

Computational Physics

Molecular Dynamics Simulations

E. Carlon, M. Laleman and S. Nomidis – Academic year 2015/2016

Contents

1	Introduction	3
2	Integration schemes	4
2.1	On the symplectic nature of the Velocity Verlet algorithm	6
3	Initialization	8
4	Interatomic Potentials	9
4.1	Lennard-Jones systems	9
4.2	Electrostatic interactions	9
4.3	Molecules	10
5	Measured quantities	10
5.1	Temperature	10
5.2	Diffusion coefficient	11
5.3	Pressure	11
6	Constraints	12
7	Simulations at constant temperature	14
7.1	Velocity rescaling	15
7.2	Berendsen thermostat	15
7.3	Andersen thermostat	15
7.4	Nosé-Hoover thermostat	16
8	Langevin dynamics	16
8.1	Numerical integration of the Langevin equation	18
9	Barrier-Crossing Problems	22
9.1	Smoluchowski Equation	22
9.2	Kramers' Theory	24

10 Introduction to LAMMPS	26
10.1 NVE Integration	27
10.2 Langevin Integration	28
10.3 LAMMPS files	29
11 Acknowledgements	32
12 References	32
13 Assignments	33
13.1 Halley's comet	33
13.2 Symplectic vs.non-symplectic integrators	33
13.3 Falling springs	35
13.4 Testing Kramers' theory	35
13.5 Lennard-Jones fluid	37
13.6 Langevin dynamics of Gaussian polymers	38

1 Introduction

The idea of Molecular Dynamics (MD) is that of numerically integrating the classical equations of motion to generate a trajectory of a system with N particles in time. The N particles are characterized by $6N$ coordinates, namely positions and momenta in a three dimensional volume space. Let a particle i be subject to a force \vec{f}_i at time t . Molecular Dynamics solves Newton's equations

$$\frac{d^2\vec{r}_i}{dt^2} = \frac{\vec{f}_i(\vec{r}_1, \vec{r}_2, \dots, \vec{v}_1, \vec{v}_2, \dots)}{m_i} \quad (1)$$

where the force depends on the particles positions and eventually velocities.

A typical Molecular Dynamics program follows this scheme:

1. At time $t = 0$ initialize the system by choosing positions and velocities of the particles. Choose also a value for Δt , the time step used in the numerical integration of the equations of motion.
2. Compute the forces on all particles.
3. Integrate the Newton's equations of motion using an appropriate integration scheme. Update positions and velocities of particles at time $t + \Delta t$.
4. Go back to step 2.

Molecular Dynamics is essentially a *deterministic* method, differently from Monte Carlo simulations which have a *stochastic* nature. Given an initial condition a molecular dynamics program will always generate the same trajectory in phase space. There are however versions of a MD algorithm with features. Examples are discussed in Sections 7.3 and 8.

In MD simulations physical quantities are calculated from time averages along the trajectories. Let A be a quantity depending on positions and velocities $A(t) = A(\vec{r}_1(t) \dots \vec{r}_N(t), \vec{v}_1(t), \dots \vec{v}_N(t))$. Its average value is defined as

$$\langle A \rangle = \frac{1}{N} \sum_{n=0}^{N-1} A(t_0 + n\Delta t) \quad (2)$$

where t_0 is an initial time. Typically, a physical quantity is characterized by some relaxation time τ_A , which is the time one has to wait until A reaches an average equilibrium value. If one is interested in equilibrium quantities it is then safer to take averages starting from a given time $t_0 > \tau_A$.

There are several integration schemes based on some discretization of the differential equation (1), as we will show in the next Section.

2 Integration schemes

The simplest integration scheme and also the mostly used is the so-called Verlet's algorithm. Let us consider the development of the positions up to the fourth order for $\vec{r}_i(t + \Delta t)$ and $\vec{r}_i(t - \Delta t)$:

$$\vec{r}_i(t + \Delta t) = \vec{r}_i(t) + \vec{v}_i(t)\Delta t + \frac{\vec{a}_i(t)}{2}\Delta t^2 + \frac{\vec{g}_i(t)}{6}\Delta t^3 + O(\Delta t^4) \quad (3)$$

$$\vec{r}_i(t - \Delta t) = \vec{r}_i(t) - \vec{v}_i(t)\Delta t + \frac{\vec{a}_i(t)}{2}\Delta t^2 - \frac{\vec{g}_i(t)}{6}\Delta t^3 + O(\Delta t^4) \quad (4)$$

where $\vec{v}_i = d\vec{r}_i/dt$ (velocity), $\vec{a}_i = d^2\vec{r}_i/dt^2$ (acceleration) and $\vec{g}_i = d^3\vec{r}_i/dt^3$.

By summing up the two equations one arrives at the

Verlet algorithm

$$\vec{r}_i(t + \Delta t) = 2\vec{r}_i(t) - \vec{r}_i(t - \Delta t) + \vec{a}_i(t)\Delta t^2 + O(\Delta t^4) \quad (5)$$

The positions are calculated accurately up to the fourth order in Δt . The velocities are not explicitly calculated, but they can be derived from the knowledge of the trajectory as

$$\vec{v}_i(t) = \frac{\vec{r}_i(t + \Delta t) - \vec{r}_i(t - \Delta t)}{2\Delta t} + O(\Delta t^2) \quad (6)$$

which is accurate only to the order Δt^2 . Hence, quantities depending on the velocities, as the total kinetic energies are not very accurately determined.

The position of the particles at time step $t + \Delta t$ depends on positions at times t and $t - \Delta t$, which gives the problem of initialization: we cannot directly apply Eq. (5) using initial conditions in which the $t = 0$ positions and velocities are given. The problem is solved in the following way. Given the initial positions and velocities $\vec{r}_i(0)$ and $\vec{v}_i(0)$ one approximates

$$\vec{r}_i(\Delta t) \approx \vec{r}_i(0) + \vec{v}_i(0)\Delta t + \frac{\vec{a}_i(0)}{2}\Delta t^2 \quad (7)$$

From it one can calculate the forces, hence $\vec{a}_i(\Delta t)$, and then apply iteratively Eq. (5).

An alternative and much better integration scheme, where positions and velocities are simultaneously updated, is the so-called

Velocity Verlet algorithm

$$\vec{r}_i(t + \Delta t) = \vec{r}_i(t) + \vec{v}_i(t)\Delta t + \frac{\vec{a}_i(t)}{2}\Delta t^2 + O(\Delta t^4) \quad (8)$$

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t) + \frac{\vec{a}_i(t + \Delta t) + \vec{a}_i(t)}{2}\Delta t \quad (9)$$

Although it may not be clear from Eqs.(8) and (9) it can be shown that in the velocity Verlet algorithm the positions are calculated accurately up to the fourth order in time. In order to show this let us replace $t \rightarrow t - \Delta t$ in Equation (8). We find:

$$\vec{r}_i(t - \Delta t) = \vec{r}_i(t) - \vec{v}_i(t - \Delta t)\Delta t - \frac{\vec{a}_i(t - \Delta t)}{2}\Delta t^2 \quad (10)$$

Summing the previous equation to Eq. (8) one gets

$$\begin{aligned} \vec{r}_i(t + \Delta t) &= 2\vec{r}_i(t) - \vec{r}_i(t - \Delta t) + (\vec{v}_i(t) - \vec{v}_i(t - \Delta t))\Delta t + \frac{\vec{a}_i(t) - \vec{a}_i(t - \Delta t)}{2}\Delta t^2 \\ &= 2\vec{r}_i(t) - \vec{r}_i(t - \Delta t) + \vec{a}_i(t)\Delta t^2 \end{aligned} \quad (11)$$

where we have used Eq. (9) at time $t - \Delta t$ to simplify the expression. Equation (11) is identical to the position update in the standard Verlet algorithm (Eq. (5)), hence of fourth order in Δt .

An important feature of the Verlet and velocity Verlet algorithms is that they are fully time reversal¹: Eq. (5) remains invariant by interchanging $\Delta t \rightarrow -\Delta t$. What is impractical of the velocity Verlet algorithm as given in Eqs. (8) and (9) is that it requires storage of accelerations at two different time steps. An implementation avoiding this is the following

¹There exist other Molecular Dynamics integration schemes which are not time reversal, or better they are time reversal only up to some given order in Δt .

Implementation of the Velocity Verlet algorithm

- 1) Given $\vec{r}_i(t)$, $\vec{v}_i(t)$ and $\vec{a}_i(t)$ update the positions to

$$\vec{r}_i(t + \Delta t) = \vec{r}_i(t) + \vec{v}_i(t)\Delta t + \frac{\vec{a}_i(t)}{2}\Delta t^2 \quad (12)$$

- 2) Calculate the velocities at an intermediate time step

$$\vec{v}_i(t + \Delta t/2) = \vec{v}_i(t) + \frac{1}{2}\vec{a}_i(t)\Delta t \quad (13)$$

- 3) From the positions at time $t + \Delta t$ calculate the forces and hence accelerations. If forces are derived from a (velocity independent potential) $V()$ one has

$$\vec{a}_i(t + \Delta t) = -\frac{1}{m}\nabla V_i(\vec{r}_1(t + \Delta t) \dots \vec{r}_N(t + \Delta t)) \quad (14)$$

- 4) Update the velocities to

$$\vec{v}_i(t + \Delta t) = \vec{v}_i(t + \Delta t/2) + \frac{1}{2}\vec{a}_i(t + \Delta t)\Delta t \quad (15)$$

Note that the algorithm is identical to that of Eqs. (8) and (9). The difference is that Eq. (9) is implemented in two steps. The new implementation requires less memory: there is no need to store data at two different time steps. For a system of N particles in three dimensions one needs $9N$ memory locations: $3N$ for positions, $3N$ for velocities and $3N$ for accelerations.

The timestep in a MD simulation should be chosen so that the simulations is long enough to be relevant to the time scales of the natural processes being studied. But Δt should be small enough to avoid discretization errors, for instance should be smaller than the fastest vibrational frequency of the system.

2.1 On the symplectic nature of the Velocity Verlet algorithm

A property of the velocity Verlet algorithm is that it is a *symplectic integrator*. Such algorithms have the property that their trajectories conserve exactly a “pseudo-energy” which differs from the true energy by a small amount (vanishing as $\Delta t \rightarrow 0$). This avoids long time drifts of the trajectories from the true ones, due to the accumulation of numerical errors during the time evolution. The problem 13.2 illustrate the importance of a symplectic algorithm.

To see this in some more general details we start from the Hamilton formalism of

classical mechanics which recasts the Newton's equations of motions as follows:

$$\begin{cases} \frac{d\vec{q}_i}{dt} = -\frac{\partial H}{\partial \vec{p}_i} \\ \frac{d\vec{p}_i}{dt} = \frac{\partial H}{\partial \vec{q}_i} \end{cases} \quad (16)$$

where \vec{q} and \vec{p} are the positions and momenta and the Hamiltonian H is given by:

$$H = \sum_{i=1}^N \frac{\vec{p}_i^2}{2m} + V(\vec{q}_1, \dots, \vec{q}_N) \quad (17)$$

and it is the total energy of the system. By plugging in (17) into (16) one finds back the Newton's equations (1).

