Principles of Robot Autonomy I

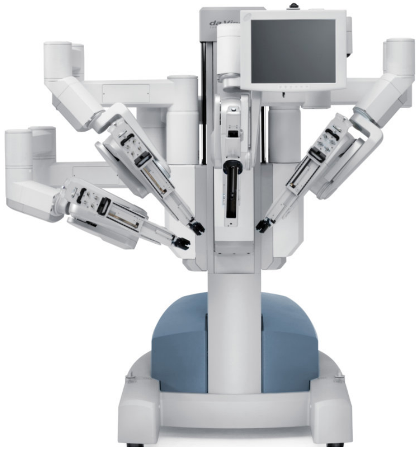
Course overview, mobile robot kinematics

Stanford University

Autonomous Systems Lab

From automation…

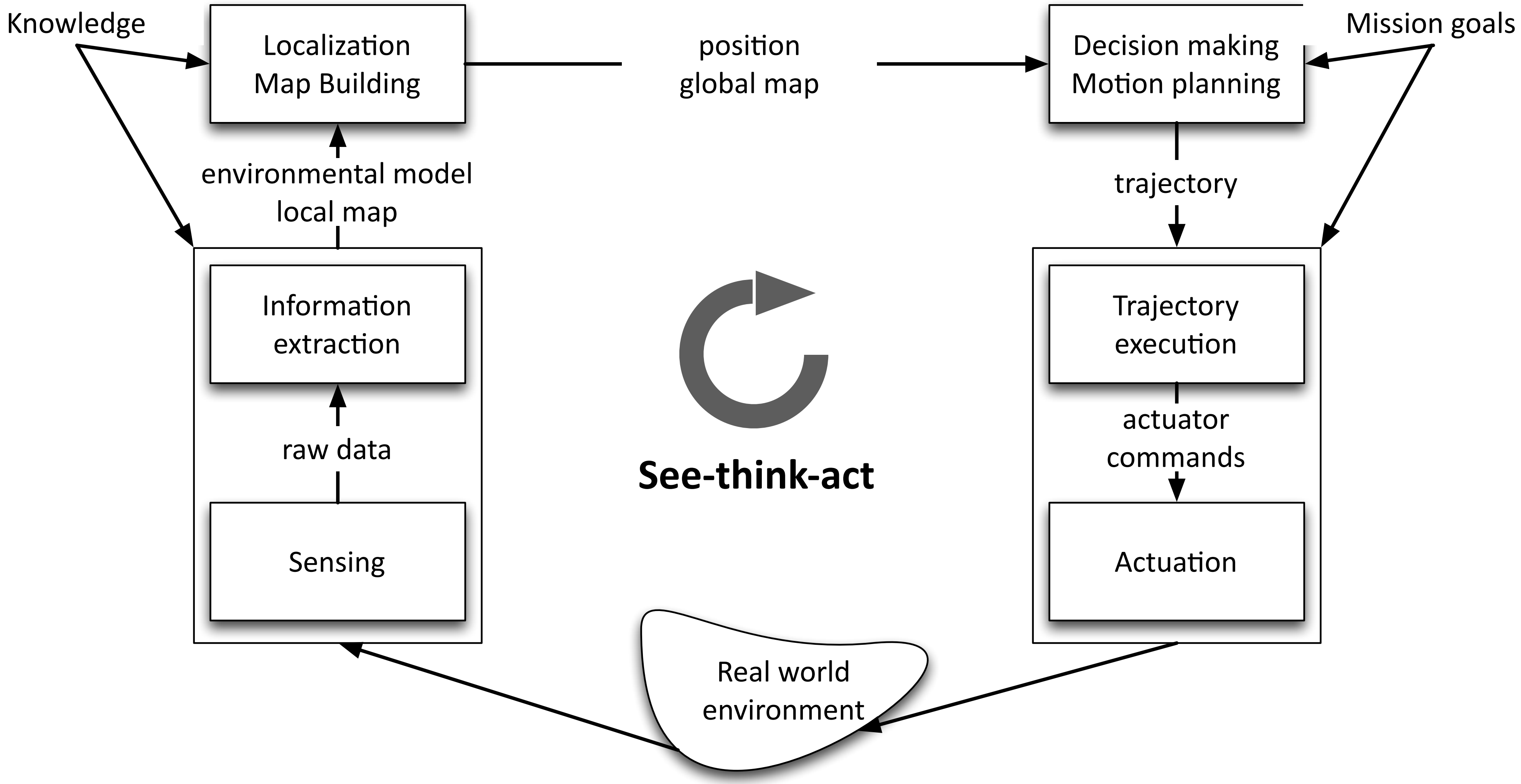
…to autonomy



Course goals

- To learn the *theoretical*, *algorithmic*, and *implementation* aspects of main techniques for robot autonomy. Specifically, the student will
    1. Gain a fundamental knowledge of the “autonomy stack”
    2. Be able to apply such knowledge in applications / research by using ROS
    3. Devise novel methods and algorithms for robot autonomy

The see-think-act cycle



Course structure

- Four modules, roughly of equal length
    1. motion control and planning
    2. robotic perception
    3. localization and SLAM
    4. state machines, decision making, and system architecture

- Extensive use of the Robot Operating System (ROS)

- Requirements
    - CS 106A or equivalent
    - CME 100 or equivalent (for linear algebra)
    - CME 106 or equivalent (for probability theory)

Logistics

- Lectures:
  - Tuesday and Thursday, 10:30am -11:50am (Zoom)
  - Friday 10:00am - 11:20am (Zoom)
  - Friday lectures are optional for those enrolled in AA 174A
  - Course will be taught in a “flipped classroom” format; pre-recorded lecture videos for each week will be released on the preceding Friday

- Sections
  - Monday, Wednesday, 10:30am - 12:30pm (Zoom)
  - Monday, 3:00pm – 5:00pm (Zoom)
  - Tuesday, 4:00pm – 6:00pm (Zoom)
  - Thursday, 2:00pm – 4:00pm (Zoom)

Logistics

- Office hours:
  - Prof. Pavone: Tuesday, 1:00 – 2:00pm (Zoom) and by appointment
  - CAs: Tuesday, 2:00 – 4:00pm; Thursday, 4:00 – 6:00pm (Zoom)
- Course websites:
  - For course content and announcements: http://asl.stanford.edu/aa274a/
  - For course-related questions: https://piazza.com/stanford/fall2020/aa274a
  - For homework submissions: https://www.gradescope.com/courses/175144
