Newton - Euler Equations of Motion

Note Title

Newton:

$$\sum F = \frac{dP}{dt} = \frac{d}{dt}(m \cdot \dot{p}) = \cancel{\dot{m}}^{\,0}\dot{p} + m\ddot{p}$$

$$\sum F = m\ddot{p}$$

Euler:

$$\sum \tau = \frac{dL}{dt} = \frac{d}{dt}(I\omega) = \dot{I}\omega + I\dot{\omega}$$

$$\sum \tau = \omega \times I\omega + I\dot{\omega}$$

$\longrightarrow$ set of linear equations in the unknowns $\ddot{p}, \dot{\omega}$

$$\ddot{p} = \sum F / m$$

$$\dot{\omega} = I^{-1}\left(\sum \tau - \omega \times I\omega\right)$$

Newton - Euler Equations in Matrix Form

$$\sum F = m\ddot{x} \qquad \text{3 eqns}$$

$$\sum \tau = I\dot{\omega} + \omega \times I\omega \qquad \text{3 eqns}$$

unknowns

$$\begin{bmatrix} m & & & \\ & m & & \\ & & m & \\ & & & I \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \sum F \\ \sum \tau - \omega \times I\omega \end{bmatrix}$$

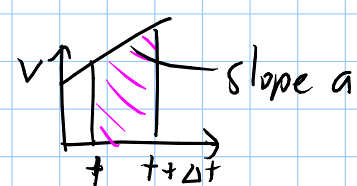
Simulating Particle Motion

state $Y = \begin{bmatrix} p \\ \dot{p} \end{bmatrix}$ $\quad \dot{Y} = \begin{bmatrix} \dot{p} \\ \ddot{p} \end{bmatrix}$ $\quad Y = Y + \dot{Y}\Delta t$

unknown

Euler integration

$t = 0$
$p = \langle 0, 0, 0 \rangle$
$\dot{p} = \langle 10, 20, 10 \rangle$
$g = \langle 0, -9.8, 0 \rangle$
while (1) {
  $F = m \cdot g - k\dot{p}$ drag force   compute external forces
  $\ddot{p} = F/m$ solve equations of motion
  $p \mathrel{+}= \dot{p}\Delta t$
  $\dot{p} \mathrel{+}= \ddot{p}\Delta t$ } Euler integration to update the state
  $t \mathrel{+}= \Delta t$
}



Note for better integration we can use:

$$p \mathrel{+}= \dot{p}\Delta t + \frac{(\ddot{p}\Delta t)\Delta t}{2}$$

Conversions Between Frames

Newton-Euler eq'ns only hold in an inertial frame!!

e.g., can use the world frame

$P_w = M\,P$
$V_w = R\,V$

$$M \triangleq \begin{bmatrix} \boxed{R} & \boxed{T} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

$\omega_w = R\,\omega$
$L_w = R\,L_w$

$I_w = ?\ I$

(1) $L_w = RL$
(2) $L_w = I_w \omega_w$

(3) $L = I\omega$
(4) $\omega_w = R\omega$

$RL = I_w \omega_w$

$RI\omega = I_w \omega_w$

$RI\omega = I_w R\omega$

$RI = I_w R \qquad R^{-1} = R^T$

$$\Rightarrow I_w = RIR^T$$

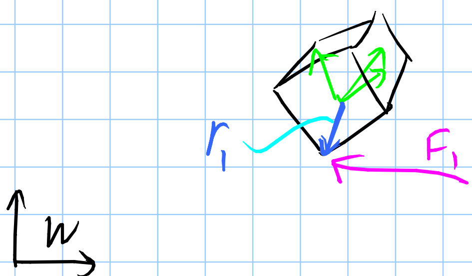
Rigid Body Simulation

$$Y = \begin{bmatrix} p \\ q \\ \dot{p} \\ \omega \end{bmatrix} \qquad \dot{Y} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \ddot{p} \\ \dot{\omega} \end{bmatrix}$$

$$\dot{q} = \frac{1}{2}\hat{\omega} \cdot q$$

$$\hat{\omega} = \langle 0, (\omega_x, \omega_y, \omega_z) \rangle$$

Conversions between frames



$$R = \text{quat\_to\_mat}(q)$$

$$r_{1_w} = R\, r_1$$

$$I_w = R\, I\, R^T$$

$$F_{1_w} = R\, F_1$$

Simulation Loop

$t = 0$

$p = \langle 0, 0, 0 \rangle$
$\dot{p} = \langle 10, 20, 10 \rangle$
$q = \langle 1, (0, 0, 0) \rangle$
$\omega = \langle 1, 2, 3 \rangle$

