Robotics 311 : How to build robots and make them move

Prof. Elliott Rouse

GSI Yves Nazon MS

Fall 2022

Photo credit: Brenda Ahearn / U-M CoE

ROB 311 – Lecture 13

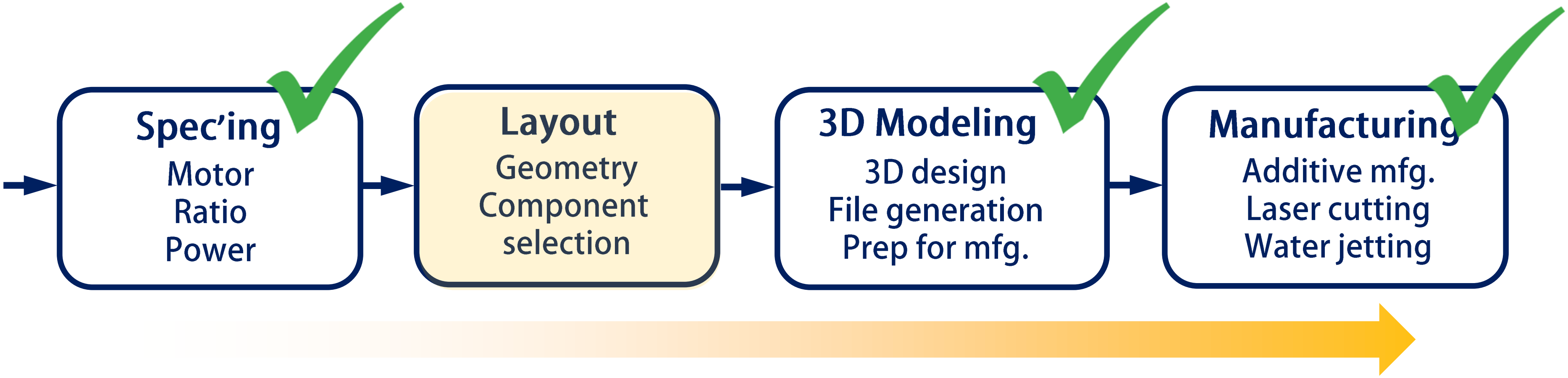
- Today:
  - Review planar model
  - Review torque conversion
  - MATLAB exercise
  - Kinematic conversions

- Announcements
  - IMU issue – we will replace all IMUs in lab tomorrow
  - HW 3 posted, due 10/20 at class start
  - Yves graded HW 1
  - He will grade HW 2 shortly and post all solutions
  - We will also post the quizzes and their solutions
  - Midterm exam – 11/8

ROB 311 – Lab 7

- We will be learning Git and how to code in Python!
- This knowledge is critical for developing the ball-bot control
- Git
  - Revisit forking repositories
  - Authentication
  - Saving data
- Python
  - Looping and indexing
  - Importing packages
  - Creating functions
  - Saving data

Manufacturing Types



- We've learned how to spec and make robots, now lets talk about design layouts
- This is often moving motion from one place to another (kinematics)
- In robots, motion moving from the actuator to the end effector
- It begins with understanding the geometry of your robot and transmissions
- Very application specific!
- Coming up:
  - Introduce transmissions and linkages
  - In-depth example of ball-bot geometry and kinematics
  - Move to mechatronics, ball-bot dynamics, and control

Manufacturing Types

Spec'ing → **Layout** → **3D Modeling** → **Manufacturing**

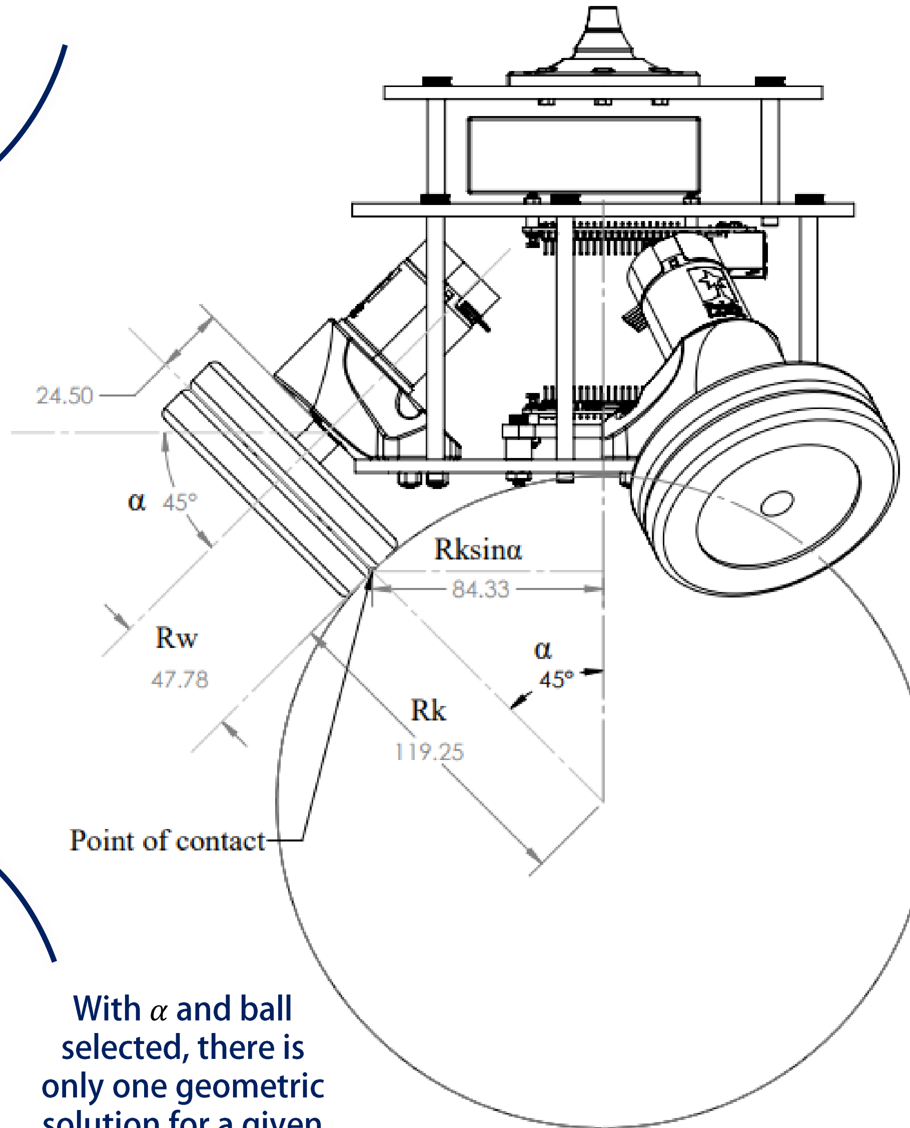
- **Spec'ing**: Motor, Ratio, Power
- **Layout**: Geometry, Component selection
- **3D Modeling**: 3D design, File generation, Prep for mfg.
- **Manufacturing**: Additive mfg., Laser cutting, Water jetting

