AA 274
Principles of Robotic Autonomy

Course overview, mobile robot kinematics,

introduction to motion control

Stanford University

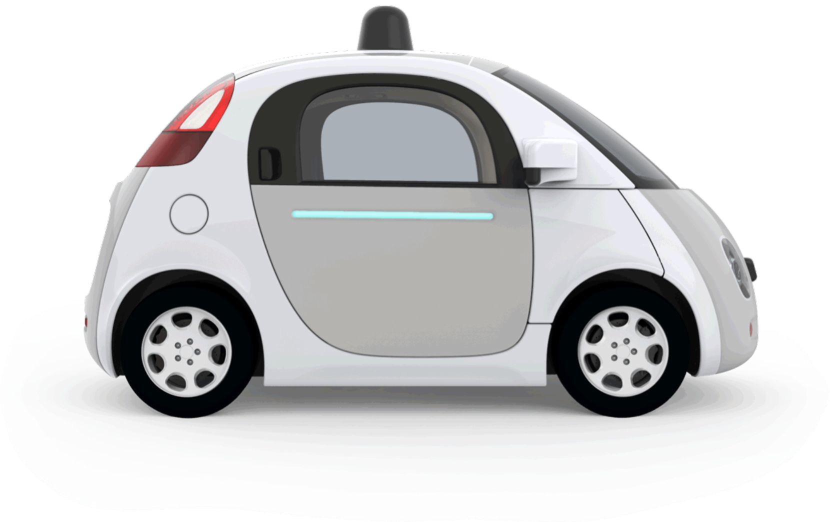
Autonomous Systems Lab

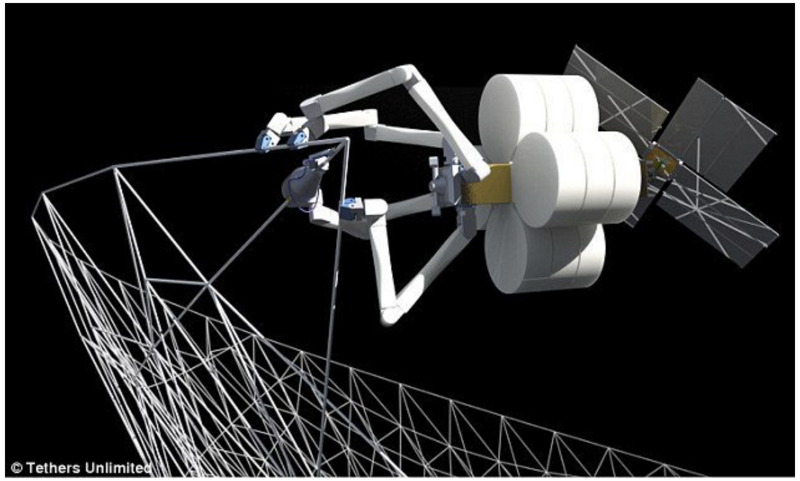
Course goals

- 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

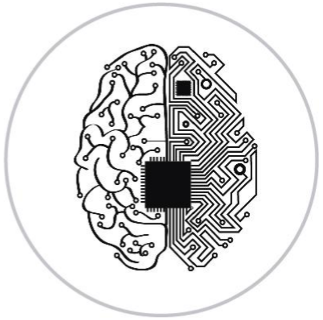
