Université Concordia University

Animation for Computer Games
COMP 477/6311

Prof. Tiberiu Popa

Collision Handling

Acknowledgements

- Some images were taken from the web for illustrations

Baraff, D. (2001). Physically based modeling: Rigid body simulation. *SIGGRAPH Course Notes, ACM SIGGRAPH*, *2*(1), 2-1.

http://graphics.cs.cmu.edu/courses/15-869-F08/lec/14/notesg.pdf

Collision Handling

- Challenges:

1. We detect two bodies intersecting at time t
   - They collided probably earlier t-dt
   - Moment of detection is not exactly the moment of collision
2. What to do?
   - Colliding contact(e.g. objects are in contact for a fraction of time and then they get separated – pool)
   - Resting contact (i.e. have contact but contact point does not move)

Collision Handling

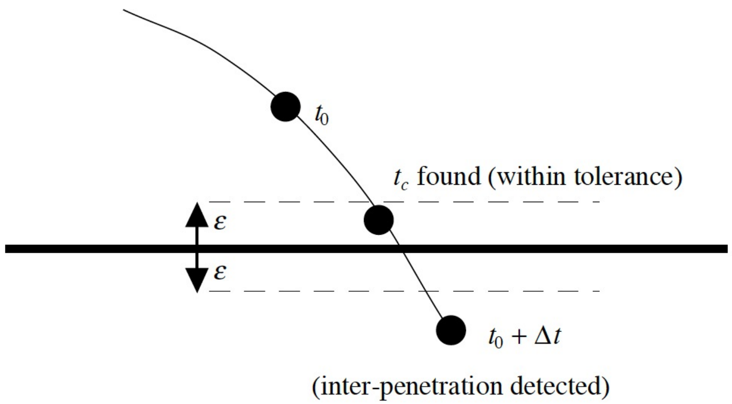
- Main approaches:
  1. Penalty forces
     - Add forces to change the motion away from an object
     - Depends on the distance field
  - Pros:
     - Easy to integrate (Remember: changing the forces is the main control mechanism)
     - Very efficient on rigid bodies (i.e. distance field can be precomputed)
     - Takes care of both challenges
  - Cons:
     - Heuristic (i.e. not physical)
       - Will fail sometimes (e.g. interpenetration)

Collision Handling

- Main approaches:
  2. Impulse ”forces”
     - Improperly called forces
     - Instantly change the velocity field
  - Pros:
    - More physically “correct”
    - Provides Guarantees
  - Cons:
    - Complex and slower
    - Numerical issues (discontinuities in the velocity field)

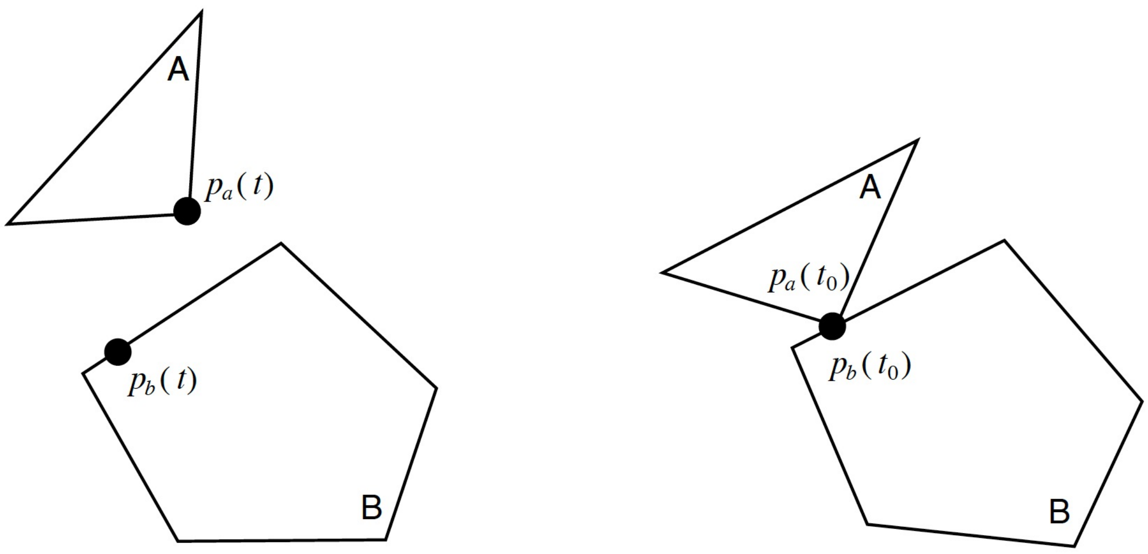
Collision Handling

- Challenges:
1. We detect two bodies intersecting at time t
   - They collided probably earlier t-dt
   - Moment of detection is not exactly the moment of collision
- Binary search for the exact contact time
   - Several evaluation of the integrator (fairly slow)

