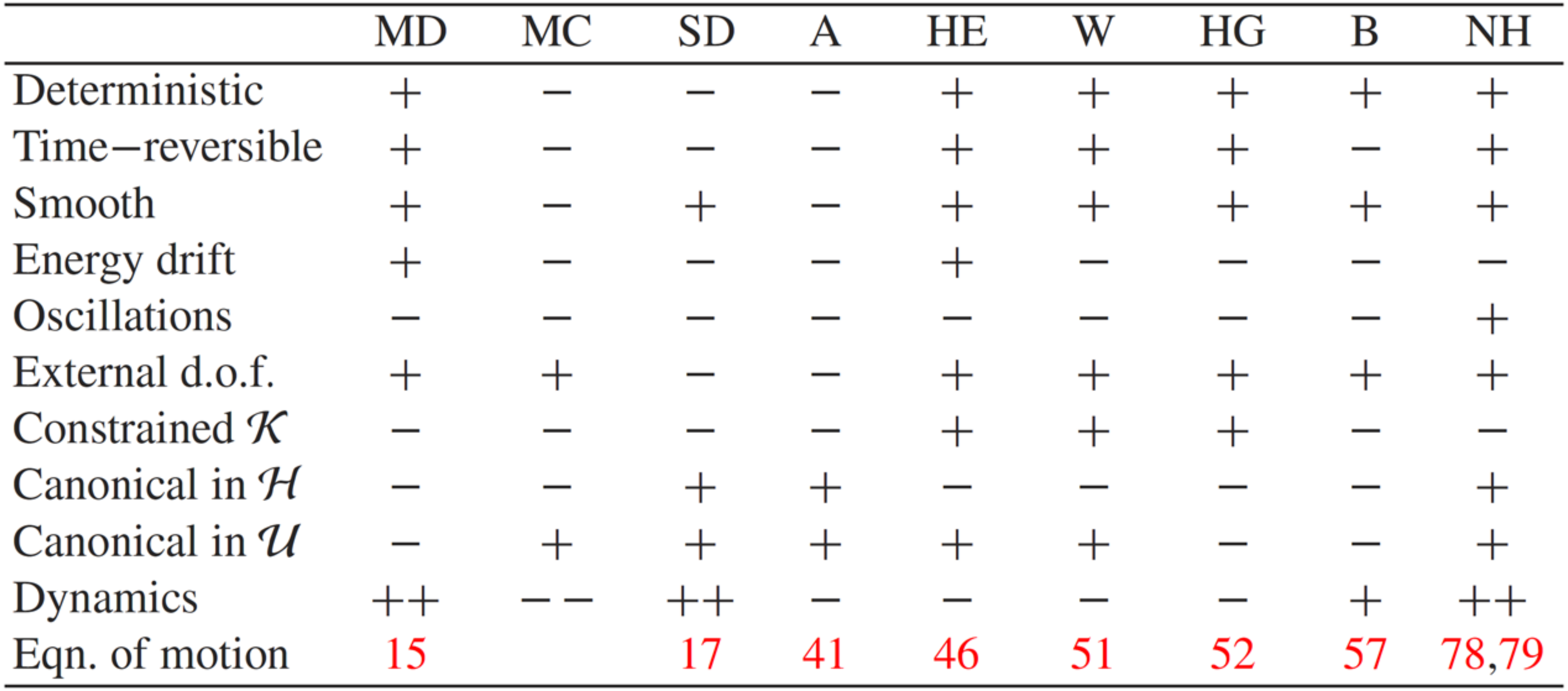
$$\langle \tilde{F}_{i\mu}\tilde{F}_{j\nu}\rangle = 2m_i\gamma_i k_{\mathrm{B}} T_0 \delta_{ij}\delta(t'-t)$$

- Nose-Hoover

$$\tilde{H} = \sum_i \frac{\tilde{\mathbf{p}}_i^2}{2m_i\tilde{s}^2} + U(\tilde{\mathbf{X}}) + \frac{\tilde{p}_s}{2Q} + (N_{\mathrm{df}}-1)\,k_{\mathrm{B}}T_0\log(\tilde{s})$$