From automation…

...to autonomy





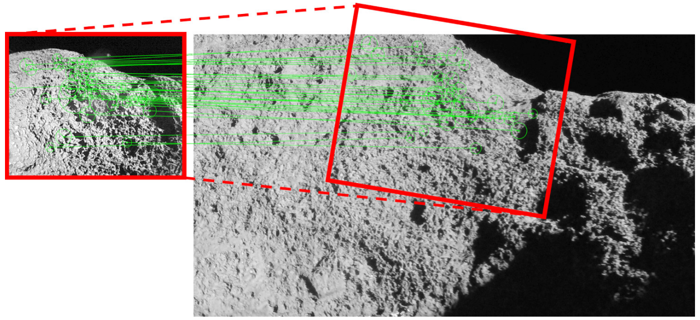
ASL research



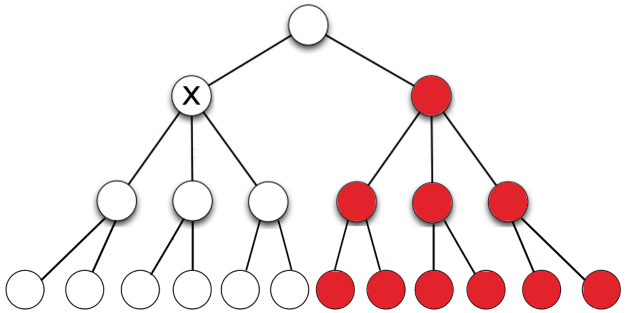
Trustworthy autonomy

Agile motion planning

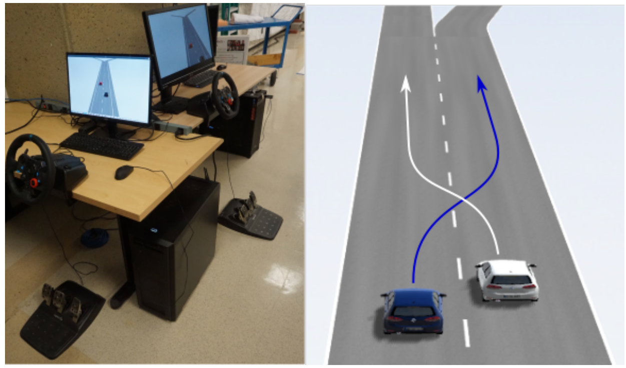
System-level coordination



Interplay with perception



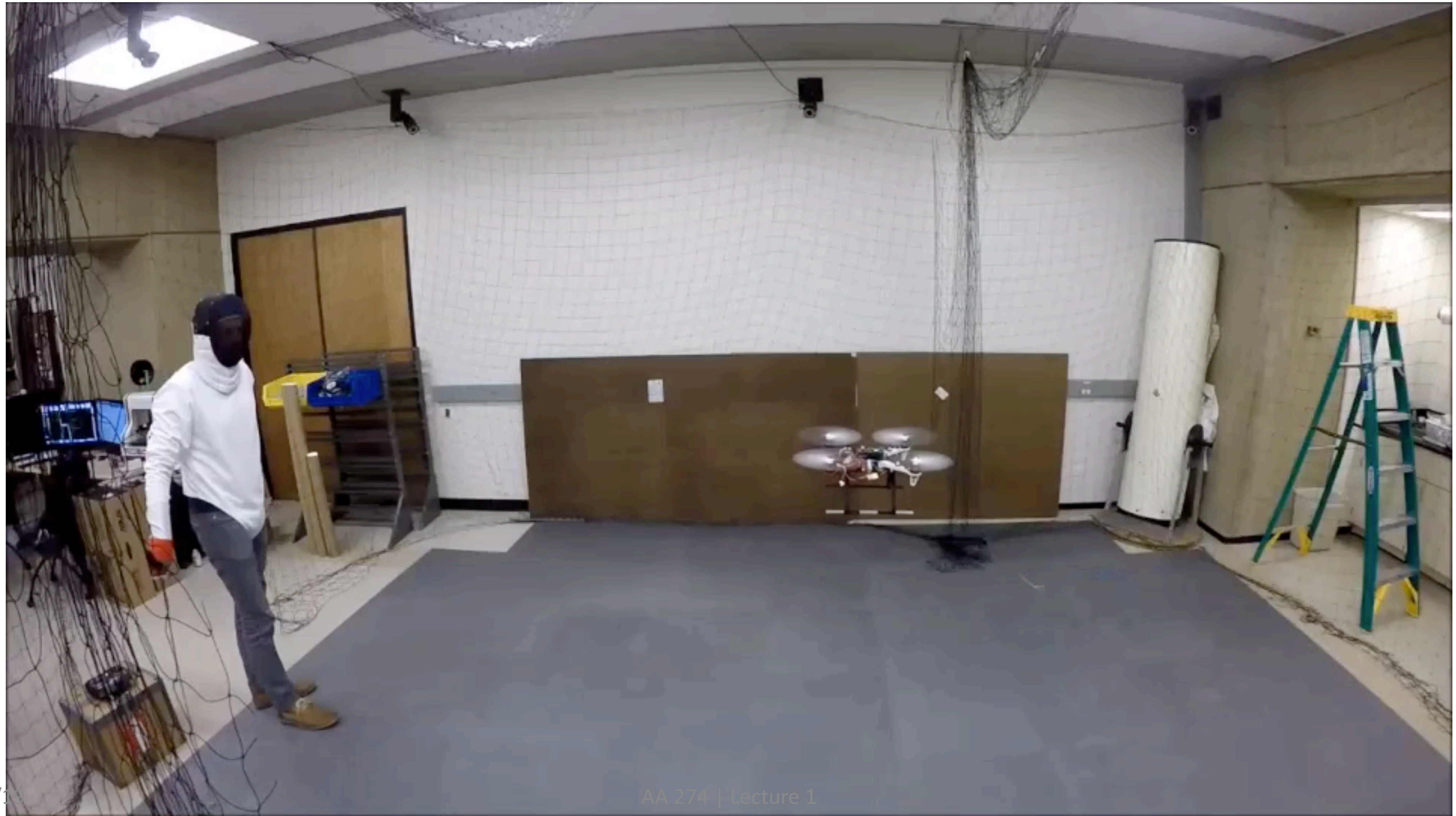
Safe and robust decision making and learning



