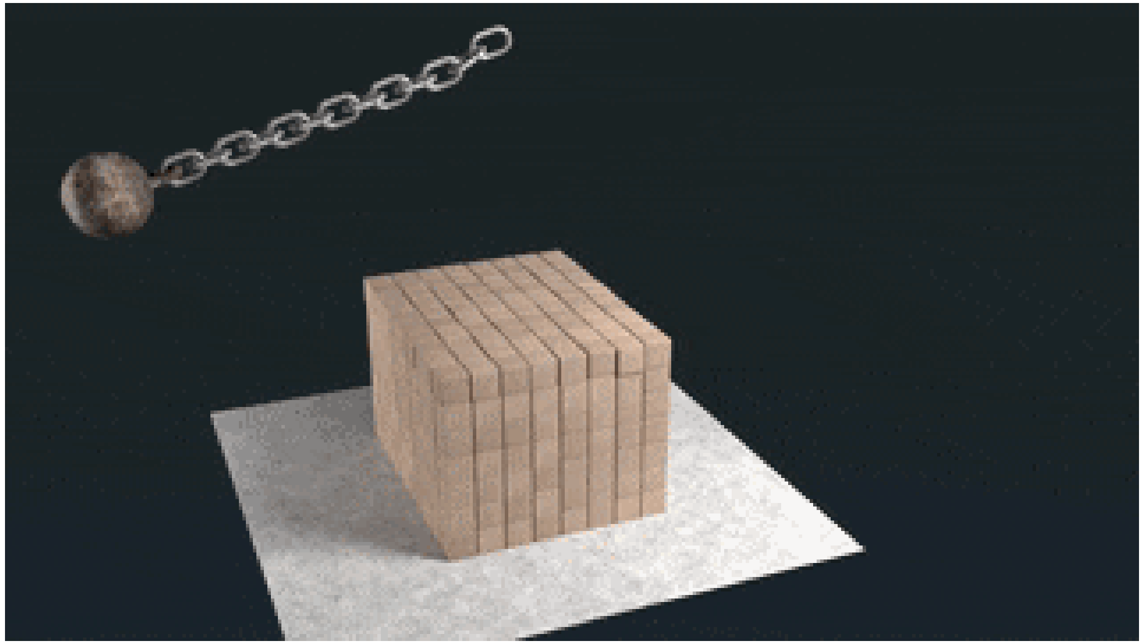


Rigid Body Dynamics

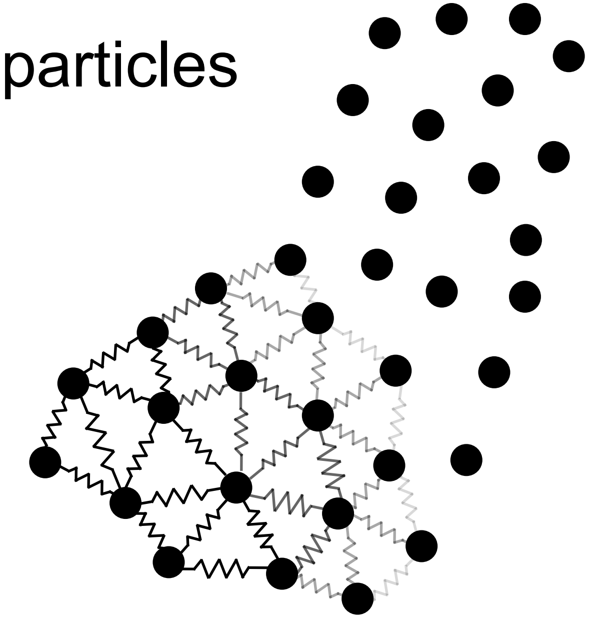
and a brief introduction to rotation representations

Learning objectives

- Understand rigid body representations and core physical concepts
  - Center of mass, local and global coordinate frames, 3d rotation representations, etc
  - Linear angular momenta, mass and moment of inertia, forces and torques, etc
- Understand the differential equations underlying rigid body dynamics
- Learn typical numerical integration schemes for rigid bodies

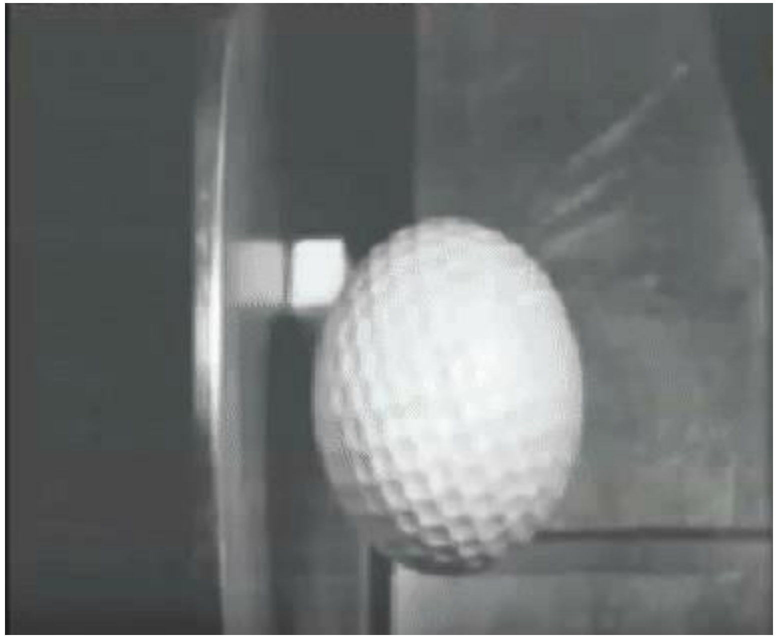
Particle Dynamics

- We've seen what it takes to simulate particles:
  - Keep track of positions **x** and velocities **v**
  - Underlying ODE: f = m**a**
- Let's assume intermolecular forces couple the motion of these particles
- As these forces get stronger…
  - How can you ensure the simulation remains numerically stable?
  - What happens in the limit?



ETH zürich

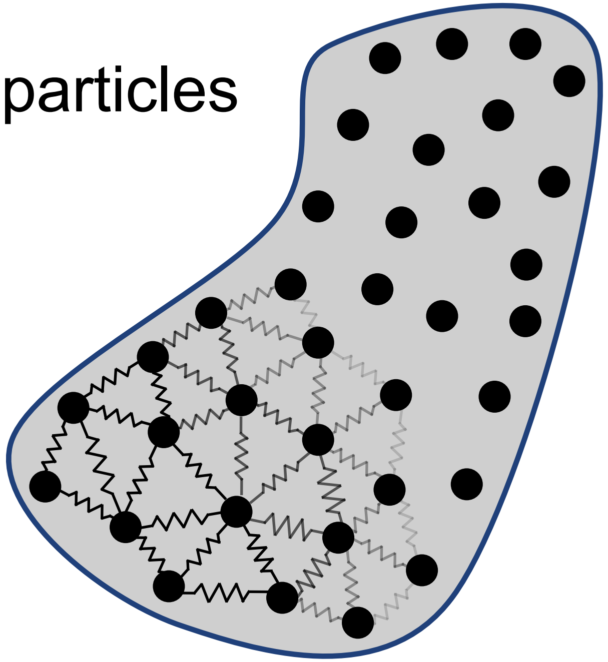
No object is truly rigid, but this is nevertheless a VERY convenient approximation!



CRL

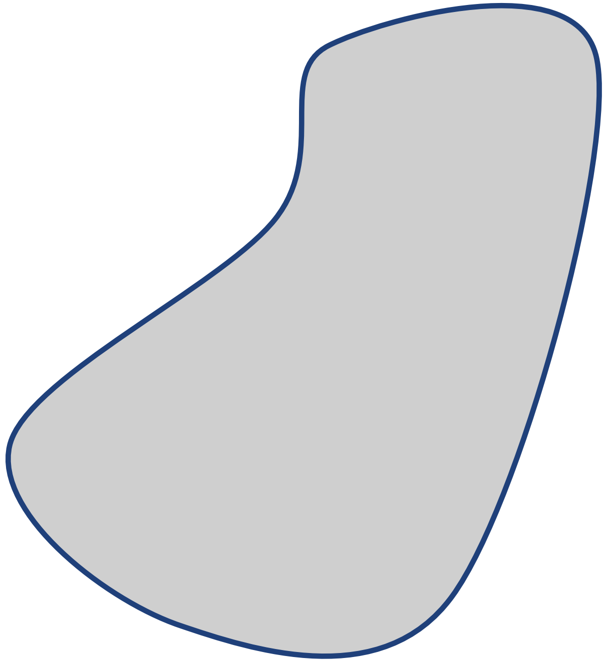
Particles moving through space

- We’ve seen what it takes to simulate particles:
  - Keep track of positions **x** and velocities **v**
  - Underlying ODE: f = m**a**
- Let’s assume intermolecular forces couple the motion of these particles
- As these forces get stronger…
  - How can you ensure the simulation remains numerically stable?
  - What happens in the limit?
- Rather than simulating lots of particles connected via stiff springs, we will model the entire group as one *rigid body*



What is a rigid body?

