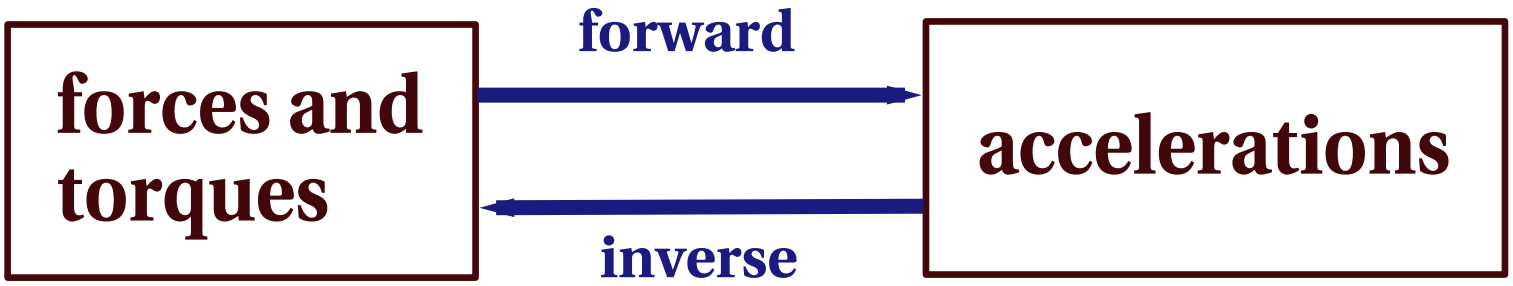
Dynamic Simulation

- point mass
- spring /mass systems
- linkages of rigid bodies
- other physical phenomena
  - aerodynamics
  - fluids
  - fracture
  - explosions

Control

Forward and Inverse Dynamics



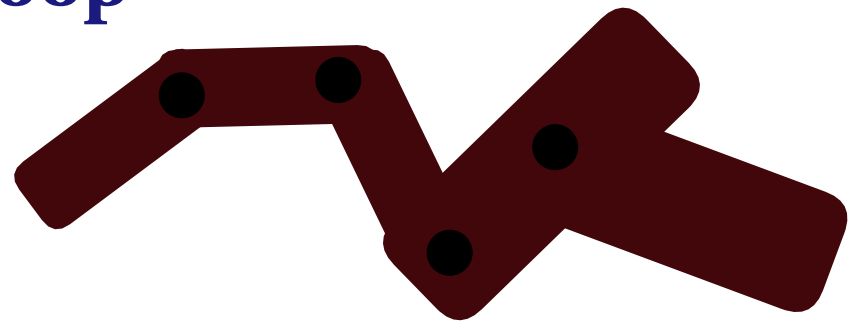
forward: given forces and torques
what is the motion?

inverse: given prescribed motion
what are the forces and torques?

Rigid Body Dynamics

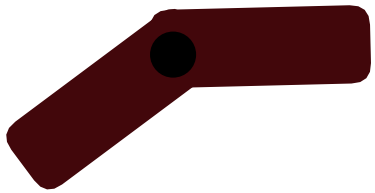
what can we simulate?

open loop

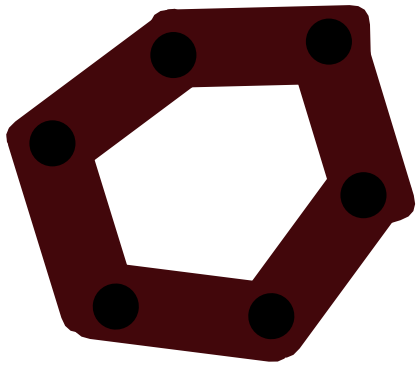


joints

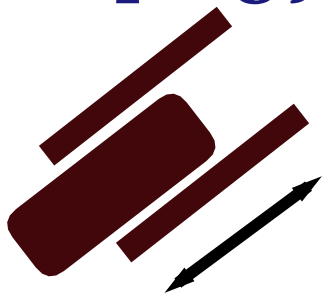
rotary joints (1,2,3d)



