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)

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

- \( p(t) \) is the position of point \( p \) at time \( t \).
- \( x(t) \) is the position of the body's center of mass at time \( t \).
- \( R(t) \) is the orientation matrix at time \( t \).

\( x(t) \in \mathbb{R}^3 \)
\( R(t) \in \mathbb{R}^{3\times3} \)
Body Orientation

Rotates vectors from body to world coordinates

- Columns of $R(t)$ encode world coordinates of body a

$$R(t) = [x', y', z']$$
Center of Mass (COM)

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

Geometric center: \(\sum m_i p_{0i} := (0,0,0)\)

\[ M = \sum m_i \]

\[ \text{COM: } \frac{\sum m_i p_i(t)}{M} = x(t) \]
Body velocities

How do the COM position and orientation change with time?

Linear velocity: \( \mathbf{v}(t) = \frac{d\mathbf{x}(t)}{dt} = \dot{\mathbf{x}}(t) \)

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

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

What is the relationship between \( \omega(t) \) and \( \mathbf{R}(t) \)?
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 \( \dot{\mathbf{R}}(t) = \frac{d\mathbf{R}(t)}{dt} \)?

\[
\dot{\mathbf{R}}(t) = \frac{d\mathbf{R}(t)}{dt} = \omega(t) \times \mathbf{r}(t)
\]
Angular Velocity

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

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

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

\[ a \times b = \begin{pmatrix} a_y b_z - b_y a_z \\ -a_x b_z + b_x a_z \\ a_x b_y - b_x a_y \end{pmatrix} = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix} \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} \]

\[ a \times \]
Summary

- **Kinematics**: how does the body move?

  \[ p(t), R(t) \]
  \[ v(t) = \frac{dx(t)}{dt} = \dot{x}(t) \]
  \[ \dot{R}(t) = \omega(t) \times R(t) \]
  \[ p(t) = x(t) + R(t)p_0 \]
  \[ \dot{p}(t) = \dot{x}(t) + \omega(t) \times R(t)p_0 \]
  \[ = \dot{x}(t) + \omega(t) \times (p(t) - x(t)) \]

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

- External forces: acting on individual particles

  Net force on body: \( F = \sum F_i \)

- Conservation of linear momentum (Newton’s 2\(^{nd}\) law):

  \( p = Mv; \dot{p} = F \)

- Analogous concepts for angular motion
Forces and Torques

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

Net torque on body: \( \mathbf{\tau} = \sum \mathbf{\tau}_i = \sum \mathbf{r}_i \times \mathbf{f}_i \)
Forces and Torques

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

Net torque on body: \( \tau = \sum \tau_i = \sum r_i \times f_i \)

• Conservation of angular momentum:
  \[ L = I \omega; \dot{L} = \tau \]

• What is \( I \)?
  • Moment of Inertia tensor
The Inertia Tensor

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

\[ I = R I_b R^T \]
\[ I_b = \sum_i m_i(p_{0i}^T p_{0i} 1 - p_{0i} p_{0i}^T) \]

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

• Linear Momentum:

\[ \mathbf{p} = M \mathbf{v}; \quad \dot{\mathbf{p}} = \mathbf{F}; \quad \mathbf{v} = \frac{1}{M} \mathbf{F} \]

• Angular Momentum:

\[ \mathbf{L} = I \omega; \quad \dot{\mathbf{L}} = \mathbf{\tau}; \quad \dot{\omega} = I^{-1}(\mathbf{\tau} - \omega \times I \omega) \]

• Note: they are decoupled!
Numerical Integration

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

- 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

\[ R_{t+1} = R_t + \Delta t \hat{R}_{t+1} \]

\[ = R_t + \Delta t \omega_{t+1} \times R_t = (I + \Delta t \omega_{t+1} \times) R_t \]

No longer a rotation matrix!

- Option 1: orthonormalize (Gram–Schmidt)
- Option 2: explicitly compute rotation \( R_{\Delta t} \) due to spinning with angular speed \( \omega_{t+1} \) for \( \Delta t \) seconds, apply incremental rotations: \( R_{t+1} = R_{\Delta t} R_t \)

- 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**

$\Delta t$

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
- Velocity

$\Delta t$
Collision Response

A Rigid Body Collision

impulsive force

velocity

\( f_{imp} = \infty \)

\( \Delta t = 0 \)
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

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

Interactive Browsing – various criteria

Input

Positive

Negative
Many Worlds Browsing
For more information

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