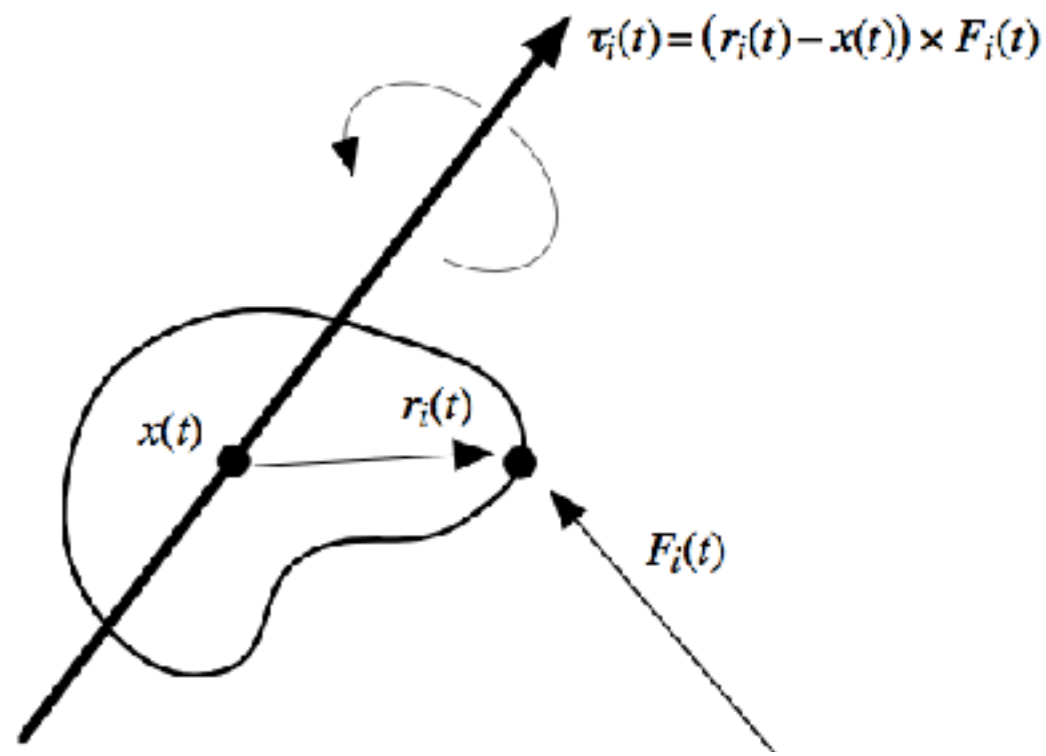
Rigid body dynamics

Rigid body dynamics

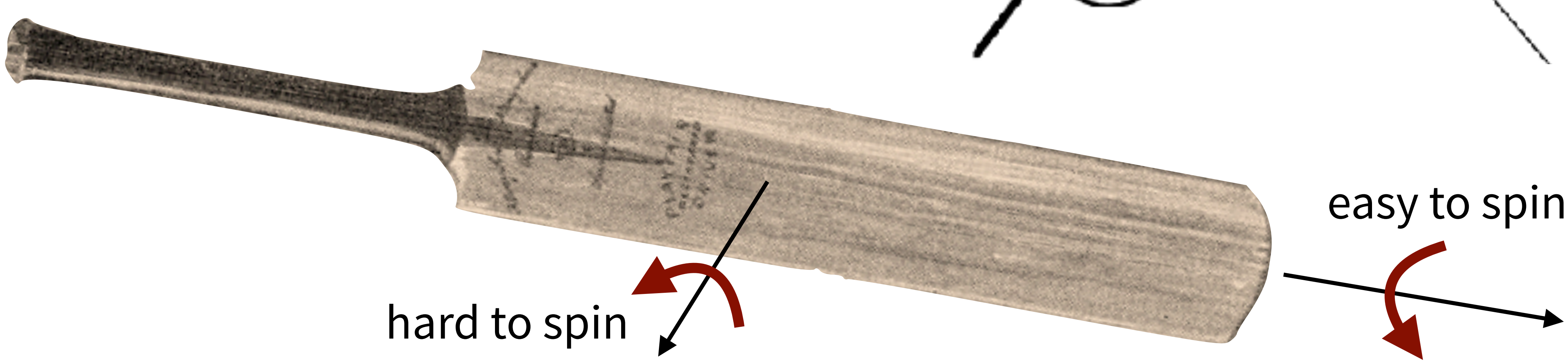
$d\mathbf{v}/dt = m^{-1}\,\mathbf{f}$

$d\boldsymbol{\omega}/dt = ?$

Force acting off-center creates rotational ***torque***, $\boldsymbol{\tau} = (\mathbf{p} - \mathbf{x}) \times \mathbf{f}$

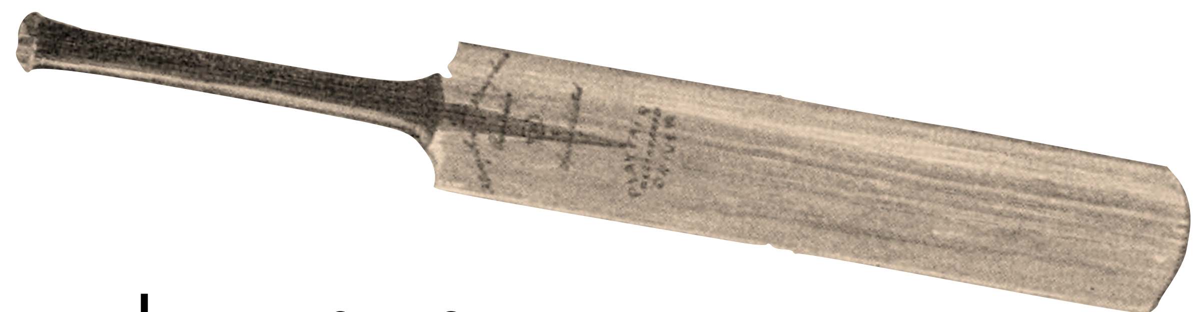


Effect on $\boldsymbol{\omega}$ depends on mass distribution:



Rigid body dynamics

Mass distribution can be encapsulated by the ***inertia matrix***



$$\mathbf{I} = \int [\mathbf{r}]_\times^T [\mathbf{r}]_\times \,\mathrm{d}m$$

$$= \int \left((\mathbf{r}^T\mathbf{r})\mathbf{1} - \mathbf{r}\mathbf{r}^T\right) \mathrm{d}m \quad \text{where } \mathbf{r} = \mathbf{p} - \mathbf{x}$$

Rotational KE = ½ $\boldsymbol{\omega}^T \mathbf{I}\, \boldsymbol{\omega}$, angular momentum $\mathbf{L} = \mathbf{I}\, \boldsymbol{\omega}$

Equations of motion:

$$\mathrm{d}/\mathrm{d}t\, (m\, \mathbf{v}) = \mathbf{f}$$
$$\mathrm{d}/\mathrm{d}t\, (\mathbf{I}\, \boldsymbol{\omega}) = \boldsymbol{\tau}$$

(Full derivation in Witkin & Baraff appendix)

Time integration issues

$$\mathrm{d}/\mathrm{d}t\,(m\,\mathbf{v}) = \mathbf{f}$$
$$\mathrm{d}/\mathrm{d}t\,(\mathbf{I}\,\boldsymbol{\omega}) = \boldsymbol{\tau}$$

Can we reduce this to $m\,\dot{\mathbf{v}} = \mathbf{f}$ and $\mathbf{I}\,\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}$?