} initial state, world coords Y_0

while (1)

    [convert quantities to world coordinates]

    $F = m \cdot g + F_{1_w}$
    $\tau = 0 + r_{1_w} \times F_{1_w}$

} accumulate forces and torques

    $\ddot{p} = F/m$
    $\dot{\omega} = I_w^{-1}(\tau - \omega \times I_w \omega)$

} equations of motion

    $p \mathrel{+}= \dot{p}\Delta t + \ddot{p}\Delta t^2/2$
    $\dot{p} \mathrel{+}= \ddot{p}\Delta t$
    $q \mathrel{+}= \hat{\omega} \cdot q\, \Delta t/2$
    $\omega \mathrel{+}= \dot{\omega}\Delta t$

} integration to update the state

[should also renormalize q]

    $t \mathrel{+}= \Delta t$

}

Another choice of state for rigid body simulation

$$Y = \begin{bmatrix} x \\ q \\ p \\ L \end{bmatrix} \qquad \dot{Y} = \begin{bmatrix} \dot{x} \\ \dot{q} \\ \dot{p} \\ \dot{L} \end{bmatrix}$$

p — linear momentum

L — angular momentum

$p = m\dot{x} \rightarrow \dot{x} = p/m$

$L = I\omega \rightarrow \omega = I^{-1}L$

$\dot{q} = \dfrac{\hat{\omega} \cdot q}{2}$

$$\left\{ \begin{array}{l} \sum F = \dot{p} \\ \sum \tau = \dot{L} \end{array} \right.$$

$t = 0$

initialize x, q, p, L

while (1) {

[convert quantities to world coords]

$F = mg + F_{1_w}$

$\tau = 0 + r_{1_w} \times F_{1_w}$

total forces and torques

$\dot{p} = F$

$\dot{L} = \tau$

equations of motion (trivial!)

$\dot{x} = p/m; \; \ddot{x} = \dot{p}/m$

$x \mathrel{+}= \dot{x}\Delta t + \ddot{x}\Delta t^2/2$

$p \mathrel{+}= \dot{p}\Delta t$

$q \mathrel{+}= \dot{q}\Delta t$ — $\omega = I^{-1}L, \; \dot{q} = \hat{\omega} \cdot q/2$

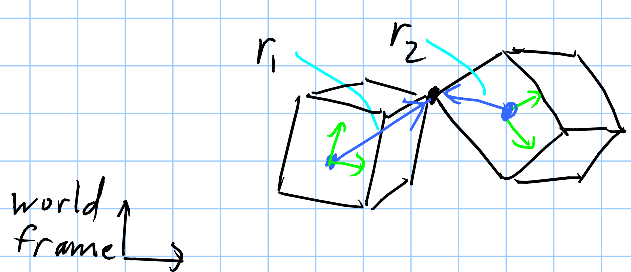
$L \mathrel{+}= \dot{L}\Delta t$

$t \mathrel{+}= \Delta t$

Euler integration to update the state

}

Simulation of two rigid bodies with a constraint



point-to-point constraint

Introduce:

- a constraint force, F_h (3 extra unknowns)
- a constraint equation (3 extra equations)