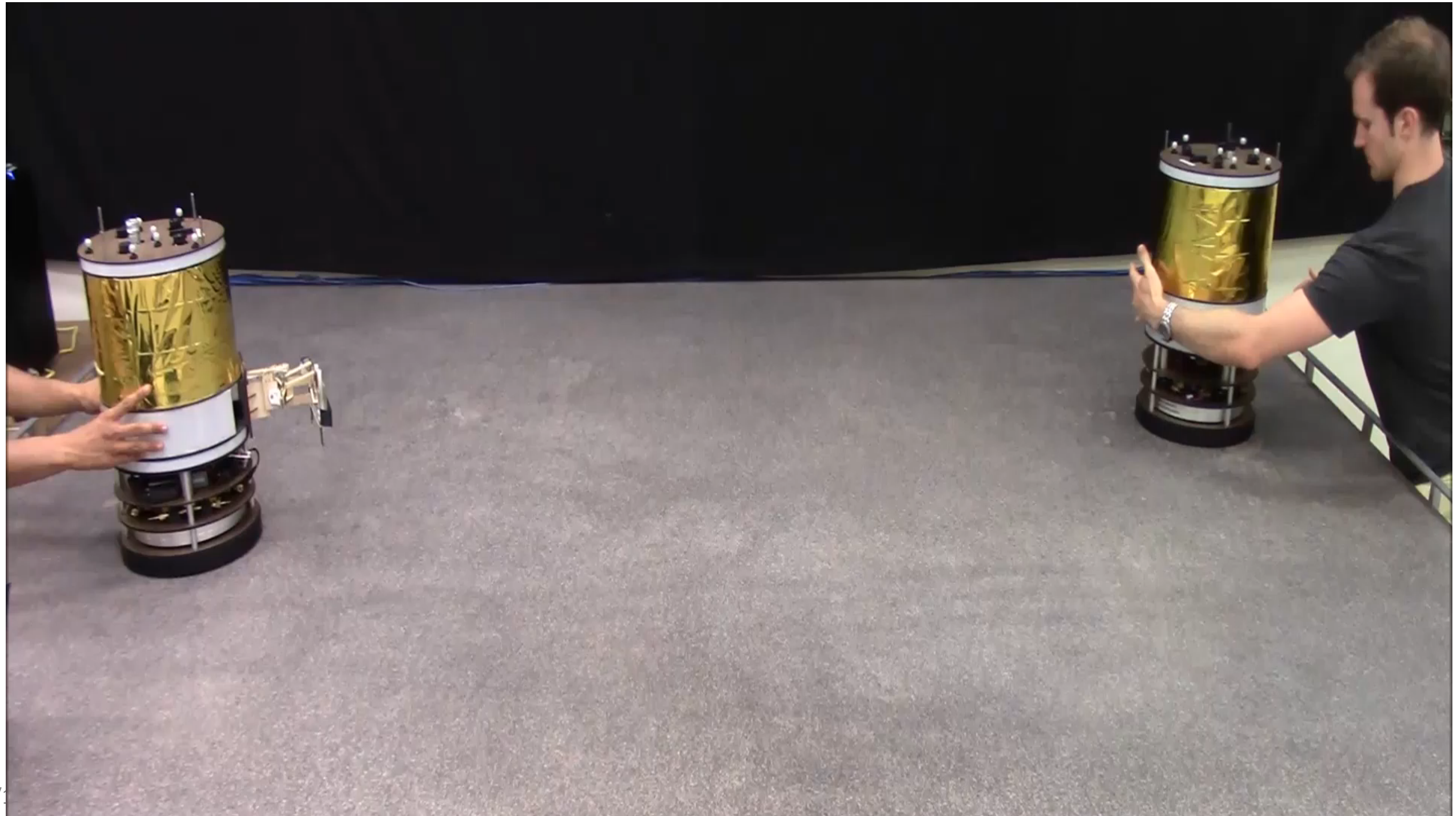
Safe interactions with humans

Sample of ASL research

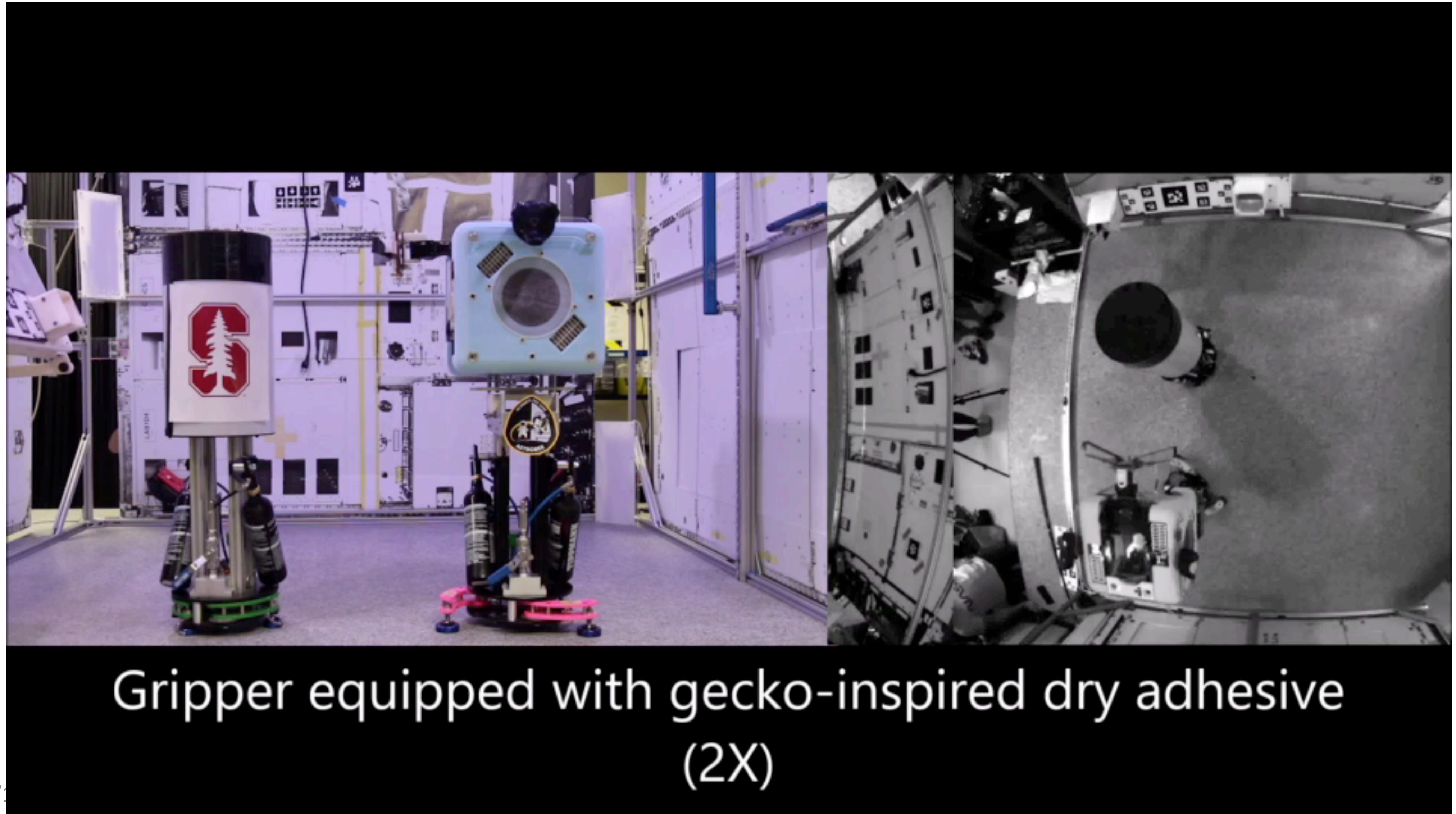
Sample of ASL research



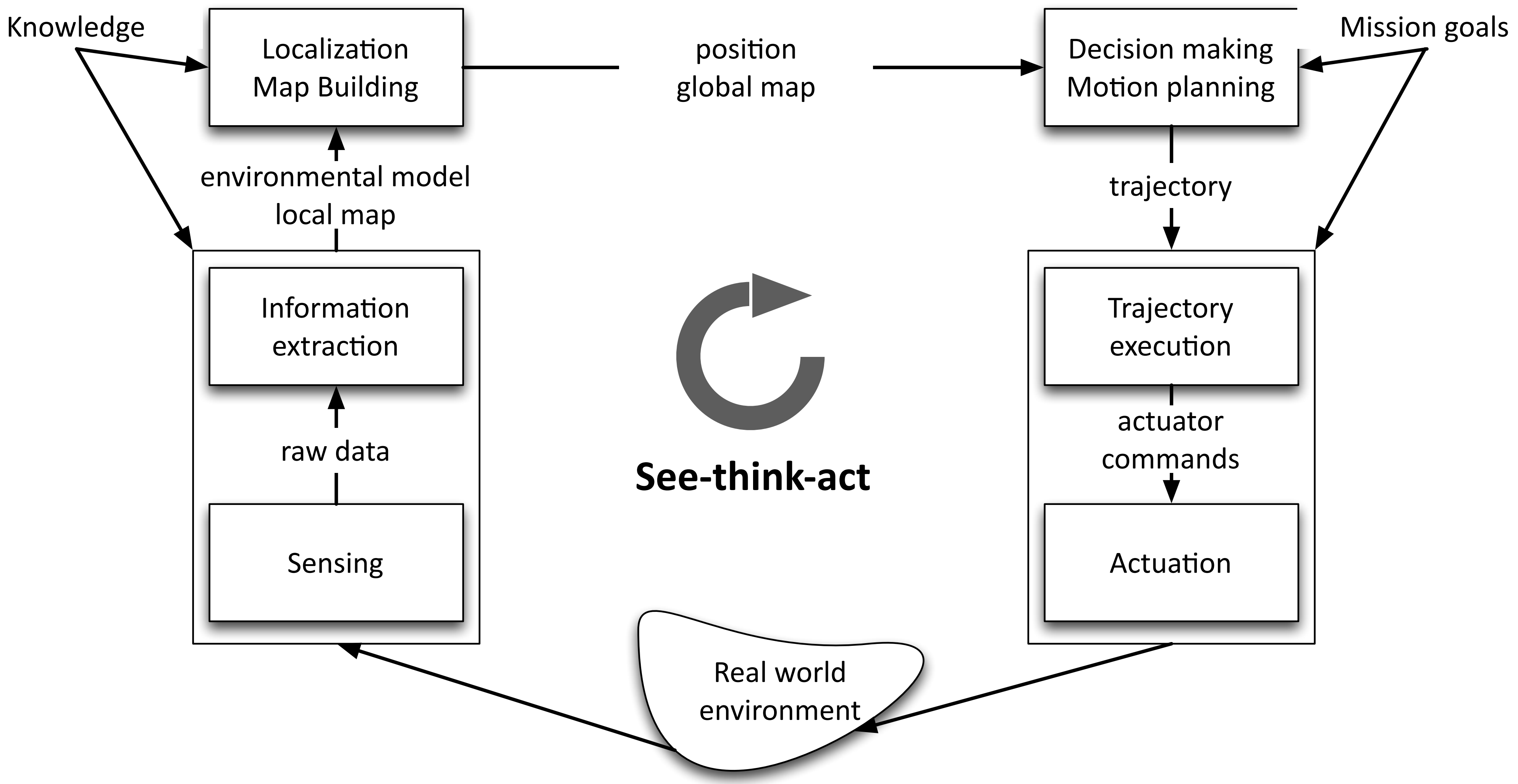
Sample of ASL research



Sample of ASL research

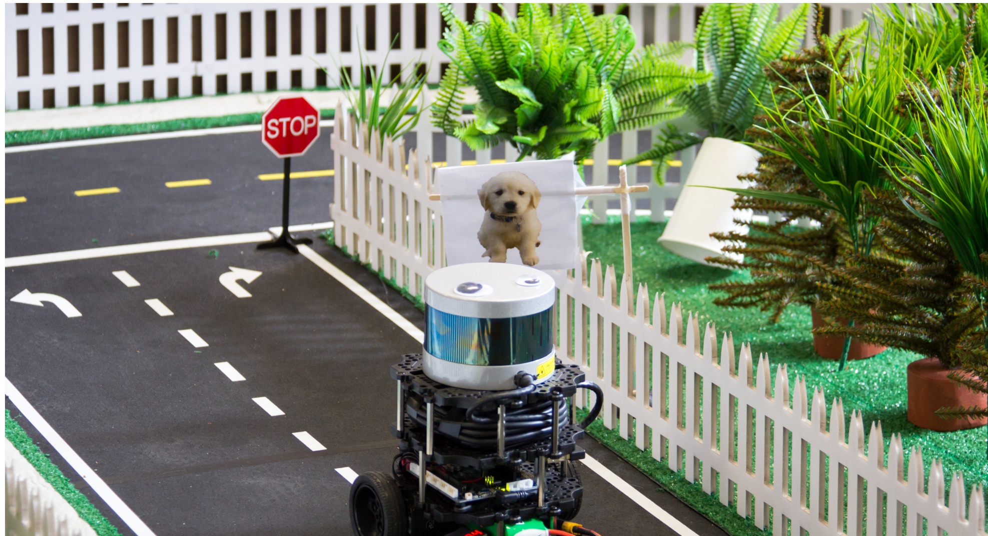


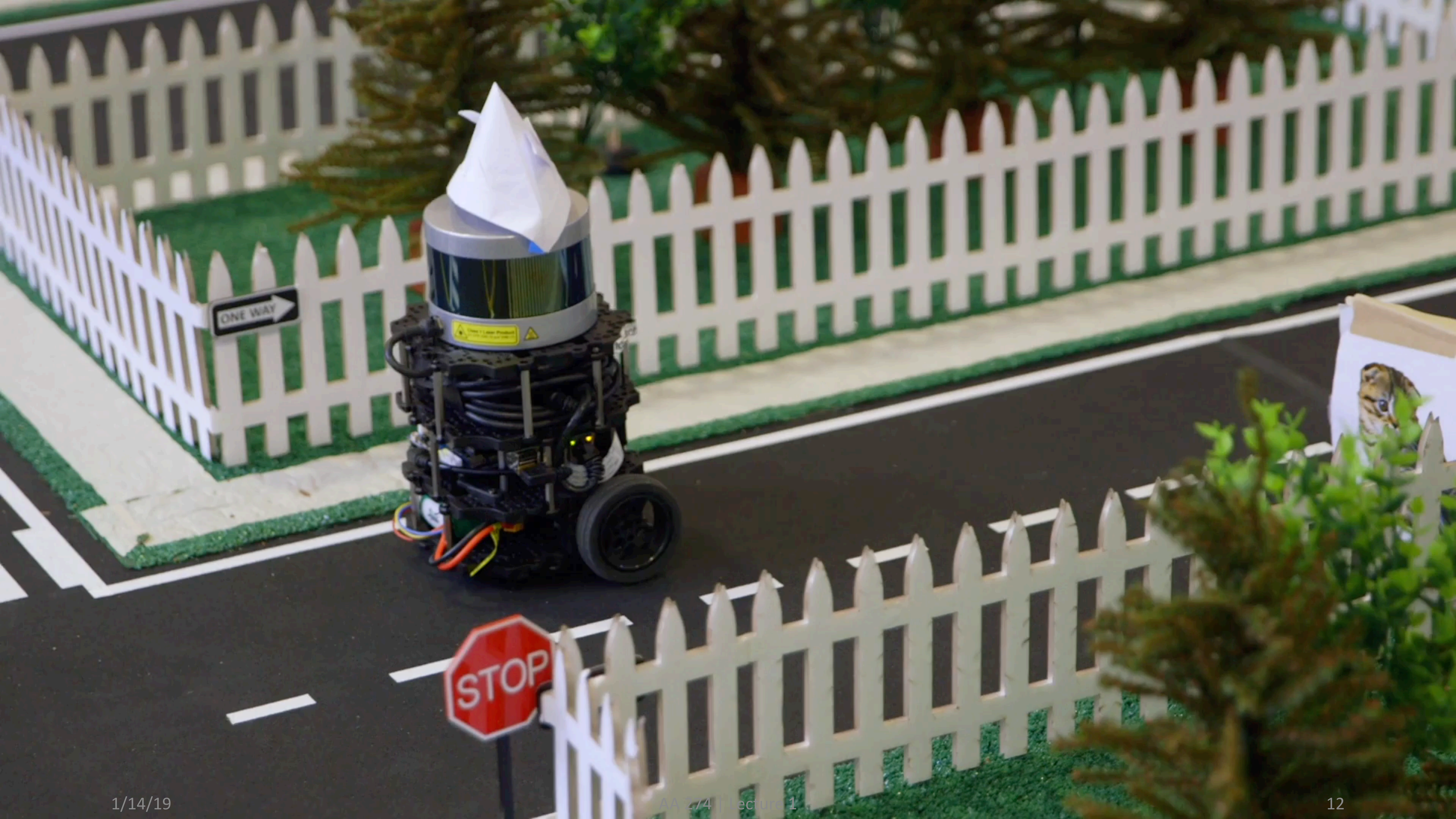
The see-think-act cycle



Course structure

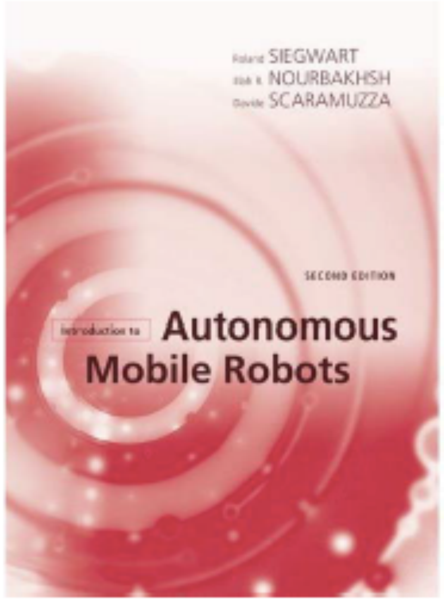
- Four modules, roughly of equal length
    1. motion control
    2. perception, from classic to deep learning approaches
    3. localization and SLAM
    4. planning, decision making, and system architecting
- Extensive use of the Robot Operating System (ROS)
    - Requirement: familiarity with programming (e.g., CS 106A or equivalent)
- Course grade calculation
    - 72% homework (four problem sets)
    - 23% final project
    - 5% scribe quality
