Design Industrial Robotics Applications with MATLAB and Simulink

Presenter Name Here

Trends in Industrial Robotics

Cobots Grow fastest in shipment terms with CAGR of **20%** from **2017 - 2023**

Growing trend toward compact robots

Increasing share of units shipped in 2023 will be payload <10kg

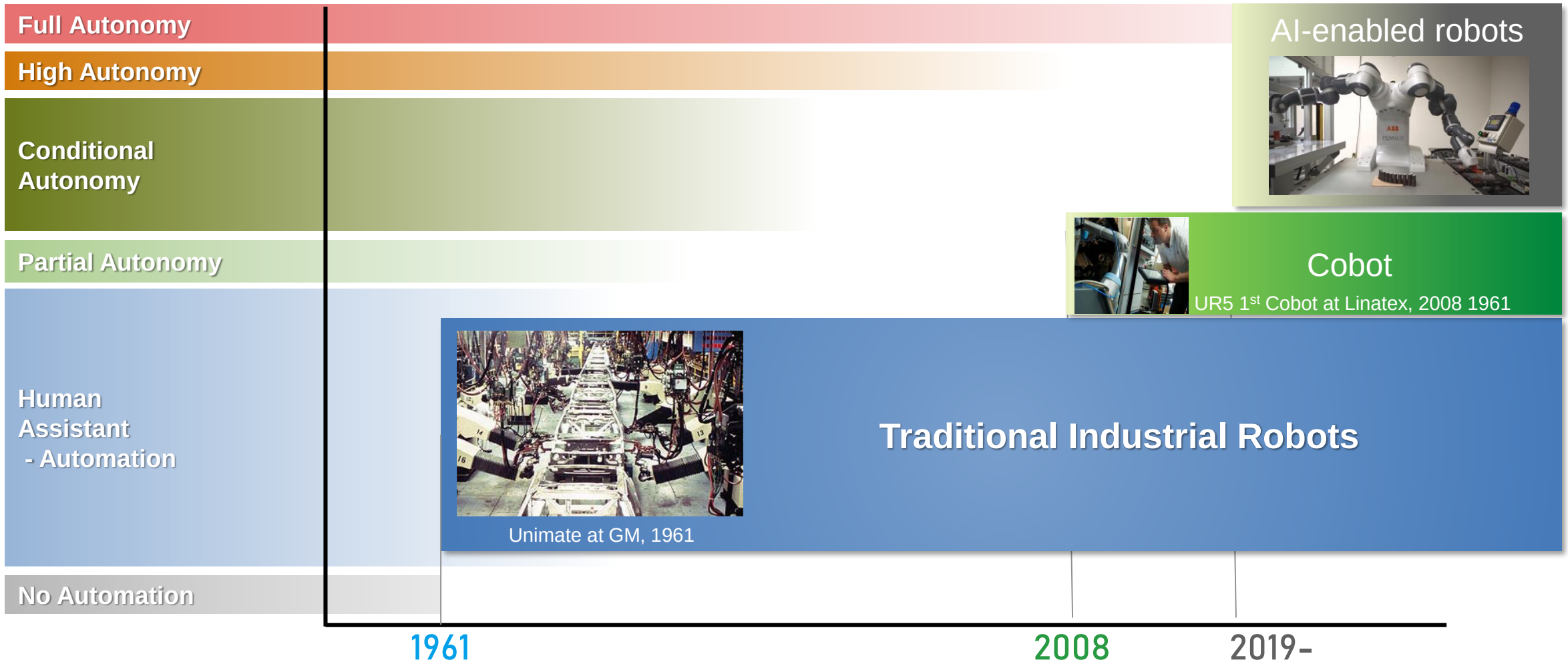
40% of Articulated Robots

80% of SCARA Robots

82% of Cobots

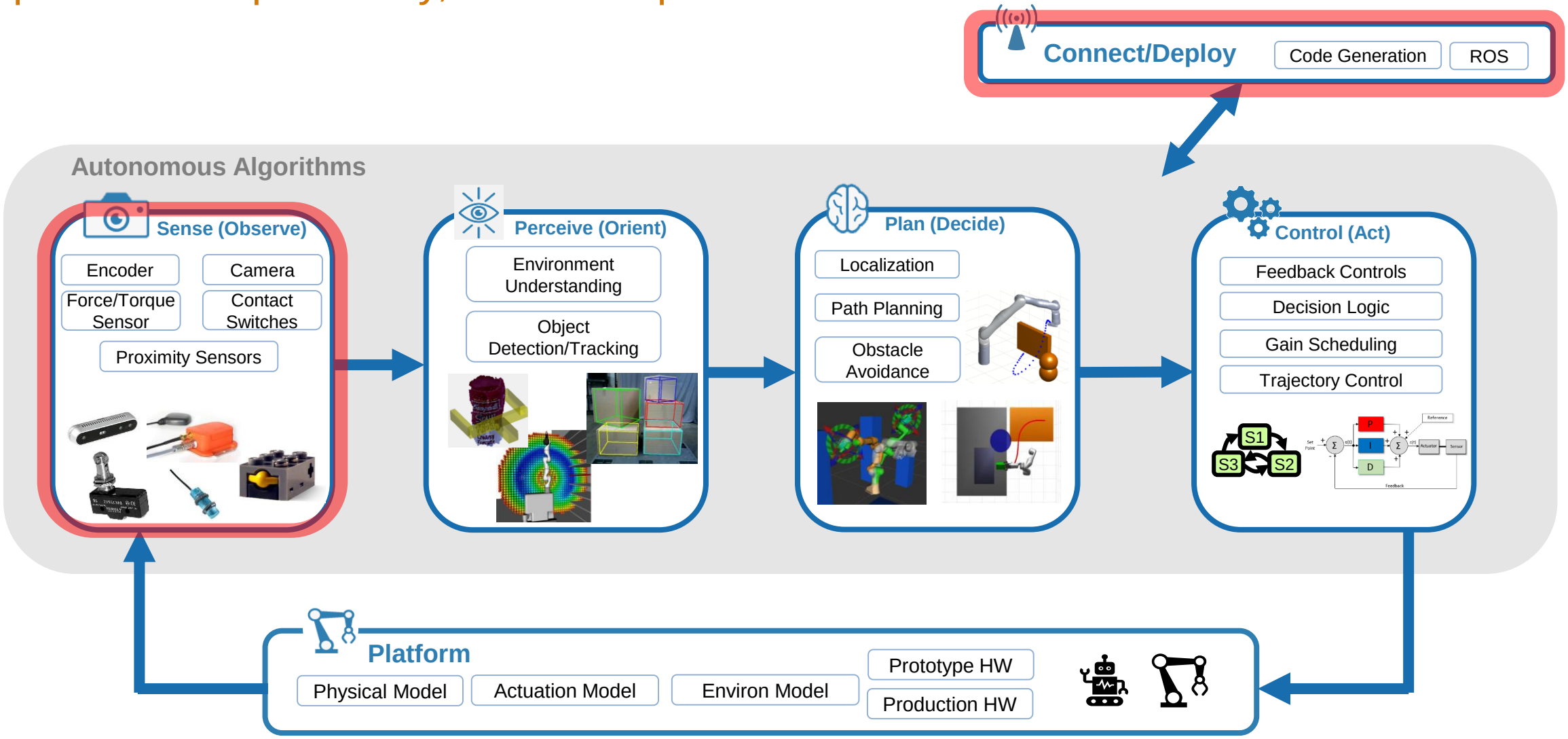
Source: Interact Analysis

Evolution of Industrial Robotics Technologies



Autonomous Industrial Robotics Systems Workflow

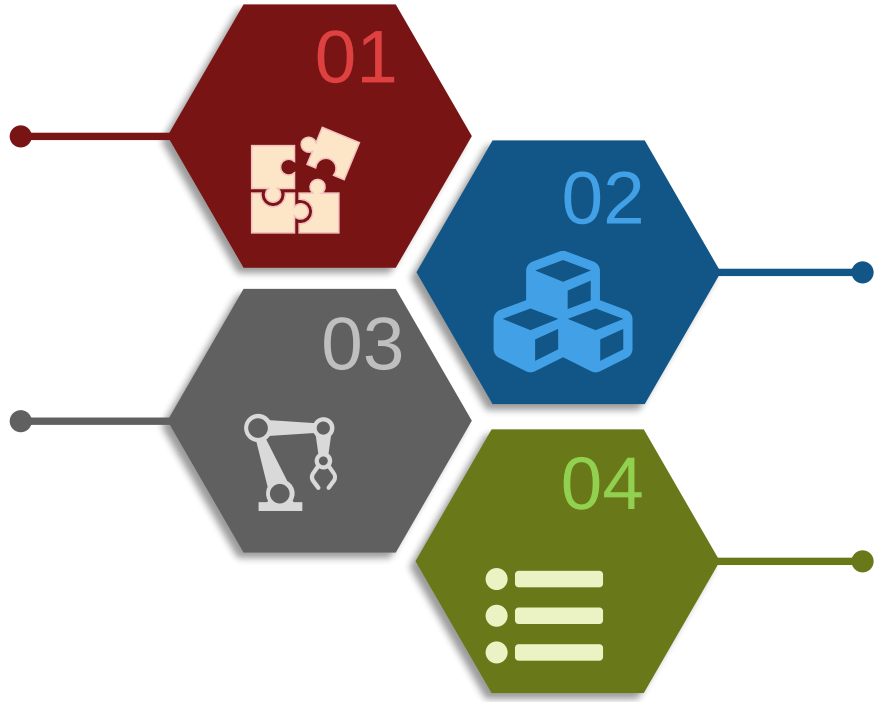
Operates independently, without explicit instructions from a human



What we'll discuss today

Challenges in Industry Robot System Design

Case Study: Pick-and-Place Manipulation Application



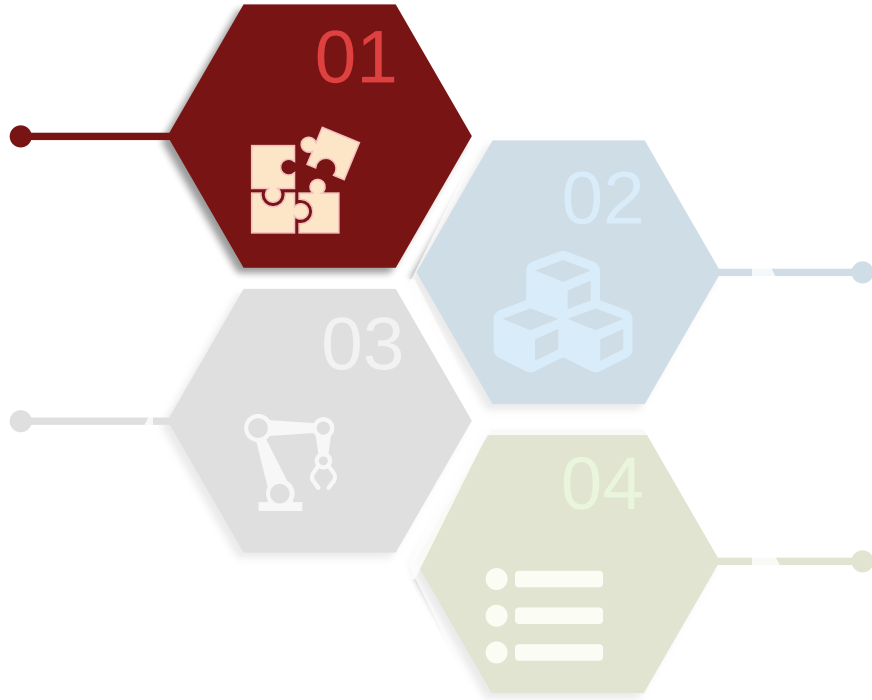
Model-Based Design for Autonomous System

Concluding Remarks

What we'll discuss today

Challenges in Industry Robot System Design

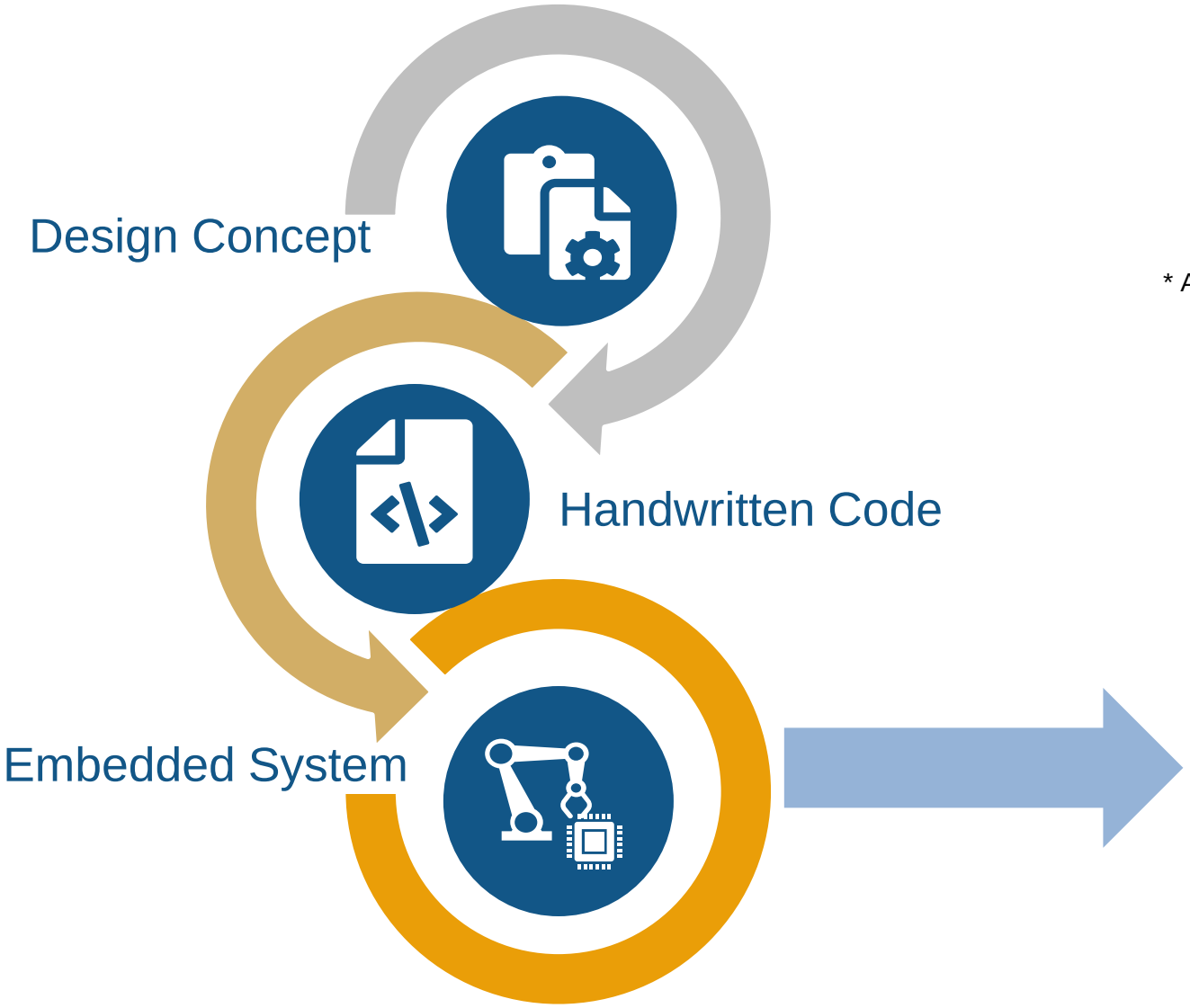
Case Study: Pick-and-Place Manipulation Application