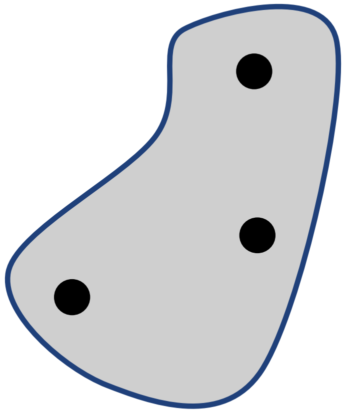
- Rigid Body: an ensemble of particles that cannot move relative to each other
  - Distance between any two particles on a rigid body remains constant
  - Shape does not change



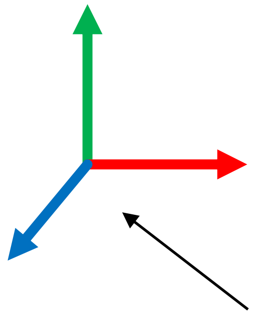
- How can a rigid body move through space?
  - It can translate
  - It can rotate
  - Or a composition of the two

- As we model the motion of the rigid body, we must therefore keep track of its *position* and *orientation* relative to a global frame of reference

Local and global coordinate frames



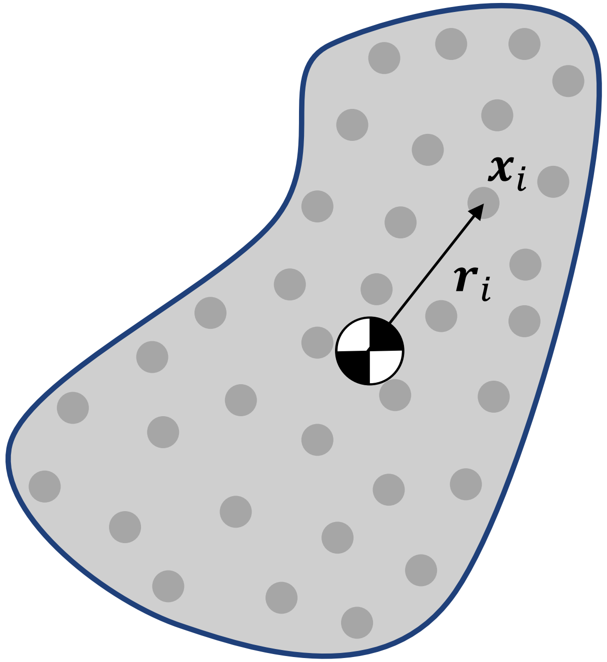
- What is the position of this rigid body in the global coordinate frame?
  - Every point on the rigid body has different world frame coordinates…
- Choose one point on the rigid body and store its position explicitly
  - Typically, this is the *center of mass*



global or **world**
coordinate frame

Center of mass: $\boldsymbol{p}$

- Think of it as the mass-weighted geometric center of the rigid body:



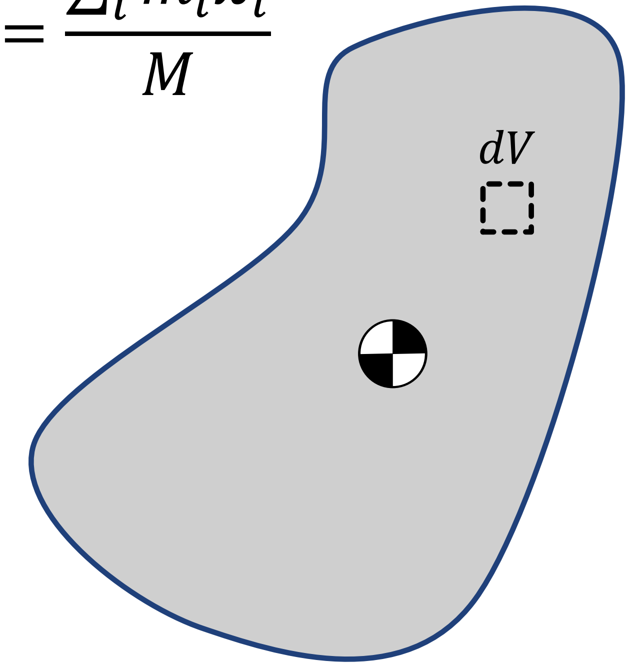
Center of mass: $\boldsymbol{p}$

- Think of it as the mass-weighted geometric center of the rigid body:

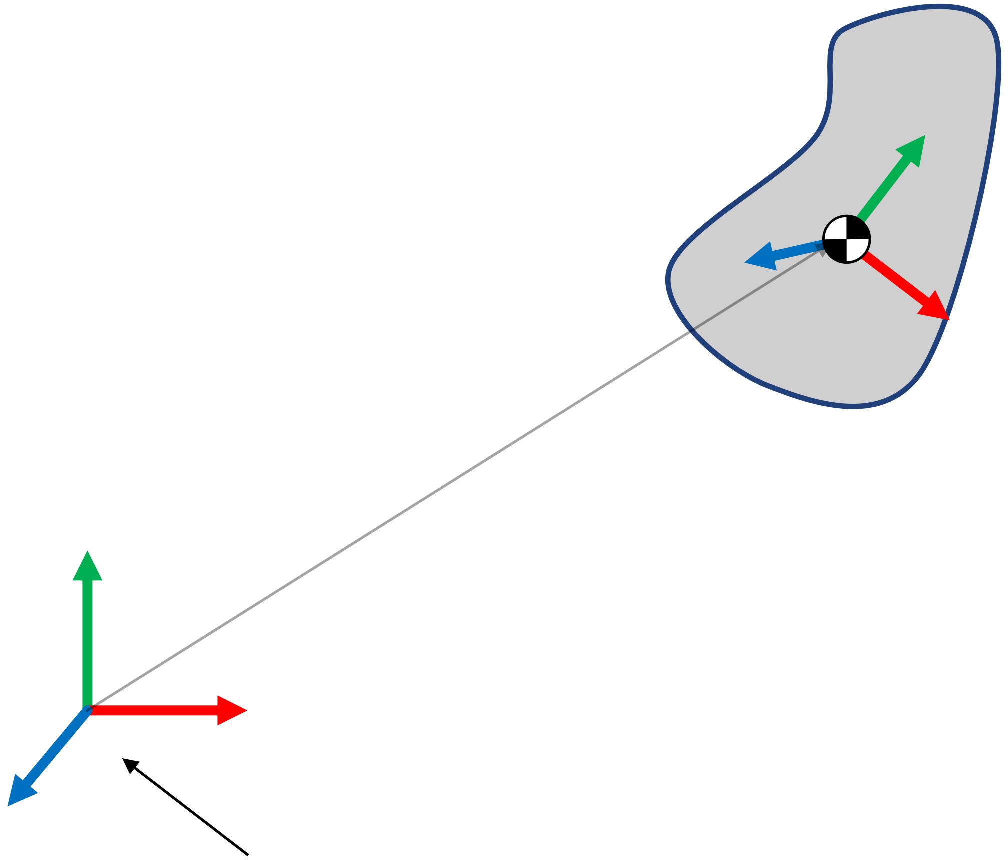
$$\sum_i m_i \boldsymbol{r}_i = \boldsymbol{0} \Rightarrow \sum_i m_i \overrightarrow{(\boldsymbol{p}, \boldsymbol{x}_i)} = \sum_i m_i (\boldsymbol{x}_i - \boldsymbol{p}) = \boldsymbol{0} \Rightarrow \boldsymbol{p} = \frac{\sum_i m_i \boldsymbol{x}_i}{M}$$

- In general:

$$\boldsymbol{p} = \frac{\int_\Omega \boldsymbol{x}_{dV} \overbrace{\rho}^{\text{local density}} dV}{\underbrace{\int_\Omega \rho dV}_{\text{total mass}}}$$



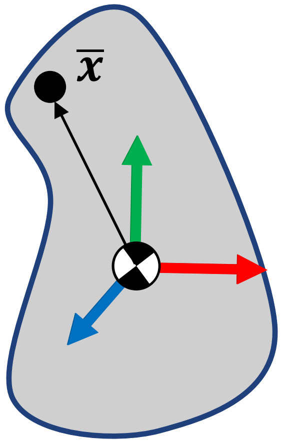
Local and global coordinate frames



- Choose one point on the rigid body and store its position explicitly
  - Typically, this is the *center of mass* $\boldsymbol{p}(t)$
- This point will act as the *origin* of the RB's local coordinate frame (i.e. it has local coordinates (0, 0, 0))
- Also need an orientation $\boldsymbol{R}(t)$ which tells us how the rigid body has rotated relative to the world frame
- $\boldsymbol{R}$ transforms vectors from local frame to global coordinates!
- Using $\boldsymbol{p}(t)$ and $\boldsymbol{R}(t)$ we can compute the world coordinates of any point on the RB

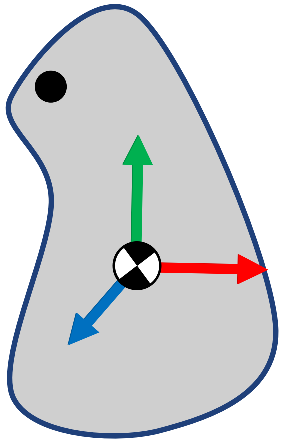
Rigid Body Kinematics

- What are the world coordinates of an arbitrary point $\bar{\boldsymbol{x}}$ that is expressed in the local coordinate frame of the rigid body?



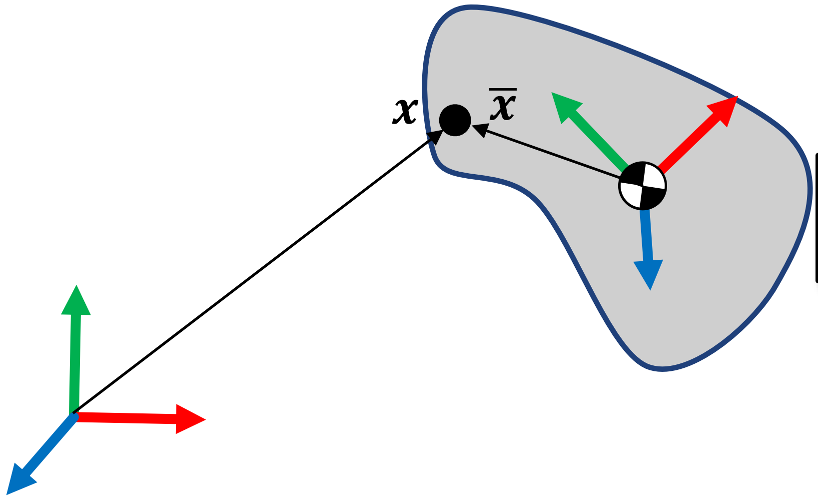
Rigid Body Kinematics

- What are the world coordinates of an arbitrary point $\bar{x}$ that is expressed in the local coordinate frame of the rigid body?