    - Extra 5%: participation on Piazza




ONE WAY
STOP

Recommended textbook

- R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza. Introduction to Autonomous Mobile Robots. MIT Press, 2nd Edition, 2011.



- Additional reading material:
  - S. Thrun, W. Burgard, D. Fox. Probabilistic Robotics. MIT Press, 2005.
  - P. Goebel. ROS By Example. 2013.

Schedule

Date	Topic	Assignment	Readings
01/07	Course overview, mobile robot kinematics, introduction to motion control		SNS:3.1-3:5;
01/09	The Robot Operating System (ROS)	HW1 out	Lecture notes
01/11	*Recitation*: Python (optional)		
01/14	Open-loop and closed-loop motion control		SNS:3.6
01/16	Robotic sensors and introduction to computer vision		SNS:4.1-4.2
01/18	*Recitation*: dynamical systems (optional)		
01/21	Martin Luther King, Jr., Day (no classes)		
01/23	Camera models and camera calibration	HW2 out, HW1 due	SNS:4.2
01/25	*Recitation*: advanced Python (optional)		
01/28	Stereo vision and image processing		SNS:4.3-4.5
01/30	Feature detection & description, information extraction, and "classic" visual recognition		SNS:4.7
02/05	Machine learning for robot autonomy		Lecture notes
02/06	Deep learning for visual recognition		Lecture notes
02/11	Localization I	HW3 out, HW2 due	SNS:5:1-5:4
02/13	Localization II		SNS:5.5-5.6
02/18	Presidents' Day (no classes)		
02/20	Localization III		SNS:5.6-5.7
02/25	SLAM I		SNS:5.8
02/27	SLAM II		SNS:5.8
03/04	Motion planning I: combinatorial motion planning	HW4 out, HW3 due	SNS:6.1-6.5
03/06	Motion planning II: sampling-based motion planning		Lecture notes
03/11	Decision making and reinforcement learning		Lecture notes
03/13	State machines and "architecting" the autonomy stack	HW4 due	Lecture notes
TBD	**Final project**, TBD		

Team

Instructor

Marco Pavone