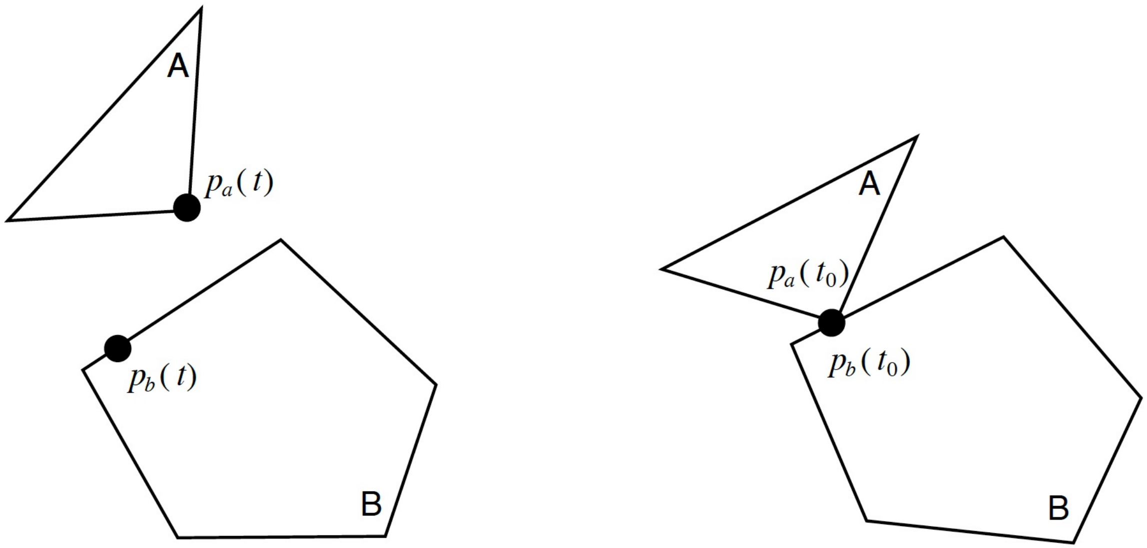


Collision Handling

1. Collision modeling:
   1. Triangular meshes
   2. Exact point to plane or edge to edge
      1. Allow for degeneracies



Collision Handling



$$\dot{p}_a(t_0) = v_a(t_0) + \omega_a(t_0) \times (p_a(t_0) - x_a(t_0))$$

$$\dot{p}_b(t_0) = v_b(t_0) + \omega_b(t_0) \times (p_b(t_0) - x_b(t_0))$$

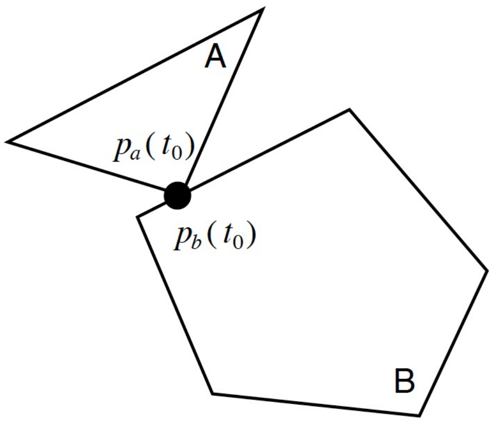
$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0))$$

