Omni-Directional Drive and Mecanum: Team 1675 Style

Jon Anderson

jon.c.anderson@gmail.com

FRC Mentor

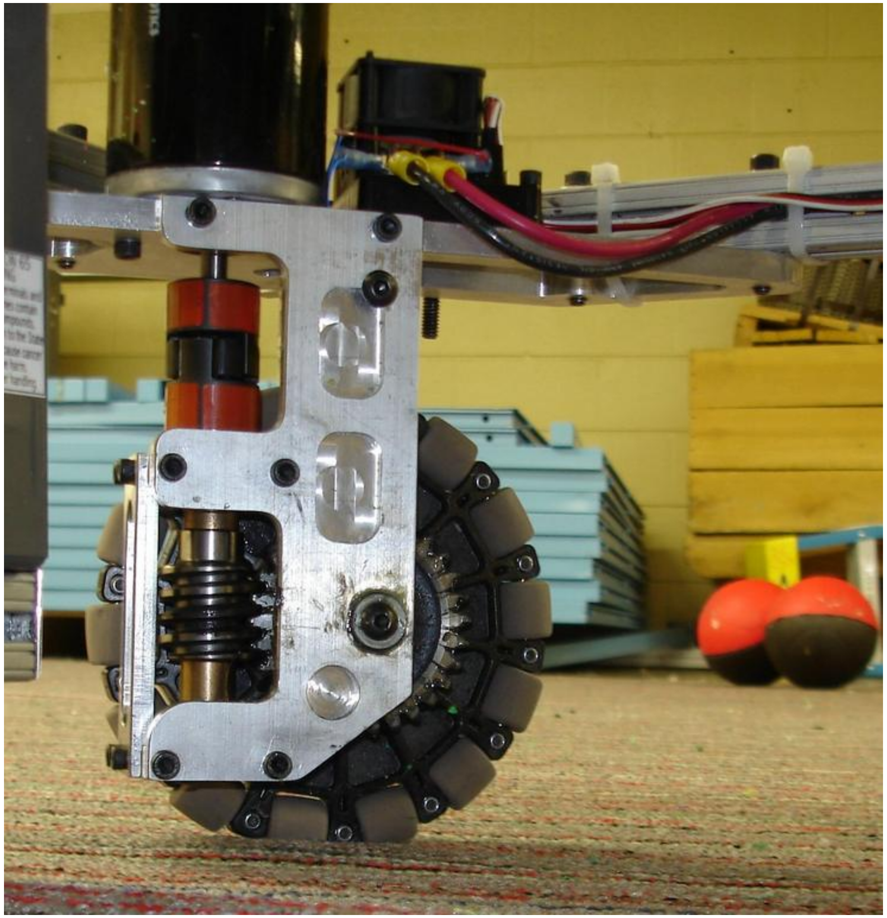
Omni-Directional Drive

- Omni-Directional Drive is Holonomic
  - The controllable degrees of freedom is equal to the total degrees of freedom:
    - Position in two axis (Forward-Backward, Side-to-Side)
    - Orientation
- Why would you want Omni-Directional Drive?
  - High maneuverability
  - Strafing
  - Travel in any direction
  - Maintain orientation while turning
  - Zero-radius turning

