

Q&A Lecture 1

The ability to implement the velocity Verlet algorithm is dependent on the fact that we know initial conditions, such as position and momentum. How is this determined?

In principle, the choice of initial conditions is arbitrary since - if the dynamics is integrated correctly - the trajectory will eventually start to visit the correct phase space points with the right frequency - more on this tomorrow. In other words, the dynamics relaxes the system towards the correct statistical situation. In practice, since this “equilibration phase” cannot be used to accumulate averages and is therefore wasteful numerically, one tries to find sensible initial coordinates (e.g. by using known crystal structures for the system) and momenta (e.g. by ensuring that they provide an estimated temperature close to the physical one). We may say more on this in future classes.

And regarding molecular dynamics, would this be a realistic choice of algorithm to determine the time evolution of a larger system (considering computational cost and time efficiency)?

With empirical potentials, velocity Verlet is the most common algorithm used nowadays. It enables simulations of millions of degrees of freedom for several nano seconds at reasonable numerical cost.

How did we derive Velocity Verlet Algorithm in the lecture? I understand the two equations about $r(dt)$ and $r(2dt)$, but how can I deal with the two equations to derive Velocity Verlet Algorithm?

You will find a derivation of the velocity part in the book by Marc Tuckerman, Statistical Mechanics Theory and Molecular Simulation, Chapter 3, section 3.8.2. Let me know if it is unclear.

I understand that $H = K + V$ is a scalar equation, but in the end, the two Hamilton's equation must be solved in the three dimensions for every particle, similarly to Newton ones. So, it looks like we probably reduce interdependency of the variable in the Hamiltonian formulation (by separating the momenta and positions, but not sure if that really helps), but I do not really get why and how this formulation of classical physics has an advantage over Newton in this context, why and how does it effectively reduces computational costs ?

The advantage is conceptual and lies in the generality of the forms of Hamiltonians that one can adopt. Also, the notion of generating the evolution from a Hamiltonian (not necessarily scalar) transfers to other mechanics - notably quantum and relativistic - while Newton does not. You are correct in saying that the computational cost of the numerical solution is the same in both formulations. However, adopting a more general formulation enables - for example - to derive evolutions beyond the microcanonical ensemble. In a different context, still numerical though, one can use other formalisms - Lagrangian - to have the more convenient framework for the evolution equations of systems subject to constraints.

You mentioned during class that if the time step chosen in the velocity verlet algorithm was too high, then the energy would diverge to large values very quickly, and I don't really see why it is the case. Could you help me on that ?

One way to see it is as follows: think about the approximation that we have used to obtain the expression for the update of the positions. It is based on a Taylor series expansion whose accuracy is controlled by the time step size. Too large a time step means a very inaccurate set of positions and - because momenta are (for the separable Hamiltonians that we are considering) related to time derivatives of the positions - a bad finite difference approximation of the momenta. Thus, the updated phase space point bears no dynamical relationship to the previous one and energy is not conserved. The divergence is a result of the fact that when one feeds "bad" input to the finite difference solution of the evolution equation, the algorithm becomes unstable.

When should we use classical mechanics or quantum mechanics when simulating atoms and molecules are there some results that can only be obtained via classical methods because the quantum computation get too heavy ?

There are some criteria - to be discussed in part in future classes - to decide when it is necessary to adopt a quantum description of the system. For example: one can look at typical energy scales (e.g. as identified by the thermal energy of the system) and compare it with typical quantum energy scales (e.g. as identified by zero point energy effects); alternatively, one can look at typical distance scales (e.g. as identified by decay of distance dependent interactions, or typical bondlengths) and compare them with typical quantum length scales (e.g. the de Broglie wavelength of a particle). Whenever these quantities are comparable (or the typical scales smaller than the quantum counterparts), it is necessary to adopt a quantum description. If the quantum calculation is too heavy - as you say - there are three choices, all of which we shall explore: (1) we give up and wait for more powerful computers (rare...), (2) we create approximate quantum dynamical schemes that aim at reducing the numerical cost while retaining the most relevant features of the phenomenon under study (in this case, which is one of the focuses of the class, it is important to be aware of the approximations and their ranges), (3) we use classical mechanics, knowing that this is not the full story.

You said during class that if the time step in the velocity verlet algorithm is too big, then the energy explodes, but I am not sure if I really understood why. Could you clarify this please ?

One way to see it is as follows: think about the approximation that we have used to obtain the expression for the update of the positions. It is based on a Taylor series expansion whose accuracy is controlled by the time step size. Too large a time step means a very inaccurate set of positions and - because momenta are (for the separable Hamiltonians that we are considering) related to time derivatives of the positions - a bad finite difference approximation of the momenta. Thus, the updated phase space point bears no dynamical relationship to the previous one and energy is not conserved. The divergence is a result of the fact that when one feeds "bad" input to the finite difference solution of the evolution equation, the algorithm becomes unstable.

