An Introduction to N-body simulations

Many body/N-body methods

Toni Collis
EPCC
toni.collis@ed.ac.uk

Overview

- What are N-body simulations and what are they used for?
- The basics behind setting up a simulation
- Forces
- Discretisation and time-steps
- Energy conservation

What is an N-body simulation?

- N-body simulations approximate the motion of particles in a dynamic system
- Interaction of particles through physical forces (e.g. gravity, electrostatics)
- Particle-particle interactions (or n-body) are a general class of algorithms with applications ranging from chemistry to astrophysics

Star formation: Density

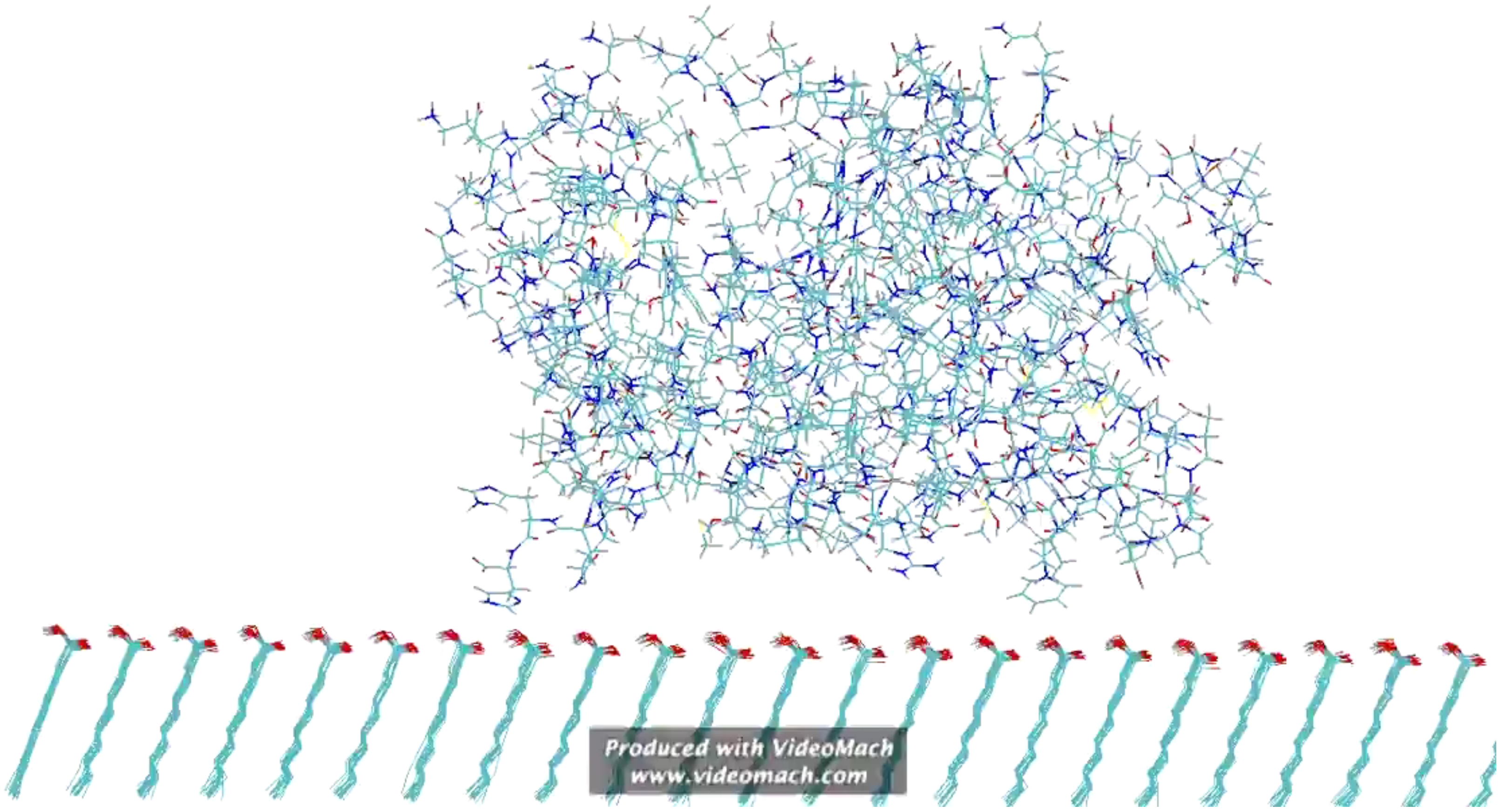
Matthew Bate – University of Exeter

Star formation : Temperature

Matthew Bate – University of Exeter

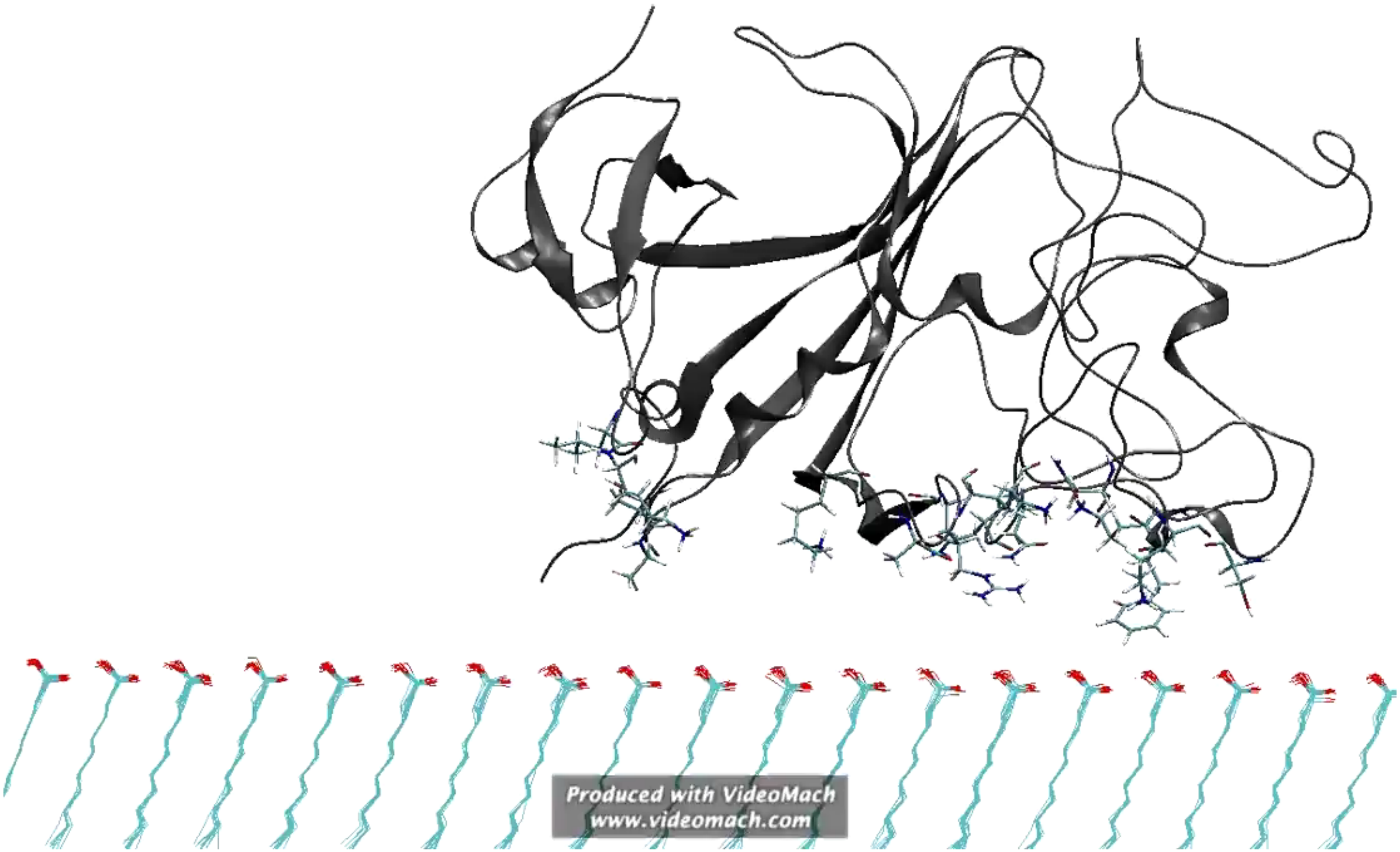
Molecular Simulation

Fibrinogen Protein Adsorption on Hydrophilic Surface
Agashe M., Raut V., Stuart S.J., and Latour R.A., Molecular simulation to characterize the adsorption behaviour of a fibrinogen gamma-chain fragment, Langmuir, 21: 1103-1117 (2005)



Molecular Simulation

Fibrinogen Protein Adsorption on Hydrophilic Surface
Agashe M., Raut V., Stuart S.J., and Latour R.A., Molecular simulation to characterize the adsorption behavior of a fibrinogen gamma-chain fragment, Langmuir, 21: 1103-1117 (2005)



