

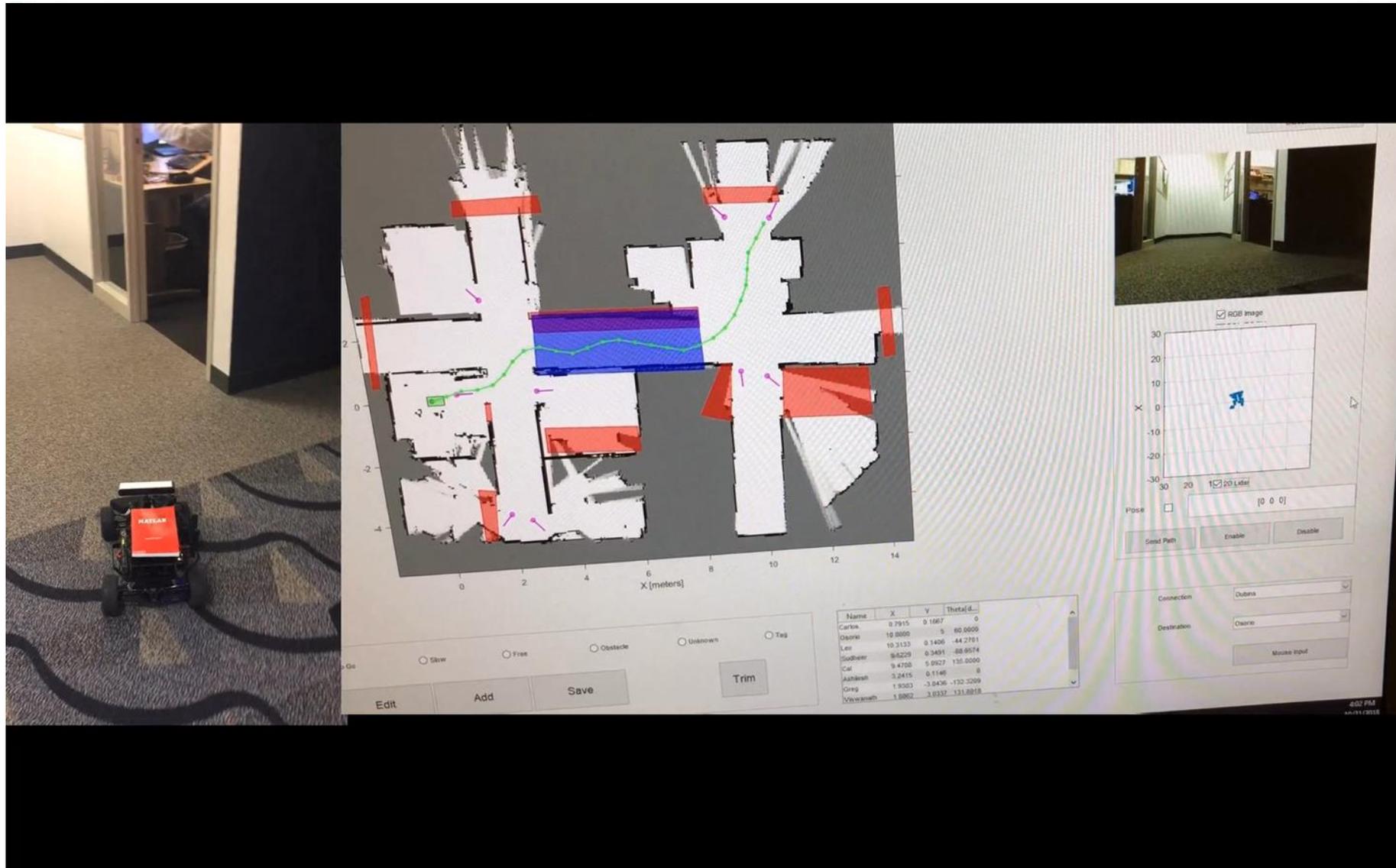
# MATLABおよびSimulinkによる自律移動ロボット開発

MathWorks Japan

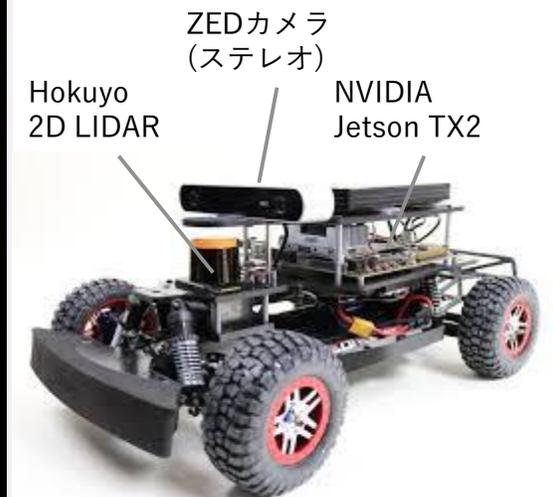
アプリケーションエンジニアリング部

木川田 亘

# 自律移動ロボットの構築例：自動搬送システム



## MIT RACECAR/J



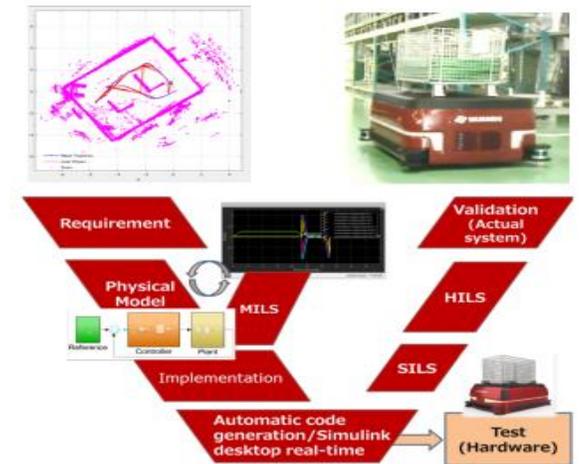
# 自律移動型ロボット開発のユーザ事例



Clearpath Roboticsによる  
産業用ロボット向け  
アルゴリズム開発の高速化  
Clearpath Robotics



自動運転タクシーの  
縦方向制御の開発  
Voyage Auto



工場内の自律搬送ロボットを  
半年間で試作  
武蔵精密工業株式会社

# 自律ロボット開発を加速化するアドオン

**System**

**Sense**

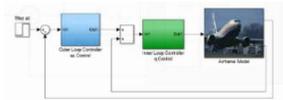
**Perceive**

**Decide & Plan**

**Control**

**Connect**

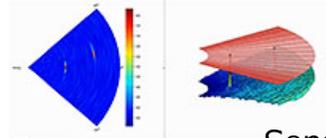
Control System Toolbox™



HW Support Packages



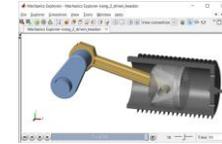
Phased Array System Toolbox™



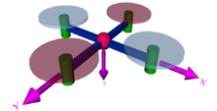
Data Acquisition Toolbox™



Simscape™



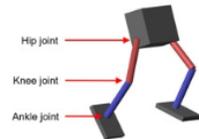
Aerospace Blockset™



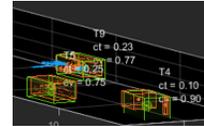
Computer Vision Toolbox™



Reinforcement Learning Toolbox™



Sensor Fusion and Tracking Toolbox™



Statistics and Machine Learning Toolbox™



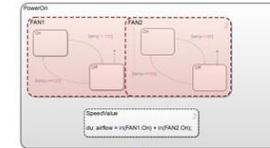
Deep Learning Toolbox™



Automated Driving Toolbox™



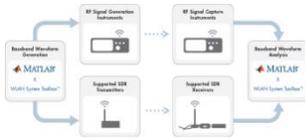
Stateflow®



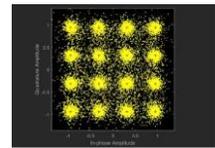
Model Predictive Control Toolbox™



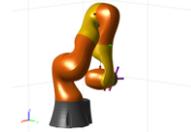
Communications Toolbox™



WLAN Toolbox™



Robotics System Toolbox™



Navigation Toolbox™



ROS Toolbox  
ROS 2

Simulink Real-Time™



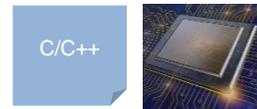
MATLAB Coder™



Simulink Coder™



Embedded Coder™



HDL Coder™



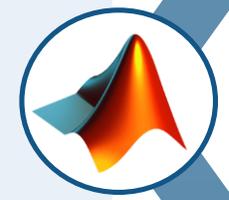
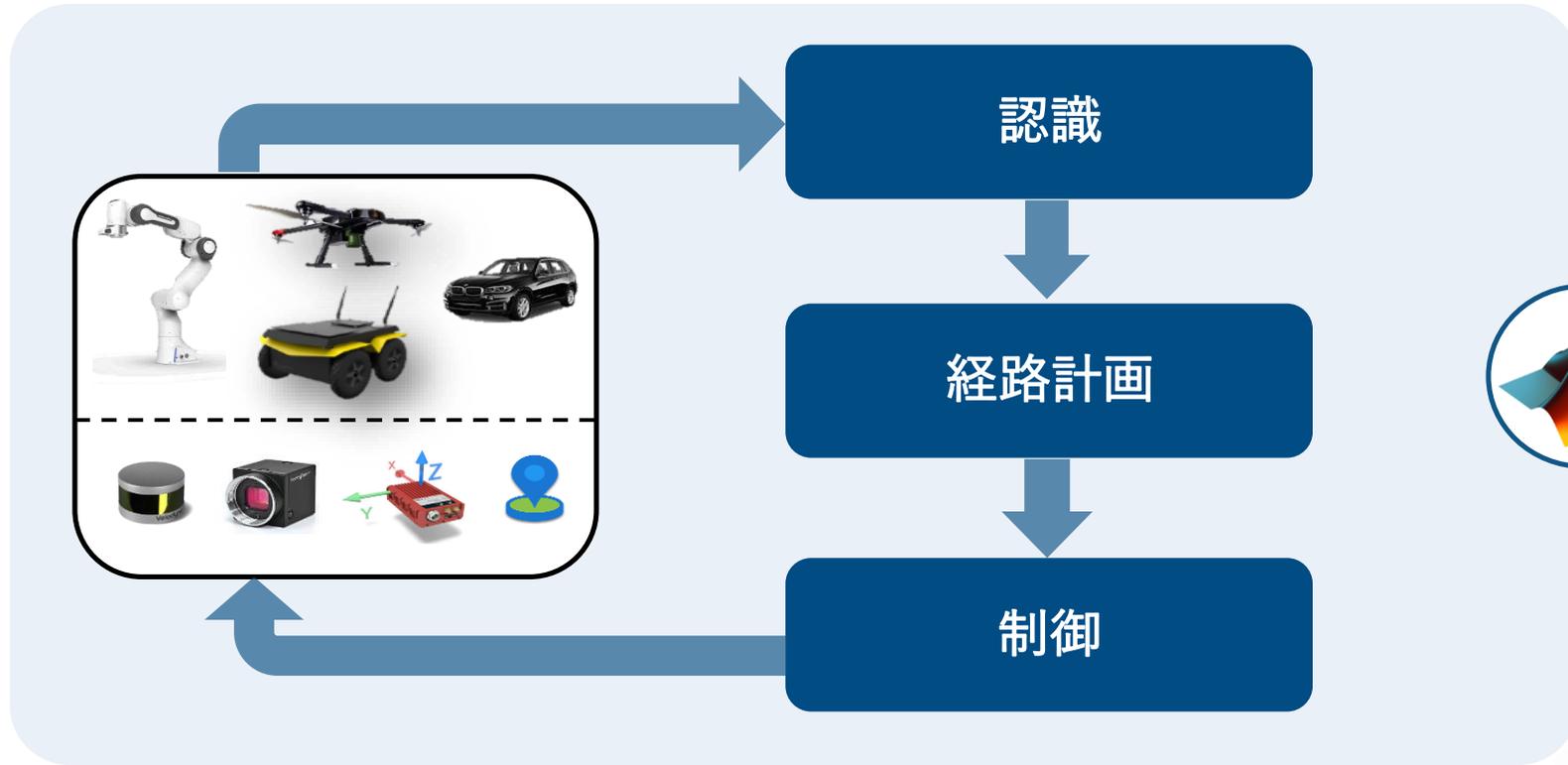
PLC Coder™