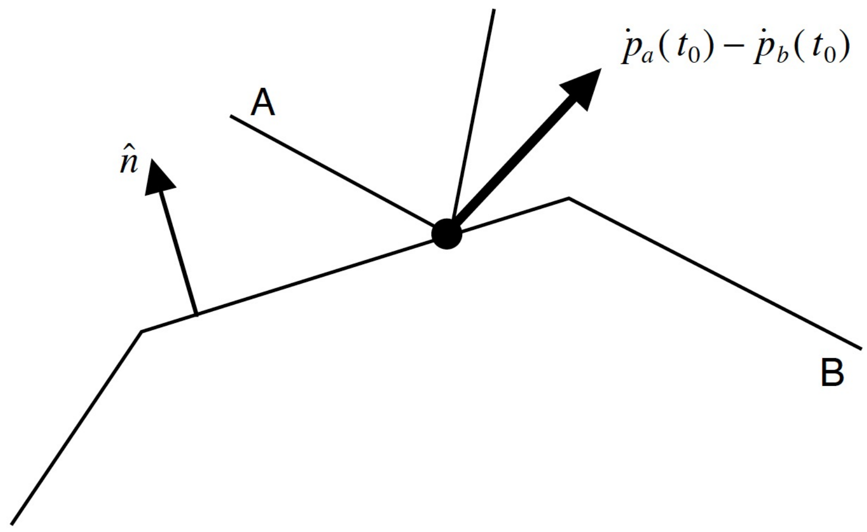
Collision Handling

$$\dot{p}_a(t_0) = v_a(t_0) + \omega_a(t_0) \times (p_a(t_0) - x_a(t_0))$$

$$\dot{p}_b(t_0) = v_b(t_0) + \omega_b(t_0) \times (p_b(t_0) - x_b(t_0))$$

$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0))$$



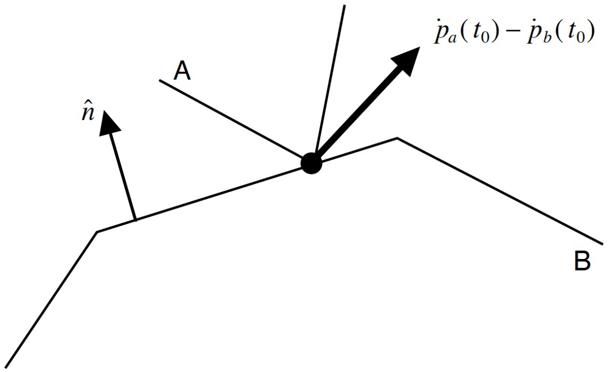


Collision Handling

$$\dot{p}_a(t_0) = v_a(t_0) + \omega_a(t_0) \times (p_a(t_0) - x_a(t_0))$$

$$\dot{p}_b(t_0) = v_b(t_0) + \omega_b(t_0) \times (p_b(t_0) - x_b(t$$

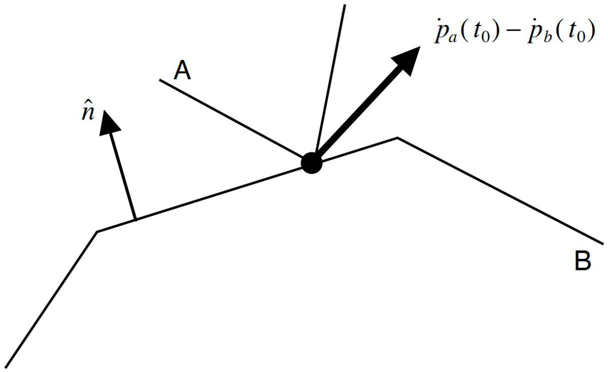
$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0))$$



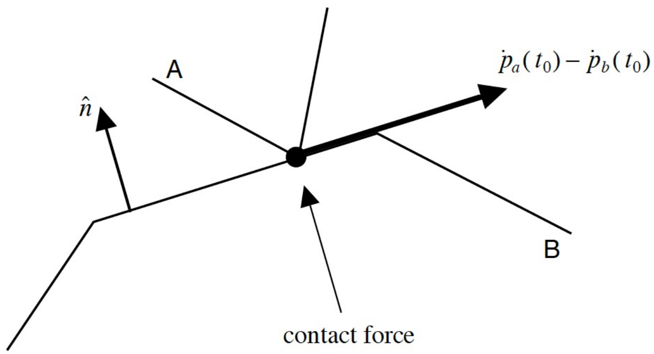
$v_{rel} > 0$	Objects move away from each other
$v_{rel} = 0$	Resting contact (later)
$v_{rel} < 0$	**Impulsive Contact**