No, $\mathbf{I}$ depends on current orientation!

$$\mathbf{I} = \mathbf{R}\,\mathbf{I}_0\,\mathbf{R}^\mathsf{T}$$

inertia in body space

Actually, $\mathrm{d}/\mathrm{d}t\,(\mathbf{I}\,\boldsymbol{\omega}) = \mathbf{I}\,\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\,\boldsymbol{\omega}$

Time integration issues

$$\mathrm{d}/\mathrm{d}t\,(m\,\mathbf{v}) = \mathbf{f}$$
$$\mathrm{d}/\mathrm{d}t\,(\mathbf{I}\,\boldsymbol{\omega}) = \boldsymbol{\tau}$$

Two options:

- Store and update $\mathbf{L} = \mathbf{I}\,\boldsymbol{\omega}$ instead: $\mathrm{d}\mathbf{L}/\mathrm{d}t = \boldsymbol{\tau}$

  Recover $\boldsymbol{\omega} = \mathbf{I}^{-1}\,\mathbf{L} = \mathbf{R}^{\mathsf{T}}\,\mathbf{I}_0^{-1}\,\mathbf{R}\,\mathbf{L}$ when needed

- Use expansion of $\mathrm{d}/\mathrm{d}t\,(\mathbf{I}\,\boldsymbol{\omega})$

$$\dot{\mathbf{v}} = m^{-1}\,\mathrm{f}$$
$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1}\,(\boldsymbol{\tau} - \boldsymbol{\omega} \times \mathbf{I}\,\boldsymbol{\omega})$$

"gyroscopic term"

The inertia matrix

$$\mathbf{I} = \int \left((\mathbf{r}^T \mathbf{r})\mathbf{1} - \mathbf{r}\mathbf{r}^T \right) \mathrm{d}m$$

How to compute $\mathbf{I}$?

- ***Simple shapes*** (sphere, box, cylinder, …): look up formula
- ***Polygon meshes***: Apply Green's theorem to reduce integral to sum over surface polygons [Mirtich, "Fast and Accurate Computation of Polyhedral Mass Properties", 1996]

Eigenvectors of $\mathbf{I}$ = ***principal axes***

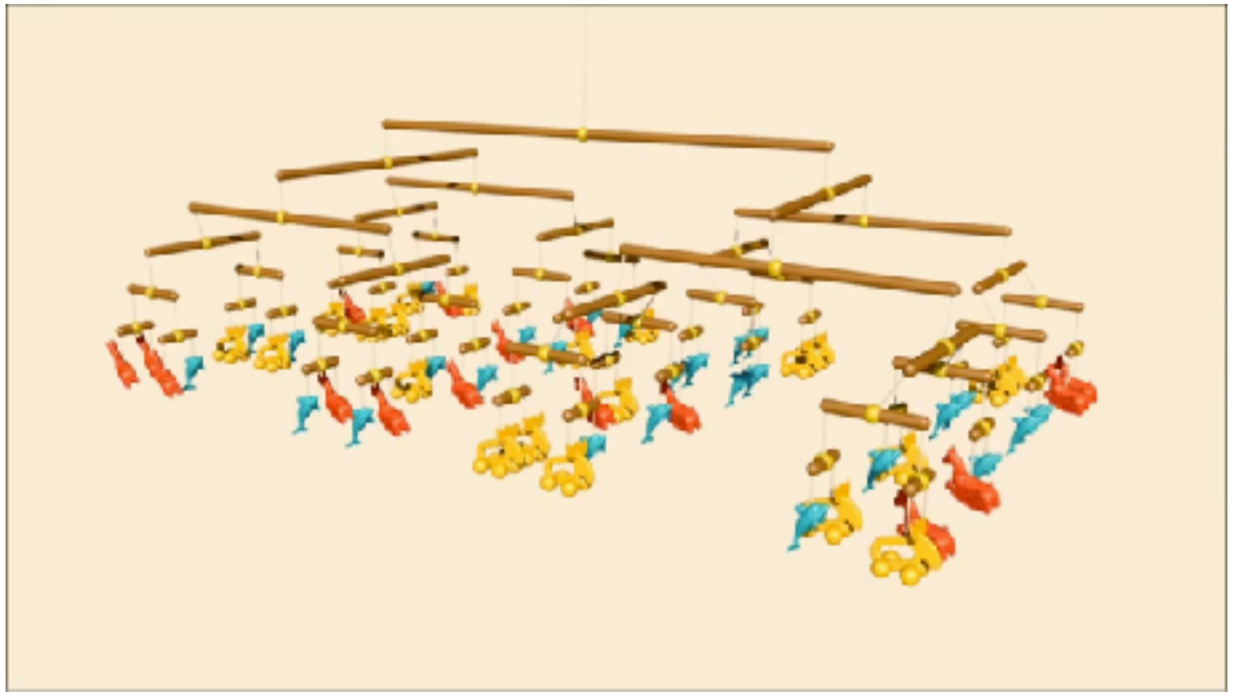
- Choose reference pose aligned with principal axes
  $\Rightarrow$ $\mathbf{I}_0$ is diagonal, easy to invert

