CS-184: Computer Graphics

Lecture #22: Rigid Body Dynamics

Prof. James O'Brien
University of California, Berkeley

V2006-F-22-1.0

Today

- Rigid-body dynamics
- Articulated systems

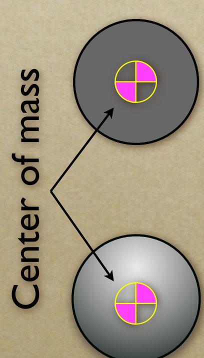
A Rigid Body

- A solid object that does not deform
  - Consists of infinite number of infinitesimal mass points...
  - ...that share a single RB transformation
    - Rotation + Translation (no shear or scale)

$$\boldsymbol{x}^W = \boldsymbol{R} \cdot \boldsymbol{x}^L + \boldsymbol{t}$$

    - Rotation and translation vary over time
  - Limit of deformable object as $k_S \rightarrow \infty$

A Rigid Body



In 2D:

Translation 2 "directions"

Rotation 1 "direction"

3 DOF Total

In 3D:

Translation 3 "directions"

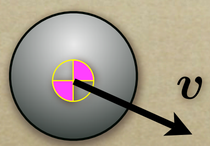
Rotation 3 "direction"

6 DOF Total

Translation and rotation are *decoupled*

2D is boring... we'll stick to 3D from now on...

Translational Motion



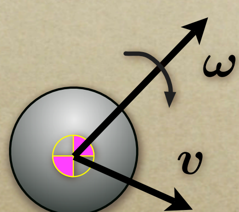
Just like a point mass:

$$\dot{\boldsymbol{p}} = \boldsymbol{v}$$

$$\dot{\boldsymbol{v}} = \boldsymbol{a} = \boldsymbol{f}/m$$

Note: Recall discussion on integration...

Rotational Motion



Rotation gets a bit odd, as well see...

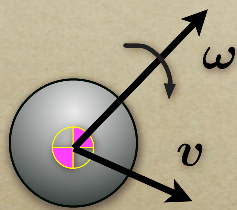
Rotational "position" $\boldsymbol{R}$

- Rotation matrix
- Exponential map
- Quaternions

Rotational velocity $\boldsymbol{\omega}$

- Stored as a vector
- (Also called angular velocity...)
- Measured in radians / second

Rotational Motion



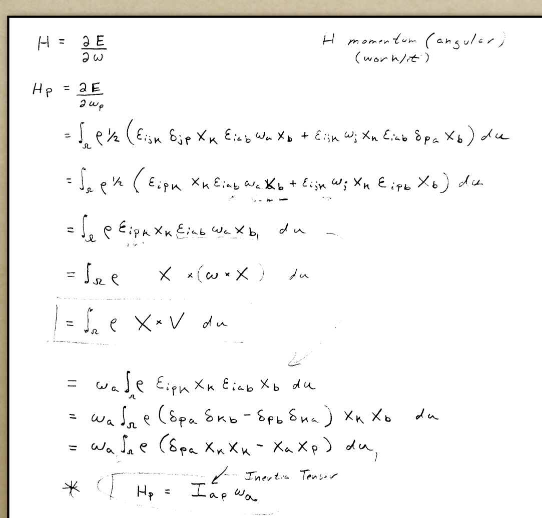
Kinetic energy due to rotation:

"Sum energy (from rotation) over all points in the object"

$$E = \int_{\Omega} \frac{1}{2} \rho \, \dot{\boldsymbol{x}} \cdot \dot{\boldsymbol{x}} \; du$$

$$E = \int_{\Omega} \frac{1}{2} \rho ([\omega \times] \boldsymbol{x}) \cdot ([\omega \times] \boldsymbol{x}) \; du$$

Rotational Motion



... momentum
... to linear momentum
... derived from rotational energy

$$\boldsymbol{H} = \int_{\Omega} \rho \, \boldsymbol{x} \times \dot{\boldsymbol{x}} \; du$$

$$\boldsymbol{H} = \int_{\Omega} \rho \, \boldsymbol{x} \times (\omega \times \boldsymbol{x}) \; du$$

$$\boldsymbol{H} = \left(\int_{\Omega} \cdots \; du \right) \boldsymbol{\omega}$$

$$\boldsymbol{H} = \boldsymbol{I} \omega$$

"Inertia Tensor" not identity matrix...

Inertia Tensor

$$\mathbf{I} = \int_\Omega \rho \begin{bmatrix} y^2+z^2 & -xy & -xz \\ -xy & z^2+x^2 & -yz \\ -xz & -yz & x^2+y^2 \end{bmatrix} du$$

See example for simple shapes at
http://scienceworld.wolfram.com/physics/MomentofInertia.html

Can also be computed from polygon models by transforming volume integral to a surface one.
See paper/code by Brian Mirtich.

Rotational Motion

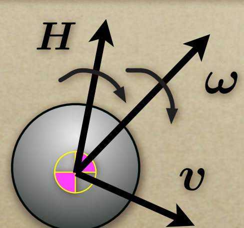


Figure is a lie if this really is a sphere...