What happens if we have 2 interacting systems? Is there a way to compute our parameters? Find the equation of evolution?

Not sure to understand the question. If we have the interaction potential between the two systems (as is the case in classical molecular dynamics) then yes, we can find the evolution

equations and integrate them using Verlet. What do you mean by compute our parameters? The parameters of the potential or something else?

The formula of velocity Verlet Algorithm is similar to the function expansion, where the first equation of Eq. (2) is essentially the Taylor expansion. Does this mean that we can utilize different schemes, such as central difference, in numerical differentiation to derive different version of velocity Verlet algorithm? If so, how to decide which version to use in practice?

The velocity Verlet algorithm is uniquely defined by the equations that we discussed in class. There are other algorithms (e.g. forward Euler, the simple Verlet, the leap frog, Gear etc) that use different approximations to the short time dynamics and/or the derivatives. The choice of a given algorithm depends - a bit - on taste and - a lot - on its properties. Relevant properties are linked to the ability to preserve properties of the continuous equations (i.e. the analytical evolution equations). Velocity Verlet, for example, is time-reversible and symplectic (preserves the measure of phase space) and third order accurate in dt , requiring only first derivatives of the potential for its implementation and enabling a straightforward definition of the estimator of the energy (important because we can use it in a check of conservation to decide if we are using a good time step). These are all good properties of an algorithm and explain why velocity Verlet is the most commonly used evolution algorithm in classical mechanics. For example, Gear and forward Euler are not time-reversible. Leap frog does not give positions and momenta at the same time, so it complicates the estimate of the energy...

In the derivation of the Velocity Verlet method, we assume time reversibility when we write the co-ordinate at time t , that is $\vec{x}(t)$, in terms of momentum and force at time $t + \delta t$, that are $p(t + \delta t)$ and $F(x(t + \delta t))$. This means that the forces are conservative, right? Can we adapt this method to also incorporate dissipative effects?

Yes, the forces we have considered so far are conservative (we are in the constant energy ensemble!). Algorithms that include dissipative effects, e.g. friction forces (sometimes "balanced" by a random force as in Langevin dynamics) exist. Note, however, that time reversibility was not assumed in the derivation of the Verlet algorithm. In fact, it is a nice property of the algorithm to be time-reversible, but this is a product of the time-stepping equations, not an hypothesis in their derivation.

While we can use a computer to predict new values of $r(t)$ and $p(t)$, we have seen that the magnitude of their associated errors depends on the chosen time increment dt . While the error decreases when choosing a smaller dt , the time the computer takes to compute these values will be longer. Instead of by iteration, is there an analytical method that can be used to estimate the precision of the solution (and not by just verifying the conservation of energy)? How do we optimise the choice of the value of dt ?

There are bound formulas that link the size of the time step to the error. These are rather complex to derive and often not very general (in the sense that their applicability depends strongly on the characteristics of the potential). A good rule of thumb is that the time step needs to be one order of magnitude smaller than the fastest oscillation frequencies in the system. These can be estimated - in particular for bound systems like molecules - by considering the Hessian of the potential that - in a local harmonic approximation - is related

to such frequencies. Note that second (and higher) derivatives of the potential are also the terms that are neglected in the velocity Verlet. The energy conservation remains, in all cases, the key test. Your point is, however, well taken, the smaller the time step, the more iterations are needed to cover a given time interval, and therefore the higher the numerical cost. Consequently, the choice of the time step is based on the best compromise between accuracy (e.g. as fixed by the threshold setting the boundary on energy conservation) and cost. To the best of my knowledge, this is done in practice by set up calculations in which the time step is tuned.

By measuring an observable like the temperature, we make several measures of a microscopic estimator of this observable (each measure requires a 'microscopic time' dt) and we make an average of these measures (so, the macroscopic measure require $M dt$ with M the number of microscopic measures)?

Yes, you understood correctly. To be more precise, the macroscopic average as written in class requires M statistically independent microscopy measurements. Since successive microscopic observations are correlated (one can predict positions and momenta based on the previous one), typically microscopic recordings are taken at intervals that guarantee statistical independence (say every 50 steps — this can be made more rigorous), which means that we need “longer trajectories” to acquire good sampling.

Furthermore, aren't collisions between molecules significant for predicting the position and speed of a molecule in a glass of water (too simplistic model)?

Collisions are modelled by repulsive terms in the interaction potential. These are not - however - collisions in the sens of “hard spheres” in which you really have atoms banging against each other but rather result in stiff - but still continuous - repulsive forces between the elements.

How stable is the Verlet velocity algorithm? In other words, if the initial conditions ($r(0)$ and $p(0)$) of each molecules are slightly disturbed, to what extent is the final result (and after how long) impacted?