  - For lecture videos: https://canvas.stanford.edu/courses/123351
  - To submit pre-lecture questions: https://forms.gle/8snGsRR6eiYcqsjf7
- To contact the AA274 staff, use the email: aa274a-aut2021-staff@lists.stanford.edu

Grading

- Course grade calculation
  - (20%) final project
  - (60%) homework
  - (20%) sections
  - (extra 5%) participation on Piazza

Team

Instructor

Marco Pavone
Associate Professor AA, and CS/EE (by courtesy)

CAs

Somrita Banerjee

Abhishek Cauligi

Boris Ivanovic

Mengxi Li

Joseph Lorenzetti

Collaborators

- Benoit Landry
- Daniel Watzenig

Labs

ASL Autonomous Systems Lab

CARS Center for Automotive Research at Stanford

Schedule

Date	Topic	Assignment
09/15	Course overview, mobile robot kinematics	
09/17	Introduction to the Robot Operating System (ROS)	
09/18	*No lecture*	HW1 out
09/22	Trajectory optimization	
09/24	Trajectory tracking & closed loop control	
09/25	⋆ Advanced methods for trajectory optimization	
09/29	Motion planning I: graph search methods	HW1 due, HW2 out
10/01	Motion planning II: sampling-based methods	
10/02	*No lecture*	
10/06	Robotic sensors & introduction to computer vision	
10/08	Camera models & camera calibration	
10/09	⋆ Stereo vision	HW2 due, HW3 out
10/13	Image processing, feature detection & description	
10/15	Information extraction & classic visual recognition	
10/16	⋆ Modern robotic perception	
10/20	Intro to localization & filtering theory	HW3 due, HW4 out
10/22	Parameteric filtering (KF, EKF, UKF)	
10/23	⋆ Nonparameteric filtering (PF)	
10/27	EKF localization	Final project released
10/29	EKF SLAM	
10/30	⋆ Monte Carlo localization and particle filter SLAM	
11/03	Multi-sensor perception & sensor fusion	
11/05	Software for autonomous systems	
11/06	*No lecture*	
11/10	State machines	HW4 due
11/12	Decision making under uncertainty	
11/13	*No lecture*	Final project check-in
11/17	Reinforcement learning	
11/19	Conclusions	
11/20	**Final Project Demo**	

