Principles of Robot Autonomy I

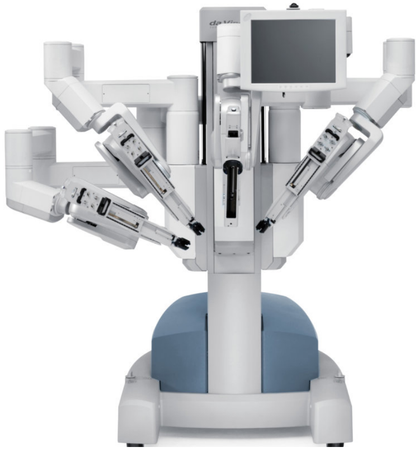
Course overview, mobile robot kinematics

Stanford University

Autonomous Systems Lab

From automation…

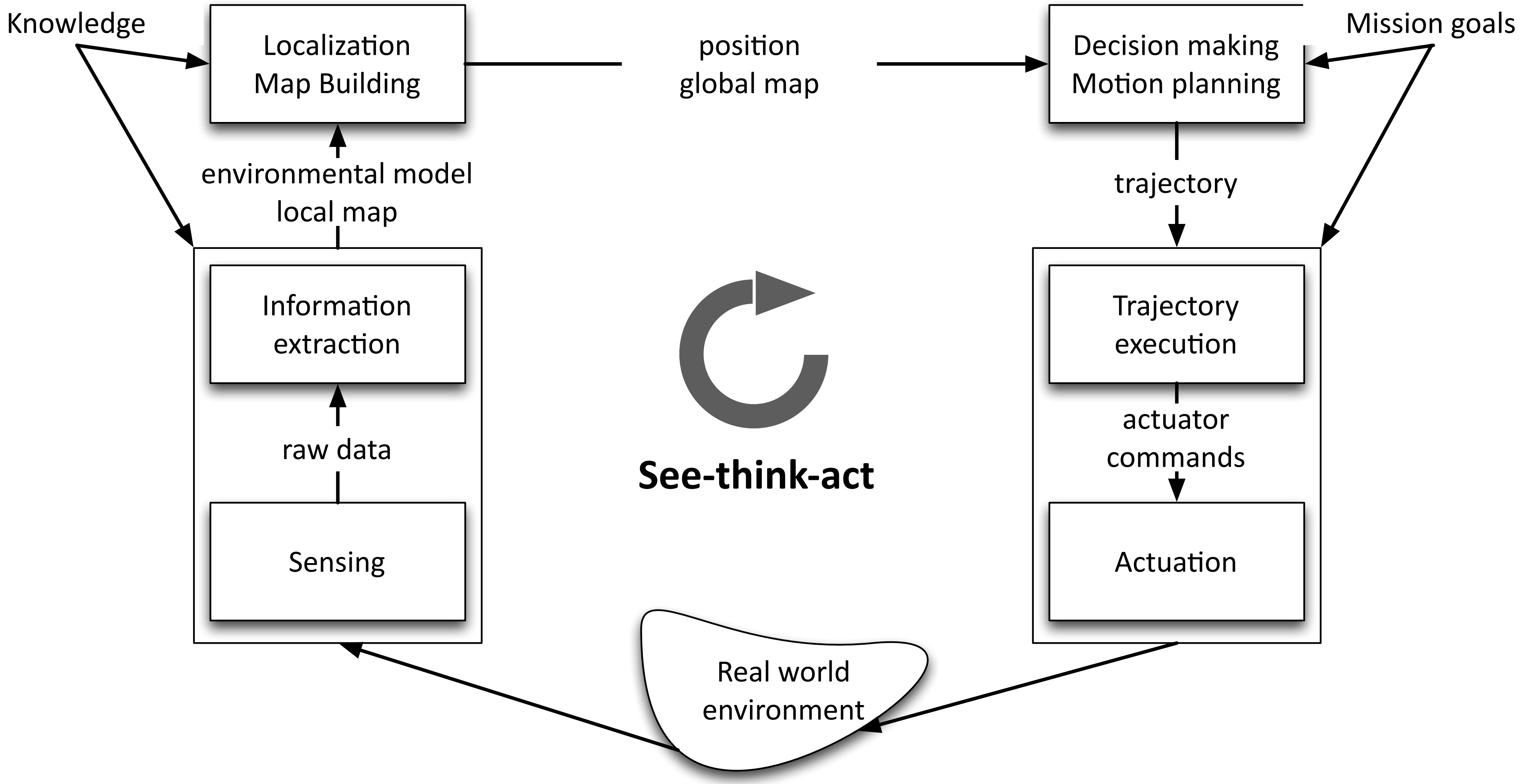
…to autonomy



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 and system architecture

- Extensive use of the Robot Operating System (ROS)

- Requirements
  - CS 106A or equivalent
  - CME 100 or equivalent (for calculus, linear algebra)
  - CME 106 or equivalent (for probability theory)
  - See also the pre-knowledge quiz on the course website

Logistics

- Lectures:
  - Tuesdays and Thursdays, 9:45am – 11:15am (NVIDIA Auditorium)
  - Recordings will be made available to all students on Canvas.

