

Enseigner la Robotique avec MATLAB & Simulink

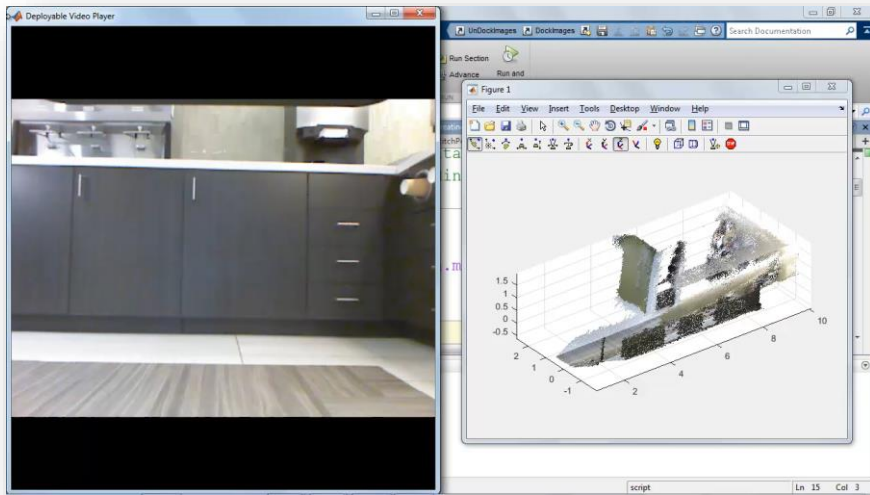
Valerie Leung
Application Engineer

Ninon Candalh-Touta
Ingénieure Pédagogique

Developing Autonomous Robotics and Navigation Systems

Valerie Leung
Application Engineer

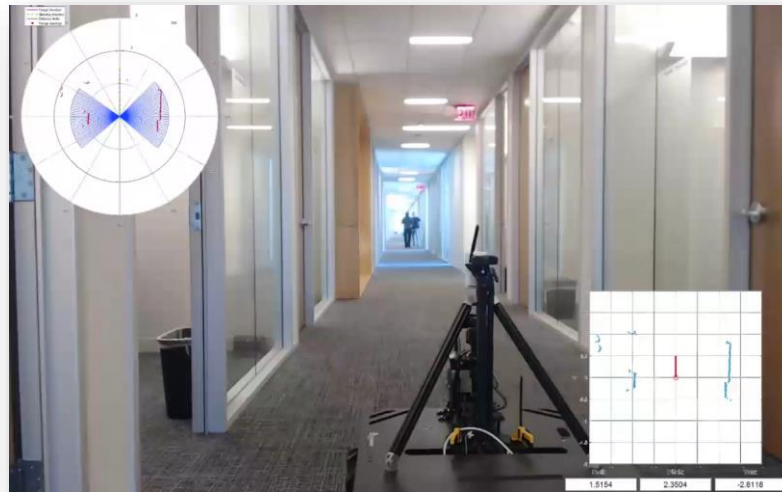
Examples of Autonomous Systems



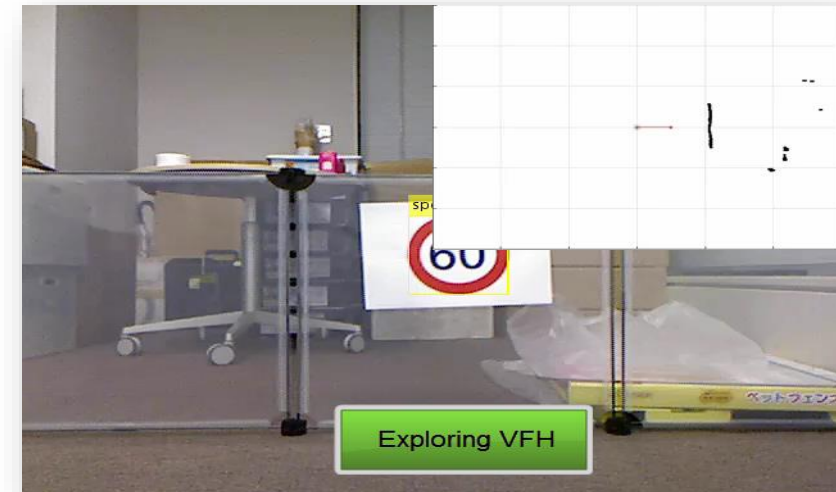
Mapping the Environment



Manipulator Arms/Humanoids



Autonomous Ground Robots



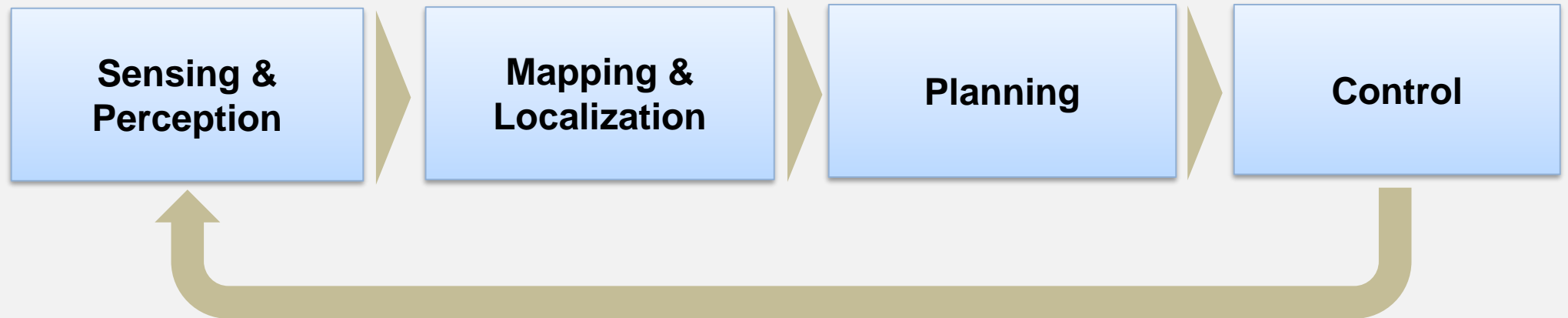
Machine Learning and Controls

Key Components for Enabling Full Autonomy

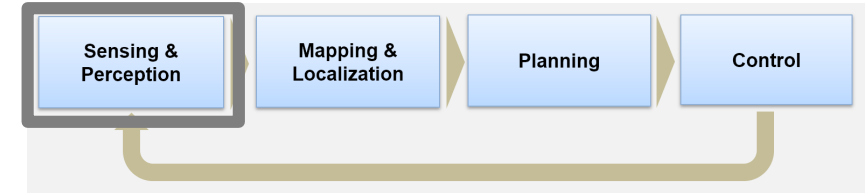
Physical Systems &
Environment



Algorithms



Examples of automated driving sensors

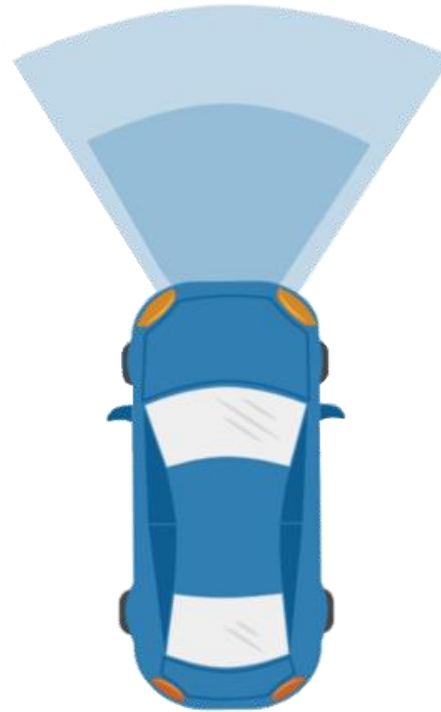


Camera

Radar-based object detector

Vision-based object detector

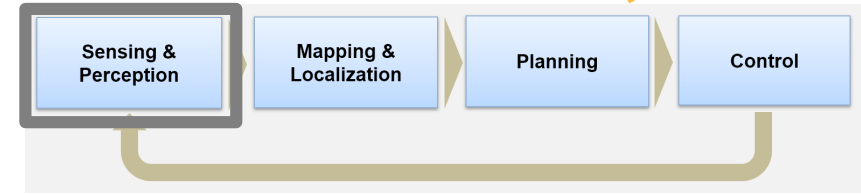
Lidar



Lane detector

Inertial measurement unit

