Week 3: Wheeled Kinematics

AMR - Autonomous Mobile Robots

Paul Furgale

Margarita Chli, Marco Hutter, Martin Rufli, Davide Scaramuzza, Roland Siegwart

AMRx Flipped Classroom

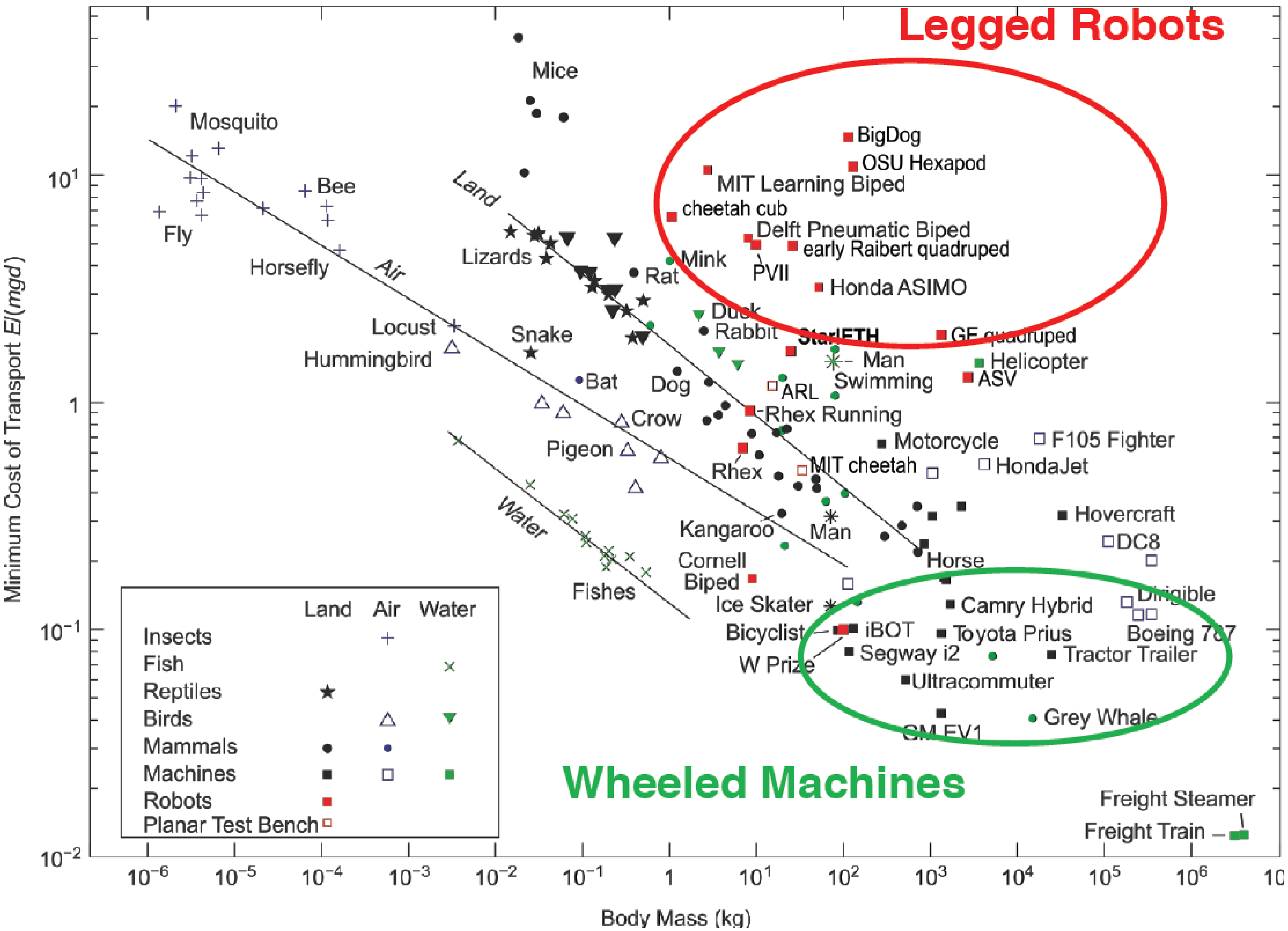
- A Matlab exercise is coming later in the class. Download it now!
  - **http://tinyurl.com/amrx-wheels**

AMRx
A review

- Please register in at edx.org and watch the lecture segments
- Please try the problem sets online (they don’t count for your grade but they are useful)
- Bring your laptop along for the lectures
  - We may provide short examples or problems for you to work out
  - We may ask for feedback on the worked exercises
- Flipped Classroom
  - You ask us, we try to help
  - No recording of the lecture, no questions are stupid
  - Feel free to interrupt

Why Wheeled Robots?

+ Requires small control action (in comparison to legs)
+ Energetically very efficient