GPU Coder™



# 自律ロボティクスシステムの開発

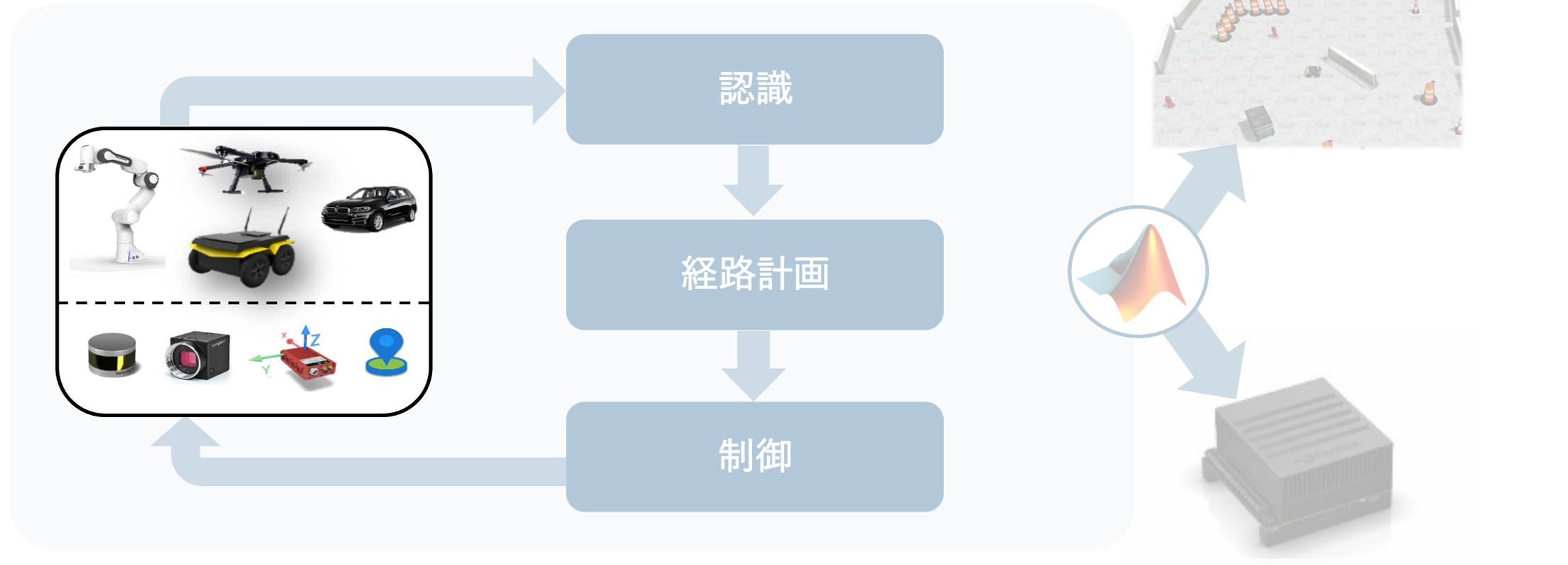


ロボット・センサーの  
モデリングと  
シミュレーション

自律アルゴリズムの  
設計

アプリケーションの  
テストと実装

# 自律ロボティクスシステムの開発



ロボット・センサーの  
モデリングと  
シミュレーション

自律アルゴリズムの  
設計

アプリケーションの  
テストと実装

# ロボット・センサーのモデリングとシミュレーション

## ロボットモデリング手法

高精度モデル

抽象モデル

外部ロボットモデル

タスクに最適な手法を選択

## センサーモデリング手法

センサーデータ取込

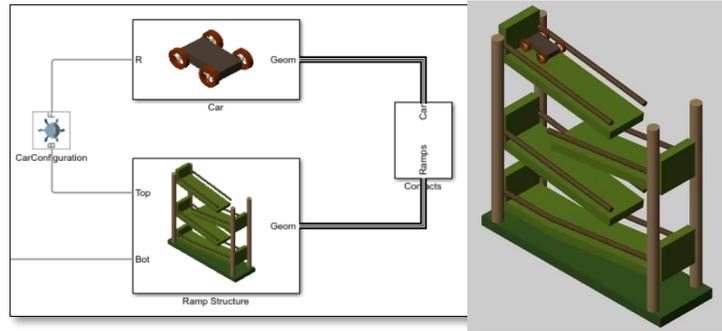
センサーシミュレーションモデル

外部センサーモデル

開ループ・閉ループワークフローを実現

# ロボットモデリング手法

## 動力学モデル

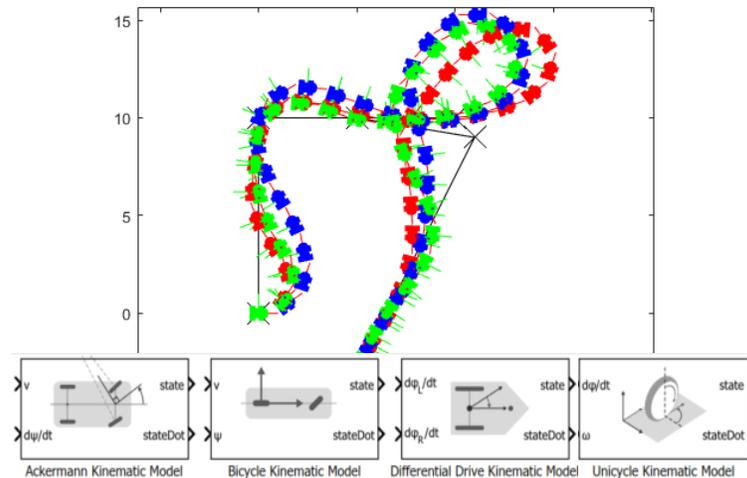


Spatial Contact Force  
ブロックの使用  
- バンパーカー

*Simscape Multibody™*

R2019b

## 運動学モデル

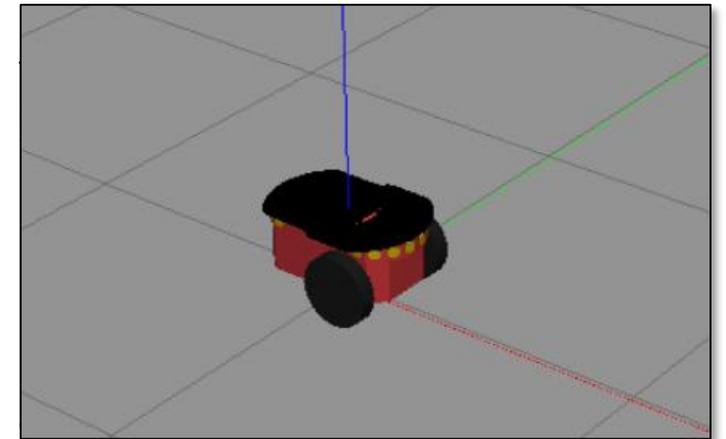


モバイルロボットの  
さまざまな運動学モデルの  
シミュレーション

*Robotics System Toolbox™*

R2019b

## Gazeboモデル



Simulinkを使用した  
Gazeboでの  
差動駆動型ロボットの制御

*Robotics System Toolbox™*

R2019b

# ロボット・センサーのモデリングとシミュレーション

## ロボットモデリング手法

高精度モデル

抽象モデル

外部ロボットモデル

タスクに最適な手法を選択

## センサーモデリング手法

センサーデータ取込

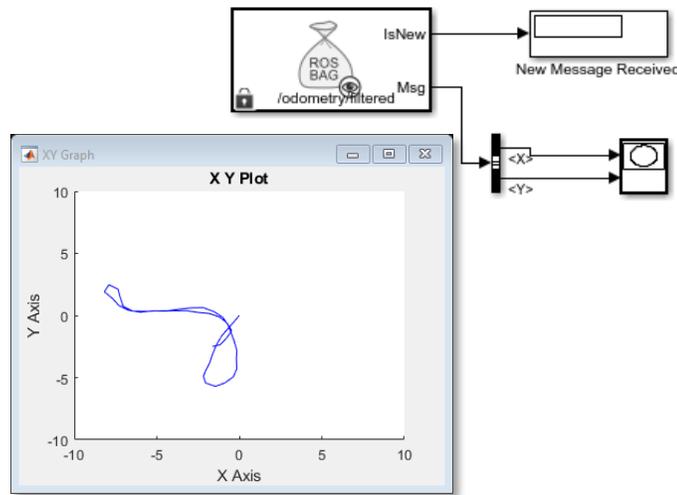
センサーシミュレーションモデル

外部センサーモデル

開ループ・閉ループワークフローを実現

# センサーモデリング手法

## rosvagのセンサーデータ インポート

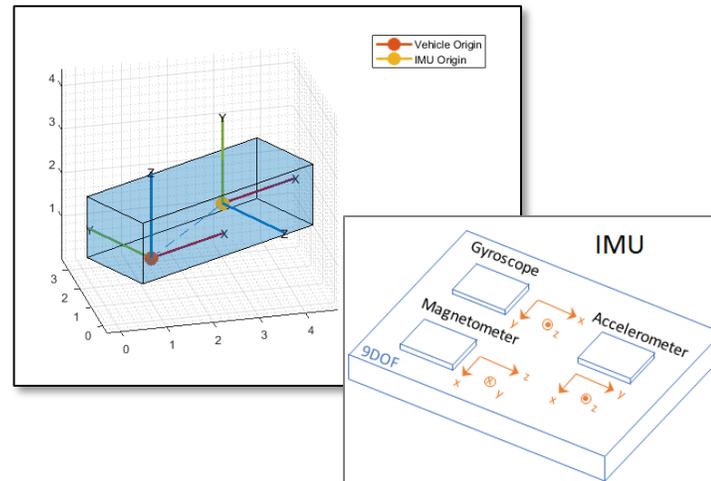


[Play Back Data from Jackal  
rosvag Logfile in Simulink](#)

*ROS Toolbox*

**R2020a**

## シミュレーション用モデル

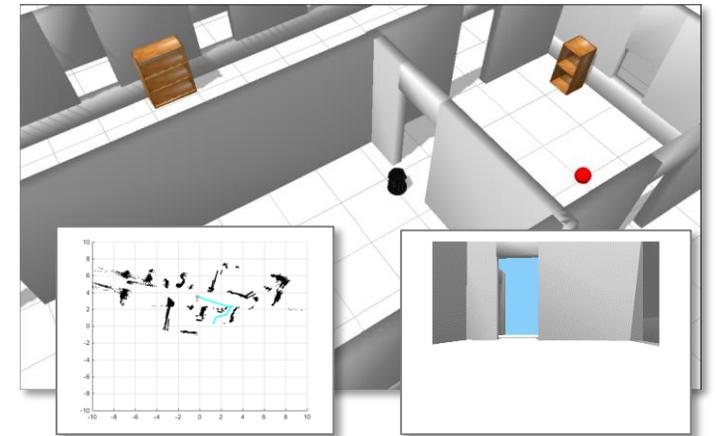


[Generate Off-centered IMU  
Readings](#)

*Navigation Toolbox™*

**R2020a**

## Gazeboセンサーモデル

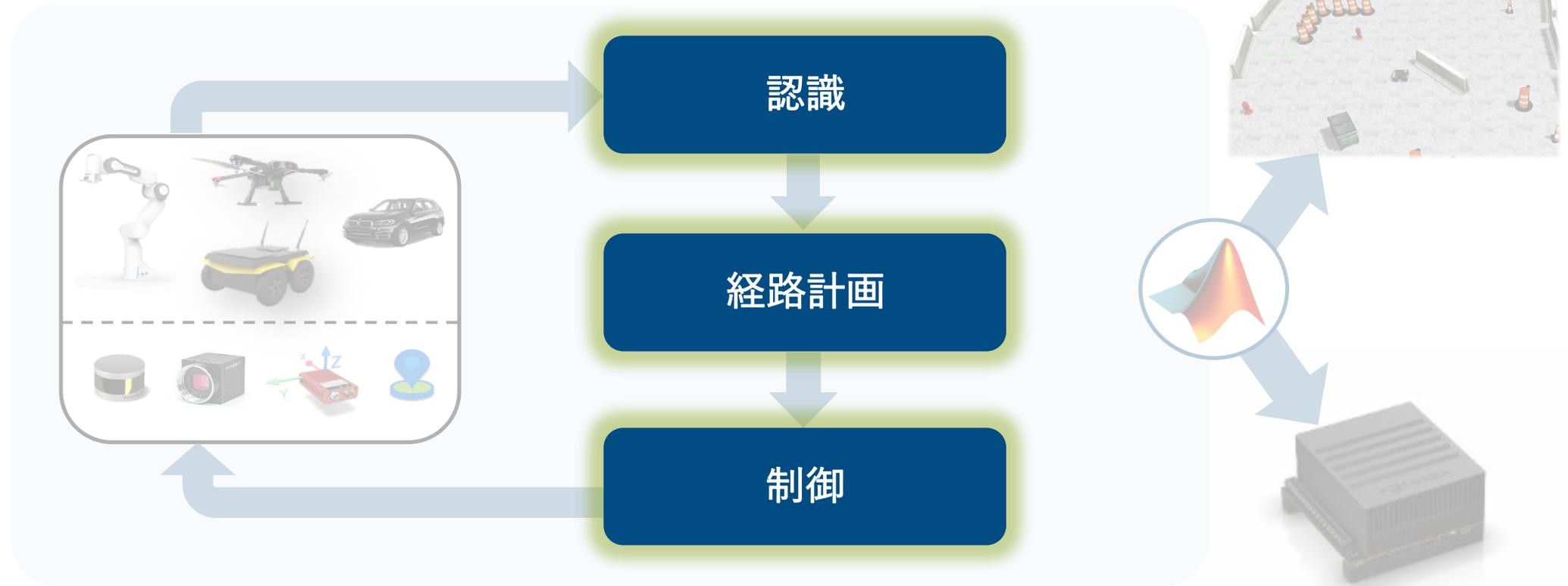


[Get Started with Gazebo and  
a Simulated TurtleBot](#)