closed loop



telescoping joints



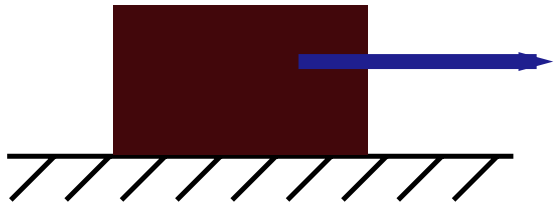
Forces / Torques

gravity

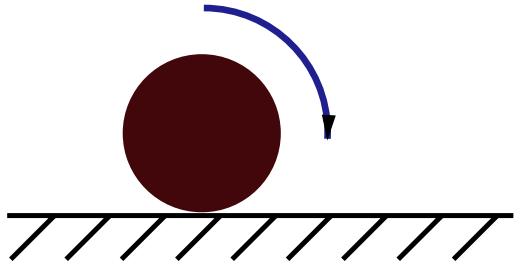
wind

collisions/contact

sliding



rolling



Dynamic Simulation

physical description of objects =>

    equations of motion

loop

    solve equations for accelerations

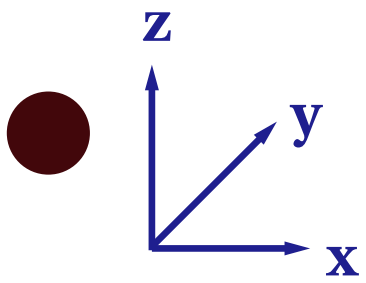
    integrate to find velocities and positions