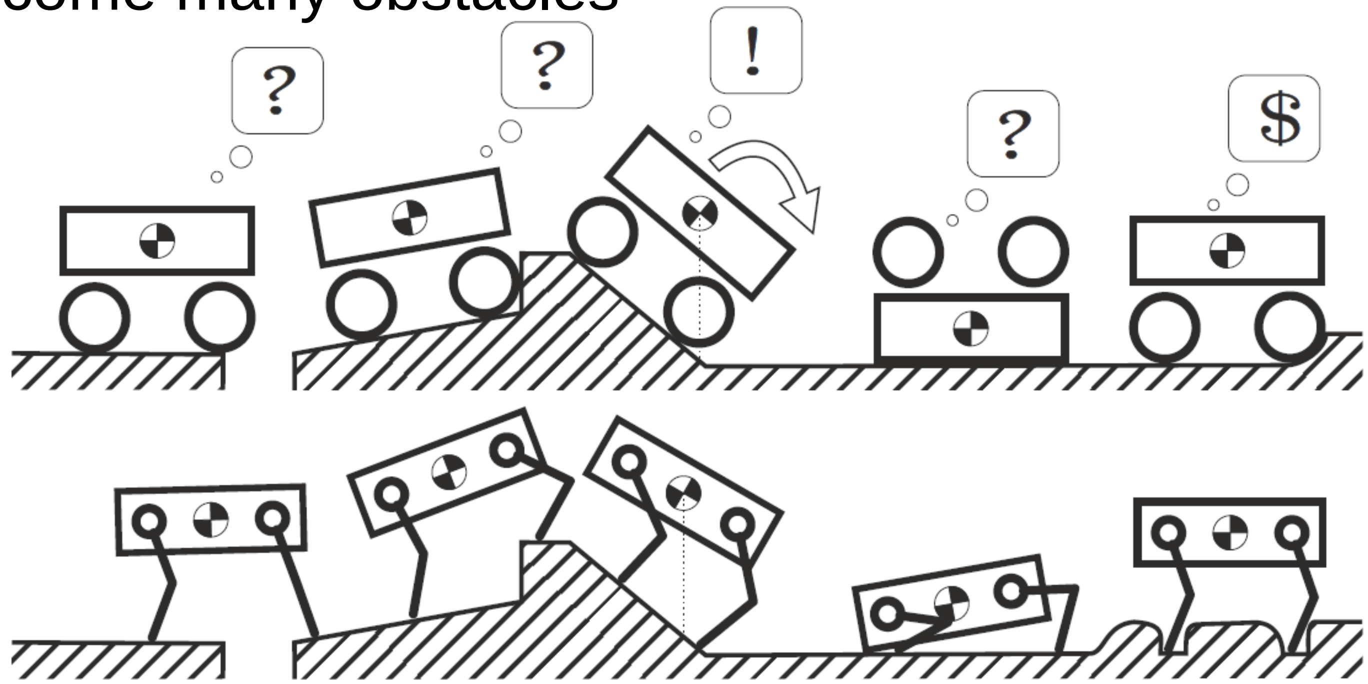


Why Wheeled Robots?

+ Requires small control action

+ Energetically very efficient

- Unable to overcome many obstacles



AMRx
Today's lecture

- Teacher: Paul Furgale, Deputy Director @ ASL

- Review
- Practical problems:
  - Intuition about matrices
  - Computing good steering angles
  - Computing wheel odometry

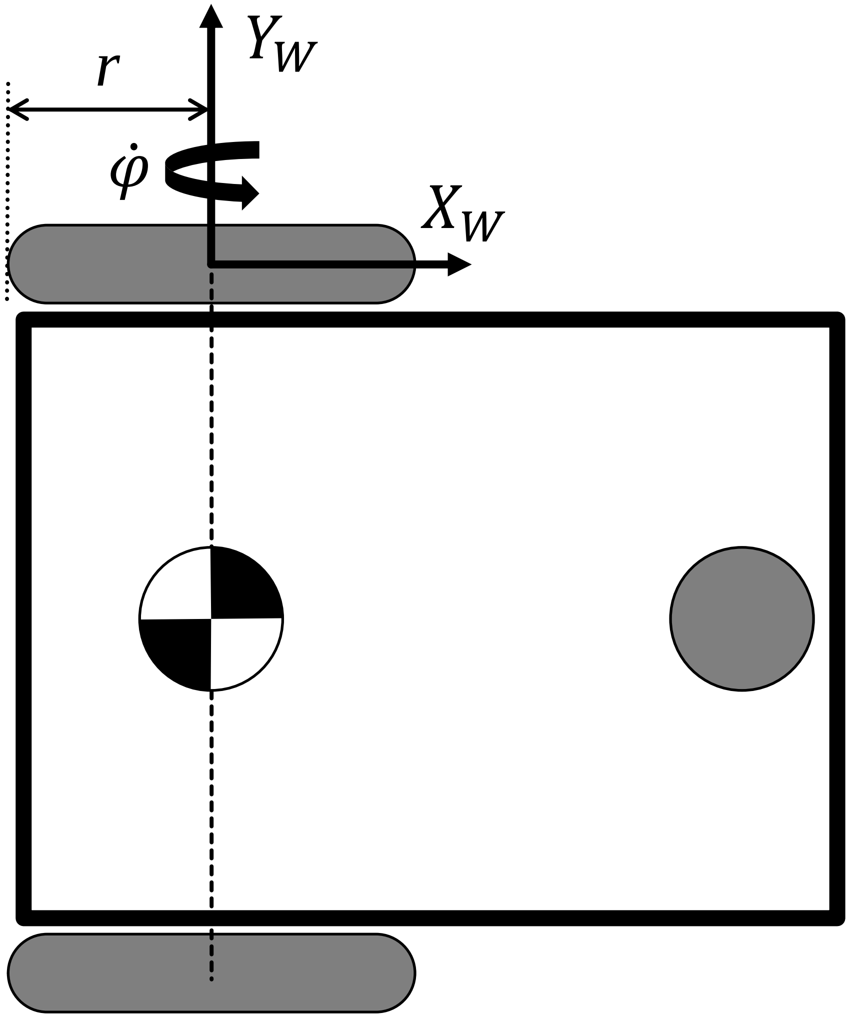
Wheeled Kinematics

- Not all degrees of freedom of a wheel can be actuated or have encoders
- Wheels can impose **differential constraints** that complicate the computation of kinematics