Omni-Directional Drive Trains

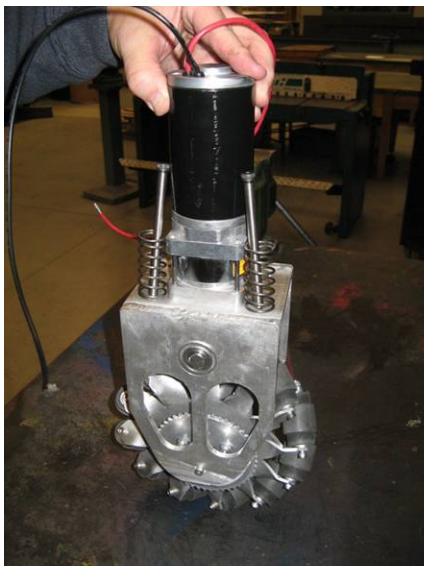
- **Omni-Wheels**

- Mecanum
- Swerve/Crab Drive



Omni-Directional Drive Trains

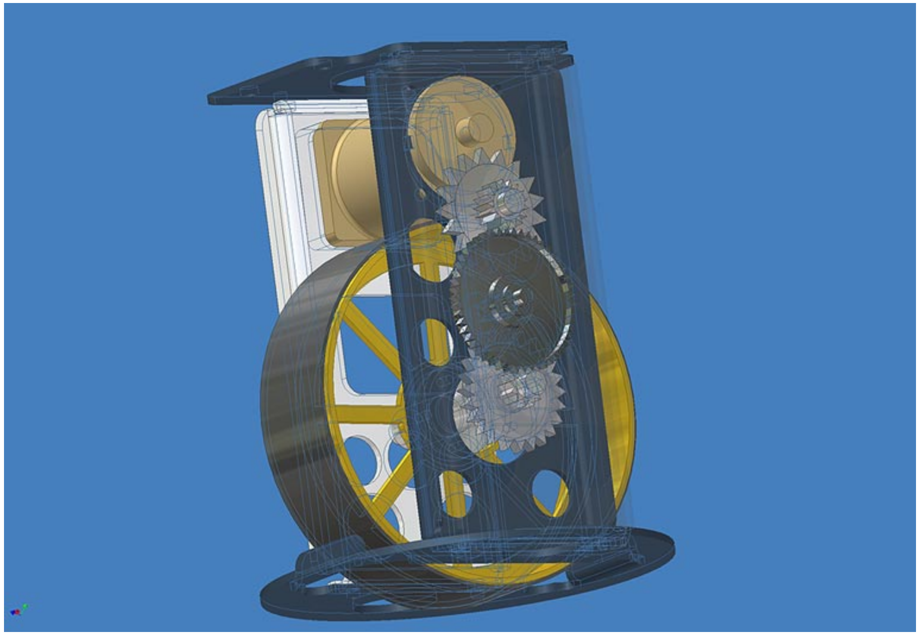
- Omni-Wheels
- **Mecanum**

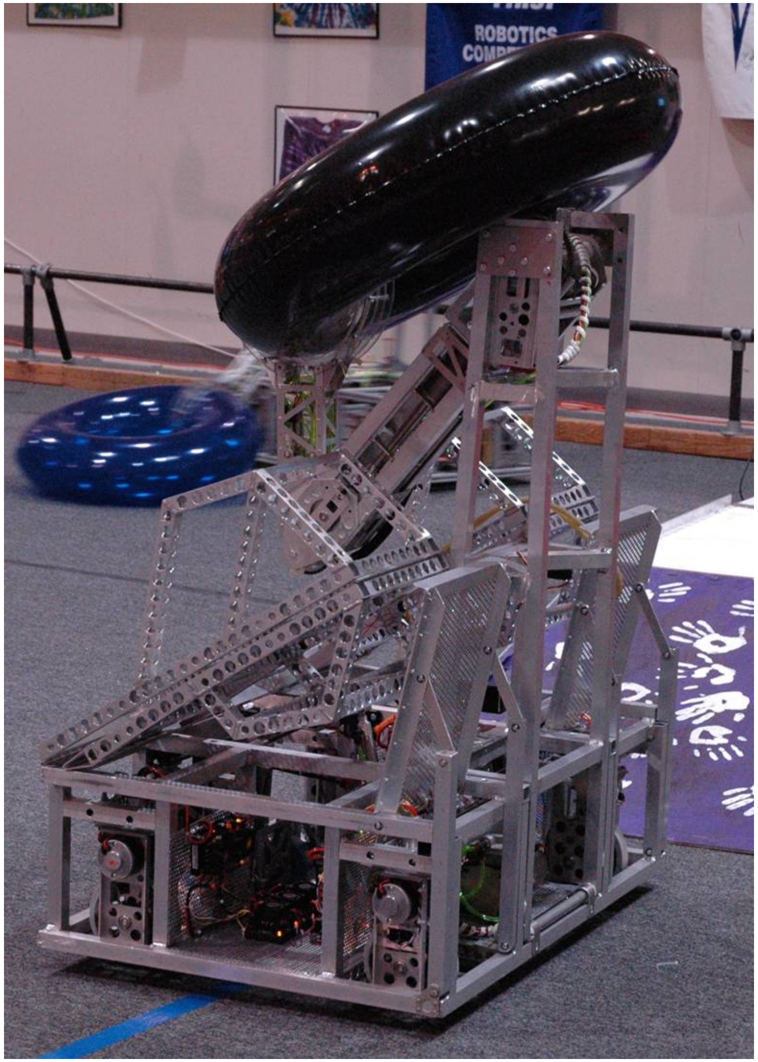


- Swerve/Crab Drive

Omni-Directional Drive Trains

- Omni-Wheels
- Mecanum
- **Swerve/Crab Drive**

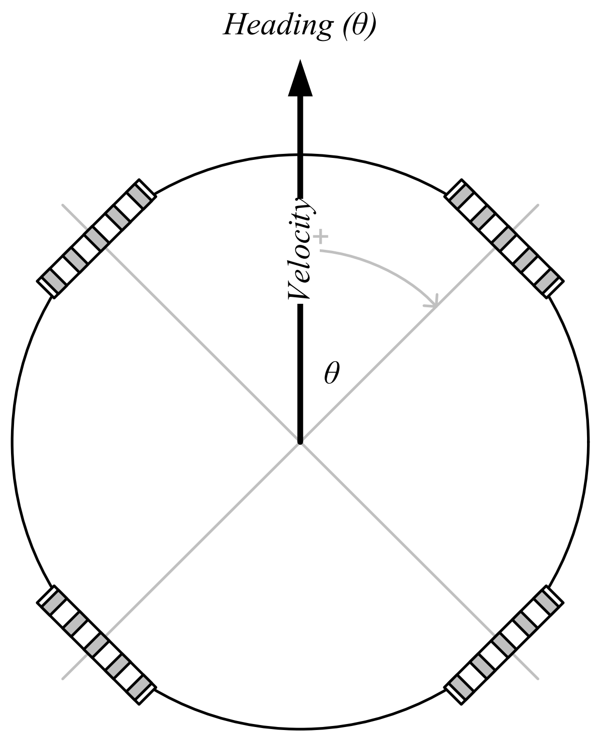




Implementation Considerations

	Speed	Pushing Power	Climbing Ability	Build Complexity	Robustness	Motors Required	Control Complexity	Weight
Omni	Full Speed on Diagonals	Low Torque	None	Low	Low	4	Medium	Light
Mecanum	Full Speed Backward and Forward	Medium Torque	Low	Very Low	High	4	Medium	Moderate
Swerve (Crab)	Full Speed in all Directions	Full Torque	Medium	High	Low	5 to 8	High	Heavy