Real problems are difficult to solve

- A simple analytical problem: apple falling
  - Gravity
    - Newton's universal law of gravitation
    - Newton's second law of motion
  - Simple solution
    - All objects fall at the same rate

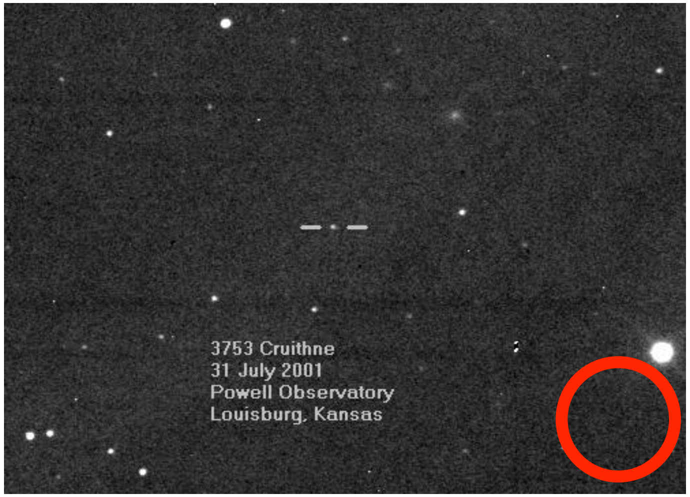
- Most real problems are too complex for an analytical solution

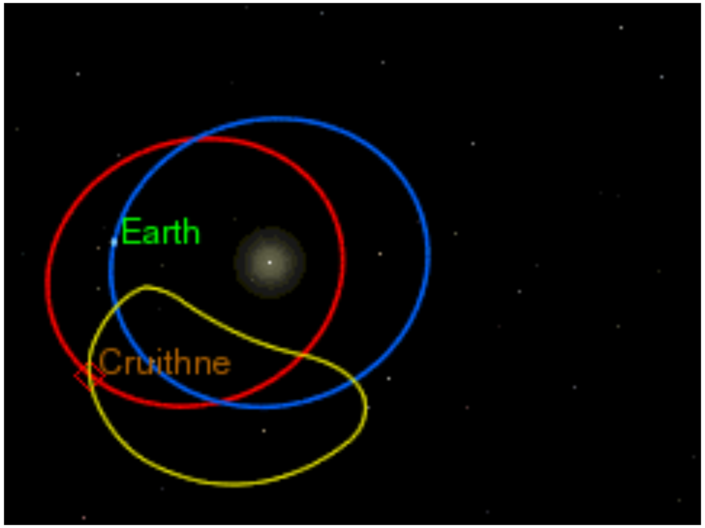
  - How many moons does the Earth have?

Earth's Second Moon

3575 Cruithne

- D. Waldron 10 Oct 1986
- Near Earth Asteroid
- 5Km diameter
- Orbits sun in 1:1 orbital resonance with Earth (i.e. co-orbiting)
- 770 years to complete full movement around the Earth





Orbit not determined until 1997

- Not a true moon, it orbits the Sun
- Almost same orbital period as the Earth
  - Orbit stable for several hundred years
- Figure shows orbit from Earth's perspective

Numerical solution

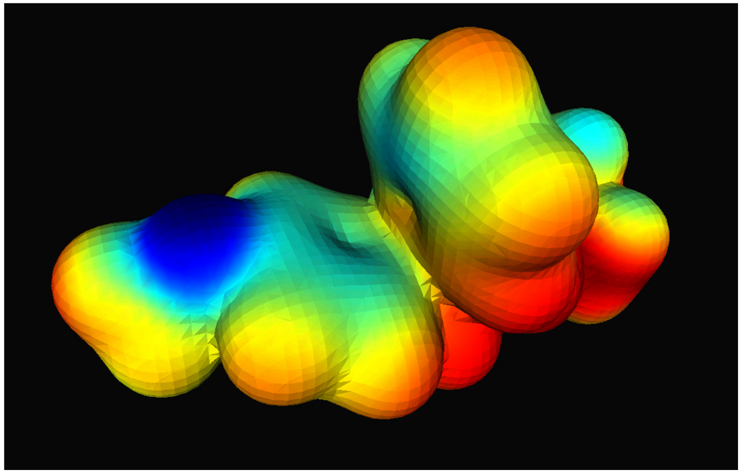
- Two body (apple and Earth)
  - Can solve analytically

- Three body problem with no symmetry
  - Mass of: Sun (10^{30}Kg) >> Earth (10^{24}Kg) >> Cruithne (10^{14}Kg)
  - No analytic solution
  - Horseshoe or “bean shaped” orbits typical
- Actual calculation included all planets except Pluto in simulation
  - Many body problem