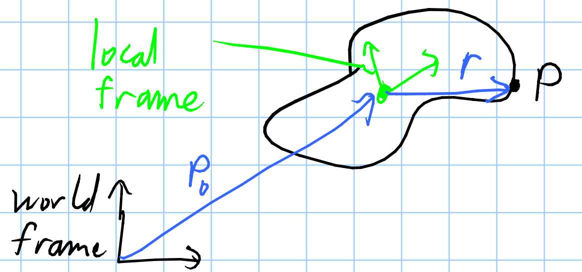
$r_{1_w} - r_{2_w} = 0 \qquad C(q) = 0$

$\dot{r}_{1_w} - \dot{r}_{2_w} = 0 \qquad C(q, \dot{q}) = 0$

$\boxed{\ddot{r}_{1_w} - \ddot{r}_{2_w} = 0} \qquad C(q, \dot{q}, \ddot{q}) = 0$

constraint on accelerations

Computing the acceleration of a point on a rigid, rotating body



$$p = p_0 + R r$$

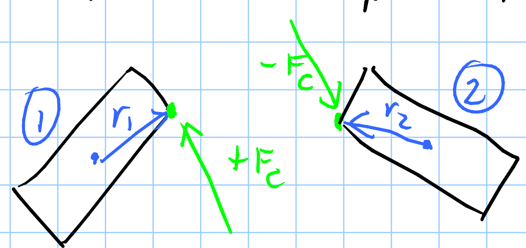
$$\dot{p} = \dot{p}_0 + \underbrace{\dot{R} r}_{\text{movement due to rotation}} + \cancel{R \dot{r}} \rightarrow 0 \text{ because } \dot{r} = 0$$

$$= \dot{p}_0 + \omega \times r$$

$$\ddot{p} = \underbrace{\ddot{p}_0}_{\text{linear accel of body}} + \underbrace{\dot{\omega} \times r}_{\text{tangential accel}} + \underbrace{\omega \times \omega \times r}_{\text{centripetal accel}}$$

chain rule

Newton-Euler structure for two blocks with pt-to-pt constraint



Block 1 Newton

$$\Sigma F = M_1 \ddot{p}_1 \qquad M_1 = \begin{bmatrix} m_1 & & \\ & m_1 & \\ & & m_1 \end{bmatrix}$$

$$F_c + M_1 g = M_1 \ddot{p}_1$$

$$M_1 \ddot{p}_1 - F_c = M_1 g$$

Block 1 Euler

$$\Sigma \tau = I_1 \dot{\omega}_1 + \omega_1 \times I_1 \omega_1$$

$$r_1 \times F_c = I_1 \dot{\omega}_1 + \omega_1 \times I_1 \omega_1$$

$$I_1 \dot{\omega}_1 - r_1 \times F_c = -\omega_1 \times I_1 \omega_1$$

Block 2 Newton

$$M_2 \ddot{p}_2 + F_c = M_2 g$$

Block 2 Euler

$$I_2 \dot{\omega}_2 + r_2 \times F_c = -\omega_2 \times I_2 \omega_2$$

Two Blocks (continued)

