Designing Autonomous Robotic Systems with MATLAB and Simulink

Carlos Santacruz-Rosero, PhD
Sr Application Engineer – Robotics

Trudy Rippel
Account Management Team

Pulkit Kapur
Sr Industry Manager – Robotics
Robotics Applications using MATLAB and Simulink

- Autonomous Ground Robots
- Manipulator Arms
- Visual Odometry
- Advanced Perception and Planning
Some challenges….

- Applying Multidomain Expertise
- Complexity of Algorithms
- End-to-End workflows
- Technical Depth and System Stability
- IP Protection
Agenda

- Introduction
- Demo
- Conclusion
Ground Vehicle Co-simulation with Gazebo
Modeling: Import CAD and URDF
MATLAB, Simulink, and ROS

MATLAB Code

ROS Bag import

Networking

Code Generation

ROS Bag

Simulation environment

Robot

Built-in algorithms

SM Models

ROS node
Fusion & Tracking Algorithm Development Workflow

1. Create new scenario or refine sensor model
2. Collect lab data
3. Refine algorithm

Expected Behavior:
- C Code
- Generate code
- Integrate with embedded environment
Humanoid Simulation
Planning and Control for Ground Robots
Integrate with common robotics frameworks

• Import and filter ROS bags
• Connect MATLAB and Simulink to ROS
• Automatically generate C++ for ROS nodes

Clear separation between your algorithm and the middleware/hardware
Path Tracking for Unmanned Ground Vehicle
Pure Pursuit with UGV: Multi-domain simulation

Robotics Algorithm  Low-Level Control  Robot Simulation
Underwater Vehicle Co-simulation with UWsim
UAV Co-simulation with Gazebo
Integrate Robotics Algorithms
Apps: Ground Truth Labeler

Why?
1. Verifying Algorithms
2. Training Classifiers
Perception with MATLAB

“How do I label my data?”
New App for Ground Truth Labeling
Label pixels and regions for semantic segmentation

“How do I access the latest models?”
Caffe model importer
LSTM (time series, text)
DAG Networks
Library of pretrained models

“How do I make training and prediction faster?”
Multi-GPUs in parallel
Optimized GPU code
Training plots

“How do I deploy my new model?”
NEW PRODUCT:
GPU Coder-
Convert to NVIDIA CUDA code

Data
Models
Train / Predict
Deploy / Share
Key Takeaways

1. Productivity increased through an interactive environment to prototype, develop, debug, and implement algorithms in robotics.

2. Model-Based Design for motion control systems allows early verification of requirements, production code generation, and continuous testing.
Takeaways

- Workflows in robotics design and simulation, perception, planning, and controls.
  - ROS, ROS 2.0 and DDS support
  - Image Processing and Computer Vision
  - Machine Learning
  - Deep Learning
  - Path Planning
  - SLAM
  - Sensor Models and Sensor Fusion
  - FDA Software Validation
  - Ground truth labeling with LiDAR and Vision
  - 3D Simulation with Physics
  - Real-time support
  - Code-generation for C/C++/CUDA/HDL/PLC

- Meet us in San Jose during MATLAB EXPO (November 7th)