- Not all problems are tractable even on latest computers

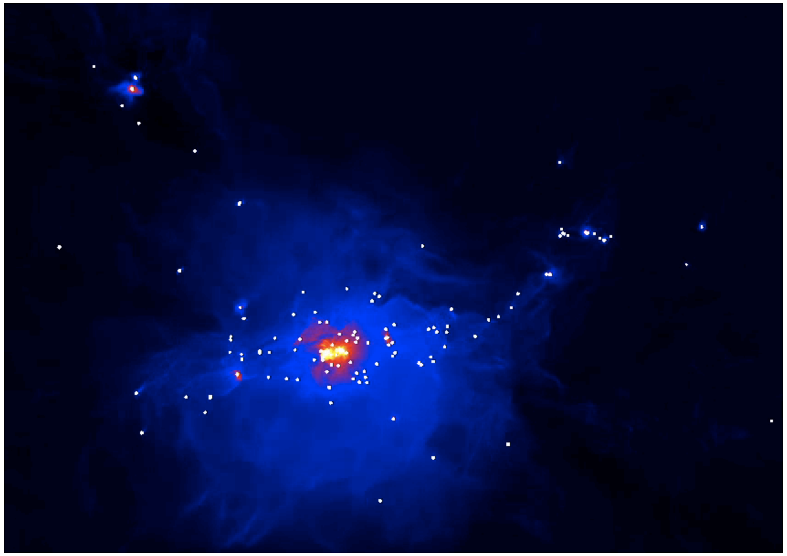
Applications which use N-body

- Molecular dynamics - Chemistry
  - Vasp
  - CASTEP
  - CP2K
  - NAMD



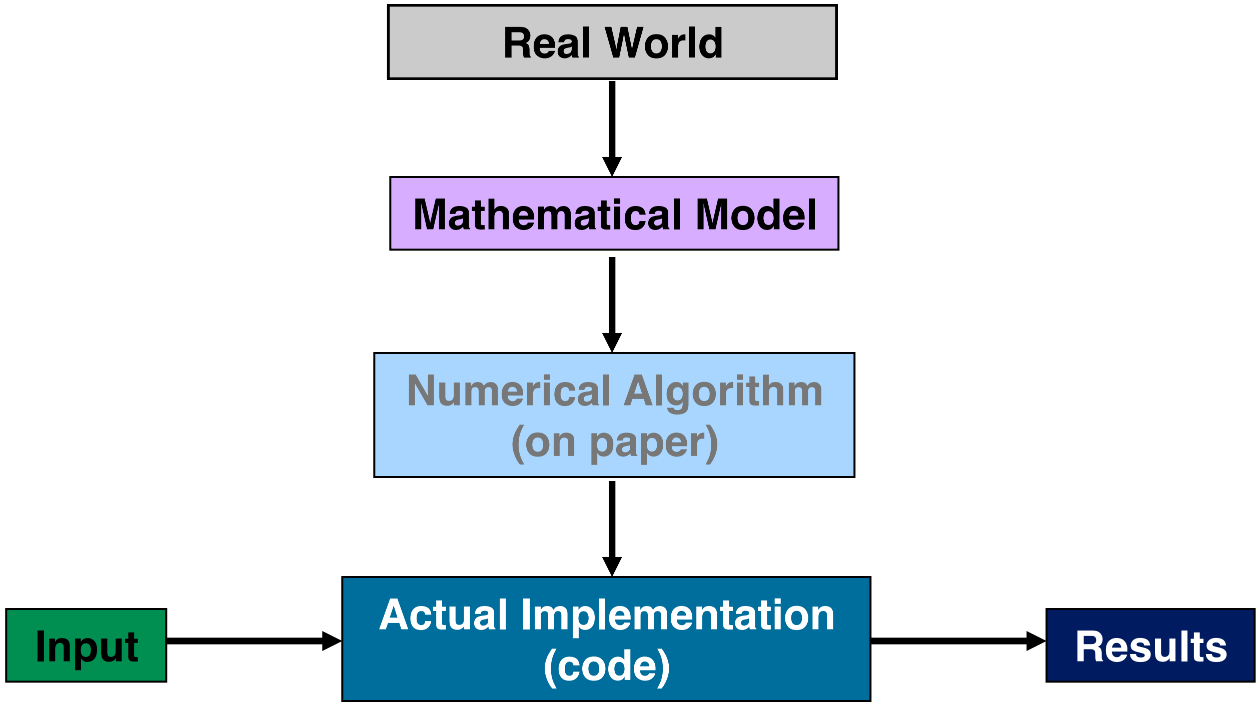
credit RPI/Curt Breneman

- Orbital dynamics- Astrophysics
  - Star formation
  - galaxy formation
  - cosmology



Matthew Bate, U. Exeter

Sources of uncertainty



Questions

- How would you simulate an apple falling under Earth's gravity?
  - Convert into a numerical problem
  - What is the force?
  - What is the input?
  - Initial conditions?
  - Discretisation?
  - How would you verify the result?

Questions

- How would you simulate an apple falling under Earth's gravity?
  - Convert into a numerical problem
    - F=ma
  - What is the force?
  - What is the input?
  - Initial conditions?
  - Discretisation?
  - How would you verify the result?

Gravitational force

- Newton's Universal Law of Gravitation
  - Force between two massive bodies (in vector form)

Gravitational Constant

Masses of the two bodies

$$\vec{F}_{ij} = G\frac{m_i m_i}{|\vec{r_i} - \vec{r_j}|^2} \times \frac{\vec{r_i} - \vec{r_j}}{|\vec{r_i} - \vec{r_j}|}$$

Unit vector in direction of resultant force

Separation of the two bodies, squared

$$\vec{F}_{ij} = G\frac{m_i m_j}{|\vec{r_i} - \vec{r_j}|^3} \left(\vec{r_i} - \vec{r_j}\right)$$

Questions

- How would you simulate an apple falling under Earth's gravity?
  - *Convert into a numerical problem*
  - *What is the force?*
  - What is the input?
  - Initial conditions?
    - Position of the atom and the Earth
    - Can approximate that the Earth is not actually a particle
    - Assume that an apple is a point particle
  - Discretisation?
  - How would you verify the result?

Questions

- How would you simulate an apple falling under Earth's gravity?
  - *Convert into a numerical problem*
  - *What is the force?*
  - *What is the input?*
  - *Initial conditions?*
  - Discretisation?

  - How would you verify the result?

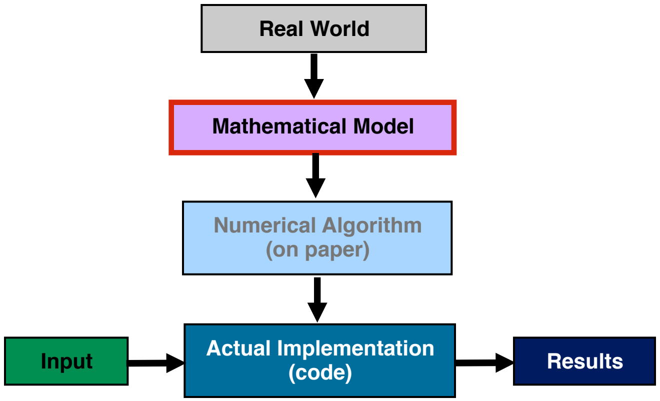
Classical mechanics

- Newton's Laws on Motion:

$$\vec{F} = m\vec{a} \qquad \vec{v} = \frac{\partial \vec{x}}{\partial t}$$

$$\vec{a} = \frac{\partial \vec{v}}{\partial t} = \frac{\partial^2 \vec{x}}{\partial t^2}$$

- Given:
  - Some space with appropriate boundary conditions,
  - A set of initial positions for a number of particles.
  - Some interaction between them, i.e. a force calculation.
- We can
  - Estimate F(x) ⇒ a ⇒ v ⇒ x ⇒ F(x) and repeat......
  - Thus simulating the motion of the particles over time.