Collaborators:

- Andrew Bylard
- Patrick Goebel
- Boris Ivanovic
- Benoit Landry
- Joseph Steven Lorenzetti

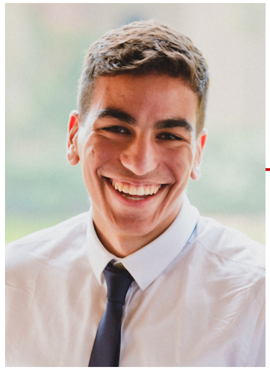
Labs:

Autonomous Systems Lab

Center for Automotive Research at Stanford

CAs

Christopher Covert



Amine Elhafsi

Karen Leung

Apoorva Sharma

Logistics

- Location / time: 420-041, Mondays and Wednesdays, 1:30 -- 2:50pm
- Office hours:
    - Prof. Pavone: Mondays, 3:00 -- 5:00pm, after class, and by appointment
    - CAs (homework support): Tuesdays and Fridays, 2:00 -- 4:00pm, location TBD
    - CAs (ROS support): Mondays, Wednesdays, and Thursdays, 5:00 -- 7:00pm, location TBD
- Course website:
    - http://asl.stanford.edu/aa274/
    - https://piazza.com/stanford/winter2019/aa274
    - https://www.gradescope.com/courses/35120
- To contact the AA274 staff, use the email: aa274-win1819-staff@lists.stanford.edu