- To determine the layout, we need to know the geometry of our robot
- The geometry of the ball-bot is more complex / important than usual
- To describe the geometry, we need an understanding of how motion is applied to the ball-bot
- This requires some in-depth descriptions of the ball-bot
    - Torque / motion applied from the wheels to the basketball
- This is also critical for control

Ball-Bot Layout

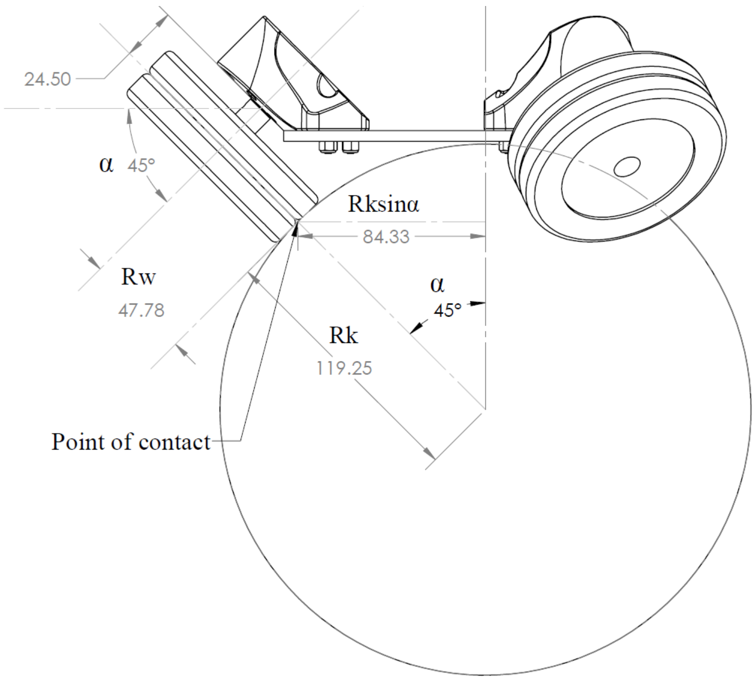
Balances base width with radial force / friction increases

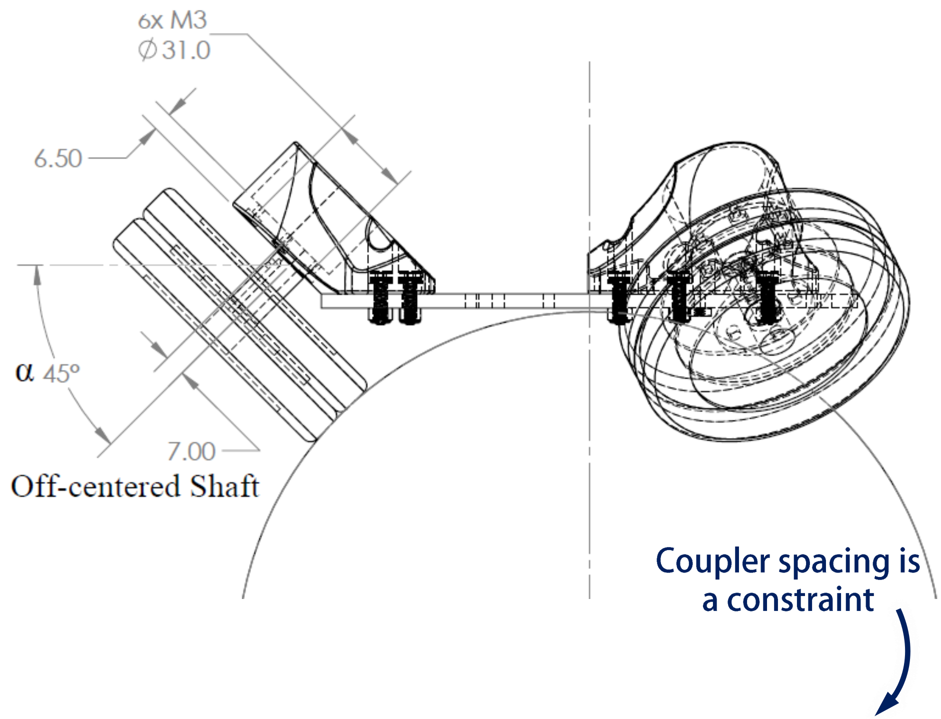
- Major design decisions are α angle, ball radius, and wheel radius
- We chose $\alpha = 45^{\circ}$
  - As α is increased, radial force increases → more friction
  - Larger α designs have a greater lean angle
- We chose a basketball for its cost, size, and texture
- Basketball radius (R_k) is 119.25 mm
- Wheel radius (R_W) is 47.8 mm
- Coupler to wheel center distance is 24.5 mm
- Wheel contacts spaced on circle with radius 84.3 mm from Z axis $(R_k \sin(\alpha))$



With α and ball selected, there is only one geometric solution for a given wheel radius