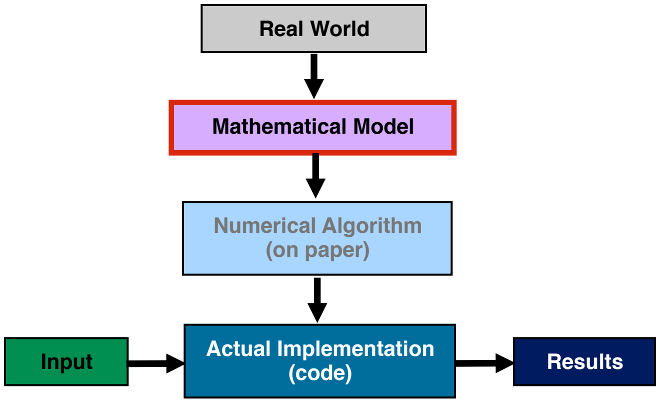


Questions

- How would you simulate an apple falling under Earth's gravity?
  - *Convert into a numerical problem*
  - *What is the force?*
  - *What is the input?*
  - *Initial conditions?*
  - *Discretisation?*
  - How would you verify the result?
    - Compare with analytical solution
    - For complicated systems with no analytical solution need to compare with experimental data (often complicated!)

A different system: 3575 Cruithne

- What to simulate for Earth's second moon?
  - Use Newtonian gravitation, not general relativity
  - No significant error, much simpler to calculate
  - Newtonian gravitation gets the advance of the perihelion of Mercury wrong
- Every massive body in the Universe contributes
  - Only include significant ones
  - Sun, Earth, Moon, Mercury, Venus, Mars, Jupiter, Saturn, Uranus and Neptune
  - Model truncates reality
  - Introduces a systematic uncertainty
  - Only accurate to size of largest omission



Interactions

- If each person in this room interacts pair-wise with each other person, how many interactions are there between us?
  - How many calculations would you need to perform to calculate this?

Aside- O(N) notation

- Who has heard of O(N) notation?
- What is the limiting behaviour of the following algorithm and how would you parallelise it?

```
for (i=0;i<N;i++){
      for (j=0;j<N;j++){
            for (k=0; k<N; k++){
                  c[i][j] += a[i][k]*b[k][j];
            }
      }
}
```

Asymptotic analysis

- Asymptotic ("big-O") notation captures this idea as "upper", "lower", and "tight" bounds.
    - $f(n) = O(g(n)) \Leftrightarrow \exists c > 0, n0$ such that $f(n) \leq c\, g(n)$, $\forall n > n0$ ("**no more than**")
    - $f(n) = \Omega(g(n)) \Leftrightarrow g(n) = O(f(n))$ ("**at least**")
    - $f(n) = \Theta(g(n)) \Leftrightarrow$ both $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$ ("**roughly**")

Note that we are throwing away constant (c) factors! In practice these are sometimes crucial. Asymptotics are useful as (precisely defined) rules of thumb.

Aside- O(N) notation

For N-body simulation the naive calculation is

```
for (i=0;i<N;i++){
      for (j=0;j<N;j++){
            if (i /= j) {
                  force[i] += force[i][j];
            }
      }
}
```

This scales as $O(N^2)$

Updating Positions

- Particles interact via pair-wise forces
  - sum up the forces
  - update positions using Newton's Equations

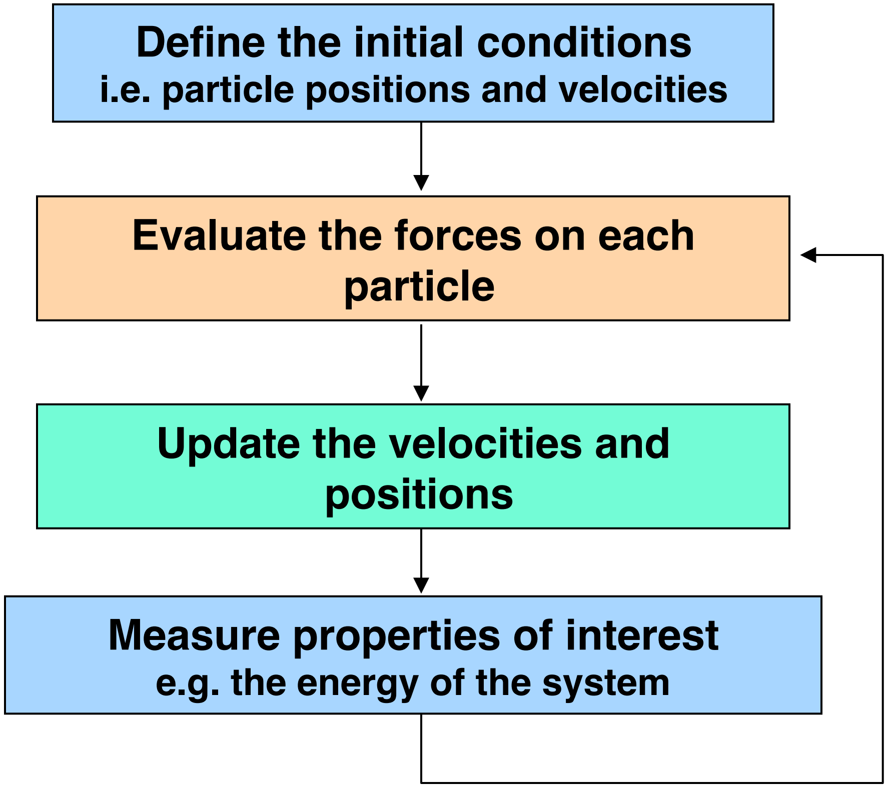
$$\vec{F} = m\vec{a}$$

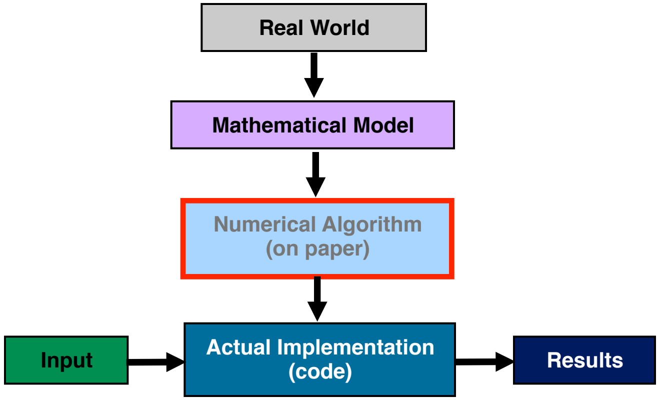
  - many ways of doing this with varying stability and accuracy
- Calculating forces is the most costly part
  - have up to $N(N\text{-}1)/2$ pairs for N particles
  - force may be gravity, electromagnetism, forces between atoms in a molecule, contact friction,
- Such simulations often called Molecular Dynamics
  - abbreviated to MD