Consider now a function of positions and momenta² $f(\vec{q}, \vec{p})$ and consider its time evolution along a trajectory $\vec{q}(t), \vec{p}(t)$ which is a solution of the equations of motion. We get

$$\frac{df(\vec{q}, \vec{p})}{dt} = \left[\dot{\vec{q}} \cdot \frac{\partial}{\partial \vec{q}} + \dot{\vec{p}} \cdot \frac{\partial}{\partial \vec{p}} \right] f(\vec{q}, \vec{p}) = \left[\frac{\vec{p}}{m} \cdot \frac{\partial}{\partial \vec{q}} + m\vec{a} \cdot \frac{\partial}{\partial \vec{p}} \right] f(\vec{q}, \vec{p}) \equiv iL f(\vec{q}, \vec{p}) \quad (18)$$

where iL defines the so-called Liouville operator. The formal solution of the previous equation of motion can be formally written as

$$f(\vec{q}(t), \vec{p}(t)) = e^{iLt} f(\vec{q}, \vec{p}) \quad (19)$$

It can be shown that the Liouville operator is Hermitian, which implies that the time evolution conserves the volume in the phase space. The Liouville operator is the sum of two parts, one containing derivatives with respect of momenta $iL_p \equiv m\vec{a} \cdot \partial/\partial \vec{p}$ and the other containing derivatives with respect to the positions $iL_q \equiv \vec{p}/m \cdot \partial/\partial \vec{q}$, however notice that

$$e^{i(L_p + L_q)t} \neq e^{iL_p t} e^{iL_q t} \quad (20)$$

since L_p and L_q do not commute. As we are interested in a discretization of the equations of motion in small time steps Δt , we can always decompose products of exponentials of non-commuting operators as follows:

$$e^{(A+B)\Delta t} = e^{A\Delta t} e^{B\Delta t} + \mathcal{O}(\Delta t^2) \quad (21)$$

$$e^{(A+B)\Delta t} = e^{A\Delta t/2} e^{B\Delta t} e^{A\Delta t/2} + \mathcal{O}(\Delta t^3) \quad (22)$$

The left and right hand sides of the two expression match at different orders in Δt and can be verified by expanding both sides in powers of Δt . We will show that Eq. (22) with $A = L_p$ and $B = L_q$ generates the velocity Verlet algorithm. We start from the relation:

$$e^{\frac{\Delta t p_x}{m} \frac{\partial}{\partial x}} f(\vec{q}(t), \vec{p}(t)) = \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{p_x \Delta t}{m} \right)^n \frac{\partial^n}{\partial x^n} f(\vec{q}(t), \vec{p}(t)) = f\left(\vec{q}(t) + \frac{p_x \Delta t}{m} \hat{x}, \vec{p}(t)\right) \quad (23)$$

²To keep the notation simpler we omit the index i for the particles in the following, but the equations generalize simply to the case of N particles

where \hat{x} is the unity vector in the x -direction. Since $\partial/\partial x$, $\partial/\partial y$ and $\partial/\partial z$ commute we can generalize the previous equation to get

$$e^{iL_q\Delta t}f(\vec{q}(t),\vec{p}(t)) = f\left(\vec{q}(t) + \frac{\Delta t}{m}\vec{p}(t),\vec{p}(t)\right) \quad (24)$$

Using this relation and the analogous one for L_p one gets

$$e^{iL_p\Delta t/2}e^{iL_q\Delta t}e^{iL_p\Delta t/2}f(\vec{q}(t),\vec{p}(t)) = f\left(\vec{q}(t) + \frac{\vec{p}(t)}{m}\Delta t + \vec{a}(t)\frac{\Delta t^2}{2}, \vec{p}(t) + m\frac{\vec{a}(t) + \vec{a}(t + \Delta t)}{2}\Delta t\right) \quad (25)$$

which is exactly the velocity Verlet algorithm³.

Note that in general one can write:

$$e^{iL_p\Delta t/2}e^{iL_q\Delta t}e^{iL_p\Delta t/2} = e^{iL\Delta t} + \varepsilon \quad (26)$$

with ε an operator which can be expressed in terms of the commutators of L_p and L_q and it is of some higher order in Δt . Such ε can be constructed by expanding Eq. (26) in powers of Δt . This means that we can define a pseudo-Liouville operator

$$iL_{\text{pseudo}} \equiv iL + \varepsilon/\Delta t \quad (27)$$

which corresponds to an exactly conserved pseudo-Hamiltonian.

3 Initialization

A MD simulation begins with giving a set of initial positions and velocities to the particles ($\vec{r}_1(0), \vec{r}_2(0), \dots, \vec{v}_1(0), \vec{v}_2(0), \dots$). The positions can be initialized by putting for instance the particles in some lattice points. The velocities can be selected from some distribution. For instance every velocity component could be taken uniformly from the interval $[-0.5v_0, 0.5v_0]$, where v_0 is some velocity scale. One imposes the condition that the total momentum vanishes. So if the initial selection yields

$$\vec{P}_{\text{TOT}} = \sum_{i=1}^N m_i \vec{v}_i(0) \neq 0 \quad (28)$$

we can subtract from all momenta a component

$$\vec{v}_i(0) \rightarrow \vec{v}_i(0) - \frac{\vec{P}_{\text{TOT}}}{m_i N} \quad (29)$$

One important test that one needs to perform to check the stability of MD simulation is the conservation of the total energy. The energy is a sum of kinetic K and potential V energies ($E = K + V$). Both K and V fluctuate during the simulation, but their sum should be constant⁴.

³Note that the lower order decomposition (21) generates a symplectic Euler integrator which is discussed in the assignment (13.2) for a one-dimensional harmonic oscillator, see Eq. (117)

⁴Actually only approximately constant, because of discretization errors accumulating during the run, due to a finite Δt

4 Interatomic Potentials

4.1 Lennard-Jones systems

A widely studied system is the Lennard-Jones fluid, which is defined by the potential

$$V_{\text{LJ}}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (30)$$

The potential has a short range repulsive term $\sim 1/r^{12}$ and a long range attractive term $\sim 1/r^6$. The origin of attractive $\sim 1/r^6$ force is quantum-mechanical and due to fluctuating dipoles (van der Waals interactions). The choice of a power $\sim 1/r^{12}$ at short distances has no theoretical justification. Its origin is related to the Pauli principle: when the electronic clouds surrounding the atoms start to overlap, the energy of the system increases abruptly. The exponent $\sim 1/r^{12}$ was chosen exclusively because of ease of computation (it is the square of the attractive term). The Lennard-Jones potential approximates very well interaction in rare gases as Ar or Kr whose interactions are dominated by van der Waals forces and electrons are in closed shells⁵. The Lennard-Jones potential has also played a central role in the development of molecular dynamics methods and has been widely used to investigate fundamental issues as phase transitions, the effect of surfaces, small clusters etc.

Usually infinitely ranged potentials are truncated to a finite radius R_c so that one approximates the LJ potential with:

$$\hat{V}_{\text{LJ}}(r) = \begin{cases} V_{\text{LJ}}(r) - V_{\text{LJ}}(R_c) & \text{if } r \leq R_c \\ 0 & \text{if } r > R_c \end{cases} \quad (31)$$

The term $V_{\text{LJ}}(R_c)$ is a small constant shift to avoid jumps in the potential which would lead to infinite forces.

4.2 Electrostatic interactions

If electrostatic charges are present, one should consider the appropriate Coulomb potentials

$$V_{\text{Coulomb}}(r) = \frac{Q_1 Q_2}{4\pi\epsilon_0 r} \quad (32)$$

where Q_1 , Q_2 are the charges and ϵ_0 the permittivity in free space. Note that the Coulomb potential has a much slower decay ($\sim 1/r$) at large distances compared to the Lennard-Jones case ($\sim 1/r^6$). Truncations as that given in Eq. (31) are not applicable to the Coulomb potential case. However, methods like the Ewald summation scheme (for more details see [1]) have been devised to deal with the long range forces arising from Coulomb potentials.

⁵Note that in lighter noble gases as He and Ne quantum mechanical effects are usually dominant close to their liquid-gas transition point. The molecular dynamics methods we discussed here apply exclusively to classical systems. To know whether quantum mechanical effects are of importance in a system one can compare the de Broglie wavelength $\Lambda = h/\sqrt{2\pi m k_B T}$ (h is the Planck's constant) to the typical interatomic distance a . Quantum effects dominate if $\Lambda \gg a$, i.e. for light atoms and molecules and at low temperatures.

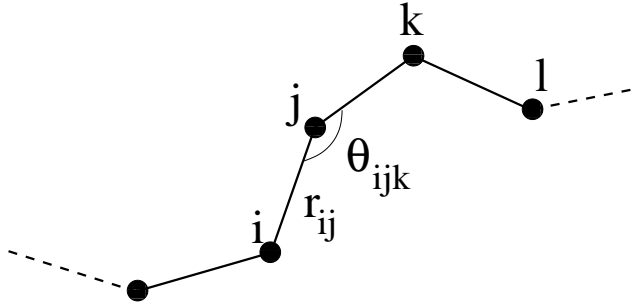


Figure 1: Schematic representation of a linear molecule.

4.3 Molecules

In molecular systems neighboring atoms form molecular bonds which are rather stable and are not disrupted easily. The intramolecular bonding interaction is usually described by simple terms of the following kind:

$$V_{bond}(r) = \frac{1}{2} \sum_{bonds} k_{ij} (r_{ij} - r_0)^2 + \frac{1}{2} \sum_{angles} k_{ijk} (\theta_{ijk} - \theta_0)^2 + \dots \quad (33)$$

where ijk denote neighboring atoms, r_{ij} the distance between the two bonded atoms i and j , and θ_{ijk} the angle formed between the three consecutive atoms i , j and k (see Fig. 1). r_0 and θ_0 denote the equilibrium value assumed by these quantities, while k_{ij} and k_{ijk} the associated “spring” constants. Typically these quantities are obtained from experimental data, or from first principles quantum mechanical calculations. The potential describes small oscillations of the various parameters around their equilibrium values. In addition there are usually other terms in Eq. (33) which involve four neighboring atoms. Atoms which are not bonded (as the atoms i and l in Fig. 1) can still interact via Lennard-Jones, Coulomb or other types of interactions.

5 Measured quantities

5.1 Temperature

For a system in equilibrium at some given temperature T , the theorem of equipartition states that the average kinetic energy per particle is related to T as

$$\frac{m}{2} \langle v^2 \rangle = \frac{3}{2} k_B T \quad (34)$$

One can then compute the temperature in a MD simulation from the total kinetic energy of the system using the relation⁶

$$T_K = \frac{1}{3Nk_B} \sum_i m_i \vec{v}_i^2 \quad (35)$$

⁶We use the subscript K to stress that this temperature is defined from the Kinetic energy K of the system

Note that as the kinetic energy fluctuates in time, the previous relation defines a so-called “instantaneous temperature” which is also a time fluctuating quantity. Usually if one simulates a sufficiently large number of particles T_K (after some equilibration time) converges to a roughly constant values as fluctuations for large N are small⁷.

