DynamO Workshop

Introduction to Event-Driven Dynamics and DynamO

Dr Marcus N. Bannerman & Dr Leo Lue

m.campbellbannerman@abdn.ac.uk leo.lue@strath.ac.uk

Section Outline

Agenda

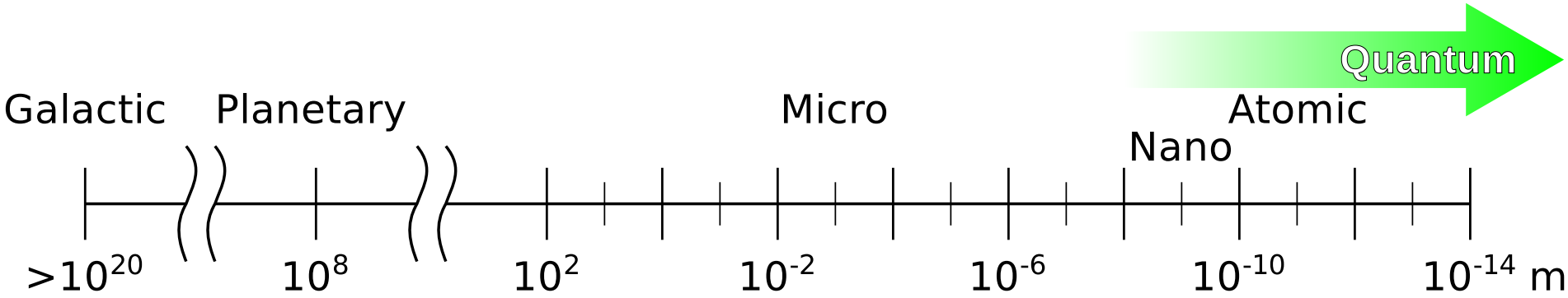
Agenda

Time	
9:30–10:00	Registration
10:00–11:00	Introductory talk
11:00–12:00	Equilibrium simulation of simple discrete fluids
12:00–13:00	Lunch
13:00–14:00	Transport properties of mixtures
14:00–15:00	Complex systems/Polymeric fluids
15:00–15:30	Coffee break
15:30–16:30	Models for protein folding
16:30–17:00	Questions

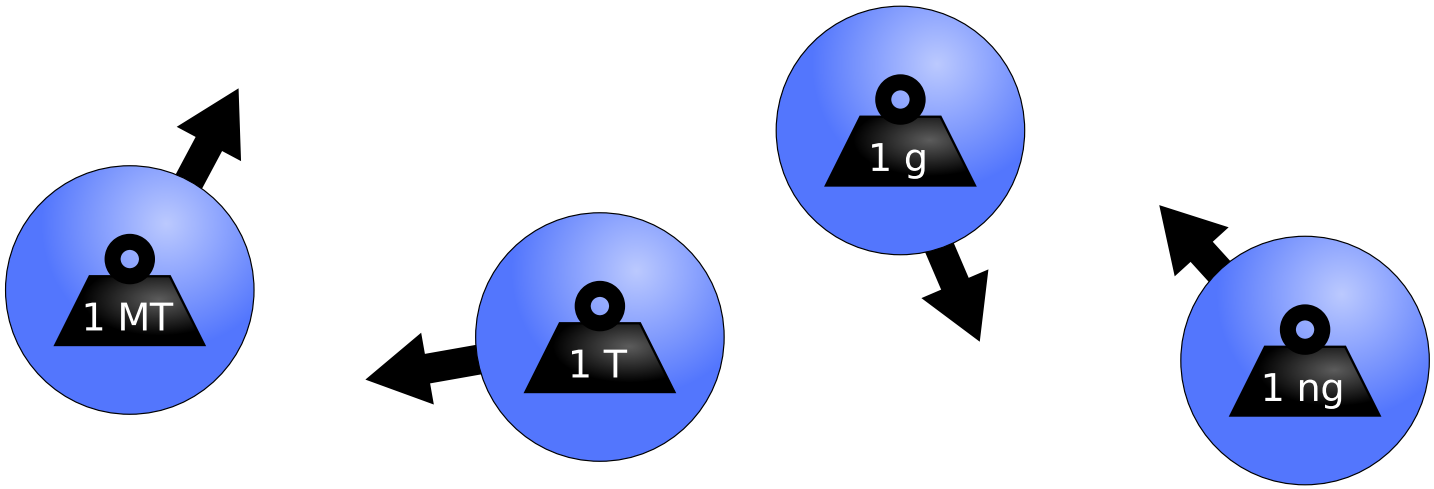
Section Outline

What is DynamO?

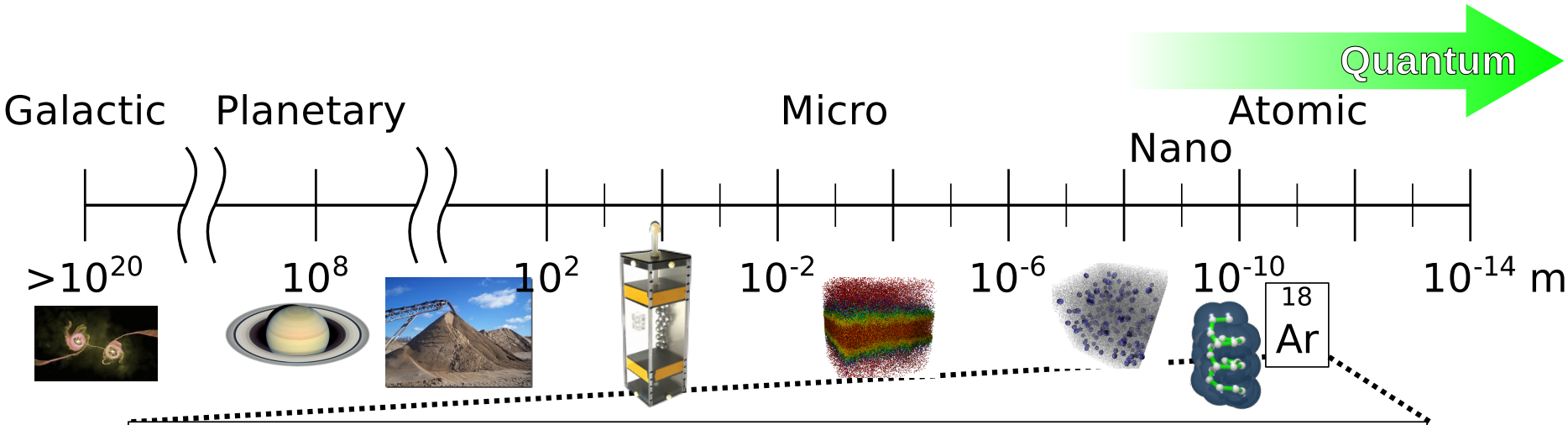
- DynamO stands for **Dynam**ics of discrete **O**bjects.
- It is a *particle dynamics package* and is one of the very few which uses an event-driven simulation approach.
- Event-driven dynamics is mainly applied to relatively simple potentials (hard-sphere, square-well) but the approach is more general than it first appears.
- To illustrate this, we introduce particle dynamics using more traditional time-stepping methods and demonstrate how results from the two approaches may be made equivalent.

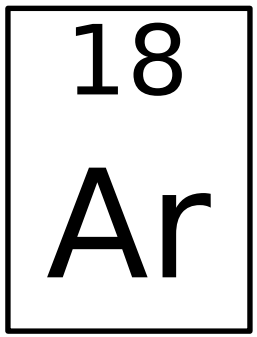


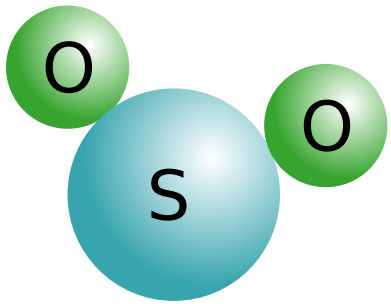
- Particle dynamics is a *classical mechanics* approach to simulating physical systems.
- To model a system, its mass is divided into a number of discrete particles:

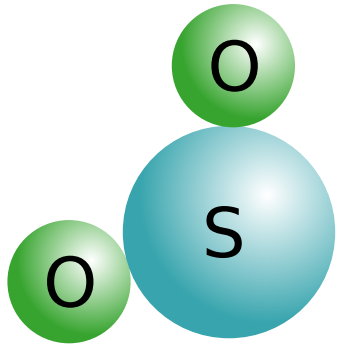


- These particles typically represent some fundamental unit of mass in the system studied...