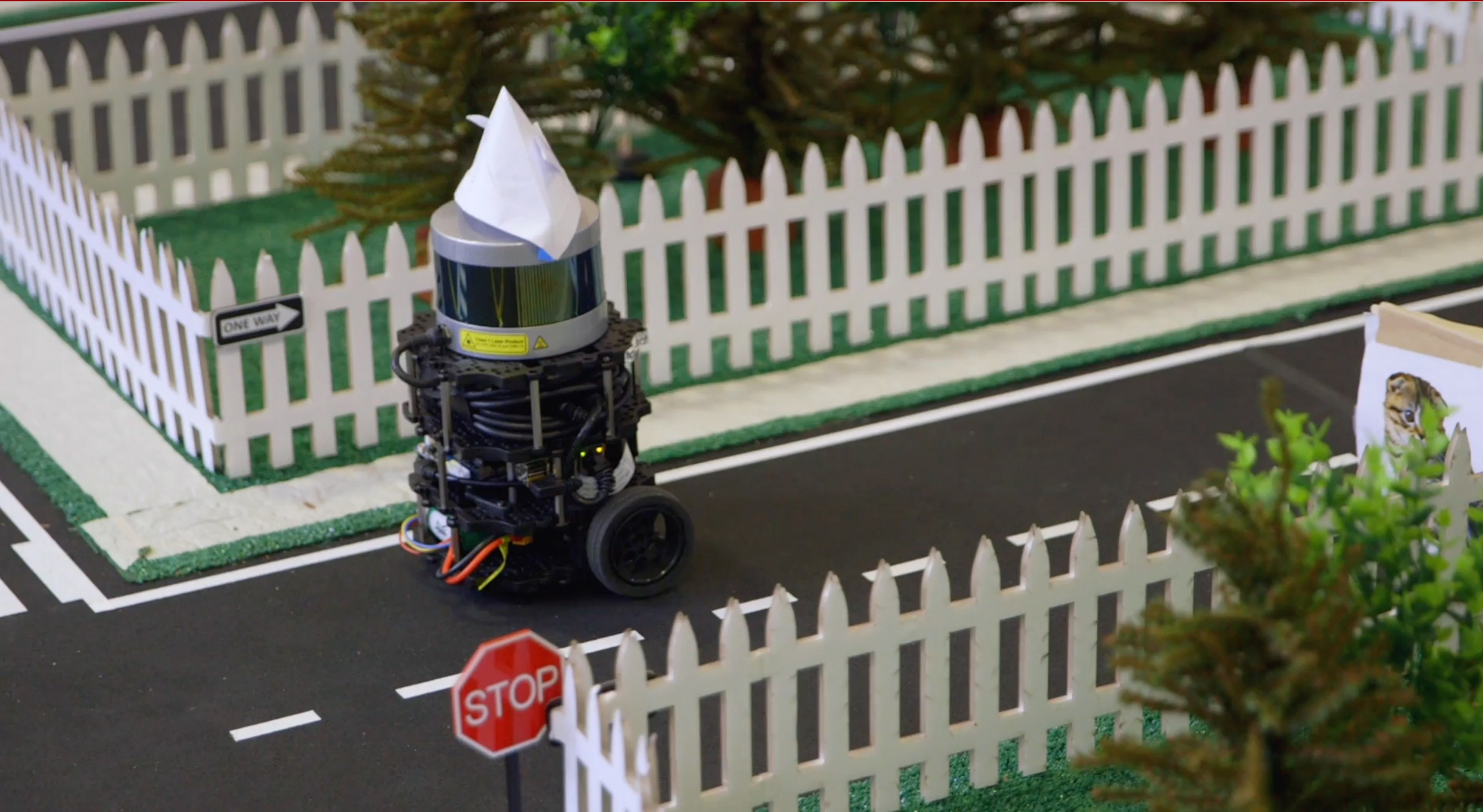
- Sections
  - 2-hour, once-a-week sessions starting Week 2
  - Hands-on exercises that complement the lecture material, build familiarity with ROS, develop skills necessary for the final project