rolling constraint

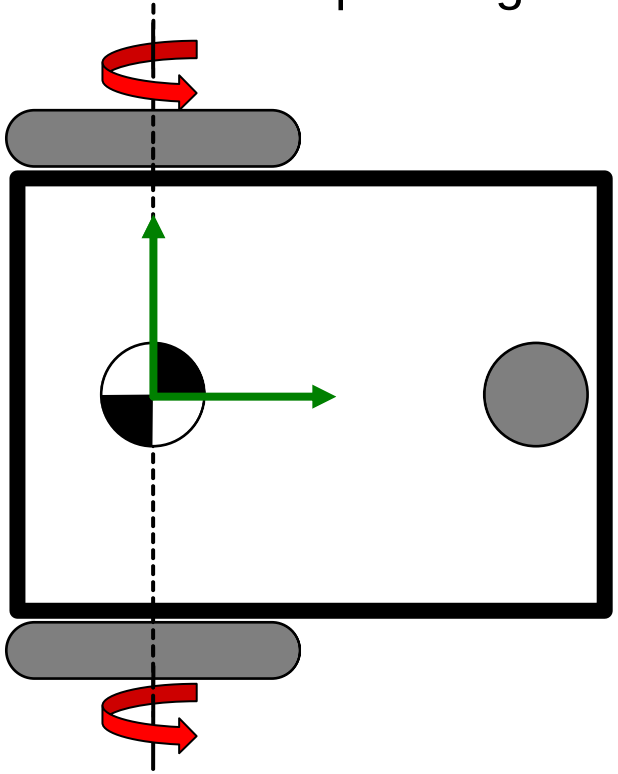
$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \dot{\varphi} r \\ 0 \end{bmatrix}$$

no-sliding constraint

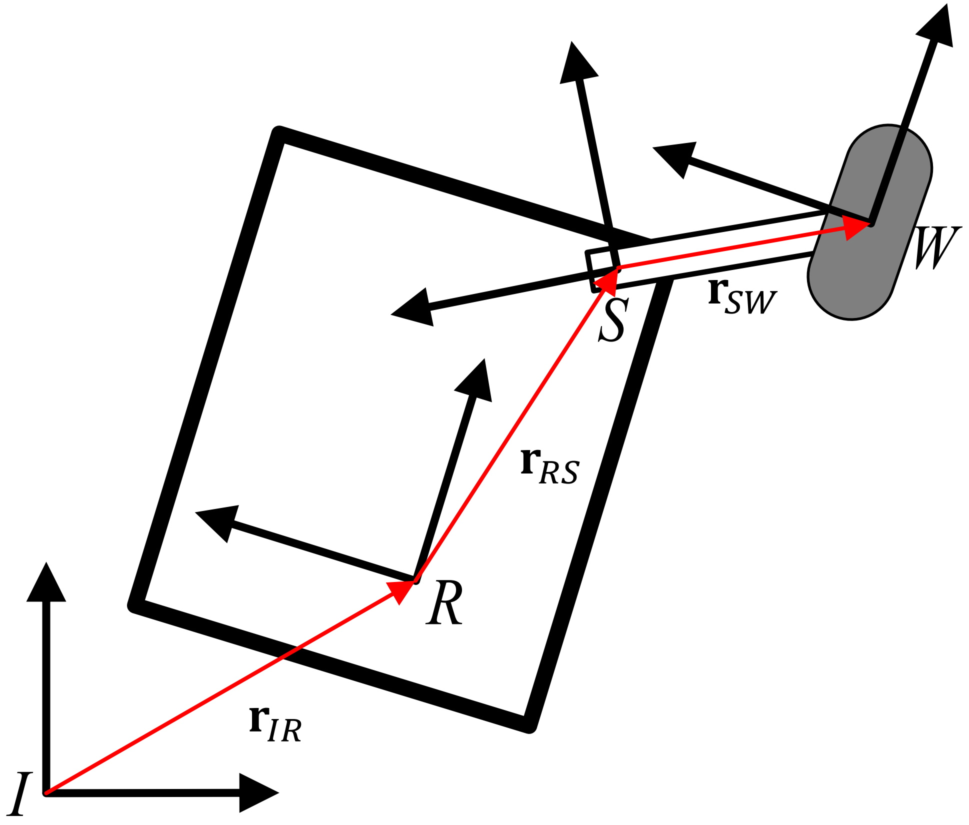


Differential Kinematics

- Differential forward kinematics
  - Given a set of actuator speeds, determine the corresponding velocity
- **Differential inverse kinematics**
  - Given a desired velocity, determine the corresponding actuator speeds

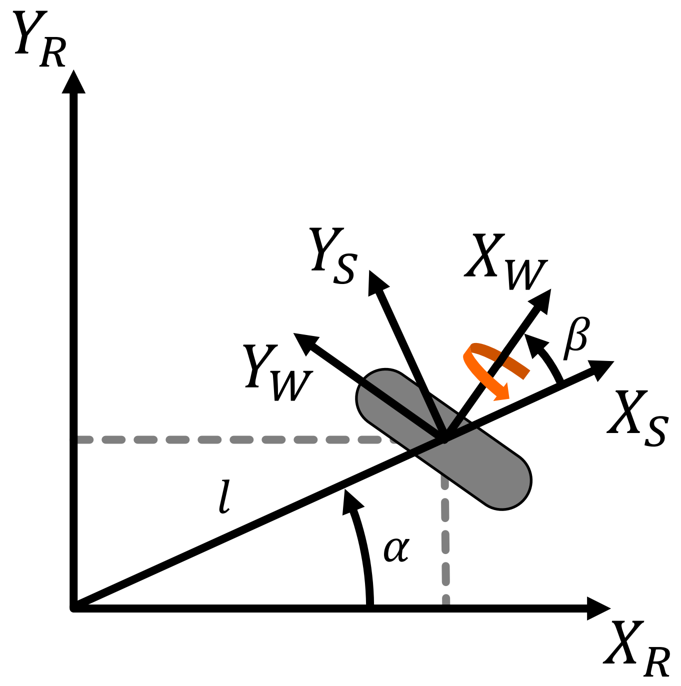


Deriving a general wheel equation



$$\mathbf{v}_{IW} = \underbrace{\mathbf{v}_{IR}}_{\text{robot velocity}} + \overbrace{\boldsymbol{\omega}_{IR} \times \underbrace{\mathbf{r}_{RS}}_{\text{steering offset}} + \boldsymbol{\omega}_{IR}}^{\text{robot angular velocity}} \times \underbrace{\mathbf{r}_{SW} + \overbrace{\boldsymbol{\omega}_{RS}}^{\text{steering rate}} \times \mathbf{r}_{SW}}_{\text{wheel offset}}$$