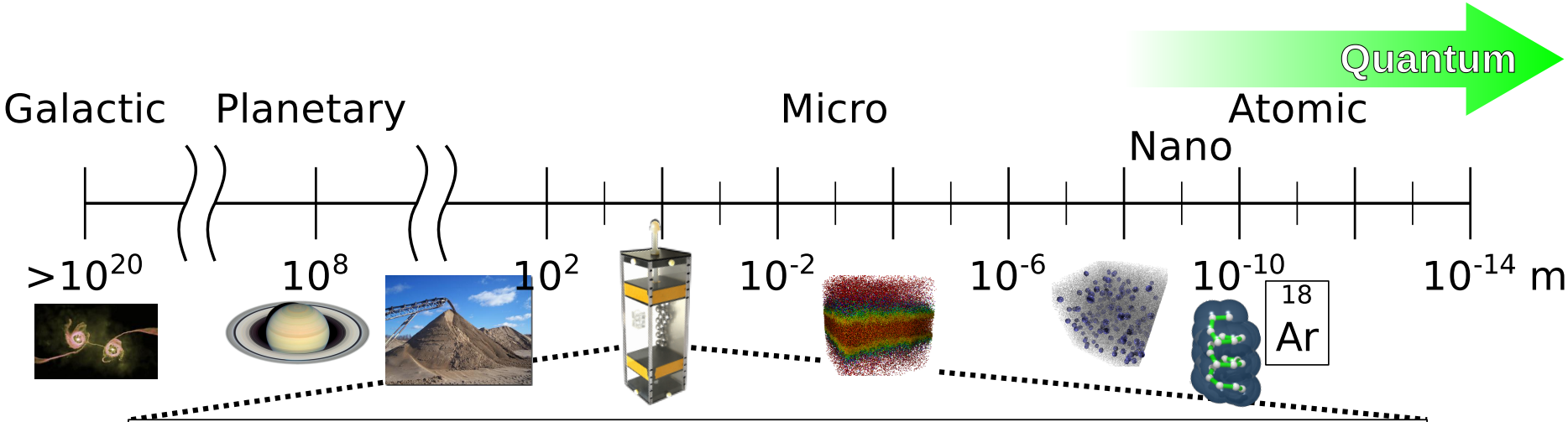


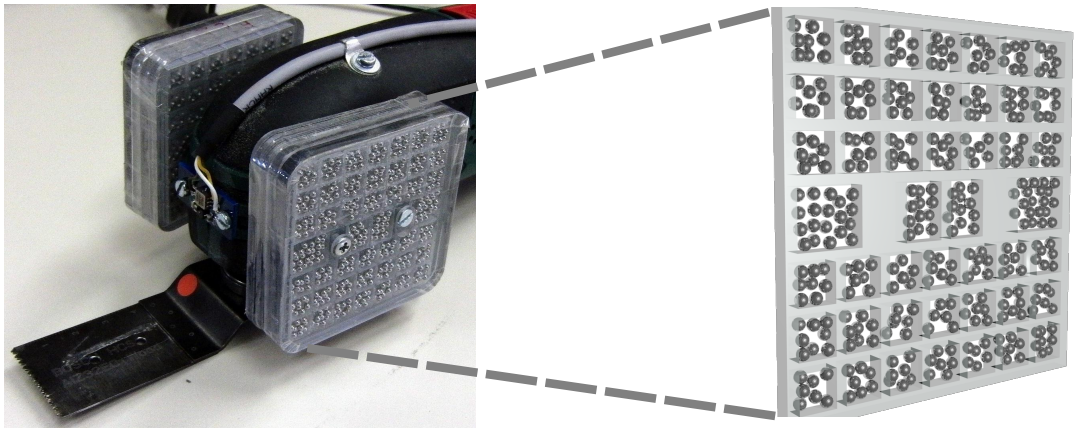




At the molecular scale, each particle may represent a single atom or a functional group.

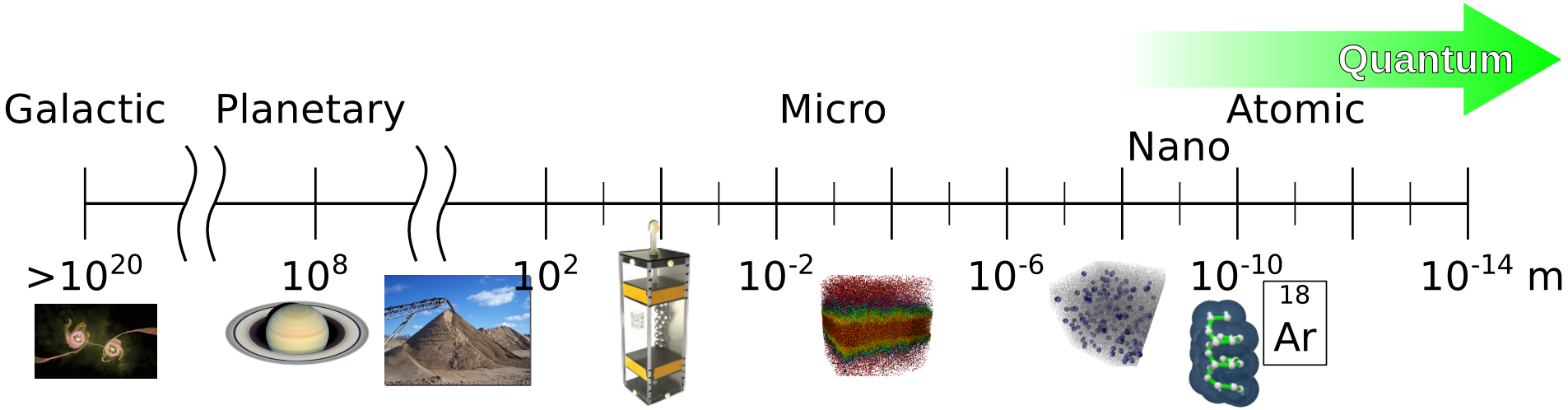
Loup Verlet, "'Computer "Experiments' on Classical Fluids. I," *Phys. Rev.*, **159**, 98 (1967)





At larger scales, a single computational particle may represent one or more "natural" particles.

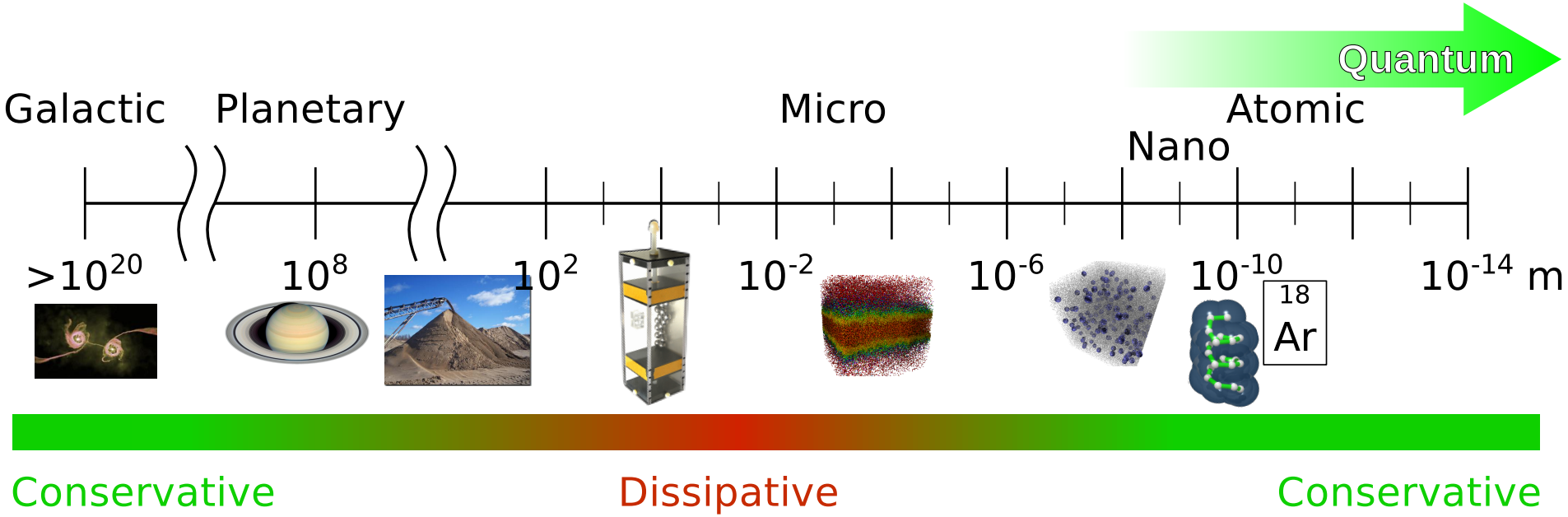
M. Bannerman, J. E. Kollmer, A. Sack, M. Heckel, P. Müller, T. Pöschel, "*Movers and shakers: Granular damping in microgravity*," *Phys. Rev. E*, **84**, 011301 (2011)