Rigid Body Kinematics

- What are the world coordinates of an arbitrary point $\bar{\boldsymbol{x}}$ that is expressed in the local coordinate frame of the rigid body?



position of the RB's COM in world coordinates

point in local coordinate frame

$$\boldsymbol{x} = \boldsymbol{p} + \boldsymbol{R}\bar{\boldsymbol{x}}$$

same point in world coordinates

Orientation of the RB relative to the global frame of reference

Note: $\boldsymbol{R} = \begin{bmatrix} r_{xx} & r_{yx} & r_{zx} \\ r_{xy} & r_{yy} & r_{zy} \\ r_{xz} & r_{yz} & r_{zz} \end{bmatrix}$

Rigid Body Kinematics

- What is the meaning of $\boldsymbol{p}(t)$?
  - Position of the rigid body's Center of Mass at time t.

- What does $\mathbf{R}(t)$ tell us?
  - Consider the x-axis in body space, (1, 0, 0), what is the direction of this vector in world space at time *t*?

$$\mathbf{R}(t)\begin{bmatrix}1\\0\\0\end{bmatrix} = \begin{bmatrix}r_{xx}\\r_{xy}\\r_{xz}\end{bmatrix}$$

The first column of $\mathbf{R}(t)$

  - Columns of $\mathbf{R}(t)$ encode global coordinates of body space ***x***, ***y*** and ***z*** vectors at time t.

Rigid Body Kinematics

- We know how to compute world coordinates for any point on the RB:

$$\boldsymbol{x}(t) = \boldsymbol{p}(t) + \boldsymbol{R}(t)\overline{\boldsymbol{x}}$$

- How do we compute this point's velocity, $\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{dt}} = \dot{\boldsymbol{x}}(t)$?

$$\dot{\boldsymbol{x}}(t) = \dot{\boldsymbol{p}}(t) + \dot{\boldsymbol{R}}(t)\overline{\boldsymbol{x}}$$

- $\dot{\boldsymbol{p}} \equiv \boldsymbol{v}$ is the velocity of the COM (aka the *linear velocity* of the rigid body)
- $\dot{\boldsymbol{R}}$ is a matrix that tells us how $\boldsymbol{R}$ changes over time
  - But what does it *mean*?

Rigid Body Kinematics – angular velocity

- Assume a rigid body has 0 COM velocity, but it is spinning
  - rotation is about the COM
  - some points on the RB are moving faster than others
  - which points on the RB have 0 velocity (i.e. they don't move in space)?

- Define spin as *angular velocity* – a vector $\boldsymbol{\omega}(t)$
  - Direction of $\boldsymbol{\omega}$ gives the axis (in world coordinates!) that the RB is rotating about
  - Magnitude of $\boldsymbol{\omega}$ encodes the speed with which the RB is spinning (rad/s)

Quiz

- Consider a 2D rigid body that is circling around a fixed point
  - What's the *speed* of the rigid body?

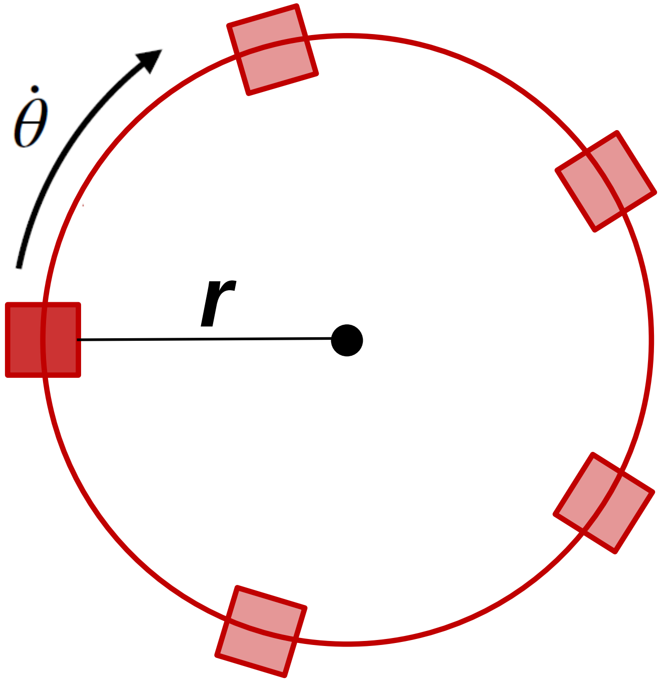
$$|\boldsymbol{v}| = |\boldsymbol{r}||\dot{\theta}|$$

  - And it's velocity?

direction is an in-plane vector orthogonal to **r**

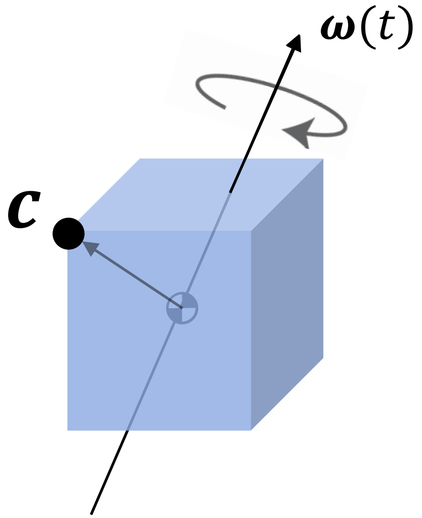
  - What's the angular velocity of the rigid body?

$$\dot{\theta}$$



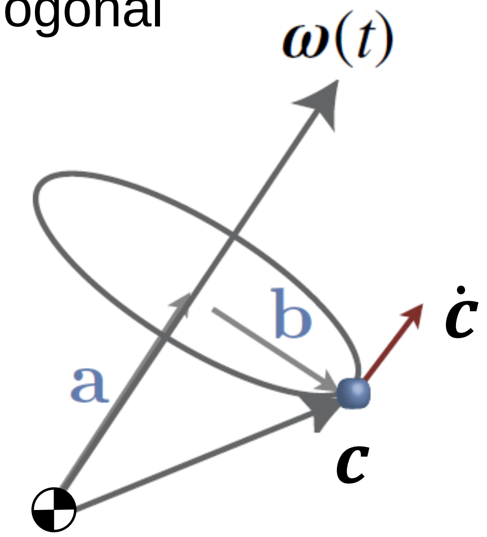
Rigid Body Kinematics

- So, an RB in 3D is spinning in place, about the COM
  - How fast is this corner point moving?



Rigid Body Kinematics

- So, an RB in 3D is spinning in place, about the COM
  - How fast is this corner point moving?
  - Let $\boldsymbol{c}$ be the world coordinates vector from the COM to the point of interest
  - Decompose $\boldsymbol{c}$ into two vectors, $\boldsymbol{a}$ and $\boldsymbol{b}$, one aligned with $\boldsymbol{\omega}$, one orthogonal
    - $\boldsymbol{a}$ does not change over time as RB spins (i.e. $\dot{\boldsymbol{a}} = \mathbf{0}$)
    - but $\boldsymbol{b}$ does:
      - $|\dot{\boldsymbol{b}}| = |\boldsymbol{b}||\boldsymbol{\omega}|$
      - $\dot{\boldsymbol{b}}$ is perpendicular to $\boldsymbol{\omega}$ and to $\boldsymbol{b}$
      - So, $\dot{\boldsymbol{b}} = \boldsymbol{\omega} \times \boldsymbol{b}$
  - Therefore, $\dot{\boldsymbol{c}} = \boldsymbol{\omega} \times \boldsymbol{a} + \boldsymbol{\omega} \times \boldsymbol{b} = \boldsymbol{\omega} \times \boldsymbol{c}$



Rigid Body Kinematics: what is $\dot{R}$?

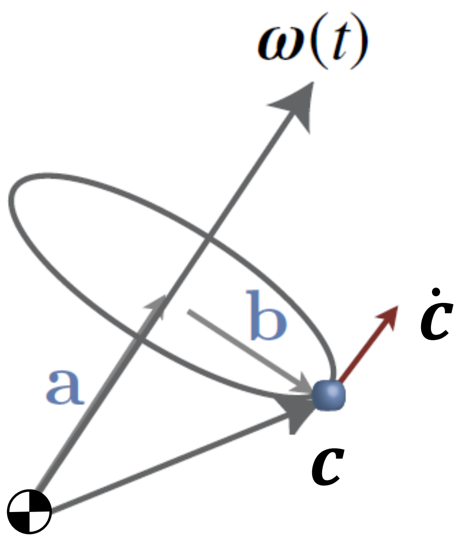
- Recall that columns of $\mathbf{R}(t)$ store world coords of local ***x***, ***y*** and ***z*** axes
  - What do columns of $\dot{\mathbf{R}}$ represent?

$$\begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix}$$

first column of $\mathbf{R}(t)$

$$\dot{\begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix}} = \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix}$$

first column of $\dot{\mathbf{R}}(t)$



  - So:

$$\dot{\mathbf{R}}(t) = \begin{bmatrix} \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix} & \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{yx} \\ r_{yy} \\ r_{yz} \end{bmatrix} & \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} \end{bmatrix}$$

Rigid Body Kinematics

- Consider two vectors in 3D, $\mathbf{a}$ and $\mathbf{b}$. Their cross product is:

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_y b_z - b_y a_z \\ -a_x b_z + b_x a_z \\ a_x b_y - b_x a_y \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}}_{\text{skew-symmetric matrix } [\mathbf{a}]} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = [\mathbf{a}]\mathbf{b}$$

- Therefore:

$$\dot{\mathbf{R}}(t) = \begin{bmatrix} \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix} & \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{yx} \\ r_{yy} \\ r_{yz} \end{bmatrix} & \boldsymbol{\omega}(t) \times \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} [\boldsymbol{\omega}(t)] \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix} & [\boldsymbol{\omega}(t)] \begin{bmatrix} r_{yx} \\ r_{yy} \\ r_{yz} \end{bmatrix} & [\boldsymbol{\omega}(t)] \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} \end{bmatrix} = [\boldsymbol{\omega}(t)]\mathbf{R}(t)$$

Rigid Body Kinematics

- We know how to compute world coordinates for any point on the RB:

$$\boldsymbol{x}(t) = \boldsymbol{p}(t) + \boldsymbol{R}(t)\overline{\boldsymbol{x}}$$

- How do we compute this point's velocity, $\dot{\boldsymbol{x}}(t)$?