Model-Based Design for Autonomous System

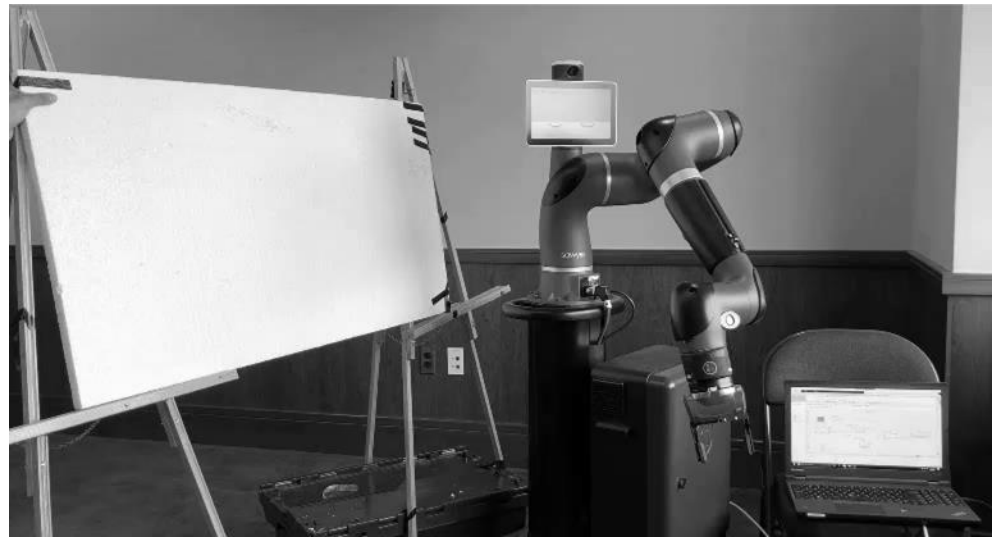
Concluding Remarks

Traditional Software Development Cycle



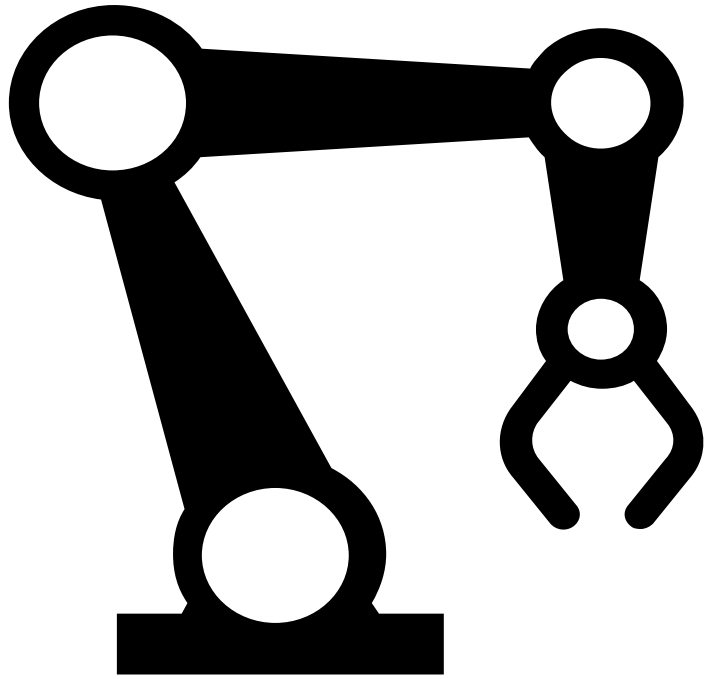
Only 6% Of Design/Development Time is Spent on Simulation*

* AspenCore - EETimes, “2019 embedded markets study,” EETimes, Tech. Rep., 2019



Common Challenges of Industrial Robotics Systems Development

Multidomain Expertise



End-to-End workflows

Complexity of Algorithms

Technical Depth and System Stability

Key Takeaways

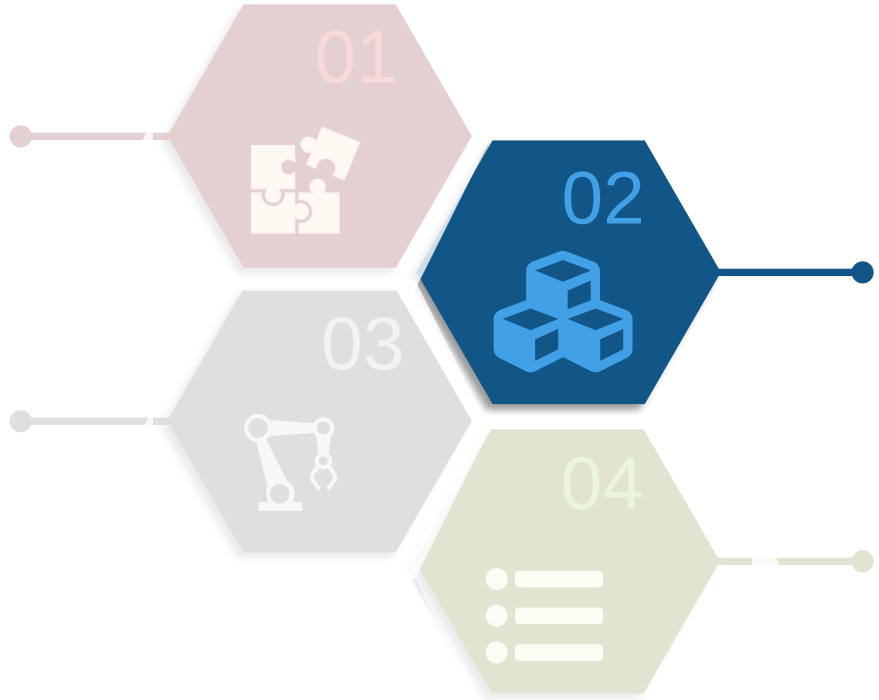
In this talk, you will learn

- Reference workflow for industrial robot development
- Multi-domain functional areas of Platform, Sensing, Perception, Planning and Control
- MATLAB and Simulink capabilities to develop new robot algorithms
  - » Kinematic and dynamic models of robots
  - » Perception algorithm design using deep learning
  - » Gazebo co-simulation for sensor models and environment simulation
  - » Path planning with obstacle avoidance
  - » Supervisory logic and control using Stateflow / RL
  - » C/C++ code / ROS nodes generation

What we'll discuss today

Challenges in Industry Robot System Design

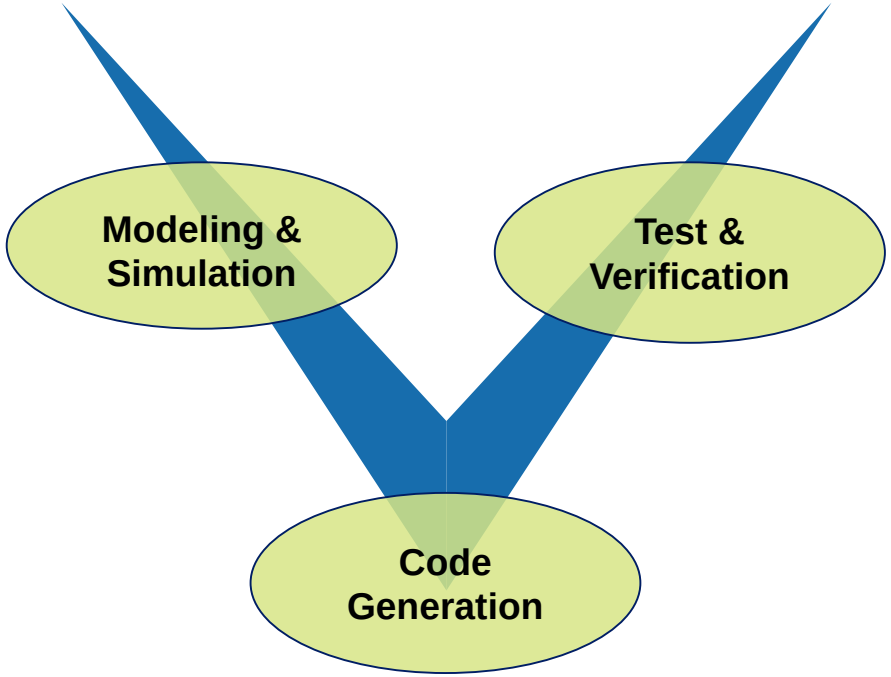
Case Study: Pick-and-Place Manipulation Application



Model-Based Design for Autonomous System

Concluding Remarks

Key to developing robust autonomous system

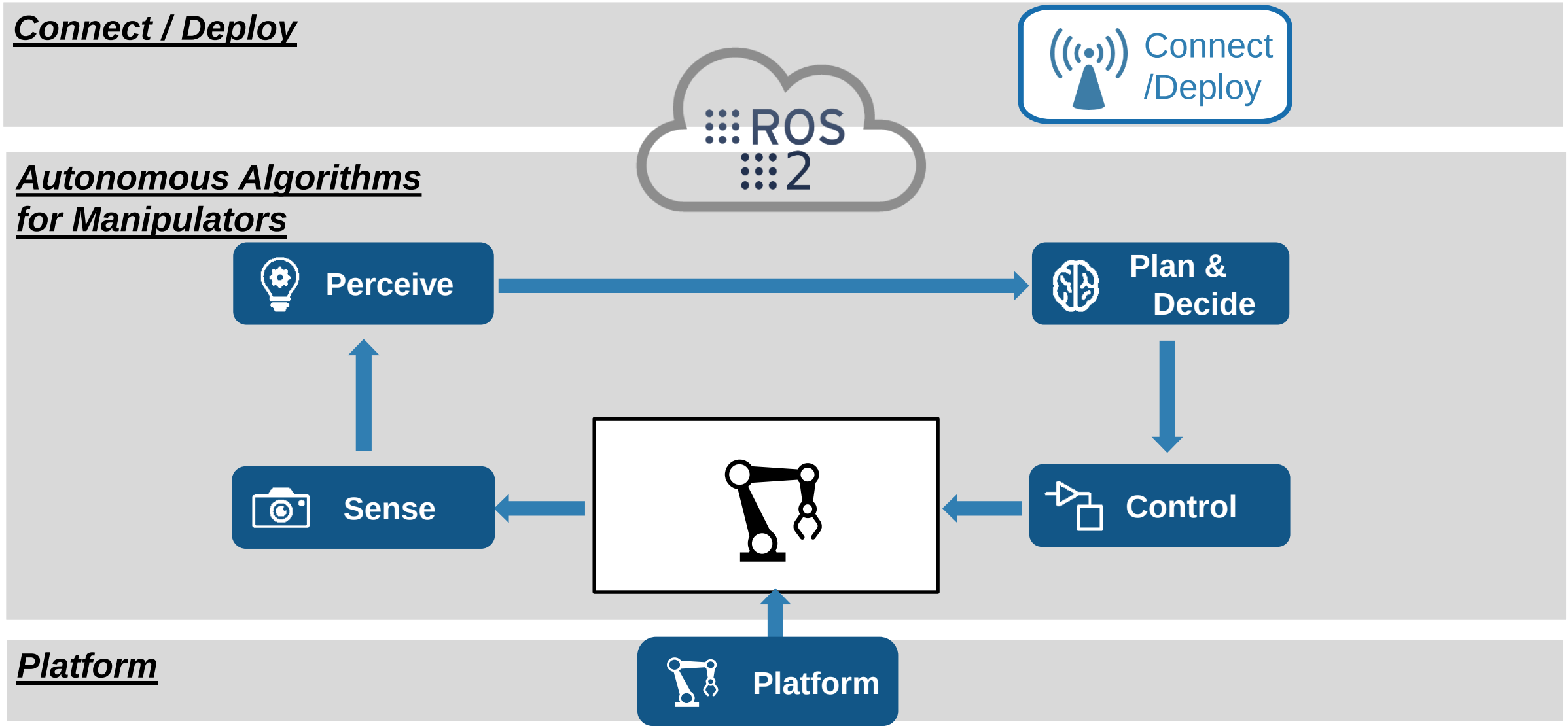


Complete Model-Based Design Workflow

Need: An end-to-end development solution that includes modeling & simulation, code generation and test & verification.

Simulate First and Simulate Often!

Full Model-Based Design Workflow



What we'll discuss today

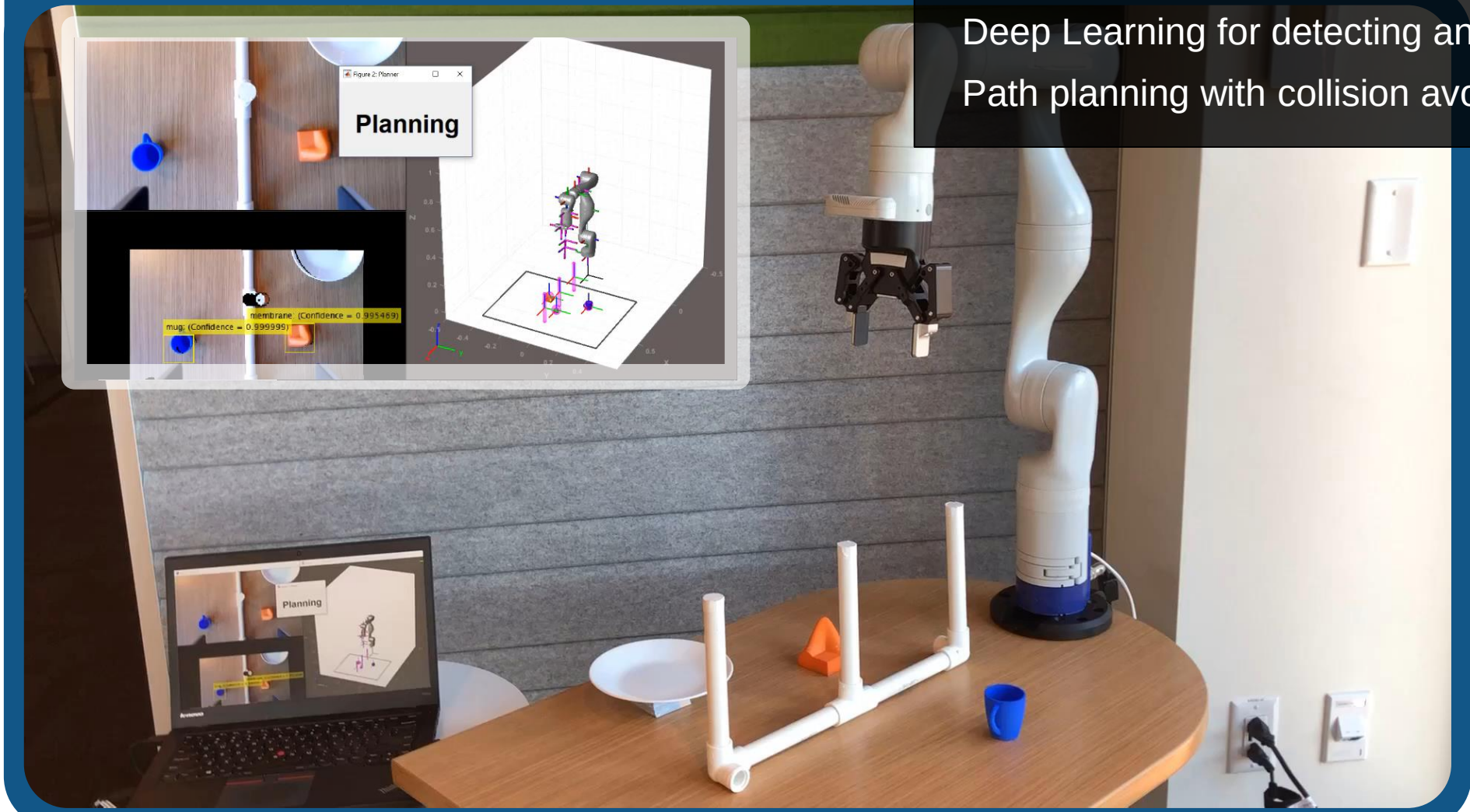
Challenges in Industry Robot System Design