Example: Standard wheel



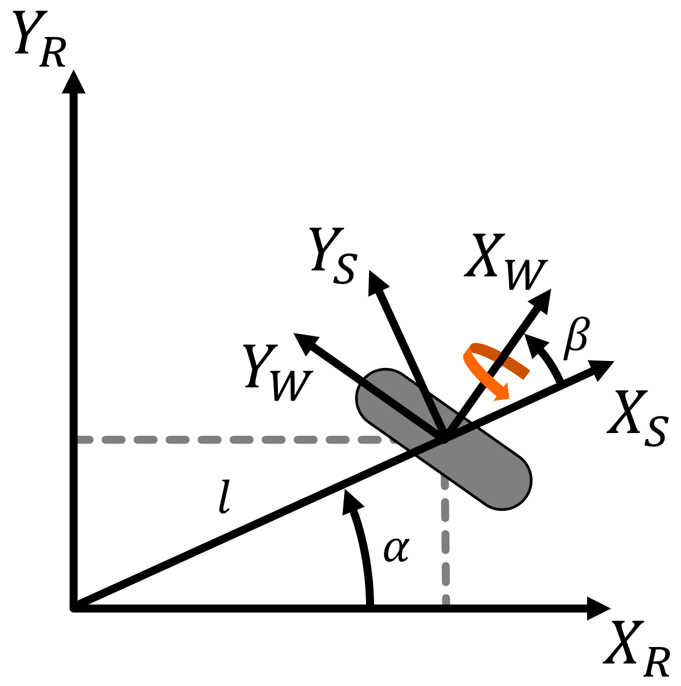
- Rolling constraint

$$[\sin \alpha + \beta \quad -\cos \alpha + \beta \quad -l \cos \beta] R(\theta) \dot{\xi}_I - \dot{\varphi} r = 0$$

- No-sliding constraint

$$[\cos \alpha + \beta \quad \sin \alpha + \beta \quad l \sin \beta] R(\theta) \dot{\xi}_I = 0$$

Example: Standard wheel



- This equation had a mistake in the online lecture. The slides have been updated and we will update the video soon
- No-sliding constraint

$$[\cos\alpha+\beta \quad \sin\alpha+\beta \quad l\sin\beta]R(\theta)\dot{\xi}_I = 0$$

Differential Kinematics

- Given a wheeled robot, each wheel imposes n constraints
- Only fixed and steerable standard wheels impose no-sliding constraints
- Suppose a robot has n wheels of radius r_i, the individual wheel constraints can be concatenated in matrix form:
  - Rolling constraints

$$J_1(\beta_S)R(\theta)\dot{\xi}_I - J_2\dot{\varphi} = 0, \quad J_1(\beta_S) = \begin{bmatrix} J_{1f} \\ J_{1S}(\beta_S) \end{bmatrix}, \quad J_2 = diag(r_1, \cdots, r_N), \quad \dot{\varphi} = \begin{bmatrix} \dot{\varphi}_1 \\ \vdots \\ \dot{\varphi}_N \end{bmatrix}$$

  - No-sliding constraints

$$C_1(\beta_S)R(\theta)\dot{\xi}_I = 0, \quad C_1(\beta_S) = \begin{bmatrix} C_{1f} \\ C_{1S}(\beta_S) \end{bmatrix}$$

Differential Kinematics

- Stacking the rolling and no-sliding constraints gives an expression for the differential kinematics

$$\begin{bmatrix} J_1(\beta_s) \\ C_1(\beta_s) \end{bmatrix} R(\theta)\dot{\xi}_I = \begin{bmatrix} J_2 \\ 0 \end{bmatrix} \dot{\varphi}$$

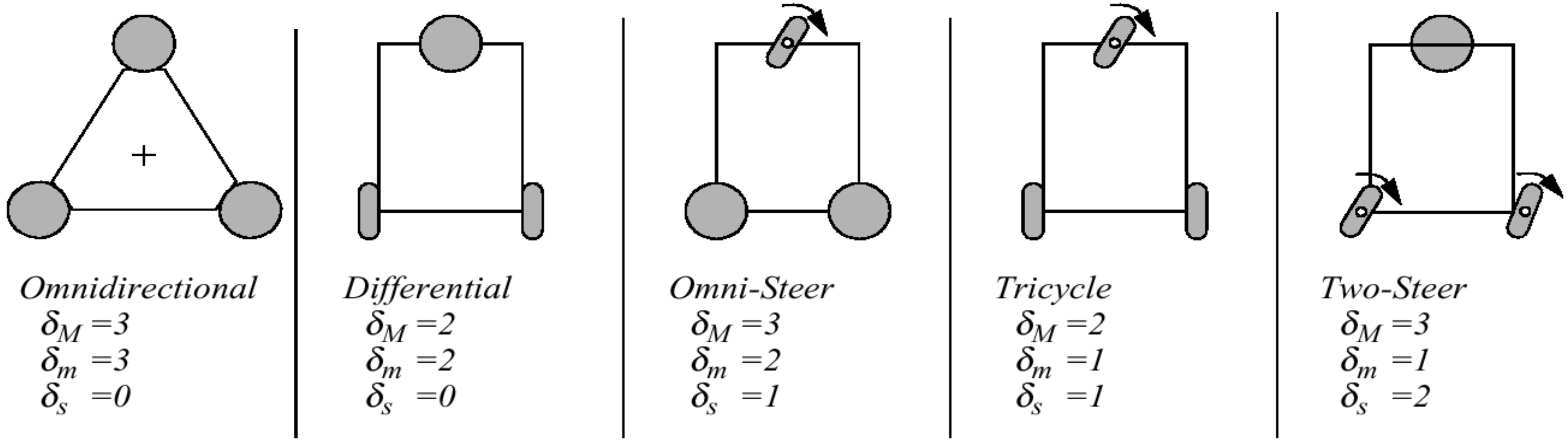
- Solving this equation for $\dot{\xi}_I$ yields the **forward differential kinematics** equation needed for computing wheel odometry
- Solving this equation for $\dot{\varphi}$ yields the **inverse differential kinematics** equation needed for control