5.2 Diffusion coefficient

The diffusion coefficient can be computed from the mean squared displacement of the particles as

$$D = \lim_{t \rightarrow \infty} \frac{\langle [\vec{r}(t) - \vec{r}(0)]^2 \rangle}{6t} \quad (36)$$

Alternatively one can use the Green-Kubo relations which relates it to the integral of the velocity autocorrelation function:

$$D = \frac{1}{3} \lim_{t \rightarrow \infty} \int_0^t \langle \vec{v}(\tau) \cdot \vec{v}(0) \rangle d\tau \quad (37)$$

5.3 Pressure

Usually pressure is calculated from the virial theorem

$$PV = NK_b T + \frac{1}{3} \sum_{i=1}^N \langle \vec{r}_i \cdot \vec{F}_i \rangle \quad (38)$$

To prove this relation one starts from computing the time average of

$$W = \sum_{i=1}^N \vec{r}_i \cdot \vec{F}_i^{TOT} \quad (39)$$

where \vec{F}_i^{TOT} is the total force acting on the i -th particle. Using the Newton's equation ($\vec{F}_i^{TOT} = m_i \ddot{\vec{r}}_i$) and an integration by parts one finds

$$\langle W \rangle = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t \sum_{i=1}^N \vec{r}_i \cdot \vec{F}_i^{TOT} d\tau = \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t \sum_{i=1}^N \vec{r}_i \cdot m_i \ddot{\vec{r}}_i = - \lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t \sum_{i=1}^N m_i \left| \dot{\vec{r}}_i \right|^2 d\tau \quad (40)$$

So that the average of W is equal to twice the total kinetic energy of the system, hence using the equipartition theorem

$$\langle W \rangle = -3Nk_B T \quad (41)$$

Note that the total force has two components:

$$\vec{F}_i^{TOT} = \vec{F}_i + \vec{F}_i^{EXT} \quad (42)$$

⁷Fluctuations of T_K in equilibrium are discussed in Section 7

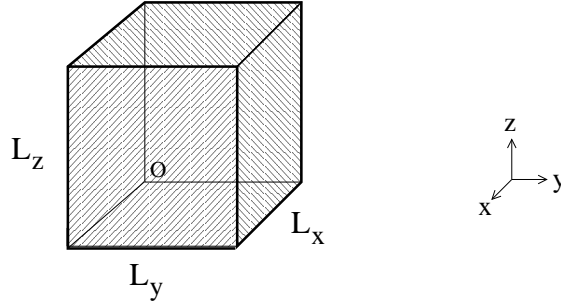


Figure 2: The contribution of the average of the left hand side of Eq. (43) equals to $-PV$ for each of the three dashed sides in the figure.

the first component, \vec{F}_i , is due to the interaction with all other particles in the system, the second component, \vec{F}_i^{EXT} , arises from interactions with the walls of the container.

We prove next that

$$\left\langle \sum_{i=1}^N \vec{r}_i \cdot \vec{F}_i^{EXT} \right\rangle = -3PV \quad (43)$$

where P is the pressure and V the volume of the container.

To show this consider a container of the form of a rectangular parallelepiped with sides of lengths L_x , L_y and L_z . Setting the origin of the axes in the edge as indicated in Fig. 2 one finds that the average in Eq. (43) gets three contribution from the three dashed sides of the container. The force is perpendicular to the sides. For the side perpendicular to the x -axis the average force is PL_yL_z (recall the pressure equals the force per unit area), whereas the position of the particles hitting the side is $\vec{r}_i = (L_x, y_i, z_i)$. The side perpendicular to the x -axis contributes to the average of Eq. (43) with a term $-PL_yL_zL_x = -PV$. The factor 3 in Eq. (43) comes from the contributions from the two other sides perpendicular to the y - and z -axes.

Combining Eqs. (39), (41), (42) and (43) we obtain the virial theorem of Eq. (38).

If the interparticle forces arise from a pair potential $V(r)$ where r is the distance between the pairs, then the virial theorem can be written as follows:

$$PV = Nk_B T - \frac{1}{3} \sum_{i < j} \left\langle r_{ij} \frac{dV(r_{ij})}{dr_{ij}} \right\rangle \quad (44)$$

where the sum is extended to all pairs of particles. The proof is left to the reader.

6 Constraints

In simulations of molecular systems one has multiple timescales in the problem: short timescales are due, for instance, to vibrations of bond lengths around their equilibrium values, long timescales involve the dynamics of the whole molecule. It is customary to take the bonds as being constrained to have fixed length. In this way one ignores short timescales and focuses on slower dynamics phenomena. Ignoring short timescales

means also that one can take longer time steps Δt , which leads to a longer time sampling of the physical quantities and thus more efficient simulations. In classical mechanics, constraints are introduced through the Lagrangian or Hamiltonian formalisms. The constraint imposes an additional force to the system⁸. A constraint of fixed bond length b between atoms 1 and 2 may be written as follows

$$\sigma(\vec{r}_1, \vec{r}_2) = (\vec{r}_1 - \vec{r}_2)^2 - b^2 = 0 \quad (45)$$

and the associated force to the i -th particle is $\vec{G}_i = -\lambda \nabla_i \sigma$, where λ is a so-called Lagrange multiplier. Its value can be fixed using Lagrangian formalism of classical mechanics. Here we show how λ can be fixed in the context of the integration scheme of an MD simulation. For simplicity we restrict ourselves to a particle constrained to remain at a fixed distance d from the origin, with constraint:

$$\sigma(\vec{r}) = \vec{r}^2 - d^2 \quad (46)$$

For simplicity, we restrict ourselves to the Verlet algorithm, although it is possible to extend the discussion of constraint to the velocity Verlet or other integration schemes. We assume that the positions at time steps $t - \Delta t$ and t satisfy the constraint thus $\sigma(\vec{r}(t - \Delta t)) = \sigma(\vec{r}(t)) = 0$. The Verlet algorithm gives:

$$\vec{r}(t + \Delta t) = 2\vec{r}(t) - \vec{r}(t - \Delta t) - \lambda \vec{\nabla} \sigma(t) \Delta t^2 = \vec{r}_u(t + \Delta t) - \frac{\lambda}{m} \vec{r}(t) \Delta t^2 \quad (47)$$

where we have introduced the unconstrained evolution $\vec{r}_u(t + \Delta t) \equiv 2\vec{r}(t) - \vec{r}(t - \Delta t)$. The Lagrange multiplier λ is fixed from the requirement that the position at time $t + \Delta t$ satisfy the constraints:

$$\sigma(\vec{r}(t + \Delta t)) = \vec{r}_u^2(t + \Delta t) - 2\frac{\lambda \Delta t^2}{m} \vec{r}(t) \cdot \vec{r}_u(t + \Delta t) + \frac{\lambda^2 \Delta t^4}{m^2} \vec{r}^2(t) - d^2 = 0 \quad (48)$$

which we obtained by squaring the right hand side of Eq. (47). As the constraint is satisfied at times t then $\vec{r}^2(t) = d^2$. We then obtain from (48) the following second order equation for λ :

$$\lambda^2 \frac{d^2 \Delta t^4}{m^2} - 2\lambda \frac{\Delta t^2}{m} \vec{r}(t) \cdot \vec{r}_u(t + \Delta t) + \vec{r}_u^2(t + \Delta t) - d^2 = 0 \quad (49)$$

We plug then the value of λ which is solution of this equation into Eq. (47), so that the constraint is exactly satisfied. This is however a rather simple case of a single particle with one constraint. In the most general case we will have N particles and l constraints the Verlet algorithm will take the form:

$$\vec{r}_i(t + \Delta t) = \vec{r}_{u,i}(t + \Delta t) - \frac{\Delta t^2}{m} \sum_{k=1}^l \lambda_k \vec{\nabla}_i \sigma_k(t) \quad (50)$$

⁸A known example is the centripetal force.

where $\vec{r}_{u,i}$ is the unconstrained position, where σ_k ($k = 1, 2, \dots, l$) denote the constraints⁹ and λ_k are the Lagrange multiplier which remain to be fixed. Following the same reasoning as done above for a single constraint we would obtain a set of coupled quadratic equations in the λ 's. Differently from the single constraint case there is no analytical solution. In the general case, several different methods have been devised to fix the values of the Lagrange multipliers in an approximate way. The discussion of these methods goes beyond the purposes of this course.

7 Simulations at constant temperature

The MD simulations discussed so far are performed in the so-called microcanonical ensemble: the number of particles (N), volume (V) and energy (E) are fixed. For this reason it is customary to indicate the microcanonical as the NVE ensemble. This setup is quite different from experimental conditions in which the temperature is fixed.

When the temperature is fixed the total energy is not conserved anymore, and in equilibrium it is distributed according to

$$p(E) = \frac{e^{-E/k_B T}}{Z} \quad (51)$$

where k_B is the Boltzmann's constant and Z is the partition function

$$Z = \int d\Gamma e^{-E/k_B T} \quad (52)$$

where the integral is extended to all position and momenta of the system ($d\Gamma = dr_1 dr_2 \dots dr_N dp_1 \dots dp_N$, with N the number of particles in the system). Considering that E is a sum of kinetic and potential energies

$$E = K + V = \sum_{i=1}^N \frac{p_i^2}{2m} + V(r_1, r_2 \dots r_N) \quad (53)$$

If we integrate out over all positions of the particles in Eq. (51) we find that the probability distribution of the velocity of a single particle is

$$p(\vec{v}) \propto \exp\left(-\frac{m\vec{v}^2}{2k_B T}\right) \quad (54)$$

where we used $p = mv$. In Section 5.1 we have introduced the instantaneous temperature T_K from the average total kinetic energy. A simple calculation shows that the relative variance of T_K in the canonical ensemble is given by [1]

$$\frac{\sigma_{T_K}^2}{\langle T_K \rangle^2} \equiv \frac{\langle T_K^2 \rangle - \langle T_K \rangle^2}{\langle T_K \rangle^2} = \frac{2}{3N} \quad (55)$$

⁹We write $\sigma_k(t)$ as a shorthand notation for $\sigma_k(\vec{r}_1(t), \dots, \vec{r}_N(t))$. $\vec{\nabla}_i$ is the gradient of σ_k with respect to the position of the i -th particles. For constraints on bond lengths σ involves only pairs of particles.

where the averages are taken according to the canonical ensemble. The instantaneous temperature defined through Eq. (35) in the canonical NVT ensemble fluctuates.

In the next sections we discuss about different ways of “fixing” the temperature in a MD simulation. These methods are referred to as *thermostats*. Only one of the methods presented (the Nosé-Hoover thermostat discussed in Sec. 7.4) reproduces the fluctuations of the canonical ensemble.

7.1 Velocity rescaling

The simplest thermostat consists in a rescaling of all the velocities after each integration step. If we obtain from an integration scheme as the velocity Verlet $\vec{v}_i(t)$ and $\vec{r}_i(t)$ we rescale all velocities by a factor λ as follows:

$$\vec{v}_i'(t) = \lambda \vec{v}_i(t) \quad (56)$$

while the positions are unaffected. We proceed with the integration of the equations of motion at $t + \Delta t$ using these new velocities. The factor λ is obtained from

$$\lambda = \sqrt{\frac{T_0}{T_K(t)}} \quad (57)$$

where $T_K(t)$ is the instantaneous temperature defined by Eq. (35) and T_0 is the desired temperature. The velocity rescaling fixes $T_K(t)$ to a constant value T_0 at each integration step, therefore it does not reproduce the fluctuations expected from the canonical ensemble (Eq. (55)).