Case Study: Pick-and-Place Manipulation Application

Model-Based Design for Autonomous System

Concluding Remarks

Robot Arm Demo

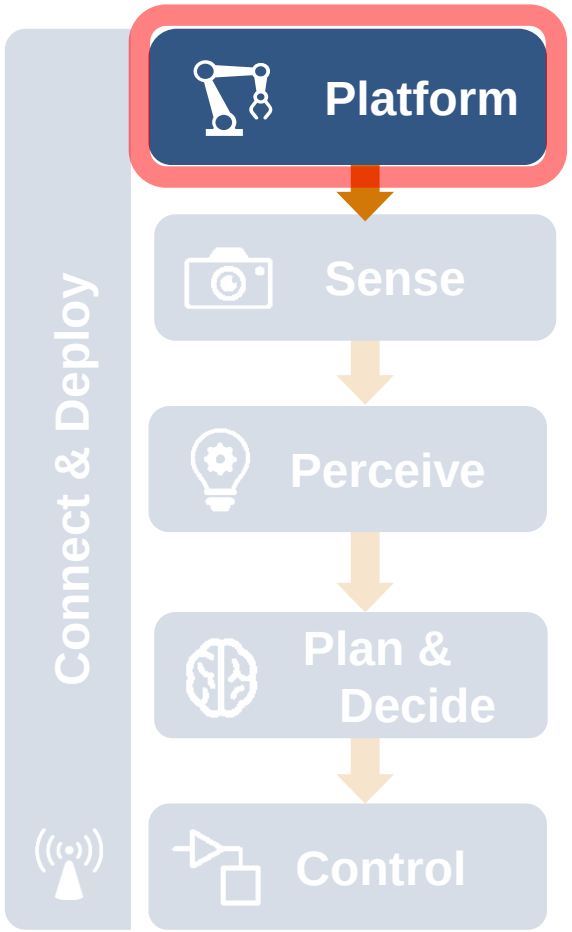


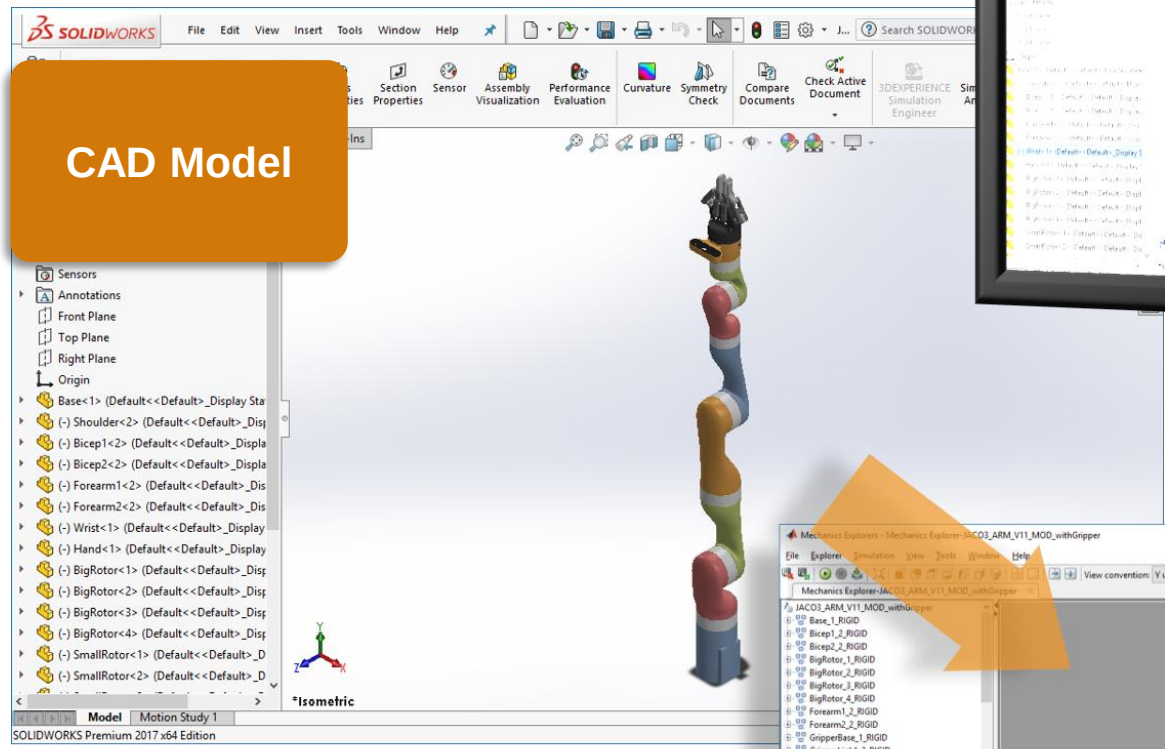
Pick-and-Place Manipulators

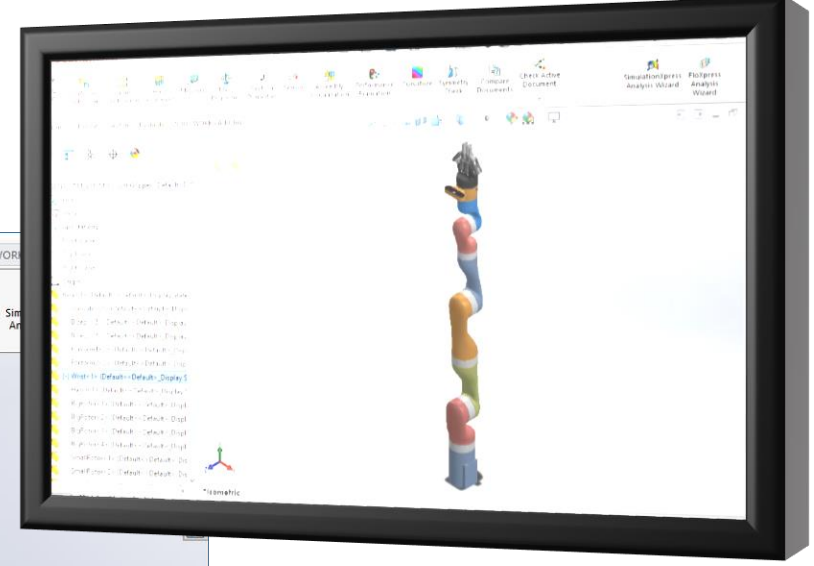
- Model-Based Design
- Deep Learning for detecting an object
- Path planning with collision avoidance