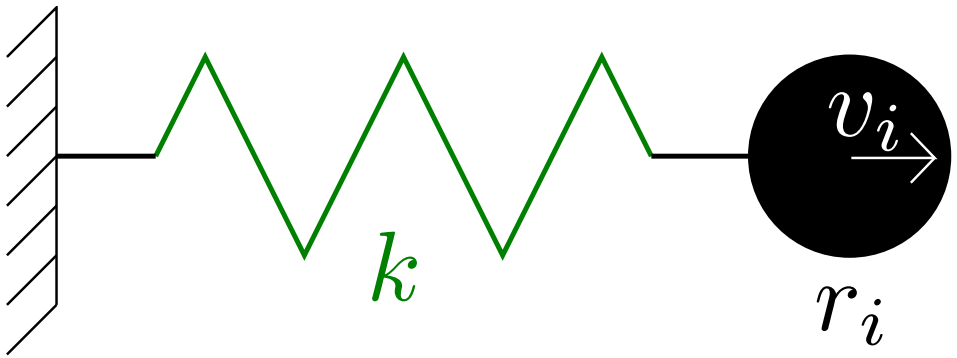
- Each of these systems are simulated by integrating *Newton's equation of motion* (EOM) as expressed for each particle:

$$\boldsymbol{F}_i = m_i \, \boldsymbol{a}_i = m_i \, \dot{\boldsymbol{v}}_i = m_i \, \ddot{\boldsymbol{r}}_i$$

  where $\boldsymbol{F}_i$ is the force acting on particle i, m_i is its mass, $\boldsymbol{a}_i$ is its acceleration, $\boldsymbol{v}_i$ is its velocity, and $\boldsymbol{r}_i$ is its position.
- It is the model expressions used for the forces, $\boldsymbol{F}_i$, which distinguishes which system is under study.



- Although force models are common in time-stepping simulations, the forces in event-driven simulation are not easily defined as each event generates an instantaneous *impulse*. Certain classes of finite forces may also be included in event-driven dynamics (e.g., gravity, oscillating objects).
- Impulsive and continuous forces may be dissipative or conservative, but we will only consider conservative forces today.
- This allows us to compare time-stepping and event-driven approaches through their potential energy function.



- To illustrate this, consider the simplest one-dimensional particle system: a mass, m_i, bound to an immobile wall by a spring.
- Inserting Hooke's law for the force of a spring (rest position of $r_i = 0$) into Newton's equation of motion, we have:

$$F_i = m_i \ddot{r}_i = -k\, r_i$$

- Taking the initial conditions that the spring is at rest $r_i(t = 0) = 0$ and in motion $v_i(t = 0) = A\,\omega$, the solution to this ODE is:

$$r_i = A \sin(\omega\, t) \qquad v_i = A\,\omega \cos(\omega\, t)$$

where $\omega = \sqrt{k/m_i}$ is the frequency of oscillation.

- This is the goal of particle dynamics: to determine the time-evolution of the system phase variables $[r_i,\, v_i]$.

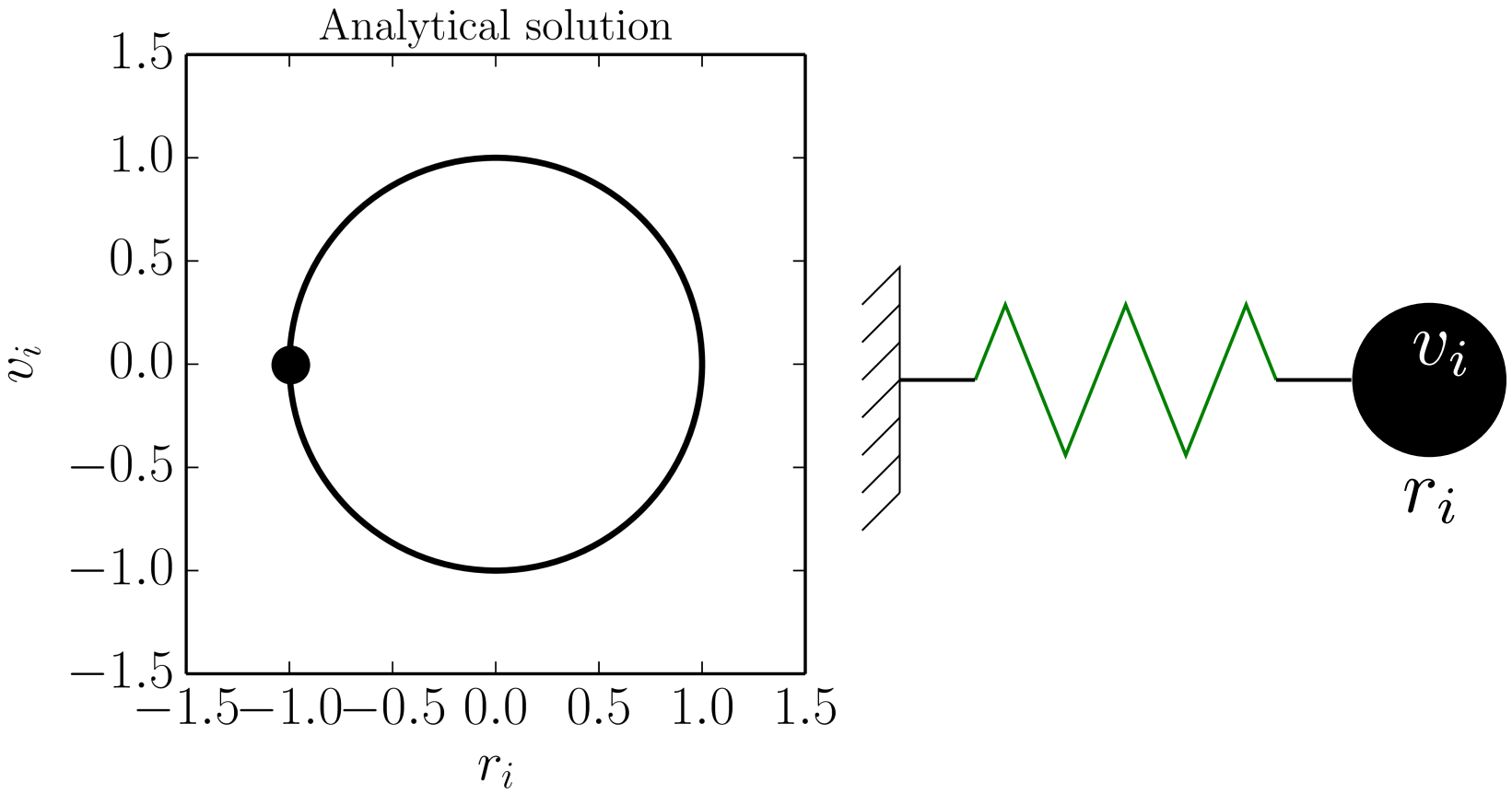
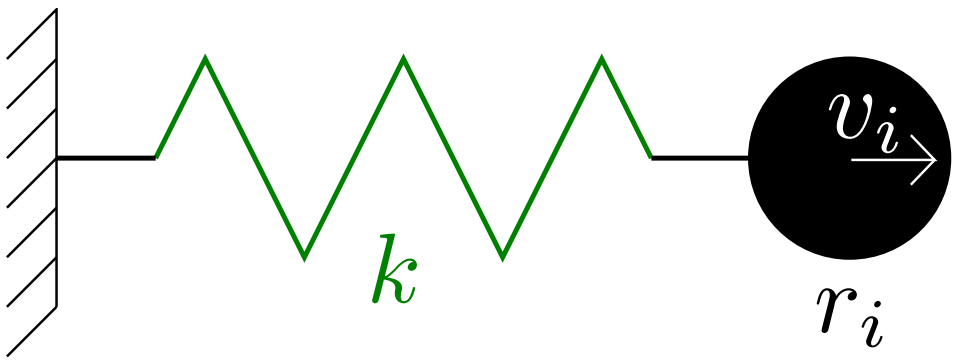


Figure: The exact phase space trajectory of the spring mass system. The parameters k, m_i, and initial velocity $v_i(t = 0)$ are set to 1 and the initial position is set to $r_i(t = 0) = 0$ which gives the solution as a circle of radius 1.



- Assume that Newton's EOM cannot be analytically integrated due to its complexity.
- In time-stepping simulations, numerical integration is used to solve Newton's EOM.
- For example, take a Taylor series of r_i and v_i at the current time t and truncate high order terms:

$$r_i(t + \Delta t) = r_i(t) + \Delta t\, v_i(t) + \cancelto{0}{\mathcal{O}(\Delta t^2)}$$

$$v_i(t + \Delta t) = v_i(t) + \Delta t\, a_i(t) + \cancelto{0}{\mathcal{O}(\Delta t^2)}$$

  where formally Δt is a small "time-step".
- This forward-Euler integration allows us to "take a time step" and estimate $r_i(t + \Delta t)$ and $v_i(t + \Delta t)$ from the initial conditions $r_i(t)$, $v_i(t)$.