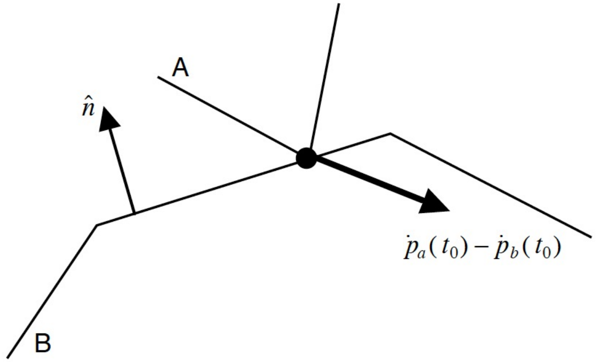
Collision Handling

$$\begin{cases} v_{rel} > 0 & \text{Objects move away from each other} \\ v_{rel} = 0 & \text{Resting contact (later)} \\ v_{rel} < 0 & \textbf{Impulsive Contact} \end{cases}$$



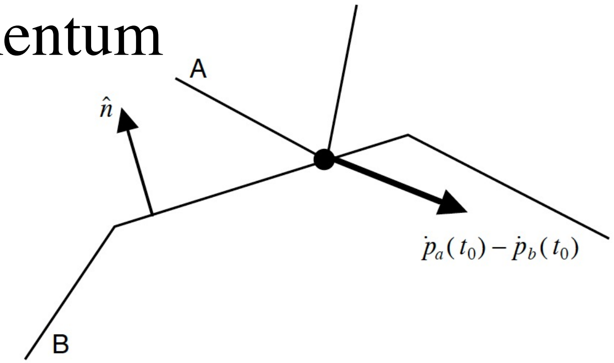
$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0))$$





Collision Handling

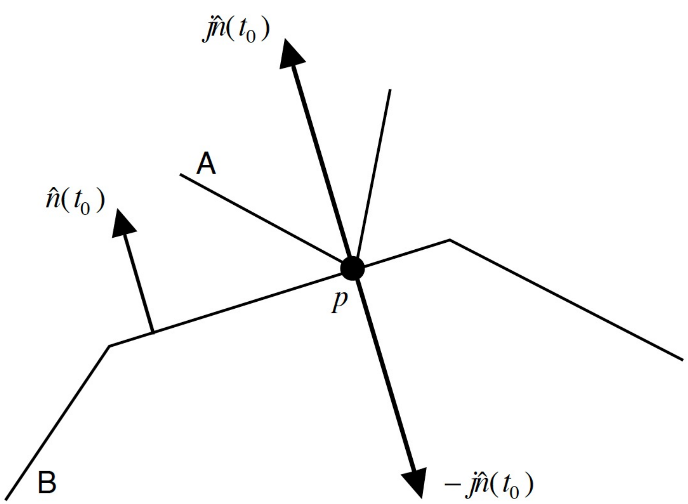
We introduce impulse (sudden change in linear momentum)
Applied to one point only, so not angular momentum



$$\Delta P = J \qquad \Delta v = \frac{J}{M}$$

Need to compute J

$$J = j\hat{n}(t_0)$$



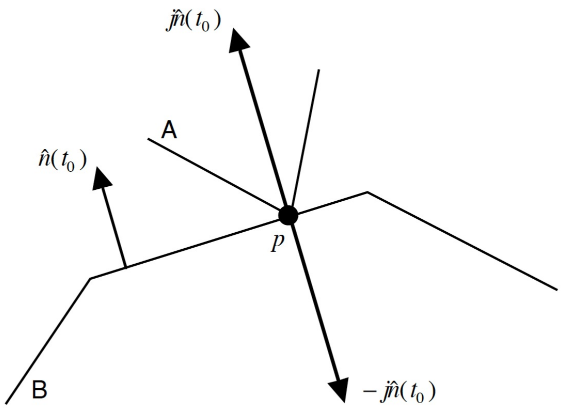
*no friction

Collision Handling

$$v_{rel}^{-} = \hat{n}(t_0) \cdot (\dot{p}_a^{-}(t_0) - \dot{p}_b^{-}(t_0))$$

$$v_{rel}^{+} = \hat{n}(t_0) \cdot (\dot{p}_a^{+}(t_0) - \dot{p}_b^{+}(t_0))$$

$$v_{rel}^{+} = -\epsilon v_{rel}^{-}$$



$$j = \frac{-(1+\epsilon)v_{rel}^{-}}{\frac{1}{M_a} + \frac{1}{M_b} + \hat{n}(t_0) \cdot \left(I_a^{-1}(t_0)\left(r_a \times \hat{n}(t_0)\right)\right) \times r_a + \hat{n}(t_0) \cdot \left(I_b^{-1}(t_0)\left(r_b \times \hat{n}(t_0)\right)\right) \times r_b}$$