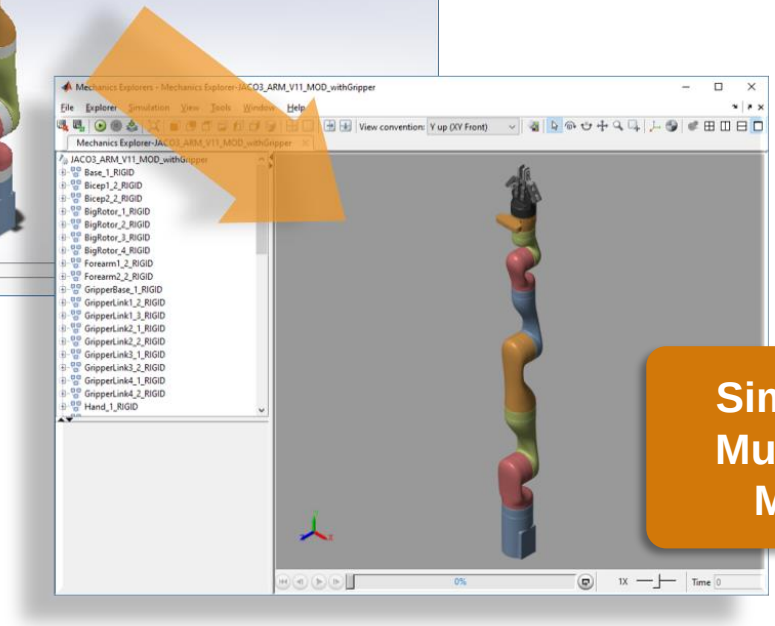
Mechanical Modeling

Automatic import from CAD Tools







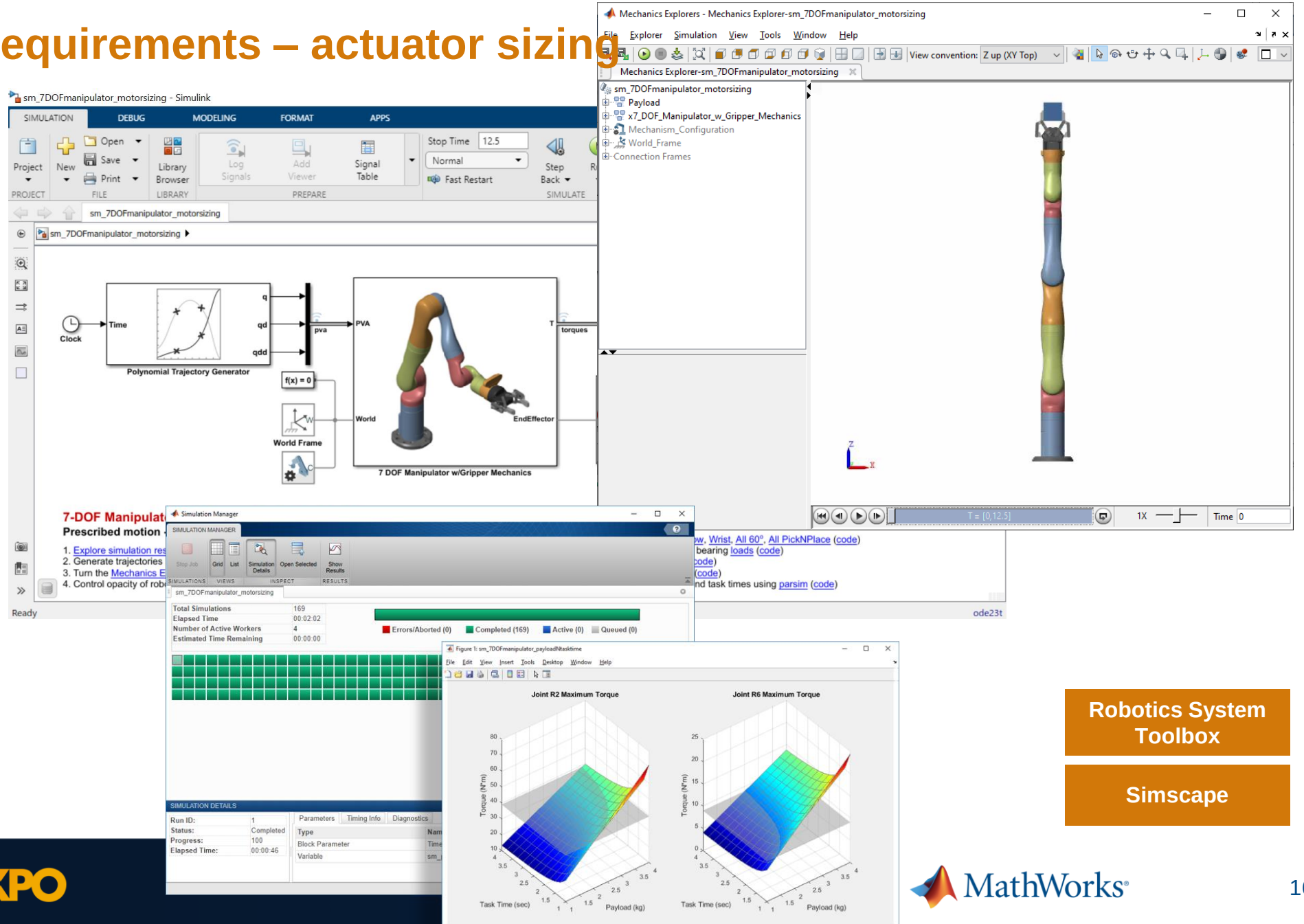


Actuators

Evaluating motor requirements – actuator sizing

Connect & Deploy

- Platform
- Sense
- Perceive
- Plan & Decide
- Control



Robotics System Toolbox

Simscape

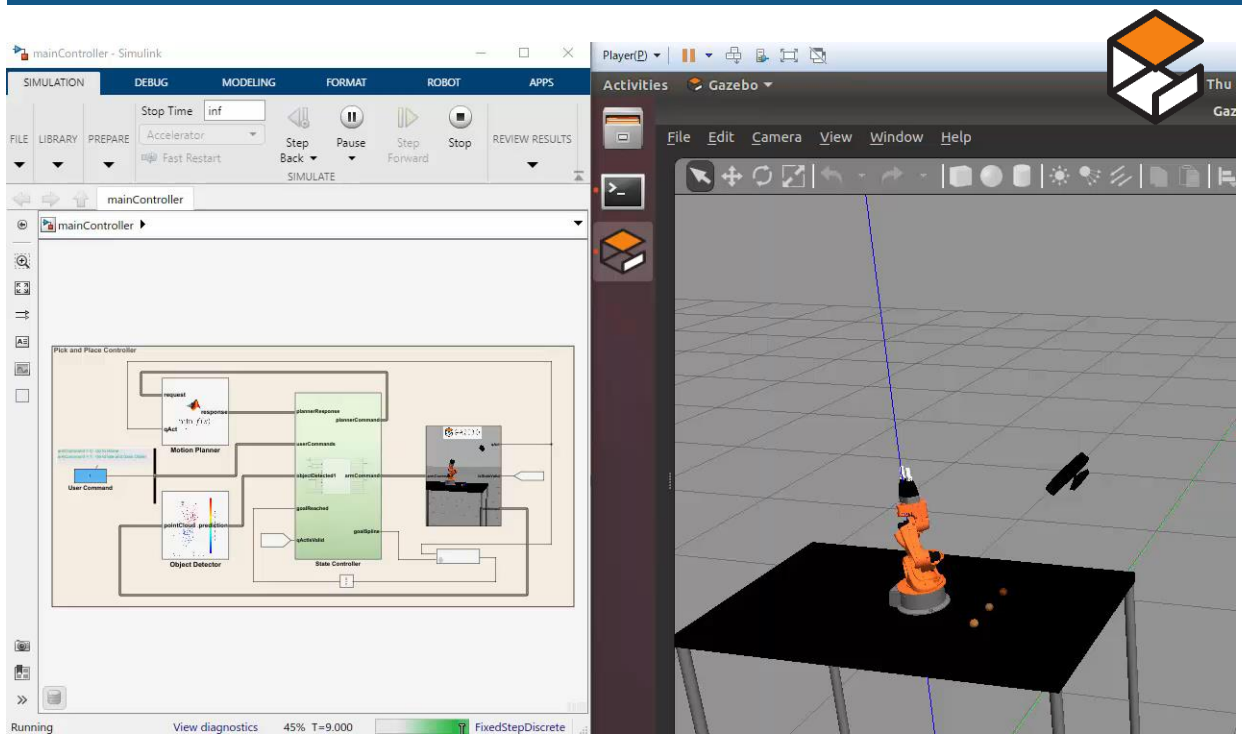
Environment Modeling

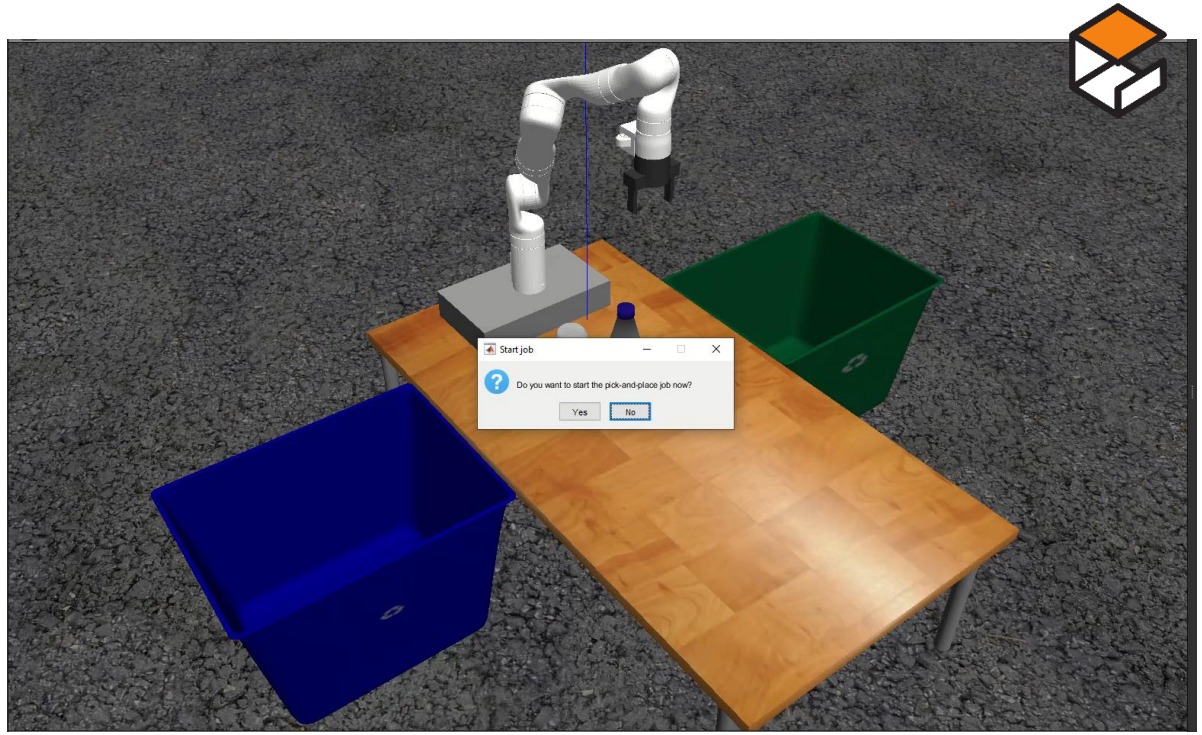
Connect to an external robotics simulator

Robotics System Toolbox

ROS Toolbox

Robot arm simulation with Gazebo



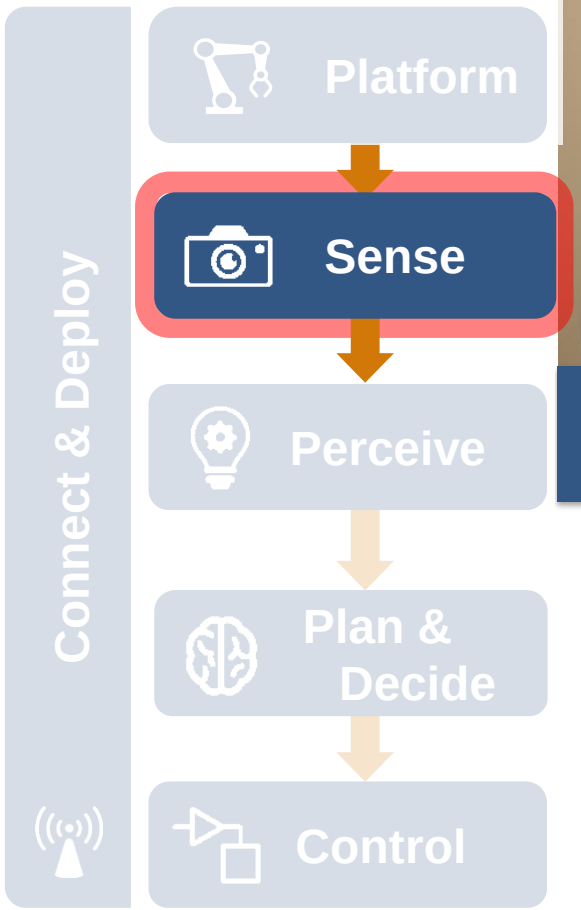


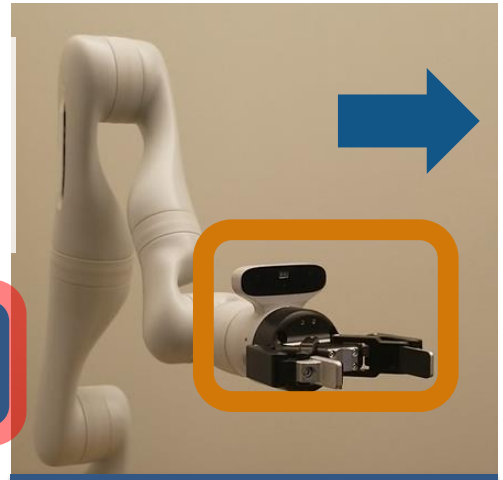
Gazebo: Physics-based simulator with sensors and noise

Sensing

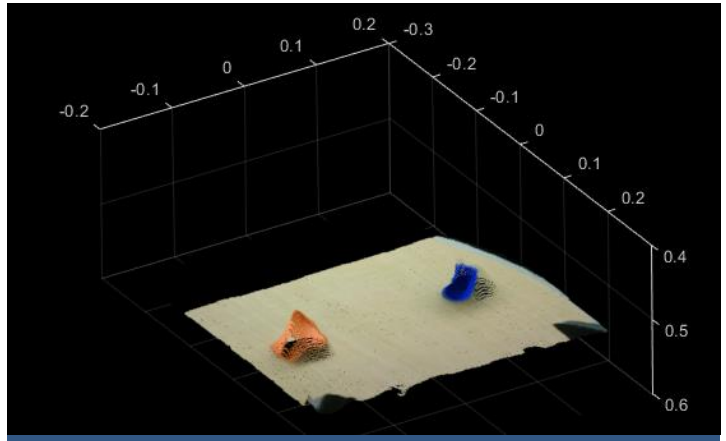
Point cloud processing for pose estimation

Computer Vision Toolbox

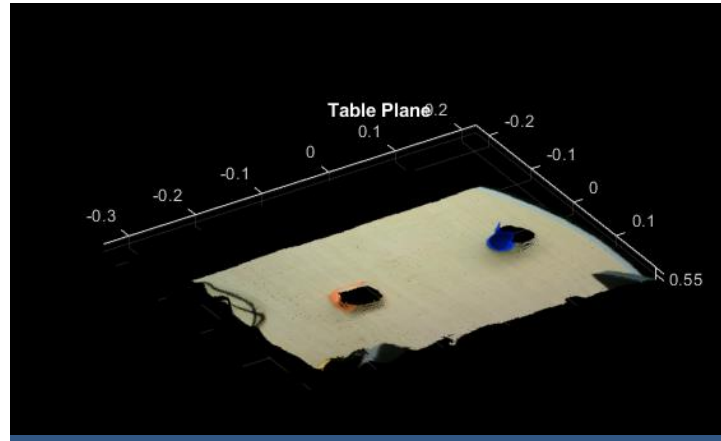




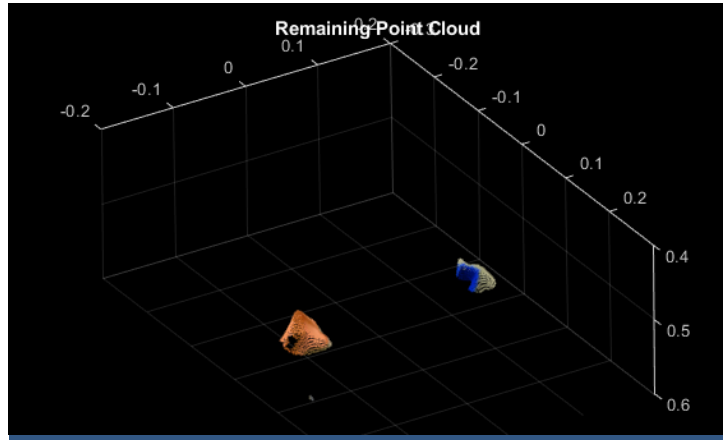
Intel® RealSense™ RGB-D camera



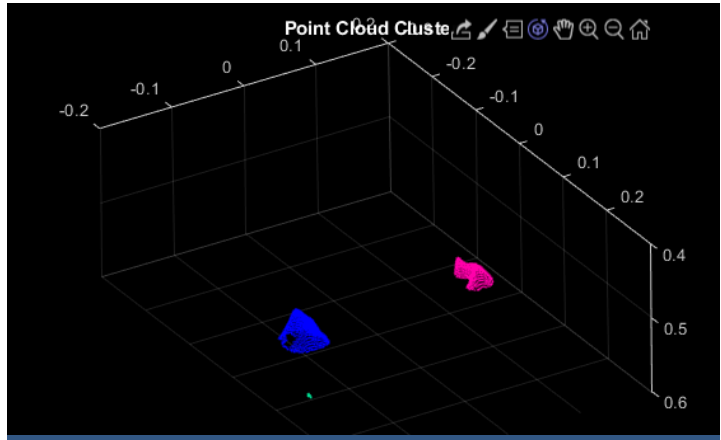
Colorized point cloud



Detect table



Point clouds of objects



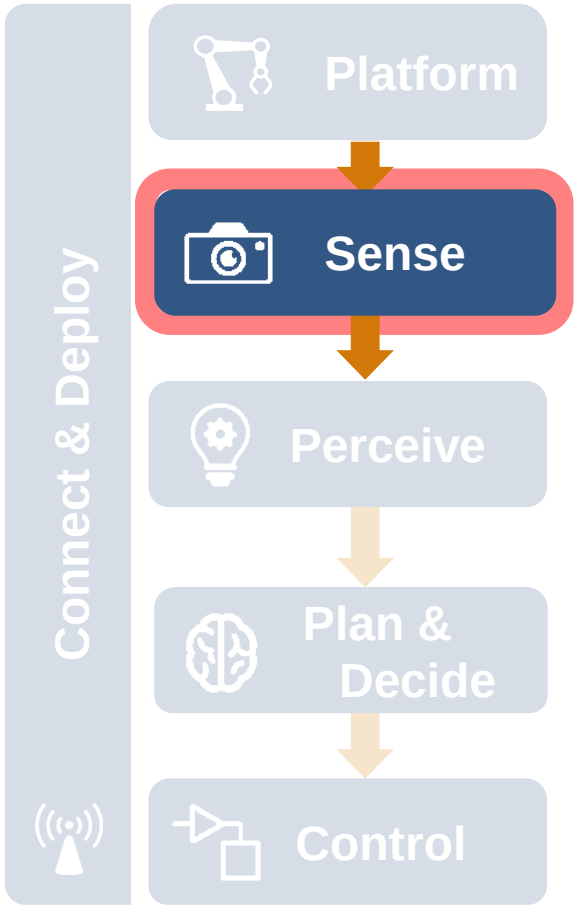
Remove noise and cluster

Sensing

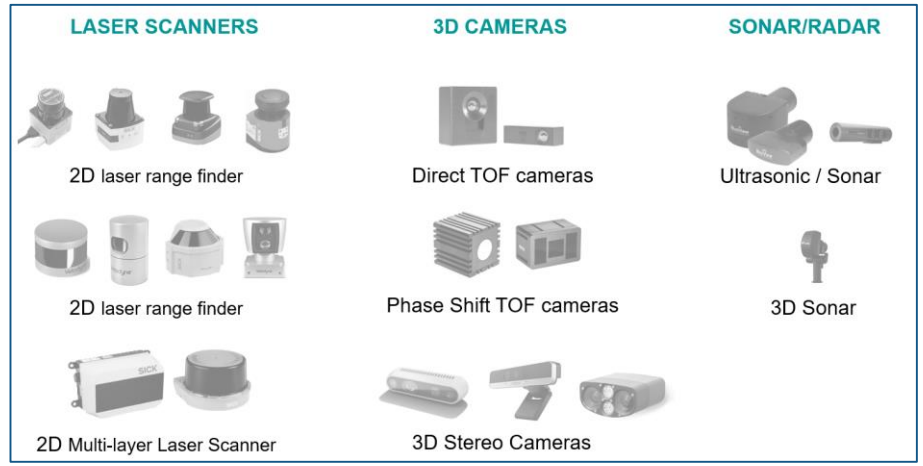
Common sensors and sensing functionalities for autonomous systems

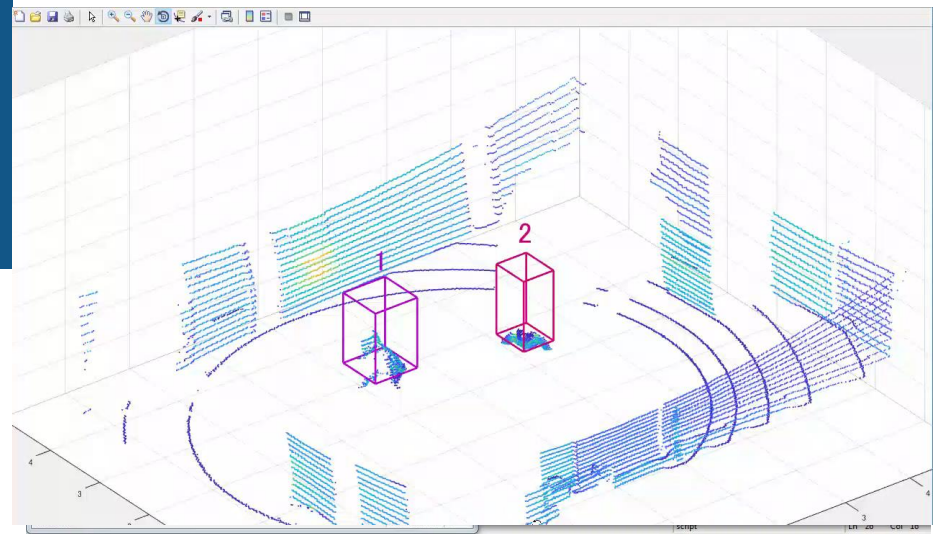
Computer Vision Toolbox

Image Processing Toolbox



- Support for Common Sensors
- Image analysis
- Image enhancement
- Visualizing Point Clouds
- Apps