P. Huenenberger, Adv. Polym. Sci **173** (2005).

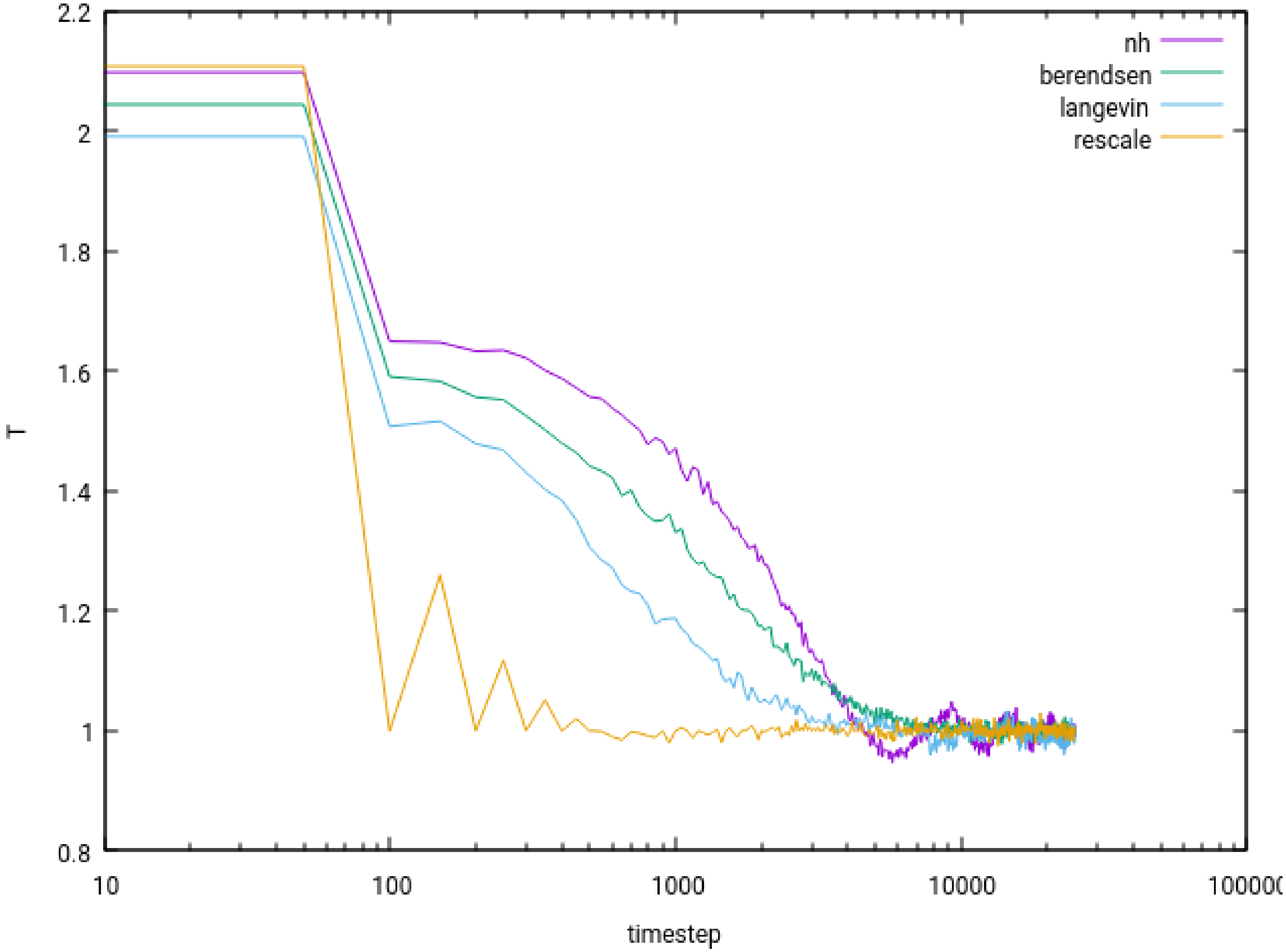
COMPARISON OF THERMOSTATTING ALGORITHMS



	MD	MC	SD	A	HE	W	HG	B	NH
Deterministic	+	−	−	−	+	+	+	+	+
Time−reversible	+	−	−	−	+	+	+	−	+
Smooth	+	−	+	−	+	+	+	+	+
Energy drift	+	−	−	−	+	−	−	−	−
Oscillations	−	−	−	−	−	−	−	−	+
External d.o.f.	+	+	−	−	+	+	+	+	+
Constrained $\mathcal{K}$	−	−	−	−	+	+	+	−	−
Canonical in $\mathcal{H}$	−	−	+	+	−	−	−	−	+
Canonical in $\mathcal{U}$	−	+	+	+	+	+	−	−	+
Dynamics	++	−−	++	−	−	−	−	+	++
Eqn. of motion	15		17	41	46	51	52	57	78,79

P. Huenenberger, Adv. Polym. Sci **173** (2005).

COMPARISON (INCLUDING NAÏVE TEMPERATURE SCALING)



SUM

- Molecular Dynamics is a powerful tool for obtaining information about molecular model systems.
- All-atom systems access tens of nanometers in length and hundreds of nanoseconds in time; this can be accelerated through coarse-graining methods. The use of advanced sampling algorithms can also make the most of the limited time (we’ll see how shortly).
- The MD simulation protocol requires you to
  - Choose a model to simulate
  - Build a system
  - Choose an ensemble
  - Choose an integrator [or, have one chosen for you]
  - Measure averages for the trajectory
- Next Up: Accessing larger length scales, time scales, and free energy computation!