Constraint equation

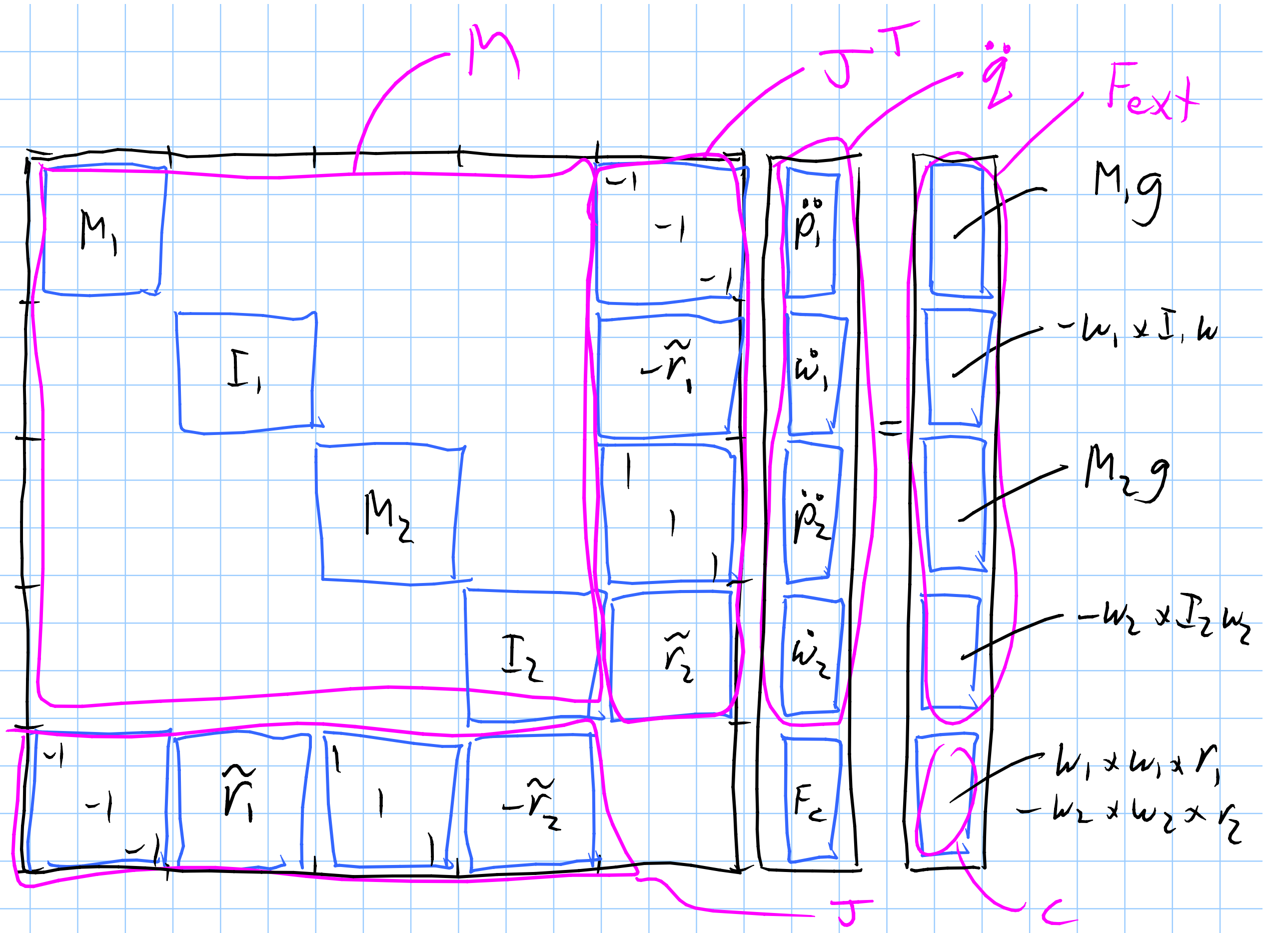
$$\ddot{p}_1 + \dot{\omega}_1 \times r_1 + \omega_1 \times \omega_1 \times r_1$$

$$= \ddot{p}_2 + \dot{\omega}_2 \times r_2 + \omega_2 \times \omega_2 \times r_2$$

$$\longrightarrow \quad -\ddot{p}_1 + r_1 \times \dot{\omega}_1 + \ddot{p}_2 - r_2 \times \dot{\omega}_2 = \omega_1 \times \omega_1 \times r_1 + \omega_2 \times \omega_2 \times r_2$$

Two Blocks – Equation in Matrix Form

$$\begin{bmatrix} M_1 & & & & -1 \\ & I_1 & & & -\tilde{r}_1 \\ & & M_2 & & 1 \\ & & & I_2 & \tilde{r}_2 \\ -1 & \tilde{r}_1 & 1 & -\tilde{r}_2 & \end{bmatrix} \begin{bmatrix} \ddot{p}_1 \\ \dot{\omega}_1 \\ \ddot{p}_2 \\ \dot{\omega}_2 \\ F_c \end{bmatrix} = \begin{bmatrix} M_1 g \\ -\omega_1 \times I_1 \omega \\ M_2 g \\ -\omega_2 \times I_2 \omega_2 \\ \omega_1 \times \omega_1 \times r_1 - \omega_2 \times \omega_2 \times r_2 \end{bmatrix}$$



$O(n^3) \longrightarrow O(n)$

$$\left.\begin{aligned} M\ddot{q} + J^T F_c &= F_{ext} \\ J\ddot{q} &= C \end{aligned}\right\} \text{exploit sparse structure to solve efficiently}$$

→ see "Linear Time Dynamics using Lagrange Multipliers" by D. Baraff, SIGGRAPH 1996