Collision Handling

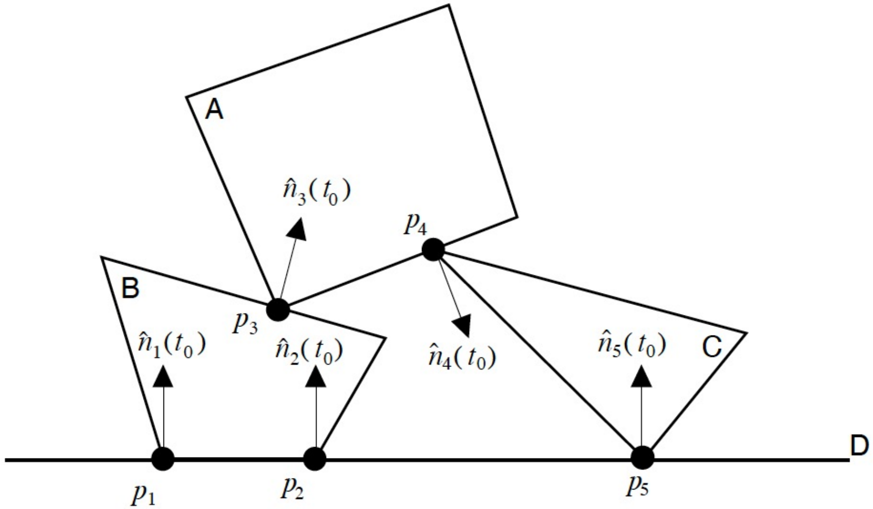
$$\begin{cases} v_{rel} > 0 \\ v_{rel} = 0 \\ v_{rel} < 0 \end{cases}$$

Objects move away from each other

Resting contact

Impulsive Contact

$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0))$$



Collision Handling

Resting contact

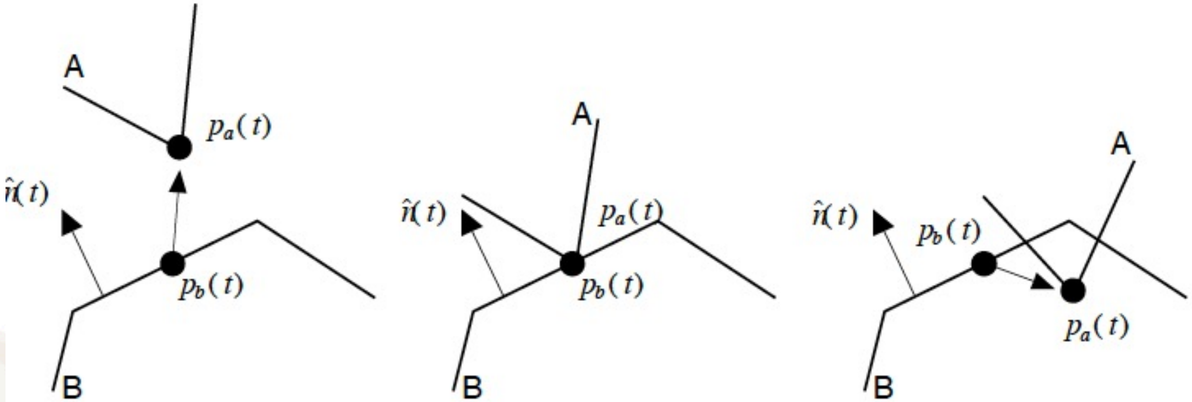
Add contact forces at intersection points → normal direction