7.2 Berendsen thermostat

The Berendsen thermostat is also based on some velocity rescaling. However the rescaling is “softer”, compared to that of Eq. (57). The rescaling factor is given by:

$$\lambda^2 = 1 + \frac{\Delta t}{\tau} \frac{T_0 - T_K(t)}{T_K(t)} \quad (58)$$

where τ is a characteristic time to be fixed at the beginning of the simulation. Note that the smallest allowed value $\tau = \Delta t$ yields the velocity rescaling method of Section 7.1. If $\tau \rightarrow \infty$ then no rescaling is done as $\lambda = 1$. For finite and not too small τ the rescaling is $\lambda = 1 + \mathcal{O}(\Delta t)$. If $T_K(t) > T_0$ then $\lambda < 1$, whereas if $T_K(t) < T_0$ then $\lambda > 1$ so that the velocities are increased or decreased according to the value of the instantaneous temperature. The system reaches the desired temperature T_0 only after a characteristic time of the order τ . As the method of Section 7.1, the Berendsen thermostat does not reproduce the canonical distribution of energies.

7.3 Andersen thermostat

In the Andersen thermostat some particles undergo random “collisions” with external degrees of freedom. When they do so their velocity is changed and taken from the

Gaussian distribution of Eq. (54). If the collisions are sufficiently frequent this procedure induces a canonical distribution of the velocities. In practice the thermostat is implemented as follows. First a collision frequency ν is chosen. All velocities and positions are updated according to the desired scheme (for instance the velocity Verlet). One then runs through all the N particles of the system and select for each of them a random number uniformly distributed in $[0, 1[$. Let $0 \leq r_i < 1$ the random number corresponding to the i -th particle. If $r_i < \nu\Delta t$ the velocity of the i -th particle is replaced from a value selected from the distribution of Eq. (54). Because of the random update to the velocities, the Andersen thermostat is an example of a MD algorithm in which the deterministic evolution is combined with a stochastic update.

7.4 Nosé-Hoover thermostat

In this scheme the system is brought in contact with a “thermal bath”. The system plus bath form a closed universe, so their total energy is conserved. The system however exchanges energy with the bath. The net result of this exchange is that the energy of the system is distributed according to the canonical distribution.

Differently from the Andersen thermostat the Nosé-Hoover thermostat is deterministic. Time evolution is described by the following equations of motion:

$$\ddot{\vec{r}}_i = \frac{\vec{F}_i}{m_i} - \gamma \dot{\vec{r}}_i \quad (59)$$

$$\dot{\gamma} = \frac{1}{Q} \left(\sum_i \frac{p_i^2}{m_i} - 3Nk_B T_0 \right) \quad (60)$$

Equation (59) is the Newton equation with an additional friction term with a time-dependent “friction” coefficient equal to γ , whose time evolution is described by Eq. (60). The friction coefficient γ can be either positive or negative in the course of time, for this reason it cannot be considered as a real friction coefficient. If γ is positive energy is taken from the system, whereas if γ is negative energy is given to the system. The quantity between parenthesis becomes twice the kinetic energy of the system minus $3Nk_B T_0$. The form of Equation (60) is such that γ tends to a value such that the quantity between parenthesis vanishes so that the kinetic energy is forced to be close to $\frac{3}{2}Nk_B T_0$. Fluctuations around this value are however possible. It can be rigorously shown that the Nosé-Hoover thermostat reproduces the canonical fluctuation exactly [1]. The Equations (59) and (60) follow from an extended Hamiltonian in which system and bath degrees of freedom are coupled.

8 Langevin dynamics

The Langevin equation is a stochastic differential equation which describes a system coupled to fast degrees of freedom, where the latter are not explicitly taken into account, but implicitly by means of a random force. The typical example is a molecule immersed in a solvent (like water), where the molecule has much larger mass compared

the solvent molecules and therefore moves much faster. The solvent molecules are in constant motion due to thermal agitation, they collide with the large particle and exchange energy with it. If we are not interested in the motion of the solvent molecules (fast degrees of freedom), but only in the large particle (slow degrees of freedom), we may assume that the solvent molecules exert some random forces on the latter. Collisions occur at a characteristic time $\tau_{coll.}$. The Langevin equation description applies at times $t \gg \tau_{coll.}$, when a very large number of collisions have occurred.

In the one dimensional case the Langevin equation takes the form:

$$m\dot{v} = F - m\gamma v + f_R \quad (61)$$

where m is the particle mass, v its velocity, F is a deterministic force applied to the particle, γ the friction coefficient and f_R a random force. In absence of friction and random forces ($\gamma = 0$ and $f_R = 0$) Eq. (61) reduces to the Newton's equation of motion. The random force is assumed to have the following properties:

$$\langle f_R(t) \rangle = 0 \quad (62)$$

$$\langle f_R(t)f_R(t') \rangle = 2A\delta(t - t') \quad (63)$$

where $\langle \rangle$ denotes the average value and A is a constant to be determined. The δ -function indicates that the random forces at different times are uncorrelated, which should happen if $t \gg \tau_{coll.}$. The parameter A is related to the characteristic amplitude of the random force.

Consider now the case $F = 0$. The equation (61) can be integrated exactly. Given v_0 the initial velocity we find:

$$v(t) = v_0 e^{-\gamma t} + \frac{1}{m} e^{-\gamma t} \int_0^t e^{\gamma s} f_R(s) ds \quad (64)$$

Which implies that the average velocity vanishes asymptotically as

$$\langle v(t) \rangle = v_0 e^{-\gamma t} \quad (65)$$

where $\tau = 1/\gamma$ is a characteristic relaxation time. One can compute the fluctuation around the average:

$$\left\langle (v(t) - v_0 e^{-\gamma t})^2 \right\rangle = \frac{A}{m^2 \gamma} (1 - e^{-2\gamma t}) \quad (66)$$

At times $t \gg 1/\gamma$ the system relaxes to equilibrium and the equipartition theorem¹⁰ holds:

$$m \langle v(t)^2 \rangle = k_B T \quad (67)$$

which implies:

$$A = m\gamma k_B T \quad (68)$$

¹⁰The equipartition theorem states that the average kinetic energy per particle in a three dimensional system in equilibrium is $\varepsilon_{kin} = 3k_B T/2$. In one dimension (the case considered here) $\varepsilon_{kin} = k_B T/2$.

We can integrate equation (64) further to get the position

$$x(t) = x_0 + \int_0^t v(s) ds \quad (69)$$

to obtain the mean square displacement

$$\langle (x(t) - x_0)^2 \rangle = \int_0^t ds ds' \langle v(s) v(s') \rangle \quad (70)$$

For the velocity-velocity correlator we find from (64):

$$\langle v(s) v(s') \rangle = \left(v_0^2 - \frac{A}{m^2 \gamma} \right) e^{-\gamma(s+s')} + \frac{A}{m^2 \gamma} e^{-\gamma|s-s'|} \quad (71)$$

which substituted into Eq. (70) gives

$$\langle (x(t) - x_0)^2 \rangle = \left(v_0^2 - \frac{A}{m^2 \gamma} \right) \left(\frac{1 - e^{-\gamma t}}{\gamma} \right)^2 + \frac{2A}{m^2 \gamma} \left(t - \frac{1 - e^{-\gamma t}}{\gamma} \right) \quad (72)$$

This expression has two interesting limiting behaviors. At short times $t \ll 1/\gamma$ one finds:

$$\langle (x(t) - x_0)^2 \rangle \approx v_0^2 t^2 \quad (73)$$

which is the ballistic motion, i.e. the particle moves with a constant speed v_0 . At much longer times $t \gg 1/\gamma$ one has

$$\langle (x(t) - x_0)^2 \rangle \approx \frac{2A}{m^2 \gamma^2} t \quad (74)$$

which describes a diffusive behavior, with the mean squared displacement increasing linearly with time. The prefactor is $2D$ with D the diffusion constant which then takes the form:

$$D = \frac{k_B T}{m \gamma} \quad (75)$$

which is known as the Einstein relation. Having summarized the basic properties of the Langevin equation we present next an algorithm which is used for its numerical integration.

8.1 Numerical integration of the Langevin equation

We show now how the Langevin equation can be discretized to implement it in molecular dynamics simulations. From the analysis of the previous section we have fixed the amplitude A and we can write the equation in the following form

$$m \ddot{r}(t) = F(r(t)) - m \gamma \dot{r}(t) + \sqrt{2m\gamma k_B T} \eta(t) \quad (76)$$

with $\langle \eta(t) \eta(t') \rangle = \delta(t' - t)$. Although $\eta(t)$ is not differentiable, we can define integrals of these processes, and therefore, it is useful to introduce a process $w(t)$, known as a

Wiener process, such that $\eta(t) = dw/dt$. We can derive some properties of $w(t)$ if we assume a small time interval Δt .

One of the main properties of the Wiener process is the following:

$$\langle w(s)w(s') \rangle = \int_0^s \int_0^{s'} \langle \eta(t)\eta(t') \rangle dt dt' = \int_0^s \int_0^{s'} \delta(t' - t) dt dt' = \min(s, s') \quad (77)$$

(to prove the last step one can consider separately the two cases $s > s'$ and $s' > s$.) Using the previous relation we can show that

$$\langle (w(t + \Delta t) - w(t))^2 \rangle = \langle w^2(t + \Delta t) + w^2(t) - 2w(t + \Delta t)w(t) \rangle = \Delta t \quad (78)$$

We can generate the random variable $w(t + \Delta t) - w(t)$ using at any time step a gaussian distributed random number ξ with zero mean and unit variance (i.e. $\langle \xi \rangle = 0$ and $\langle \xi^2 \rangle = 1$) as follows:

$$w(t + \Delta t) - w(t) = \sqrt{\Delta t} \xi \quad (79)$$

From this relation we can easily derive a first order integrator of the Langevin equation (76) in the overdamped limit (i.e. the high friction limit in which the inertial term $m\ddot{r}$ is neglected). The first order discretization of the derivatives gives:

$$m\gamma (r(t + \Delta t) - r(t)) = F(r(t))\Delta t + \sqrt{2mk_B T} [w(t + \Delta t) - w(t)] \quad (80)$$

and using (79) we get the

First order integrator of the overdamped Langevin equation

$$r(t + \Delta t) = r(t) + \frac{F(r(t))}{m\gamma} \Delta t + \sqrt{\frac{2k_B T}{m\gamma}} \xi \sqrt{\Delta t} \quad (81)$$

In practice at any time step one needs to generate a random number ξ with zero mean and unit variance¹¹. The peculiar feature of this integrator is the presence of a square root of the time step Δt . This factor is due to the property (79), which has a clear physical origin. The Wiener process is the integral of the random noise $w(\tau) = \int_0^\tau \eta(t)dt$. We can view $\eta(t)$ as the steps of a random walk and $w(\tau)$ is then the sum of all increments i.e. the total distance covered by the walker after a time τ . Due to the diffusive nature of random walks this distance scales as $\sqrt{\tau}$.

Having discussed the main features of the integrator in the simple overdamped case we are ready to present a more detailed integrator which takes the inertial term into account and it is accurate to order Δt^2 . The algorithm has been derived by Vanden-Eijnden and Ciccotti [4]. We express the Langevin equation in the following differential form:

$$dr(t) = v(t)dt \quad (82)$$

$$dv(t) = f(r(t))dt - \gamma v(t)dt + \sigma dw(t) \quad (83)$$

¹¹In Matlab you can use the command `randn(N,1)` to generate a vector with N random numbers selected from a gaussian distribution with zero mean and unit variance.

where $\sigma = \sqrt{2k_B T \gamma / m}$ and $f(r) = F(r)/m$ and $w(t)$ is the Wiener process. This expression can be easily expanded to a system of n particles in three dimensions. Here, for simplicity, we focus on a single particle in one dimension.