Omni Control: Overview

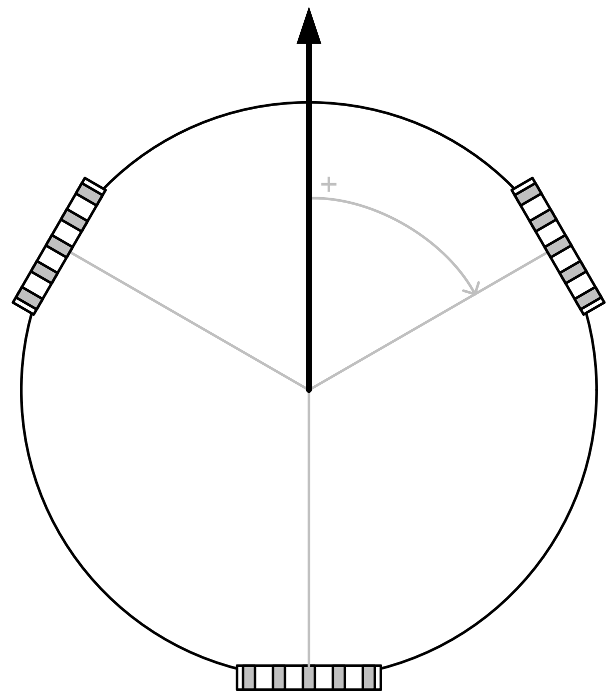


$$Heading = Velocity \angle \theta$$

Orientation and Wheel Location

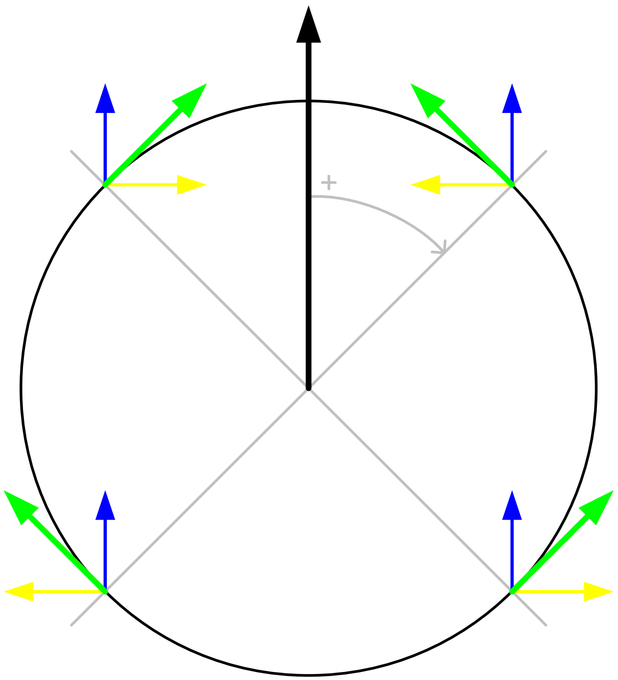
Omni Control: Lots of Wheels?