Mobile robot kinematics

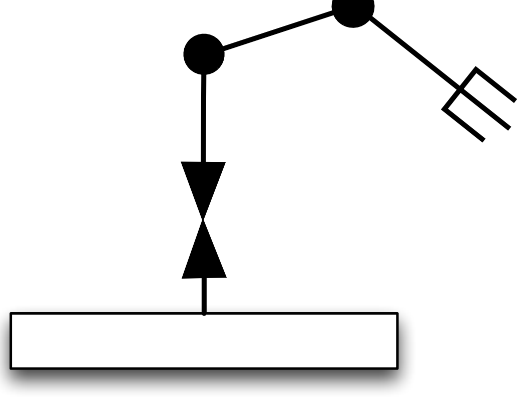
- Aim
  - Understand motion constraints
  - Learn about basic motion models for wheeled vehicles
  - Gain insights for motion control

- Readings
  - SNS: 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

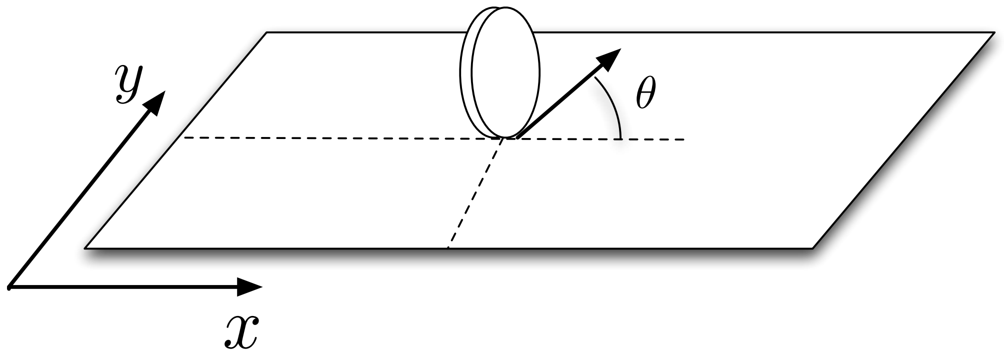
- If a kinematic constraint is not integrable in the form $h_i(\xi) = 0$, then it is said *nonholonomic* -> nonholonomic mechanical system
- Nonholonomic constraints reduce mobility in a completely different way. Consider a single Pfaffian constraint

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

- 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 $a^T(\xi) = 0$
  - No loss of accessibility

Example of nonholonomic system

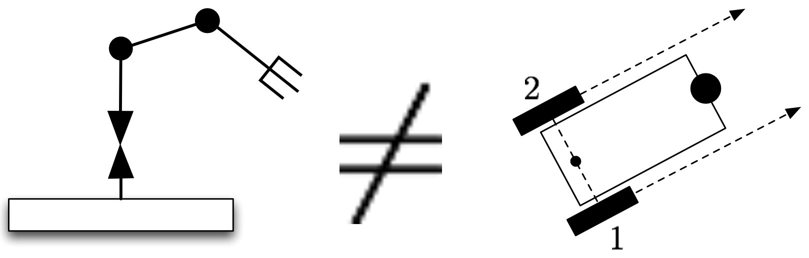
- System: disk that rolls without slipping
- $\xi = [x, y, \theta]^T$



- No side slip constraint

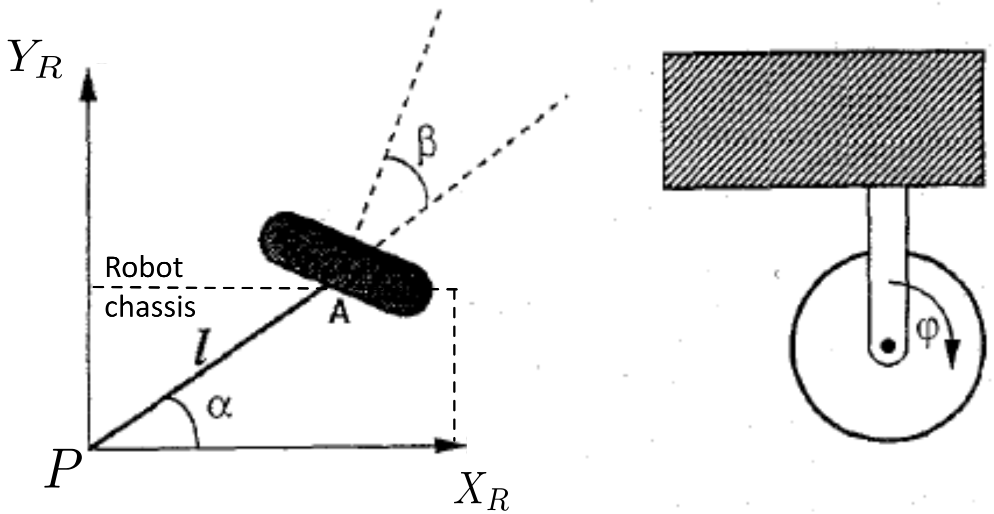
$$[\dot{x},\, \dot{y}] \cdot \begin{bmatrix} \sin\theta \\ -\cos\theta \end{bmatrix} = \dot{x}\sin\theta - \dot{y}\cos\theta = [\sin\theta, -\cos\theta, 0]\,\dot{\xi} = 0$$