Recall Gauss and series summation
Force calculation complexity

$$\mathcal{O}(N^2)$$

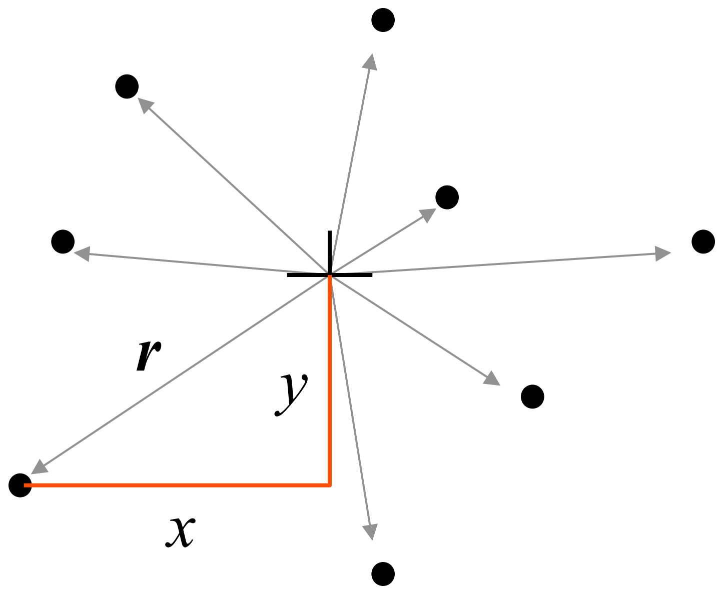
Molecular Dynamics

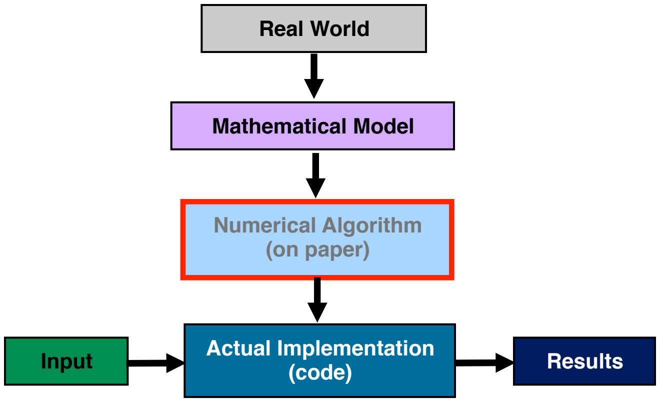




The Space: Positions

- Positions stored as arrays of real numbers
  - e.g. in 2-D, vector position ***r*** of i = 1 to N particles stored as x[i] and y[i] displacements:

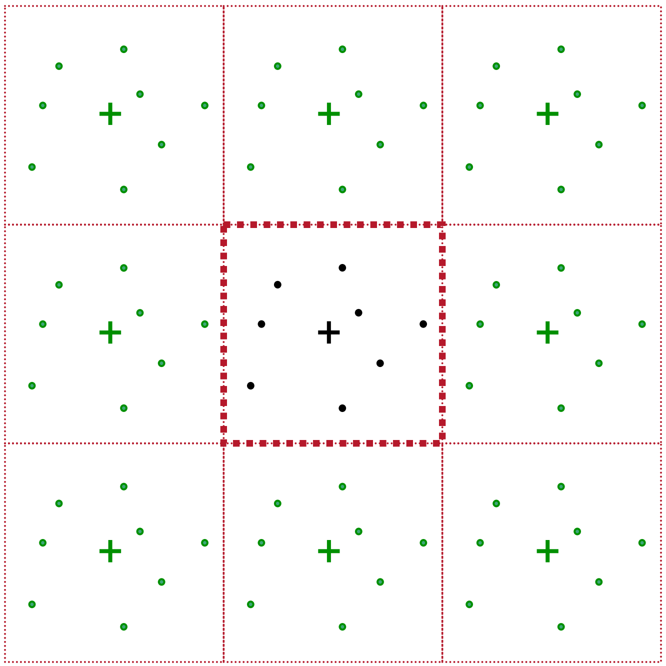




The Space: Boundaries

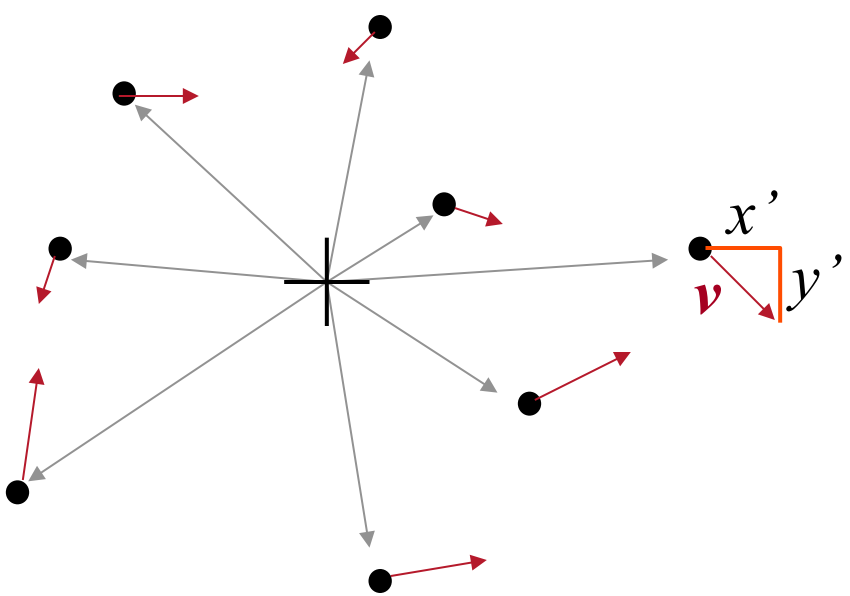
- Examples
  - open:
    - particles can go anywhere.

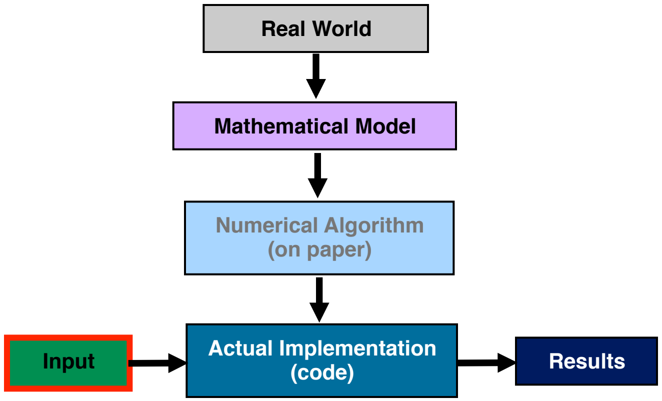
  - periodic:
    - particles stay in a fixed volume, by wrapping around the edges.
    - The system ‘sees’ images of itself tiled out in all directions.