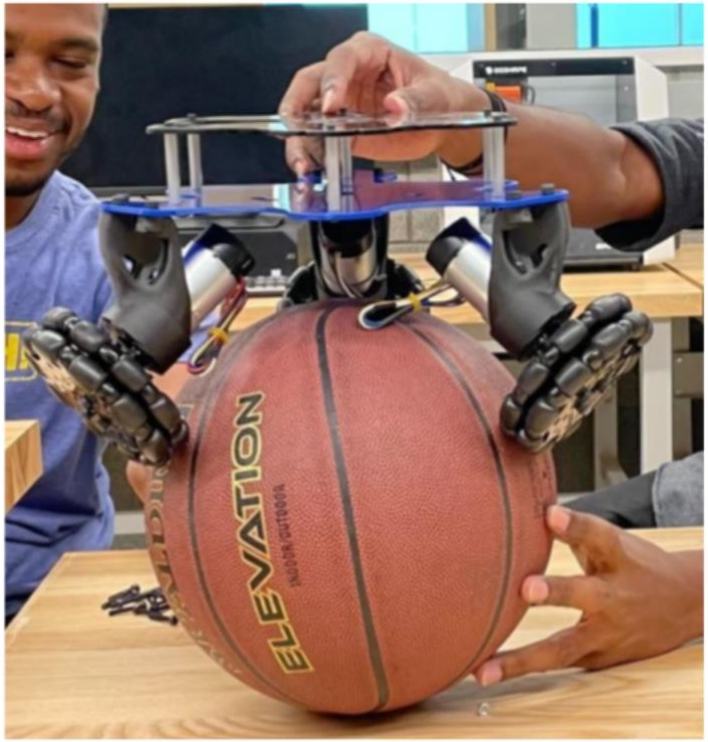
Ball-Bot Layout

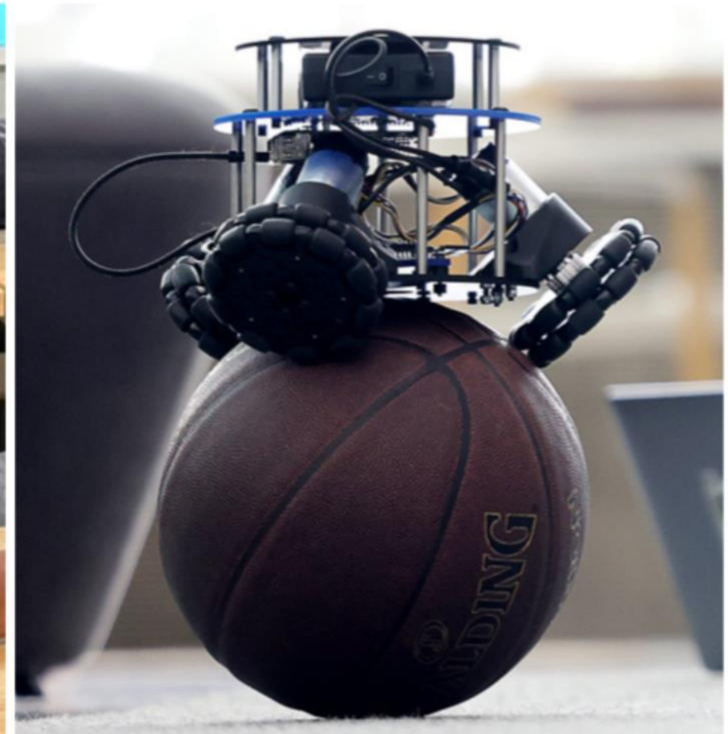




- Now we know many of the dimensions, driven by α, ball radius, wheel radius, and coupler spacing
- To design the motor mount sketches, we need to know more about the motor dimensions
- Shaft is off center by 7 mm
- These dimensions define how your sketches were provided
- Now lets discuss motion and torque transfer

Ball-Bot Layout



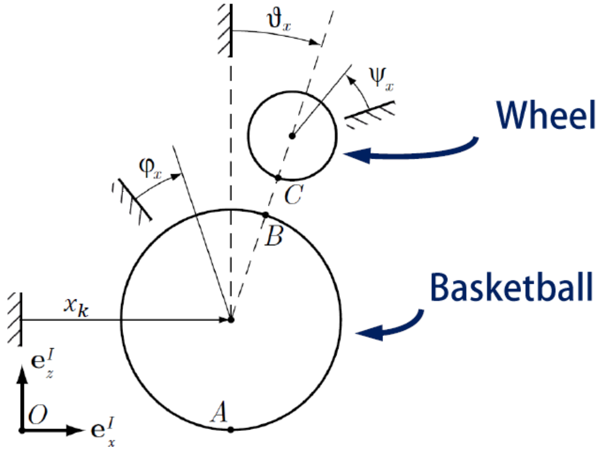


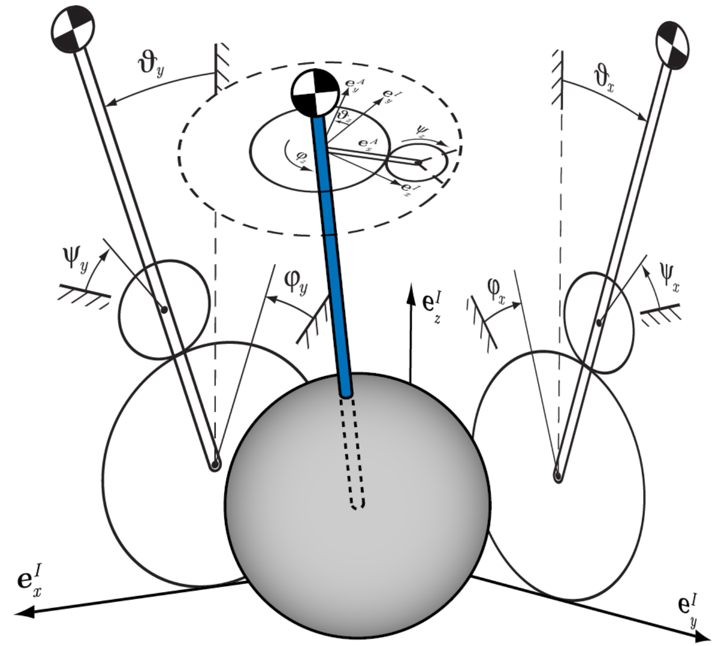
Design v1
Mounts are under
the base plate

