MAY 2, 2024

TEXAS
The University of Texas at Austin

CS 384G Final Project

Rigid Body Physics Simulation in Minecraft

Lisong Sun, Pranav Venkatesh
The University of Texas at Austin

Project Goals

- Minecraft: Rigid Body Physics Simulation
- Block-Terrain Interaction
- Block-Block Interaction

Velocity Verlet

- (as described in lecture)
- Include rotational component

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2$$

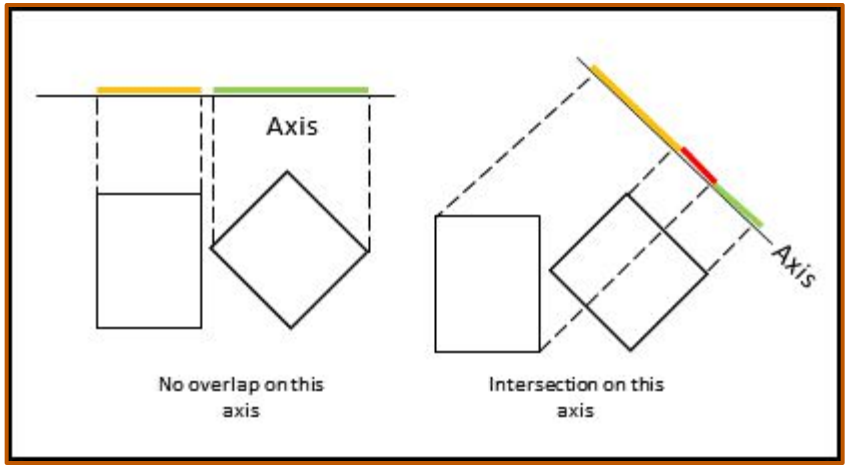
$$v(t + \Delta t) = v(t) + a(t)\Delta t$$

$$\theta(t + \Delta t) = \theta(t) + \omega(t)\Delta t + \frac{1}{2}\alpha(t)\Delta t^2$$

$$\omega(t + \Delta t) = \omega(t) + \alpha(t)\Delta t$$

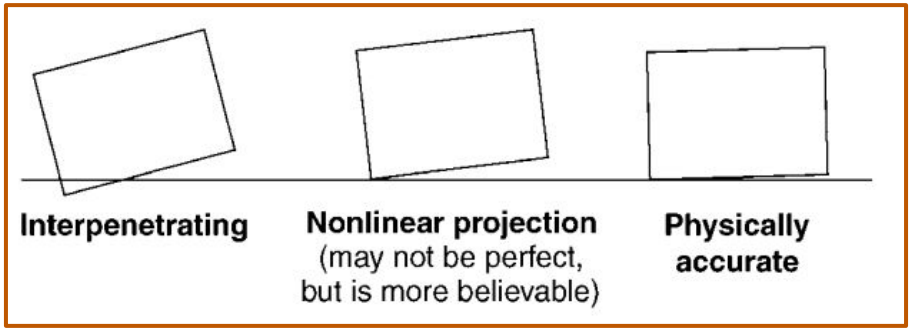
Collision Detection

- Cube-cube collisions
- Two types of collisions: point-face and edge-edge
- Handled with **separating axis theorem**
- Results in contact points, contact normals, and penetration depth



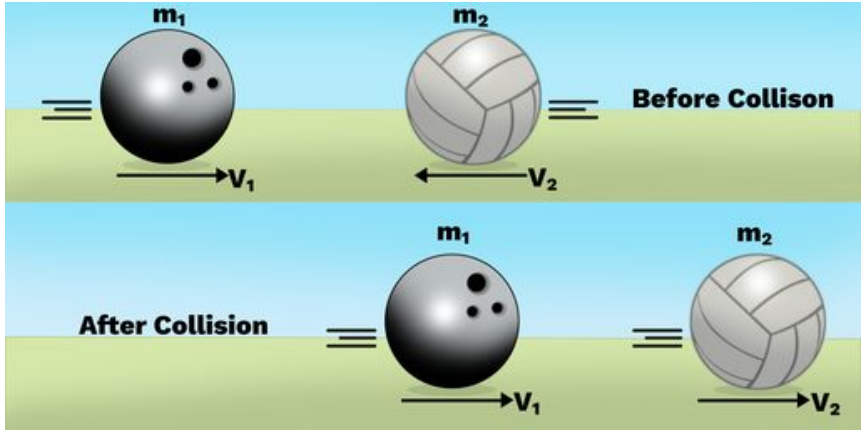
Collision Resolution

- Follow *Game Physics Engine Development* by Ian Millington
- Resolve each contact
- 1) Interpenetration resolution
  - Caused by discretizing timesteps
  - Resolved with combination of translation and rotation



Collision Resolution (cont.)

- 2) Velocity resolution
  - Apply force due to elastic collisions
  - Modeled as an impulse and impulsive torque
  - Calculations involve mass, moment of inertia, distance to center of mass, etc.
  - Friction force added perpendicular to contact normal



Velocity resolution in detail

1) Calculate desired velocity change to preserve momentum

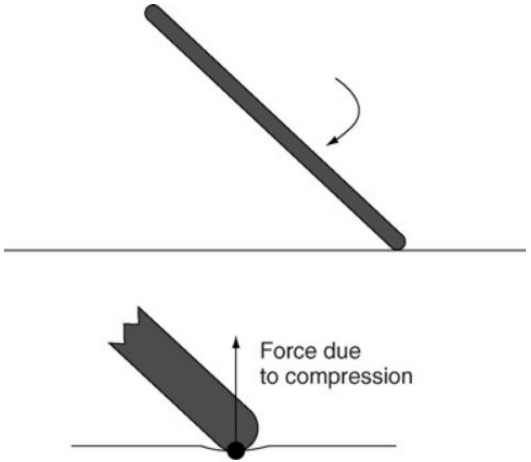
   $p = mv$, $L = I\omega$

2) Calculate velocity change per unit impulse

   Add linear velocity of body + velocity @ point due to rotation

3) Using 1) and 2), calculate impulse needed to generate desired velocity change

4) Convert impulse @ point to impulsive torque



Additional Challenges

- Multiple-collision resolution
  - Iterative approach
- Stability
  - Kill low velocity movement
- Optimization
  - Brute force collision detection
  - Sleep mechanism

Demo Video