Initial Conditions

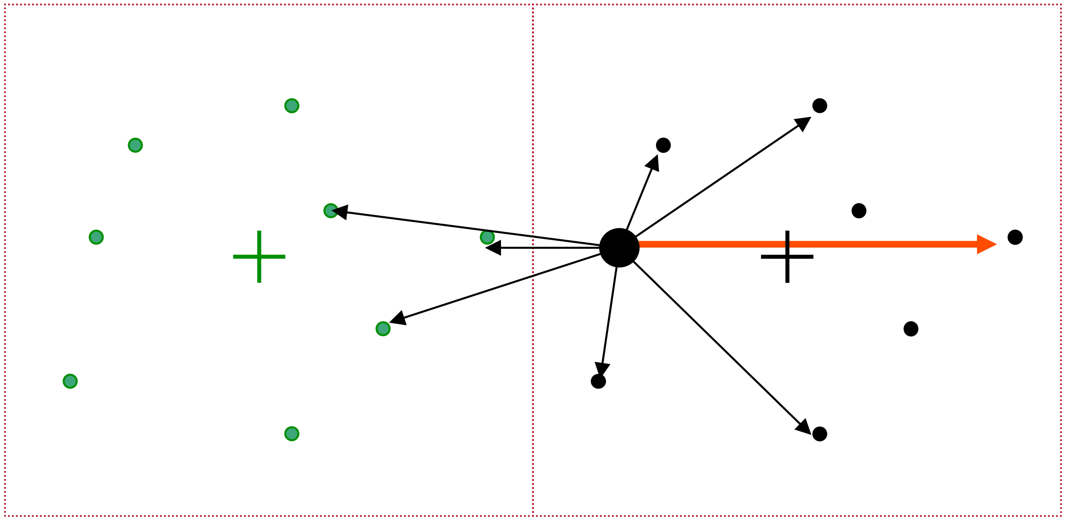
- We need initial positions and velocities
  - position ***r*** stored as x and y displacements,
    - e.g. miles, metres.
  - velocity ***v*** stored as x' and y' "speeds":
    - units of displacement per unit time, e.g. mph, m/s.





The Total Force

- For each particle, numbered *i* from 1 to *N*
  – we add up all of the forces acting upon that particle due to all of the other particles.
    – when using periodic boundary conditions, we usually use the nearest image of the other particles.



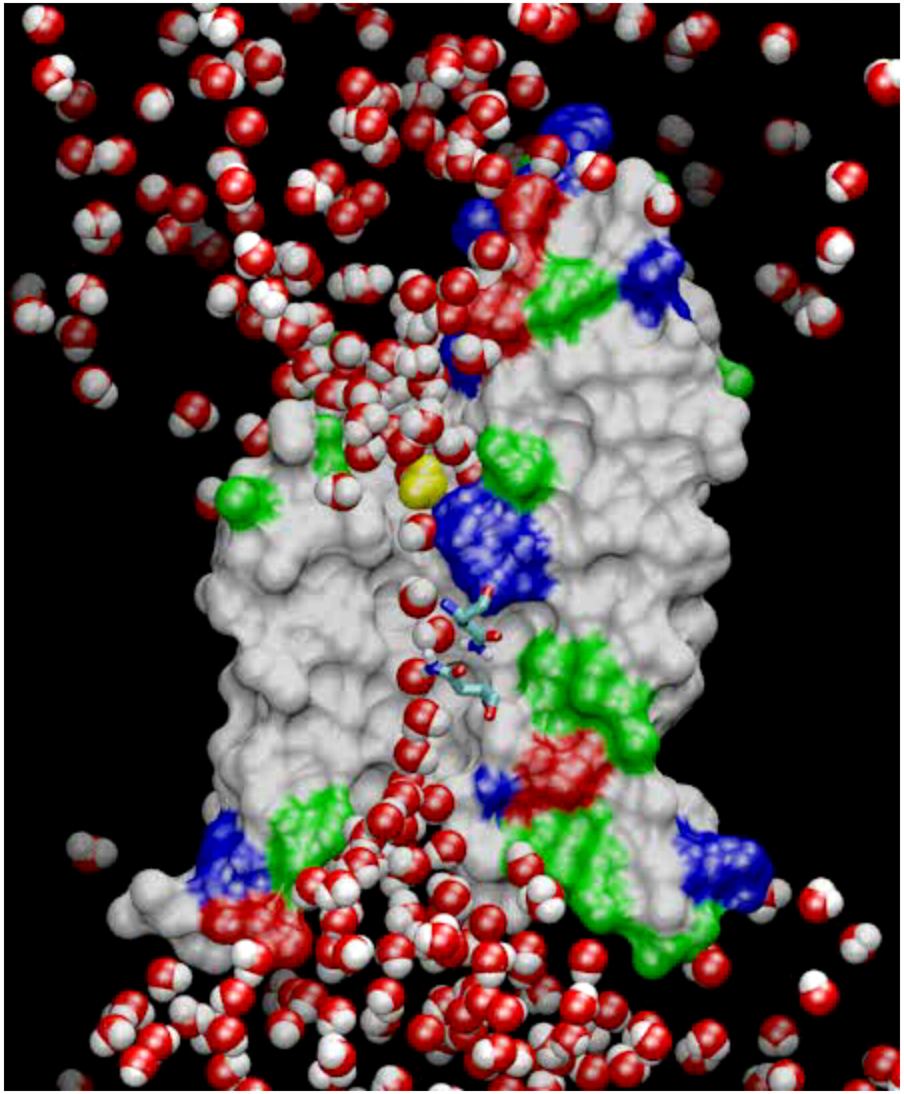
  – n.b. particles should not interact directly with themselves.

What is a ‘particle’?

In N-body simulations each ‘particle’ has physical significance

- Particles treated by the simulation may or may not correspond to physical objects which are particulate in nature
- N-body simulation of a star cluster: particle per star
- Molecular simulation: each atom is treated as a particle
  - N.B. *ab initio* Molecular simulation treats each electron as a particle but uses quantum mechanics instead of classical forces
- N-body simulation of a gas cloud:
  - Atom by atom simulation would require 10^{23} particles for each mole of material
  - Instead use larger quantity of gas for each particle (often implemented using Smoothed Particle Hydrodynamics)

Water permeation in an aquaporin



Tajkhorshid, E., et al. Science **296**:525-530 (2002)

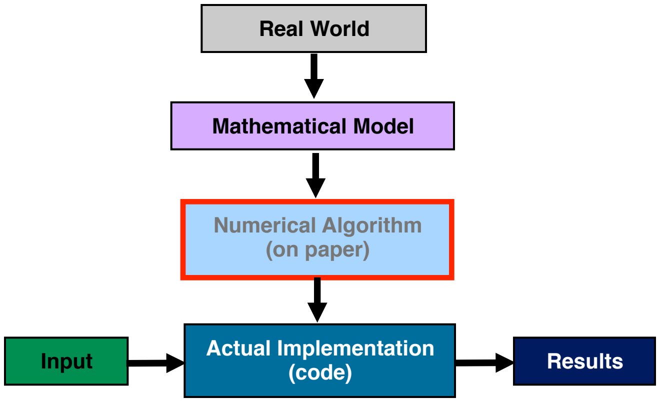
The Dynamics

- Given the forces,
  - we can determine the acceleration of each particle
  - to update the velocity and position

$$\vec{v}(t+\delta t) = \vec{v}(t) + \vec{a}(t) \times \delta t$$
$$\vec{x}(t+\delta t) = \vec{x}(t) + \vec{v}(t) \times \delta t$$

$$\vec{a}_i = \frac{\vec{F}_i}{m_i}$$
$$\vec{a}_i = \frac{d\vec{v}_i}{dt}$$
$$\vec{v}_i = \frac{d\vec{x}_i}{dt}$$

- Small incremental step = discretisation of the problem
- Then start all over again.