As many as you would like…
Works from 2 to infinity



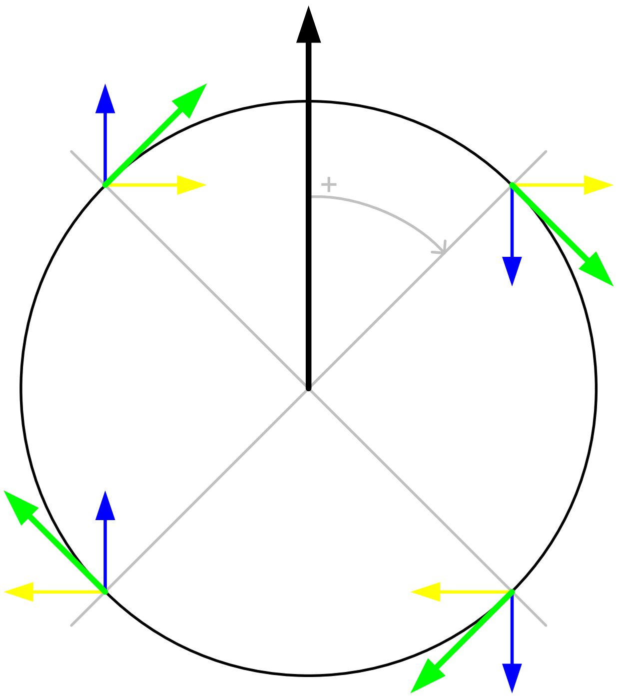
3-Wheel Omni-Drive

Omni Control: Logical Forces



Logical Force Vectors

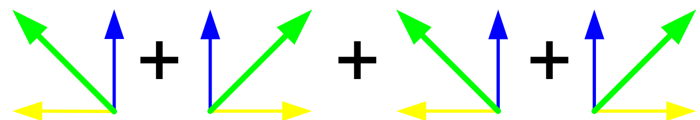
Omni Control: Actual Forces



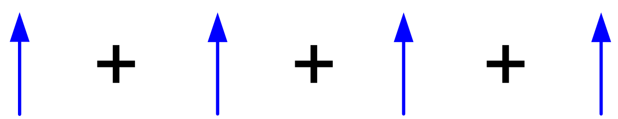
Wheel Force Vectors

Omni Control: Forces on Robot

$$\sum_{i=1}^{n} \vec{F}_i =$$



$$\sum_{i=1}^{n} \vec{F}_i =$$



Force Vector Summation

Omni Control: Motor Outputs

$$MotorOutput_n = Velocity \times \sin(\,Heading - MotorOffset_n\,) - Rotation$$

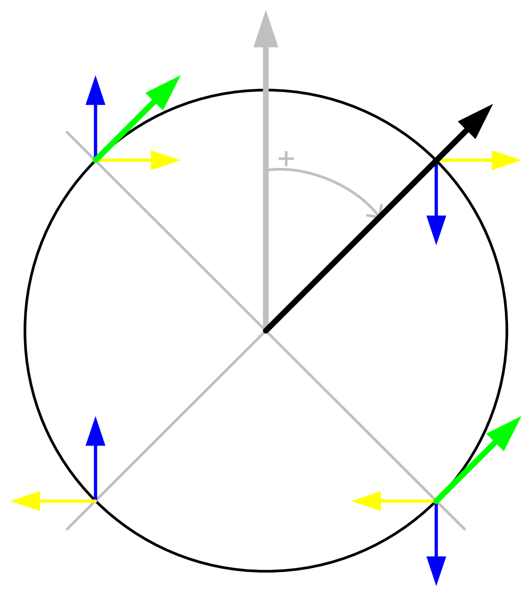
Example 1: 4 motors (Labeled Clockwise), (½ @ ∠π/4)

$MotorOutput_1 = 1/2 \times \sin(\,\pi/4 - 1\pi/4\,) - 0 = 0$

$MotorOutput_2 = 1/2 \times \sin(\,\pi/4 - 3\pi/4\,) - 0 = -1/2$

$MotorOutput_3 = 1/2 \times \sin(\,\pi/4 - 5\pi/4\,) - 0 = 0$

$MotorOutput_4 = 1/2 \times \sin(\,\pi/4 - 7\pi/4\,) - 0 = 1/2$



Omni Control: Motor Outputs

$$MotorOutput_n = Velocity \text{ x } \sin(\,Heading - MotorOffset_n\,) - Rotation$$

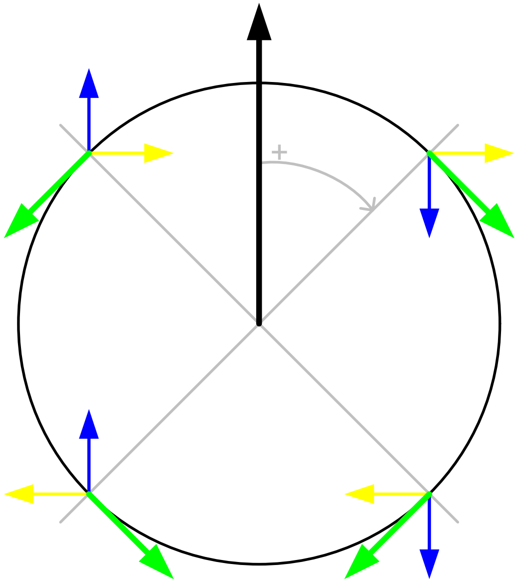
Example 2: 4 motors (Labeled Clockwise), (1 @ ∠-π)

$MotorOutput_1 = 1 \text{ x } \sin(\,-\pi - 1\pi/4\,) - 0 = \sqrt{2}/2$

$MotorOutput_2 = 1 \text{ x } \sin(\,-\pi - 3\pi/4\,) - 0 = \sqrt{2}/2$

$MotorOutput_3 = 1 \text{ x } \sin(\,-\pi - 5\pi/4\,) - 0 = -\sqrt{2}/2$

$MotorOutput_4 = 1 \text{ x } \sin(\,-\pi - 7\pi/4\,) - 0 = -\sqrt{2}/2$

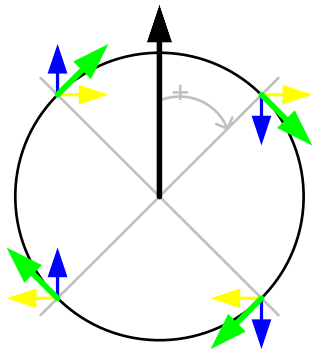


Motor Output
Calculation 2

Omni Control: The Twist

$$MotorOutput_n = Velocity \text{ x } \sin(\, Heading - MotorOffset_n\,) - Rotation$$