Mobile robot kinematics

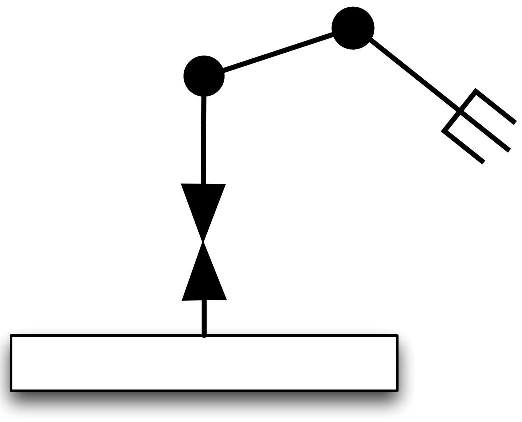
- Aim
  - Understand motion constraints
  - Learn about basic motion models for wheeled vehicles
  - Gain insights for motion control

- Readings
  - R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to Autonomous Mobile Robots. MIT Press, 2nd Edition, 2011. Sections 3.1-3.3.
  - B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo. Robotics: Modelling, Planning, and Control. Springer, 2008 (chapter 11).

Holonomic constraints

- Let $\xi = [\xi_1, \dots, \xi_n]^T$ denote the configuration of a robot (e.g., $\xi = [x, y, \theta]^T$ for a wheeled mobile robot)

- *Holonomic* constraints
  - $h_i(\xi) = 0$, for $i = 1, \dots, k < n$
  - Reduce space of accessible configurations to an $n - k$ dimensional subset
  - If all constraints are holonomic, the mechanical system is called holonomic
  - Generally the result of mechanical interconnections



Kinematic constraints

- Kinematic constraints

$$a_i(\xi, \dot{\xi}) = 0, \qquad i = 1, \ldots, k < n$$

  - constrain the instantaneous admissible motion of the mechanical system
  - generally expressed in Pfaffian form, i.e., linear in the generalized velocities

$$a_i^T(\xi)\,\dot{\xi} = 0, \qquad i = 1, \ldots, k < n$$

- Clearly, k holonomic constraints imply the existence of an equal number of kinematic constraints

$$\frac{d\,h_i(\xi)}{dt} = \frac{\partial\,h_i(\xi)}{\partial \xi}\dot{\xi} = 0, \qquad i = 1, \ldots, k < n$$

- However, the converse is not true in general…

Nonholonomic constraints

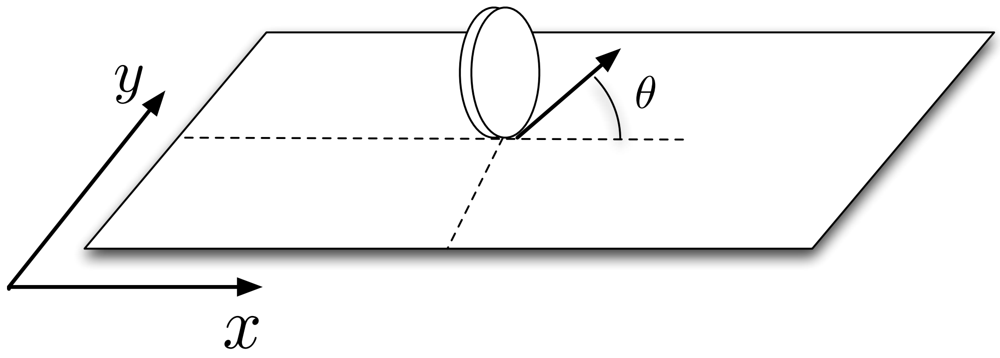
- If a kinematic constraint is not integrable in the form $h_i(\xi) = 0$, then it is said *nonholonomic* -> nonholonomic mechanical system
- Nonholonomic constraints reduce mobility in a completely different way. Consider a single Pfaffian constraint