Design v2
Mounts are above
the base plate

- What effect did this have? Lowering the center of mass and stiffer mounts

Full Planar Model

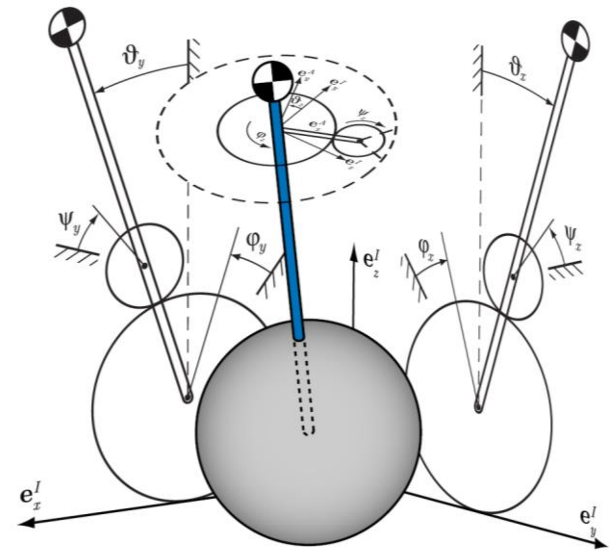




- When spec’ing, we developed a planar model of the ball-bot
- We will build on our planar model to describe the 3D motion / torque of the robot
- We project the 3D ball-bot into three planes
    - X-Z plane and Y-Z plane, and X-Y plane
- Each plane contains one virtual wheel
- This is simplification is helpful analytically

Assumptions and Rolling Physics

- No slip: The contact points at the wheel-ball and ball-ground interfaces do not slip
  - Velocity at ball-ground interface = 0
- No deformation: We will not consider deformation of the ball
- Consider a ball rolling with some velocity
- Tangential velocity from rolling is $r\dot{\theta}$

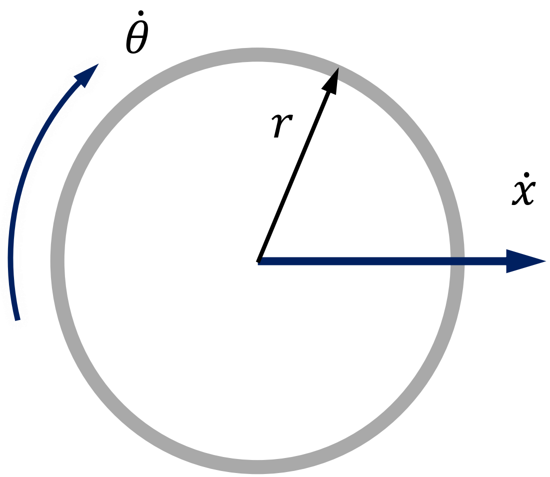


What is the x-component of velocity at the top of the ball?

$$\dot{x}_{top} = \dot{x} + r\dot{\theta}$$

How does the no-slip condition relate rolling and linear velocities?

$$r\dot{\theta} = \dot{x}$$

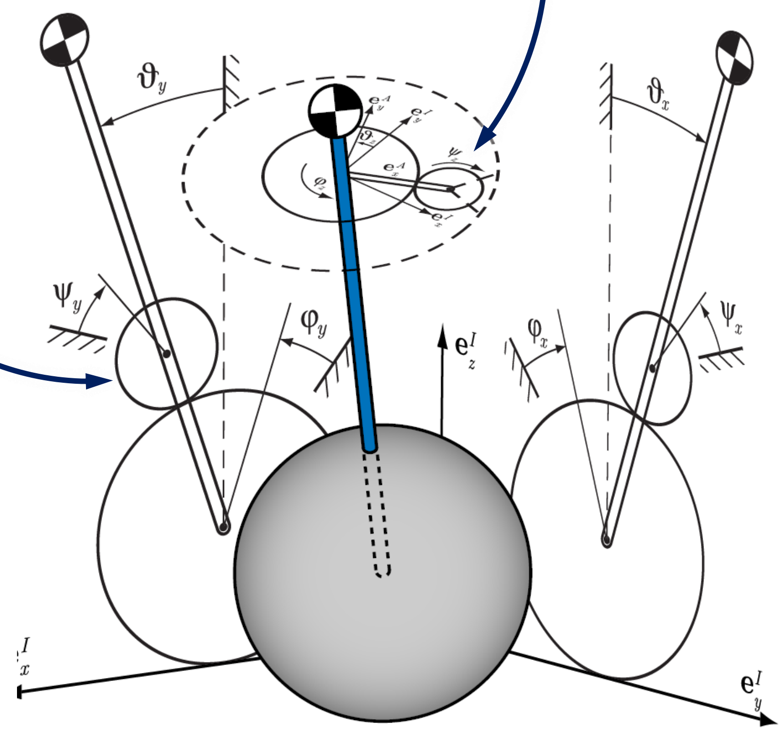


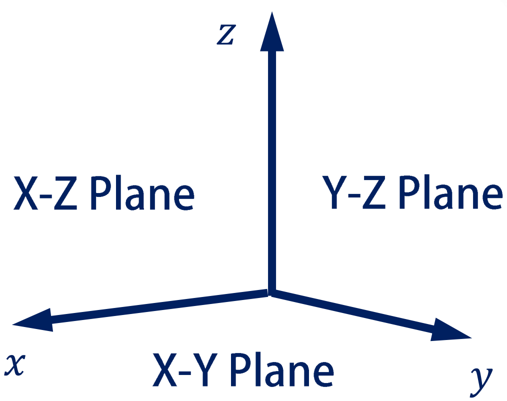
Full Planar Model

The X-Y plane contains a single wheel that rotates the ball-bot

ϑ_{axis}	Body rotation
ψ_{axis}	Wheel rotation
φ_{axis}	Ball rotation

The X-Z and Y-Z planes have a single wheel beneath the center of mass





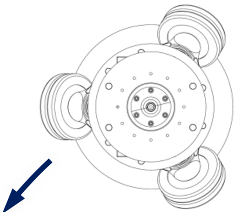
Since we want to balance and drive, the rotation of the ball (i.e. X-Y plane) is mostly irrelevant