    Monday: 5:30 – 7:30pm (virtual) `rabrown1`
    Tuesday: 4:30 – 6:30pm (in-person) `lewt`
    Wednesday: 10am – 12pm (virtual) `somrita`
    Wednesday: 12pm – 2pm (in-person) `schneids`
    Wednesday: 5 – 7pm (in-person) `rabrown1`
    Thursday: 11:45am – 1:45pm (in-person) `somrita`
    Thursday: 4:30 – 6:30pm (virtual) `lewt`
    Friday: 12 – 2pm (virtual) `schneids`

  - Link to the section sign-up sheet

Logistics

- Office hours:
  - Dr. Schmerling: Thursday, 12:45 – 1:45pm (Durand 217) and by appointment
  - CAs: Monday, 1:00 – 3:00pm (Skilling Lab); Tuesday 11:30am – 1:30pm (Zoom); Friday, 10:00am – 12:00pm (Zoom)
- Course websites:
  - For course content and announcements: http://asl.stanford.edu/aa274a/
  - For course-related questions: https://edstem.org/us/courses/14340
  - For homework submissions: https://www.gradescope.com/courses/309846
  - For lecture videos: https://canvas.stanford.edu/courses/142088
- To contact the AA274 staff, use the email: aa274a-aut2122-staff@lists.stanford.edu


ONE WAY
STOP

Team

Instructor

Ed Schmerling
Research Engineer
Associate Director, Stanford Autonomous Systems Lab

Collaborators

- Daniel Watzenig

Labs

Autonomous Systems Lab

Center for Automotive Research at Stanford

CAs

Somrita Banerjee

Robin Brown

Thomas Lew

Stephanie Schneider

Schedule

Date	Topic	Assignment
09/21	Course overview, mobile robot kinematics	
09/23	Introduction to the Robot Operating System (ROS)	HW1 out
09/28	Trajectory optimization	
09/30	Trajectory tracking & closed loop control	
10/05	Motion planning I: graph search methods	HW1 due, HW2 out
10/07	Motion planning II: sampling-based methods	
10/12	Robotic sensors & introduction to computer vision	
10/14	Camera models & camera calibration	
10/19	Image processing, feature detection & description	HW2 due, HW3 out
10/21	Information extraction & classic visual recognition	
10/26	Intro to localization & filtering theory	
10/28	Parametric filtering (KF, EKF, UKF)	
11/02	*No lecture (Democracy Day)*	HW3 due, HW4 out
11/04	Nonparametric filtering (PF)	Final project released
11/09	Object detection/tracking, EKF localization	
11/11	Simultaneous localization and mapping (SLAM)	
11/16	Multi-sensor perception & sensor fusion I	HW4 due
11/18	Multi-sensor perception & sensor fusion II	
11/23	*No lecture (Thanksgiving)*	
11/25	*No lecture (Thanksgiving)*	
11/30	Stereo vision	Final project check-in
12/02	State machines	
12/08	*Final exam slot (8:30 – 11:30am)*	Final project demo

Mobile robot kinematics

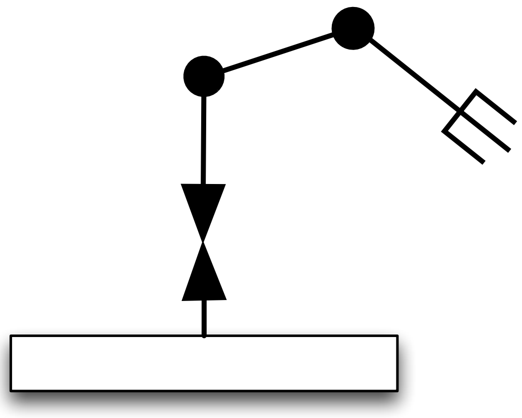
- Aim
  - Understand motion constraints
  - Learn about basic motion models for wheeled vehicles
  - Gain insights for motion control

- 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 $\xi = [\xi_1, \ldots, \xi_n]^T$ 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 $i = 1, \ldots, k < n$
  - 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