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(t) + \dot{\boldsymbol{R}}(t)\overline{\boldsymbol{x}}$$

where $\dot{\boldsymbol{R}}(t) = [\boldsymbol{\omega}(t)]\boldsymbol{R}(t)$, and $\boldsymbol{v} = \dot{\boldsymbol{p}}$. Expanding this out gives us (and leaving out time dependence):

$$\dot{\boldsymbol{x}} = \boldsymbol{v} + [\boldsymbol{\omega}]\boldsymbol{R}\overline{\boldsymbol{x}} = \boldsymbol{v} + \boldsymbol{\omega} \times \boldsymbol{R}\overline{\boldsymbol{x}}$$

$$= \boldsymbol{v} + [\boldsymbol{\omega}](\boldsymbol{R}\overline{\boldsymbol{x}} + \boldsymbol{p} - \boldsymbol{p}) = \boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x} - \boldsymbol{p}))$$

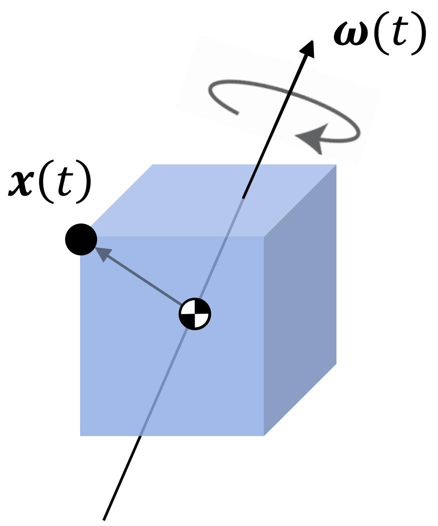
- **Always be careful which coordinate frame quantities are expressed in!!!**

Rigid Body Kinematics

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(t) + \boldsymbol{\omega}(t) \times (\boldsymbol{x}(t) - \boldsymbol{p}(t))$$

linear component

angular component



Do you know how to compute the acceleration, $\ddot{\boldsymbol{x}}(t)$?

Quiz – true or false

- If a (uniform density) cube has non-zero angular velocity:
  - a corner point always moves faster than the COM
  - a corner point can move slower than the COM
  - a corner point always moves at the same speed as the COM

- If a (uniform density) cube has non-zero angular velocity and zero linear velocity
  - the COM may or may not be moving
  - a corner point may or may not be moving

Rigid Body Dynamics

$$"F = ma"$$

Mass

Forces

Accelerations

Kinetic Energy of a Rigid Body

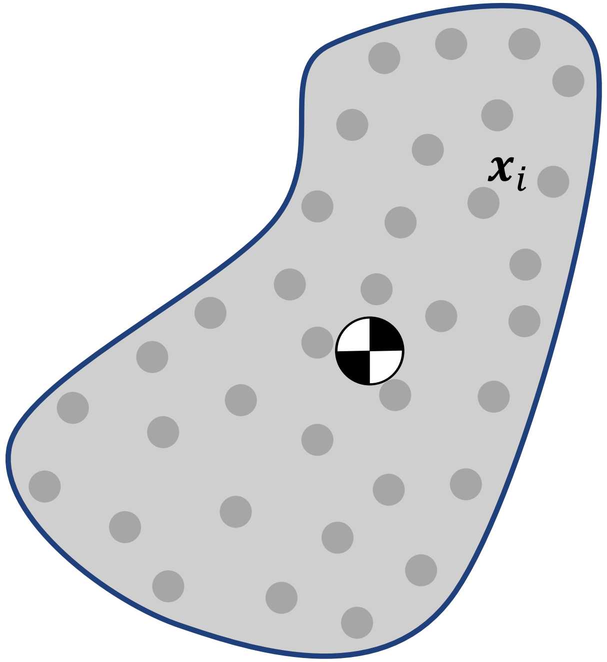
- Recall the kinetic energy of a mass point:

$$K = \frac{1}{2}\dot{\boldsymbol{x}}^T m \dot{\boldsymbol{x}}$$

- What is the kinetic energy of a rigid body?

$$K = \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i$$

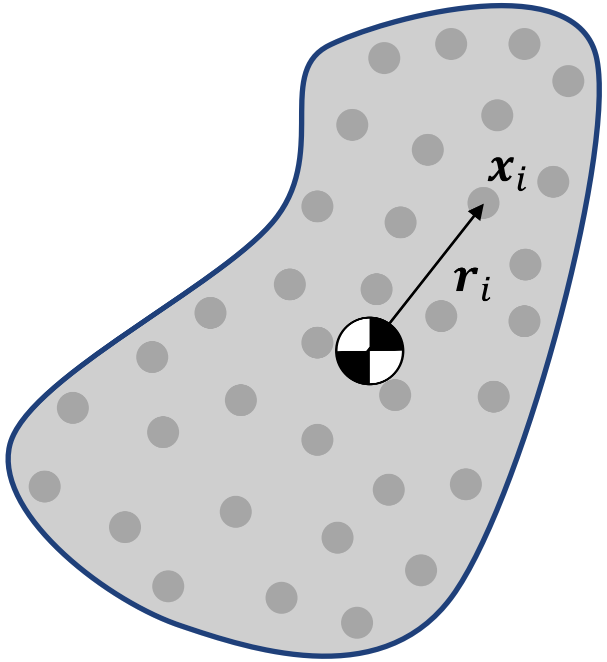
- But recall, $\dot{\boldsymbol{x}}_i = \boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})$, so let's work this out…



Kinetic Energy of a Rigid Body

- Kinetic energy of a rigid body:

$$
\begin{aligned}
K &= \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i \\
&= \frac{1}{2}\sum_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})\right)^T m_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times \overbrace{(\boldsymbol{x}_i - \boldsymbol{p})}^{\boldsymbol{r}_i}\right) \\
&= \frac{1}{2}\sum_i \boldsymbol{v}^T m_i \boldsymbol{v} + 2\,\boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i\,(\boldsymbol{\omega} \times \boldsymbol{r}_i) \\
&= \frac{1}{2}\boldsymbol{v}^T \left(\sum_i m_i\right)\boldsymbol{v} + \sum_i \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + \frac{1}{2}\sum_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)
\end{aligned}
$$



Kinetic Energy of a Rigid Body

- Kinetic energy of a rigid body:

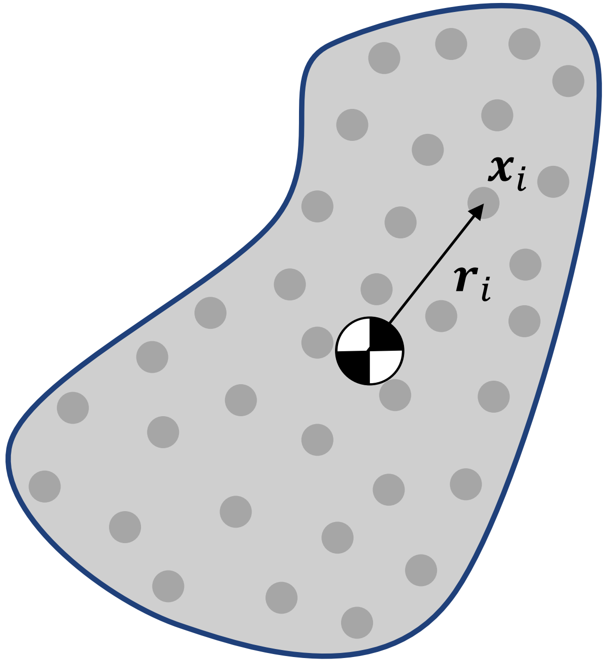
$$K = \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i$$

$$= \frac{1}{2}\sum_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})\right)^T m_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times \overbrace{(\boldsymbol{x}_i - \boldsymbol{p})}^{\boldsymbol{r}_i}\right)$$

$$= \frac{1}{2}\sum_i \boldsymbol{v}^T m_i \boldsymbol{v} + 2\,\boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i\,(\boldsymbol{\omega} \times \boldsymbol{r}_i)$$

$$= \underbrace{\frac{1}{2}\boldsymbol{v}^T \left(\sum_i m_i\right) \boldsymbol{v}}_{} + \sum_i \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + \frac{1}{2}\sum_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)$$

$\frac{1}{2}\boldsymbol{v}^T M \boldsymbol{v}$ – kinetic energy due to the linear motion of the RB's COM



Kinetic Energy of a Rigid Body

- Kinetic energy of a rigid body:

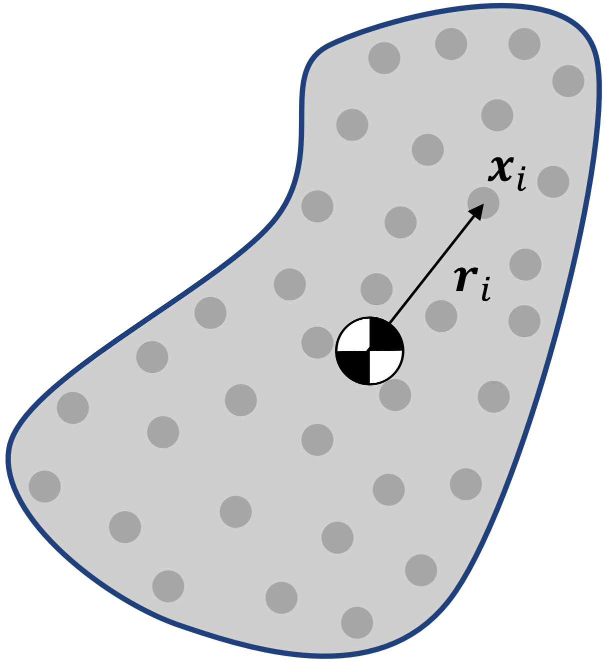
$$K = \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i$$

$$= \frac{1}{2}\sum_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})\right)^T m_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times \overbrace{(\boldsymbol{x}_i - \boldsymbol{p})}^{\boldsymbol{r}_i}\right)$$

$$= \frac{1}{2}\sum_i \boldsymbol{v}^T m_i \boldsymbol{v} + 2\,\boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i\,(\boldsymbol{\omega} \times \boldsymbol{r}_i)$$

$$= \frac{1}{2}\boldsymbol{v}^T \left(\sum_i m_i\right) \boldsymbol{v} + \underbrace{\sum_i \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)} + \frac{1}{2}\sum_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)$$

$$= \boldsymbol{v}^T \sum_i m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) = -\boldsymbol{v}^T \sum_i m_i (\boldsymbol{r}_i \times \boldsymbol{\omega}) = -\boldsymbol{v}^T \left(\sum_i m_i\,\boldsymbol{r}_i\right) \times \boldsymbol{\omega} = 0$$



Kinetic Energy of a Rigid Body

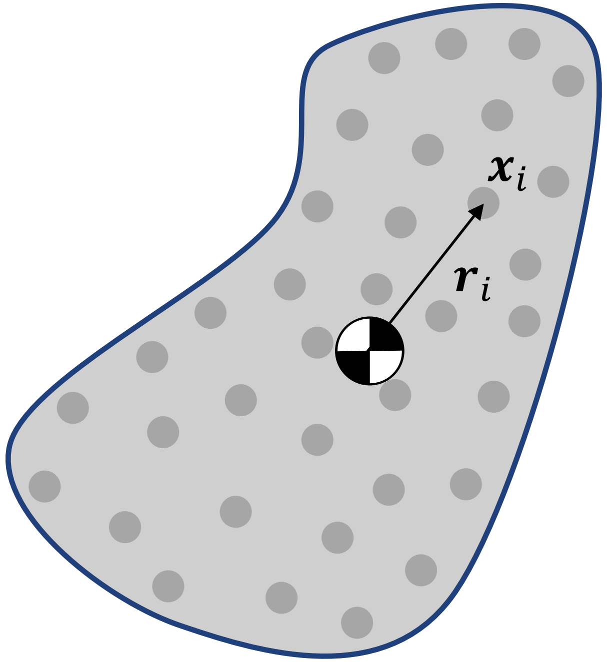
- Kinetic energy of a rigid body:

$$K = \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i$$

$$= \frac{1}{2}\sum_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})\right)^T m_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times \overbrace{(\boldsymbol{x}_i - \boldsymbol{p})}^{\boldsymbol{r}_i}\right)$$

$$= \frac{1}{2}\sum_i \boldsymbol{v}^T m_i \boldsymbol{v} + 2\,\boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i\,(\boldsymbol{\omega} \times \boldsymbol{r}_i)$$

$$= \frac{1}{2}\boldsymbol{v}^T \left(\sum_i m_i\right)\boldsymbol{v} + \sum_i \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + \underbrace{\frac{1}{2}\sum_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)}_{?}$$



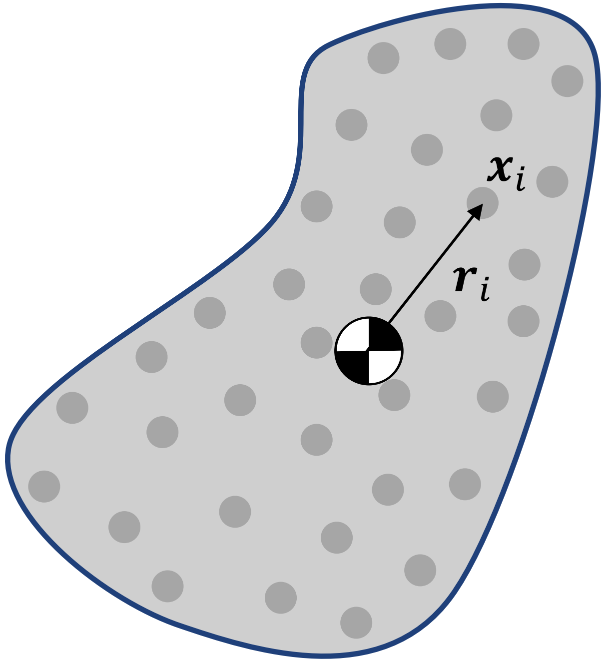
Kinetic Energy of a Rigid Body

$$(\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) = m_i([\boldsymbol{\omega}]\boldsymbol{r}_i)^T [\boldsymbol{\omega}]\boldsymbol{r}_i$$
$$= m_i([\boldsymbol{r}_i]\boldsymbol{\omega})^T [\boldsymbol{r}_i]\boldsymbol{\omega}$$
$$= \boldsymbol{\omega}^T m_i [\boldsymbol{r}_i]^T [\boldsymbol{r}_i]\boldsymbol{\omega}$$

Recall: if $\boldsymbol{r}_i = (x, y, z)^T$, then $[\boldsymbol{r}_i] = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$, so

$$[\boldsymbol{r}_i]^T [\boldsymbol{r}_i] = -\begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix} * \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$
$$= \begin{bmatrix} y^2 + z^2 & -xy & -xz \\ -xy & x^2 + z^2 & -yz \\ -xz & yz & x^2 + y^2 \end{bmatrix} = \boldsymbol{r}_i^T \boldsymbol{r}_i \mathbb{I} - \boldsymbol{r}_i \, \boldsymbol{r}_i^T$$

a 3x3 identity matrix



Kinetic Energy of a Rigid Body

So: $\frac{1}{2}\sum_i(\boldsymbol{\omega}\times\boldsymbol{r}_i)^T m_i(\boldsymbol{\omega}\times\boldsymbol{r}_i) = \frac{1}{2}\boldsymbol{\omega}^T \underbrace{\sum_i m_i\left(\boldsymbol{r}_i^T\boldsymbol{r}_i\mathbb{I} - \boldsymbol{r}_i\,\boldsymbol{r}_i^T\right)}_{\text{\textit{Moment of Inertia} tensor } \boldsymbol{I}\text{, expressed in world coordinates}}\boldsymbol{\omega}$

Note that $\boldsymbol{r}_i = \boldsymbol{x}_i - \boldsymbol{p} = \boldsymbol{p} + \boldsymbol{R}\overline{\boldsymbol{x}}_i - \boldsymbol{p} = \boldsymbol{R}\overline{\boldsymbol{x}}_i$, therefore:

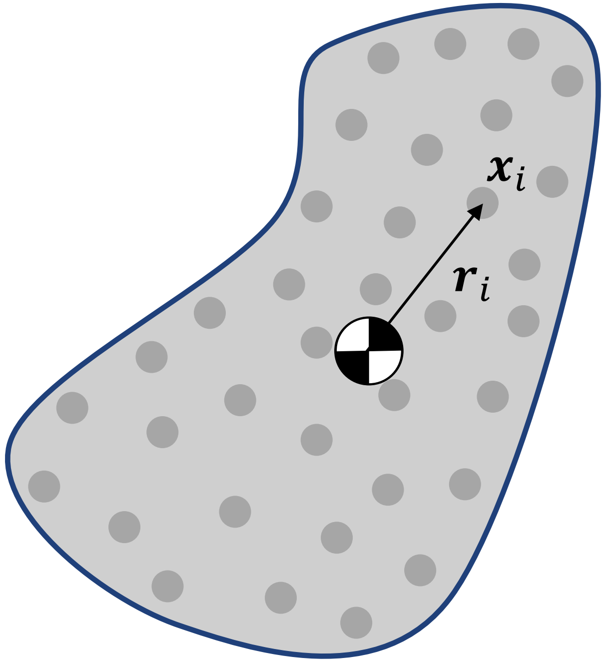
$$\boldsymbol{r}_i^T\boldsymbol{r}_i\mathbb{I} = (\boldsymbol{R}\overline{\boldsymbol{x}}_i)^T(\boldsymbol{R}\overline{\boldsymbol{x}}_i)\mathbb{I} = \overline{\boldsymbol{x}}_i^T\overline{\boldsymbol{x}}_i\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{R}\overline{\boldsymbol{x}}_i^T\overline{\boldsymbol{x}}_i\boldsymbol{R}^T$$

and

$$\boldsymbol{r}_i\,\boldsymbol{r}_i^T = (\boldsymbol{R}\overline{\boldsymbol{x}}_i)(\boldsymbol{R}\overline{\boldsymbol{x}}_i)^T = \boldsymbol{R}\overline{\boldsymbol{x}}_i\overline{\boldsymbol{x}}_i^T\boldsymbol{R}^T$$