Degree of Maneuverability

- The Degree of Maneuverability, δ_M, combines mobility and steerability

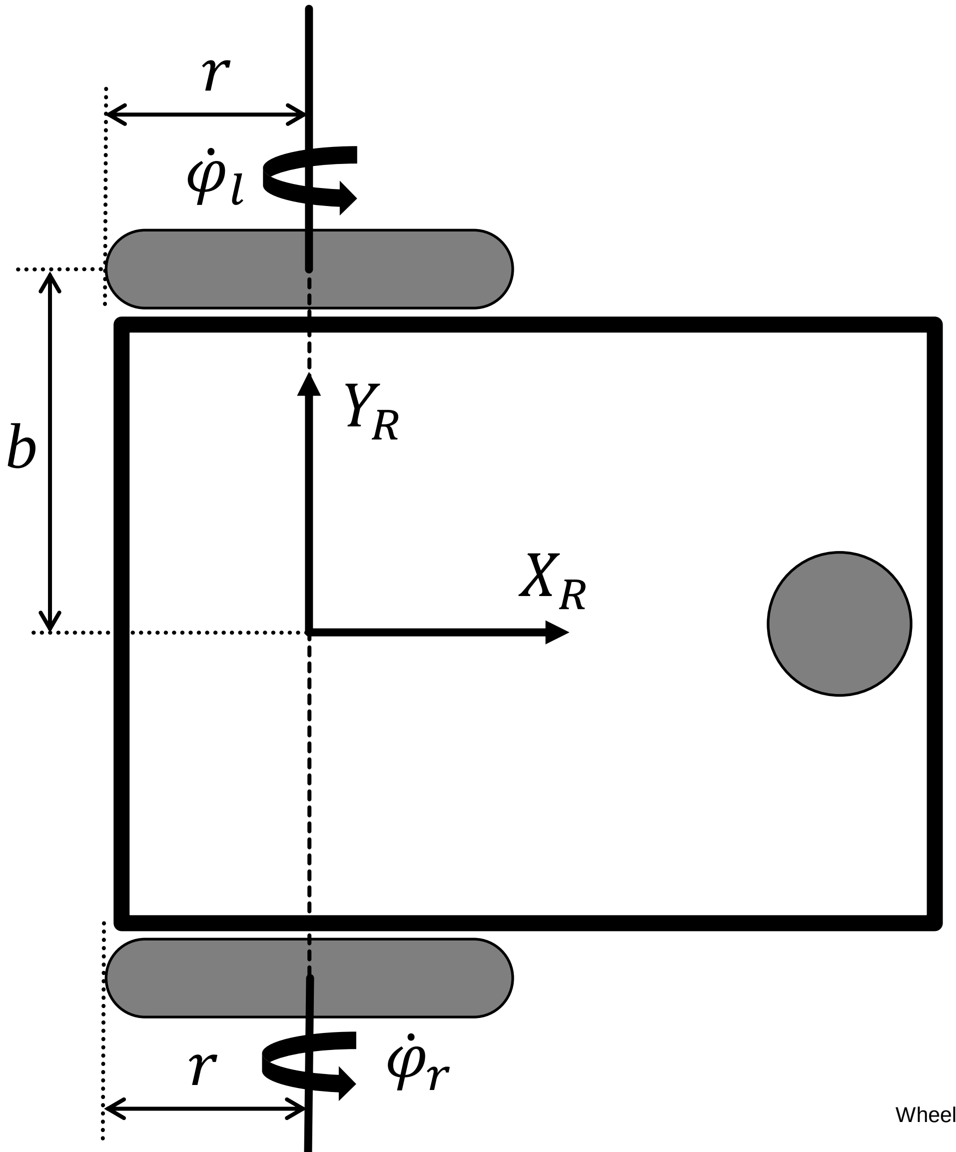
$$\delta_M = \delta_m + \delta_s$$

- Examples



Worked Exercise | A Differential Drive Robot

- Two fixed standard wheels
- The robot frame (R) in between the wheels
- Stack the wheel equations for this configuration