Conservation or momentum:

$$\boldsymbol{H}^W = \boldsymbol{I}^W \boldsymbol{\omega}^W$$

$$\boldsymbol{H}^W = \boldsymbol{R}\boldsymbol{I}^L\boldsymbol{R}^\mathsf{T}\boldsymbol{\omega}^W$$

$$\dot{\boldsymbol{H}}^W = \dot{\boldsymbol{R}}\boldsymbol{I}^L\boldsymbol{R}^\mathsf{T}\boldsymbol{\omega}^W + \boldsymbol{R}\boldsymbol{I}^L\dot{\boldsymbol{R}}^\mathsf{T}\boldsymbol{\omega}^W + \boldsymbol{R}\boldsymbol{I}^L\boldsymbol{R}^\mathsf{T}\boldsymbol{\alpha}^W$$

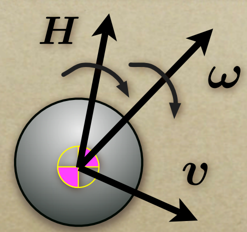
$$\dot{\boldsymbol{H}}^W = 0$$

$$\dot{\boldsymbol{R}} = \boldsymbol{\omega} \times \boldsymbol{R}$$

$$\boldsymbol{\alpha}^W = (\boldsymbol{R}\boldsymbol{I}^L\boldsymbol{R}^\mathsf{T})^{-1}(-\boldsymbol{\omega}^W \times \boldsymbol{H}^W)$$

In other words, things wobble when they rotate.

Rotational Motion



$$\dot{\boldsymbol{R}} = [\omega \times] \boldsymbol{R}$$

$$\dot{\boldsymbol{\omega}} = \boldsymbol{\alpha}$$

Figure is a lie if this really is a sphere...

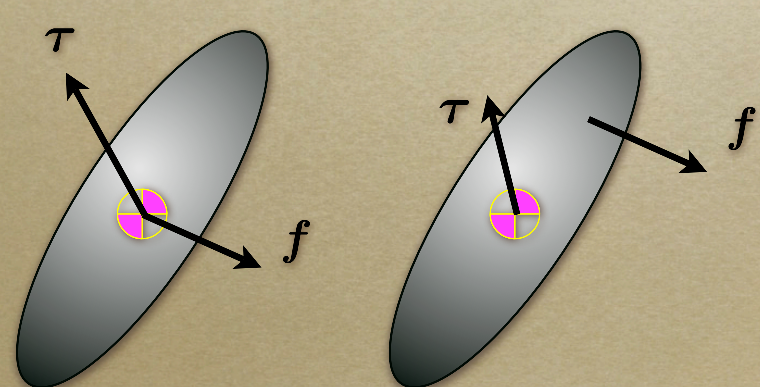
$$\boldsymbol{\alpha}^W = (\boldsymbol{R}\boldsymbol{I}^L\boldsymbol{R}^\top)^{-1} \left((-\boldsymbol{\omega}^W \times \boldsymbol{H}^W) + \boldsymbol{\tau} \right)$$

$$\boldsymbol{\tau} = \boldsymbol{f} \times \boldsymbol{x}$$

Take care when integrating rotations, they need to stay rotations.

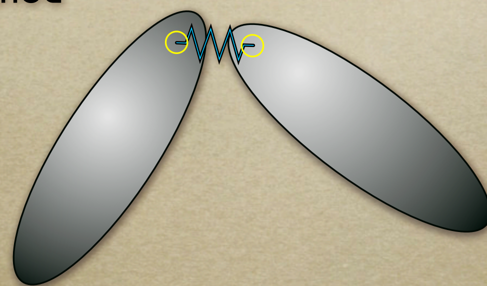
Couples

- A force / torque pair is a couple
  - Also a wrench (I think)
- Many couples are equivalent



Constraints

- Simples method is to use spring attachments
  - Basically a penalty method



  - Spring strength required to get good results may be unreasonably high
    - There are ways to cheat in some contexts...

Constraints

- Articulation constraints
  - Spring trick is an example of a full coordinate method
    - Better constraint methods exist
  - Reduced coordinate methods use DOFs in kinematic skeleton for simulation
    - Much more complex to explain
- Collisions
  - Penalty methods can also be used for collisions
  - Again, better constraint methods exist

Suggested Reading

- Brian Mirtich, ``Fast and Accurate Computation of Polyhedral Mass Properties,'' Journal of Graphics Tools, volume 1, number 2, 1996. http://www.cs.berkeley.edu/~jfc/mirtich/papers/volInt.ps

- Brian Mirtich and John Canny, ``Impulse-based Simulation of Rigid Bodies,'' in Proceedings of 1995 Symposium on Interactive 3D Graphics, April 1995. http://www.cs.berkeley.edu/~jfc/mirtich/papers/ibsrb.ps

- D. Baraff. Linear-time dynamics using Lagrange multipliers. Computer Graphics Proceedings, Annual Conference Series: 137-146, 1996. http://www.pixar.com/companyinfo/research/deb/sig96.pdf

- D. Baraff. Fast contact force computation for nonpenetrating rigid bodies. Computer Graphics Proceedings, Annual Conference Series: 23-34, 1994. http://www.pixar.com/companyinfo/research/deb/sig94.pdf