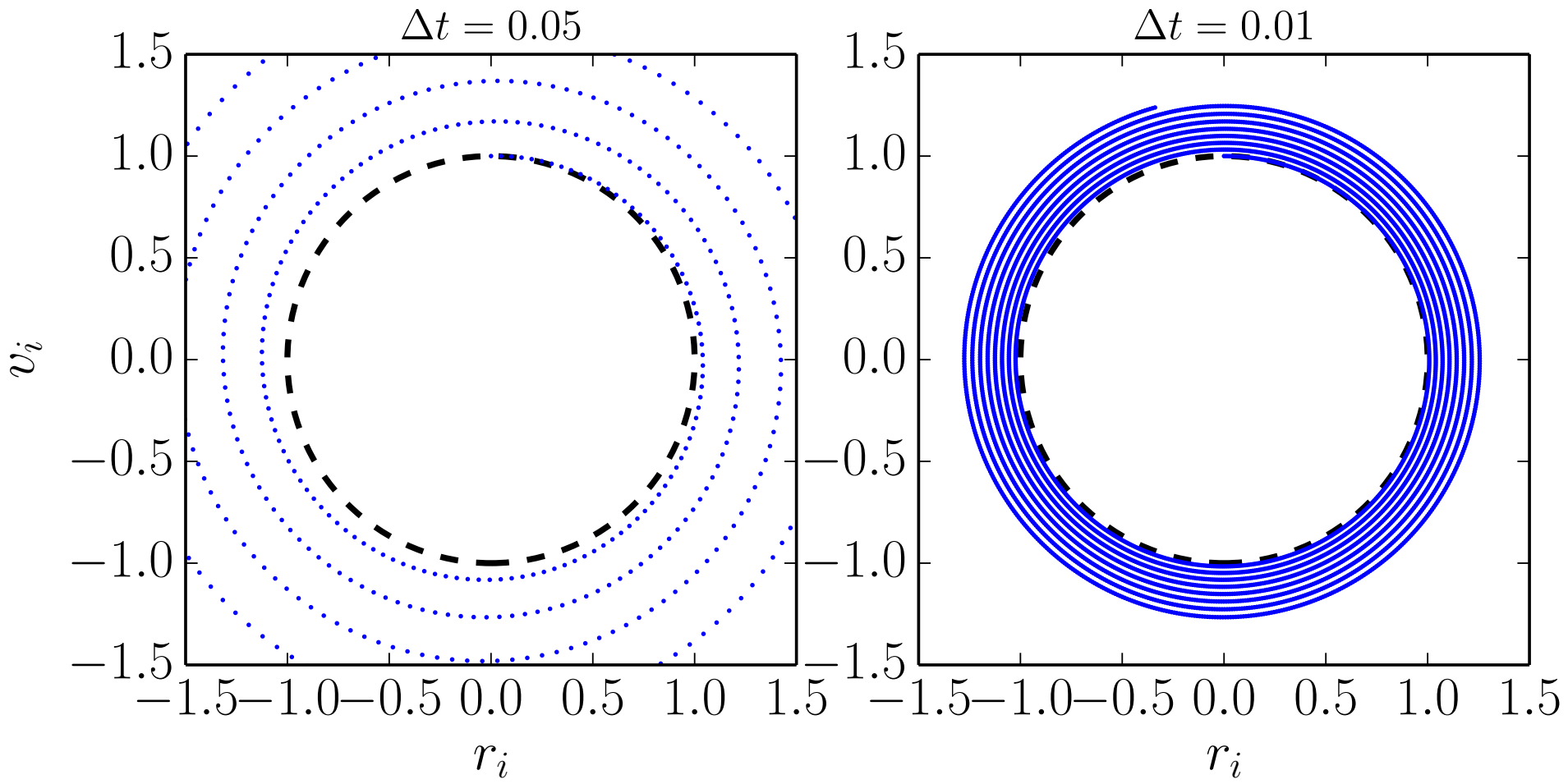


Figure: Numerical solution of the spring mass system over 500 time steps using the Euler integrator and two different step sizes Δt . The error of truncating higher order terms has a consistent bias causing a steady increase in total energy.

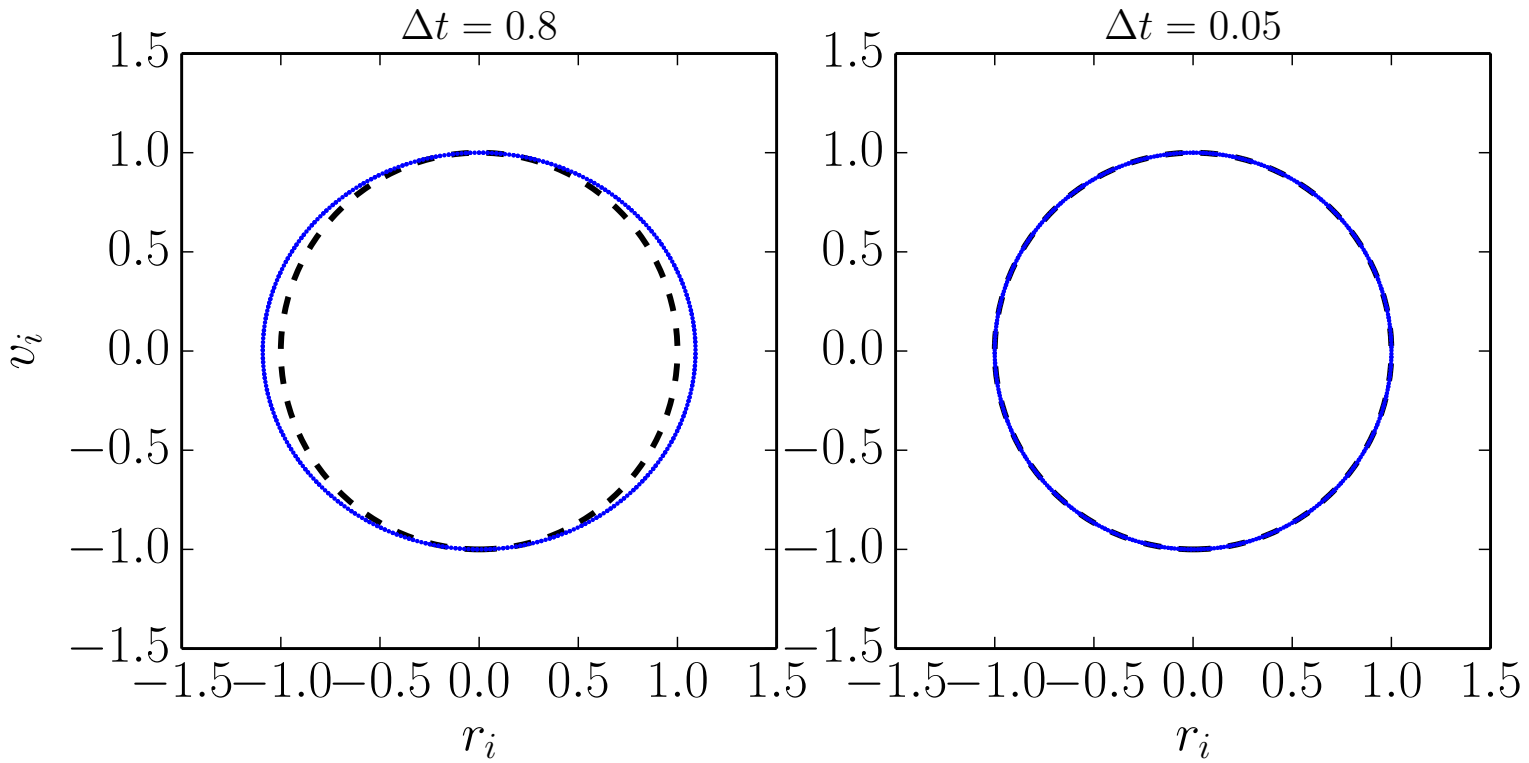
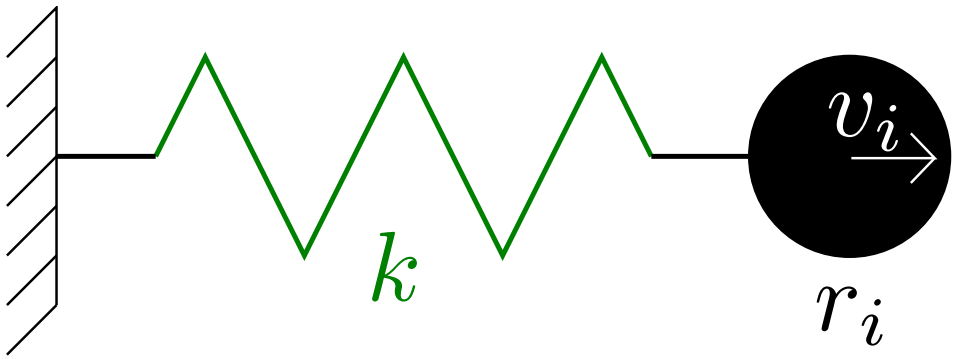


Figure: As before, but using the Velocity Verlet integrator (below) which is symmetric in time. This significantly improves conservation of energy but still simulates a perturbed system; however, for this system even relatively large time steps are extremely close to the exact solution.