We start by integrating Eqs. (82) and (83) from t to $t + \Delta t$:

$$r(t + \Delta t) = r(t) + \int_t^{t+\Delta t} v(s) ds \quad (84)$$

$$v(t + \Delta t) = v(t) + \int_t^{t+\Delta t} f(r(s)) ds - \gamma \int_t^{t+\Delta t} v(s) ds + \sigma[w(t + \Delta t) - w(t)] \quad (85)$$

Note that these contains integrals in v and f which produce terms of the order Δt and higher in the discretization. We start from the integral in the velocity for which Eq. (85) can be used to lowest order:

$$v(s) \approx v(t) + (s - t)f(r(t)) - (s - t)\gamma v(t) + \sigma[w(s) - w(t)] \quad (86)$$

where $t \leq s \leq t + \Delta t$. Integrating the previous result from t to $t + \Delta t$ we get:

$$\int_t^{t+\Delta t} v(s) ds = \Delta t v(t) + C(t) \quad (87)$$

where we have defined

$$C(t) \equiv \frac{\Delta t^2}{2} [f(r(t)) - \gamma v(t)] + \sigma \int_t^{t+\Delta t} [w(s) - w(t)] ds \quad (88)$$

This expression can be used as approximant of the integrals of the velocity on the Eqs. (84) and (85).

To approximate the time integral of the force appearing in equation (85) we proceed in a similar manner. Using the relation

$$\frac{df}{dt} = \frac{\partial f}{\partial r} \dot{r} = \frac{\partial f}{\partial r} v \quad (89)$$

we obtain:

$$f(r(s)) = f(r(t)) + \int_t^s v(y) f'(r(y)) dy \approx f(r(t)) + (s - t)v(t)f'(r(t)) \quad (90)$$

Again we integrate this equation from t to $t + \Delta t$ and get

$$\int_t^{t+\Delta t} f(r(s)) ds = \Delta t f(r(t)) + \frac{\Delta t^2}{2} v(t) f'(r(t)) = \Delta t \frac{f(r(t + \Delta t)) + f(r(t))}{2} \quad (91)$$

where the last step is performed to eliminate the inconvenient derivative of the force using the following relation:

$$f(r(t + \Delta t)) = f(r(t)) + f'(r(t))(r(t + \Delta t) - r(t)) = f(r(t)) + f'(r(t))v(t)\Delta t \quad (92)$$

For the computation of the term $w(t + \Delta t) - w(t)$ in Eq. (85) we use again (79). In order to end the calculation we still need to compute $C(t)$ in Eq. (88), which contains integral of the difference between two Wiener functions. We note that using (77) one can easily prove that:

$$\left\langle (w(t + \Delta t) - w(t)) \int_t^{t+\Delta t} ds' (w(s') - w(t)) \right\rangle = \frac{\Delta t^2}{2} \quad (93)$$

$$\left\langle \int_t^{t+\Delta t} ds (w(s) - w(t)) \int_t^{t+\Delta t} ds' (w(s') - w(t)) \right\rangle = \frac{\Delta t^3}{3} \quad (94)$$

which can be viewed as higher order counterparts of Eq. (78). Using the same strategy as for the first order integrator we see that Eq. (94) can be fulfilled if:

$$\int_t^{t+\Delta t} ds (w(s) - w(t)) = \Delta t^{3/2} \eta \quad (95)$$

with η a random number with the properties $\langle \eta \rangle = 0$ and $\langle \eta^2 \rangle = 1/3$. Plugging in (95) and (79) in Eq. (93) we find $\langle \xi \eta \rangle = \frac{1}{2}$, hence the two random numbers are correlated. To solve this issue in practice we can decompose η into two uncorrelated random numbers as follows:

$$\eta = a\xi + b\theta \quad (96)$$

with $\langle \xi \theta \rangle = 0$ and $\langle \theta^2 \rangle = 1$. The coefficients a and b can be determined by imposing $\langle \xi \eta \rangle = \frac{1}{2}$ and $\langle \eta^2 \rangle = 1/3$, from which we get $a = 1/3$ and $b = 1/(2\sqrt{3})$. This implies:

$$\int_t^{t+\Delta t} ds (w(s) - w(t)) = \Delta t^{3/2} \left(\frac{\xi}{\sqrt{3}} + \frac{\theta}{2\sqrt{3}} \right) \quad (97)$$

Now plugging in (87), (91), (79) and (97) in (84) and (85) we get the:

Second order integrator of the Langevin equation

$$\begin{aligned} r(t + \Delta t) &= r(t) + \Delta t v(t) + C(t) \\ v(t + \Delta t) &= v(t) + \frac{\Delta t}{2} [f(r(t + \Delta t)) + f(r(t))] - \Delta t \gamma v(t) \\ &\quad + \sigma \sqrt{\Delta t} \xi(t) - \gamma C(t) \\ C(t) &= \frac{\Delta t^2}{2} [f(r(t)) - \gamma v(t)] + \sigma \Delta t^{3/2} \left(\frac{1}{2} \xi(t) + \frac{1}{2\sqrt{3}} \theta(t) \right) \end{aligned} \quad (98)$$

Note: The integrator reduces to the Velocity Verlet integrator if $\gamma = 0$ and $\sigma = 0$.

We are going to use this algorithm in problem 13.4 to investigate the dynamics of a particle crossing a potential barrier (see Fig. 3). The next section discusses the theory of the stochastic dynamics of barrier crossing.

9 Barrier-Crossing Problems

The transition of a system from one state into another can be generally viewed as a barrier-crossing process. In this approach, it is very common to reduce the complexity of the system, usually having many degrees of freedom, into a single parameter x called *reaction coordinate*. An appropriate choice of x should allow for a clear distinction between the two states, i.e. one should be able to know the state of the system, just by looking at the value of x .

In this section, we will calculate the rate at which a system will be crossing the energetic barrier, i.e. it will make the transition between two states. For this purpose, we will first present a phenomenological derivation of the Smoluchowski equation, which we will use to calculate the rate.

9.1 Smoluchowski Equation

Suppose we have a large ensemble of identical and non-interacting systems, placed within a solvent, which are described by a probability density function $p(x, t)$. By definition, if we choose randomly a system from that ensemble at time t , then $p(x, t)dx$ will express the probability of it having its reaction coordinate in the interval $[x, x + dx]$.¹² In absence of external forces, p will evolve in time according to the diffusion equation

$$\frac{\partial p}{\partial t} = D \frac{\partial^2 p}{\partial x^2}$$

where D is the so-called *diffusion coefficient*. Moreover, p must satisfy the continuity equation at all times

$$\frac{\partial p}{\partial t} + \frac{\partial j}{\partial x} = 0 \quad (99)$$

where $j = j(x, t)$ is the probability current, i.e. the rate at which the probability “flows” through x at time t .¹³ By comparing those two expressions, we identify

$$j_{\text{diffusion}} = -D \frac{\partial p}{\partial x} \quad (100)$$

as the diffusion current, i.e. the current that tends to balance any inhomogeneities in p .

If we now introduce a potential energy¹⁴ $V(x)$, a general force $F = -dV/dx$ will act on every system, depending on its reaction coordinate x . This will give rise to an additional probability current of the form

$$j_{\text{drift}} = p v_{\text{drift}} \quad (101)$$

¹²In order to make the notation simple, we will use $p(x, t) = p$ from now on, keeping in mind that it will, in general, be a function of both x and t .

¹³Although this may seem rather abstract, it has a more intuitive interpretation. More specifically, if one is studying a very large number N of identical and non-interacting systems, $Nj(x, t)$ will represent how many systems per unit time pass through x , i.e. they go from $x - dx$ to $x + dx$.

¹⁴Note here that, owing to the reduction of the degrees of freedom into a single reaction coordinate, $V(x)$ will not simply be the mechanical energy of the system, but it will represent a generalised energy, which may also contain entropic contributions etc.

where v_{drift} is the “velocity” of the probability density flow, due to the force.

As already mentioned, we are assuming that every system is placed within a solvent, which has a friction coefficient γ . In the previous section, we discussed the effect of such a solvent, for the case of a single particle. In particular, we saw that it leads to two additional terms in the equation of motion

$$m \frac{dv}{dt} = F - m\gamma v + f_R$$

where $-m\gamma v$ represents the friction force that the particle feels as it moves with velocity v through the solvent and $f_R = f_R(t)$ is the random force that mimics the collisions with the solvent particles.

In analogy with that, the drift velocity v_{drift} of the probability density will obey a similar equation

$$\mu \frac{dv_{\text{drift}}}{dt} = F - \gamma v_{\text{drift}}$$

where μ is a generic mass,¹⁵ which we have incorporated in γ . We also used the fact that the force f_R , having random orientation, has no contribution to the net drift velocity of the probability density.¹⁶ If we now take the friction coefficient to be very large, which is the so-called *overdamped limit*, it can be shown that the acceleration term becomes negligible, so that the velocity is simply proportional to the force

$$v_{\text{drift}} = \frac{F}{\gamma}$$

With this approximation, we can now rewrite the expression (101) for the drift current in the following manner

$$j_{\text{drift}} = -\frac{1}{\gamma} \frac{dV}{dx} p \quad (102)$$

Using the expressions (100) and (102) we have found, we can now write the net probability current $j = j_{\text{diffusion}} + j_{\text{drift}}$ as follows

$$j = -D \frac{\partial p}{\partial x} - \frac{1}{\gamma} \frac{dV}{dx} p \quad (103)$$

An important property can be derived from this result. More specifically, if we take the case of an equilibrium distribution, for which we know that $p = \exp(-V/k_B T)/Z$, the current (103) vanishes (see continuity equation (99)) and we obtain the *Einstein relation*

$$D = \frac{k_B T}{\gamma} = \frac{1}{\gamma \beta} \quad (104)$$

¹⁵Apart from the total mass of the system, μ will also depend on the nature of the reduction we make, when describing the whole system with a single reaction coordinate x .

¹⁶Note, however, that it is this force which gives rise to the diffusion current of equation (100). The fact that the solvent is responsible for both $j_{\text{diffusion}}$ and j_{drift} will become more apparent when we derive the Einstein relation (104), which shows that the diffusion constant is connected to the friction coefficient.

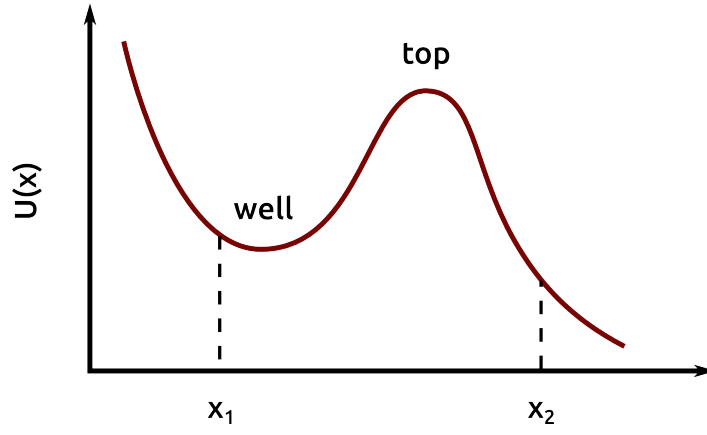


Figure 3: Energy landscape of a typical barrier-crossing problem. We assume that the system is initially inside a potential well (W) and, in order to reach a lower-energy state and thus minimise its energy, it first needs to cross an energy barrier (T). With x_1 and x_2 we denote two reference points, which we will use in order to calculate the barrier-crossing rate.

where we have introduced the inverse temperature $\beta \equiv 1/k_B T$. We can now plug this in equation (103) and rewrite it in the following compact way

$$j = -D \left(\frac{\partial p}{\partial x} + \beta \frac{dV}{dx} p \right) = -D e^{-\beta V} \frac{\partial}{\partial x} (p e^{\beta V}) \quad (105)$$

Finally, we can use this expression, in order to rewrite the continuity equation (99) and obtain the so-called

Smoluchowski equation

$$\frac{\partial p(x, t)}{\partial t} = -\frac{\partial}{\partial x} \left[-D e^{-\beta V(x)} \frac{\partial}{\partial x} (p(x, t) e^{\beta V(x)}) \right] \quad (106)$$

which gives the time evolution of the probability density.