A Differential Drive Robot | Summary

- Degree of Maneuverability

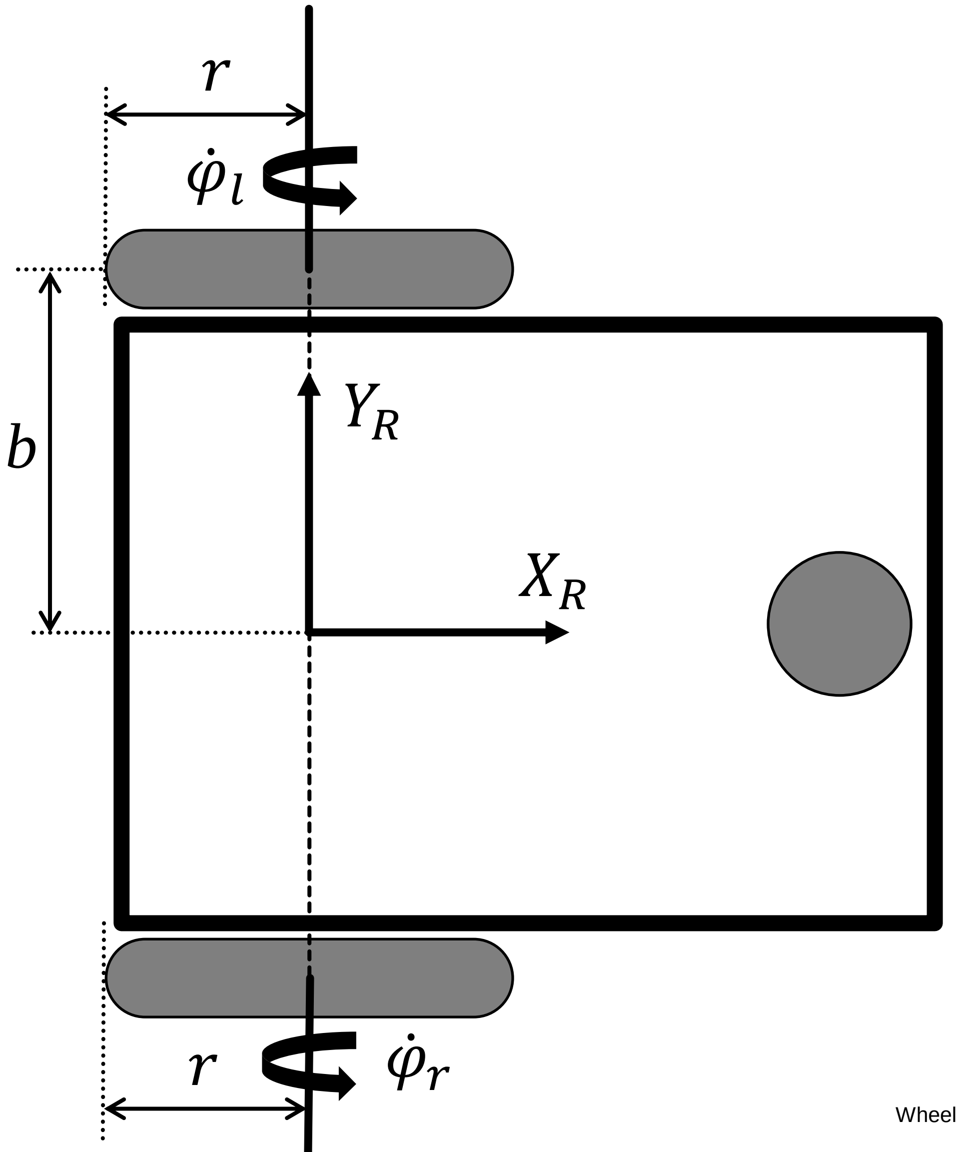
$$\delta_m = 2, \qquad \delta_s = 0, \qquad \delta_M = 2$$

- Forward differential kinematics

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} r/2 & r/2 \\ 0 & 0 \\ r/2b & -r/2b \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix}$$

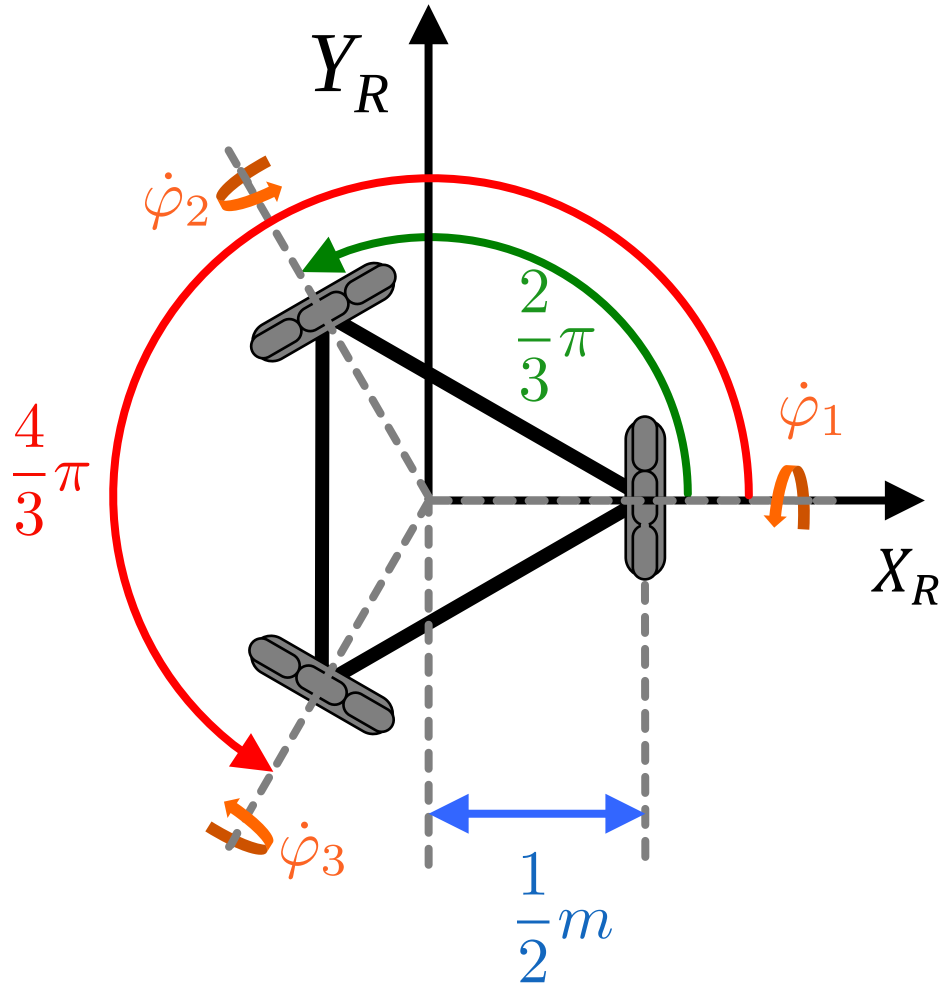
- Inverse differential kinematics

$$\begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} = \begin{bmatrix} 1/r & 0 & b/r \\ 1/r & 0 & -b/r \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