*ROS Toolbox*

**R2019b**

# 自律ロボティクスシステムの開発



ロボット・センサーの  
モデリングと  
シミュレーション

自律アルゴリズムの  
設計

アプリケーションの  
テストと実装

# 自律アルゴリズムの設計

## 認識ワークフロー

データ収集

環境認知

自己位置推定

マッピング  
SLAM

## 経路計画・制御ワークフロー

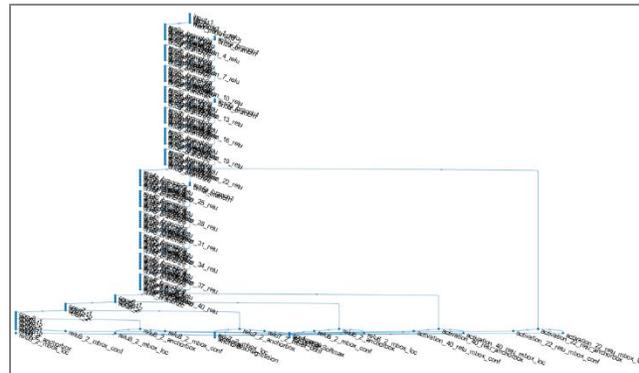
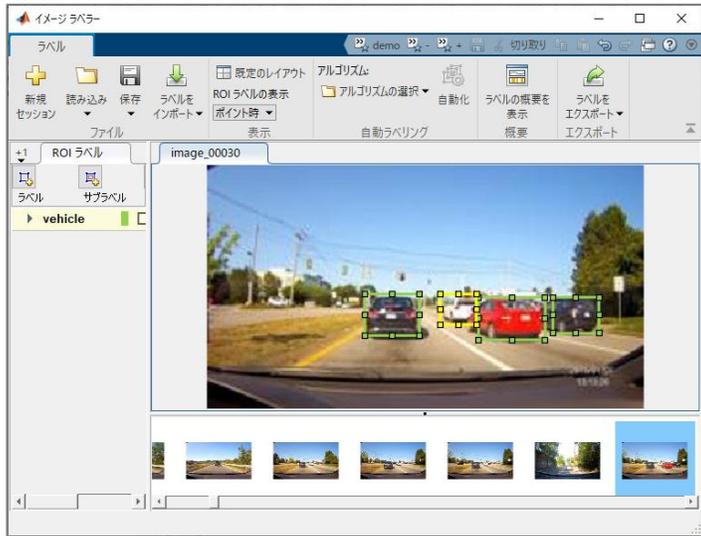
意思決定  
ロジック

モーション  
プランニング

パス追従

アドバンスト  
制御

# データ収集 & 環境認知



次のオブジェクト クラスに対する SSD オブジェクト検出器の学習を実行中:

```
* vehicle
```

最初の CPU で学習中。  
入力データの正規化を初期化しています。

エポック	反復	経過時間 (hh:mm:ss)	ミニバッチ損失	ミニバッチの精度	ミニバッチ RMSE	基準学習率
1	1	00:00:23	1.2314	48.73%	1.92	0.1000

- SSD (Single Shot Detector) ネットワークの作成
- データの拡張と前処理
- SSDのトレーニングと評価
- 自動コード生成

[Object Detection Using SSD Deep Learning](#)

*Computer Vision Toolbox™*

*Deep Learning Toolbox™*

*Parallel Computing Toolbox™*

**R2020a**

# 画像処理とディープラーニング

## 物体検出 (YOLO v3)

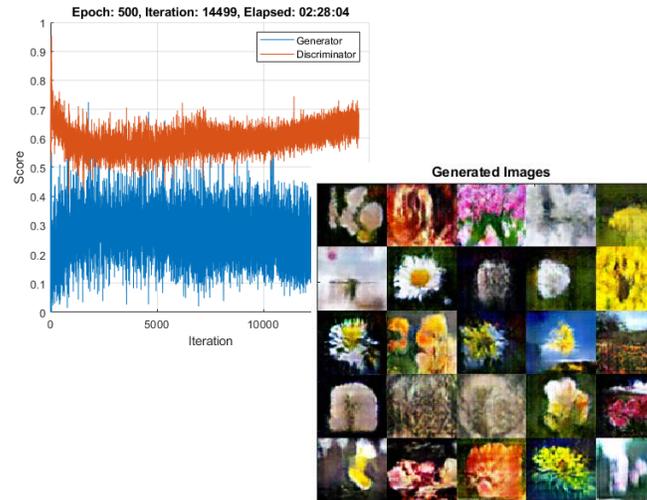


### Object Detection Using YOLO v3 Deep Learning

Computer Vision Toolbox™  
Deep Learning Toolbox™

R2020a

## GAN

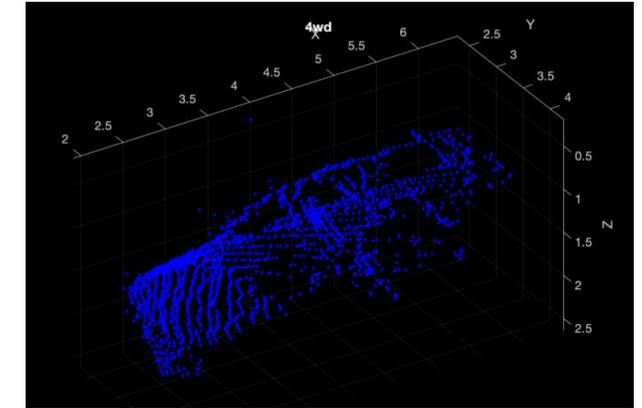


### Train Generative Adversarial Network (GAN)

Deep Learning Toolbox™  
Parallel Computing Toolbox™

R2019b

## 点群分類 (PointNet)



### Point Cloud Classification Using PointNet Deep Learning

Computer Vision Toolbox™  
Deep Learning Toolbox™

R2020a

# 検出した周辺物体の追尾

カメラによる  
周辺物体の追尾

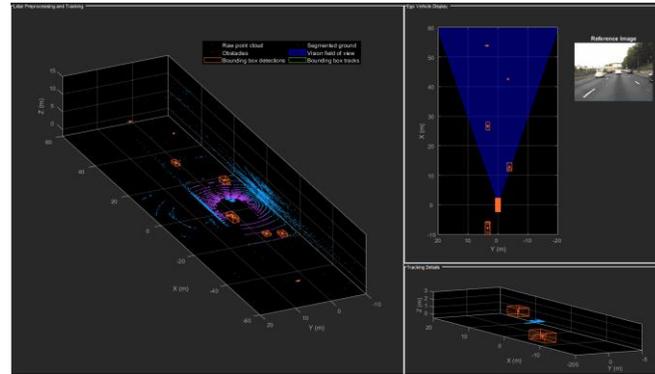


[Track Pedestrians from a Moving Car](#)

Automated Driving Toolbox™

R2020a

Lidarによる  
周辺物体の追尾



[Track Vehicles Using Lidar Data in Simulink](#)

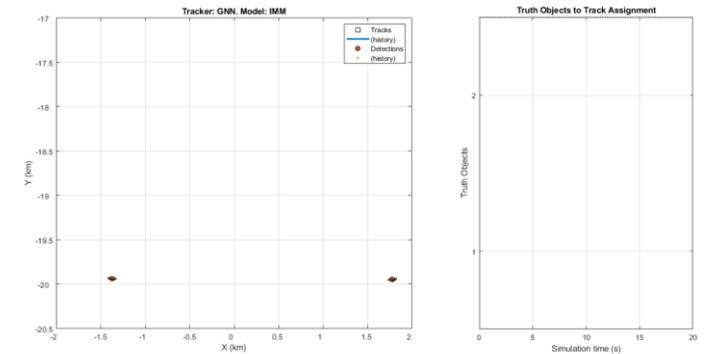
Automated Driving Toolbox™

Sensor Fusion and Tracking Toolbox™

Computer Vision Toolbox™

R2020a

隣接する物体の追尾



[Track Closely Spaced Targets Under Ambiguity in Simulink](#)

Simulink®

Sensor Fusion and Tracking Toolbox™

R2020a

# 自己位置推定 & マッピング・SLAM

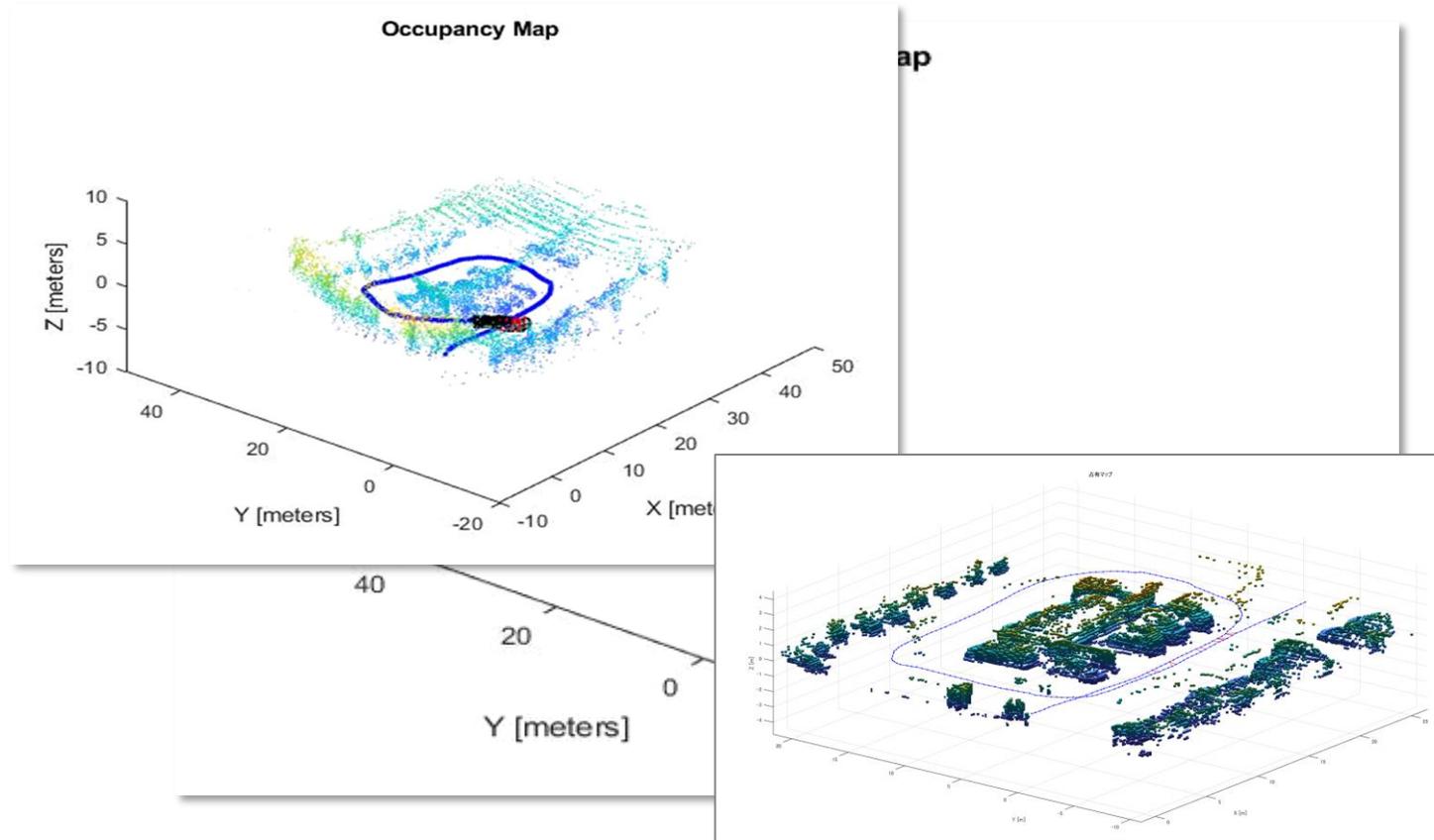
データ読込

同期

オフライン  
地図作成

SLAM  
自己位置推定

障害物回避



- ループ閉じ込みの推定
- ポーズグラフの最適化によるカメラ軌道の補正
- 3次元占有マップの作成と可視化

[Perform SLAM Using 3-D Lidar Point Clouds](#)