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

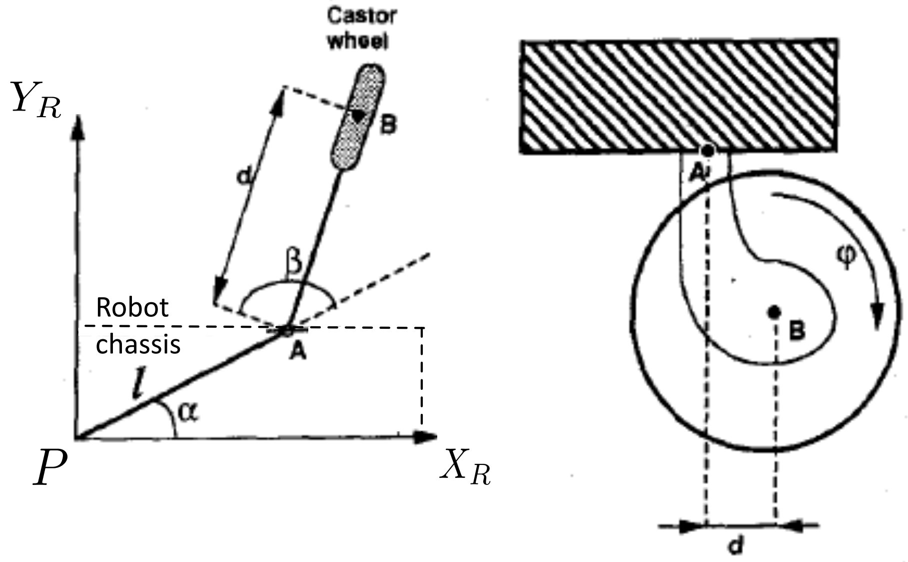


Types of wheels

- 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

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

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

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

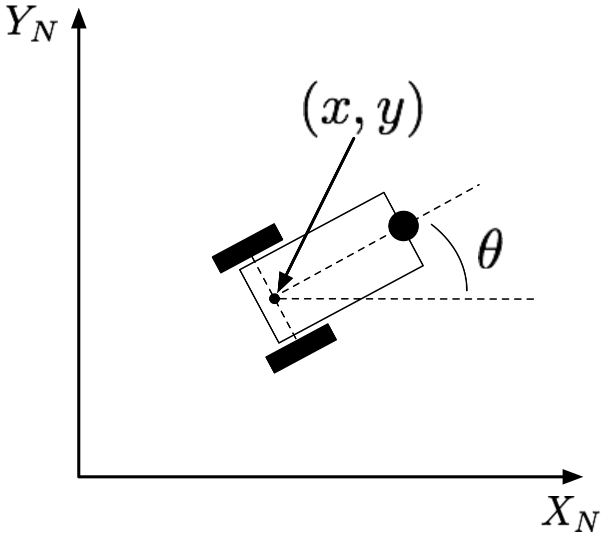
where $[g_1(\xi), g_2(\xi)]$ is a basis of the null space of $a^T(\xi)$

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

Kinematic models (examples)

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

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\zeta} \end{pmatrix} = \begin{pmatrix} \cos\zeta\ \cos\theta \\ \cos\zeta\ \sin\theta \\ \sin\zeta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2$$

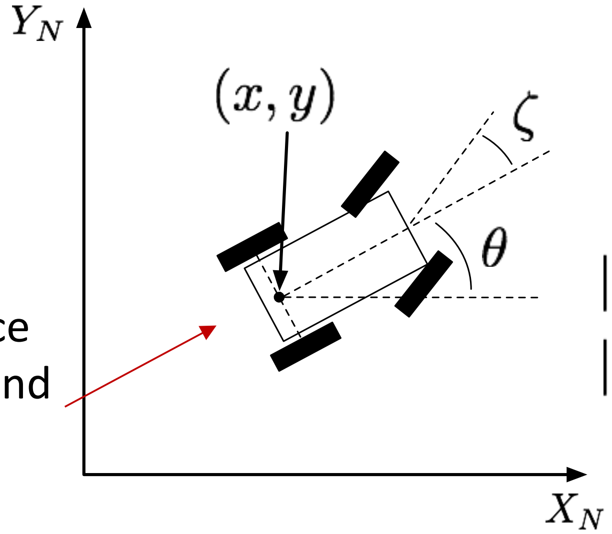


Unicycle or differential drive

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

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

Assume distance between rear and front axles is 1



Tricycle or simple car

$$|u_1| \leq u_{1,\max}$$

$$|u_2| \leq u_{2,\max}$$

$$|\zeta| \leq \zeta_{\max}$$

Warning: a kinematic state space model should be interpreted only as a subsystem of a more general dynamical model

Simplified car models

- Assuming we do not care about the direction of the front wheels, set

$$v = u_1 \cos\zeta, \qquad \omega = u_1 \sin\zeta$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\zeta} \end{pmatrix} = \begin{pmatrix} \cos\zeta \ \cos\theta \\ \cos\zeta \ \sin\theta \\ \sin\zeta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2 \Rightarrow \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \omega$$

$|v| \leq u_{1,\max} \cos\zeta_{\max}, \ |\omega| \leq |v| \tan\zeta_{\max}$ ⟶ Car-like robot

$|v| = u_{1,\max} \cos\zeta_{\max}, \ |\omega| \leq |v| \tan\zeta_{\max}$ ⟶ Reeds&Shepp’s car

$v = u_{1,\max} \cos\zeta_{\max}, \ |\omega| \leq |v| \tan\zeta_{\max}$ ⟶ Dubins’ car

- Reference: J.-P. Laumond. Robot Motion Planning and Control. 1998.

Next time

ROS