Articulated bodies

Articulated bodies

Collection of rigid bodies with joint constraints

- Chains of objects
- Ragdoll physics



[Bender et al. 2014]

[Secret Exit]

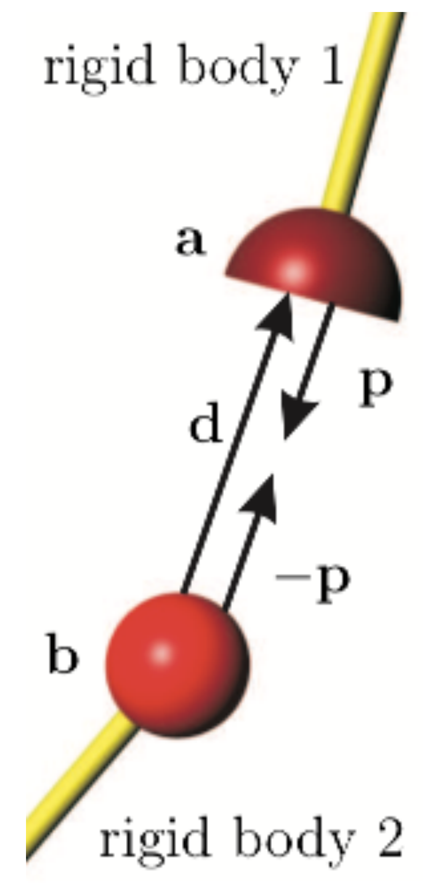
Articulated bodies

Collection of rigid bodies with joint constraints

e.g. spherical joint: body 1 point **a** = body 2 point **b**

$$\mathbf{g}(\mathbf{q}_1, \mathbf{q}_2) = \mathbf{a}(\mathbf{q}_1) - \mathbf{b}(\mathbf{q}_2)$$
$$= (\mathbf{x}_1 + \mathbf{R}_1\, \mathbf{a}_0) - (\mathbf{x}_2 + \mathbf{R}_2\, \mathbf{b}_0)$$