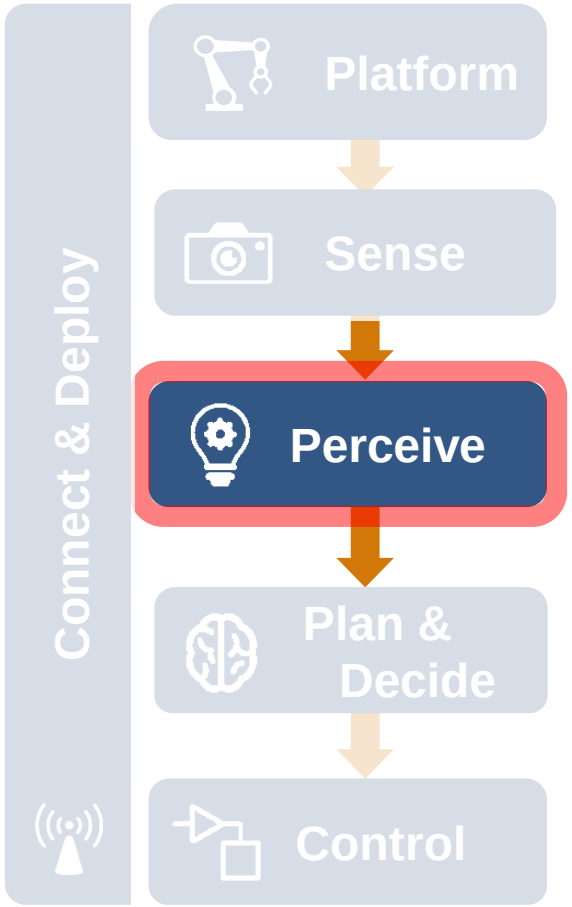


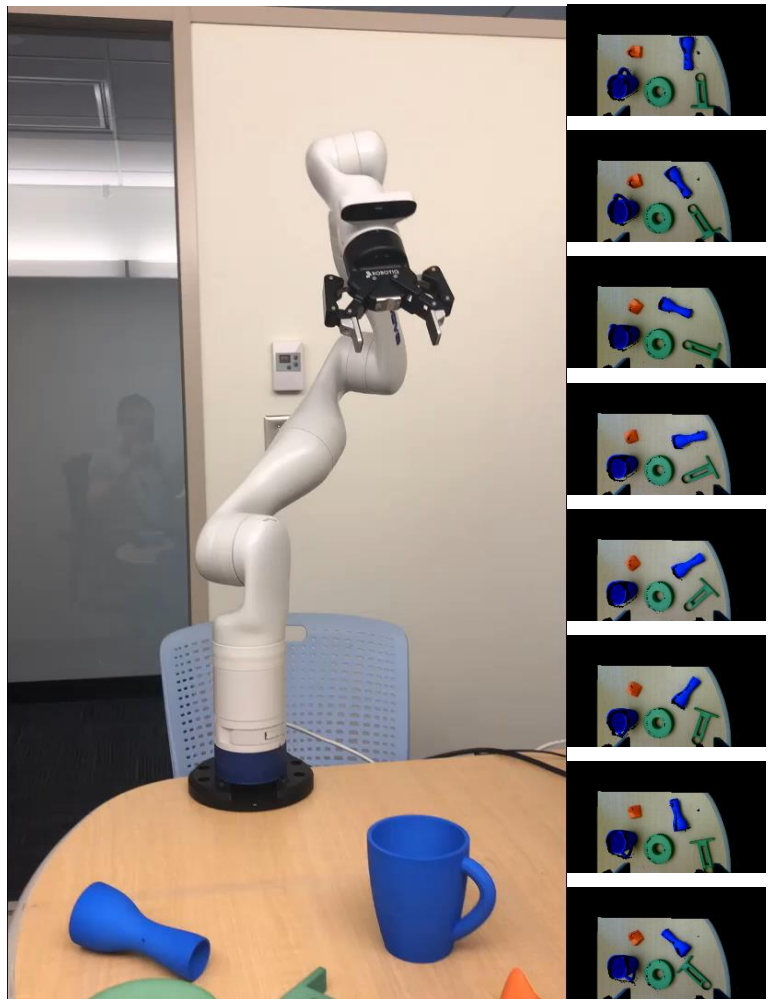


Perception

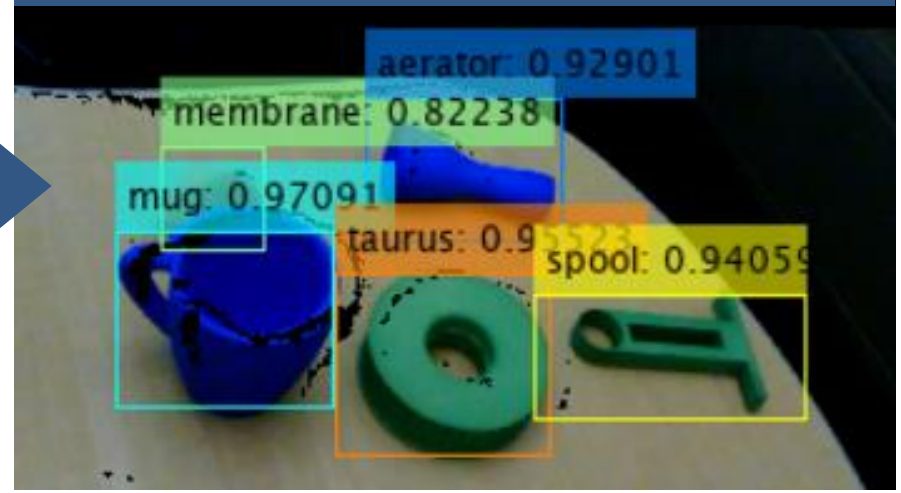
Deep learning for object classification

Deep Learning Toolbox

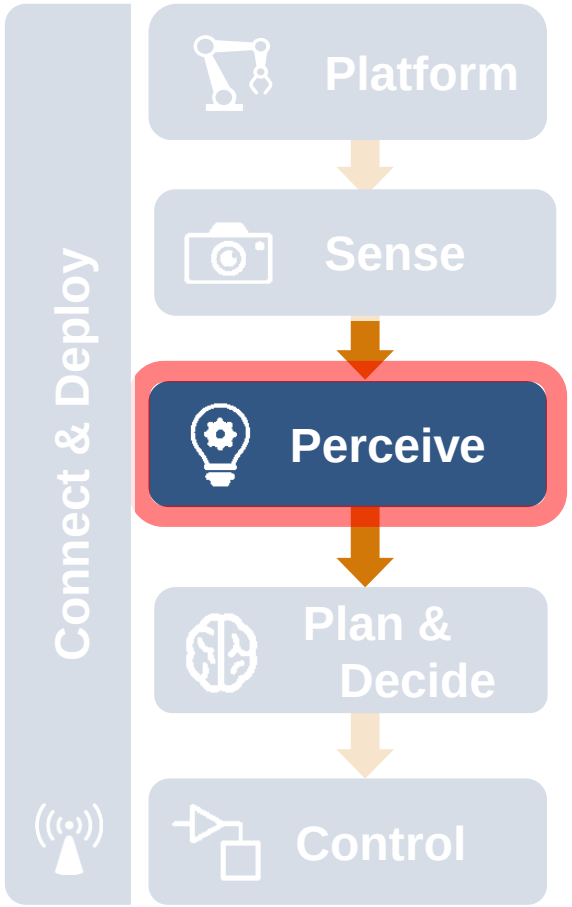


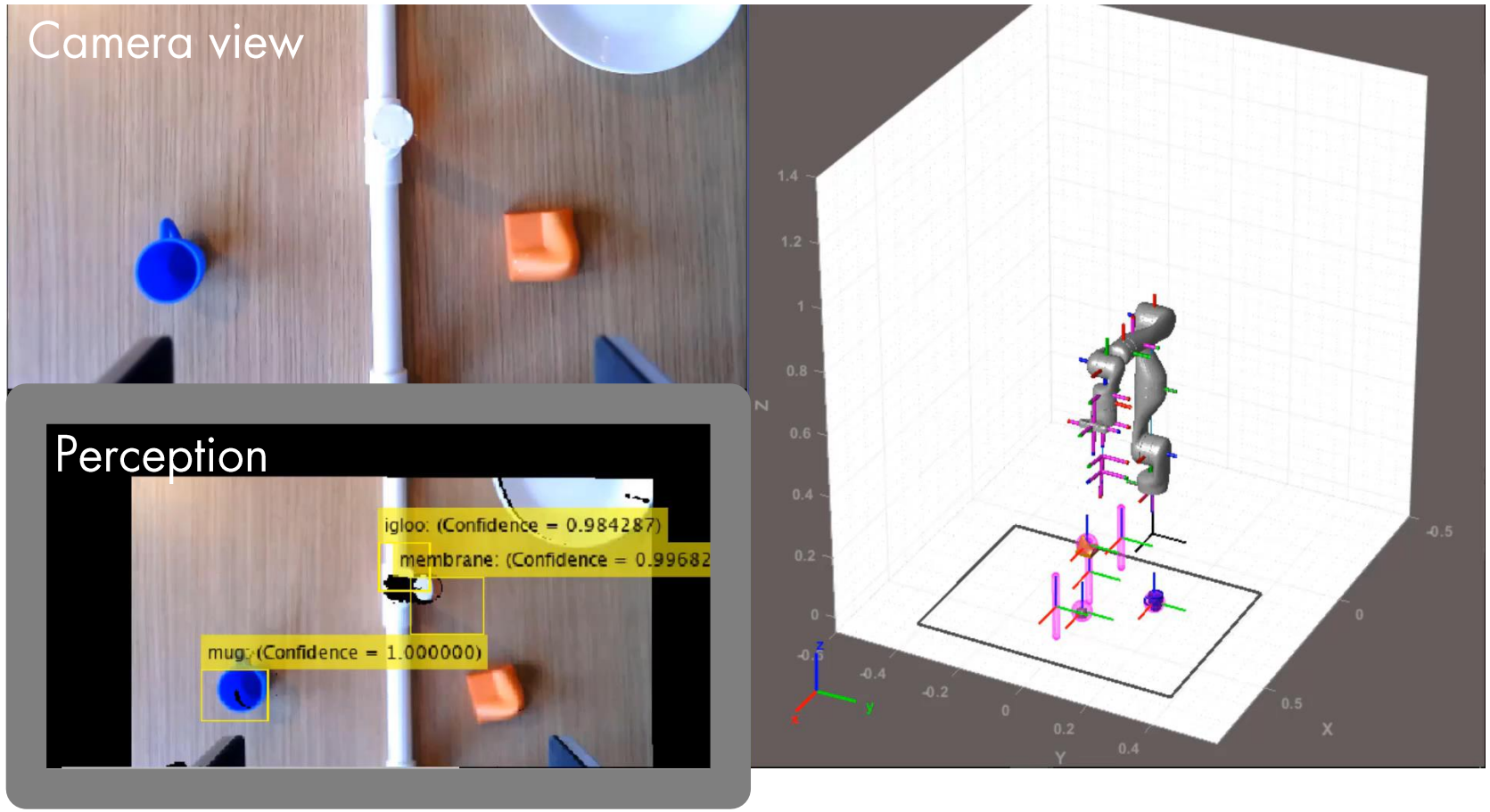


Object detector using Deep Learning (YOLO v2)

