Cloth Simulation

Douwe van Gijn
Christian Knaapen

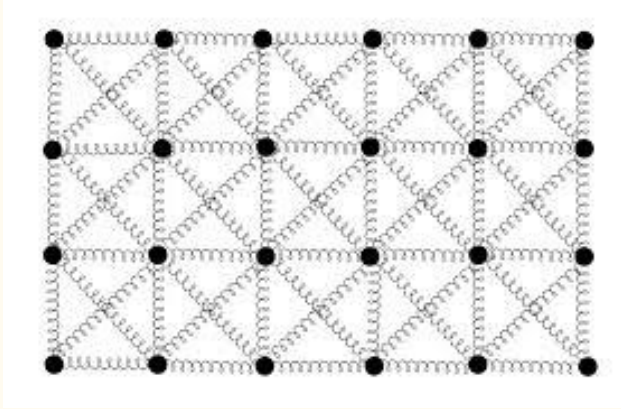
Mass-spring system

Grid of vertices connected with springs

$$F_{spring} = K_{spring}(|\vec{x}| - l_0)$$

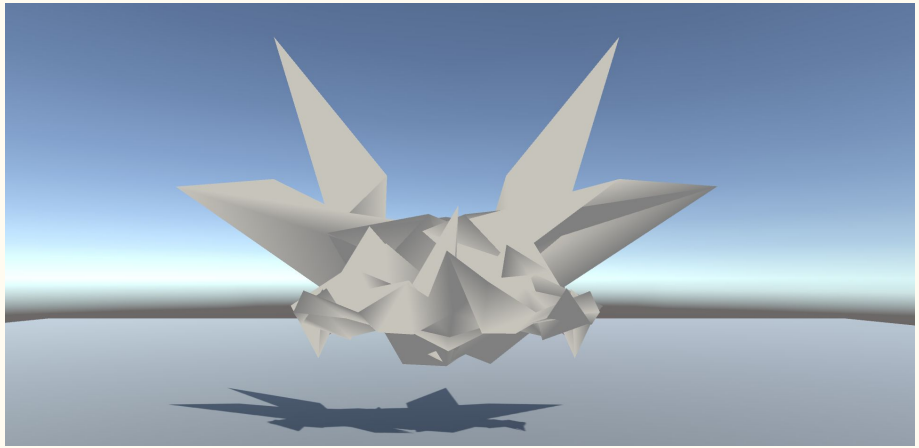
$$F_{damp} = K_{damp}\frac{\vec{v}\cdot\vec{x}}{|\vec{x}|}$$

$$\vec{F} = (F_{spring} + F_{damp})\frac{\vec{x}}{|\vec{x}|}$$



Explicit Euler integration

- Everything exploded
- Glitchy mess
- Infinite energy



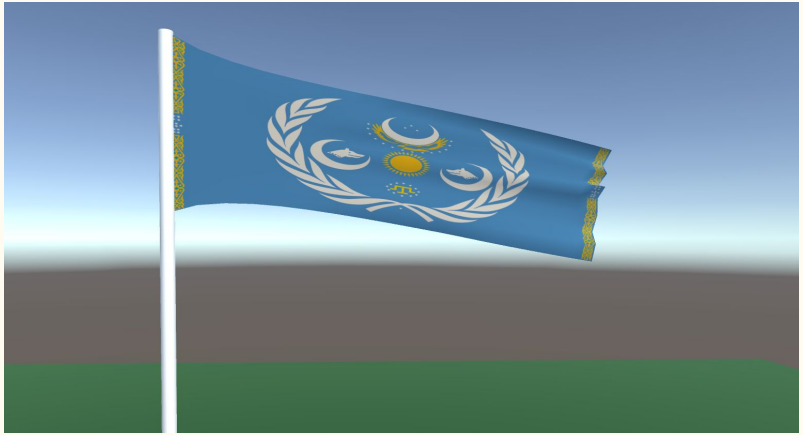
Verlet [vɛʁˈlɛ] integration

- Incredibly stable!
- 4th order accurate → as good as Runge-Kutta
- As cheap as Euler
- Does not have velocity terms

$$\vec{x}_{t+\Delta t} = \vec{x}_t + (\vec{x}_t - \vec{x}_{t-\Delta t}) + \vec{F}\frac{\Delta t^2}{m}$$

$$\vec{v} = \frac{\vec{x}_t - \vec{x}_{t-\Delta t}}{\Delta t}$$

Drag and Wind



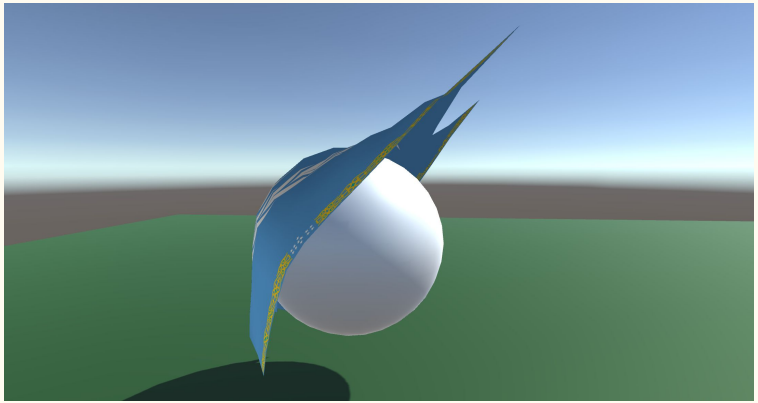
- Highly scientific
- Totally not a hack

$$\vec{F}_{drag} = -K_{drag}\vec{v}$$

$$\vec{F}_{wind} = \vec{K}_{wind}R_1R_2$$

- R_1: random number in range [0.2, 1], renewed every 30 frames
- R_2: random number in range [0.8, 1.2], renewed every frame

Collision



- Ground

$$x_y = \max(\varepsilon, x_y)$$

- Sphere
  - If inside sphere
  - epsilon: small number to prevent z-fighting and visual glitches

$$\vec{x} = \vec{x}_{sphere} + \frac{\vec{x} - \vec{x}_{sphere}}{|\vec{x} - \vec{x}_{sphere}|}(r + \varepsilon)$$

Polish

- Unity3D
  - Custom mesh
- Tweakable parameters
- Textures
- ‘Professional’ 3D modeling
- Decades of tweaking