Animation & Simulation

He Wang (王鹤)

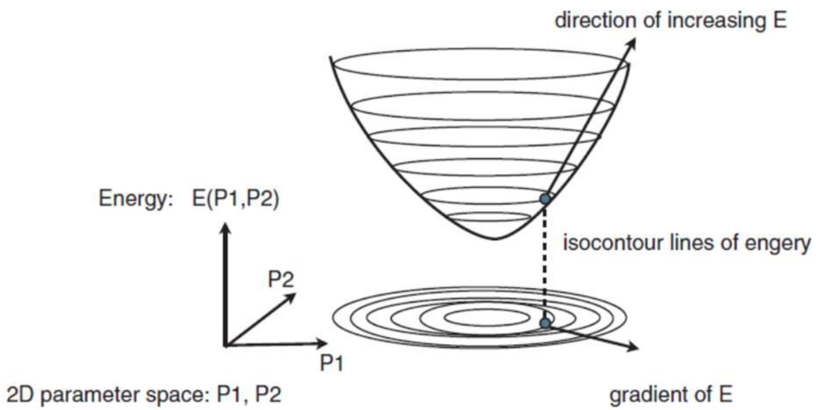
Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Strictly enforced-hard constraints (joint angles, penetrations, foot planting)
      - Numerically more challenging
      - The more, the more difficult to satisfy all
    - Better to satisfy-soft constraints (not too fast, not too rigid, etc)
      - Can be formed as energy minimisation problem, deviation causes non-zero energy

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Energy minimisation
      - Desired motions can be formed as non-negative smooth function $E(\psi)$
      - Find the minima, then constraints will be satisfied (fully or partially)

$$F(0) = \psi_0 \quad F(t_{i+1}) = F(t_i) - h\nabla E$$



Physically-based Animation

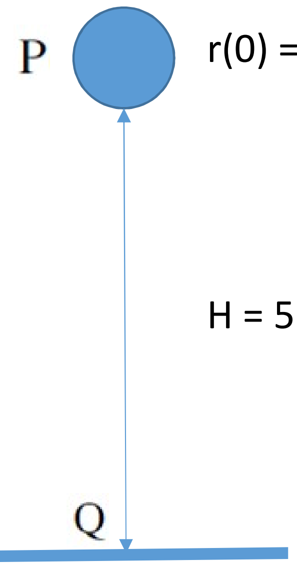
- Rigid body simulation
  - Enforcing soft and hard constraints
    - Energy minimisation

Initial distance 5m, initial velocity 0m/s, time step h=0.01s

$E = |\mathrm{P} - \mathrm{Q}|^2$ so F(0) = H

$$F(t_{i+1}) = F(t_i) - h\nabla E$$

What is F(0.01) = ?
F(0.02) = ?



r(0) = 5, v(0) = 0

H = 5m

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Energy minimisation
      - Constraints can be formed as non-negative smooth function $E(\psi)$
      - Find the minima, then constraints will be satisfied (fully or partially)
      - Three useful functions
        - $P(u, v)$ computes the positions given u and v
        - $N(u, v)$ computes the surface normal at u, v
        - $I(x)$ computes the signed distance to a surface

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Energy minimisation
      - Constraints can be formed as non-negative smooth function $E(\psi)$
      - Find the minima, then constraints will be satisfied (fully or partially)
      - Three useful functions

Point-to-fixed-point

$$E = |\mathrm{P}(u, v) - \mathrm{Q}|^2$$

Point-to-point

$$E = |\mathrm{P}^a(u_a, v_a) - \mathrm{P}^b(u_b, v_b)|^2$$

Point-to-point locally abutting

$$E = |\mathrm{P}^a(u_a, v_a) - \mathrm{P}^b(u_b, v_b)|^2 + \mathrm{N}^a(u_a, v_a) \cdot \mathrm{N}^b(u_b, v_b) + 1.0$$

Floating attachment

$$E = (\mathrm{I}^b(\mathrm{P}^a(u_a, v_a)))^2$$

Floating attachment locally abutting

$$E = (I^b(\mathrm{P}^a(u_a, v_a)))^2 + \mathrm{N}^a(u_a, v_a) \cdot \frac{\nabla I^b(\mathrm{P}^a(u_a, v_a))}{|\nabla I^b(\mathrm{P}^a(u_a, v_a))|} + 1.0$$