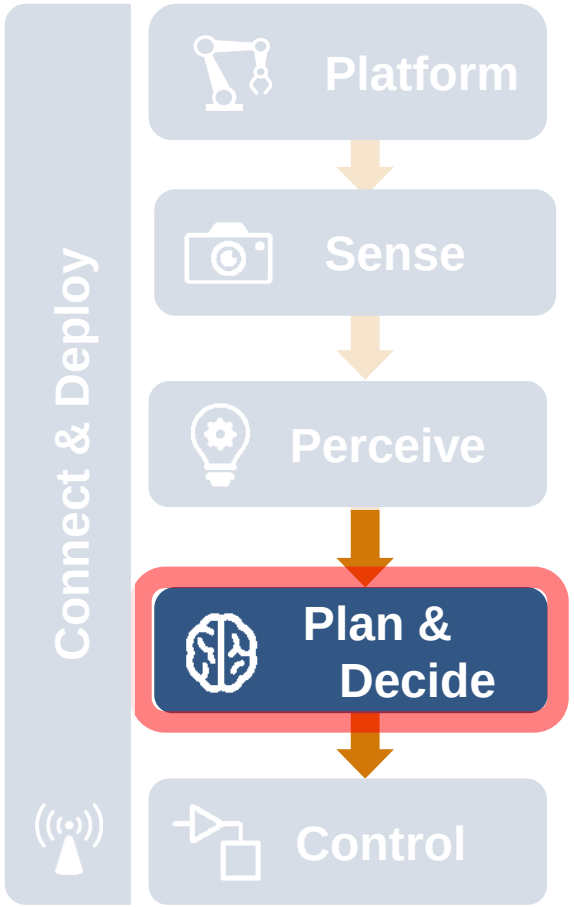


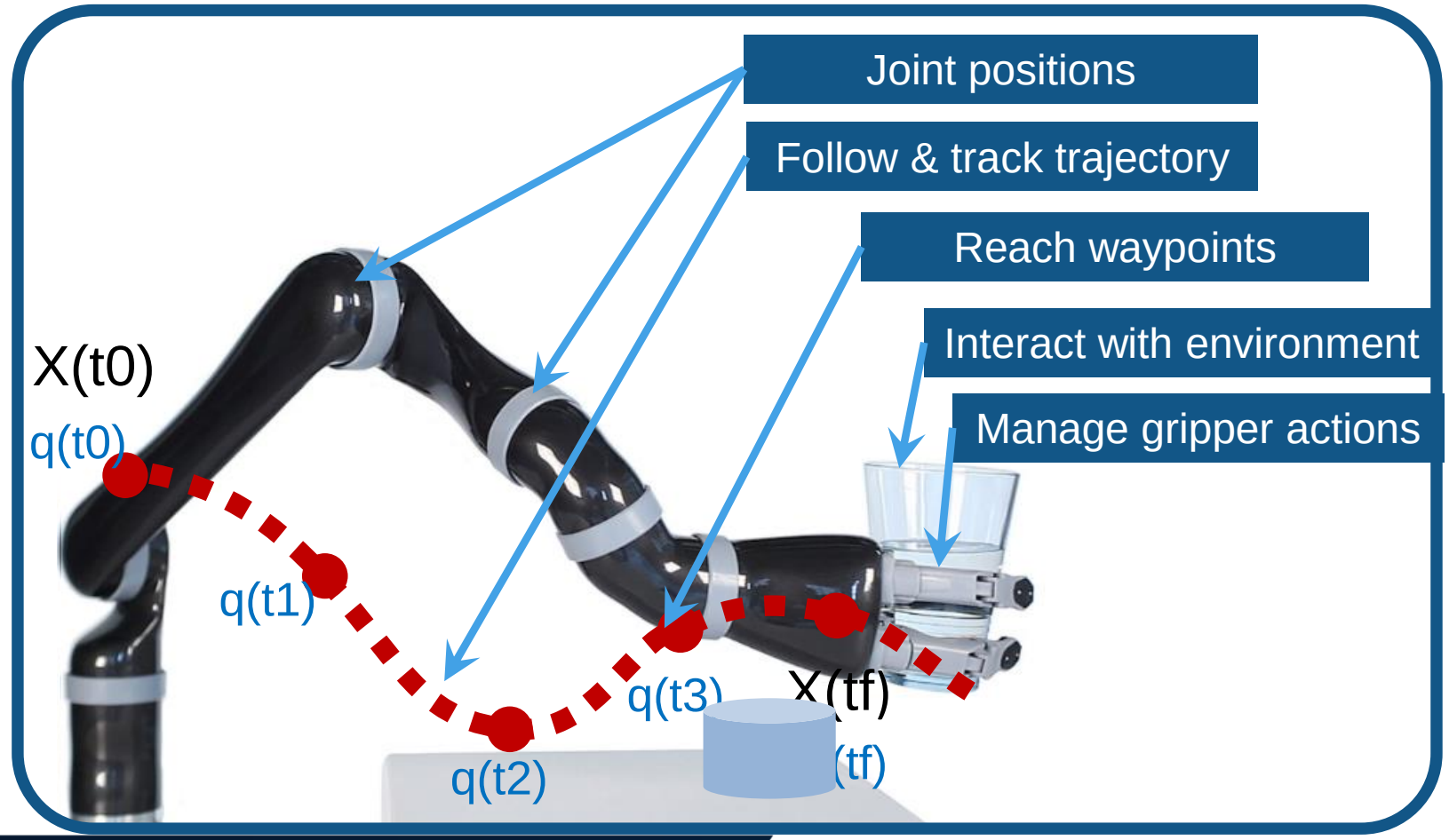
Perception
Object Classification





Motion Planning



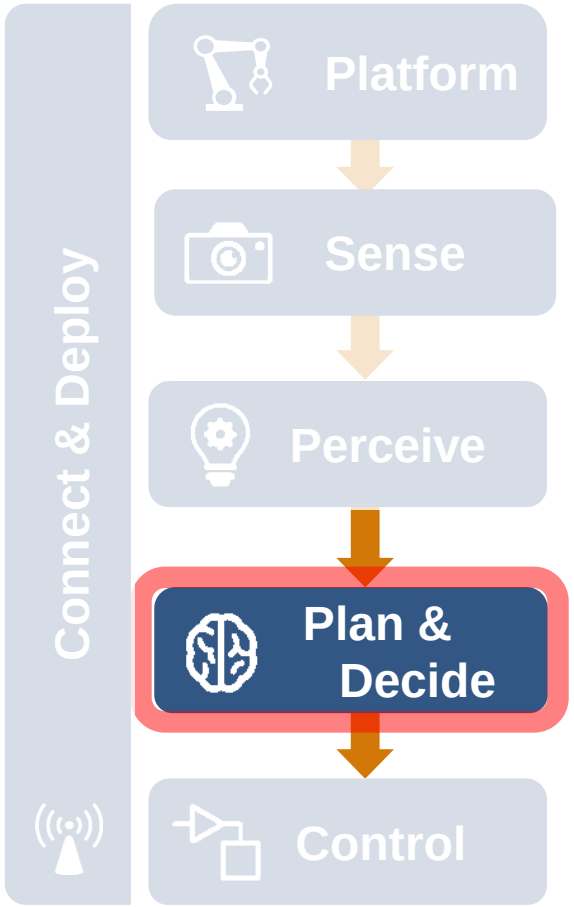


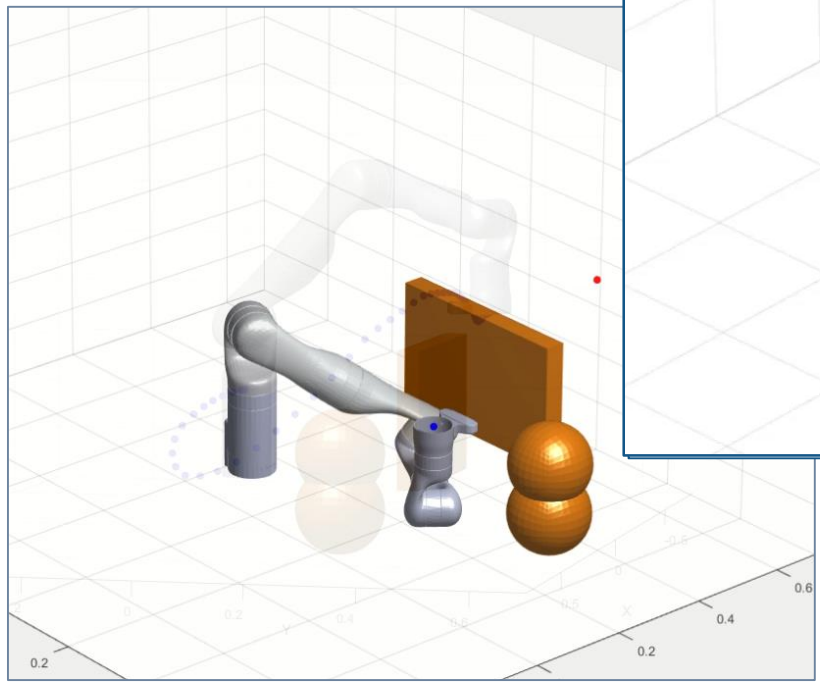
Motion Planning

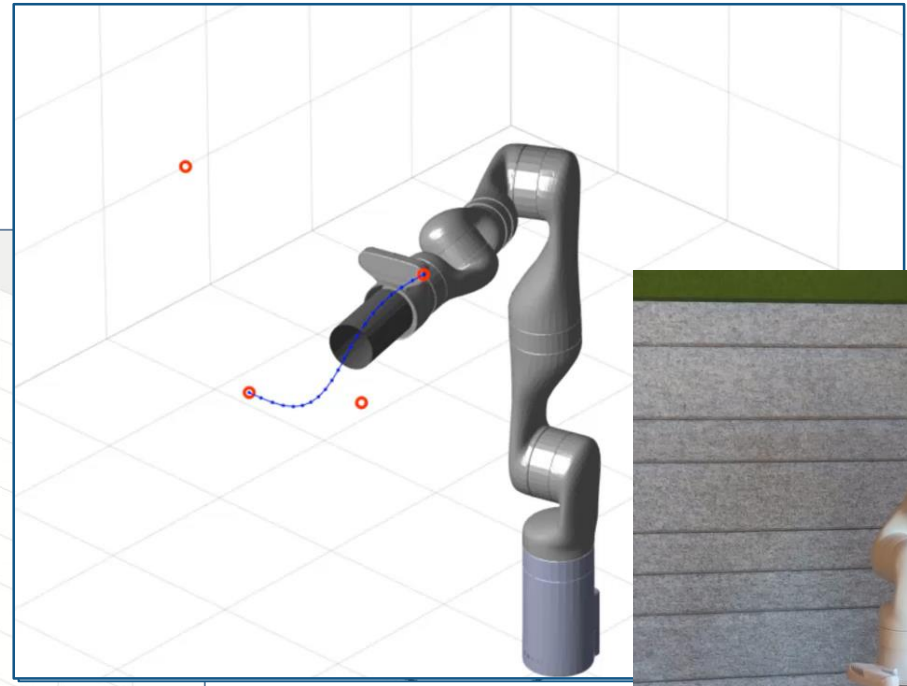
Path Planning + Trajectory Gen + Trajectory Following