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

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

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

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

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

- However, the converse is not true in general…

Nonholonomic constraints

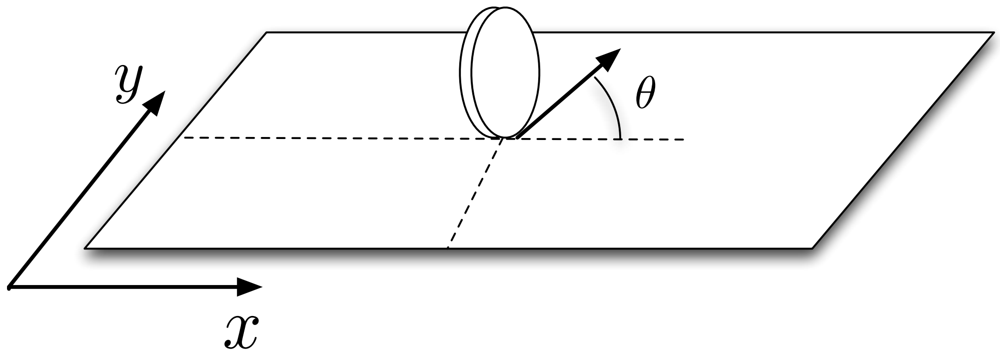
- If a kinematic constraint is not integrable in the form $h_i(\xi) = 0$, then it is said *nonholonomic* -> nonholonomic mechanical system
- Nonholonomic constraints reduce mobility in a completely different way. Consider a single Pfaffian constraint

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

- 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

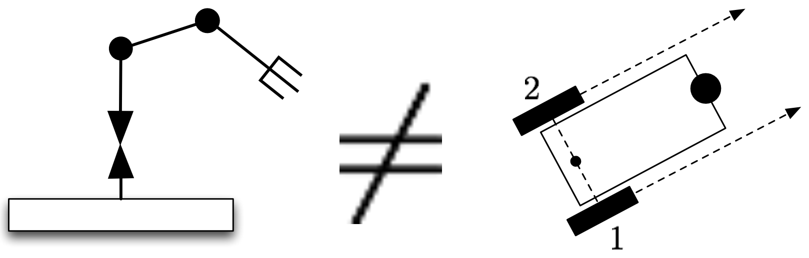
- System: disk that rolls without slipping
- $\xi = [x, y, \theta]^T$



- No side slip constraint

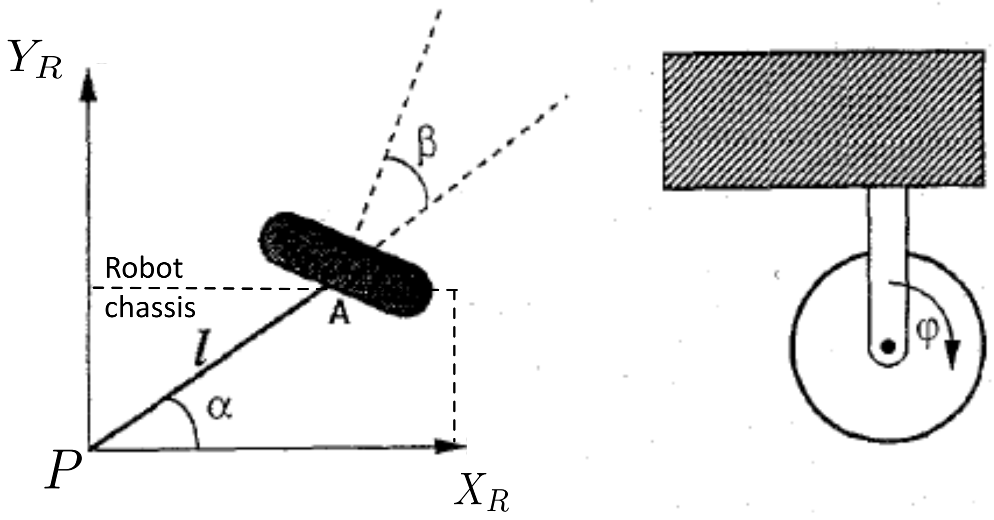
$$[\dot{x},\ \dot{y}] \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} = \dot{x}\sin\theta - \dot{y}\cos\theta = [\sin\theta, -\cos\theta, 0]\,\dot{\xi} = 0$$