9.2 Kramers' Theory

We will now present a simplified version of Kramers' theory, valid in the overdamped limit. Suppose that the energy $V(x)$ of the system, as a function of the reaction coordinate, has the general form of Figure 3. We consider a large ensemble of identical and non-interacting systems, which are all initially placed at the energetic well on the left. We would like to calculate the average rate at which a crossing of the barrier takes place.

Since we are only interested in the transitions that take place in one direction, i.e. from left to right, we can take periodic boundary conditions. More specifically, we will

“collect” any system that reaches the right part of the barrier (e.g. beyond x_2) and will “place” it again inside the well. This way, we can sustain a steady flow of probability from left to right, which is translated to a constant current, both in time and space. Thus, we can start from the expression (105) for the current

$$j = -De^{-\beta V} \frac{\partial}{\partial x} (pe^{\beta V})$$

and integrate it, with respect to x , bearing in mind that j is a constant

$$\frac{j}{D} \int_{x_1}^{x_2} e^{\beta V(x)} dx = p(x_1)e^{\beta V(x_1)} - p(x_2)e^{\beta V(x_2)}$$

where x_1 is somewhere within the energetic well and x_2 is at the right of the barrier, as shown in Figure 3.

In order to proceed, we will need to make some approximations. More specifically, we will assume that *the height of the barrier is large enough*, so that the probability of finding a system in the vicinity of x_1 , i.e. within the well, is much higher than finding it around x_2 . Thus, we can neglect the second term in the right-hand side of the previous expression,¹⁷ so that we will have

$$\frac{j}{D} \int_{x_1}^{x_2} e^{\beta V(x)} dx \simeq p(x_1)e^{\beta V(x_1)} \quad (107)$$

Moreover, we can approximate the integral in the left-hand side as follows

$$\int_{x_1}^{x_2} e^{\beta V(x)} dx \simeq \int_{\text{top}} e^{\beta V(x)} dx$$

since $e^{\beta V(x)}$ is much larger at the top of the barrier, compared to its sides. Note also that, owing to the large height of the barrier, every system will have enough time to equilibrate¹⁸ within the well, before it crosses the barrier. In other words, we can approximate

$$p(x_1) \simeq \frac{1}{Z_{\text{well}}} e^{-\beta V(x_1)}, \quad \text{where } Z_{\text{well}} = \int_{\text{well}} e^{-\beta V(x)} dx$$

Plugging the above results in equation (107), we obtain

$$\frac{j}{D} \simeq \left[\int_{\text{well}} e^{-\beta V(x)} dx \int_{\text{top}} e^{\beta V(x)} dx \right]^{-1} \quad (108)$$

We now need to calculate the integrals appearing in equation (108). For this purpose, let us first expand the potential energy $V(x)$ around the bottom of the well x_w

$$V(x) \simeq V_w + \frac{\omega_w^2}{2} (x - x_w)^2$$

¹⁷This is also supported by the fact that $e^{\beta V(x_1)} \gg e^{\beta V(x_2)}$.

¹⁸A system that is placed within the deep energetic well has no means to “know” that there are more states available beyond the barrier, without crossing it. If the barrier is high enough, this will require long times, which will enable the system to reach a *local equilibrium*.

where we have introduced $V_w \equiv V(x_w)$ the energy at the bottom of the well and $\omega_w^2 \equiv V''(x_w)$ its curvature. Using this, we can rewrite the first integral in (108) as follows

$$\int_{\text{well}} e^{-\beta V(x)} dx \simeq e^{-\beta V_w} \int_{\text{well}} \exp \left[-\beta \frac{\omega_w^2}{2} (x - x_w)^2 \right] dx$$

Moreover, since the integrand decays exponentially around x_w , we can expand the limits of the integration to $-\infty$ and $+\infty$ without large errors. This way, carrying out the integration¹⁹ yields the following result

$$\int_{\text{well}} e^{-\beta V(x)} dx \simeq \sqrt{\frac{2\pi}{\beta \omega_w^2}} e^{-\beta V_w}$$

Following a similar procedure for the second integral of equation (108), we find

$$\int_{\text{top}} e^{\beta V(x)} dx \simeq \sqrt{\frac{2\pi}{\beta \omega_t^2}} e^{\beta V_t}$$

where we defined $V_t \equiv V(x_t)$ the energy at the top x_t of the barrier and $\omega_t^2 = -V''(x_t)$ its curvature. Finally, we can plug these results in equation (108) and use the Einstein relation (104), $D = 1/\gamma\beta$, so as to obtain the final result

$$\boxed{j \simeq \frac{\omega_w \omega_t}{2\pi\gamma} e^{-\beta \Delta V}} \quad (109)$$

where $\Delta V \equiv V_t - V_w$ is the height of the energetic barrier.

10 Introduction to LAMMPS

Molecular dynamics simulations can also be performed using special packages. Within these packages one typically finds preimplemented interactions and integrators. Instead of having to explicitly write the code for the system of interest, you can, in a simple way, specify the system and the integrator. The package then takes care of combining the correct, lower-level parts of the code and performs the simulation. In this section the focus will be lie on the freely available Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS).²⁰ Extensive information and a manual can be found on <http://lammps.sandia.gov>.

LAMMPS typically uses four kind of files:

¹⁹For this purpose, one needs to make a change of variable and use the expression for the Gaussian integral

$$\int_{-\infty}^{+\infty} e^{-x^2} dx = \sqrt{\pi}$$

²⁰S. Plimpton, **Fast Parallel Algorithms for Short-Range Molecular Dynamics**, J Comp Phys, 117, 1-19 (1995)

- The *input file*. The input file specifies the details of the system and the integrator. E.g. what kind of interactions are there, what are the numeric constants used in these interactions, what kind of integration should be used, ...
- The *configuration file*. The configuration file contains the details of the initial configuration.

Note that it is not always necessary to have this file present. Using the input file it is possible to generate initial configurations, thus eliminating the need of a configuration file. (See e.g. problem 13.5.)

- The *log file*. The log file keeps track of some useful quantities like energy and temperature and also contains the warnings and errors thrown during execution.
- The *output file*. This file contains snapshots of the system on times specified by the user during the simulations.

There are different types of output files and one can have more than one output file per simulation run.

The harmonic oscillator is an excellent example to illustrate the basic working of LAMMPS. In this section the NVE and Langevin integration of two particles connected by a Hookean spring will be presented. The spring constant k is set to 2 and the equilibrium length r_0 of the spring is 0. The initial extension of the spring is 1. The input and configuration file for this system can be found at the end of this section. They should be sufficiently self-explanatory.

10.1 NVE Integration

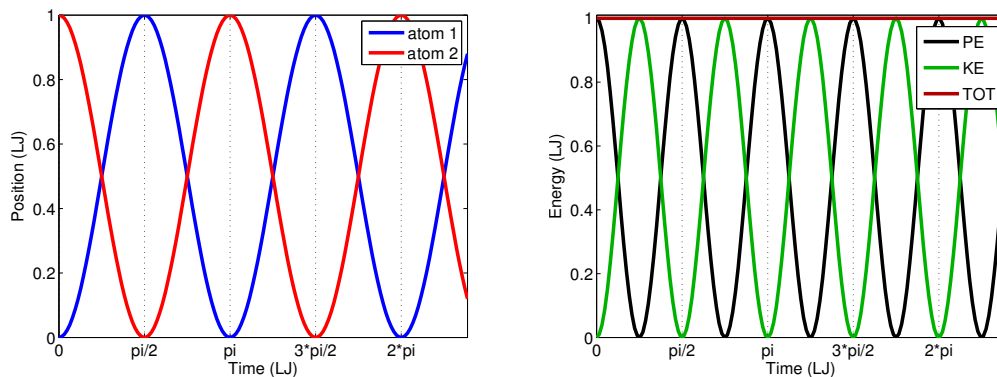


Figure 4: Left: The position of the two particles through time for NVE integration. Right: The energy of the whole system through time for NVE integration.

If we perform NVE integration of this system we know that the two particles start to oscillate. This is illustrated in 4. The period of the motion is 2π , as can be seen on the graph.

When looking at the energy, we see that it is indeed conserved, as expected for NVE integration.

10.2 Langevin Integration

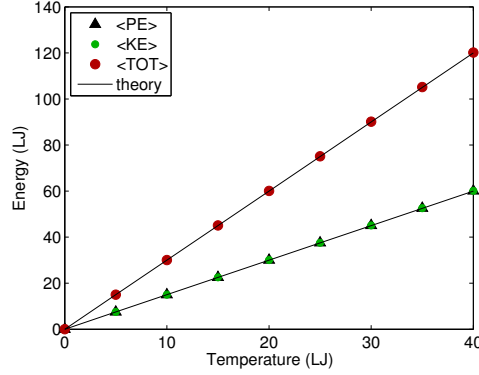


Figure 5: The average energy of the system during Langevin integration.

By turning on the Langevin integrator we are simulating a harmonic oscillator coupled to a heat bath. On 5 we see that the equipartition theorem is nicely satisfied:

$$\langle T \rangle + \langle V \rangle = \frac{3}{2}k_B T + \frac{3}{2}k_B T. \quad (110)$$

To obtain these results an average over many timesteps was taken. The first part of the simulation was discarded, since it was influenced by the initial conditions. I.e. the system was not thermalized yet.

Note: because in the code we impose *zero yes* on our Langevin integrator, we eliminate the degree of freedom the system has to move in space. This thus effectively means that we only look at the degrees of freedom of the system.

