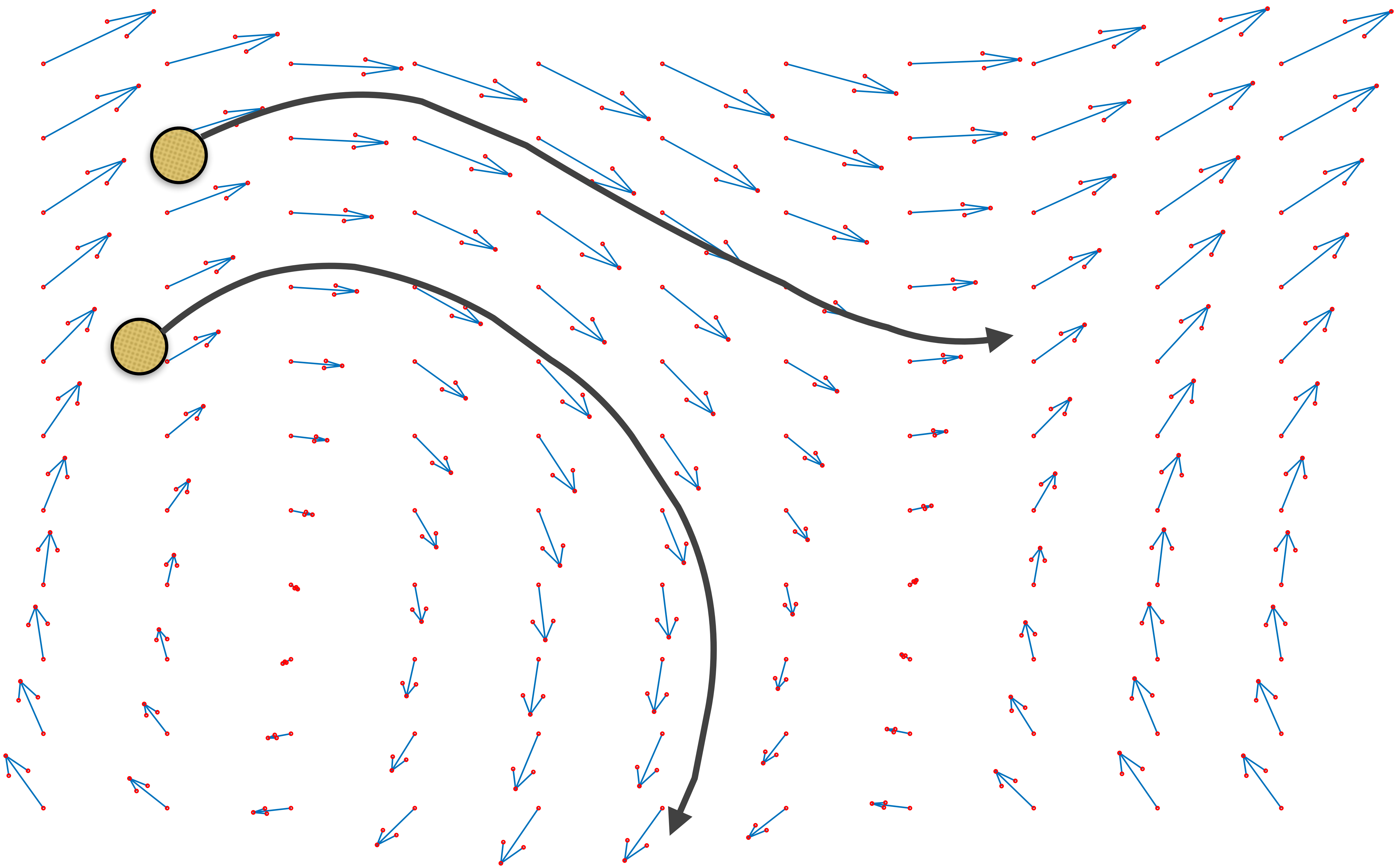
I. A Simple Start: Particle Dynamics

slides: Adam Bargteil

Let’s jump right in and consider the problem of tracing a particle through a velocity field