Robotics System Toolbox

Model Predictive Control Toolbox







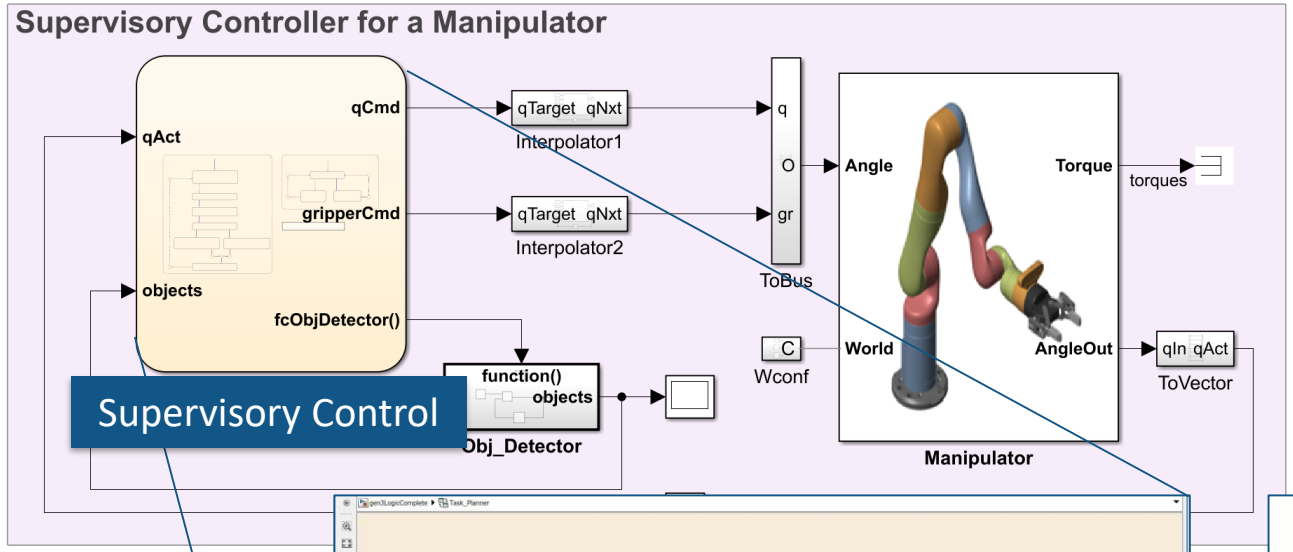
Motion Control

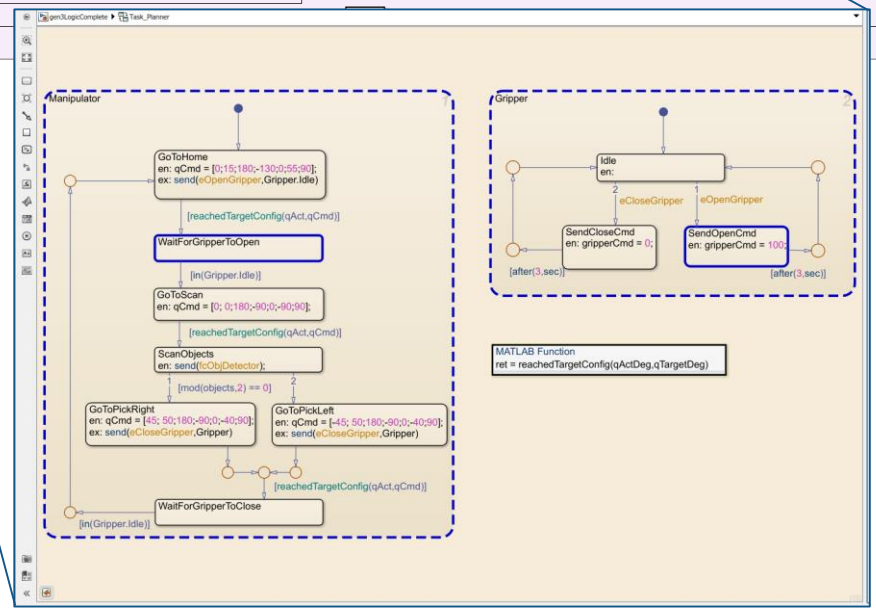
Decision Logic

Stateflow

Connect & Deploy

- Platform
- Sense
- Perceive
- Plan & Decide
- Control

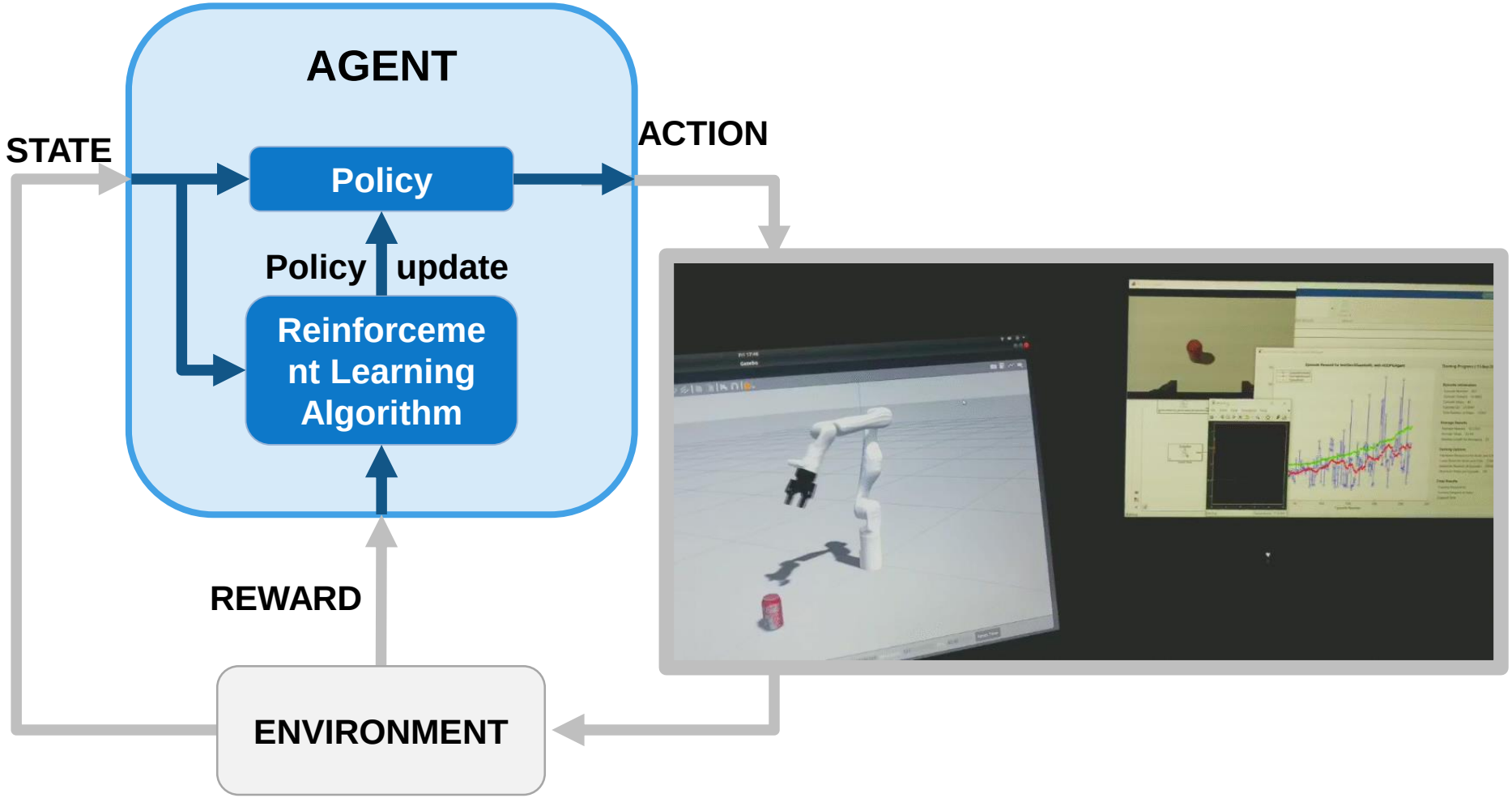




Advanced Control: Reinforcement Learning

Grasping an object with image inputs

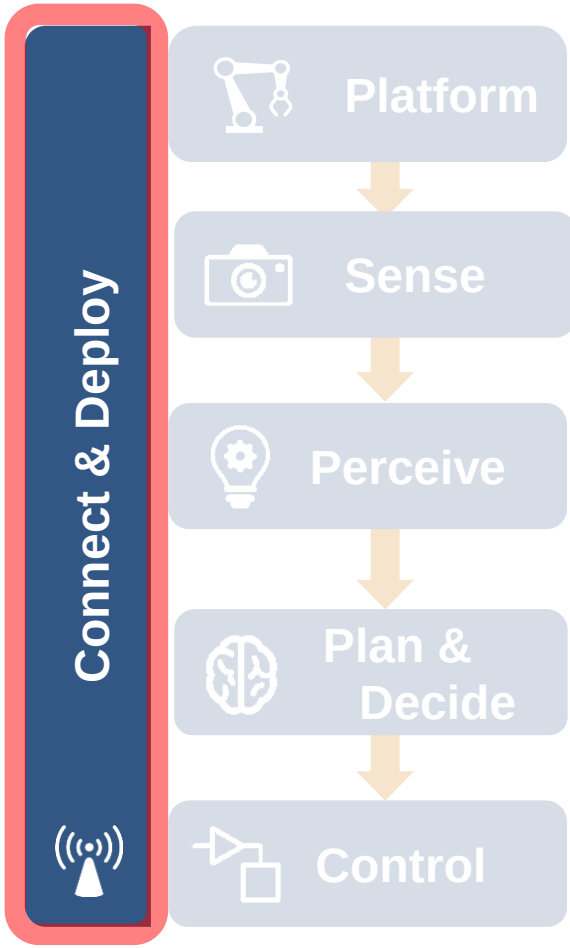
Reinforcement Learning Toolbox

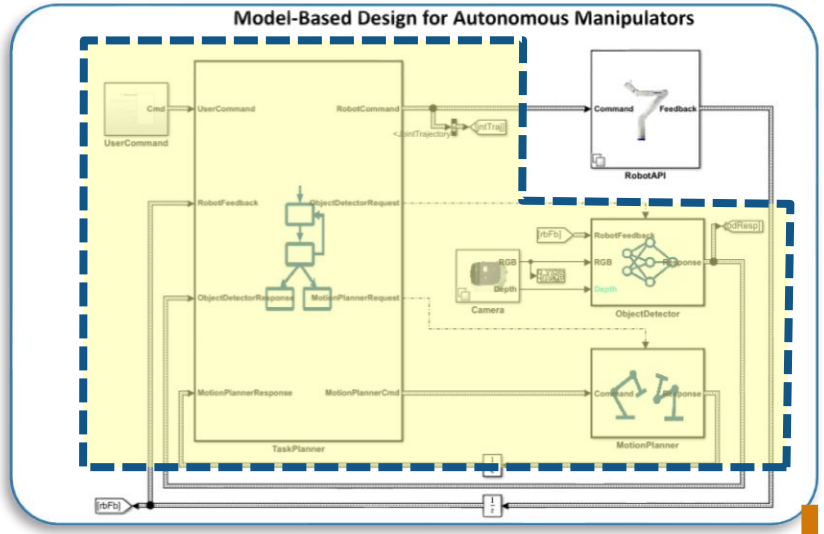


Hardware Connectivity

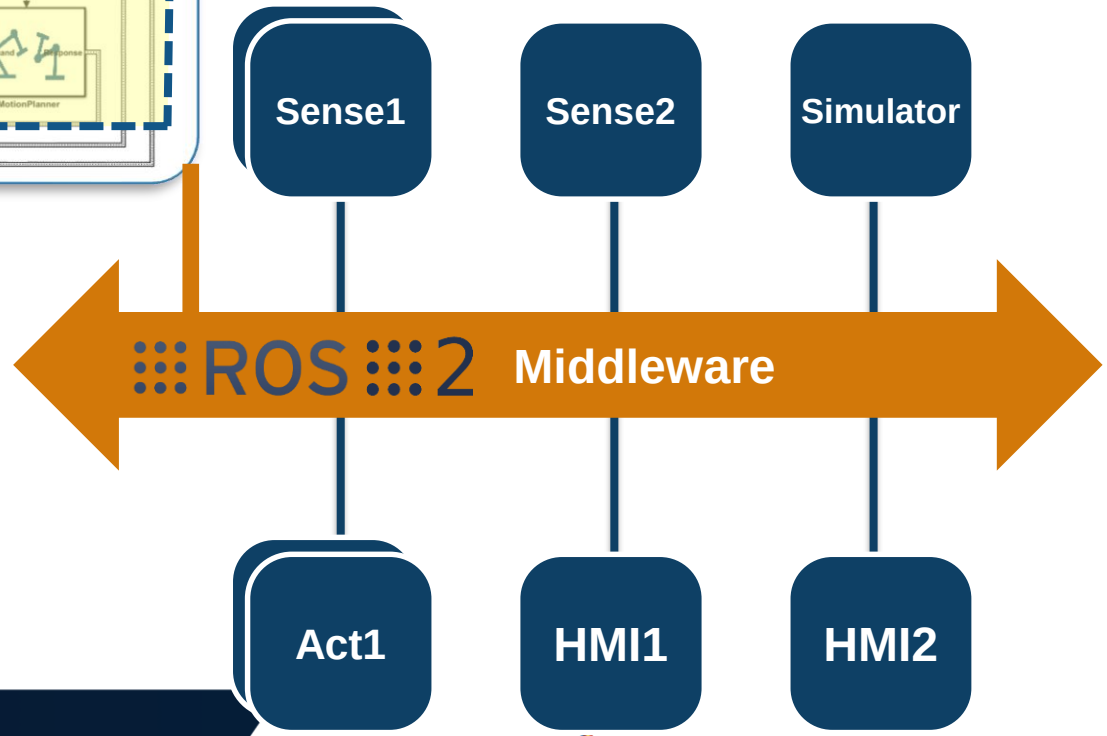
Code Generation Support

ROS Toolbox





Application



Hardware Connectivity

Code Generation Support

Stateflow

Connect & Deploy

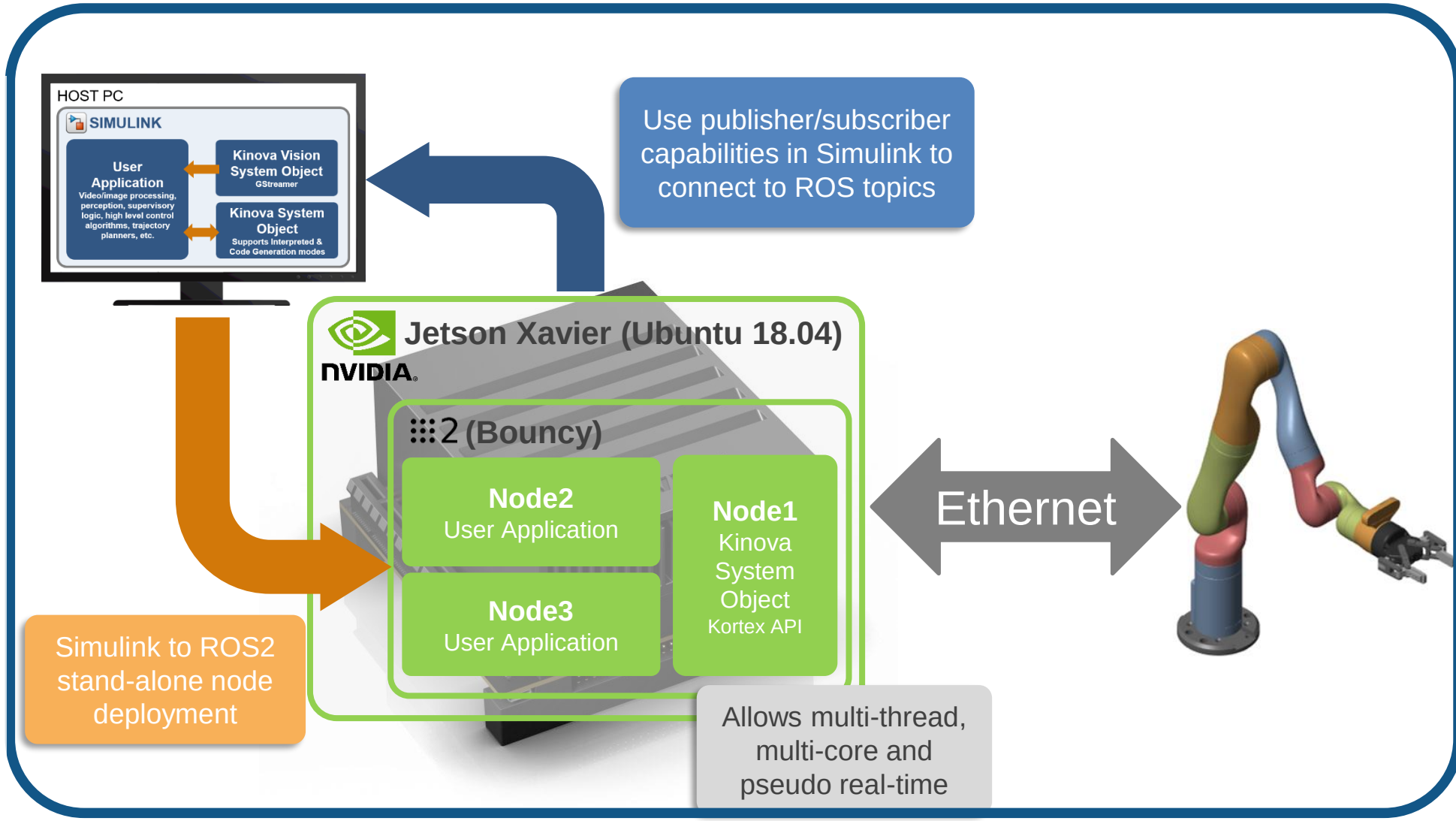
Platform

Sense

Perceive

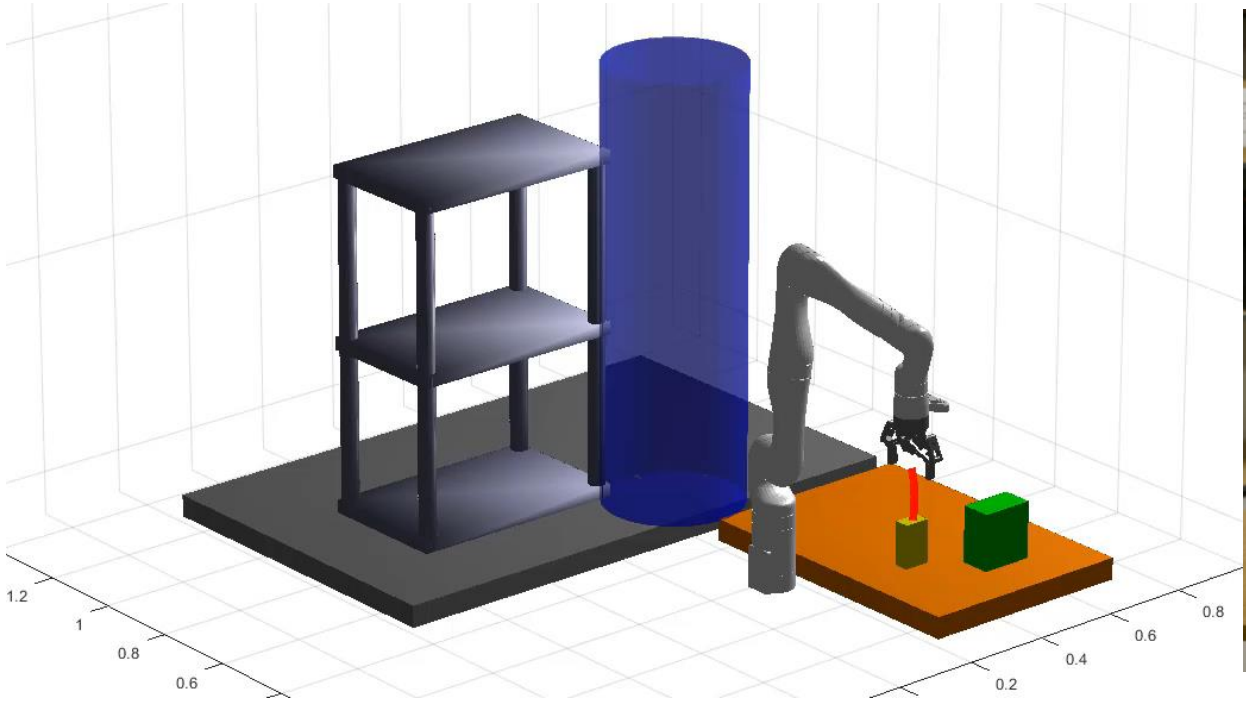
Plan & Decide

Control



Use the same reference workflow

For warehouse pick-and-place (storage shelf)

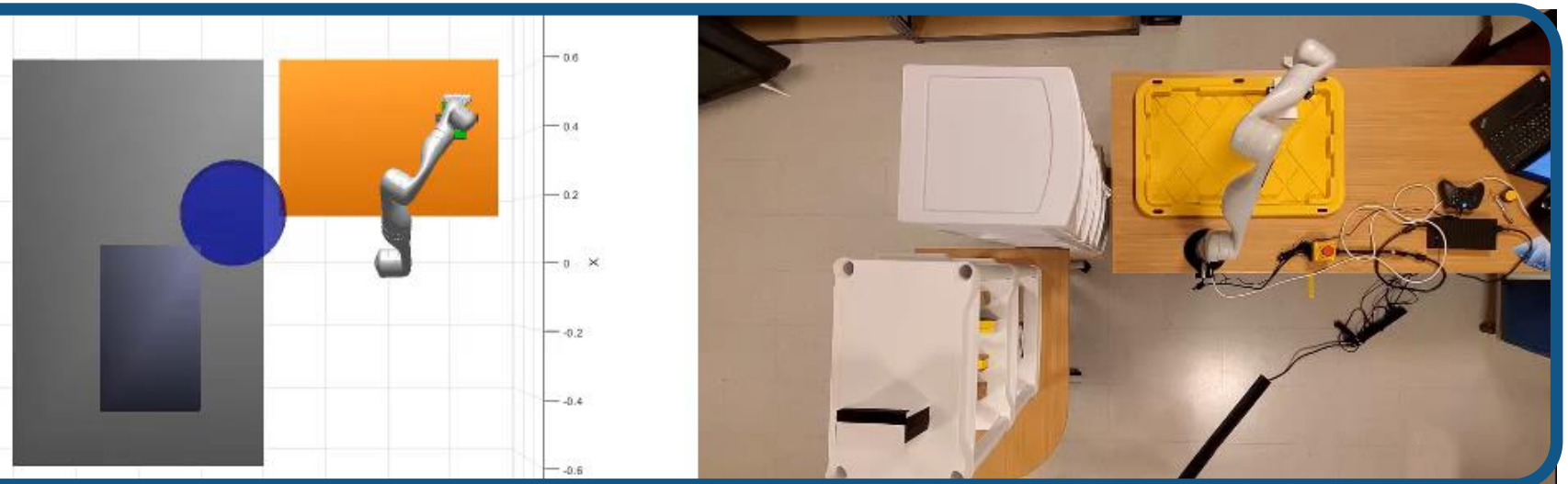


This workflow example highlights the use of Robotics System Toolbox collision-checking algorithms, nonlinear MPC, and Stateflow for MATLAB