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

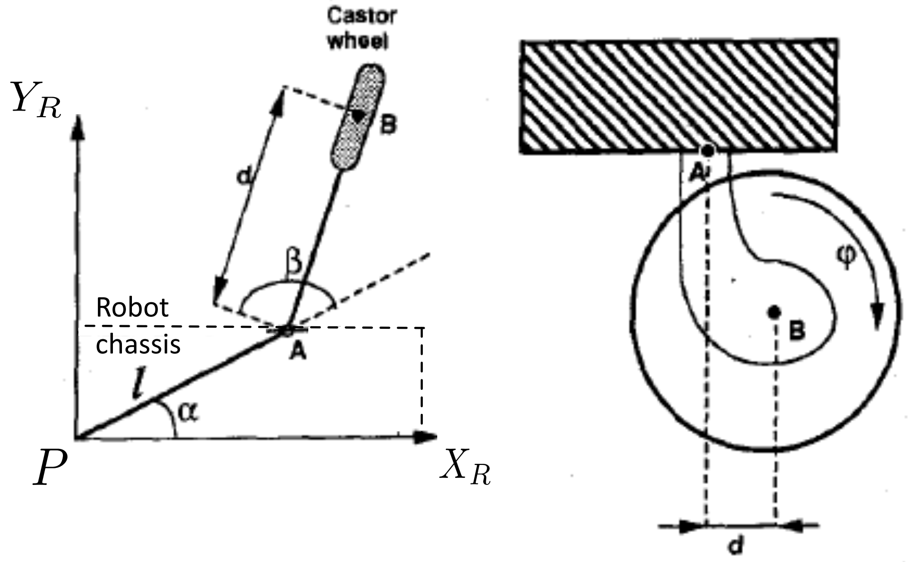


Types of wheels

- 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 $\{g_1(\xi), \dots, g_{n-k}(\xi)\}$ 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

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

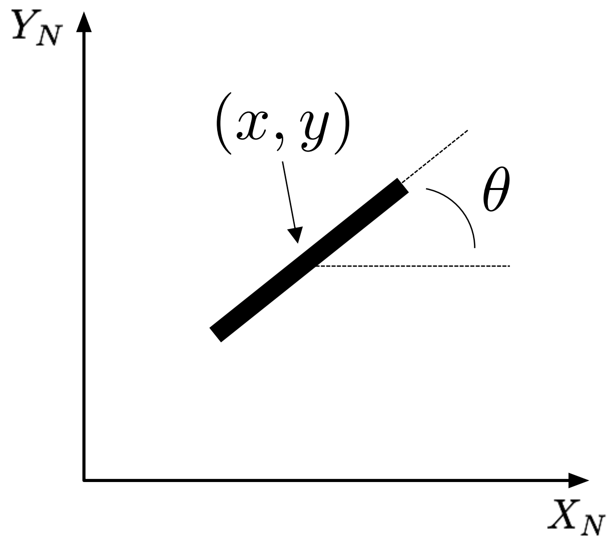
where $[g_1(\xi), g_2(\xi)]$ is a basis of the null space of $a^T(\xi)$

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

Unicycle and differential drive models

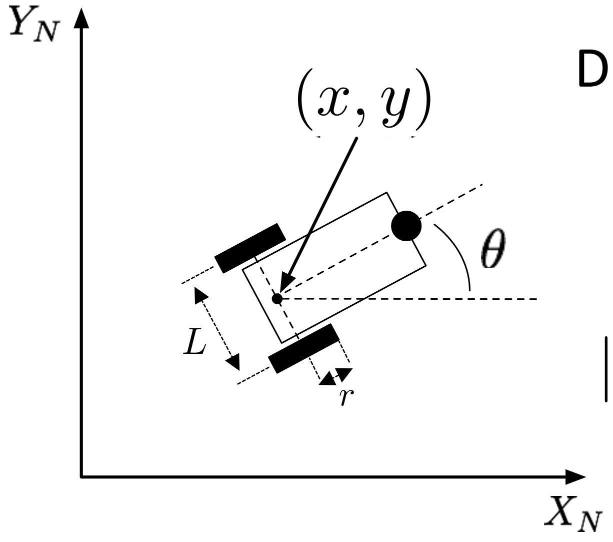
$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega$$

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



Unicycle

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



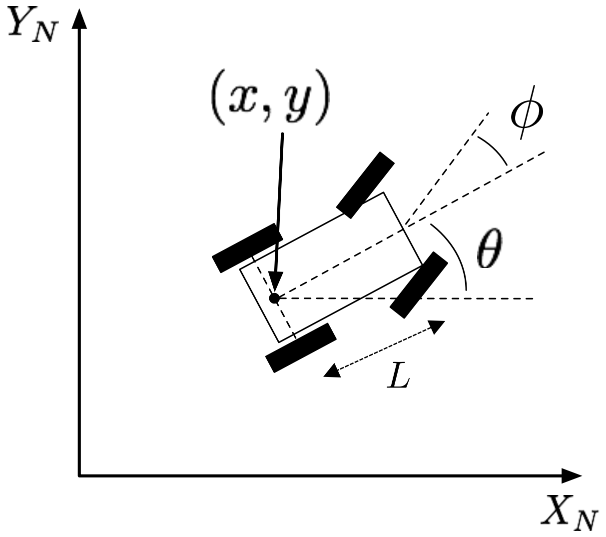
Differential drive

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

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

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