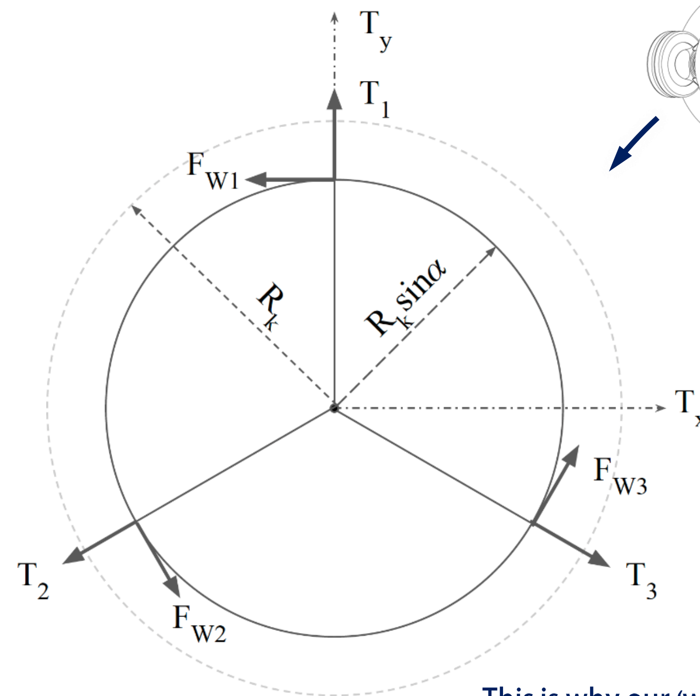
- Each plane has two DOFs: Rotations of the chassis and ball
- X-Z and Y-Z planes are required for balance and motion—we will focus on these
- The torques and motion of the virtual wheels are different from the real wheels

Torque Transfer

F_{W1}	Wheel-ball force 1	R_k	Ball radius
F_{W2}	Wheel-ball force 2	T_1	Motor 1 torque
F_{W3}	Wheel-ball force 3	T_2	Motor 2 torque
		T_3	Motor 3 torque

- How does the motor torque get transferred to each plane?
- Lets look at how the geometry affects resolving torques between coordinate systems
  - Planar torques are around the x and y axes
  - Wheel torques are T_1, T_2, and T_3
- Real wheels on contact circle defined by $R_k \sin(\alpha)$
- We aligned Motor 1 torque to be along y axis
- We defined α at 45° and each torque vector spaced at 120°





This is why our 'worst case' approximation during modeling for motor selection was helpful. We planned for this! (Lecture 3)

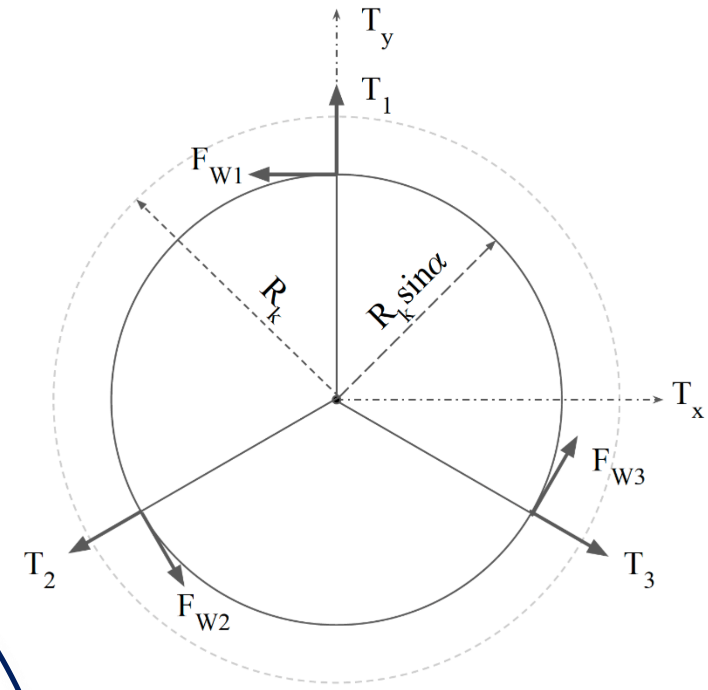
Will this have consequences on the motor? Yes, we're defining one of the axis torques to be fully borne by one motor, while splitting the torques from the other plane to two motors

Virtual and Real Wheel Contact

- Torque on the wheel - cross product

$$T_{Wi} = r_{Wi} \times F_{Wi} \text{ for } i = 1,2,3$$

$$T_{Wj} = r_{Wj} \times F_{Wj} \text{ for } j = x, y, z$$



- Virtual wheel contact points in (X-Z and Y-Z planes)

$$r_{Wx} = R_k \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad r_{Wy} = R_k \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad r_{kWz} = R_k \begin{bmatrix} 0 \\ \sin(\alpha) \\ 0 \end{bmatrix}$$