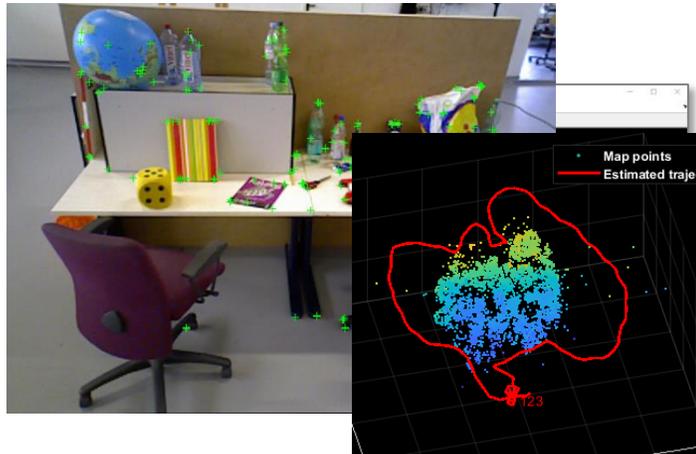
*Navigation Toolbox™*

*Computer Vision Toolbox™*

**R2019b**

# SLAMアルゴリズム設計

## 単眼カメラVisual SLAM (ORB-SLAM)

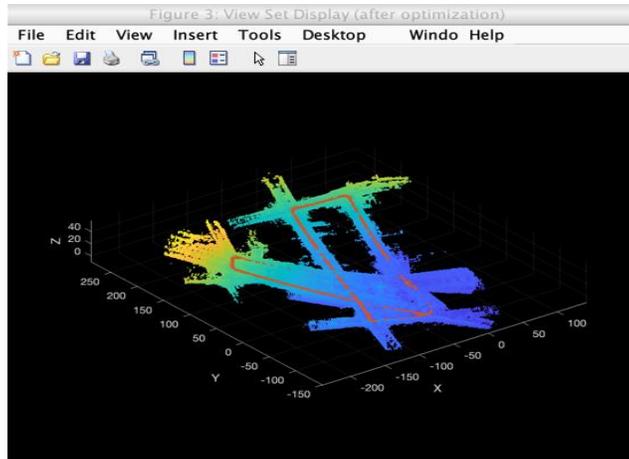


### Monocular Visual Simultaneous Localization and Mapping

Computer Vision Toolbox™

R2020a

## Lidar SLAM (実データ)

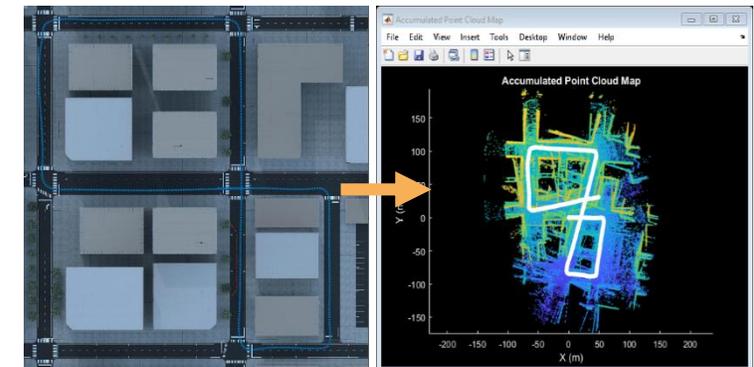


### Build a Map from Lidar Data Using SLAM

Navigation Toolbox™  
Automated Driving Toolbox™  
Computer Vision Toolbox™

R2020a

## Lidar SLAM (合成データ)



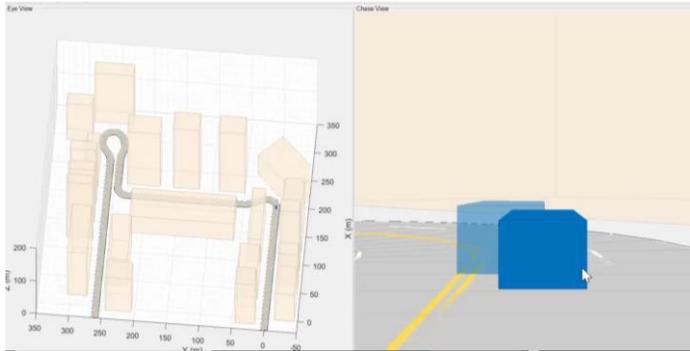
### Design Lidar SLAM Algorithm using 3D Simulation Environment

Navigation Toolbox™  
Automated Driving Toolbox™  
Computer Vision Toolbox™

R2020a

# センサーフュージョンによる位置・姿勢推定

## Vision + IMUによる位置・姿勢推定



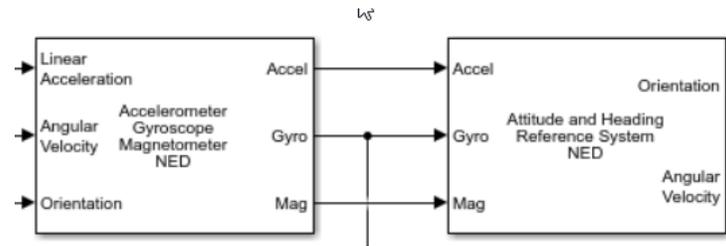
### Visual-Inertial Odometry Using Synthetic Data

*Sensor Fusion and Tracking Toolbox™*

*Automated Driving Toolbox™*

**R2019b**

## IMUによる姿勢推定

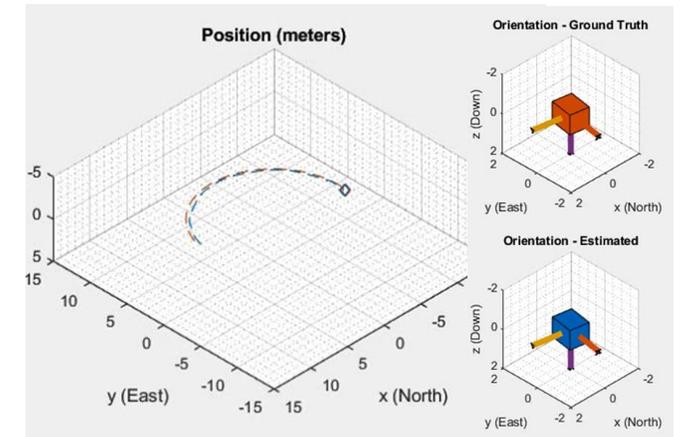


### IMU Sensor Fusion with Simulink

*Sensor Fusion and Tracking Toolbox™*

**R2020a**

## IMU + GPSによる位置・姿勢推定



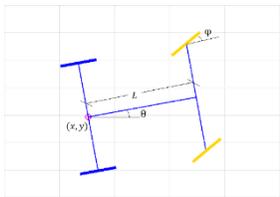
### Estimate Position and Orientation of a Ground Vehicle

*Sensor Fusion and Tracking Toolbox™*

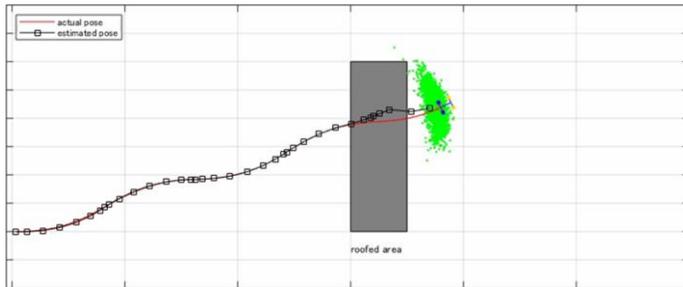
**R2019b**

# その他の位置・姿勢推定

## 粒子フィルターによる位置・姿勢推定



$$\begin{aligned} \dot{x} &= v \cos(\theta) \\ \dot{y} &= v \sin(\theta) \\ \dot{\theta} &= \frac{v}{L} \tan \phi \\ \dot{\phi} &= \omega \end{aligned}$$

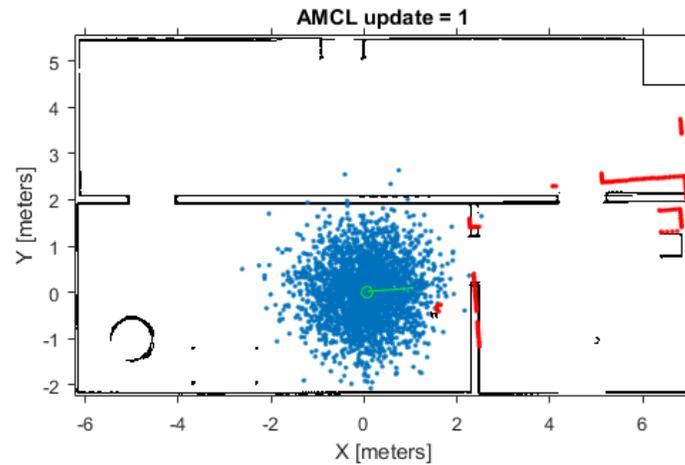


粒子フィルターを使用した自動車型ロボットの追跡

Robotics System Toolbox™

R2017a

## Lidarと粒子フィルターによる位置・姿勢推定

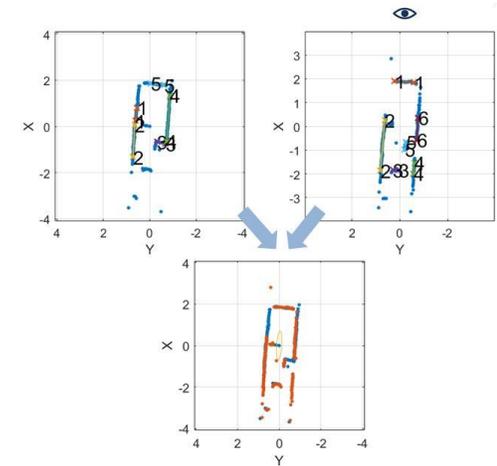


Localize TurtleBot Using Monte Carlo Localization

Navigation Toolbox™

R2019b

## ライン特徴量による位置・姿勢推定



Estimate Pose of Scans with Line Features

Navigation Toolbox™

R2020a

# 行動計画・制御ワークフロー

## 認識ワークフロー

データ収集

環境認知

自己位置推定

マッピング  
SLAM

## 行動計画・制御ワークフロー

意思決定  
ロジック

モーション  
プランニング

パス追従

アドバンスト  
制御

# 行動計画・制御ワークフロー

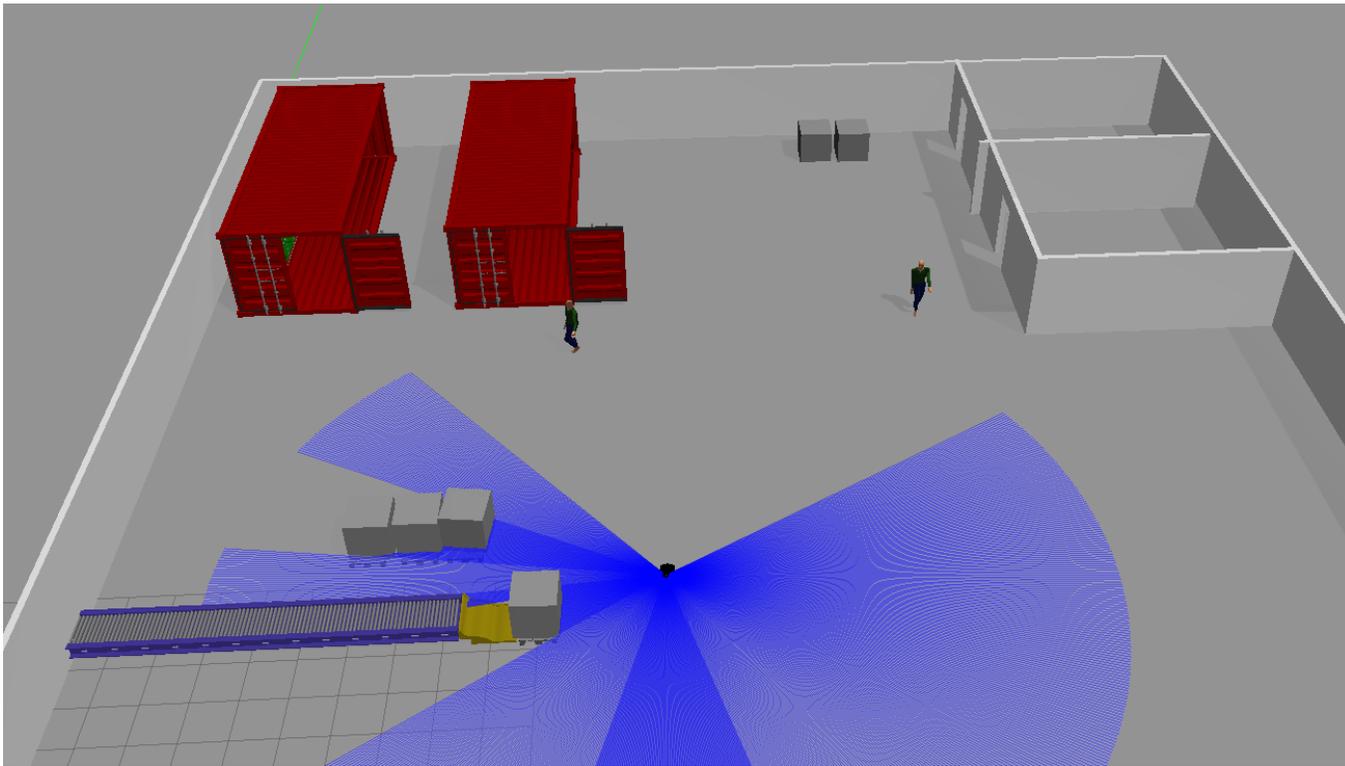
タスク  
スケジューリング

経路計画

軌道計画

経路追従

アドバンスド  
制御



- Gazeboセンサーの読み取り
- タスクスケジューリングと経路計画
- パス追跡と障害物回避
- ロボットのシミュレーション

[Simulate a Mobile Robot in a Warehouse Using Gazebo](#)