Taylor's Expansion

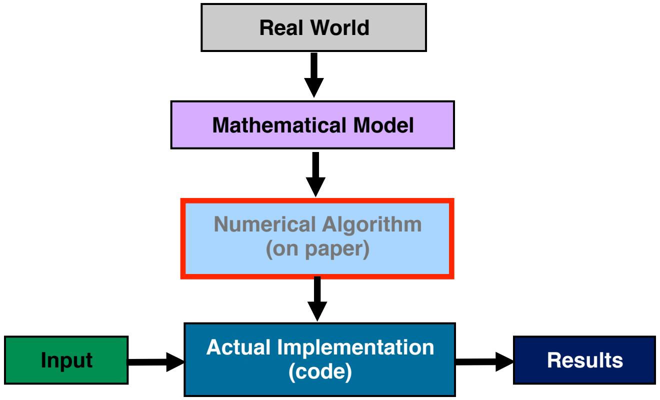
Non-examinable mathematics

- Express a function at a single point as a infinite series of its derivatives evaluated at that point.

$$f(x+h) = \sum_{n=0}^{\infty} \frac{d^n f(x)}{dx^n} \frac{h^n}{n!}$$

Brook Taylor
1685-1731

- Re-express dynamical equations using Taylor's Expansion
- Truncate to some chosen order



The Euler scheme – forward difference

- The simplest is Euler integration:
- Consider Taylor expansion to first order

$$f(x+h) = f(x) + f'(x)h + \mathcal{O}(h^2)$$

$$f'(x) \equiv f^{(1)} \equiv \frac{df(x)}{dx}$$

Ignore terms of this power (and higher)

Leonhard Euler
1707-1783

$$x \to t \quad h \to \Delta t$$

$$f'(x) \to \frac{f(t)}{dt}$$

differential of displacement is velocity
differential of velocity is acceleration

$$\vec{v}(t+\Delta t) = \vec{v}(t) + \vec{a}(t) \times \Delta t$$

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t) \times \Delta t$$

Euler equations

$$f(x+h) = f(x) + f\prime(x)h + f\prime\prime(x)h^3 + \mathcal{O}(h^4)$$

$$\begin{aligned}
\vec{v}(t+\delta t) &= \vec{v}(t) + \vec{v}\prime(t) \times \delta t + \vec{v}\prime\prime(t) \times \delta t^2 + \mathcal{O}(t^4) \\
&\sim \vec{v}(t) + \vec{v}\prime(t) \times \delta t + \mathcal{O}(t^2) \\
&= \vec{v}(t) + \vec{a}(t) \times \delta t + \mathcal{O}(t^2)
\end{aligned}$$

$$\begin{aligned}
\vec{x}(t+\delta t) &= \vec{x}(t) + \vec{x}\prime(t) \times \delta t + \vec{x}\prime\prime(t) \times \delta t^2 + \mathcal{O}(t^4) \\
&\sim \vec{x}(t) + \vec{x}\prime(t) \times \delta t + \mathcal{O}(t^2) \\
&= \vec{x}(t) + \vec{v}(t) \times \delta t + \mathcal{O}(t^2)
\end{aligned}$$

$$\vec{v}(t+\delta t) = \vec{v}(t) + \vec{a}(t) \times \delta t$$

$$\vec{x}(t+\delta t) = \vec{x}(t) + \vec{v}(t) \times \delta t$$

Error

- Using a truncated Taylor Expansion is an approximation.
- How would you quantify the error?

$$f(x+h) = f(x) + f\prime(x)h + f\prime\prime(x)h^3 + \mathcal{O}(h^4)$$

$$\begin{aligned}
\vec{v}(t+\delta t) &= \vec{v}(t) + \vec{v}\prime(t) \times \delta t + \vec{v}\prime\prime(t) \times \delta t^2 + \mathcal{O}(t^4) \\
&\sim \vec{v}(t) + \vec{v}\prime(t) \times \delta t + \mathcal{O}(t^2) \\
&= \vec{v}(t) + \vec{a}(t) \times \delta t + \mathcal{O}(t^2)
\end{aligned}$$

$$\begin{aligned}
\vec{x}(t+\delta t) &= \vec{x}(t) + \vec{x}\prime(t) \times \delta t + \vec{x}\prime\prime(t) \times \delta t^2 + \mathcal{O}(t^4) \\
&\sim \vec{x}(t) + \vec{x}\prime(t) \times \delta t + \mathcal{O}(t^2) \\
&= \vec{x}(t) + \vec{v}(t) \times \delta t + \mathcal{O}(t^2)
\end{aligned}$$

Euler and the error

$$\vec{v}(t+\delta t) = \vec{v}(t) + \vec{a}(t) \times \delta t$$

$$\vec{x}(t+\delta t) = \vec{x}(t) + \vec{v}(t) \times \delta t$$

- Finite difference equation looks very similar to the differential equation
  - What is the error?
- Truncated Taylor series at $\mathcal{O}(h^2) \sim \mathcal{O}(t^2)$
- Evaluate *v* and *r* at time τ in *N* steps of size $\delta(t) = \frac{1}{N}$
- The error for each step is $\mathcal{O}((t^2))$
- Total error is $N \times (\delta t)^2 = \frac{1}{\delta t}(\delta t)^2 = \delta t$

$\boxed{\text{Error is linear in step size}}$

Verlet – second order scheme

Loup Verlet
b. 1931

- A better, more stable and symmetric scheme
- Combines forward and backward Taylor expansions

$$\vec{x}(t+\delta t) = 2\vec{x}(t) - \vec{x}(t-\delta t) + \vec{a}(t) \times (\delta t)^2$$

The velocities do not appear explicitly, but can be found using $\vec{v}(t+\delta t) = \dfrac{x(t+\delta t) - r(t-\delta t)}{2\delta t}$

Discretisation errors:

Positions are correct to $\mathcal{O}((\delta t)^4)$

Velocities are correct only to $\mathcal{O}((\delta t)^2)$

This algorithm is not self starting

Requires one Euler step to be performed first

Other forms exist to address these shortcomings:

Leapfrog and Velocity Verlet

Leapfrog – second order scheme

- Uses half-step velocities:

