

MA3227 Numerical Analysis II

Lecture 21: Molecular Dynamics

Simon Etter



2019/2020

Molecular Dynamics

Disclaimer

Like Lectures 19 and 20, the main purpose of this lecture is not to introduce any new material but rather to illustrate the application of the techniques that we developed so far to real-world problems.

Introduction

It is well established today that matter consists of discrete units called atoms, and that the atomistic nature of matter has profound impacts on many everyday applications.

For illustration, watch 2:40 to 4:36 of the following video:

<https://youtu.be/7-I20Ru9BwM?t=162>.

The macroscopic effects of the microscopic composition of materials can be studied using the ODE techniques developed in Lectures 16 and 17.

For example, the following website shows computer simulations of cracks propagating through silicon crystals (essentially, rocks).

<https://warwick.ac.uk/fac/sci/eng/staff/jrk/>

See also <https://youtu.be/GC1Pr5Qpd5A> for an example from chemistry.

Molecular Dynamics

Introduction (continued)

These types of simulations are called molecular dynamics simulations.

This lecture will demonstrate how we can perform simple molecular dynamics simulations using DifferentialEquations.

Hamiltonian systems

According to Newtonian mechanics, the state of a system of n identical atoms is fully determined if we know the positions $(x_i(t) \in \mathbb{R}^d)_{i=1}^n$ and velocities $(v_i(t) \in \mathbb{R}^d)_{i=1}^n$ of the atoms.

Moreover, these quantities satisfy the ODEs

$$\dot{x}_i = v_i, \quad \dot{v}_i = F_i((x_j)_{j=1}^n)$$

where $F_i((x_j)_{j=1}^n)$ denotes the force on atom i and where I assumed that the masses of the atoms are $m = 1$.

Molecular Dynamics

Hamiltonian systems (continued)

In molecular dynamics simulations, the force

$$F_i((x_j)_{j=1}^n) = -\nabla_{x_i} V((x_j)_{j=1}^n)$$

is minus the gradient with respect to x_i of a function $V((x_j)_{j=1}^n)$, which is called the potential energy.

Adding the potential energy to the kinetic energy $\sum_{i=1}^n \frac{1}{2} \|v_i\|_2^2$ yields the total energy

$$H((x_i), (v_i)) = \sum_{i=1}^n \frac{1}{2} \|v_i\|_2^2 + V((x_i))$$

which is also called the Hamiltonian of the system.

Since $\nabla_{v_i} \frac{1}{2} \|v_i\|_2^2 = \|v_i\|_2 \frac{v_i}{\|v_i\|_2} = v_i$, the above ODEs for x_i and v_i can also be written as

$$\dot{x}_i = \nabla_{v_i} H((x_j), (v_j)), \quad \dot{v}_i = -\nabla_{x_i} H((x_j), (v_j)).$$

The equations of motion are thus fully specified once we provide an expression for Hamiltonian $H((x_i), (v_i))$, which in turn is specified once we provide an expression for the potential energy $V((x_i))$.

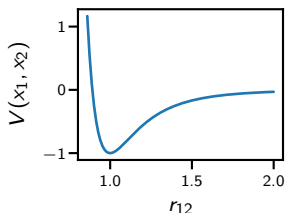
Molecular Dynamics

Lennard-Jones potential

A simple potential energy yielding qualitatively correct results is given by

$$V((x_i)) = \sum_{i < j} \left(r_{ij}^{-12} - 2r_{ij}^{-6} \right), \quad r_{ij} = \|x_i - x_j\|_2.$$

An intuition for this function can be obtained by considering a system of only $n = 2$ atoms and plotting $V(x_1, x_2)$ as a function of $r_{12} = \|x_1 - x_2\|_2^2$.



The negative slope for $r_{12} < 1$ implies that atoms repel if they get too close, and the positive slope for $r_{12} > 1$ implies that atoms attract if they are far enough apart. Since the slope goes to 0 for $r_{12} \rightarrow \infty$, the attraction vanishes if atoms are too far apart.

Molecular Dynamics

Numerical implementation

See the code file for this lecture.

Some remarks:

- ▶ Adaptive time-stepping usually yields no performance benefit for molecular dynamics simulations: the point of adaptive time-stepping is to “zoom in” on those times where something critical happens, but in a large enough molecular dynamics simulation there is always something critical happening somewhere in the simulation box. We therefore disable adaptive time-stepping by passing `adaptive = false` to the `solve()` function.
- ▶ Our simulation uses periodic boundary conditions: if an atom leaves the simulation box on one side, it reappears on the other side. If we did not do this, then the atoms would one-by-one disappear to infinity and never interact again. Periodic boundary conditions require us to map the atoms back into the simulation box after every time step. This can be done using a callback function; see `enforce_pbc`.