$$a^T(\xi)\,\dot{\xi} = 0$$

- Holonomic
  - Can be integrated to $h(\xi) = 0$
  - Loss of accessibility, motion constrained to a level surface of dimension $n - 1$
- Nonholonomic
  - *Velocities* constrained to belong to a subspace of dimension $n - 1$, the null space of $a^T(\xi)$
  - No loss of accessibility

Example of nonholonomic system

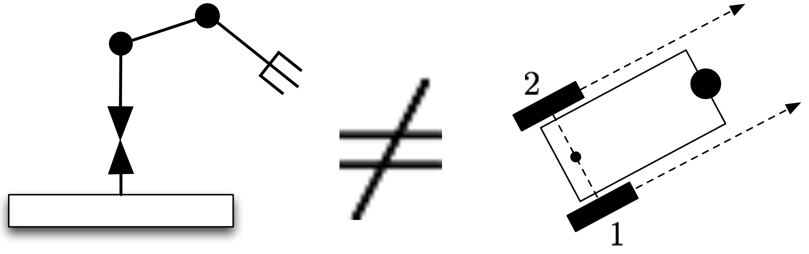
- System: disk that rolls without slipping
- $\xi = [x, y, \theta]^T$



- No side slip constraint

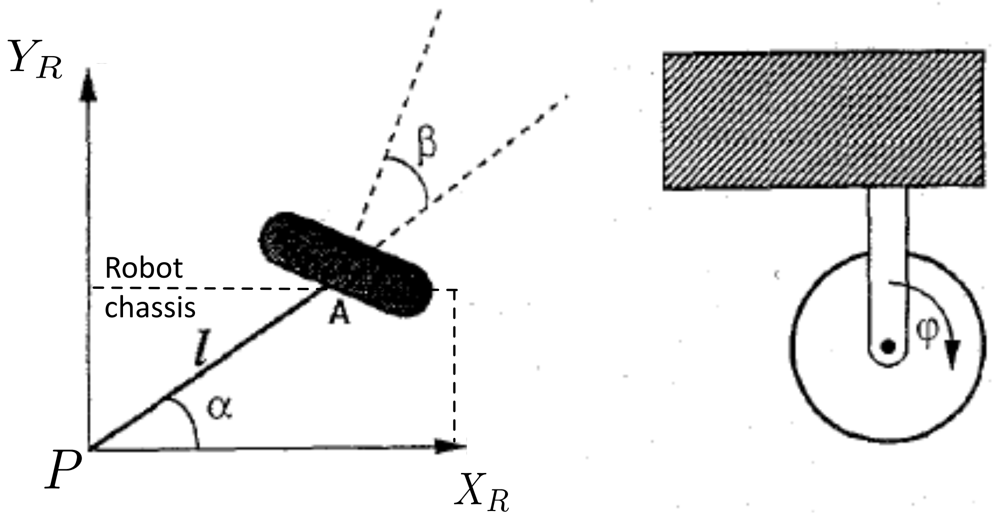
$$[\dot{x},\ \dot{y}] \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} = \dot{x}\sin\theta - \dot{y}\cos\theta = [\sin\theta, -\cos\theta, 0]\,\dot{\xi} = 0$$

- Facts:
  - No loss of accessibility
  - Wheeled vehicles are generally nonholonomic