    draw graphics

Physical Description

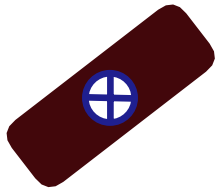
point mass

translate in x,y,z



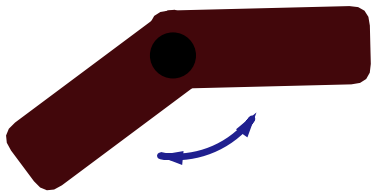
rigid body

translate in x,y,z
rotate about x,y,z



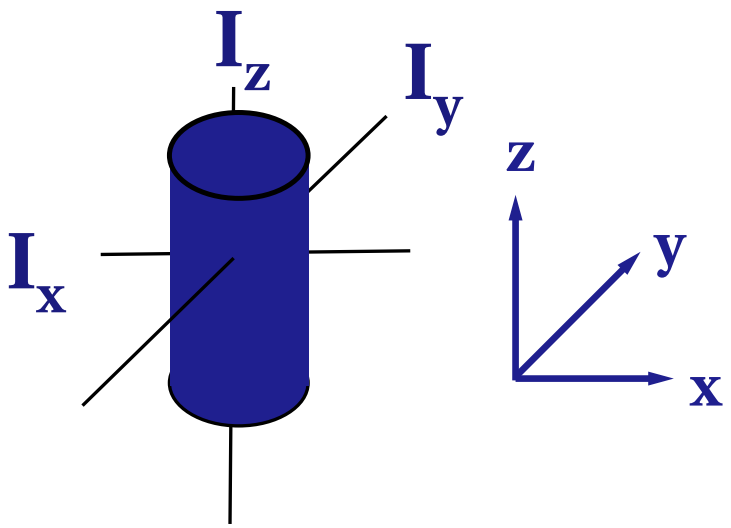
articulated rigid body

translate in x,y,z
rotate about x,y,z
constraints wrt neighbors



System Description

mass
center of mass
moment of inertia



$$I_x = I_y = 1/12\ m\ (3r^2 + L^2)$$

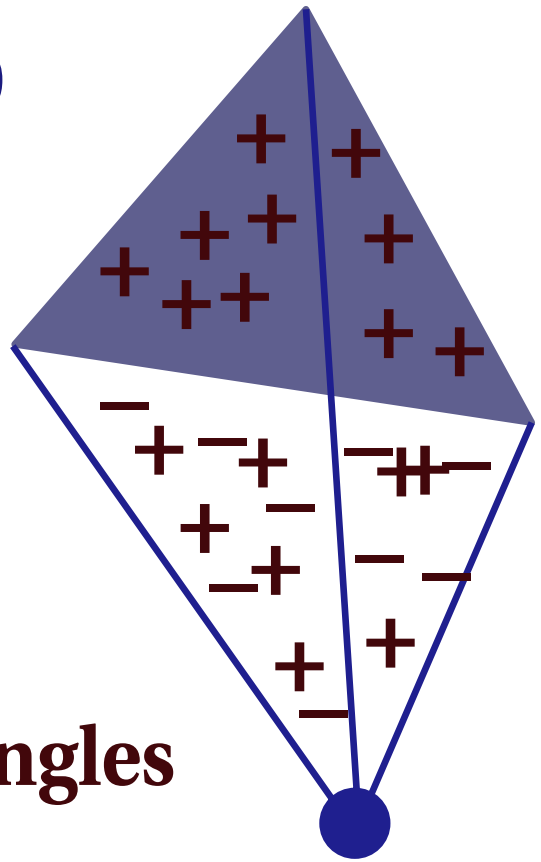
$$I_z = \frac{mr^2}{2}$$

formula for simple solids

Moment of Inertia of Polygonal Objects

Lien and Kajiya '84, IEEE CG&A

In 2D



+/– triangles

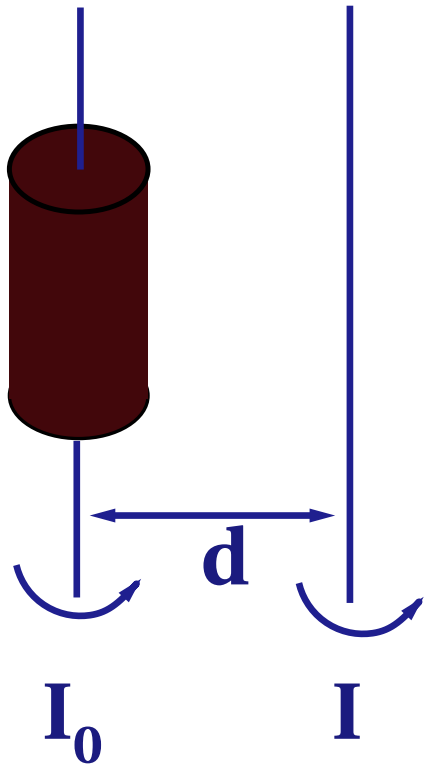
In 3D

+/– tetrahedrons

Mirtich, J Graphics Tools 1996
Algorithm to minimize numerical error and be efficient. Code on the web

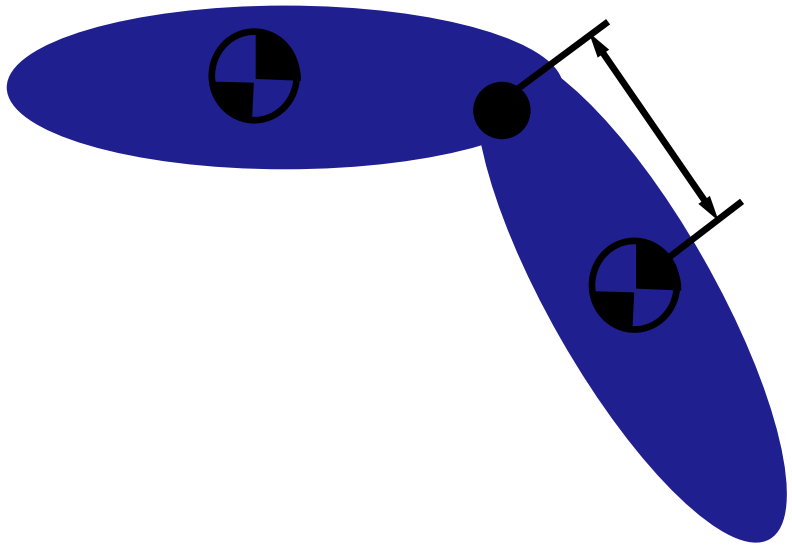
Parallel Axis Theorem

$$I = I_0 + md^2$$



SdFast, Symbolic Dynamics

link: mass, moi
joints: DOF,
distance from COM of links



code for the equations of motion
hooks for applying forces, torques