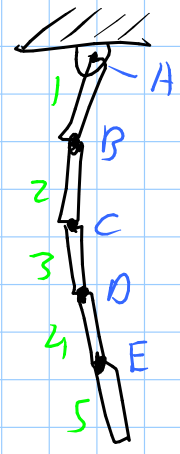
Constraint Stabilization

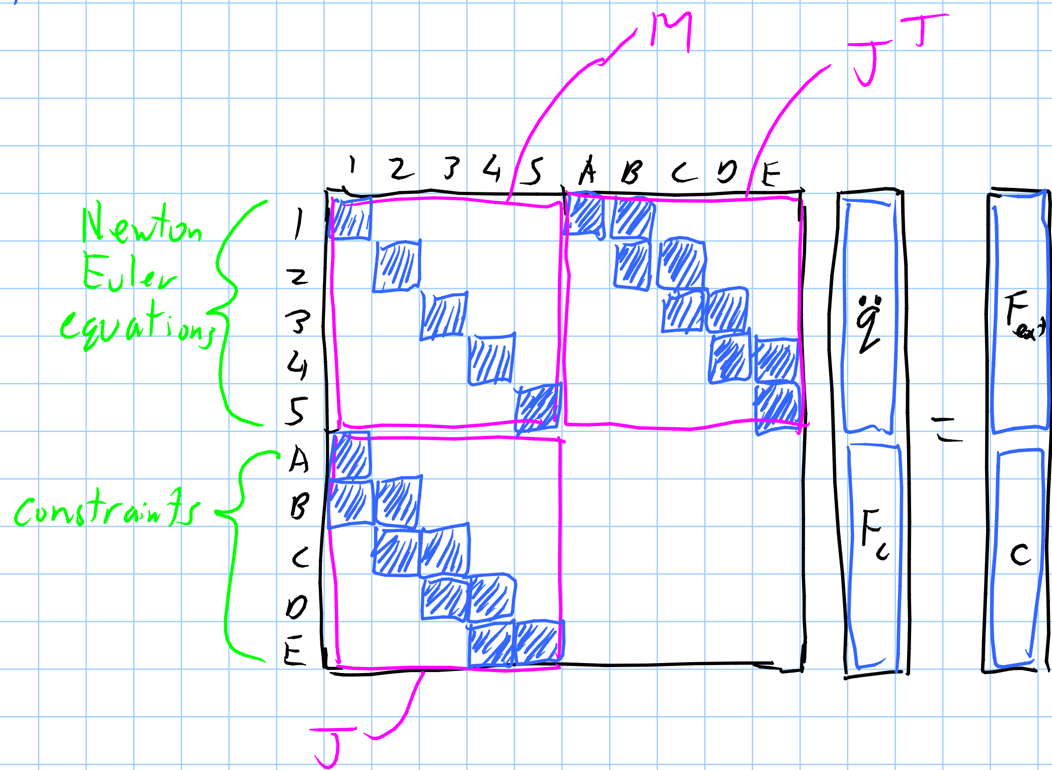
$\ddot{r}_{1_w} = \ddot{r}_{2_w}$ at the given time instant, but error will creep in during the integration step

$$\left.\begin{aligned} \dot{r}_{1_w} - \dot{r}_{2_w} &= 0 \\ r_{1_w} - r_{2_w} &= 0 \end{aligned}\right\} \text{begin to be violated}$$

