Rigid Body Dynamics and Beyond
Rigid Bodies
A rigid body

- Collection of particles
- Distance between any two particles is always constant
- What types of motions preserve these constraints?
  - Translation, rotation
Rigid Body Parameterization (reduced coordinates)

\[ p(t) = x(t) + R(t)p_{\downarrow 0} \]

- \( x(t) \in \mathbb{R}^3 \)
- \( R(t) \in \mathbb{R}^{3 \times 3} \)

position orientation
Body Orientation

Rotates vectors from body to world coordinates

- Columns of $R(t)$ encode world coordinates of body $a$.
Center of Mass (COM)

Geometric center of the body
- (0,0,0) in body coordinates
- $x(t)$ in world coordinates

Total mass:
$$M = \sum_i m_i$$

Center of mass:
$$\frac{\sum_i m_i p_i(t)}{M}$$
Body velocities

How do the COM position and orientation change with time?

Linear velocity: $v(t) = \frac{dx(t)}{dt} = \dot{x}(t)$

Angular velocity: $\omega(t) = ?$

$\omega(t)$ encodes spin direction and magnitude

What is the relationship between $\omega(t)$ and $R(t)$?
Angular velocity

- Consider vector $\mathbf{r}(t)$. What is $\dot{\mathbf{r}}(t)$?

$$\dot{\mathbf{r}}(t) = \omega(t) \times \mathbf{r}(t)$$

- Relation to $\dot{\mathbf{R}}(t) = \frac{d\mathbf{R}(t)}{dt}$?
Angular Velocity

\( \mathbf{R} \) rotates vectors from body to world coords

- Columns of \( \mathbf{R}(t) \): world coordinates of body axes
- Columns of \( \dot{\mathbf{R}}(t) \): change of body axes world coordinates wrt time

\[
\begin{align*}
\dot{x}'(t) &= \omega(t) \times x'(t) \\
\dot{y}'(t) &= \omega(t) \times y'(t) \\
\dot{z}'(t) &= \omega(t) \times z'(t)
\end{align*}
\]

Putting these all together:

\[
\dot{\mathbf{R}}(t) = \omega(t) \downarrow \times \mathbf{R}(t)
\]
Recall

\[ a \times b = \begin{pmatrix} a_y b_z - b_y a_z \\ -a_x b_z + b_x a_z \\ a_x b_y - b_x a_y \end{pmatrix} \]

\[ = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix} \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} \]
Summary

• Kinematics: *how does the body move?*

\[ p(t), R(t) \]
\[ \mathbf{v}(t) = \frac{d\mathbf{x}(t)}{dt} = \dot{x}(t) \]
\[ \dot{R}(t) = \omega(t) \times R(t) \]

\[ p(t) = x(t) + R(t)p_0 \]
\[ p(t) = x(t) + \omega(t) \times R(t)p_0 \]
\[ = x(t) + \omega(t) \times (p(t) - x(t)) \]

• Dynamics: *what causes this motion?*
Forces and Torques

- External forces: acting on individual particles
  
  \[ F = \sum_{i} F_i \]

  Net force on body:

- Conservation of linear momentum (Newton’s 2nd law):
  
  \[ p = Mv \quad \quad \dot{p} = F \]

- Analogous concepts for angular motion
Forces and Torques

- Forces on individual particles generate torques - (consequence of constant inter-particle distance)

Net torque on body:

\[ \tau = \sum_{i} \tau_i = \sum_{i} r_i \times f_i \]
Forces and Torques

- Forces on individual particles generate torques (consequence of constant inter-particle distance)

  Net torque on body:

  \[ \tau = \sum_i \tau_i = \sum_i \vec{r}_i \times \vec{f}_i \]

- Conservation of angular momentum:

  \[ \vec{L} = I \vec{\omega} \quad \dot{\vec{L}} = \tau \]

- What is \(\vec{r}\)?
  - Moment of Inertia tensor
The Inertia Tensor

- Analogous to mass, but for rotational motions
  - quantifies distribution of mass as a 2nd order tensor

\[ I = R I_b R^T \]

\[ I_b = \sum_{i} m_i \left( p_{0i}^T p_{0i} 1 - p_{0i} p_{0i}^T \right) \]