*Robotics System Toolbox™*

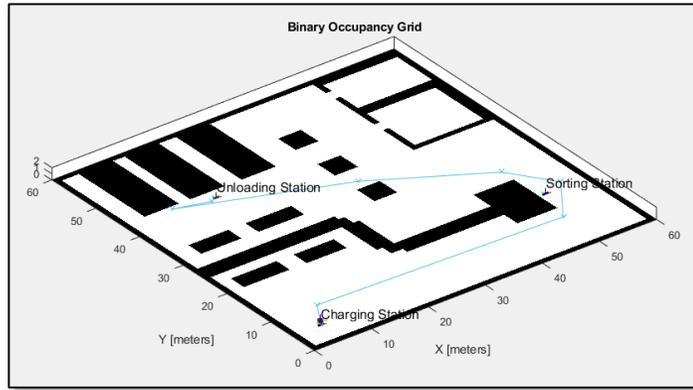
*Navigation Toolbox™*

*Stateflow®*

**R2020a**

# タスク・モーションプランニング

## タスクスケジューリング



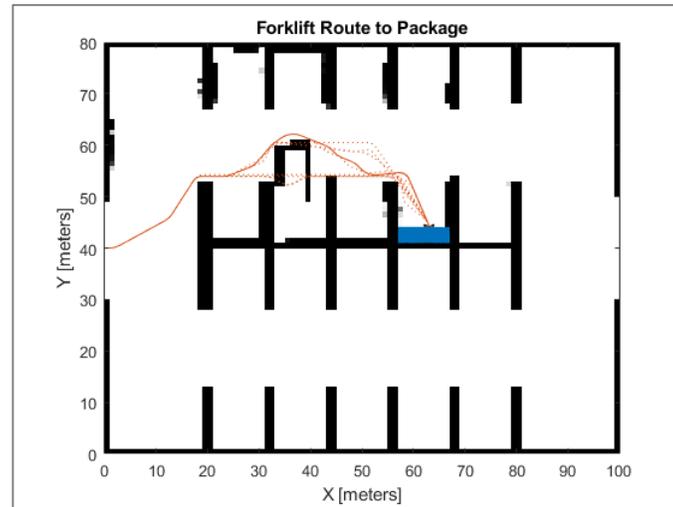
倉庫ロボットのタスクの実行

Robotics System Toolbox™

Stateflow®

R2019b

## ダイナミック経路計画

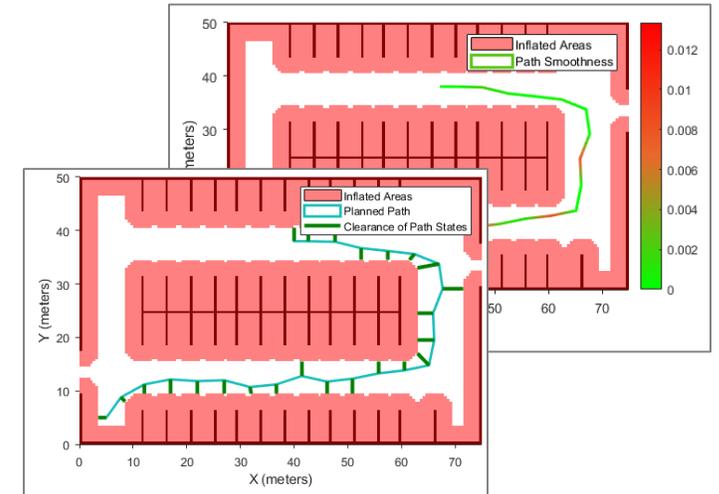


Dynamic Replanning on an  
Indoor Map

Navigation Toolbox™

R2019b

## パスメトリクス



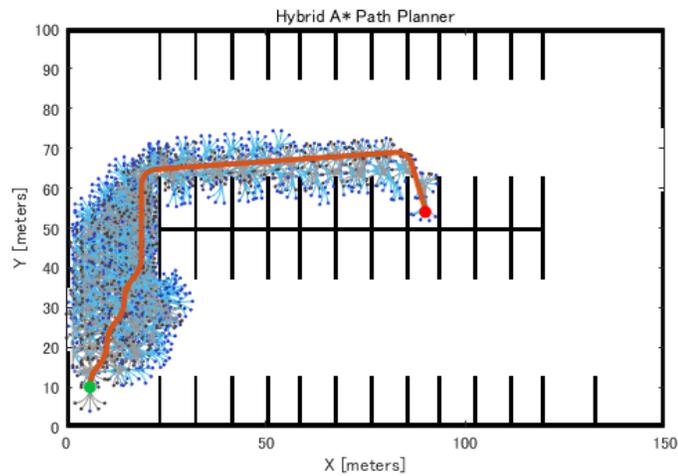
Evaluate Metrics for Planned  
Path

Navigation Toolbox™

R2020a  
Update

# モーションプランニング

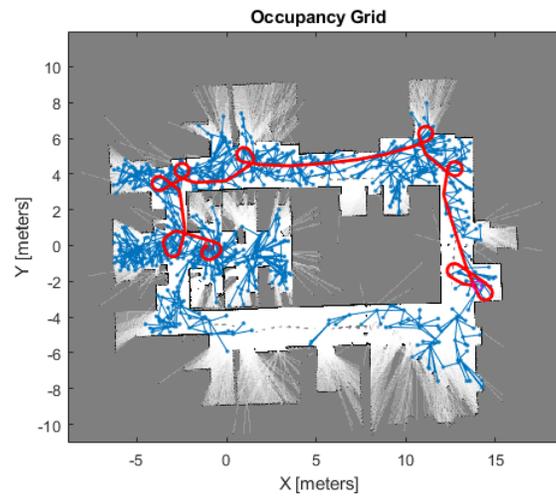
## Hybrid A\*



[plannerHybridAStar](#)  
Navigation Toolbox™

R2019b

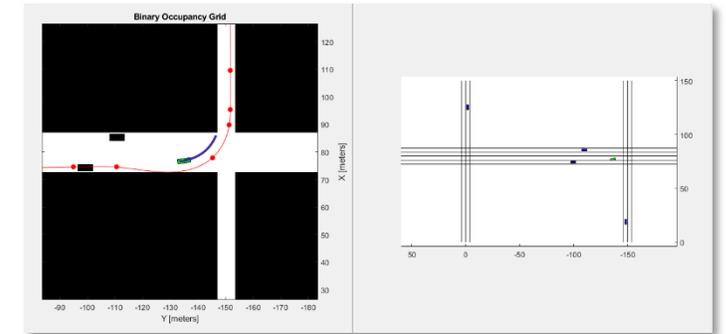
## RRT、RRT\*



[Plan Mobile-Robot Paths using RRT](#)  
Navigation Toolbox™

R2019b

## 最適化

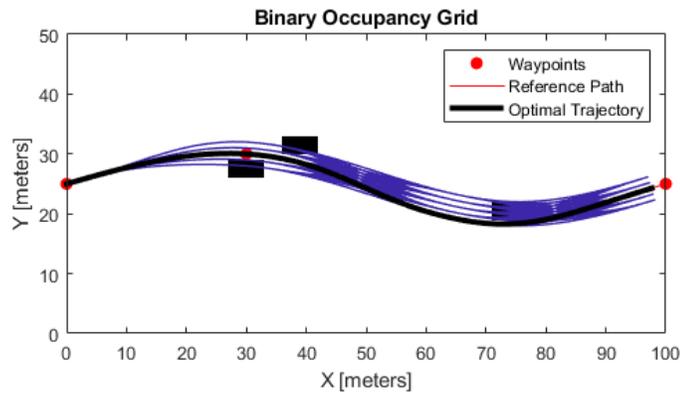


[Optimal Trajectory Generation for Urban Driving](#)  
Navigation Toolbox™  
Model Predictive Control Toolbox™

R2019b

# パス追跡とアドバンスト制御

## 最適化

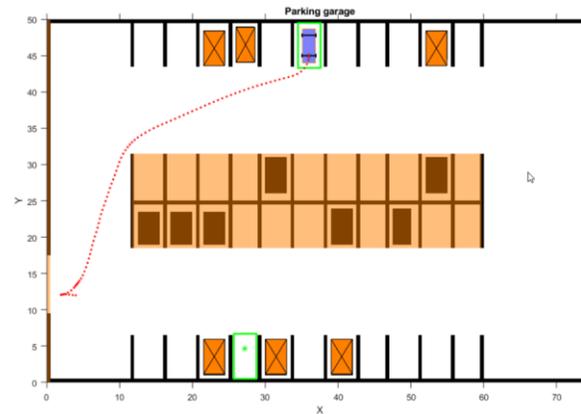


[Find optimal trajectory for reference path](#)

*Navigation Toolbox™*

**R2019b**

## 非線形MPC



[Parallel Valet using Nonlinear Model Predictive Control](#)