If the time step is small enough, the stability of the integration is the stability of the underlying Hamiltonian motion. As you probably know, for generic interaction potentials and not too low temperatures, classical dynamics is chaotic in that small perturbations in initial conditions rapidly lead to diverging trajectories (true analytically and numerically). The exact speed of divergence can be computed via the so-called Ljapunov esponents for the dynamical systems — these are related to relatively abstract objects such as the eigenvalues of the evolution operator. Can redirect to biblio if you are interested.

What modifications are needed to calculate observables from a system of molecules (that have fixed bond lengths), as opposed to individual atoms?

Molecules do not really have a fixed bond length but if we choose to model them as rigid objects, the evolution equations and the algorithm need to be modified to include the associated conditions (e.g. in a diatomic, a fixed distance between the two atoms). This is done by incorporating these conditions as constraints in the dynamical system and then adapting the velocity Verlet algorithm with an extra set of operations that ensure that the

constraints are satisfied at each time step. This generalised algorithm is called SHAKE. We shall discuss constrained evolution more in detail in future classes.

Since velocity is a vector, what does the square of a vector PHYSICALLY represent when calculating kinetic energy for example? In other terms, what is the link between the mathematical object and the physical quantity we measure in the real world ?

The square of a vector is related to its magnitude (typically via square root). So the kinetic energy (when measured in terms of velocities) provides information on the total change in positions over an infinitesimal amount of time. The kinetic energy is also - as we saw in class - related to the temperature of the system.

In what situations and/or systems would the equations based on momentum have an obvious advantage over other quantities such as velocity?

Velocities are time derivatives of positions and - as such - cannot be considered independent. Momenta, in the Hamiltonian formalism, are independent variables from positions. This is important in statistical mechanics because it allows to define microscopic estimators of macroscopic quantities in terms of the independent variables that define the microscopic state of the system at each time step. Furthermore, momenta remain the good microscopic descriptors of the state also when they are not just mass times velocities. One example of this situation is charged systems in the presence of an external magnetic field or rotating system.

The ability to implement the velocity Verlet algorithm is dependent on the fact that we know initial conditions, such as position and momentum. How is this determined?

In principle, the choice of initial conditions is arbitrary since - if the dynamics is integrated correctly - the trajectory will eventually start to visit the correct phase space points with the right frequency - more on this tomorrow. In other words, the dynamics relaxes the system towards the correct statistical situation. In practice, since this "equilibration phase" cannot be used to accumulate averages and is therefore wasteful numerically, one tries to find sensible initial coordinates (e.g. by using known crystal structures for the system) and momenta (e.g. by ensuring that they provide an estimated temperature close to the physical one). We may say more on this in future classes.

And regarding molecular dynamics, would this be a realistic choice of algorithm to determine the time evolution of a larger system (considering computational cost and time efficiency)?

With empirical potentials, velocity Verlet is the most common algorithm used nowadays. It enables simulations of millions of degrees of freedom for several nano seconds at reasonable numerical cost.

When "constants of motion" like the Hamiltonian suggest an issue in a simulation, how can we efficiently identify the source of the error, given the many potential factors such as interaction models, timestep, integration algorithms, or system size, without having to exhaustively check each one?

Ehmm, one cannot determine the source of the error without thorough check of the equations implemented in the code and the code itself.

Why do we start the lecture with Hamiltonian mechanics rather than Lagrangian mechanics? Is it because Hamiltonian formalism has better analogy in quantum mechanics?

Yes, the Hamiltonian is the generator of the time evolution also in quantum mechanics and therefore this formalism enables to establish closer counterparts. We shall consider situations in which the Lagrangian formalism is convenient in future classes.

I learned that in quantum mechanics the Hamiltonian of a system is an operator for the total energy of that system. So if the Hamiltonian is the generator of the classical mechanics, is there a link with the quantum definition ?

Also in classical mechanics, the Hamiltonian is the observable associated to the energy of the system. Also, the time derivative of a quantum operator (i.e. the equation for its evolution in time) is linked to the commutator of the operator with the Hamiltonian operator. So, the time evolution is obtained from manipulations of the operator and the Hamiltonian. The formal solution of the quantum evolution - as we'll see tomorrow - is also expressed in terms of the Hamiltonian. In classical, we have seen that the evolution equations of momenta and coordinates are obtained from derivatives of the classical Hamiltonian. It is also possible to recast this via something that is the classical counter part of the commutator, i.e. the Poisson parenthesis of the Hamiltonian with the coordinates and momenta. Finally, though we did not see it in class, also the formal solution for the evolution can be expressed in terms of derivatives of the Hamiltonian. Due to all these analogies, the generator of the motion is the same - i.e. the adequate representation of the Hamiltonian - both in classical and quantum mechanics.