10.3 LAMMPS files

Input file

```
#####
#       This input file is used in order to simulate a harmonic       #
#       oscillator in nve and Langevin ensembles.                     #
#####

# This is the correct atom type for the case of an oscillator, having only a
# bond interactions (namely the spring between them).
atom_style      bond

# We have to turn off the sorting in order to avoid an error...
atom_modify      sort 0 0.

# Read in the initial data from the correct file.
read_data        data.harm

# We use a harmonic potential for the bond.
bond_style       harmonic

# The coefficients for the harmonic bond energy are:
# bond type (1 if no other bond types are present), k and r0.
bond_coeff        1 1. 0.

# Set the mass of all (*) atoms equal to 1.
mass * 1.0

# Set the velocity (vx vy vz) of all atoms equal to 0.
velocity all set 0.0 0.0 0.0

# Perform the integration through time.
fix nve_fix all nve

# Apply a Langevin thermostat to all atoms with initial and
# final temperature equal to 4, damping constant 100, a given seed
# and we want the net random force to be zero.
# The '154' in this expression is a seed for the RNG used by the Langevin
# integrator. The 'zero yes' command ensures that the net force in the total
# system is zero.
# UNCOMMENT for Langevin integration
#fix lan_fix all langevin 4 4 100 154 zero yes

# Let LAMMPS calculate the total kinetic energy of the system
# and store it in the compute variable c_Kenergy
compute Kenergy all ke

# Let LAMMPS calculate the total potential energy of the system
# and store it in the compute variable c_Venergy
compute Venergy all pe

# Store the energies (kinetic, potential and total) in three
# variables which can be accessed by the print command. Don't
# set E equal $K+$V, because LAMMPS will complain...
variable K equal c_Kenergy
variable V equal c_Venergy
variable E equal c_Kenergy+c_Venergy

# Store the x-position of the two particles in two variables,
# which can be used by the print command.
variable x1 equal x[1]
variable x2 equal x[2]

# Variable containing the timestep.
variable dt equal 0.005

# Variable containing the current time of the simulation,
# i.e. timestep (internally stored as step) times dt.
variable t equal step*dt

# Set the timestep for integration.
timestep          ${dt}

# The 'thermo' command allows you to view some basic information in the
# terminal. Every 50 integration steps this information is printed. The same
# information will also be present in the log file of the simulation.
thermo            50

# The "print" fix prints a certain string to a file at specified intervals.
# This string can contain references to variables, of which the value can
# be obtained using the dollar sign.

# print for NVE
fix energy_print all print 10 "$t $K $V $E" file energy.dat screen no title "#Timestep Kinetic Potential Total"
fix position_print all print 10 "$t ${x1} ${x2}" file position.dat screen no title "#Timestep x1 x2"

# print for Langevin
```

```
#fix energy_print all print 100 "$t $K $V $E" file energy.dat screen no title "#Timestep Kinetic Potential Total"

# The 'run' command starts the time integration over the specified number
# of timesteps.
run      1500    #NVE
#run 1500000 #Langevin
```

Note: if this file is named in.harm, it can be run with LAMMPS using the command “lammmps_daily -in in.harm”.

Note: the configuration file, which you can find below, should be named data.harm for the code to work.

Configuration file

```
#####
#           This data file is used in order to simulate a harmonic           #
#           oscillator in nve and Langevin ensembles.                         #
#####

# Specify the number of atoms. In this example two atoms make up the two ends
# of the oscillator.
2 atoms
# Specify the number of atom types. Our atoms will be identical, so there is
# only one atom type.
1 atom types
# Specify the number of bonds. There is only one bond in the system: the spring
# between the two atoms.
1 bonds
# Specify the number of bond types. There is only one bond, so we define only
# one bond type.
1 bond types

# ----- #

# Set the boundaries of the box. The commands xlo and xhi are preceeded with the
# lower bound for x and the upper bound for x respectively. An analogous thing
# goes for ylo, yhi, zlo and zhi.
-10 10 xlo xhi
-10 10 ylo yhi
-10 10 zlo zhi

# ----- #

# Specify some information about the particles. In the in.harm file we will have
# to specify the kind of atoms we want to use. In this example the type 'bond'
# is used. The documentation [http://lammps.sandia.gov/doc/read\_data.html]
# then tells us that we have to specify the following:
# atom-ID molecule-ID atom-type x y z
# Here x, y and z are the initial positions of the atom.
#
# For the bonds we have to specify:
# ID type atom1 atom2
# Here atom1 and atom2 are the bonds between which the bond is formed.
#
# IMPORTANT: there can be no comments between sections after this!!

Atoms

1 1 1 0. 0. 0.
2 1 1 1. 0. 0.

Bonds

1 1 1 2
```

11 Acknowledgements

We are very grateful to Hans Vandebroek from U. Hasselt who took care of explicitly deriving the calculations of the Langevin integrator of Section 8.1 and for typing part of that Section.

12 References

- [1] D. Frenkel and B. Smit, *Understanding molecular simulations*, Academic Press (2002)
- [2] F. Ercolessi, *A molecular dynamics primer*, International School for advanced studies SISSA/ISAS, Trieste (1997) - available from: <http://www.fisica.uniud.it/~ercolessi/md/md.pdf>
- [3] M.P. Allen, *Introduction to Molecular Dynamics simulations*, in “Computational Soft Matter: From Synthetic Polymers to Proteins”, Lecture Notes, N. Attig, K. Binder, H. Grubmüller, K. Kremer (Eds.), John von Neumann Institute for Computing, Jlich, NIC Series, Vol. 23, ISBN 3-00-012641-4, pp. 1-28 (2004).
- [4] E. Vanden-Eijnden and G. Ciccotti, *Second-order integrators for Langevin equations with holonomic constraints*, Chem. Phys. Lett. 429, 310316 (2006).

13 Assignments

13.1 Halley's comet

We simulate here the trajectory of Halley's comet around the Sun by using the velocity Verlet algorithm

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{a(t)}{2}\Delta t^2 \quad (111)$$

$$v(t + \Delta t) = v(t) + [a(t) + a(t + \Delta t)] \frac{\Delta t}{2} \quad (112)$$

Consider the Sun as a fixed point with mass $M = 2 \cdot 10^{30}\text{kg}$. The gravitational acceleration is given by

$$a = \frac{GM}{r^2} \quad (113)$$

with $G = 6.67428 \cdot 10^{-11} \frac{\text{m}^3}{\text{kg} \cdot \text{s}^2}$. Suppose that the comet starts at its aphelion²¹ at $5.28 \cdot 10^{12}$ meter distance from the Sun, with a purely tangential speed of $9.13 \cdot 10^2$ m/s.

- Generalize the velocity Verlet algorithm to two dimensions.
- Use this algorithm to simulate the comet's trajectory (use the implementation of Eqs. (12)-(15)). Decide upon a reasonable value of Δt by using the fact that a single orbit takes many years.
- Plot the x and y component of the trajectory in time. Plot the entire trajectory in the x-y plane.
- Verify that the total (kinetic+potential) energy is conserved.
- What is the distance of the perihelion to the Sun? What is the speed there? How long does a single orbit take?

13.2 Symplectic vs.non-symplectic integrators

We compare here the performance of two MD integration schemes for a one dimensional harmonic oscillator. The Hamilton's equations of motion are:

$$\begin{cases} \dot{p} = -\frac{\partial H}{\partial x} = -kx \\ \dot{x} = \frac{\partial H}{\partial p} = p/m \end{cases} \quad (114)$$

²¹The farthest point in its orbit. The speed is minimal here, and perpendicular to the axis Sun-comet.

where the Hamiltonian is given by

$$H = \frac{p^2}{2m} + \frac{kx^2}{2} \quad (115)$$

We set for convenience $k = m = 1$.

The first integrator (Euler integrator) we consider is non-symplectic and it is obtained from the first order expansion in Δt of the equations of motion

$$\begin{cases} p(t + \Delta t) = p(t) - x(t)\Delta t \\ x(t + \Delta t) = x(t) + p(t)\Delta t \end{cases} \quad (116)$$

The second integrator is also first order, but symplectic:

$$\begin{cases} p(t + \Delta t) = p(t) - x(t)\Delta t \\ x(t + \Delta t) = x(t) + p(t + \Delta t)\Delta t \end{cases} \quad (117)$$

Note that there is only a slight difference between the two.

a) Express Eqs. (116) and (117) in matrix form ²²

$$\begin{pmatrix} x \\ p \end{pmatrix}_{t+\Delta t} = M \begin{pmatrix} x \\ p \end{pmatrix}_t \quad (118)$$

and show that for Eq. (116) $\det(M) > 1$ while for Eq. (117) $\det(M) = 1$.

Symplectic integrators do not strictly conserve the Hamiltonian H , but they conserve a so-called *shadow*-Hamiltonian H' , which differs from H by terms of order Δt^k . For this reason they do not suffer from long time drift as non-symplectic integrators. Symplectic integrators are also volume preserving. The velocity Verlet scheme is a symplectic integrator (and it is of order $k = 2$). Eq. (117) defines the so-called Euler symplectic integrator of order $k = 1$.

b) Show analytically that the symplectic integrator defined by Eq. (117) has the following constant of motion (shadow hamiltonian)

$$H' = H - \frac{px}{2}\Delta t \quad (119)$$

c) Take $\Delta t = 10^{-3}$ and $\Delta t = 10^{-2}$ with initial conditions $x(0) = 1$, $p(0) = 0$. and plot the solutions of Eqs. (116) and (117) up to time $T = 10$. Compare the relative errors of the two integration schemes to the exact solution of Eq. (114). Which of the two integrators is more accurate?

d) For the same values of the parameters given above plot the value of H as a function of t up to time $T = 10$. Plot H' as well for the symplectic integrator. Which of the two integrators has a long time drift in the energy estimate?

²² M can be view as the Jacobian matrix of the transformation connecting position and momenta at time t to those at time $t + \Delta t$. The fact that $\det M = 1$ tells us that the transformation is volume preserving, i.e. it does not alter the volume of the two dimensional $x - p$ phase space.

13.3 Falling springs

We consider a spring connecting two identical masses falling freely from a given height to the ground. To make the problem simpler we consider the system to be one dimensional: the spring is elongated in the same direction of the gravitational force (the vertical z -axis). We set the masses to $m = 1$, the spring constant to $K = 1$ and spring rest length $l = 1$. The spring is at rest at the beginning, with the particles set at positions $z = 5$ and $z = 6$. The particles are subject to a gravitational force towards the ground $F = mg$. We set $g = 10$. The ground corresponds to $z = 0$ and the collision is totally elastic: after each collision the velocity of each particle changes sign. Neglect the mutual interaction between the particles.

- a) Simulate the system with the velocity Verlet algorithm, taking care of the boundary condition at $z = 0$ when updating the positions of the particles. Test the algorithm for different values of Δt .
- b) Plot the trajectories of the two particles $z_1(t)$ and $z_2(t)$ as a function of time. Do the particles ever reach the initial height?
- c) Plot the total elastic E_{el} , gravitational E_{grav} and kinetic E_{kin} energy as a function of time. Check also that the total mechanical energy is conserved²³ during the simulation.
- d) Repeat the analysis of the potential and kinetic energies for a chain of N particles of equal masses connected by identical springs. Take as starting positions $z_1 = 5$, $z_2 = 6 \dots z_N = 4 + N$ and consider $N = 10$ and $N = 20$.
- e) We would like to study the effect of different thermostats on the system. For this purpose, repeat the previous step using a velocity rescaling (57) and a Berendsen thermostat (58), taking $\tau = 100\Delta t$ for the latter. Turn on the thermostat at $t = 0.25 t_{\text{total}}$ and then turn it off again at $t = 0.75 t_{\text{total}}$, where t_{total} is the total time of the simulation. Choose a temperature of $T = 10$.

13.4 Testing Kramers' theory

In section 9.2 we derived an analytical expression for the rate at which a system can cross an energetic barrier in order to minimize its energy. In order to test this result, we are going to simulate the one-dimensional problem of a particle of mass $m = 1$, which feels the following potential

$$V(x) = -6\Delta V \left(\frac{x^3}{3} + \frac{x^2}{2} \right)$$

From Fig. 6 we see that this potential has a local minimum (well) at $x = -1$ next to a local maximum at $x = 0$ (barrier). This means that a particle initially placed at

²³We know that the energy in the Velocity Verlet algorithm is conserved up to some order in Δt . However if your simulation is correct you should see no drift of the total energy in time.