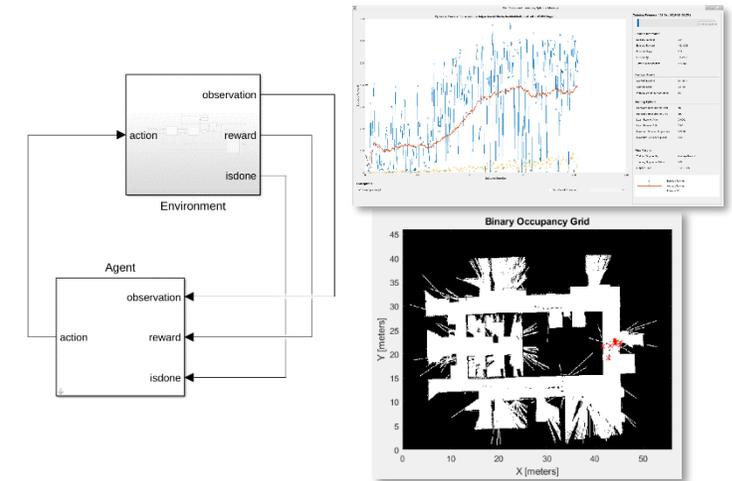
*Automated Driving Toolbox™*

*Model Predictive Control Toolbox™*

*Navigation Toolbox™*

**R2020a**

## 強化学習



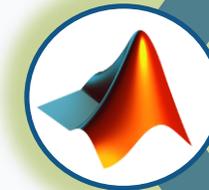
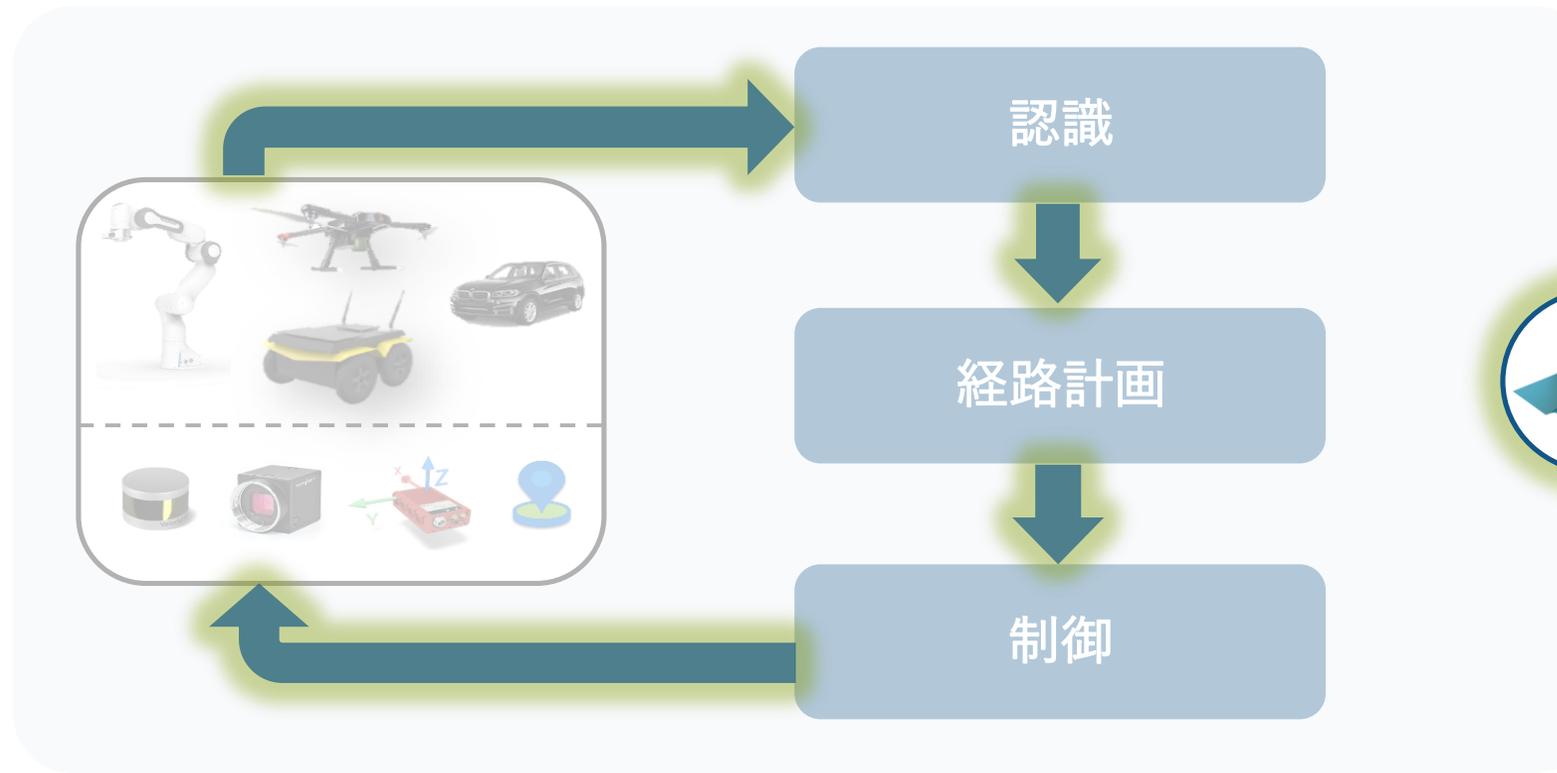
[Avoid Obstacles using Reinforcement Learning for Mobile Robots](#)

*Robotics System Toolbox™*

*Reinforcement Learning Toolbox™*

**R2020a**

# 自律ロボティクスシステムの開発



ロボット・センサーの  
モデリングと  
シミュレーション

自律アルゴリズムの  
設計

アプリケーションの  
テストと実装

# アプリケーションのテストと実装

## テスト手法

MATLAB  
Simulink

外部物理  
シミュレータ

検証タスクの  
自動化

認証

## 実装手法

ROS/ROS2

MCU  
DSP

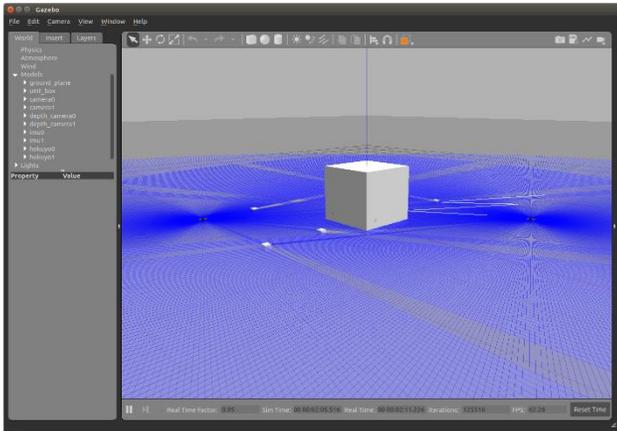
SoC

FPGA  
ASIC

GPU

# Gazebo / ROS / ROS2 連携

## Gazebo 同期シミュレーション

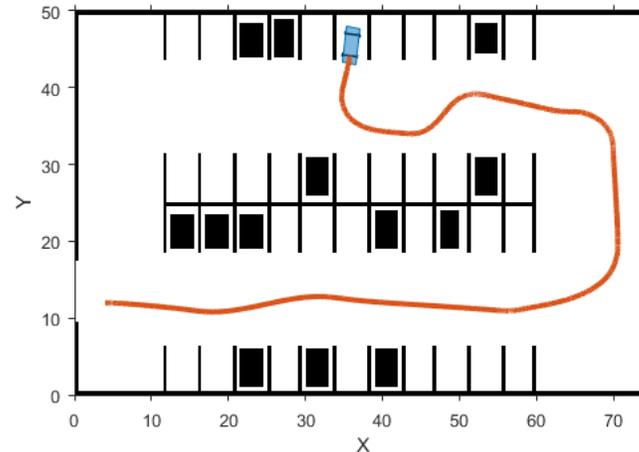


[Simulink と Gazebo 間の  
コシミュレーションの実行](#)

*Robotics System Toolbox™*

**R2019b**

## ROS / ROS2 連携

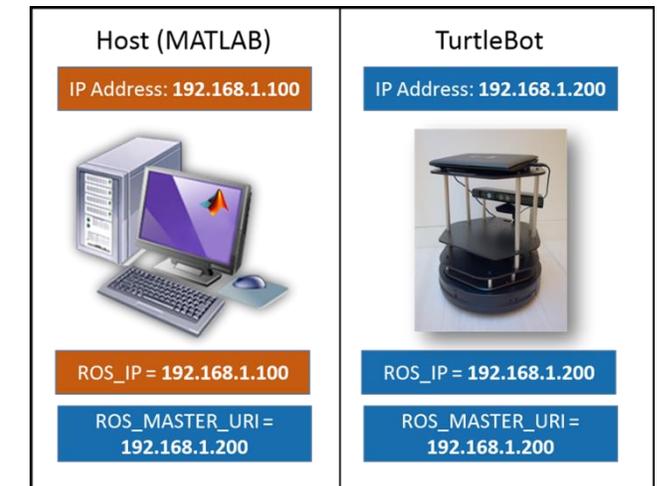


[Automated Parking Valet with  
ROS 2 in Simulink](#)

*Automated Driving Toolbox™*  
*ROS Toolbox*

**R2020a**

## ハードウェア接続

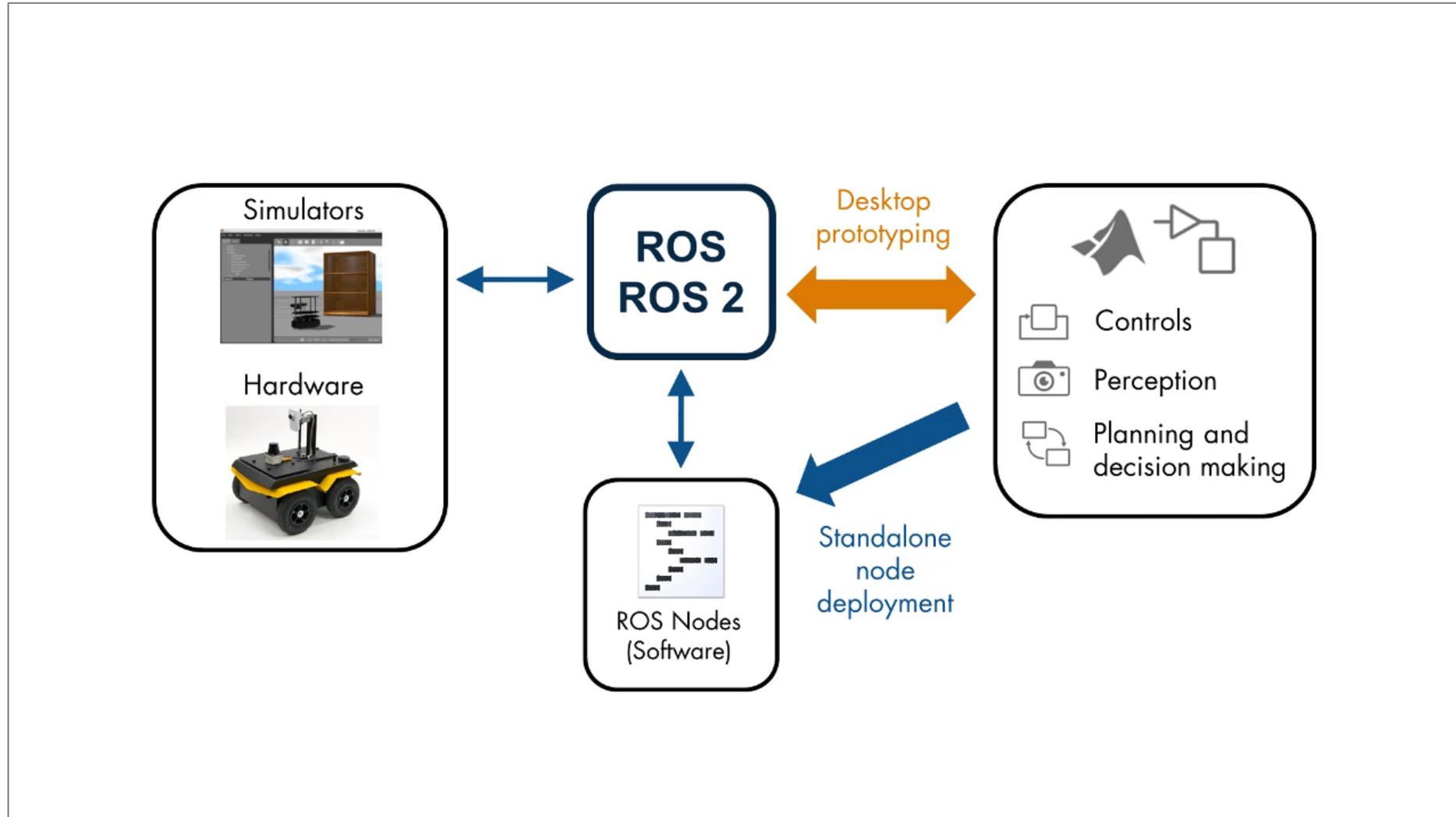


[Get Started with  
a Real TurtleBot](#)