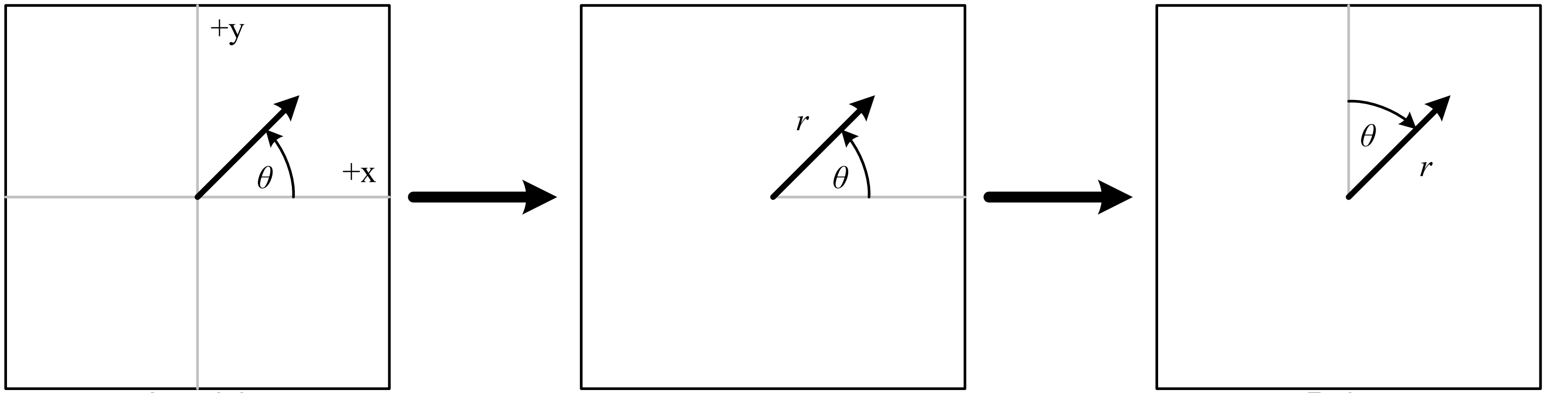
Rotation – or yaw rate – is a scalar rate added to each *MotorOutput* on top of the scaled *Velocity* calculation. As such, when *MotorOutput* has already reached 100% (when the *Heading* is an odd multiple of $\pi/4$) it has no effect. If the Omni-Drive has 4 motors as in this example there is no real problem as the opposing *MotorOutput*s are at 0%. In this example it is good practice to limit the rotation scalar to 30% to allow the rate of turn to be approximately equal when traveling in all directions



Rotation is often controlled by the yaw axis on a joystick that can twist or by a secondary input such as an additional joystick or potentiometer.

Rotation Control

Omni Control: Human Control



Joystick
(Cartesian Coordinates)

Polar Coordinates

Robot
(Navigation Coordinates)

Cartesian to Polar

$$r = \sqrt{x^2 + y^2} \qquad \theta = \arctan(\, y/x \,)$$

r should be limited to ±100% (not more than full speed)
In the case that $x = 0$:
- if $y > 0$ then $\theta = \pi/2$
- if $y < 0$ then $\theta = -\pi/2$
- if $y = 0$ then $\theta = 0$