Constraint stabilization adds terms to help adress this problem. — see Baumgarte

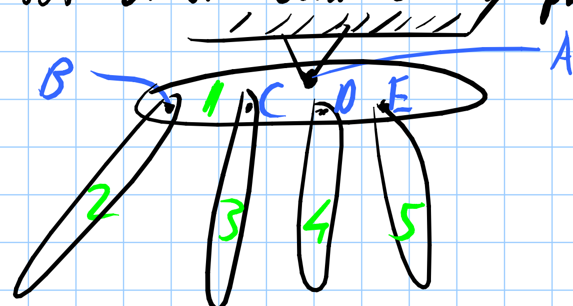
Simulation of a 5-link chain

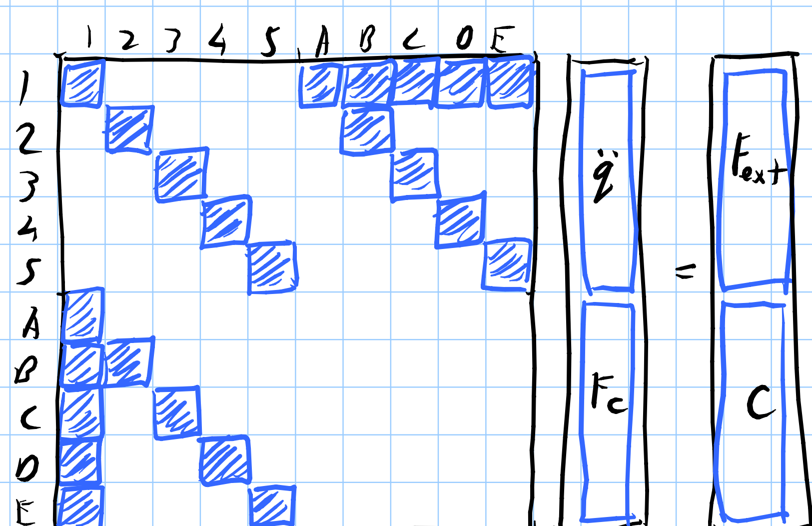




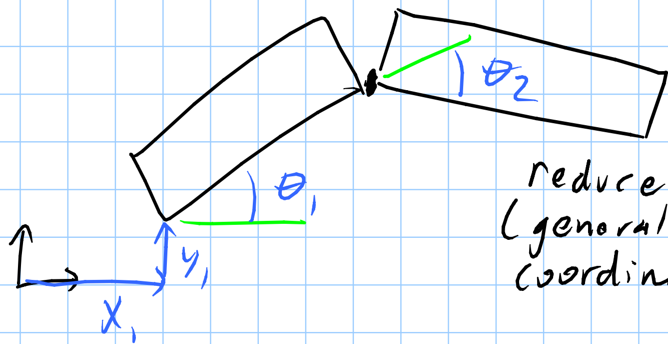
What about other connectivity patterns?

E.g.





Reduced Coordinate Formulations



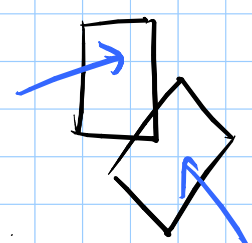
reduced (generalized) coordinates

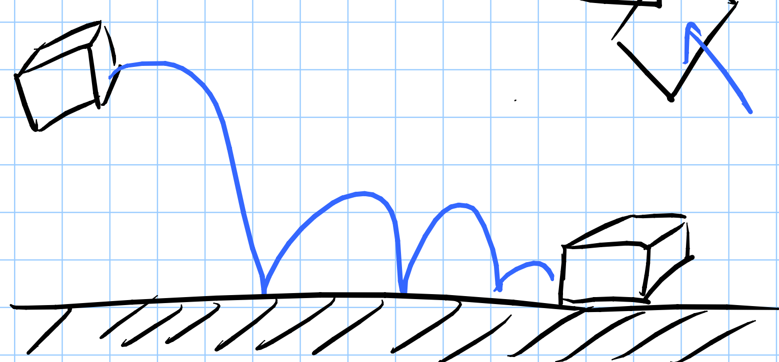
$$q = \begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \\ \theta_2 \end{bmatrix}$$

Compare to: $\begin{bmatrix} x_1 \\ y_1 \\ \theta_1 \\ x_2 \\ y_2 \\ \theta_2 \end{bmatrix}$

see Featherstone "Robot Dynamics Algorithms" for $O(n)$ algorithms for articulated chains

Collisions



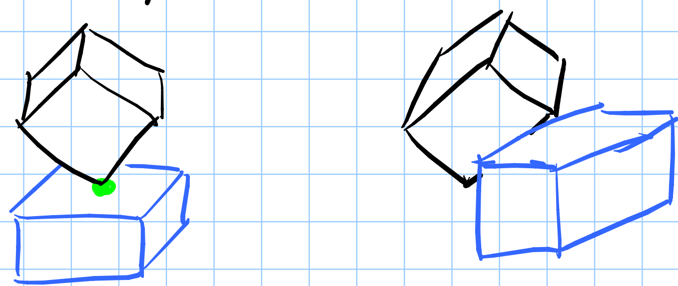


subproblems:

1. collision detection
2. collision resolution
3. resting contact

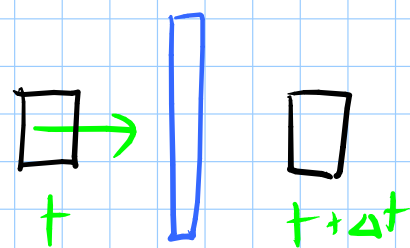
Collision Detection

In what ways can objects collide?



vertex-face edge-edge

Idea: Intersect geometry at each timestep

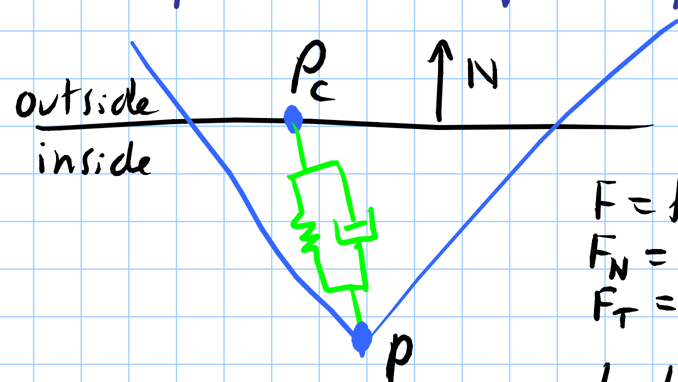


problem:
can miss collisions

→ separate literature on efficient collision detection

Collision Resolution

Apply a spring-and-damper force
that pushes the objects apart.



$$F = k_p(P_c - P) + k_d(\cancel{\dot{P}_c}^{\,0} - \dot{P})$$
$$F_N = F \cdot N$$
$$F_T = F - F_N$$

check that $F_N > 0$

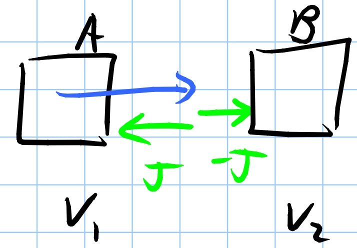
check that $\dfrac{|F_T|}{|F_N|} < \mu$

friction coeff.

Problem:
How to set k_p, k_d?
Needs to be a function of object mass

Stiff system, therefore requires small timesteps
and potentially unstable

Impulse Forces



$$F = m\frac{\Delta v}{\Delta t}$$

$$\underbrace{F\Delta t}_{\substack{\text{impulse}\\ J}} = \underbrace{m\Delta v}_{\substack{\text{momentum chge}\\ \Delta P}}$$

capital P = momentum

$$P_A^+ = P_A^- + J \quad (\Delta P)$$

$$P_B^+ = P_B^- - J$$

$$P_A^+ + P_B^+ = P_A^- + P_B^-$$

conservation of total linear momentum

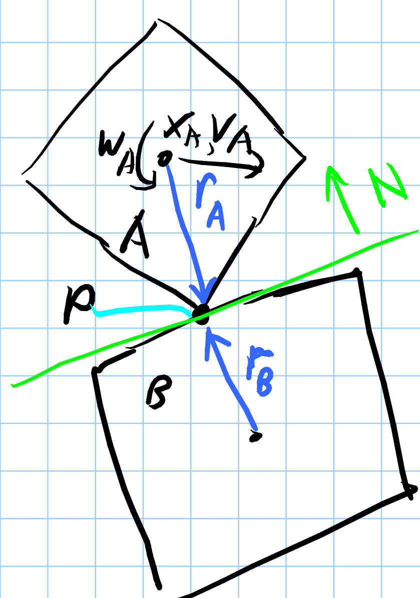
$L = I\omega$

$$L_A^+ = L_A^- + r_A \times J$$

$$L_B^+ = L_B^- - r_B \times J$$

---

x_A
v_A
ω_A



x_B
v_B
ω_B

$$p_A = x_A + r_A$$

$$\dot{p}_A = v_A + \omega_A \times r_A$$

$$\dot{p}_B = v_B + \omega_B \times r_B$$

$$v_{rel} = (\dot{p}_A - \dot{p}_B) \cdot N$$

scalar

$$\boxed{v_{rel}^+ = -\varepsilon\, v_{rel}^-}$$

coeff of restitution

scalar eq'n

$J = j\, N$

scalar unknown

① Develop expressions for V_A^+, V_B^+, w_A^+, w_B^+ as a fn of j and other known quantities