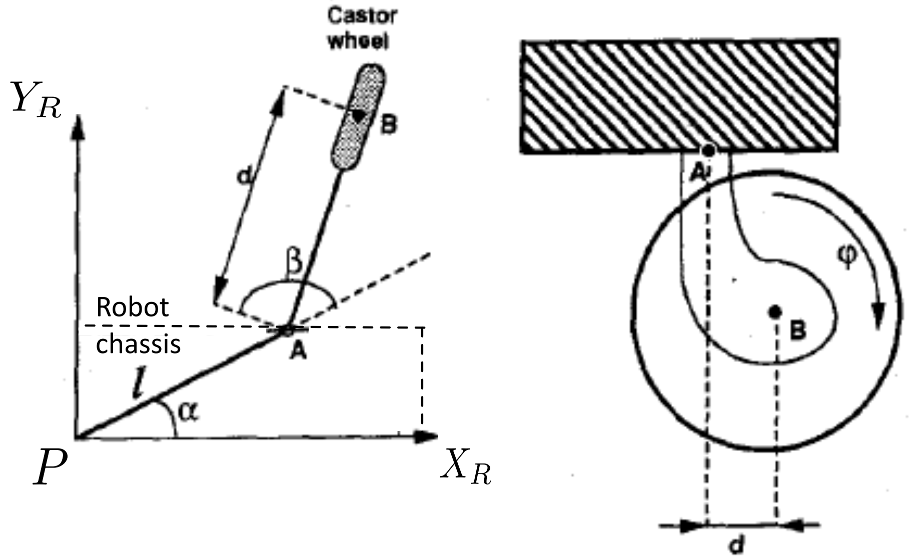


Types of wheels

- Standard wheels (four types)



Standard wheel -- fixed or steerable



Standard, off-centered wheel (caster)
-- passive or active

- Special wheels: achieve omnidirectional motion (e.g., Swedish or spherical wheels)

Kinematic models

- Assume the motion of a system is subject to *k* Pfaffian constraints

$$\begin{bmatrix} a_1^T(\xi) \\ \vdots \\ a_k^T(\xi) \end{bmatrix} \dot{\xi} := A^T(\xi)\dot{\xi} = 0$$

- Then, the admissible velocities at each configuration ξ belong to the *(n − k)*-dimensional null space of matrix $A^T(\xi)$
- Denoting by $\{g_1(\xi), \ldots, g_{n-k}(\xi)\}$ a basis of the null space of $A^T(\xi)$, admissible trajectories can be characterized as solutions to

$$\dot{\xi} = \sum_{j=1}^{n-k} g_j(\xi) u_j = G(\xi) u$$

Input vector

Example: unicycle

- Consider pure rolling constraint for the wheel:

$$\dot{x}\sin\theta - \dot{y}\cos\theta = [\sin\theta,\ -\cos\theta,\ 0]\,\dot{\xi} = a^T(\xi)\,\dot{\xi} = 0$$

- Consider the matrix

$$G(\xi) = [g_1(\xi),\ g_2(\xi)] = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}$$

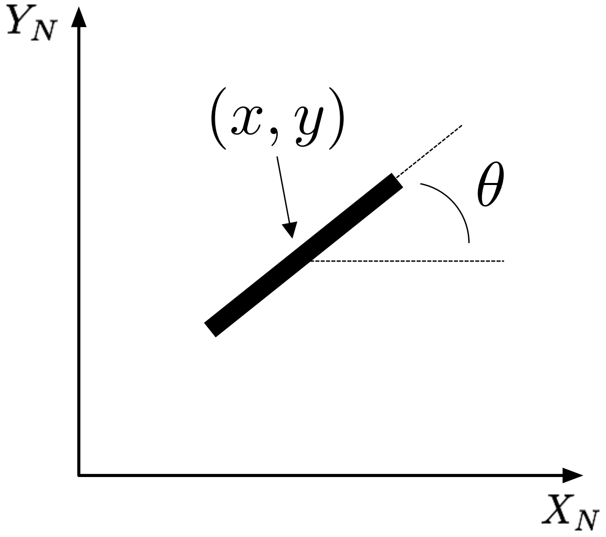
where $[g_1(\xi), g_2(\xi)]$ is a basis of the null space of $a^T(\xi)$

- All admissible velocities are therefore obtained as linear combination of $g_1(\xi)$ and $g_2(\xi)$