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

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



- Now consider the Event-Driven Particle Dynamics (EDPD) approach.
- Assuming Newton's EOM is too complex to analytically integrate, we must decouple the motion of each particle from the rest of the system (for a short period of time) to allow an **analytical solution** to its motion.
- To demonstrate this, we decouple the action of the spring.
- Consider the energetic potential of the spring:

$$U_i = k\, r_i^2/2$$

- To simulate this system using EDPD we must consider a *discrete* or *"stepped"* approximation of the spring potential...

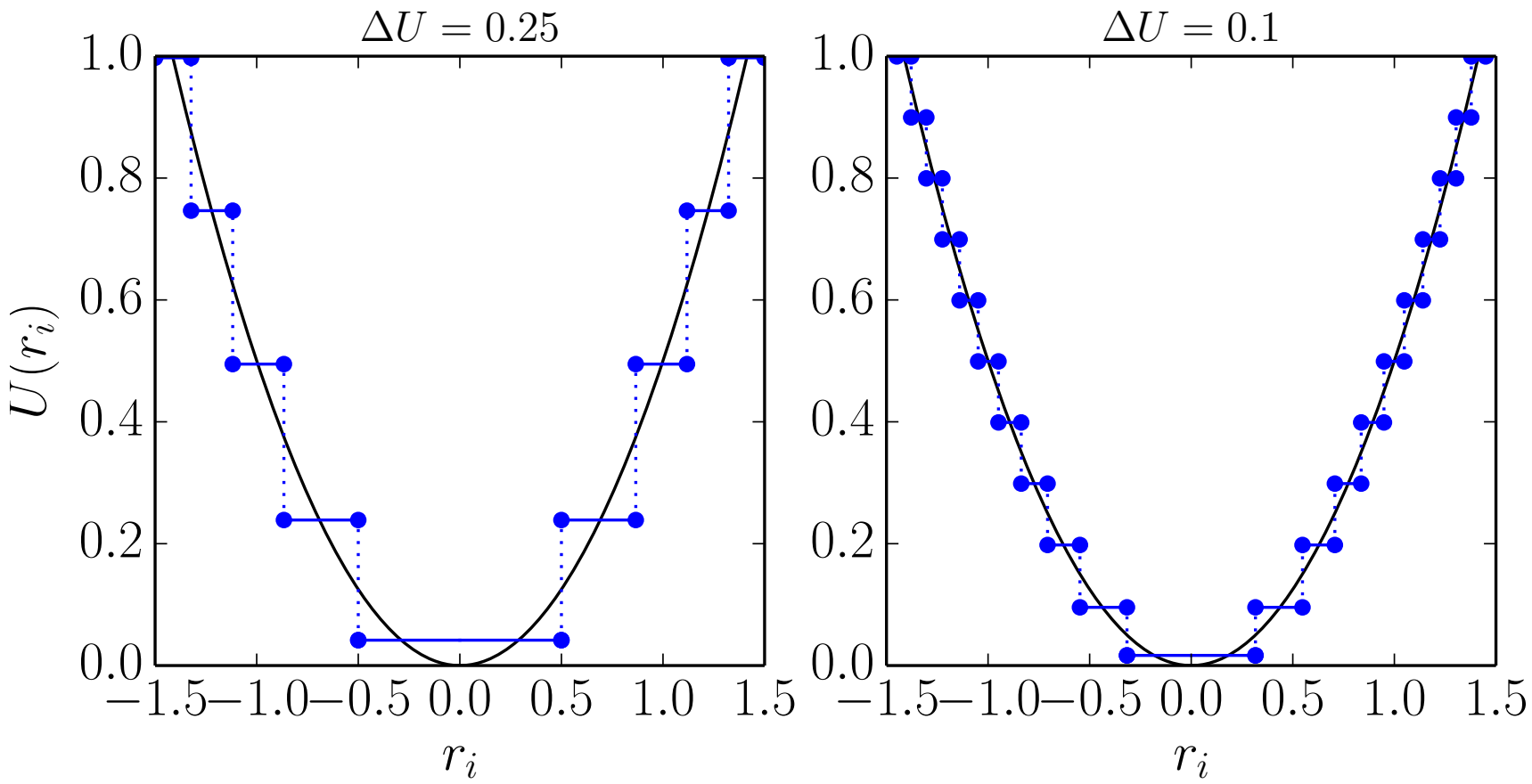


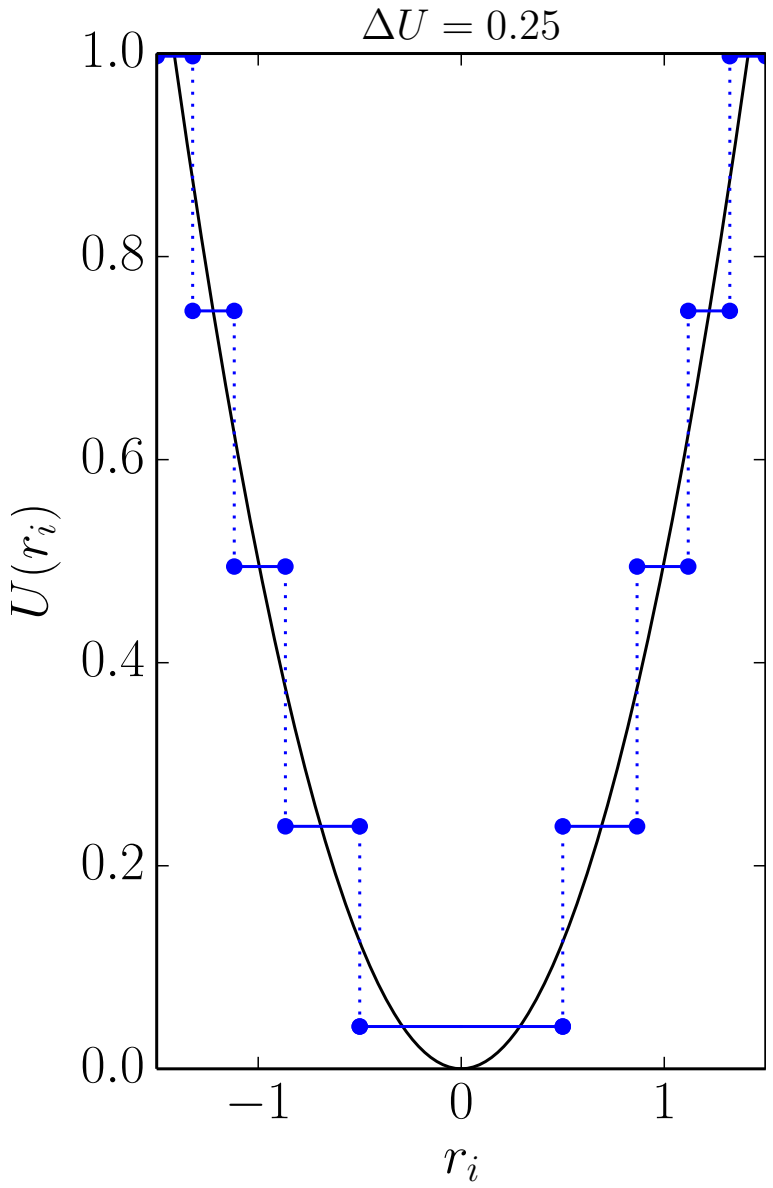
Figure: The potential energy of a spring as a function of position, and two different "stepped" approximations. A potential step[1], ΔU, is introduced as a measure of the maximum deviation between the *continuous* and *discrete* potentials. The potential step, ΔU, (like the time step Δt) controls the accuracy relative to the exact solution.