Molecular Dynamics

Energy conservation

An easy calculation reveals

$$\begin{aligned}\frac{d}{dt} H((x_i), (v_i)) &= \sum_{i=1}^n \left(\dot{x}_i \cdot \nabla_{x_i} H((x_i), (v_i)) + \dot{v}_i \cdot \nabla_{v_i} H((x_i), (v_i)) \right) \\ &= \sum_{i=1}^n \left(-\dot{x}_i \cdot \dot{v}_i + \dot{v}_i \cdot \dot{x}_i \right) = 0,\end{aligned}$$

i.e. the total energy $H((x_i), (v_i))$ of the system does not change over time.

Of course, it is unreasonable to expect that our numerical simulation preserves the energy exactly. However, `energy_conservation()` shows that not only is the total energy not preserved, it drifts over time.

You might have to hide the potential and kinetic energy from the plot to see the variation in the total energy.

Molecular Dynamics

Energy conservation (continued)

This energy drift is a problem if we are interested in simulating over very long time spans. It can be avoided using either of two approaches:

- ▶ Choose a very small time step Δt .
- ▶ Use a symplectic integrator.

The first approach is often not practical because the time step would have to be excessively small to keep the drift within reasonable bounds.

The second approach is based on the observation that the trajectory $(x_i(t)), (v_i(t))$ of a Hamiltonian system stays on a geometrical object called a symplectic manifold.

There are numerical ODE solvers called symplectic integrators which ensure that the numerical trajectory $(\tilde{x}_i(t)), (\tilde{v}_i(t))$ stay on a symplectic manifold which is a slightly perturbed copy of the original manifold.

For such symplectic integrators, the total energy typically oscillates but it does not drift. Replaced `method = Midpoint()` with `method = VerletLeapfrog()` in `energy_conservation()` for numerical demonstration.

Molecular Dynamics

Stiffness-induced time-step constraints

Recall from Lecture 18 that explicit ODE solvers have a tendency to lead to blow-up of the numerical solution even in cases where the exact solution oscillates or even converges to a steady-state.

This phenomenon also occurs in molecular dynamics simulations. Replace $dt = 0.06$ with $dt = 0.07$ in `energy_conservation()` for numerical demonstration.

In molecular dynamics simulations, it is virtually impossible to provide a rigorous mathematical analysis of this phenomenon. However, explaining it on a heuristic level is fairly straightforward.

If two atoms are a distance $r_{ij} \approx 1$ apart, they oscillate against each other with a characteristic frequency $\omega > 0$, i.e.

$$r_{ij}(t) \approx 1 + C \cos(\omega t).$$

This frequency ω roughly corresponds to an imaginary eigenvalue $\lambda = \omega \iota$ of the Jacobian $\nabla f(y_F)$ of $f(y) = \dot{y}$ at the partial fixed point $r_{ij} = 1$ (take this sentence with a very big grain of salt).

Molecular Dynamics

Stiffness-induced time-step constraints (continued)

To avoid blow-up, we must ensure that $\lambda = \omega \iota$ is in the stability region of the ODE solver, or at least close to it. For explicit ODE solvers, this effectively means that we must choose $\Delta t \lesssim \frac{1}{\omega}$, i.e. we must choose the time step Δt small enough such that we have at least a constant number of time points t_k per oscillation.

A constant number of time steps per oscillation is not an issue if the oscillation is the phenomenon that we are interested in. However, in many molecular dynamics simulations we have some components which oscillate with very high frequencies but very small amplitudes. These components force us to take excessively small time steps without contributing any significant physics to the simulation, i.e. they lead to a stiff system of ODEs.

This stiffness could be resolved by switching to an implicit ODE solver, but that would require us to solve a huge nonlinear system of equations in every time step which is not feasible.

Molecular Dynamics

Stiffness-induced time-step constraints (continued)

Instead, physicists often reduce the stiffness by “freezing” the troublesome components: high-frequency / low-amplitude oscillations typically arise as a result of very strong chemical bonds between atoms. Since the bond is strong, we lose little accuracy if we treat the bond length as fixed and forbid the atoms from oscillating against each other. This eliminates the step-size constraint induced by the high-frequency oscillation.

Example: water is often treated as a molecule with fixed bond lengths and angles rather than a system of three atoms for precisely the above reason.

Summary

- ▶ Symplectic integrators allows us to simulate Hamiltonian systems over long time spans without energy drift.
- ▶ Molecular dynamics simulations are often stiff but using an implicit ODE solver is not an option. Instead, the stiffness can be reduced by freezing high-frequency, low-amplitude bonds.