Use the same reference workflow

For warehouse pick-and-place (storage shelf)

With obstacle avoidance **ON** (obstacle shown in blue)

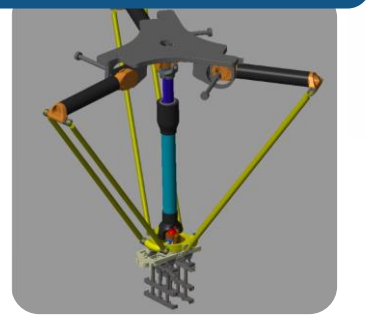


With obstacle avoidance **OFF** (obstacle shown in black for reference)

Use the same reference workflow

For Delta robot for automated parts sorting

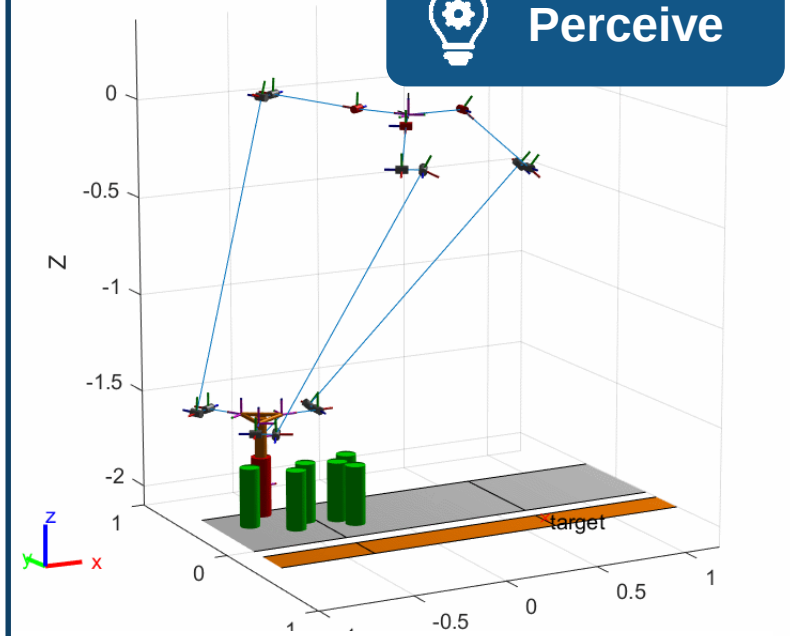
Platform



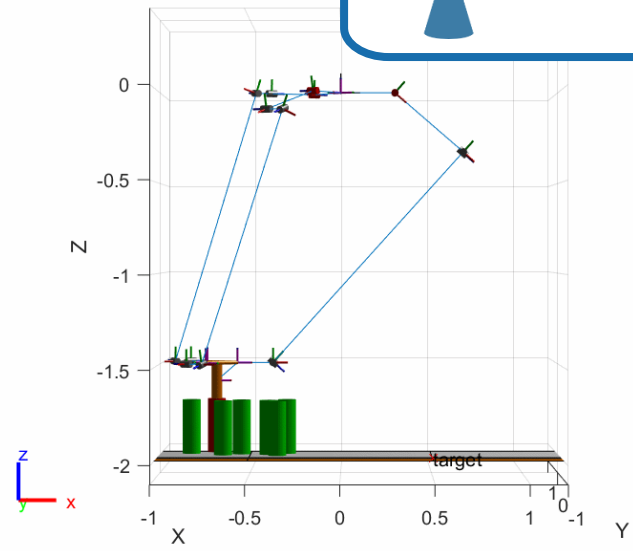
KRONES

Sense

Perceive

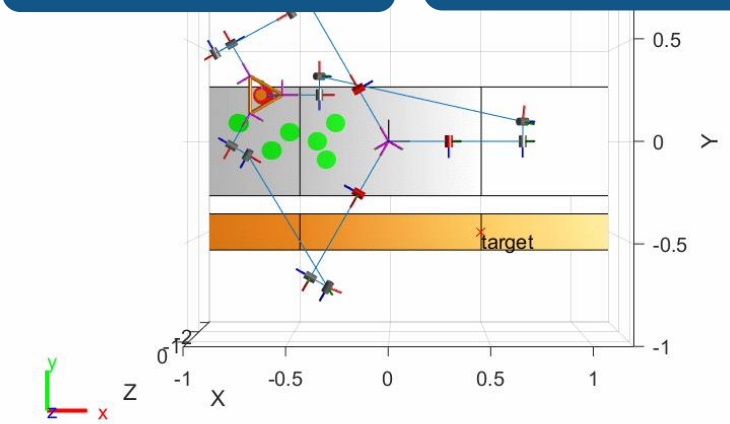


Connect



Plan & Decide

Control



What we'll discuss today

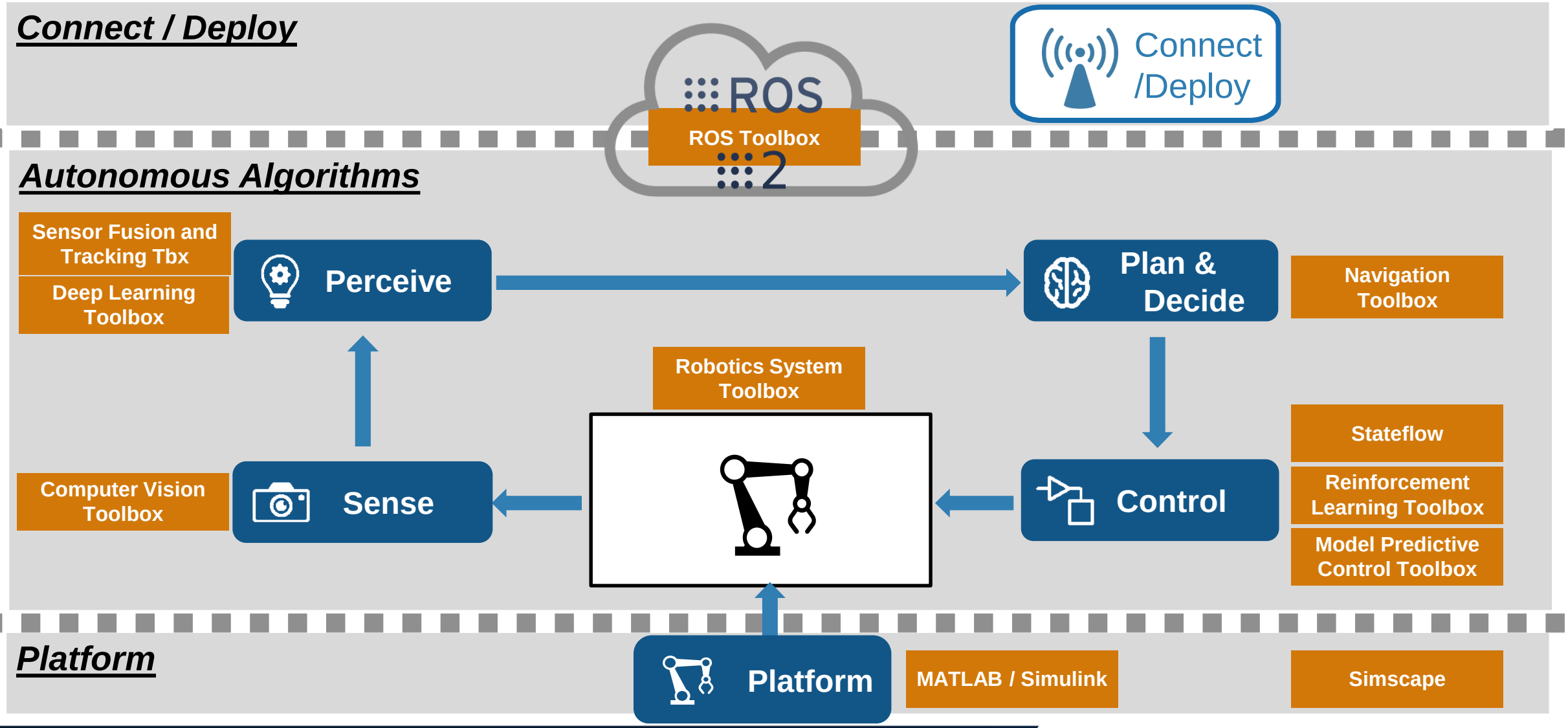
Challenges in Industry Robot System Design

Case Study: Pick-and-Place Manipulation Application

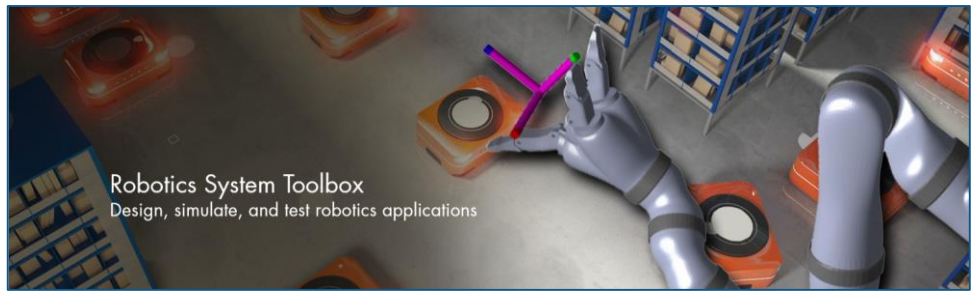
Model-Based Design for Autonomous System

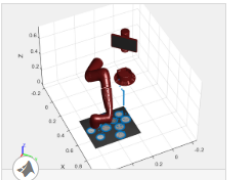
Concluding Remarks

Full Model-Based Design Workflow



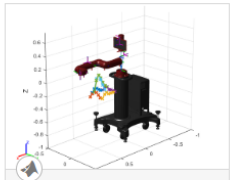
Resources to get started with





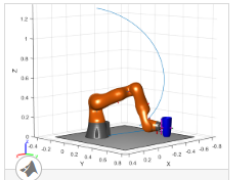
Trajectory Control Modeling With Inverse Kinematics

This Simulink example demonstrates how the Inverse Kinematics block can drive a manipulator along a specified



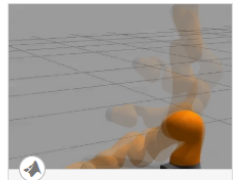
Manipulator Shape Tracing in MATLAB and Simulink

Trace a predefined 3-D shape in space. Following a smooth, distinct path is useful in many robotics applications such as welding,



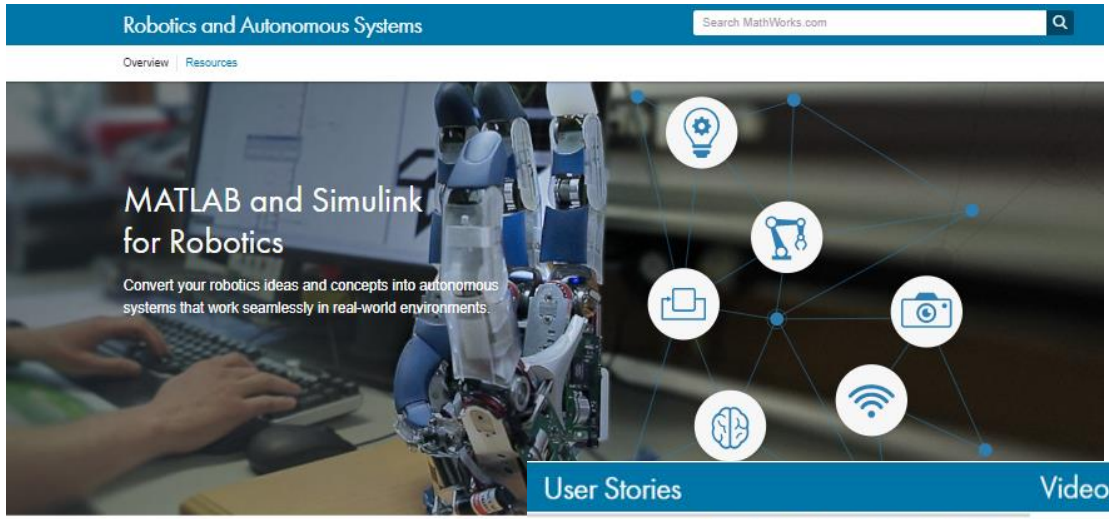
Plan a Reaching Trajectory With Multiple Kinematic Constraints

Use generalized inverse kinematics to plan a joint-space trajectory for a robotic manipulator. It combines multiple constraints to generate a



Control LBR Manipulator Motion Through Joint Torque Commands

Given a set of desired joint configuration waypoints and a torque-controlled manipulator, this example shows how to implement



User Stories

3T Develops Robot Emergency Braking Sys Based Design
3T modeled an emergency brake controller for verified the design, and generated defect-free implementation.

Clearpath Robotics Accelerates Algorithm I Industrial Robots
Clearpath Robotics shortens development time perception, computer vision, fleet managemen for industrial robots.

German Aerospace Center (DLR) Robotics Center Develops Autonomous Humanoid R Based Design
DLR developed advanced algorithms, generat and real-time operation, and automated senso armed robot.

HEBI Robotics Enables Rapid Development Algorithms for Robots Assembled from Sm
HEBI Robotics created a MATLAB based API accelerates the development of real-time contr powered by HEBI actuators.

Mitsubishi Heavy Industries Develops Robo Nuclear Fuel Debris
MHI used Model-Based Design to develop high software for a seven-meter, multi-axis robotic ar Daiichi nuclear power station clean-up effort.

Videos and Webinars

What Is Robotics System Toolbox?
Design, simulate, and test robotics applications.
Date: 13 Mar 2015

Path Planning and Navigation for Autonomous
Simplify the complex tasks of robotic path planning Simulink. This demonstration walks through how to using just three components: a path,...
Date: 31 Oct 2017

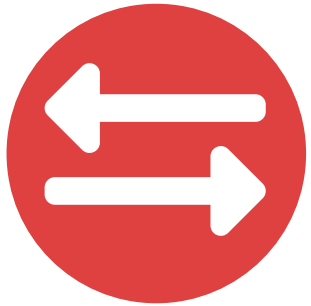
Simulink Blocks for Robot Manipulators and Sa
Use Simulink blocks to design, simulate, and imple and perform safe trajectory tracking control.
Date: 12 Jul 2018

Designing Robot Manipulator Algorithms
Accelerate the design of robot manipulator algorith Toolbox functionality and integrating robot models test manipulation tasks.

Robotics: Tools and Workflow
Professor Peter Corke describes why MATLAB and discusses using MathWorks tools for robo
Date: 1 Nov 2016

Control LBR Manipulator Motion Through Jo
Solve inverse and forward dynamics for RigidBo

Concluding Remarks



Challenges in industrial robot application development

- Multi-domain Expertise
- Complexity of Algorithms
- End-to-End workflows
- Technical Depth and System Stability

Develop Software with Model-Based Design

- Fast Iterations
- Strong Focus on Simulation

End-to-end workflow for industry robot applications development

- Platform
- Sense
- Perceive
- Plan & Decide
- Control

```
% Thank you!
```

mathworks.com/robotics