*ROS Toolbox*

**R2019b**

## MATLAB/SimulinkのROS・ROS2連携機能



# アプリケーションのテストと実装

## テスト手法

MATLAB  
Simulink

外部物理  
シミュレータ

検証タスクの  
自動化

認証

## 実装手法

ROS/ROS2

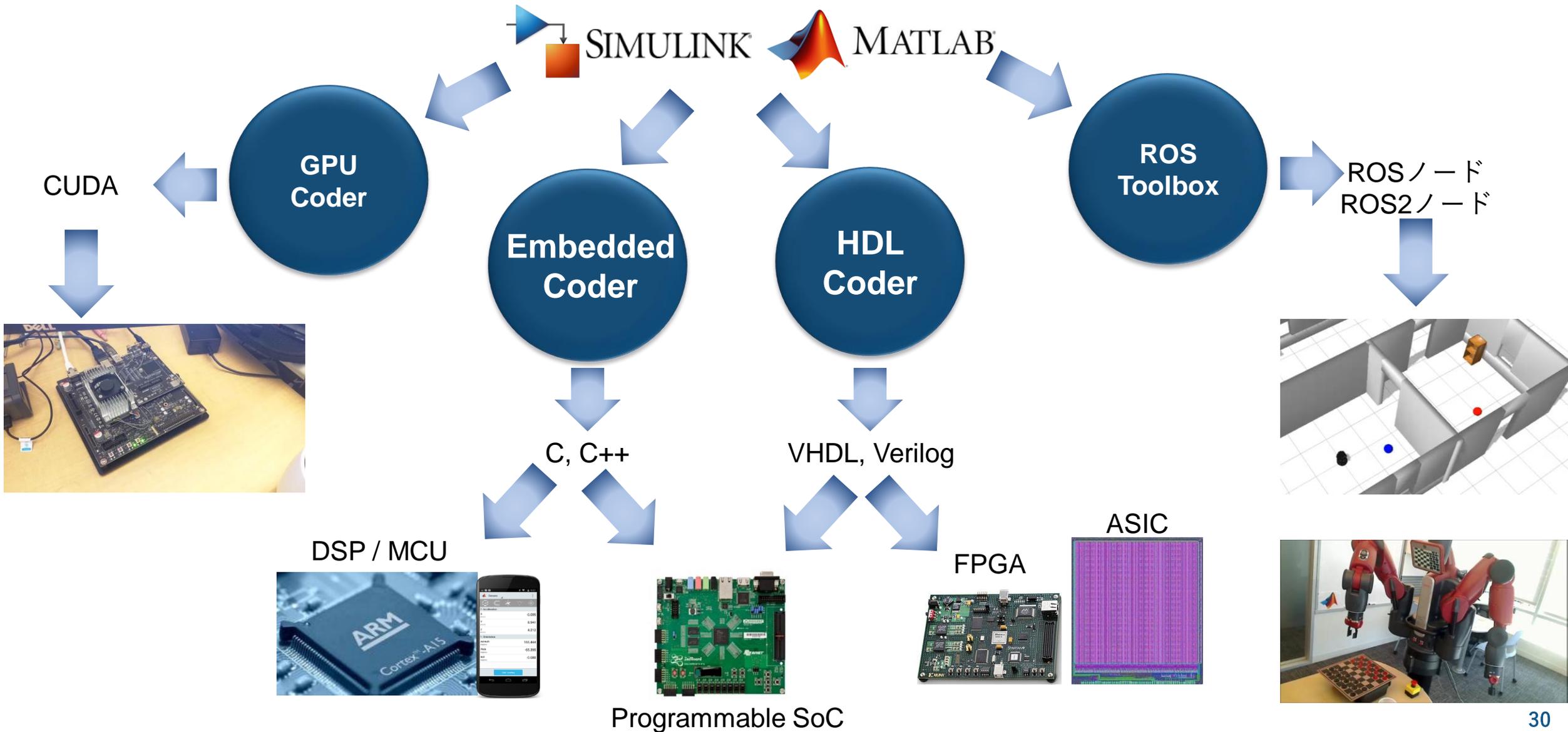
MCU  
DSP

SoC

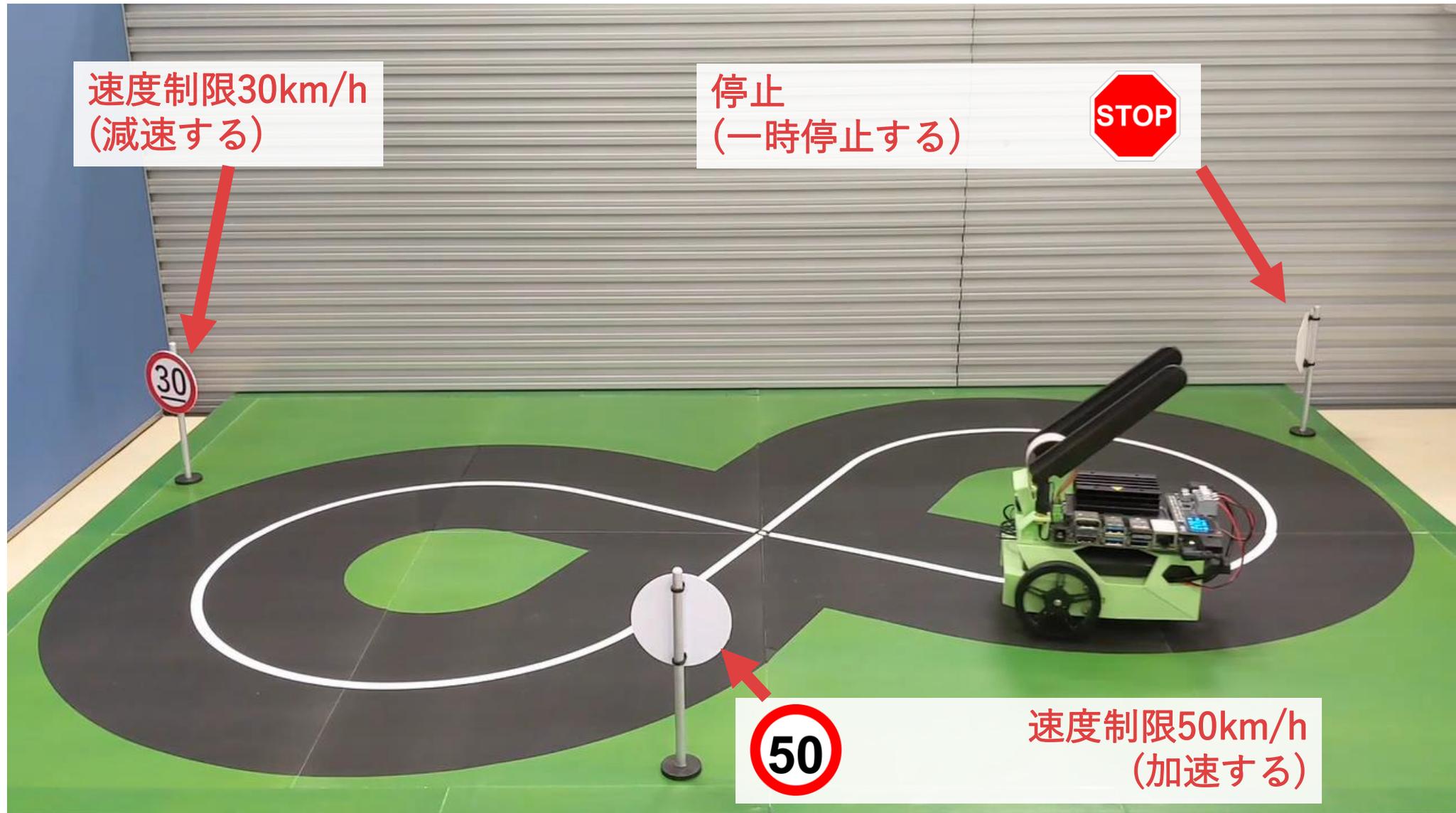
FPGA  
ASIC

GPU

# アルゴリズムのプロトタイピングと実装

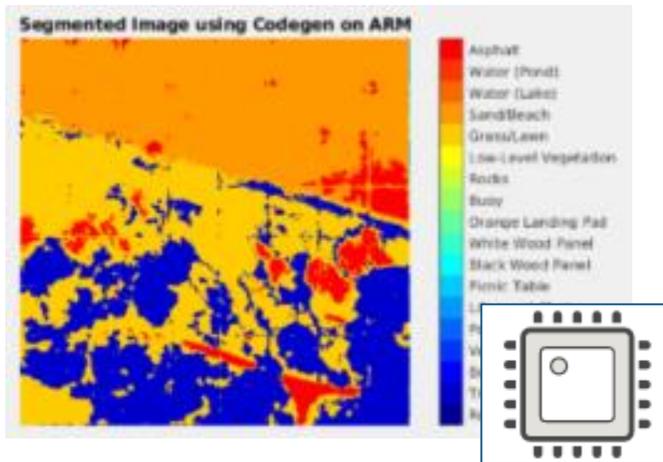


# GPU CoderによるGPUコード生成とROSノード実装



# 自動コード生成と実装

## ARM Neon コード生成

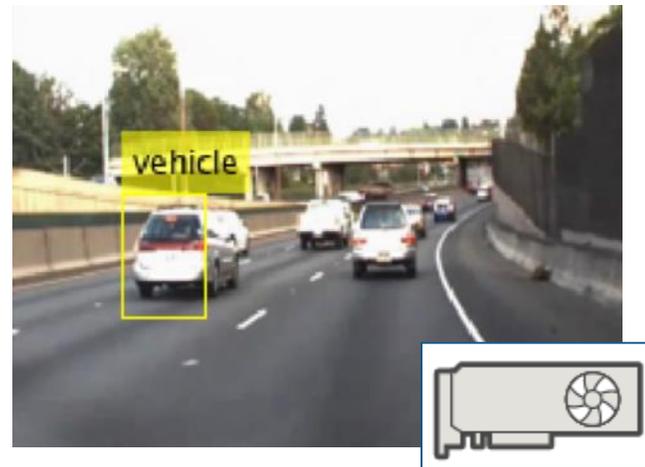


### Code Generation for Semantic Segmentation Application on ARM Neon

Deep Learning Toolbox™  
MATLAB Coder™

**R2020a**

## NVIDIA GPU CUDAコード生成

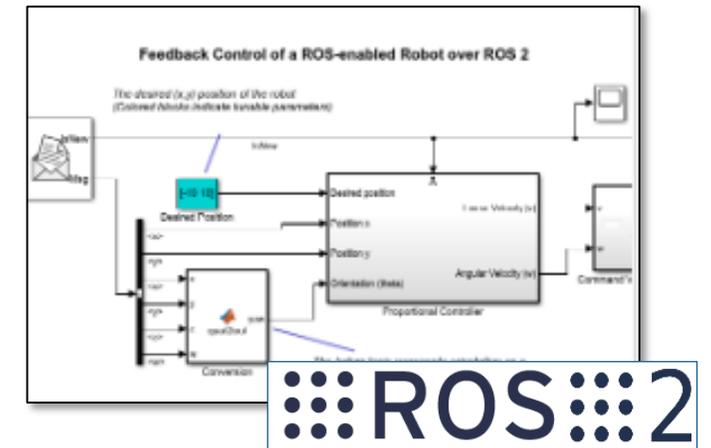


### Code Generation for Object Detection by Using Single Shot Multibox Detector

Deep Learning Toolbox™  
GPU Coder™

**R2020a**

## ROS / ROS2 ノード生成



### Generate a Standalone ROS 2 Node from Simulink®

ROS Toolbox  
Embedded Coder™

**R2020a**

# アプリケーションのテストと実装

## テスト手法

MATLAB  
Simulink

外部物理  
シミュレータ

検証タスクの  
自動化

認証

## 実装手法

ROS/ROS2

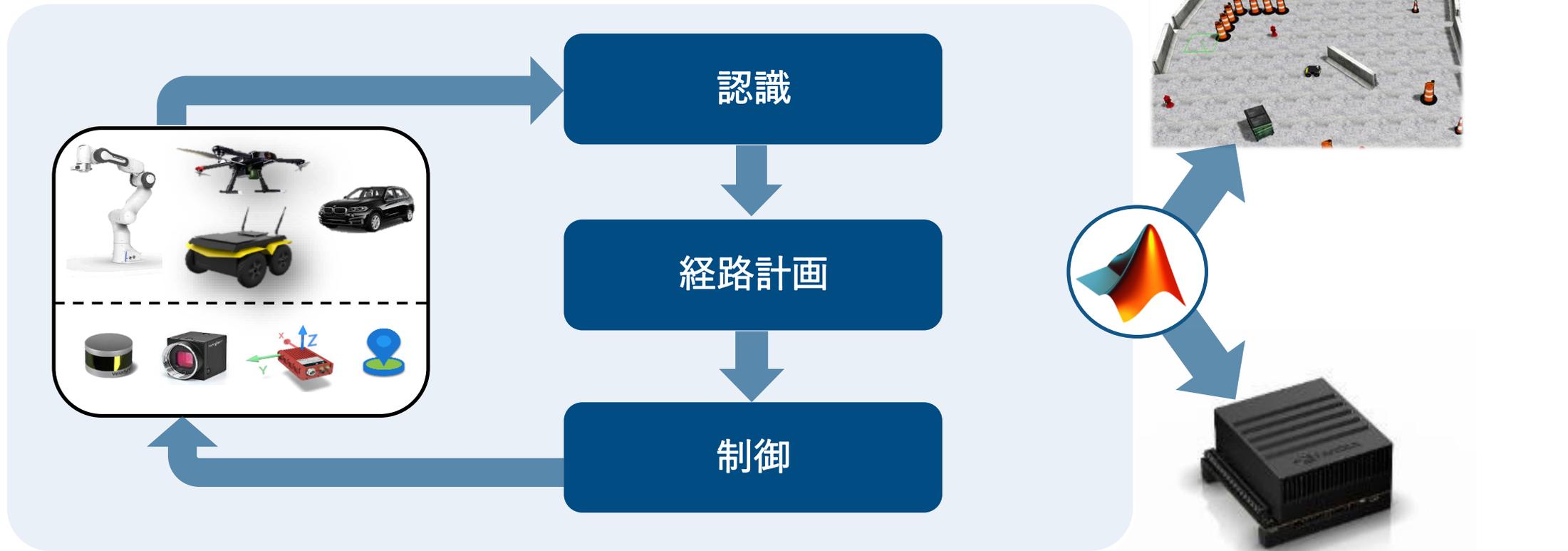
MCU  
DSP

SoC

FPGA  
ASIC

GPU

# まとめ：自律ロボティクスシステムの開発



ロボット・センサーの  
モデリングと  
シミュレーション

自律アルゴリズムの  
設計

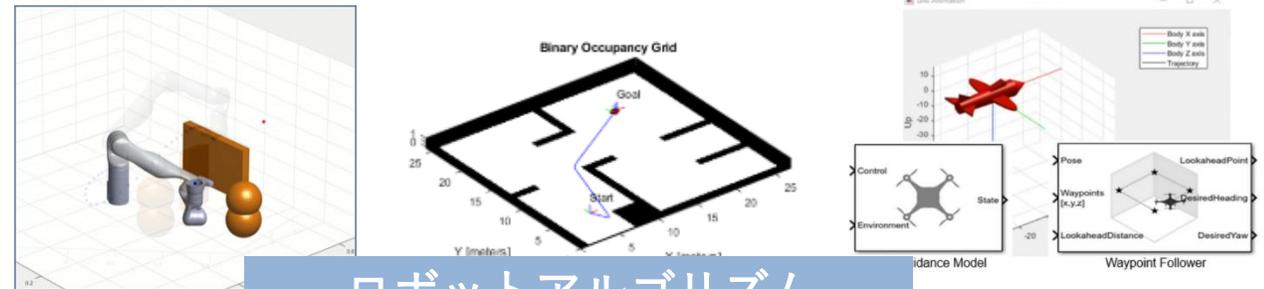
アプリケーションの  
テストと実装

# Robotics System Toolbox概要

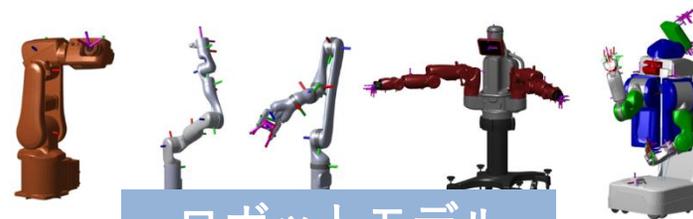
R2019b

ロボットアプリケーションの設計、シミュレーション、テスト

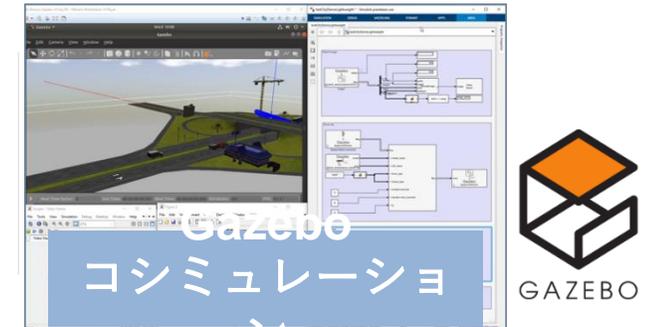
- ロボットアルゴリズム
  - ロボットアーム (逆運動学/軌道生成/衝突検知)
  - 移動ロボット (2D地図生成/自己位置推定)
  - UAV/ドローン (ガイダンス/経路追従/MavLink)
- 高抽象度ロボットモデル
  - 移動ロボット(運動学)