So:

$$\boldsymbol{I} = \sum_i m_i\left(\boldsymbol{r}_i^T\boldsymbol{r}_i\mathbb{I} - \boldsymbol{r}_i\,\boldsymbol{r}_i^T\right) = \underbrace{\boldsymbol{R}\overbrace{\sum_i m_i\left(\overline{\boldsymbol{x}}_i^T\overline{\boldsymbol{x}}_i\mathbb{I} - \overline{\boldsymbol{x}}_i\overline{\boldsymbol{x}}_i^T\right)}^{\text{Local coordinates \textit{Moment of Inertia} tensor } \boldsymbol{I}_{\boldsymbol{b}}}\boldsymbol{R}^T}_{\text{World coordinates \textit{Moment of Inertia} tensor } \boldsymbol{I} = \boldsymbol{R}\boldsymbol{I}_{\boldsymbol{b}}\boldsymbol{R}^T}$$

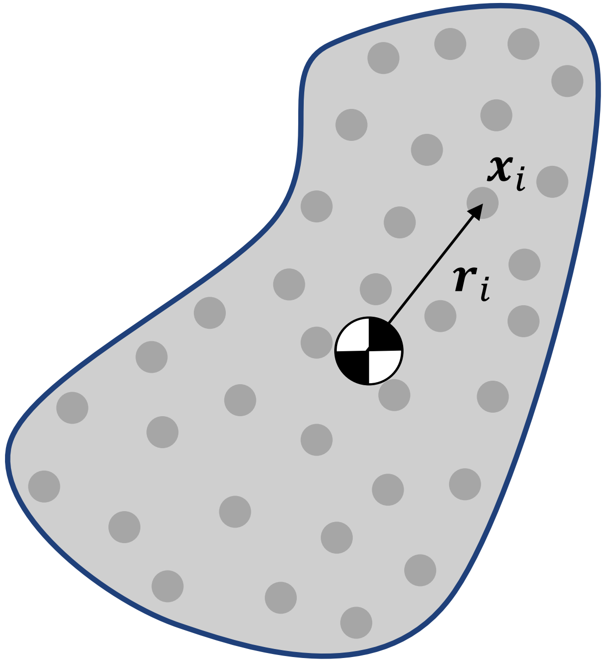


Kinetic Energy of a Rigid Body

- Kinetic energy of a rigid body:

$$
\begin{aligned}
K &= \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i \\
&= \frac{1}{2}\sum_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times (\boldsymbol{x}_i - \boldsymbol{p})\right)^T m_i \left(\boldsymbol{v} + \boldsymbol{\omega} \times \overbrace{(\boldsymbol{x}_i - \boldsymbol{p})}^{\boldsymbol{r}_i}\right) \\
&= \frac{1}{2}\sum_i \boldsymbol{v}^T m_i \boldsymbol{v} + 2\, \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i\, (\boldsymbol{\omega} \times \boldsymbol{r}_i) \\
&= \frac{1}{2}\boldsymbol{v}^T \left(\sum_i m_i{}^T\right) \boldsymbol{v} + \sum_i \boldsymbol{v}^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i) + \underbrace{\frac{1}{2}\sum_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)^T m_i (\boldsymbol{\omega} \times \boldsymbol{r}_i)}
\end{aligned}
$$

$\frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{I} \boldsymbol{\omega}$ - kinetic energy due to the angular motion of the RB



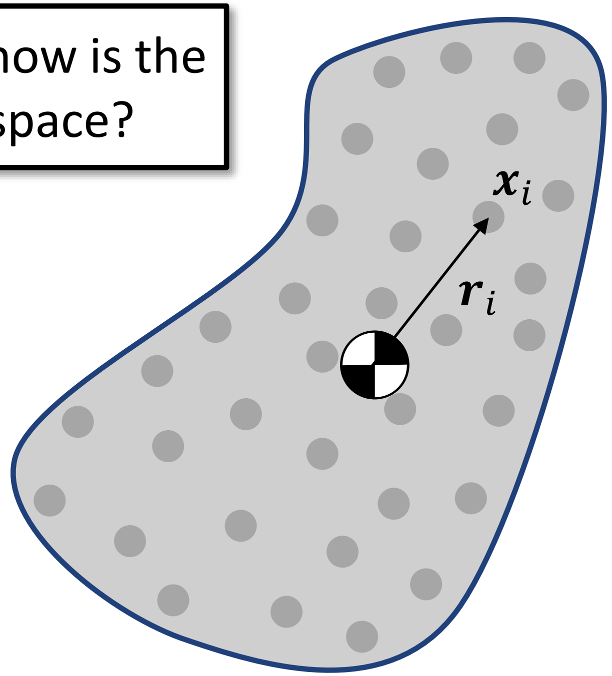
Kinetic Energy of a Rigid Body

- Kinetic energy of a rigid body:

$$K = \frac{1}{2}\sum_i \dot{\boldsymbol{x}}_i^T m_i \dot{\boldsymbol{x}}_i = \frac{1}{2}\boldsymbol{v}^T M\boldsymbol{v} + \frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{I}\boldsymbol{\omega}$$

First order of mass: where is the mass concentrated?

Second order of mass: how is the mass distributed in space?



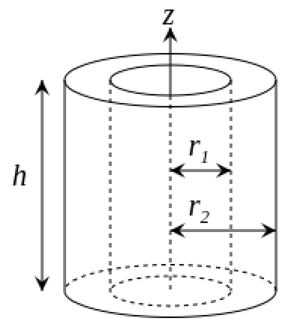
Moment of Inertia

- It is “mass” for rotational motions
  - “The more mass an object has, the more force it takes to move it”
- Is constant in body coordinates
  - Only needs to be computed once!
- In world coordinates, it changes as the RB rotates: $\boldsymbol{I} = \boldsymbol{R}\boldsymbol{I}_b\boldsymbol{R}^T$
  - Can it ever be constant in world coordinates?
- What does $\boldsymbol{I}\boldsymbol{\omega}$ mean?
  - $\boldsymbol{\omega}$ is a world coordinates vector, $\boldsymbol{I}\boldsymbol{\omega}$ is also expressed in world coordinates

Moment of Inertia

- $\boldsymbol{I_b} = \sum_i m_i\left(\overline{\boldsymbol{x}}_i^T \overline{\boldsymbol{x}}_i \mathbb{I} - \overline{\boldsymbol{x}}_i \overline{\boldsymbol{x}}_i^T\right)$, or in the continuous case $\int_\Omega \rho(\overline{\boldsymbol{x}}_{dV}^T \overline{\boldsymbol{x}}_{dV} \mathbb{I} - \overline{\boldsymbol{x}}_{dV} \overline{\boldsymbol{x}}_{dV}^T)\, dV$

- For basic shapes, closed form solutions to the integral exist: https://en.wikipedia.org/wiki/List_of_moments_of_inertia

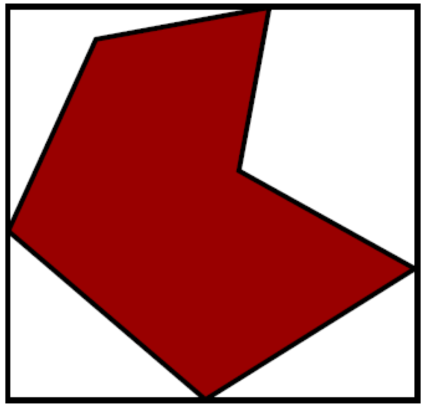


- It is always possible to choose a local coordinate frame such that $\boldsymbol{I_b}$ only has non-zero elements on the diagonal
  - These are called principal moments of inertia $\boldsymbol{I}_x, \boldsymbol{I}_y, \boldsymbol{I}_z$
  - If someone already gives you a MoI, you can do an eigenvalue decomposition on it to retrieve both the principal moments of inertia, and the axes of the local coordinate frame where off-diagonal entries vanish.

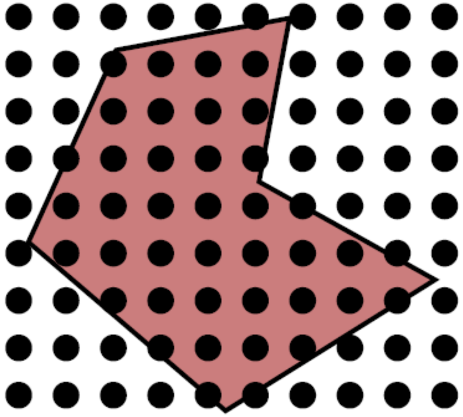
Moment of Inertia

- In practice, the local moment of inertia is often approximated. For instance:

Bounding box



Point sampling



Rigid Body Dynamics

Mass

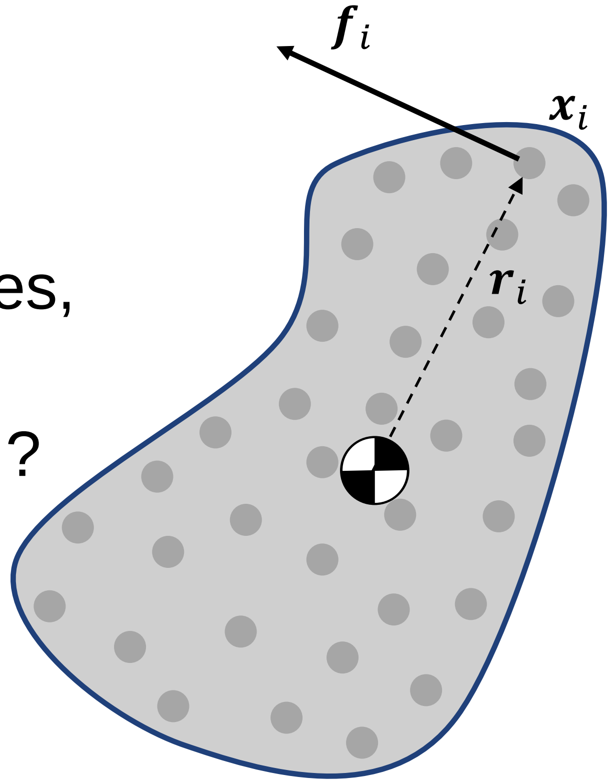
$$"F = ma"$$

Forces

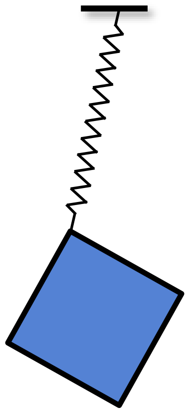
Accelerations

Rigid Bodies: forces and torques

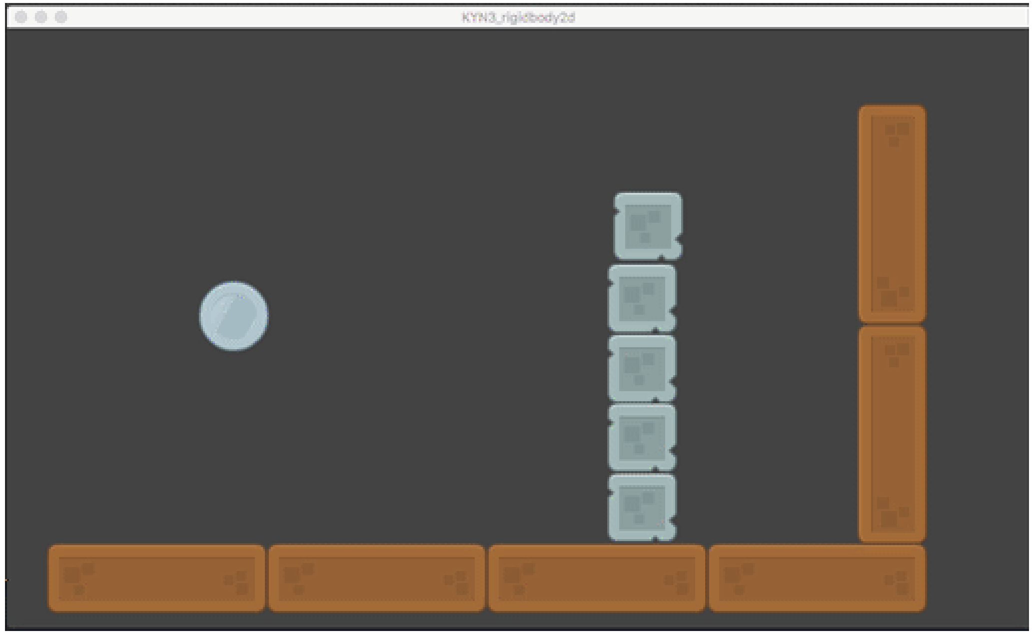
- $\boldsymbol{f}_i(t)$ denotes the total force from external forces acting on the i^{th} particle at time t.
  - Total/net force acting on the rigid body: $\boldsymbol{F}(\boldsymbol{t}) = \sum_i \boldsymbol{f}_i(t)$
  - Total/net torque on the rigid body: $\boldsymbol{\tau}(\boldsymbol{t}) = \sum_i \boldsymbol{r}_i \times \boldsymbol{f}_i(t)$
- Net torque depends on the point of application of individual forces, but net force does not
- What are the net force and torque due to gravity (e.g. $\boldsymbol{f}_i = m_i \boldsymbol{g}$)?
  - $\boldsymbol{F} = \sum_i m_i \boldsymbol{g} = \boldsymbol{g} \sum_i m_i = M\boldsymbol{g}$
  - $\boldsymbol{\tau} = \sum_i \boldsymbol{r}_i \times m_i \boldsymbol{g} = -\sum_i m_i [\boldsymbol{g}] \boldsymbol{r}_i = -[\boldsymbol{g}] \sum_i m_i \boldsymbol{r}_i = \boldsymbol{0}$



Rigid Bodies: forces and torques



- Where do forces come from?
  - Gravity
  - Springs
  - User interaction – e.g. pick an object with the mouse and drag it around
  - Joint constraints
  - Collisions and Frictional Contact
  - Motors or muscles
  - Etc.



Rigid Body Dynamics

Mass

$$"F = ma"$$

Forces

Accelerations

Linear and angular momenta

- $\mathbf{p}$: total *linear momentum* of the RB

$$\mathbf{p} = M\boldsymbol{v}$$

  - Linear momentum of an RB is the same as if it was just a particle with mass M and velocity $\boldsymbol{v}(t)$!

- $\mathbf{L}$: total *angular momentum* of the RB is defined analogously

$$\mathbf{L} = \boldsymbol{I\omega}$$

  - Angular moment does not depend on linear COM motion, only on rotation about the COM.

Newton’s second law of motion and conservation of momentum

$$\boldsymbol{F} = M\boldsymbol{a} = M\frac{d\boldsymbol{v}}{dt} = \frac{d(M\boldsymbol{v})}{dt} = \frac{d\mathbf{p}}{dt} = \dot{\mathbf{p}}$$

- This expression tells us how the linear velocity of the RB changes over time in response to net force applied to the rigid body.
  - Change in linear momentum is equivalent to the net force acting on the rigid body.

- Analogous expression holds for conservation of angular momentum
  - Change in angular momentum is equivalent to the net torque acting on the rigid body.

$$\dot{\mathbf{L}} = \frac{\mathrm{d}\mathbf{L}}{dt} = \boldsymbol{\tau}$$

Conservation of angular momentum

$$\dot{\mathbf{L}} = \boldsymbol{\tau}, \mathbf{L} = \boldsymbol{I\omega}$$
$$\dot{\mathbf{L}} = \dot{\boldsymbol{I}}\boldsymbol{\omega} + \boldsymbol{I}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}$$

Does the MoI change over time?

Angular acceleration

Conservation of angular momentum

$$\dot{\mathbf{L}} = \boldsymbol{\tau}, \mathbf{L} = \boldsymbol{I\omega}$$
$$\dot{\mathbf{L}} = \dot{\boldsymbol{I}}\boldsymbol{\omega} + \boldsymbol{I}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}$$
$$\boldsymbol{I} = \boldsymbol{R}\boldsymbol{I}_b\boldsymbol{R}^T \Rightarrow \dot{\boldsymbol{I}} = \dot{\boldsymbol{R}}\boldsymbol{I}_b\boldsymbol{R}^T + \boldsymbol{R}\dot{\boldsymbol{I}}_b\boldsymbol{R}^T + \boldsymbol{R}\boldsymbol{I}_b\dot{\boldsymbol{R}}^T$$

Recall: $\dot{\boldsymbol{R}} = [\boldsymbol{\omega}]\boldsymbol{R}$, so

$$\dot{\boldsymbol{I}} = \dot{\boldsymbol{R}}\boldsymbol{I}_b\boldsymbol{R}^T + \boldsymbol{R}\dot{\boldsymbol{I}}_b\boldsymbol{R}^T + \boldsymbol{R}\boldsymbol{I}_b\dot{\boldsymbol{R}}^T = [\boldsymbol{\omega}]\boldsymbol{R}\boldsymbol{I}_b\boldsymbol{R}^T + \mathbf{0} - \boldsymbol{R}\boldsymbol{I}_b\boldsymbol{R}^T[\boldsymbol{\omega}]$$
$$\Rightarrow \dot{\boldsymbol{I}}\boldsymbol{\omega} = \boldsymbol{\omega} \times \boldsymbol{I\omega} + \boldsymbol{I\omega} \times \boldsymbol{\omega} = \boldsymbol{\omega} \times \boldsymbol{I\omega}$$

$$\Rightarrow \boldsymbol{I}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau} - \boldsymbol{\omega} \times \boldsymbol{I\omega}$$

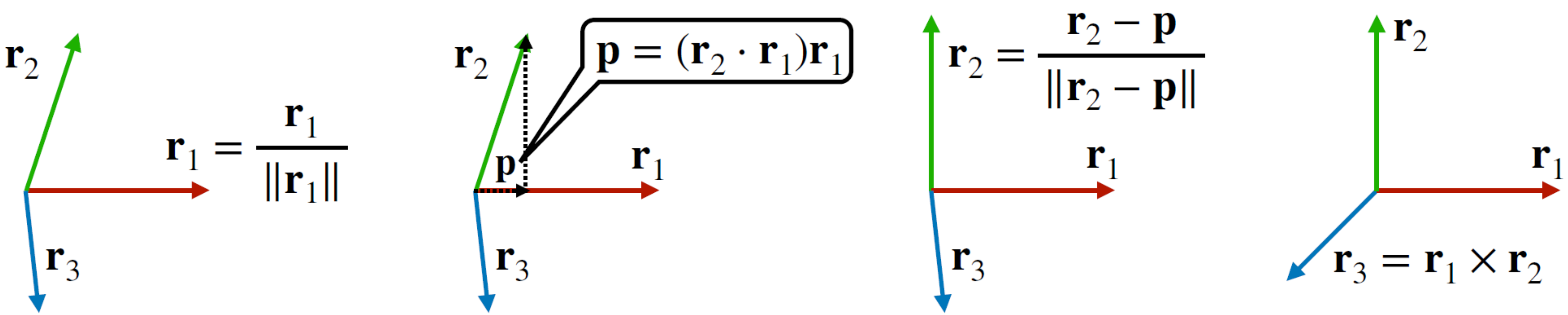
The "$\boldsymbol{F} = \boldsymbol{ma}$" of rigid body dynamics: aka the Newton–Euler equations of motion

$$\underbrace{\begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{\tau} - \boldsymbol{\omega} \times \boldsymbol{I}\boldsymbol{\omega} \end{bmatrix}}_{\text{Force}} = \overbrace{\begin{bmatrix} M\mathbb{I} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix}}^{\text{Mass}} \underbrace{\begin{bmatrix} \dot{\boldsymbol{v}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix}}_{\text{Acceleration}}$$

Note: linear and angular components of a rigid body's motion are decoupled!