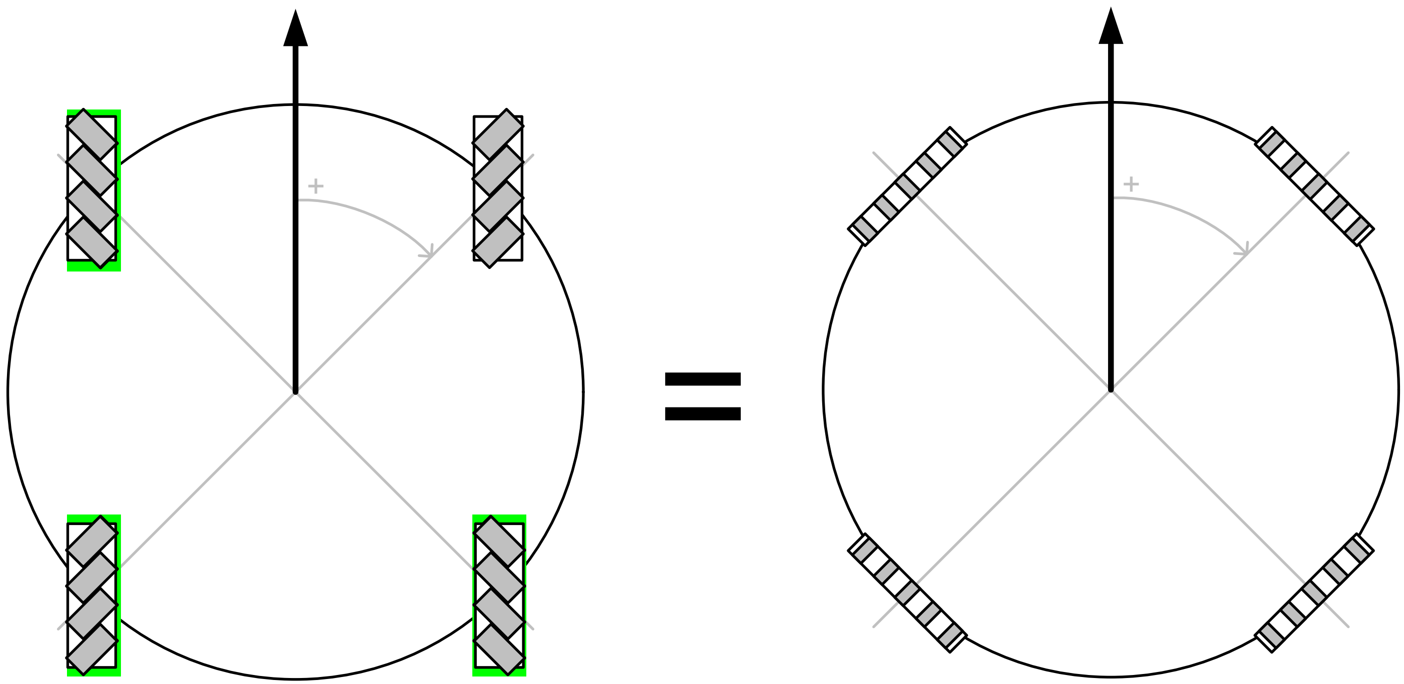
Programmatically this is mitigated by using the atan2 function

Polar to Navigation

$$\theta_{nav} = 2\pi + \pi/2 - \theta_{polar}$$

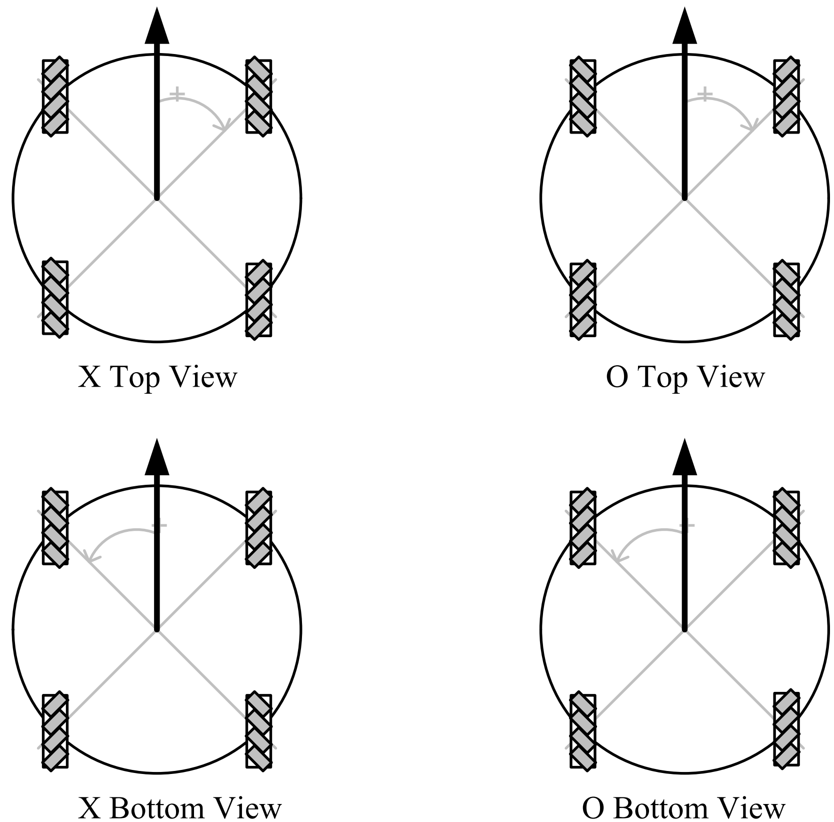
```
Calculating Heading and
Velocity
```