[1] C. Thomson, L. Lue, and M. N. Bannerman, "Mapping continuous potentials to discrete forms," *J. Chem. Phys.*, **140**, 034105 (2014)

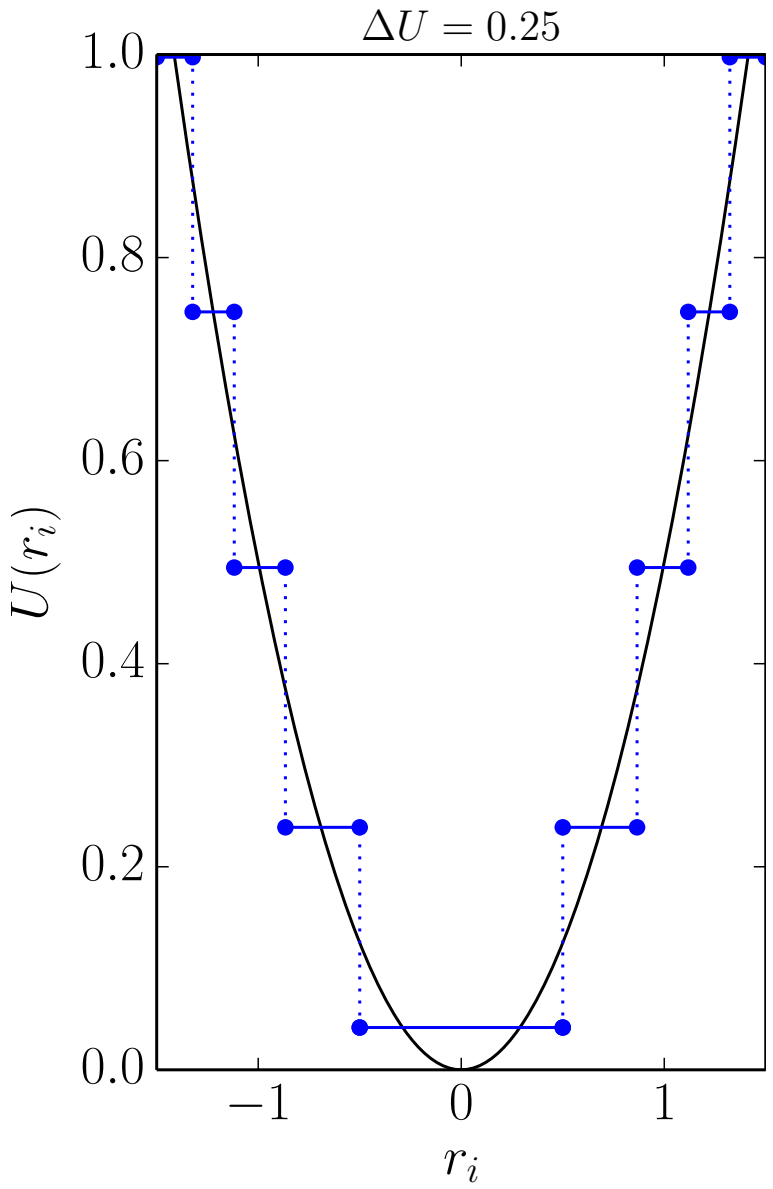
- *Between discontinuities*, $\partial U_i/\partial r_i = 0$ and therefore $\boldsymbol{F}_i = 0$.
- As the force is zero, the particle is temporarily decoupled from the spring and the "free-motion" of the system is trivial ballistic motion:

$$\boldsymbol{r}_i(t + \Delta t) = \boldsymbol{r}_i(t) + \Delta t\, \boldsymbol{v}_i(t)$$

- This is a successful decoupling as *between discontinuities* the motion of the system is analytically described by the equation above.
- We must be careful not to cross a discontinuity while using the analytical solution above.
- Instead, these must be separately treated the *instant* the discontinuity is encountered.



- If we can detect *a priori* the crossing of a discontinuity (an event),
- . . . and calculate the resulting impulse at the time of the crossing,
- we can skip the solution of the "free-motion" entirely.
- EDPD algorithm:
  1. Search for the next discontinuity crossing/event.
  2. Skip through the free motion of the system to the time of the next event.
  3. Calculate and apply the impulse.
  4. Repeat until the end of the simulation.



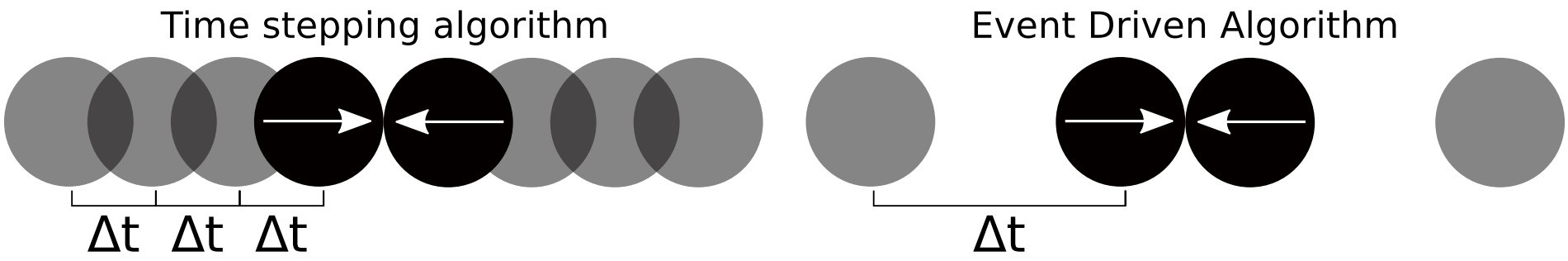


Figure: At low densities, event-driven algorithms can skip uninteresting parts of the dynamics.

- To detect a crossing of a discontinuity (event), consider two particles i and j and discontinuity at a relative separation distance of σ.
- The test for the event is expressed as a search for the (positive) roots of an overlap function, $f(t)$:

$$f(t) = |\boldsymbol{r}_i(t) - \boldsymbol{r}_k(t)| - \sigma$$

where $f(t)$ is a measure of the distance from a discontinuity in the potential.

- Solving for the event impulse, $\Delta \boldsymbol{P}$ is a search for the appropriate solution to the conservation of energy (and momentum):

$$\frac{1}{2} m_i \, \boldsymbol{v}_i^2 + \frac{1}{2} m_j \, \boldsymbol{v}_j^2 + \Delta U = \frac{1}{2} m_i \left(\boldsymbol{v}_i + \frac{\Delta \boldsymbol{P}}{m_i} \right)^2 + \frac{1}{2} m_j \left(\boldsymbol{v}_j - \frac{\Delta \boldsymbol{P}}{m_i} \right)^2$$

where ΔU is the change in internal energy due to the discontinuity.

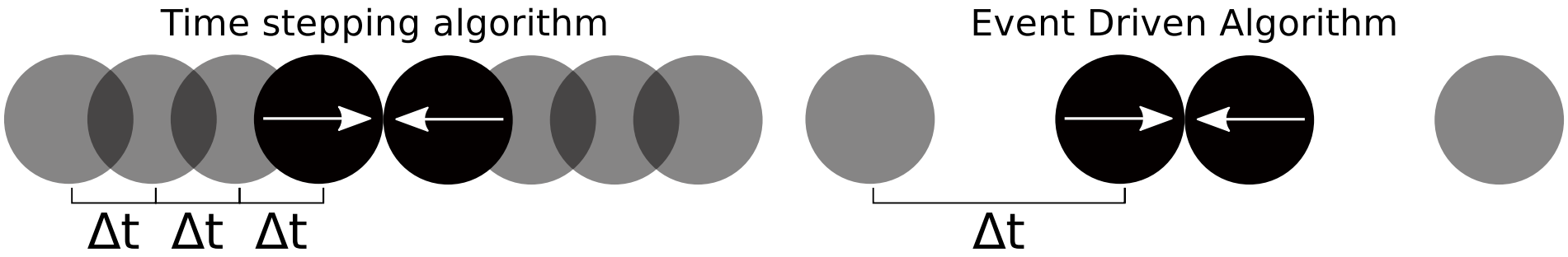


Figure: At low densities, event-driven algorithms can skip uninteresting parts of the dynamics.

- Event-driven dynamics can only be applied if stable root-detection algorithms are available for the overlap and energy balance functions.
- If these exist, event-driven dynamics is an "exact" (to machine precision) solution of the dynamics of the stepped model.
- Energy is conserved to machine precision.
- Round-off error is generally reduced as, during events which don't involve them, particles are left untouched (time-warp).
- Aside from round-off error, the simulated dynamics is reversible (preserves detailed balance) and the simulation may be safely coupled with Monte Carlo techniques.
- Although the stepped spring model is not exactly equivalent to Hooke's law, the stepped approximation may be used to approximate real systems.