We begin with contact points then discuss forces

- Real wheel contact points

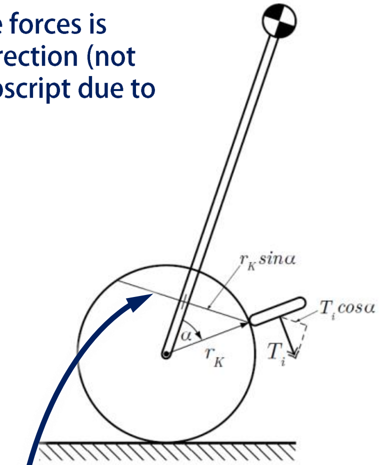
$$r_{W1} = R_k \begin{bmatrix} 0 \\ \sin(\alpha) \\ \cos(\alpha) \end{bmatrix} \quad r_{W2} = R_k \begin{bmatrix} \frac{-\sqrt{3}}{2}\sin(\alpha) \\ -\frac{1}{2}\sin(\alpha) \\ \cos(\alpha) \end{bmatrix} \quad r_{W3} = R_k \begin{bmatrix} \frac{\sqrt{3}}{2}\sin(\alpha) \\ -\frac{1}{2}\sin(\alpha) \\ \cos(\alpha) \end{bmatrix}$$

Virtual and Real Forces

- Torque on the wheel - cross product

$$T_{Wi} = r_{Wi} \times F_{Wi} \;\; \text{for} \;\; i = 1,2,3$$
$$T_{Wj} = r_{Wj} \times F_{Wj} \;\; \text{for} \;\; j = x, y, z$$

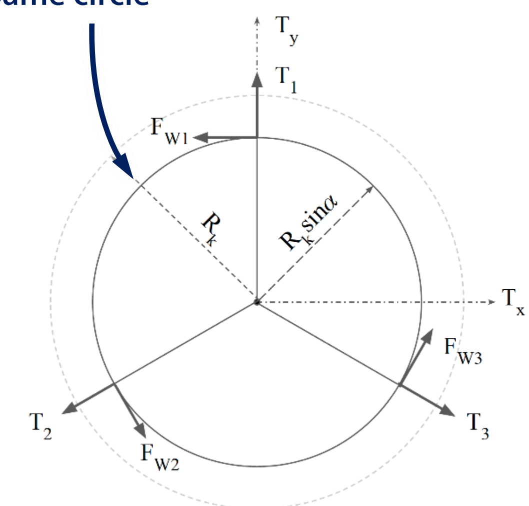
The direction of the forces is along the vector direction (not the component subscript due to the cross product)



- Virtual wheel forces

$$F_{Wx} = \frac{T_x}{r_w}\begin{bmatrix}0\\1\\0\end{bmatrix} \quad F_{Wy} = \frac{T_y}{r_w}\begin{bmatrix}1\\0\\0\end{bmatrix} \quad F_{Wz} = \frac{T_z}{r_w}\begin{bmatrix}-1\\0\\0\end{bmatrix}$$

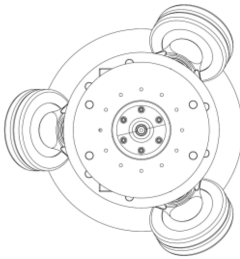
Same circle



- Real wheel forces

$$F_{W1} = \frac{T_1}{r_w}\begin{bmatrix}-1\\0\\0\end{bmatrix} \quad F_{W2} = \frac{T_2}{r_w}\begin{bmatrix}1/2\\-\sqrt{3}/2\\0\end{bmatrix} \quad F_{W3} = \frac{T_3}{r_w}\begin{bmatrix}1/2\\\sqrt{3}/2\\0\end{bmatrix}$$

Relating Virtual and Real Torques



- Torque in both coordinate systems is conserved
    - Not forces conserved—this says the torques around the ball will be equivalent

$$T_1 + T_2 + T_3 = T_x + T_y + T_z$$

- Written in terms of force and perpendicular distance

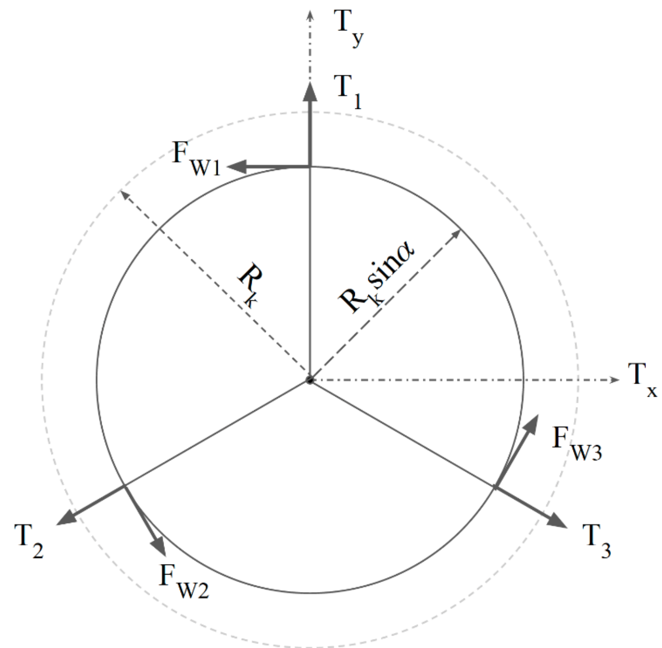
$$r_{W1} \times F_{W1} + r_{W2} \times F_{W2} + r_{W3} \times F_{W3} \dots$$
$$= r_{Wx} \times F_{Wx} + r_{Wy} \times F_{Wy} + r_{Wz} \times F_{Wz}$$

- Solving for torques

$$T_1 = \frac{1}{3}\left(T_z - \frac{2T_y}{\cos(\alpha)}\right)$$

$$T_2 = \frac{1}{3}\left(T_z + \frac{1}{\cos(\alpha)}\left(-\sqrt{3}T_x + T_y\right)\right)$$

$$T_3 = \frac{1}{3}\left(T_z + \frac{1}{\cos(\alpha)}\left(\sqrt{3}T_x + T_y\right)\right)$$



Determined by solving conservation of torque using eqns. from previous slides

Relating Virtual and Real Torques

α $\pi/4$ or 45°

β $\pi/2$ or 90°

- Conversion equations provided in ETH Zurich dissertation
- Solved for real motor torques