Mecanum Control: It's the Same



Mecanum-Drive

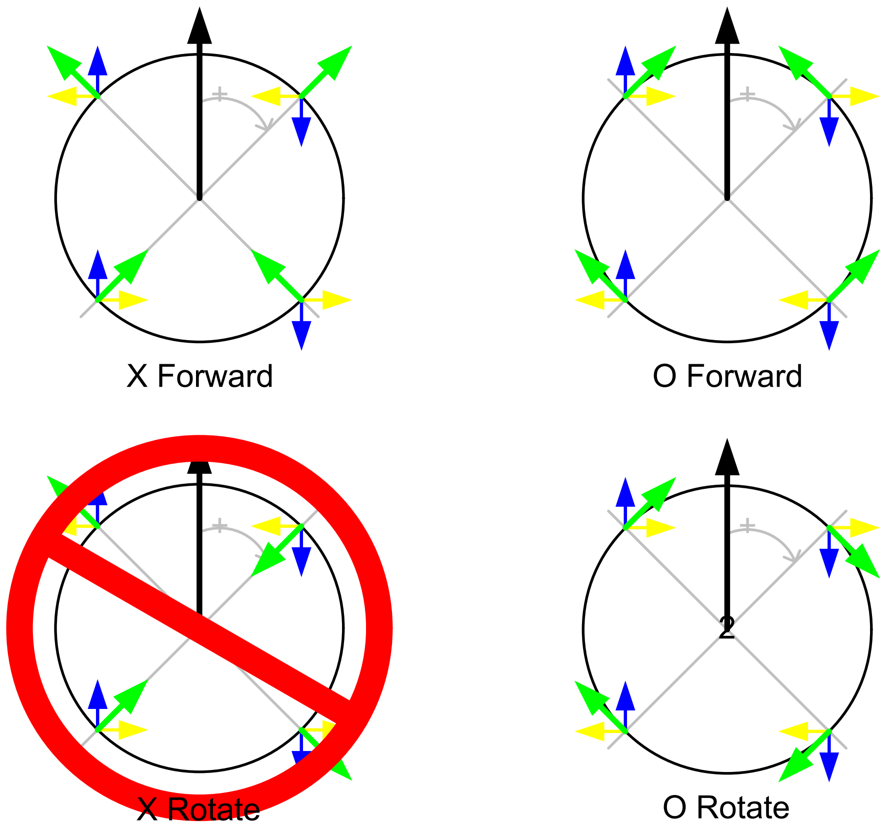
Mecanum Configuration: X vs O



X vs O Wheel Configuration

Mecanum Rotation: X vs O

No moment for X



X vs O Rotation

Mecanum: Other Considerations

- To maintain control all wheels must have equal downward force applied
- Orientation reference can be shifted from the point of view of the field to the robot using feedback (Yaw Rate Sensor, Encoders, etc)
- Full speed forward and backward can be achieved with an override to the MotorOutput algorithm’s input velocity or output velocity

Mecanum: Other Considerations

- One Joystick Control: Needs to have the z-axis on the joystick allowing for yaw control
- Two Joystick Control:
  - Tank-style control having left wheels on left stick and right wheels on right stick
  - Translation on joystick A and yaw rate on joystick B or potentiometer, etc
  - Forward/Reverse and yaw rate on joystick A and strafe on joystick B

Resources

- Mecanum Wheels
  - AndyMark 8” Mecanum Wheels http://www.andymark.com/ProductDetails.asp?ProductCode=am-0083
- More Details and Differing Control Strategy
  - Ether’s white paper “Mecanum and Omni Kinematic and Force Analysis and Programming” http://www.chiefdelphi.com/media/papers/2390
- An Introduction to Programming Holonomic Robots
  - Jack Buffington, “Holonomic Drive Platforms: How to Drive a Robot That Has No Front”, Servo Magazine, April 2005

Acknowledgements

- Omni Wheel Photo – AndyMark
- Omni Robot Photo – FRC Team 81
- Mecanum Pod and Mecanum Robot Photos – FRC Team 1675
- Crab Pod and Crab Robot – FRC Team 111