$$P_A^+ = P_A^- + J$$
$$m_A V_A^+ = m_A V_A^- + jN$$

$$V_A^+ = V_A^- + \frac{jN}{m_A}$$

similarly,

$$V_B^+ = V_B^- - \frac{jN}{m_B}$$

$$L_A^+ = L_A^- + r_A \times J$$
$$I_A w_A^+ = I_A w_A^- + r_A \times jN$$

$$w_A^+ = w_A^- + j\, I_A^{-1}(r_A \times N)$$

$$w_B^+ = w_B^- - j\, I_B^{-1}(r_B \times N)$$

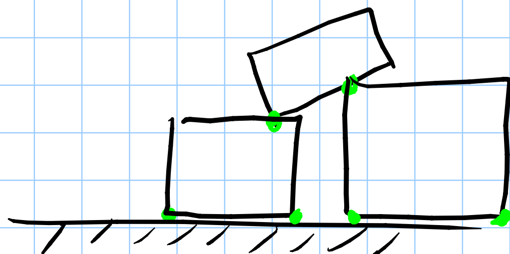
② Substitute into $\underbrace{V_{rel}^+}_{LHS} = \underbrace{-\epsilon V_{rel}^-}_{RHS}$ (known!)

$$\begin{aligned}
V_{rel}^+ &= (\dot{p}_A^+ - \dot{p}_B^+) \cdot N \\
&= (V_A^+ + w_A^+ \times r_A - V_B^+ - w_B^+ \times r_B) \cdot N \\
&= \Big(V_A^- + \frac{jN}{m_A} + (w_A^- + j\, I_A^{-1}(r_A \times N)) \times r_A \\
&\quad - V_B^- + \frac{jN}{m_B} - (w_B^- - j\, I_B^{-1}(r_B \times N)) \times r_B\Big) \cdot N \\
&= \underbrace{(V_A^- + w_A^- \times r_A - V_B^- - w_B^- \times r_B) \cdot N}_{V_{rel}^-} \\
&\quad + j\,\underbrace{\Big(\frac{1}{m_A} + \frac{1}{m_B} + (I_A^{-1}(r_A \times N) \times r_A) \cdot N + (I_B^{-1}(r_B \times N) \times r_B) \cdot N\Big)}_{c}
\end{aligned}$$

$$-\epsilon V_{rel}^- - V_{rel}^- = jc$$

$$\Rightarrow \quad j = -\frac{(1+\epsilon)\, V_{rel}^-}{c}$$

Resting Contact



$0 \le v_{rel} \le \gamma$ where γ is a small threshold velocity

compute set of forces $F_i = f_i N_i$

unknown scalar

conditions:

1. $\dot{v}_{rel_i} \ge 0$ disallow accelerations in direction of interpenetration

2. $f_i \ge 0$ forces must repel, and not "glue"

3. $f_i = 0$ if $\dot{v}_{rel} > 0$ force must be zero if contact breaks

$$f_i \, \dot{v}_{rel_i} = 0$$

To solve:

Determine $\dot{v}_{rel_i}$ as a linear fn of f_i's

$$\vec{\dot{v}}_{rel} = A\vec{f} + \vec{b}$$

$\Rightarrow$ ① $A\vec{f} \ge -\vec{b}$

③ $\sum f_i \, \dot{v}_{rel_i} = 0$

$\vec{f} \cdot \vec{\dot{v}}_{rel} = 0$

$f^T \dot{v}_{rel} = 0 \rightarrow f^T A f + f^T b = 0$

contains f_i^2 terms

scalar eq'n

Quadratic Programming for resting contact

$$
\begin{aligned}
&③ \quad f^T A f + f^T b = 0 \\
&① \quad Af + b \geq 0 \\
&② \quad f \geq 0
\end{aligned}
$$

scalar eq'n

set of linear constraints

→ more specifically, belongs to a specific subclass:

Linear Complementarity Problems (LCP)

QP: maximize $5x^2 + 3y^2$

subject to
$$
\begin{aligned}
2x + y &\geq 3 \\
-3x - 2y &\geq -2
\end{aligned}
$$