Time stepping scheme (Symplectic Euler)

- At each time step:
  - Compute net force $\boldsymbol{F}$ and net torque $\boldsymbol{\tau}$
  - Update linear and angular velocities: $\boldsymbol{v}_{i+1} = \boldsymbol{v}_i + h\frac{\boldsymbol{F}}{M}$, $\boldsymbol{\omega}_{i+1} = \boldsymbol{\omega}_i + h\boldsymbol{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega}_i \times \boldsymbol{I}\boldsymbol{\omega}_i)$
    - Note: $\boldsymbol{I} = \boldsymbol{R}\boldsymbol{I}_b\boldsymbol{R}^T$, so $\boldsymbol{I}^{-1} = \boldsymbol{R}\boldsymbol{I}_b^{-1}\boldsymbol{R}^T$
  - Update COM position: $\boldsymbol{p}_{i+1} = \boldsymbol{p}_i + h\boldsymbol{v}_{i+1}$
  - Update orientation:
    - Option 1: $\boldsymbol{R}_{i+1} = \boldsymbol{R}_i + h\dot{\boldsymbol{R}}_{i+1} = \boldsymbol{R}_i + h[\boldsymbol{\omega}_{i+1}]\boldsymbol{R}_i = (\mathbb{I} + h[\boldsymbol{\omega}_{i+1}])\boldsymbol{R}_i$
      - Main issue, over time $\boldsymbol{R}$ will no longer be an orthonormal matrix. Fix via Gram-Schmidt orthonormalization process.



Time stepping scheme (Symplectic Euler)

- At each time step:
  - Compute net force $\boldsymbol{F}$ and net torque $\boldsymbol{\tau}$
  - Update linear and angular velocities: $\boldsymbol{v}_{\mathrm{i+1}} = \boldsymbol{v}_{\mathrm{i}} + h\frac{\boldsymbol{F}}{M}$, $\boldsymbol{\omega}_{\mathrm{i+1}} = \boldsymbol{\omega}_{\mathrm{i}} + h\boldsymbol{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega_i} \times \boldsymbol{I}\boldsymbol{\omega_i})$
    - Note: $\boldsymbol{I} = \boldsymbol{R}\boldsymbol{I_b}\boldsymbol{R}^T$, so $\boldsymbol{I}^{-1} = \boldsymbol{R}\boldsymbol{I}_{\boldsymbol{b}}^{-1}\boldsymbol{R}^T$
  - Update COM position: $\boldsymbol{p}_{\mathrm{i+1}} = \boldsymbol{p}_{\mathrm{i}} + h\boldsymbol{v}_{\mathrm{i+1}}$
  - Update orientation:
    - Option 1: $\boldsymbol{R}_{\mathrm{i+1}} = \boldsymbol{R}_{\mathrm{i}} + h\dot{\boldsymbol{R}}_{\mathrm{i+1}} = \boldsymbol{R}_{\mathrm{i}} + h[\boldsymbol{\omega}_{\mathrm{i+1}}]\boldsymbol{R}_{\mathrm{i}} = (\mathbb{I} + h[\boldsymbol{\omega}_{\mathrm{i+1}}])\boldsymbol{R}_{\mathrm{i}}$
      - Main issue, over time $\boldsymbol{R}$ will no longer be an orthonormal matrix. Fix via Gram-Schmidt orthonormalization process.
    - Option 2: first compute matrix corresponding to a rotation with angular speed $\boldsymbol{\omega}_{\mathrm{i+1}}$ for a time period h, then apply this after the rotation at previous time step: $\boldsymbol{R}_{\mathrm{i+1}} = \mathbf{Rot}\left(h|\boldsymbol{\omega}_{\mathrm{i+1}}|, \frac{\boldsymbol{\omega}_{\mathrm{i+1}}}{|\boldsymbol{\omega}_{\mathrm{i+1}}|}\right)\boldsymbol{R}_{\mathrm{i}}$
    - Option 3: use a different parameterization of rotations

Quaternions

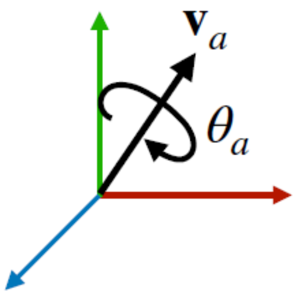
- Quaternions: a generalization of complex numbers to higher dimensions

$$\boldsymbol{q} = w + x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}, \text{ where } \boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -\mathbf{1}$$

- Quaternions are often written as $\boldsymbol{q} = [w, \mathbf{v}]$, where vector $\mathbf{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$
- Quaternion multiplication: $\boldsymbol{q_1} * \boldsymbol{q_2} = [w_1 w_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, w_1 \mathbf{v}_2 + w_2 \mathbf{v}_1 + \mathbf{v}_1 \times \mathbf{v}_2]$
- Unit quaternions (i.e. $w^2 + x^2 + y^2 + z^2 = 1$) encode a much more compact parameterization of 3D rotations

Unit quaternions

- Rotation quaternions: $\mathbf{q}(\boldsymbol{\theta}_{\mathrm{a}}, \mathbf{v}_{\mathrm{a}})$



axis angle $(\theta_a, \mathbf{v}_a)$

$$\mathbf{q} = \begin{bmatrix} w \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \cos\frac{\theta_a}{2} \\ \sin\frac{\theta_a}{2}\mathbf{v}_a \end{bmatrix}$$

- Rotating vectors:

$$\mathrm{Rot}(\boldsymbol{v}, \mathbf{q}) \equiv \mathbf{q}\boldsymbol{v} = \mathbf{q} * (\mathbf{0},\ \boldsymbol{v}) * \mathbf{q}^{-1}$$

- Let $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ be two rotation quaternions. The quaternion $\boldsymbol{q}_1 * \boldsymbol{q}_2$ represents the rotation obtained by first applying rotation $\boldsymbol{q}_2$, then rotating by $\boldsymbol{q}_1$ - analogous to the way in which rotations are composed using rotation matrices.

Time stepping scheme using Quaternions

- At each time step:
  - Compute net force $\boldsymbol{F}$ and net torque $\boldsymbol{\tau}$
  - Update linear and angular velocities: $\boldsymbol{v}_{\mathrm{i+1}} = \boldsymbol{v}_{\mathrm{i}} + h\frac{\boldsymbol{F}}{M}$, $\boldsymbol{\omega}_{\mathrm{i+1}} = \boldsymbol{\omega}_{\mathrm{i}} + h\boldsymbol{I}^{-\mathbf{1}}(\boldsymbol{\tau} - \boldsymbol{\omega}_{\boldsymbol{i}} \times \boldsymbol{I}\boldsymbol{\omega}_{\boldsymbol{i}})$
  - Update COM position: $\boldsymbol{p}_{\mathrm{i+1}} = \boldsymbol{p}_{\mathrm{i}} + h\boldsymbol{v}_{\mathrm{i+1}}$
  - Update orientation:
    - Option 1: $\boldsymbol{q}_{\mathrm{i+1}} = \mathbf{q}\left(h|\boldsymbol{\omega}_{\mathrm{i+1}}|, \frac{\boldsymbol{\omega}_{\mathrm{i+1}}}{|\boldsymbol{\omega}_{\mathrm{i+1}}|}\right)\boldsymbol{q}_{\mathrm{i}}$
    - Option 2: $\dot{\boldsymbol{q}}_{i+1} = \frac{1}{2}(0, \boldsymbol{\omega}_{\mathrm{i+1}})\boldsymbol{q}_{\mathrm{i}}$, $\boldsymbol{q}_{\mathrm{i+1}} = \boldsymbol{q}_{\mathrm{i}} + h\dot{\boldsymbol{q}}_{i+1}$
      - Can you show that in the limit, as $h \to 0$, these two options are equivalent?
      - Normalization of quaternions is still necessary, but easier/faster than orthonormalizing rotation matrices
      - Easy to extract rotation angle/axis, therefore easy to compute rotation matrix from a quaternion - useful, for example, when computing world-coordinates moment of inertia. Coordinate frame transformations, on the other hand, can be implemented using quaternion operations directly.
        - Use the right representation for your problem!

Numerical simulation models

Particles

State:
- Position
- Velocity

Physical Properties:
- Mass

Rigid Bodies

State:
- Position
- Orientation
- Linear Velocity
- Angular Velocity

Physical Properties:
- Mass
- Moment of Inertia

$$"F = ma"$$

Additional material

- Quaternions:

  https://en.wikipedia.org/wiki/Quaternion
- Skew symmetric matrix:

  https://en.wikipedia.org/wiki/Skew-symmetric_matrix
- Rigid body lecture notes from David Baraff:

  https://www.cs.cmu.edu/~baraff/sigcourse/notesd1.pdf

  https://www.cs.cmu.edu/~baraff/sigcourse/notesd1.pdf
- Brian Mirtich's thesis

  https://people.eecs.berkeley.edu/~jfc/mirtich/thesis/mirtichThesis.pdf
- Impulse-based collision processing

  "Nonconvex Rigid Bodies with Stacking", Guendelman et al., 2003

  hand-written course notes