Initial Value Problem

$$\mathbf{x}_p(0) = \mathbf{x}_0$$

$$\frac{d\mathbf{x}_p(t)}{dt} = \mathbf{v}(\mathbf{x}_p, t)$$

Change $\longrightarrow$ Difference

First-order Ordinary Differential Equation

Initial Value Problem

$$\mathbf{x}_p(0) = \mathbf{x}_0$$

$$\frac{d\mathbf{x}_p(t)}{dt} = \mathbf{v}(\mathbf{x}_p, t)$$

Simple

Powerful

Instructive

Euler’s Method

The Derivative

$$\frac{d\mathbf{x}_p(t)}{dt} = \lim_{\epsilon \to 0} \frac{\mathbf{x}_p(t+\epsilon) - \mathbf{x}_p(t)}{\epsilon}$$

$$\epsilon \longrightarrow \Delta t$$

$$\frac{d\mathbf{x}_p(t)}{dt} \approx \frac{\mathbf{x}_p(t+\Delta t) - \mathbf{x}_p(t)}{\Delta t}$$

Euler's Method

$$\frac{d\mathbf{x}_p(t)}{dt} \approx \frac{\mathbf{x}_p(t+\Delta t) - \mathbf{x}_p(t)}{\Delta t}$$

$$+$$

$$\frac{d\mathbf{x}_p(t)}{dt} = \mathbf{v}(\mathbf{x}_p, t)$$

$$\downarrow$$

$$\frac{\mathbf{x}_p(t+\Delta t) - \mathbf{x}_p(t)}{\Delta t} = \mathbf{v}(\mathbf{x}_p, t)$$

$$\downarrow$$

$$\mathbf{x}_p(t+\Delta t) = \mathbf{x}_p(t) + \Delta t \cdot \mathbf{v}(\mathbf{x}_p, t)$$

The Great Tradeoff

$$\frac{d\mathbf{x}_p(t)}{dt} \approx \frac{\mathbf{x}_p(t+\Delta t) - \mathbf{x}_p(t)}{\Delta t}$$

As Δt decreases

the approximation gets better

but

the computational cost increases

Let’s consider another problem

In the real world

$$\mathbf{f} = m\mathbf{a}$$

Another Initial Value Problem

$$\mathbf{x}_p(0) = \mathbf{x}_0$$

$$\frac{d^2\mathbf{x}_p(t)}{dt^2} = \frac{\mathbf{f}(\mathbf{x}_p, t)}{m_p}$$

Second-order Ordinary Differential Equation

Another Initial Value Problem

$$\mathbf{x}_p(0) = \mathbf{x}_0$$
$$\mathbf{v}_p(0) = \mathbf{v}_0$$

$$\frac{d\mathbf{x}_p(t)}{dt} = \mathbf{v}(\mathbf{x}_p, t)$$

$$\frac{d\mathbf{v}_p(t)}{dt} = \frac{\mathbf{f}(\mathbf{x}_p, t)}{m_p}$$

Coupled First-order Ordinary Differential Equations

Euler's Method (Again)

$$\mathbf{v}_p(t + \Delta t) = \mathbf{v}_p(t) + \Delta t \cdot \frac{\mathbf{f}(\mathbf{x}_p, t)}{m_p}$$

$$\mathbf{x}_p(t + \Delta t) = \mathbf{x}_p(t) + \Delta t \cdot \mathbf{v}_p(t + \Delta t)$$

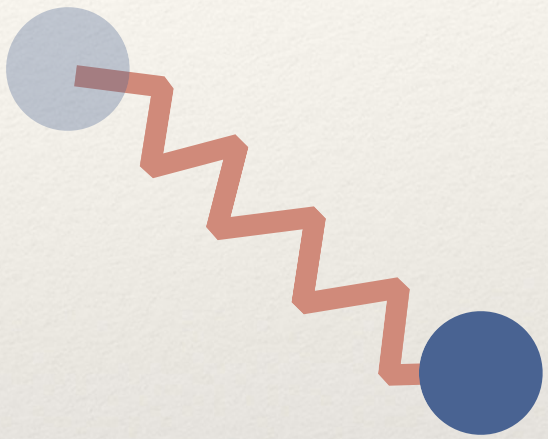
Symplectic Euler

```
struct Particle {
  double mass;
  Eigen::Vector3d pos, vel, frc;
};


foreach (p : particles) {
  p.frc = 0.0;
}

foreach (f : forces) {
  foreach (p : forces.affectedParticles) {
    p.frc += f.computeForce(p);
  }
}

foreach (p : particle) {
  p.vel += dt * p.frc / p.mass;
  p.pos += dt * p.vel;
}
```

Check our Karl Sim's *Particle Dreams*

Let’s Add Springs!