Physically-based Animation

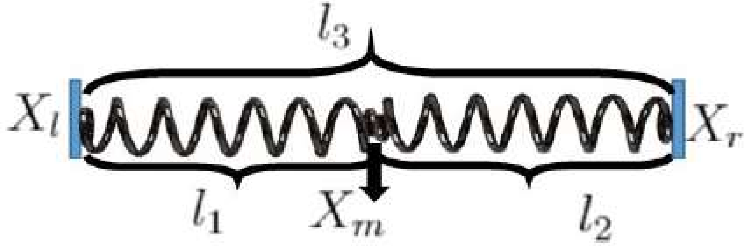
- Rigid body simulation
  - Enforcing soft and hard constraints
    - Energy minimisation
      - Constraints can be formed as non-negative smooth function $E(\psi)$
      - Find the minima, then constraints will be satisfied (fully or partially)
      - Three useful functions
      - Not hard constraints!

Physically-based Animation

- Rigid body simulation
    - Enforcing soft and hard constraints
        - Energy minimisation

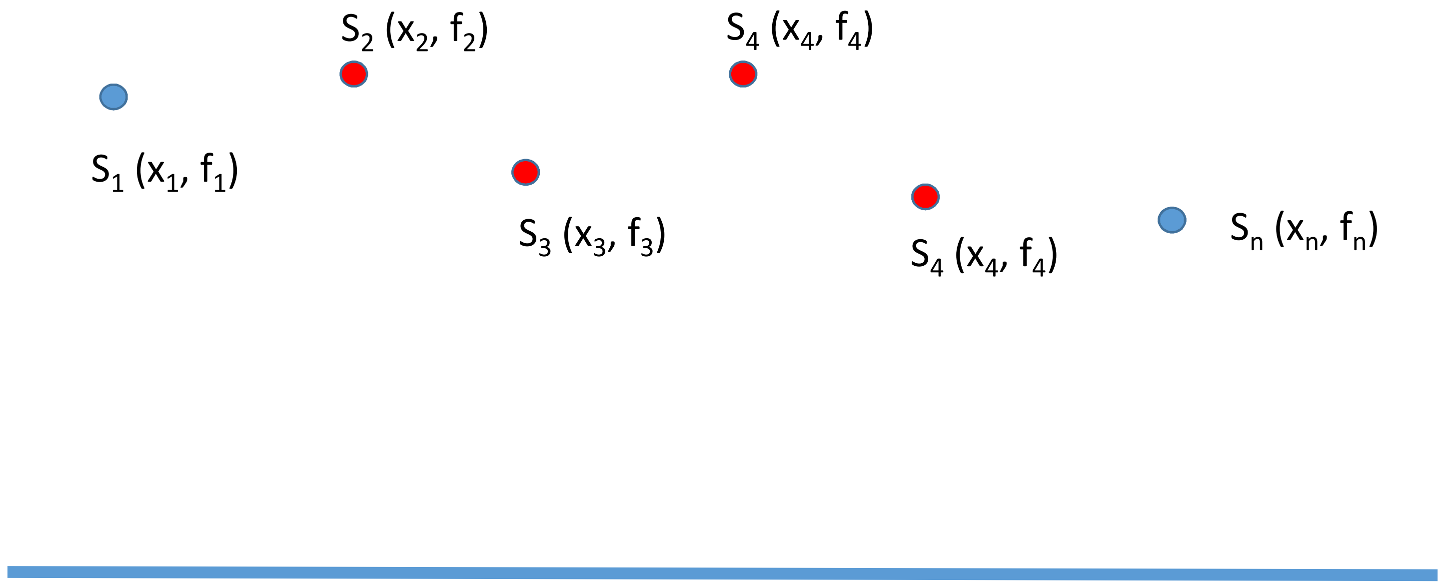
Mass-spring models can be used to simulate deformable objects. In a 1D scenario (Fig. 1), two springs are connected at X_m and are connected to two points at positions $X_l = 0$ and $X_r = 2$. The rest-lengths of the two springs are $l_1 = l_2 = 1$ and the distance between the two walls is $l_3 = l_1 + l_2 = 2$. The two stiffness coefficients of the two springs are $s_1 = s_2 = 1$. The only mass is a point mass (1 kilogram) attached to X_m. Gravity and the mass of the springs can be ignored. $X_m^0 = 0.5$ with $v^0 = 0$, where the superscript indicates the time step.

$$E = s(||p_2 - p_1|| - l)^2$$



Physically-based Animation

- Example



Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Space-time constraints Witkin and Kass (Siggraph 1988, The Luxo Lamp, 1986)