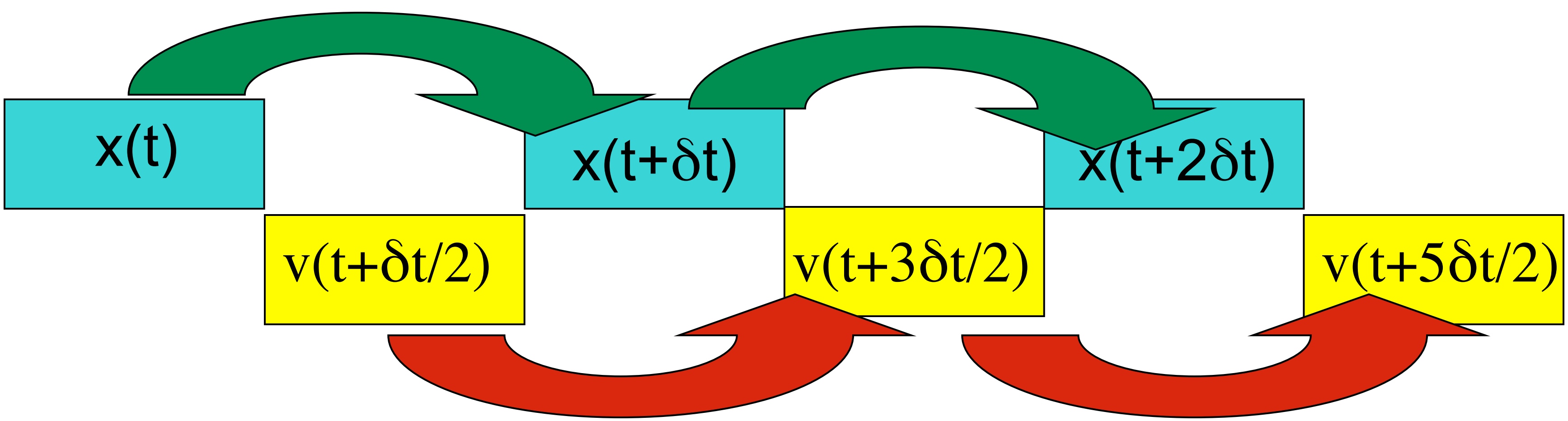
$$\vec{x}(t+\delta t) = \vec{x}(t) + \vec{v}(t + {}^1\!/\!{}_2\delta t) \times \delta t$$

$$\vec{v}(t + {}^1\!/\!{}_2\delta t) = \vec{v}(t - {}^1\!/\!{}_2\delta t) + \vec{a}(t) \times \delta t$$

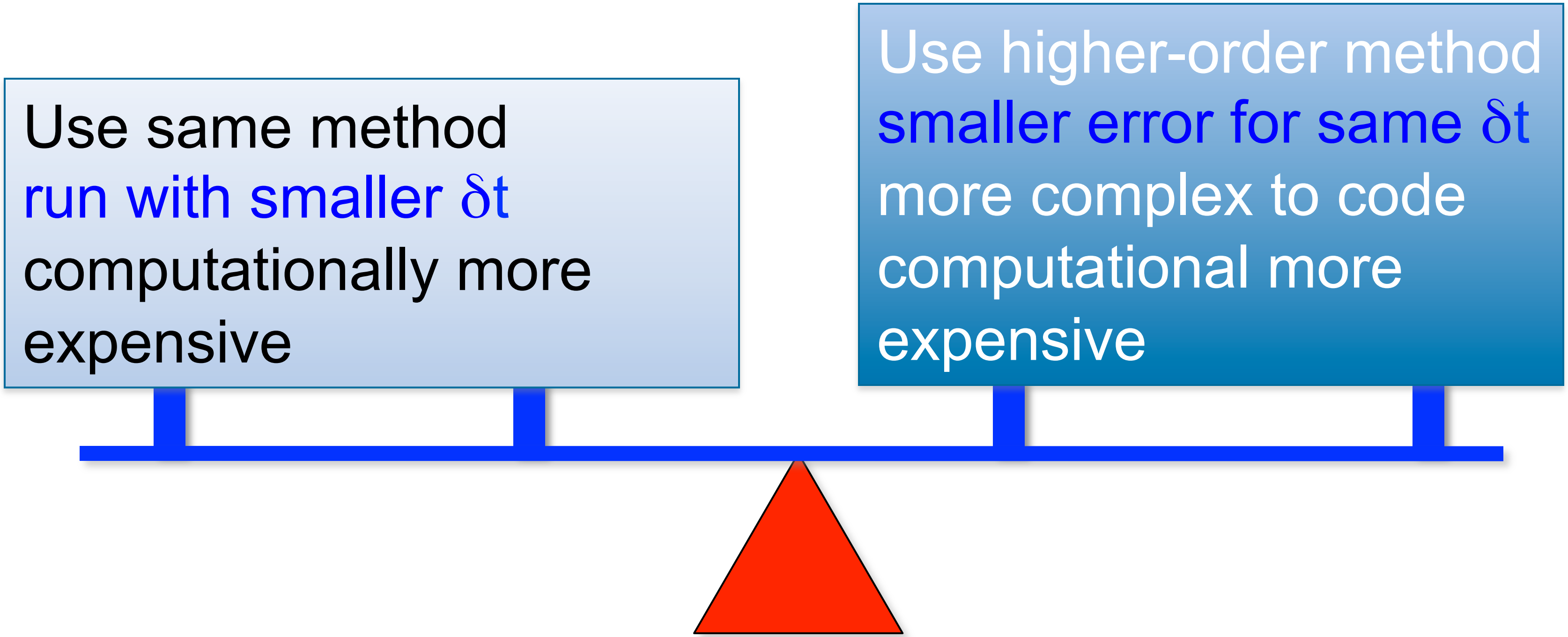
- Combines forward and backward 2nd order Taylor expansion
  - both displacement and velocity
- Error for each step is $\mathcal{O}((\delta t)^3)$

Algorithm is not self-starting: Require Euler 1st step

N steps of size δt -> Total error is O((δ)^2))



Reducing the error



- A huge variety of schemes exist
  - some may even be adaptive and choose Δt automatically
  - beyond the scope of this course
  - Earth's 2nd moon calculation uses high order integrator scheme

Question

What are the other sources of error in a simulation?

- Algorithm (Verlet etc.)
- Measurements of experiment that simulation is being compared with
- Inaccurate theory (e.g. Newtonian mechanics)
- Time-step size
- Limitations of artificial boxsize

Checking Correctness of Solution

- This is a real issue!
- May be able to monitor special quantities
  - energy must be conserved
  - momentum must be conserved
    - useful checks but not really enough
- Qualitative answer should not depend on step size
  - Multiple simulations to confirm answer

Checks & Sums

- Conservation of energy

$$E_K(t) = \sum_i \frac{1}{2} m_i \vec{v}_i(t)$$

$$E_P(t) = \sum_{\langle i,j \rangle} U(|\vec{r}_j(t) - \vec{r}_i(t)|)$$

$$E_T = E_P(t) + E_K(t) = C$$

Constant: Not time dependant

Similarly Conservation of linear momentum

$$\vec{P} = \sum_i m_i \vec{v}_i(t)$$

Linear momentum should also be conserved in each dimension individually, e.g. *x, y, z*