Springs



$$\mathbf{f} = -k\mathbf{d}$$

$$\mathbf{f}_p = -k\mathbf{x}_p$$

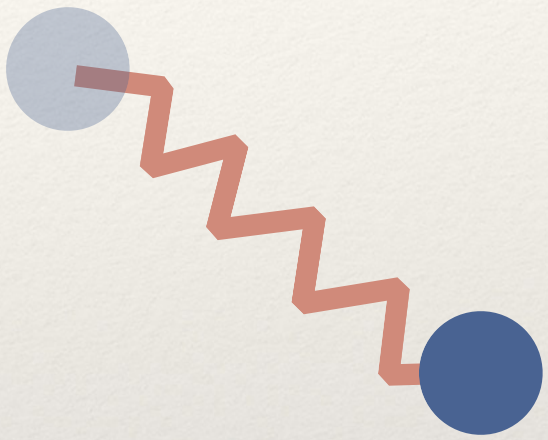
non-zero rest length?

$$\mathbf{f}_p = -k\left(\|\mathbf{x}_p\|\right)\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|}$$

$$\mathbf{f}_p = -k\left(\|\mathbf{x}_p\| - r\right)\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|}$$

Springs

$$\mathbf{f}_p = -k\left(\|\mathbf{x}_p\| - r\right)\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|}$$

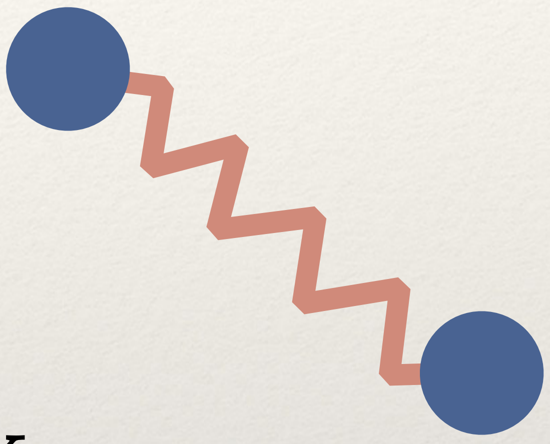


Strain $$\left(\frac{\|\mathbf{x}_p\|}{r} - 1\right)$$

$$\mathbf{f}_p = -k\left(\frac{\|\mathbf{x}_p\|}{r} - 1\right)\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|}$$

Springs

$$\mathbf{f}_p = -k\left(\frac{\|\mathbf{x}_p\|}{r} - 1\right)\frac{\mathbf{x}_p}{\|\mathbf{x}_p\|}$$



$$\mathbf{f}_p = k\left(\frac{\|\mathbf{x}_q - \mathbf{x}_p\|}{r} - 1\right)\frac{\mathbf{x}_q - \mathbf{x}_p}{\|\mathbf{x}_q - \mathbf{x}_p\|}$$

arbitrary connection?

$$\mathbf{f}_q = -\mathbf{f}_p$$

Damping

$$\mathbf{f}_p = k\left(\frac{\|\mathbf{x}_q - \mathbf{x}_p\|}{r} - 1\right)\frac{\mathbf{x}_q - \mathbf{x}_p}{\|\mathbf{x}_q - \mathbf{x}_p\|}$$

$$\mathbf{f}_p = k_d\left(\underbrace{\frac{\mathbf{v}_q - \mathbf{v}_p}{r}}_{\text{relative velocity}} \cdot \underbrace{\frac{\mathbf{x}_q - \mathbf{x}_p}{\|\mathbf{x}_q - \mathbf{x}_p\|}}_{\text{spring direction}}\right)\frac{\mathbf{x}_q - \mathbf{x}_p}{\|\mathbf{x}_q - \mathbf{x}_p\|}$$

$$\mathbf{f}_p = \left[k_s\left(\frac{\|\mathbf{x}_q - \mathbf{x}_p\|}{r} - 1\right) + k_d\left(\frac{(\mathbf{v}_q - \mathbf{v}_p)\cdot(\mathbf{x}_q - \mathbf{x}_p)}{r\|\mathbf{x}_q - \mathbf{x}_p\|}\right)\right]\frac{\mathbf{x}_q - \mathbf{x}_p}{\|\mathbf{x}_q - \mathbf{x}_p\|}$$

```
foreach (p : particles) {
  p.frc = 0.0;
}

foreach (s : springs) {
  Eigen::Vector3d d = particles[s->j].pos - particles[s->i].pos;
  double l = d.norm();
  Eigen::Vector3d v = particles[s->j].vel - particles[s->i].vel;
  Eigen::Vector3d frc = (params.k_s*((l / s->r) - 1.0) +
     params.k_d*(v.dot(d)/(l*s->r))) * (d/l);
  particles[s.i].frc += frc
  particles[s.j].frc -= frc
}

foreach (p : particle) {
  p.vel += dt * p.frc / p.mass;
  p.pos += dt * p.vel;
}
```