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Space-time constraints Witkin and Kass (Siggraph 1988, The Luxo Lamp)
      - Given a particle $m\ddot{\mathrm{x}}(t) - \mathrm{f}(t) - m\mathrm{g} = 0$
        - Given f, initial value problem, easy
        - Given initial and final position, solve for f

Dirichlet Boundary $\mathrm{x}(t_0) = \mathrm{a}$, $\mathrm{x}(t_1) = \mathrm{b}$

$$R = \int_{t_0}^{t_1} |\mathrm{f}|^2 \, dt$$

Is the energy consumption given by $|\mathrm{f}|^2$

Subject to

$$\mathrm{p}_i = m \frac{\mathrm{x}_{i+1} - 2\mathrm{x}_i + \mathrm{x}_{i-1}}{h^2} - \mathrm{f} - m\mathrm{g} = 0$$

$$c_a = |\mathrm{x}_1 - \mathrm{a}| = 0$$
$$c_b = |\mathrm{x}_n - \mathrm{b}| = 0$$

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Space-time constraints Witkin and Kass (Siggraph 1988, The Luxo Lamp
      - Given a particle $m\ddot{\mathrm{x}}(t) - \mathrm{f}(t) - m\mathrm{g} = 0$
        - Given f, initial value problem, easy
        - Given initial and final position, solve for f

Minimise $$R = \int_{t_0}^{t_1} |\mathrm{f}|^2 \, dt$$

Subject to $$\mathrm{p}_i = m \frac{\mathrm{x}_{i+1} - 2\mathrm{x}_i + \mathrm{x}_{i-1}}{h^2} - \mathrm{f} - m\mathrm{g} = 0$$

$$c_a = |\mathrm{x}_1 - \mathrm{a}| = 0$$
$$c_b = |\mathrm{x}_n - \mathrm{b}| = 0$$

S_j values are the x_i and f_i S_j values that minimize R subject to $C_i(S_j) = 0$

Physically-based Animation

- Rigid body simulation
  - Enforcing soft and hard constraints
    - Space-time constraints Witkin and Kass (Siggraph 1988, The Luxo Lamp)

Minimise $$R = \int_{t_0}^{t_1} |\mathrm{f}|^2 \, dt$$

Subject to
$$\mathrm{p}_i = m \frac{\mathrm{x}_{i+1} - 2\mathrm{x}_i + \mathrm{x}_{i-1}}{h^2} - \mathrm{f} - m\mathrm{g} = 0$$
$$c_a = |\mathrm{x}_1 - \mathrm{a}| = 0$$
$$c_b = |\mathrm{x}_n - \mathrm{b}| = 0$$

$$\mathrm{J}_{ij} = \frac{\partial C_i}{\partial S_j}$$

$$\mathrm{H}_{ij} = \frac{\partial^2 R}{\partial S_i \partial S_j}$$

Iteratively solve

$$-\frac{\partial}{\partial S_j}(R) = \sum_j H_{ij}\hat{S}_j$$ $\hat{S}_j$ minimises R irrespective to C

$$-C_i = \sum_j J_{ij}(\hat{S}_j + \tilde{S}_j)$$ $\tilde{S}_j$ drives Cs to zero and project $\hat{S}_j$ to the null space of J

finally $\Delta S_j = \hat{S}_j + \tilde{S}_j$

Physically-based Animation

- Example

$S_2\ (x_2, f_2)$ $S_4\ (x_4, f_4)$

$c_a = |\mathrm{x}_1 - \mathrm{a}| = 0$

$S_1\ (x_1, f_1)$

$c_b = |\mathrm{x}_n - \mathrm{b}| = 0$

$S_3\ (x_3, f_3)$ $S_4\ (x_4, f_4)$ $S_n\ (x_n, f_n)$

$$R = \int_{t_0}^{t_1} |\mathrm{f}|^2\, dt$$

$$\mathrm{J}_{ij} = \frac{\partial C_i}{\partial S_j}$$

$$\mathrm{H}_{ij} = \frac{\partial^2 R}{\partial S_i \partial S_j}$$

Iteratively solve

$$-\frac{\partial}{\partial S_j}(R) = \sum_j H_{ij}\hat{S}_j$$ $\hat{S}_j$ minimises R irrespective to C

$$-C_i = \sum_j J_{ij}(\hat{S}_j + \tilde{S}_j)$$ $\tilde{S}_j$ drives Cs to zero and project $\hat{S}_j$ to the null space of J

finally $\Delta S_j = \hat{S}_j + \tilde{S}_j$