  - ロボットアーム(運動学/動力学)
- Gazebo時刻同期 協調シミュレーション
- ロボットモデル
  - 剛体ボディツリーモデル (RBT)
  - 商用ロボットのモデルライブラリ
  - モデル取り込み(URDF、DHパラメータ Simscape Multibody)
- ロボットアプリケーションのリファレンスサンプル
  - ロボットアームによるピック&プレイス
  - 倉庫内の複数台自律搬送



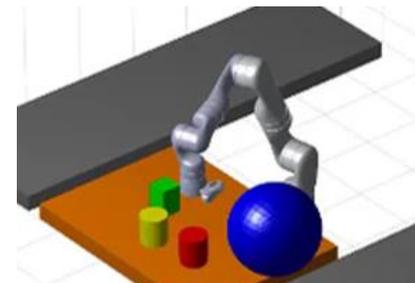
ロボットアルゴリズム



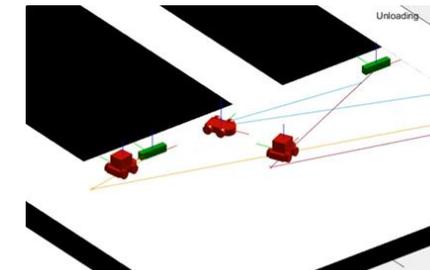
ロボットモデル



コシミュレーション



ピック&プレイス

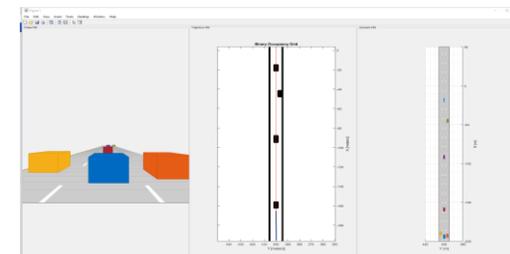
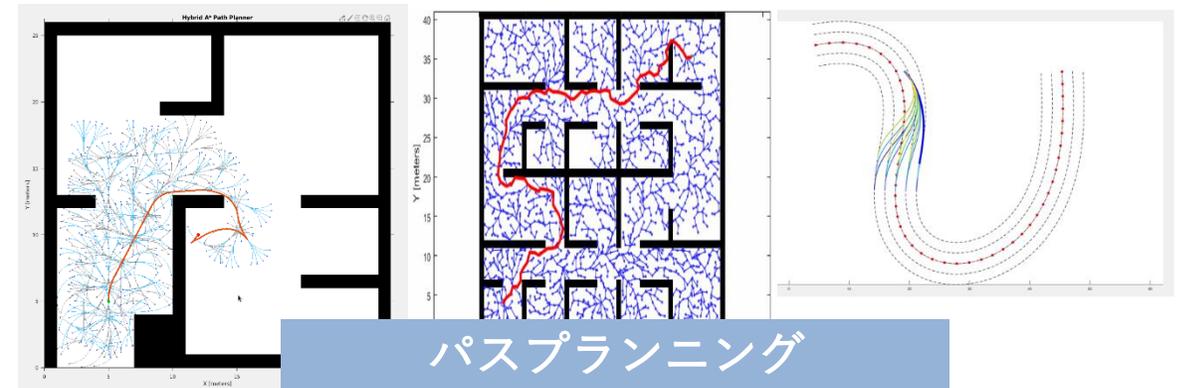
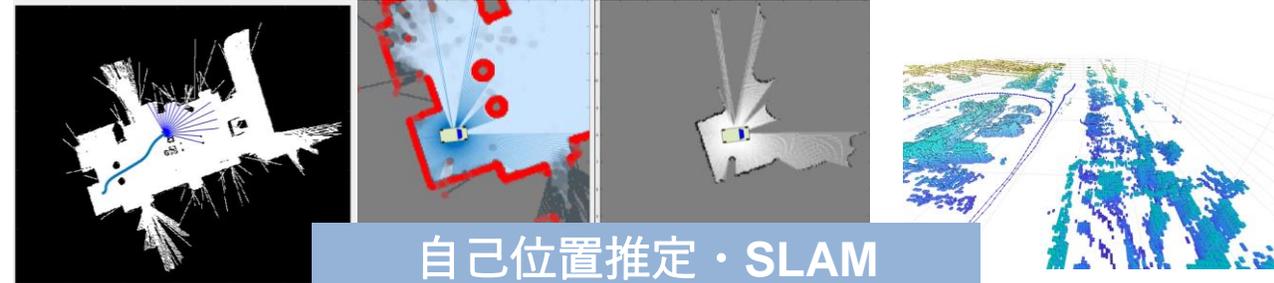


複数台自動搬送

# Navigation Toolbox概要

パスプランニング、自己位置推定、SLAMの設計、シミュレーションおよび実装

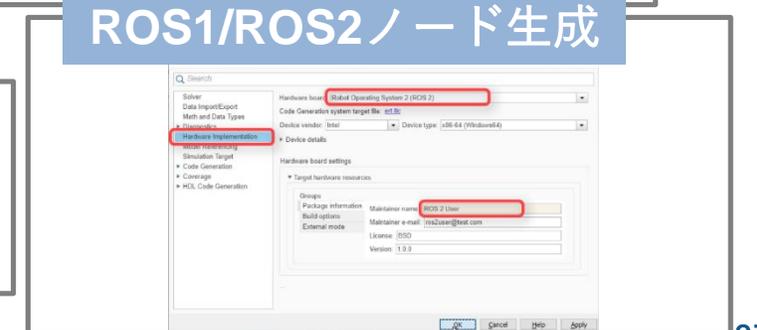
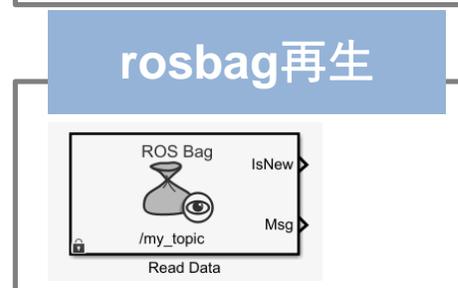
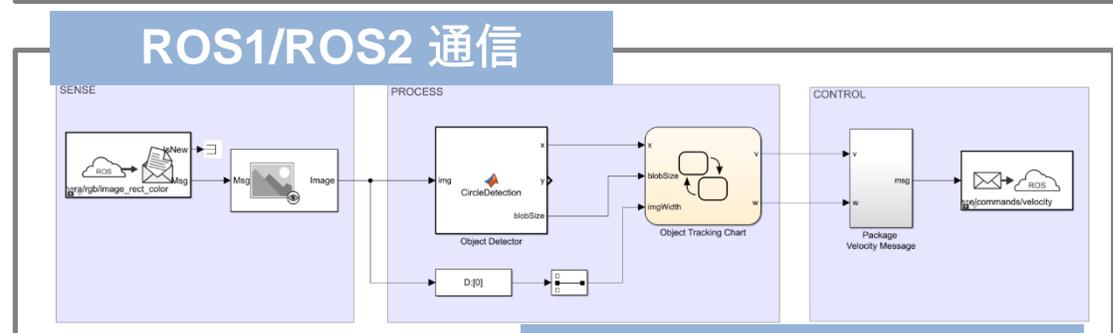
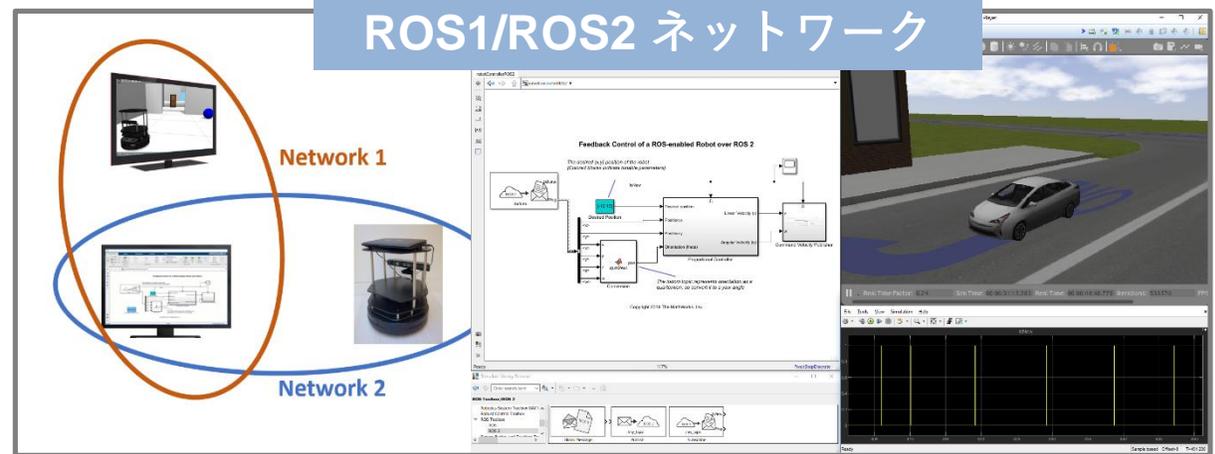
- 自己位置・姿勢推定と環境地図生成(SLAM)
  - 2D/3D占有マップ表現
  - 2D/3D LiDAR SLAMアルゴリズム
- 2D LiDAR SLAMマップビルダーアプリ
  - ループ閉じこみを対話的に
- センサーモデル
  - IMU/GPSセンサーモデル
  - 2D LiDARセンサーモデル
- 位置・姿勢推定(ローカリゼーション)
  - IMU/GNSSによる位置姿勢推定
  - モンテカルロローカリゼーション
- パスプランニング
  - Hybrid A\*
  - RRT/RRT\*(カスタマイズ可能)
- パスプランニングメトリクス
  - メトリクスを使用してパスプランナーの出力した経路を評価
- 軌跡と経路点追従アルゴリズム
  - 軌跡を追従するためのロボットの制御コマンド生成
- 他のToolboxとの連携サンプル
  - ドライビングシナリオ連携 (Automated Driving Toolbox)



# ROS Toolbox概要

ROS1/ROS2ベースのアプリケーションのデザイン、シミュレーションおよび実装

- MATLABとSimulinkをROS1およびROS2に直接接続
  - MATLAB/Simulinkの資産をROS接続
  - MATLAB/SimulinkをROSノード化
- rosbagの読み込みおよびプレイバックをサポート
  - オフラインで解析やテストが可能
- SimulinkからのROS1/ROS2ノード生成に対応
  - C++でROSのコーディング不要
  - catkin/colconでのビルドに必要なファイルも自動生成





© 2020 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.