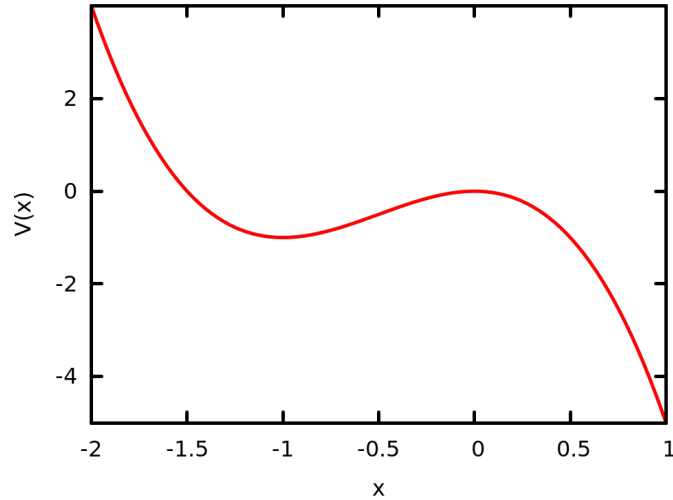


Figure 6: Energetic landscape of the barrier-crossing problem for $\Delta E = 1$. The energy has two local extrema, one at $x = -1$ (well) and one at $x = 0$ (top of barrier).

$x = -1$ will have to cross an energetic barrier of height ΔV in order to minimize its energy.²⁴

- a) If the energetic barrier is sufficiently high, the particle will practically remain fully confined within the potential well. This will provide it with enough time to equilibrate, so that the probability of finding it at x will follow the Boltzmann distribution $p(x) \sim e^{-V(x)/k_B T}$ within the well, while it will be $p(x) = 0$ beyond the barrier. To test this statement, place the particle at $x = -1$ with zero velocity, take $\Delta V = 10$ and run a sufficiently long simulation using the Langevin integrator (98) for $T = 0.25$ and $\gamma = 1$. One can then determine $p(x)$ by regularly collecting the position of the particle during the simulation and calculating the histogram at the end.
- b) We would like to verify the theory of section 9.2. For this purpose, simulate a system of many (at least 10^4) such non-interacting particles using the Langevin integrator for $\Delta V = 1$ and $\gamma = 1$. Place initially all particles at the bottom of the well with zero velocity. At every step of the simulation check which particles have crossed the barrier (e.g. satisfy the condition $x > 0.5$) and re-initialize them.²⁵ Vary the temperature around $T = 0.5$, plot the barrier-crossing rate and compare with the exponential dependence of Kramers' formula (109) by choosing the axes accordingly.

²⁴Note that in this case the reaction coordinate x is just the position of the particle.

²⁵Replenishing the particles in this way eventually leads to a constant current of barrier-crossing particles.

13.5 Lennard-Jones fluid

In this problem we are going to simulate a group of N particles of mass $m = 1$ that are confined to move within a cubic box of width $L = 10$. The particles interact with each other through a Lennard-Jones potential

$$V_{\text{LJ}}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]$$

where we choose $\sigma = \epsilon = 1$. For computational efficiency we are going to truncate this potential at $r = R_c = 2.5$ and shift it as follows

$$V(r) = \begin{cases} V_{\text{LJ}}(r) - V_{\text{LJ}}(R_c), & r \leq R_c \\ 0, & \text{else} \end{cases}$$

The appropriate pair-interaction style that implements this in LAMMPS is `lj/cut`. In order to shift the potential, you will have to use the command `pair_modify`.

The initialization of the simulation is performed as follows. First define the simulation box using the command `region`, create it using the command `create_box` and then generate N atoms randomly within the box with the command `create_atoms`. We would like to have periodic boundary conditions, but since this is the default choice in LAMMPS you will not need to specify it explicitly.

Note that there is a chance, especially for large N , that two or more particles are initially very close to each other. Owing to the Lennard-Jones repulsion term r^{-12} , this can lead to extremely high forces which may in turn lead to missing particles and LAMMPS giving an error. In order to avoid this, you can first perform an energy minimization using the command `minimize`.²⁶ Then you can initialize the velocities of the particles according to the Maxwell-Boltzmann distribution for an initial temperature T and set at the same time the total momentum equal to zero using the command `velocity` with the appropriate options.

- a) Run an NVE time integration²⁷ (`fix nve`) with time-step $\Delta t = 5 \times 10^{-3}$ and particle density $\rho = N/L^3 = 0.3$. Choose $T = 2$ as an initial temperature and plot the total energy as a function of time in order to verify that it is conserved. For the calculation of the kinetic and potential energy you will need to use the commands `compute ke` and `compute pe` respectively.
- b) Turn on a Nose-Hoover thermostat for $T = 2$, using an NVT time integration (`fix nvt`) instead of NVE, and repeat the simulation. Plot the kinetic energy as a function of time and compare with the expression $\langle K \rangle = 3Nk_B T/2$ obtained from the equipartition theorem.

²⁶Note that with this command LAMMPS does not perform an actual time-integration, but rather “artificially” drive the system into a state of minimum energy.

²⁷NVE means that the number of particles N , the volume V of the simulation box and the total energy E are kept constant during the integration.

- c) Repeat the previous simulation for different values of the particle density within the range $0 < \rho \leq 0.2$. Plot the pressure P of the system as a function of the density and compare with both the ideal gas law $P = \rho k_B T$ and its first correction $P = \rho k_B T(1 + b_2 \rho)$ coming from the virial expansion.²⁸ Do you observe a deviation for larger values of ρ ? Note that, in order to extract the pressure of the system using `compute pressure`, you will first need to obtain the current temperature with the command `compute temp`.
- d) Facultative: Calculate numerically the third virial coefficient

$$b_3 = -\frac{1}{3} \int d\vec{r} \int d\vec{r}' f(r) f(r') f(|\vec{r} - \vec{r}'|)$$

where $f(r) \equiv e^{-\beta V(r)} - 1$, and compare the previous result with the second correction $P = \rho k_B T(1 + b_2 \rho + b_3 \rho^2)$ to the ideal gas law.

13.6 Langevin dynamics of Gaussian polymers

A Gaussian polymer consists of a series of $N+1$ successive beads which are connected by N harmonic springs, similarly to what is shown in Fig. 7. The corresponding potential energy is

$$V_{\text{bond}} = \frac{3k_B T}{2a^2} \sum_{i=1}^N d_i^2$$

where d_i is the length of each spring, $k_B = 1$ is the Boltzmann's constant, T is the temperature and a is the equilibrium bond length.²⁹ For the implementation of this potential in LAMMPS one needs to use the `harmonic` bond-interaction style with the appropriate parameters.

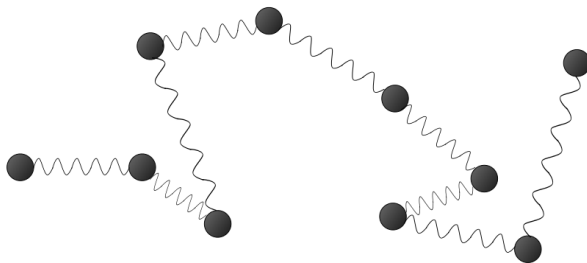


Figure 7: Bead-and-spring representation of a Gaussian polymer.

The student has to solve one of the following problems, which focus on various properties of such a polymer. All $N + 1$ beads have equal mass $m = 1$ and all N

²⁸The second virial coefficient b_2 is given by the formula

$$b_2 = -\frac{1}{2} \int d\vec{r} \left(e^{V(r)/k_B T} - 1 \right)$$

which you can calculate numerically. For $T = 2$ one finds $b_2 \simeq -0.772$.

²⁹Although the minimum energy of this potential is at $r_i = 0$, one can show that for finite temperature the equilibrium bond length is $r_i = a$.

springs have an equilibrium length of $a = 1$. The system follows Langevin dynamics, which can be implemented in LAMMPS using two successive fixes, namely the `nve` and `langevin` ones. For the latter the user has to specify a damping parameter, which is defined as the inverse of the friction coefficient γ appearing in the Langevin equation (61), which we will set here $\gamma = 1$. You will also need to turn off the sorting of atoms with the command `atom_modify sort 0 0.0`.³⁰

- a) Systems that follow Langevin dynamics typically perform diffusive motion. For instance, the Gaussian polymer is expected to undergo regular diffusion $\langle r_{\text{cm}}^2(t) - r_{\text{cm}}^2(0) \rangle \sim t$, where r_{cm} is its center of mass. However, if one focuses on a single bead of such a polymer, he or she will first observe an anomalous diffusion of the form $\langle r_i^2(t) - r_i^2(0) \rangle \sim \sqrt{t}$ at short time-scales, owing to the presence of the springs. At longer times the diffusion of the whole polymer will dominate, thus leading to an ordinary diffusion $\langle r_i^2(t) - r_i^2(0) \rangle \sim t$.

In order to verify the above statement, run simulations (about 10^7 time-steps for $\Delta t = 0.02$ and $\gamma = 1$) at $T = 1$ for a polymer consisting of $N = 200$ springs and plot the mean squared displacement of a bead as a function of time in a logarithmic scale, after letting the system equilibrate.

Hint: You can write a script that initializes the polymer as a random walk that performs jumps of length a , in order to reduce the equilibration times. Moreover, the mean squared displacement can be calculated and averaged over all beads with the compute `msd`. The current time-step of the simulation is automatically stored by LAMMPS in the variable `step`.

- b) One can make the system more realistic by introducing a resistance to bending. In order to simulate such a stiff polymer, one needs to introduce an angle energy term of the form³¹

$$V_{\text{bend}} = \frac{\kappa_b}{a} \sum_{i=1}^{N-1} [1 - \cos(\theta_i - \pi)]$$

where κ_b is the bending rigidity and θ the angle formed by three successive beads. This can be implemented in LAMMPS using the `cosine/delta` angle-interaction style with the appropriate parameters (angles in degrees).

Run simulations for $\kappa_b = 5$, $\Delta t = 0.01$, $N = 100$ and for different temperatures within the range $0.25 \leq T \leq 1.5$. Let the system equilibrate (about 10^5 steps) and then calculate the mean bending angle $\langle \cos(\theta - \pi) \rangle$. Plot it together with the theoretical expression

$$\langle \cos(\theta - \pi) \rangle = \frac{1}{\tanh(\kappa_b / ak_B T)} - \frac{ak_B T}{\kappa_b}$$

Hint: An easy way to get $\langle \cos(\theta - \pi) \rangle$ in LAMMPS is to first obtain the energy V_{bend} of every angle interaction with the `angle/local` compute, then average

³⁰You will need to do this whenever a simulation does not take pair interactions into account, otherwise an error will occur.

³¹Notice that the minimum energy corresponds to $\theta_i = \pi$, i.e. a completely straight configuration.

it along the polymer using the compute `reduce` and store it in a variable rearranging the terms appropriately.

- c) Simulate the stretching of a Gaussian polymer by applying two opposite forces F at the two ends. Set $T = 1$, $\Delta t = 0.02$ and $N = 100$, let the system equilibrate (roughly 10^5 steps) and plot the average squared extension $\langle z^2 \rangle$ as a function of the magnitude of the force within the range $0 \leq F \leq 3$. Compare with the analytical expression

$$\langle z^2 \rangle = Na^2 + \frac{N^2 a^2}{9} \left(\frac{Fa}{k_B T} \right)^2$$

Hint: In order to apply a force on a single particle in LAMMPS, one needs to define a group consisting of the particle and then use the fix `addforce` on it. Moreover, for the calculation of the extension you will need the coordinates of the first and last bead, which LAMMPS stores automatically in the variables `x[1]`, `x[101]` etc.