$$d_i(t) = \hat{n}_i(t) \cdot (p_a(t) - p_b(t))$$

"Distance" function → should never be negative

Add force to keep it positive

$$F_i(t_0) = f_i \hat{n}_i(t_0)$$



Collision Handling

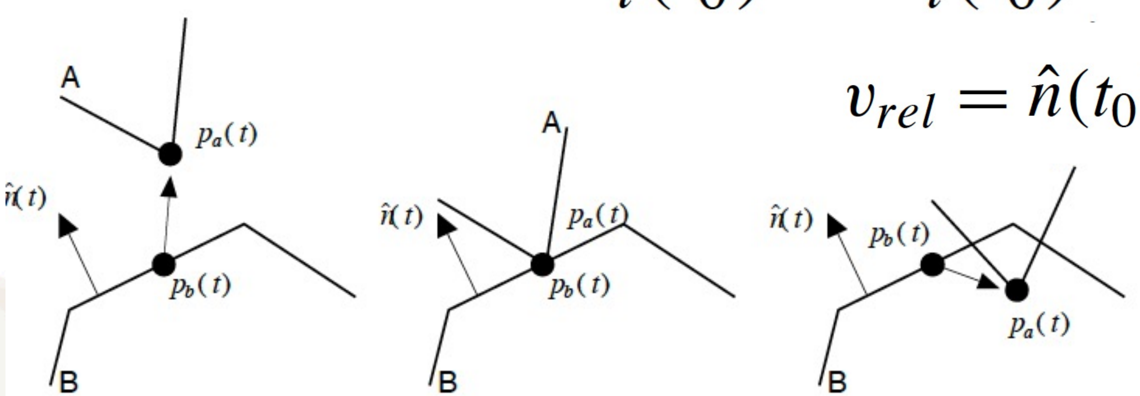
$$d_i(t) = \hat{n}_i(t) \cdot (p_a(t) - p_b(t))$$

$d_i(t_0) = 0$ follows that $\dot{d}_i(t_0) \geq 0$

$$\dot{d}_i(t) = \dot{\hat{n}}_i(t) \cdot (p_a(t) - p_b(t)) + \hat{n}_i(t) \cdot (\dot{p}_a(t) - \dot{p}_b(t))$$

$$\dot{d}_i(t_0) = \hat{n}_i(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0)) = 0$$

$$v_{rel} = \hat{n}(t_0) \cdot (\dot{p}_a(t_0) - \dot{p}_b(t_0)) = 0$$



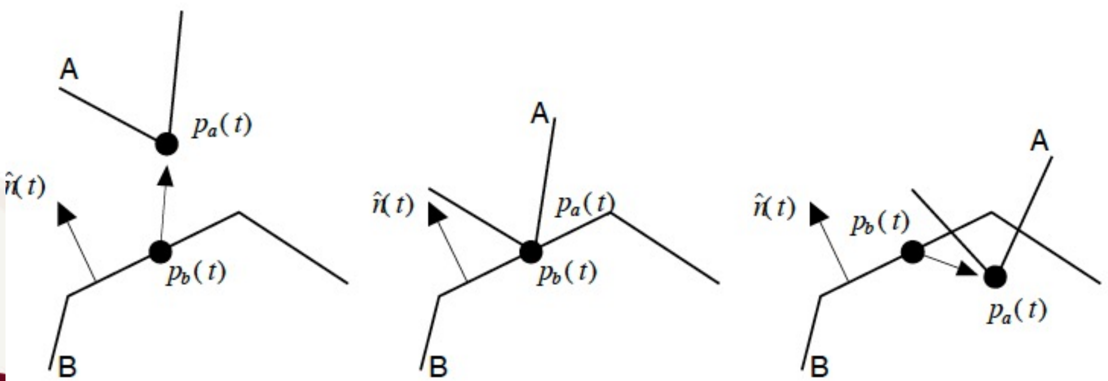
Collision Handling

$$d_i(t) = \hat{n}_i(t) \cdot (p_a(t) - p_b(t))$$

$d_i(t_0) = 0$ and $\dot{d}_i(t) = 0$ follows that $\ddot{d}_i(t_0) \geq 0$

$F_i(t_0) = f_i \hat{n}_i(t_0)$ follows that $f_i \geq 0$

$f_i \ddot{d}_i(t_0) = 0$ Because F_i should be 0 when not in contact



Collision Handling

Recap:

Need to find forces $F_i(t_0) = f_i\hat{n}_i(t_0)$ s.t. $\begin{cases} \ddot{d}_i(t_0) \geq 0 \\ f_i \geq 0 \\ \ddot{d}_i(t_0) = \sum_{k=1}^{n} a_{ik} f_k + b_i \end{cases}$

Where i is the index of the n points of contact (see Appendix D in reference below)

Solved using quadratic programming

Baraff, D. (2001). Physically based modeling: Rigid body simulation. *SIGGRAPH Course Notes, ACM SIGGRAPH*, *2*(1), 2-1.