These are the same equations as the previous slide, but written generally for any α and β

$$T_1 = \frac{1}{3} \cdot \left(T_z + \frac{2}{\cos\alpha} \cdot (T_x \cdot \cos\beta - T_y \cdot \sin\beta) \right)$$

$$T_2 = \frac{1}{3} \cdot \left(T_z + \frac{1}{\cos\alpha} \cdot \left(\sin\beta \cdot (-\sqrt{3}T_x + T_y) - \cos\beta \cdot (T_x + \sqrt{3}T_y) \right) \right)$$

$$T_3 = \frac{1}{3} \cdot \left(T_z + \frac{1}{\cos\alpha} \cdot \left(\sin\beta \cdot (\sqrt{3}T_x + T_y) + \cos\beta \cdot (-T_x + \sqrt{3}T_y) \right) \right)$$

- Solved for virtual motor torques

$$T_x = \cos\alpha \cdot \left(T_1 \cdot \cos\beta - T_2 \cdot \sin(\beta + \frac{\pi}{6}) + T_3 \cdot \sin(\beta - \frac{\pi}{6}) \right)$$

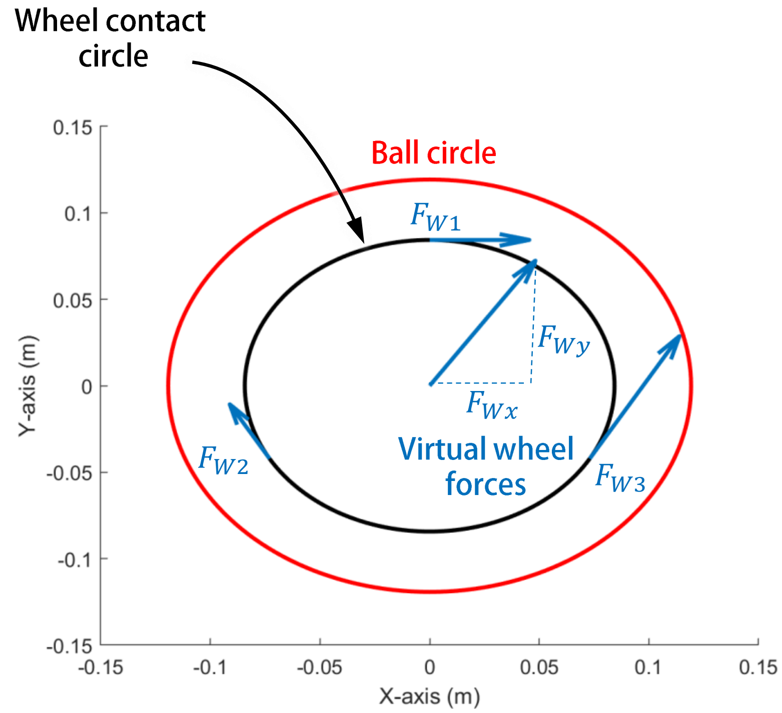
$$T_y = \cos\alpha \cdot \left(-T_1 \cdot \sin\beta - T_2 \cdot \cos(\beta + \frac{\pi}{6}) + T_3 \cdot \cos(\beta - \frac{\pi}{6}) \right)$$

$$T_z = T_1 + T_2 + T_3$$

Fankhauser and Gwerder et al, 2010

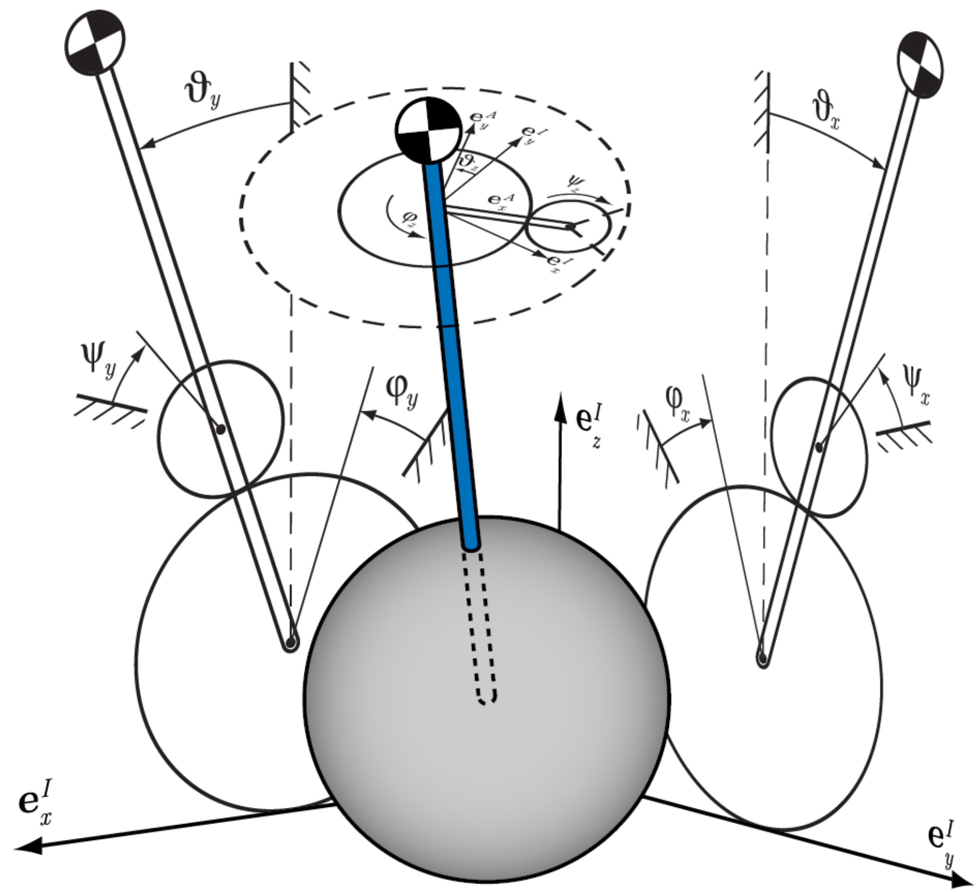
Relating Virtual and Real Torques

- We are going to create a function to convert / plot torques
- Steps
  - Define T_x and T_y values
  - Convert to motor torques T_1, T_2, T_3
  - Determine the X-Y locations of the contact points
  - I have created a version that shows quiver plots (right)
  - MATLAB skeleton uploaded to Canvas
- Find T_1, T_2, T_3 for $T_x = 3$ and $T_y = 2$
- We plan to reuse this function…