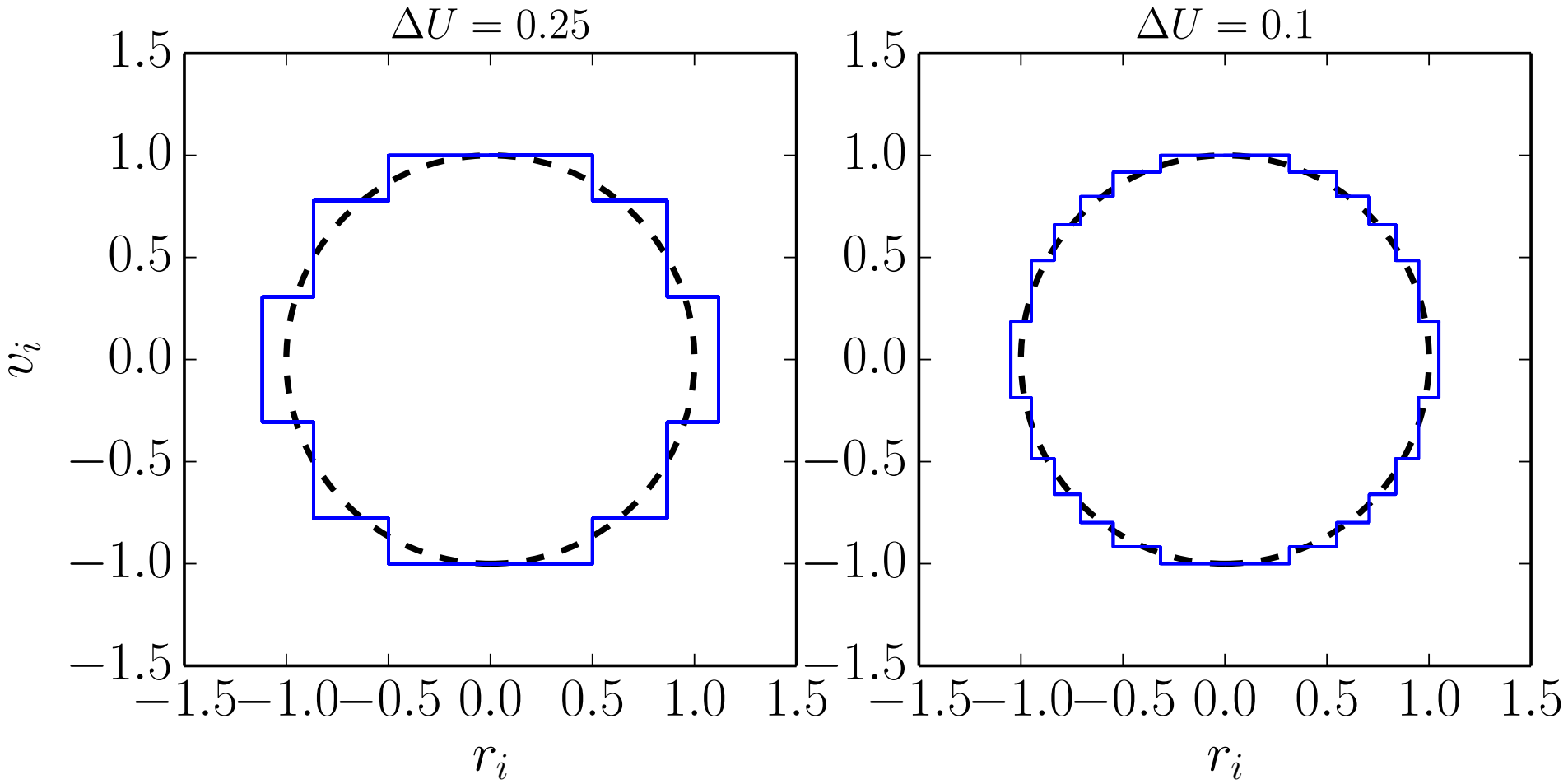


Figure: The phase space trajectory of the spring-mass system approximated using a stepped/discrete potential and EDPD. Vertical sections correspond to instantaneous jumps in the velocity due to the impulse of an event. Horizontal sections correspond to the "free-motion" of the system. In larger ($N > 1$) systems, effects of the discontinuities are "smoothed" out.

Section Outline

Comparison: Continuous vs Stepped Lennard-Jones

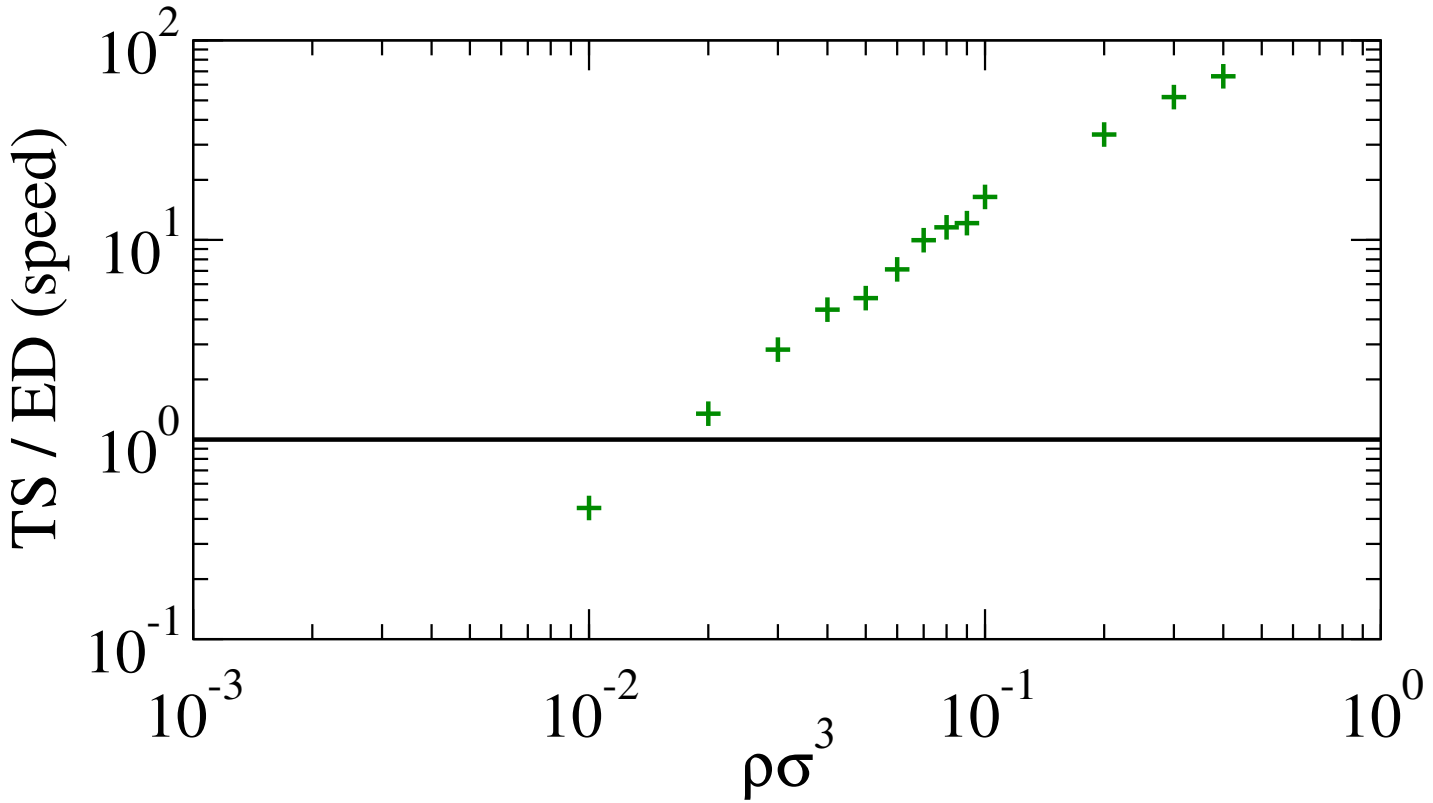


Figure: Relative speed of Time-Stepping (TS) versus Event-Driven (ED) simulation of a Lennard-Jones system with $r_{cutoff} = 3$. The stepped approximation was chosen to reproduce the liquid-vapour densities to high accuracy and transport coefficients to within 10%.[1]

[1] C. Thomson, L. Lue, and M. N. Bannerman, "Mapping continuous potentials to discrete forms," *J. Chem. Phys.*, **140**, 034105 (2014)

Performance Summary

- Force models such as the Lennard-Jones potential with a cut-off of 3.0 are too "soft" for stepped potentials to compete with at liquid densities (gas densities ED performs increasingly better).
- Although stepped potentials can approximate continuous systems, EDPD should not be applied to simulate continuous potentials (except for the rare-gas limit e.g., space vehicle re-entry).
- Time-stepping should also not be used for "hard" potentials as EDPD is significantly faster.
- EDPD is particularly fast when used on "coarse-grained" potentials, such as the hard-sphere or square well, and these are at the heart of many theoretical descriptions:

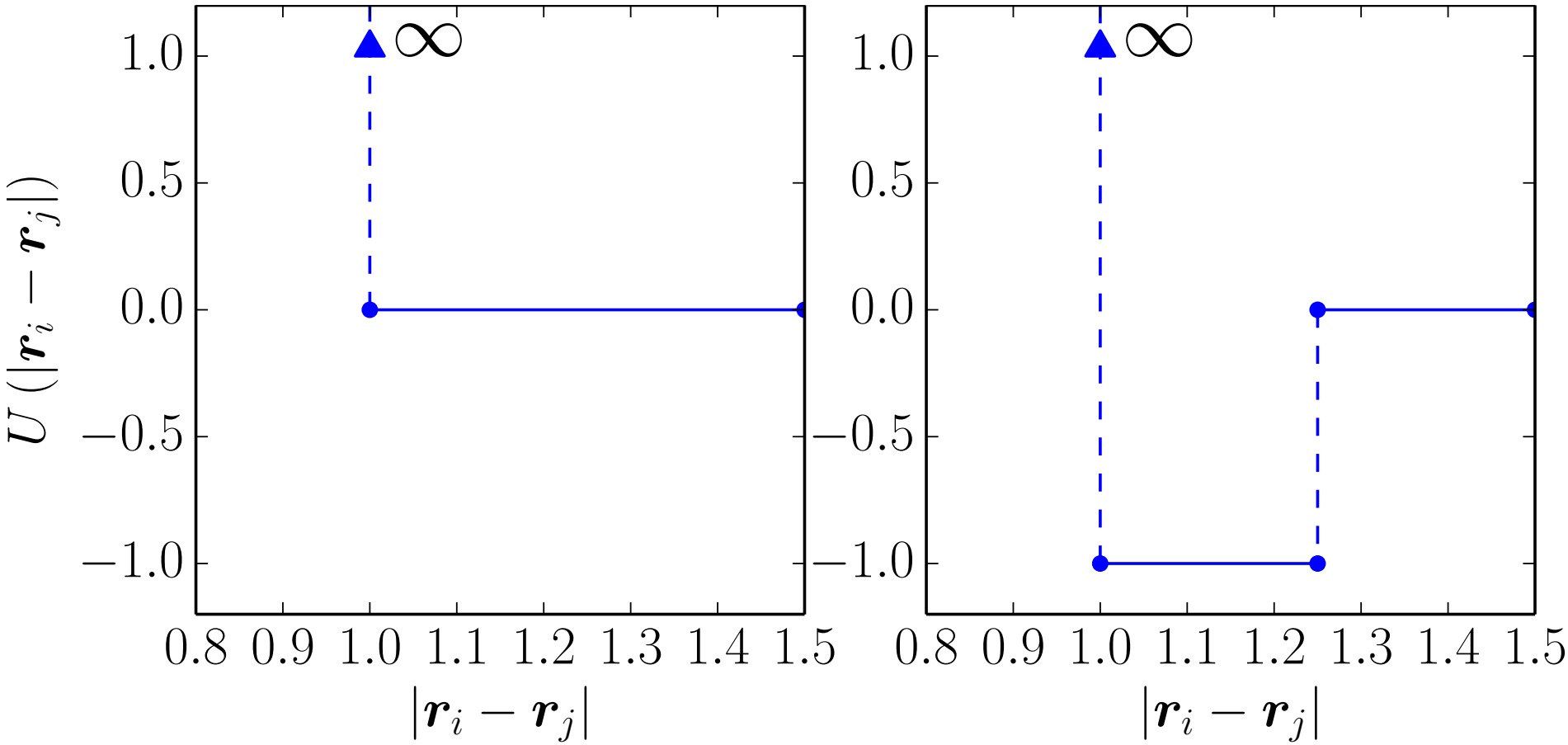


Figure: The hard sphere (left) and square-well (right) potentials. These are the most computationally efficient models available for simulation of non-ideal fluids, and the basis for much of the available theoretical molecular descriptions (kinetic theory, thermodynamic perturbation theory).

- It is clear that there are overlaps in application of time-stepping and event-driven methods.
- We can then compare time-stepping and event-driven simulation:

Time-Stepping	EDPD
✗ **Numerical** integration.	✓ **Analytical** solution of model dynamics.
✓ Simpler simulation algorithm.	? Complex simulation algorithm.
Spring example: 32 Lines of code.	Spring example: 90 Lines of code.
✓ Physical scaling laws directly compatible (e.g., Hookes law, or molecular dispersion $\propto r^{-6}$).	? Force models/Potentials are tabluated.
✓ Many validated models available.	? Fewer validated models available.
? Faster for dense systems with complex or long-ranged potentials.	✓ Strong theoretical frameworks.
	? Faster for low density or coarse-grained models.

- With the recent advent of standard software packages (like DynamO), and the TPT approaches allowing the fitting of potentials to experimental data, these disadvantages of EDPD are being overcome.

Section Outline

Time-warp algorithm

Exact time averages

- In molecular dynamics, properties such as pressure, energy, stress, are measured by taking a time average:

$$\langle A \rangle_t = t^{-1} \int_0^t A(\tau)\,\mathrm{d}t$$

  where $A(\tau)$ is the value of the property at a time τ.
- In time-stepping simulation, this integral is approximated by regular sampling.
- Some properties are sampled every timestep (stress/pressure and energy). More expensive properties are sampled at larger intervals or in post processing (e.g., Gromacs and its trajectory file).
- For discrete models many properties do not change between events, therefore the integral may be calculated exactly

$$\int_0^t A(\tau)\,\mathrm{d}t = \sum_a \Delta t_a\, A_a^-$$

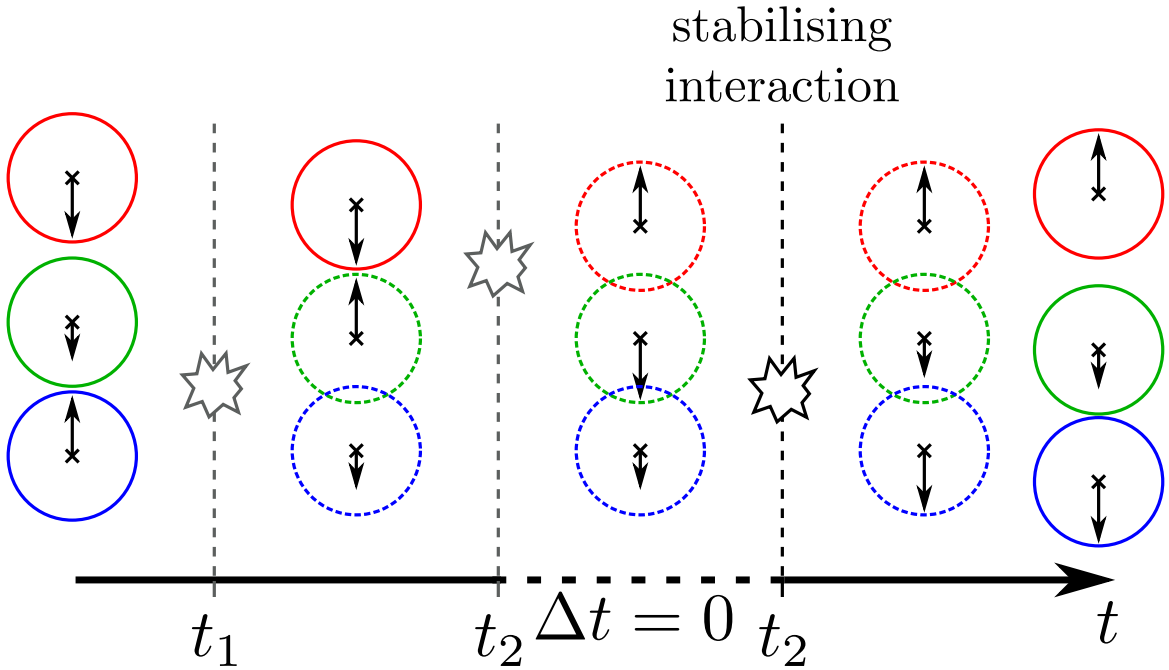
  where Δt_a is the free streaming time before event a, and A_a^- is the property just before it.

Exact time averages

- This is **required** to evaluate properties which are a function of the (impulsive) forces: pressure, viscosity, thermal conductivity.
- However, all properties which can be evaluated this way are, as it is computationally cheap and more accurate.
- This is why DynamO has an **output plugin** architecture, and many properties must be collected during a simulation rather than afterwards.

Stable algorithm and Magnet

- Although event-driven dynamics is analytic, it is extremely sensitive to round-off error.
- For example, round-off error in event times causes 50% of hard sphere collisions to overlap slightly: $U \to \infty$!
- DynamO uses "stabilising" interactions to ensure the rare (10^{-9}) but critical "triple events" (see below) are recoverable[1] .



[1] M. N. Bannerman, S. Strobl, A. Formella, and T. Poschel, "Stable algorithm for event detection in event-driven particle dynamics," *Comp. Part. Mech.*, **1**, 191-198 (2014)