Details in Bender et al. [2012], Sec. 3.2

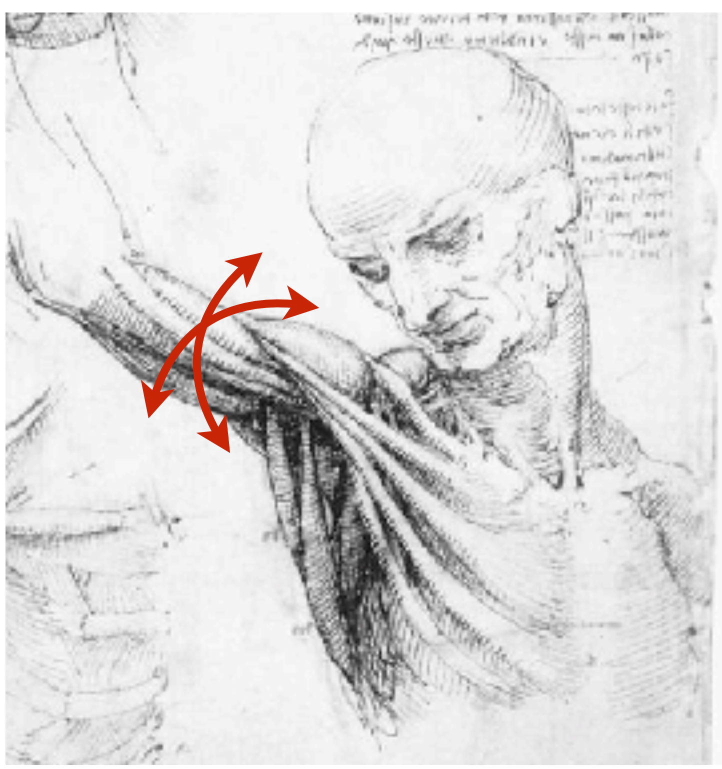


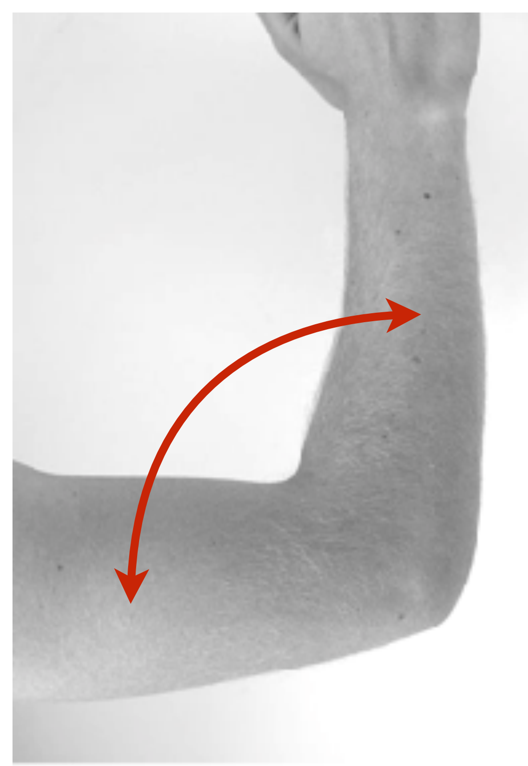
[Bender et al. 2012]

Joints

Common examples of joints (*m* = #DOFs removed):

- **Spherical** / **ball** (*m* = 3): permits arbitrary rotation
- **Revolute** / **hinge** (*m* = 5): permits rotation about one axis
- **Prismatic** / **slider** (*m* = 5): permits translation along one axis





Simulating articulated bodies

- Reduced coordinates: ***Featherstone's algorithm***
  - Works only for tree-structured systems
  - Detailed explanation given in Mirtich's PhD thesis [1996]
- ***Penalty forces***, ***projection methods***
  - More flexible, modular (can add / remove joints dynamically)
  - Require constraint Jacobian d**g**/d**q**

Joint constraint Jacobian

More convenient to define $\mathbf{J}$ such that $\dot{\mathbf{g}} = \mathbf{J}\,\mathbf{u}$

- Conceptually, $\mathbf{J} = (d\mathbf{g}/d\mathbf{q})\,\mathbf{H}$
- Constraint forces and torques: $(\mathbf{f}_c, \boldsymbol{\tau}_c) = \mathbf{J}^T \boldsymbol{\lambda}$

Example: ball joint $\mathbf{a}(\mathbf{q}_1) - \mathbf{b}(\mathbf{q}_2)$

$$\dot{\mathbf{g}} = (\mathbf{v}_1 + \boldsymbol{\omega}_1 \times (\mathbf{a} - \mathbf{x}_1)) - (\mathbf{v}_2 + \boldsymbol{\omega}_2 \times (\mathbf{b} - \mathbf{x}_2))$$

$$= \begin{bmatrix} \mathbf{1}_{3\times 3} & -[\mathbf{a} - \mathbf{x}_1]_\times & -\mathbf{1}_{3\times 3} & [\mathbf{b} - \mathbf{x}_2]_\times \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 \\ \boldsymbol{\omega}_1 \\ \mathbf{v}_2 \\ \boldsymbol{\omega}_2 \end{bmatrix}$$