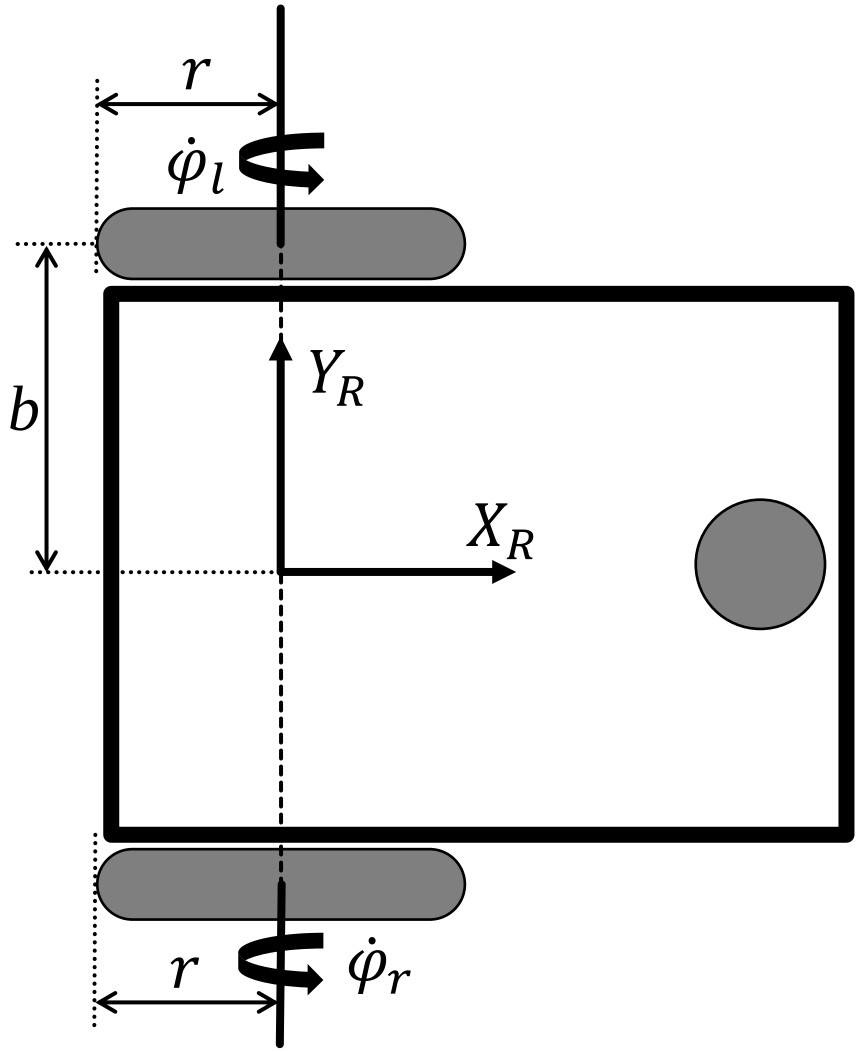


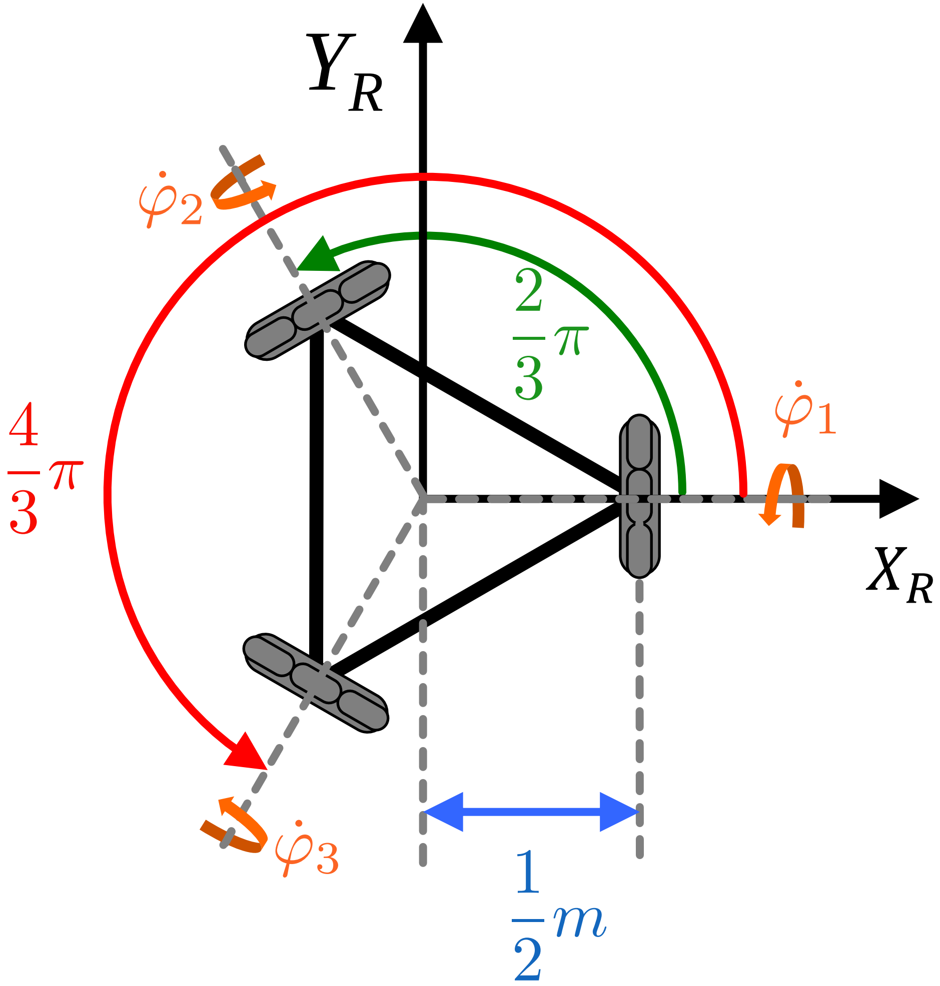
An Omnidirectional Robot

- This robot was the example in the online problem set.
- Were there any questions about the example?