Full Planar Model

- We now can describe torque from the wheels to full planar model
  - Go from x and y components to the 3 motor torques
- Can we develop a similar analysis for velocity transformation?
- The real and virtual wheels are the same size
- Do the real wheels and virtual wheels have the same velocity?
  - No, wheel orientation affects velocity
  - Wheels only spin with the component of linear velocity perpendicular to wheel axis



Full Planar Model

- Planar model has two DOFs - φ_{axis} and ϑ_{axis}
- Lets look at the velocities of points A, B, and C

$v_{A-Ccomponent}$	Velocity component of point A, B, or C
φ_{axis}	Ball angle
ϑ_{axis}	Body angle
ψ_{axis}	Wheel angle
y_k	Linear position of ball center
	Time derivative denoted by dot operator

$$v_{Ay} = 0$$

$$v_{Az} = 0$$

$$v_{By} = \dot{y}_k + \dot{\varphi}_x R_k \cos(\vartheta_x)$$

$$v_{Bz} = -\dot{\varphi}_x R_k \sin(\vartheta_x)$$

$$v_{Cy} = \dot{y}_k + \left((R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w\right)\cos(\vartheta_x)$$

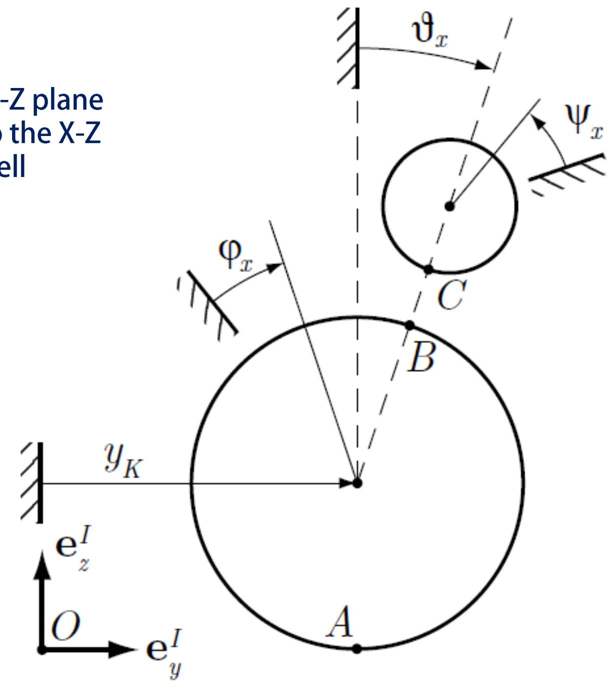
$$v_{Cz} = -\left((R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w\right)\sin(\vartheta_x)$$

$$v_{By} = v_{Cy}$$

$$v_{Bz} = v_{Cz}$$

Y-Z Plane

Derived for the Y-Z plane but applicable to the X-Z plane as well



Full Planar Model

- We can set the velocities equal at points A and B

$v_{A-Ccomponent}$	Velocity component of point A, B, or C
φ_{axis}	Ball angle
ϑ_{axis}	Body angle
ψ_{axis}	Wheel angle
y_k	Linear position of ball center
	Time derivative denoted by dot operator

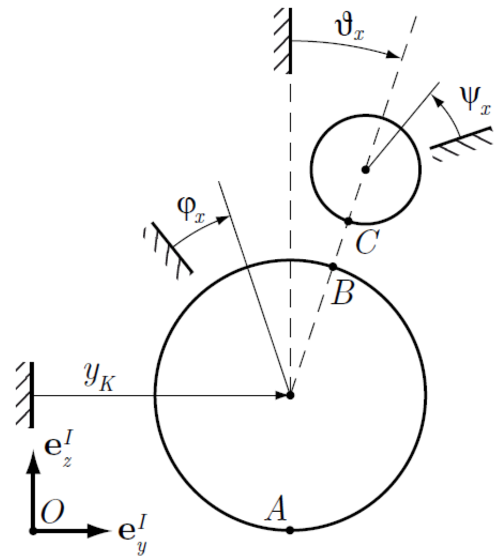
$$v_{By} = v_{Cy}$$

$$v_{Bz} = v_{Cz}$$

$$v_y = \cancel{\dot{y}_k} + \dot{\varphi}_x R_k \cancel{\cos}(\vartheta_x) = \cancel{\dot{y}_k} + \big((R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w\big)\cancel{\cos}(\vartheta_x)$$

$$v_z = -\dot{\varphi}_x R_k \cancel{\sin}(\vartheta_x) = -\big((R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w\big)\cancel{\sin}(\vartheta_x)$$

$$\dot{\varphi}_x R_k = (R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w$$



- We want to know how fast the wheels need to spin for known / induced ball and body rotations

Full Planar Model

- Planar model has two DOFs - φ_{axis} and ϑ_{axis}
- Lets look at the velocities of points A, B, and C

$v_{A-Ccomponent}$	Velocity component of point A, B, or C
φ_{axis}	Ball angle
ϑ_{axis}	Body angle
ψ_{axis}	Wheel angle
y_k	Linear position of ball center
	Time derivative denoted by dot operator

$$\dot{\varphi}_x R_k = (R_k + R_w)\dot{\vartheta}_x + \dot{\psi}_x R_w$$

$$\dot{\psi}_x = \frac{R_k}{R_w}\dot{\varphi}_x - \frac{R_k + R_w}{R_w}\dot{\vartheta}_x$$

$$\dot{\psi}_x = \frac{R_k}{R_w}(\dot{\varphi}_x - \dot{\vartheta}_x) - \dot{\vartheta}_x$$

Virtual wheel angular velocity

Ball angular velocity

Body angular velocity