Unicycle and differential drive models

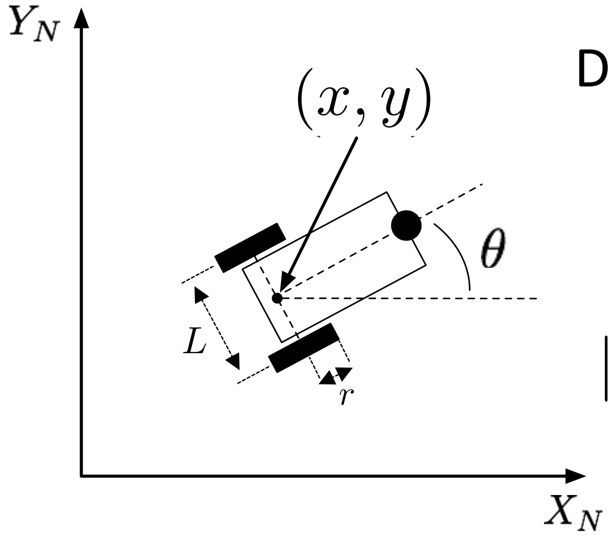
$$\begin{pmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{pmatrix} = \begin{pmatrix}\cos\theta\\ \sin\theta\\ 0\end{pmatrix} v + \begin{pmatrix}0\\ 0\\ 1\end{pmatrix}\omega$$

$$\begin{pmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{pmatrix} = \begin{pmatrix}\frac{r}{2}(\omega_l + \omega_r)\cos\theta\\ \frac{r}{2}(\omega_l + \omega_r)\sin\theta\\ \frac{r}{L}(\omega_r - \omega_l)\end{pmatrix}$$



Unicycle

$$|v| \leq v_{\max}$$
$$|\omega| \leq \omega_{\max}$$



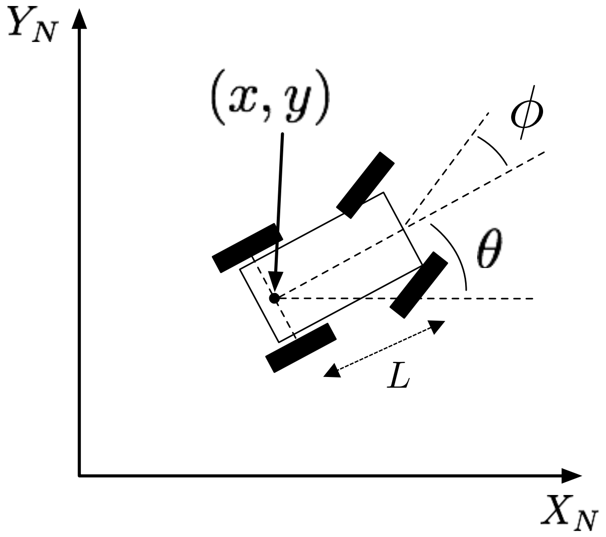
Differential drive

$$|\omega_l| \leq \omega_{l,\max}$$
$$|\omega_r| \leq \omega_{r,\max}$$

The kinematic model of the unicycle also applies to the differential drive vehicle, via the one-to-one input mappings: $v = \frac{r}{2}(\omega_r + \omega_l)$ $\omega = \frac{r}{L}(\omega_r - \omega_l)$

Simplified car model

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} v\cos\theta \\ v\sin\theta \\ \frac{v}{L}\tan\phi \end{pmatrix}$$



Simple car

$$|v| \leq v_{\max}, \; |\phi| \leq \phi_{\max} < \frac{\pi}{2}$$ → Simple car model

$$v \in \{-v_{\max}, v_{\max}\}, \; |\phi| \leq \phi_{\max} < \frac{\pi}{2}$$ → Reeds&Shepp's car

$$v = v_{\max}, \; |\phi| \leq \phi_{\max} < \frac{\pi}{2}$$ → Dubins' car

References: (1) J.-P. Laumond. Robot Motion Planning and Control. 1998. (2) S. LaValle. Planning algorithms, 2006.

From kinematic to dynamic models

- A kinematic state space model should be interpreted only as a subsystem of a more general dynamical model
- Improvements to the previous kinematic models can be made by placing integrators in front of action variables
- For example, for the unicycle model, one can set the speed as the integration of an action a representing acceleration, that is

$$\dot{x} = v \cos\theta, \quad \dot{y} = v \sin\theta, \quad \dot{\theta} = \omega, \quad \dot{v} = a$$

Next time

ROS