Simple Robots

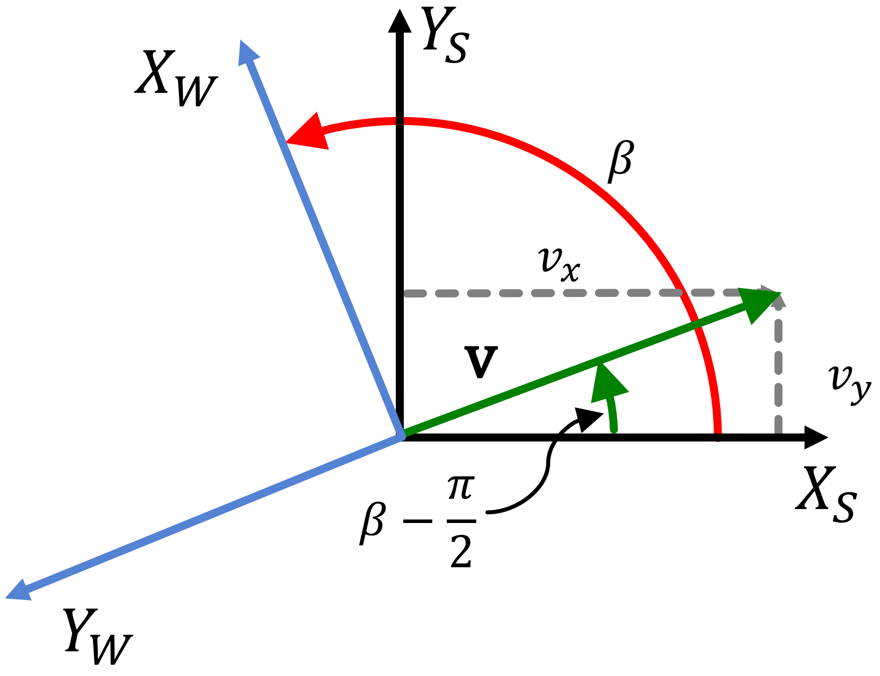




Practical Problems

- **Head over to MATLAB and get into groups**
- **http://tinyurl.com/amrx-wheels**

Computing the steering angle



$$\beta = \frac{\pi}{2} + \operatorname{atan}(v_y / v_x)$$

- To compute the steering angle, we must transform the velocity of the vehicle into the steering frame S, then determine the angle, β, that brings the velocity into the negative y direction in the wheel frame.

Computing Wheel Odometry

- Given sensed wheel speeds at time *t*, we know how to generate a predicted platform velocity:

$$\dot{\xi}_R(t) = \mathbf{F}\,\dot{\phi}(t)$$

- Recall that

$$\dot{\xi}_R = R(\theta)\dot{\xi}_I$$

- In the inertial frame, the velocity equation is:

$$\dot{\xi}_I(t) = \mathrm{R}(-\theta(t))\mathbf{F}\,\dot{\phi}(t)$$

Computing Wheel Odometry

- Assumptions:
  - Timesteps of size T
  - Constant acceleration

$$\dot{\xi}_R(t) = \dot{\xi}_R(t_0) + \frac{t-t_0}{T}\left(\dot{\xi}_R(t_1) - \dot{\xi}_R(t_0)\right), \qquad t_0 \leq t < t_1$$

- The velocity integration equation:

$$\xi_I(t_k) = \xi_I(t_{k-1}) + \int_{t_{k-1}}^{t_k} \dot{\xi}_I(t)dt$$

Computing Wheel Odometry

- Assumptions:
  - Timesteps of size T
  - Constant acceleration

$$\dot{\xi}_R(t) = \dot{\xi}_R(t_0) + \frac{t-t_0}{T}\left(\dot{\xi}_R(t_1) - \dot{\xi}_R(t_0)\right), \qquad t_0 \leq t < t_1$$

- The velocity integration equation:

$$\xi_I(t_k) \approx \xi_I(t_{k-1}) + \frac{\mathrm{T}}{2}\mathrm{R}(-\theta(t_{k-1}))(\dot{\xi}_R(\mathrm{t_{k-1}}) + \dot{\xi}_R(\mathrm{t_k}))$$

Computing Wheel Odometry

- Assumptions:
  - Timesteps of size T
  - Constant acceleration

$$\dot{\xi}_R(t) = \dot{\xi}_R(t_0) + \frac{t-t_0}{T}\left(\dot{\xi}_R(t_1) - \dot{\xi}_R(t_0)\right), \qquad t_0 \le t < t_1$$

- The velocity integration equation:

$$\xi_I(t_k) \approx \xi_I(t_{k-1}) + \frac{\mathrm{T}}{2}\mathrm{R}(-\theta(t_{k-1}))(\mathbf{F}(\mathrm{t}_{\mathrm{k-1}})\dot{\phi}(\mathrm{t}_{\mathrm{k-1}}) + \mathbf{F}(\mathrm{t}_{\mathrm{k}})\dot{\phi}(\mathrm{t}_{\mathrm{k}}))$$

Computing Wheel Odometry

- The velocity integration equation:

$$\xi_I(t_k) \approx \xi_I(t_{k-1}) + \frac{\mathrm{T}}{2}\mathrm{R}(-\theta(t_{k-1}))(\mathbf{F}(\mathrm{t_{k-1}})\dot{\phi}(\mathrm{t_{k-1}}) + \mathbf{F}(\mathrm{t_k})\dot{\phi}(\mathrm{t_k}))$$