- Body-coords MOI is constant, can be precomputed – easy to look up for common shapes!
- World-coords MOI changes with time!
Conservation of Linear and Angular Momenta

• Linear Momentum:

\[ p = Mv \quad \dot{p} = F \]

• Angular Momentum:

\[ L = I\omega \quad \dot{L} = \tau \]

• Note: they are decoupled!
Numerical Integration

• COM Acceleration $\rightarrow$ Velocity $\rightarrow$ Position
  • Easy: $v_{t+1} = v_t + \Delta t \dot{v} \quad x_{t+1} = x_t + \Delta t v_{t+1}$

• Angular Acceleration $\rightarrow$ Angular Velocity
  • Easy: $\omega_{t+1} = \omega_t + \Delta t \dot{\omega}$

• Angular Velocity to Rotations?
  • A bit trickier: $R_{t+1} = R_t + \Delta t \dot{R}_{t+1}$
Updating Rotations

\[ R_{t+1} = R_t + \Delta t \dot{R}_{t+1} \]
\[ = R_t + \Delta t \omega_{t+1}^* R_t = (I + \Delta t \omega_{t+1}^*) R_t \]

No longer a rotation matrix!

- Option 1: orthonormalize (Gram–Schmidt)
- Option 2: explicitly compute rotation \( R \downarrow \Delta t \) due to spinning with angular speed \( \omega \downarrow t+1 \) for \( \Delta t \) seconds, apply incremental rotations: \( R \downarrow t+1 = R \downarrow \Delta t R \downarrow t \)
- NOTE: same concept applies if other rotation parameterizations (i.e. quaternions) are employed
Computing forces

• Given a set of forces, you know how to compute the motion of a rigid body

• Where do forces come from?
  • User interaction
  • Gravity
Rigid Bodies
Computing forces

- Given a set of forces, you know how to compute the motion of a rigid body
- Where do forces come from?
  - User interaction
  - Gravity
  - Collisions and contacts
Collision Response

Collision Process

Δt

no force

no force
Collision Response

A Soft Collision

force

velocity

$\Delta t$
Collision Response

A Harder Collision

force

velocity

$\Delta t$
Collision Response

A Very Hard Collision

force vs. time

velocity vs. time

Δt
Collision Response

A Rigid Body Collision

Impulsive force $f_{imp} = \infty$

$\Delta t = 0$

Velocity
Collision Response

“Nonconvex Rigid Bodies with Stacking”, Guendelman et al., SIGGRAPH 2003
Computing forces

- Given a set of forces, you know how to compute the motion of a rigid body
- Where do forces come from?
  - User interaction
  - Gravity
  - Collisions and contacts
    - Easiest way to model: “spring” penalty forces
  - Articulation
Articulated Rigid Body Dynamics
Computing forces

• Given a set of forces, you know how to compute the motion of a rigid body

• Where do forces come from?
  • User interaction
  • Gravity
  • Collisions and contacts
    ▪ Easiest way to model: “spring” penalty forces
  • Articulation
    ▪ Easiest way to model: “spring” penalty forces
Artistic control over rigid body simulations

Many-Worlds Browsing for Control of Multibody Dynamics  Twigg and James, 2007
Many Worlds Browsing…

Sampling Plausible Worlds

compute and apply impulse

$V_t$ $V_{t+1}$

[O’Sullivan et al., 2003]
Many Worlds Browsing...

Interactive Browsing – various criteria
Many Worlds Browsing
For more information

An Introduction to Physically Based Modeling:
- http://www.cs.cmu.edu/~baraff/pbm/
What you should know:

- What parameters are needed to specify the configuration of a rigid body? What are the time derivatives of these parameters?
- Suppose we represent rigid body state as position, orientation, linear momentum, and angular momentum. What is the time derivative of this state vector as a function of forces and torques acting on the body?
- How do we use these derivatives to advance the position and orientation forward in time using Forward Euler?
- Express the time derivative of a rotation matrix using two methods.
- If we represent a rigid body as a collection of particles, what is its mass? Its center of mass?
- Sketch out a block diagram for simulating a rigid body similar to the block diagram used for simulating particles.
- Is inertia of a rigid body constant in world coordinates? If not, how do we compute it?
- Give two methods for handling collisions (computing collision forces).