Examples of automated driving sensors



Camera (640 x 480 x 3)

```

239 239 237 238 241 241 241 242 243
252 252 251 252 252 253 253
  
```

Vision Detector

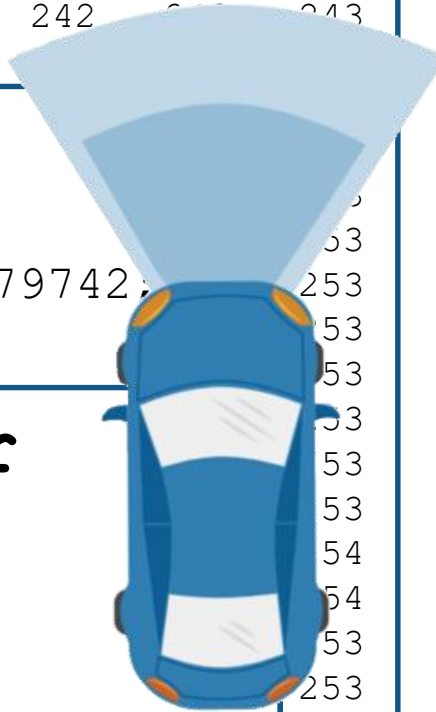
```

SensorID      = 1;
Timestamp     = 1461634696379742;
NumDetections = 6;
  
```

Lane Detector

```

Left
IsValid:      1
Confidence:   3
BoundaryType: 3
Offset:       1.68
HeadingAngle: 0.002
Curvature:   0.000
Right
IsValid:      1
Confidence:   3
  
```



Radar Detector

```

SensorID      = 2;
Timestamp     = 1461634696407521;
NumDetections = 23;
  
```

Detection

```

TrackID      -12.2911  1.4790  -0.59
TrackSt      -14.8852  1.7755  -0.64
Positio      -18.8020  2.2231  -0.73
Amplitu      -25.7033  3.0119  -0.92
Detection    -0.0632  0.0815  1.25
TrackID      -0.0978  0.0855  1.25
TrackSt      -0.2814  0.1064  1.25
  
```

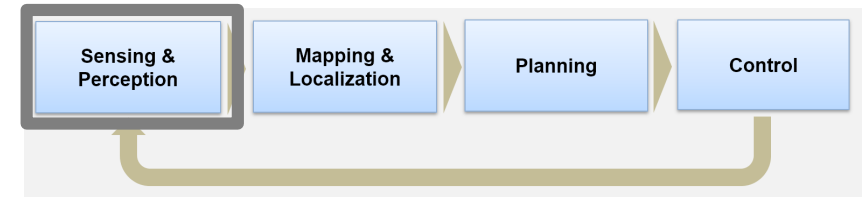
Lidar (47197 x 3)

Inertial Measurement Unit

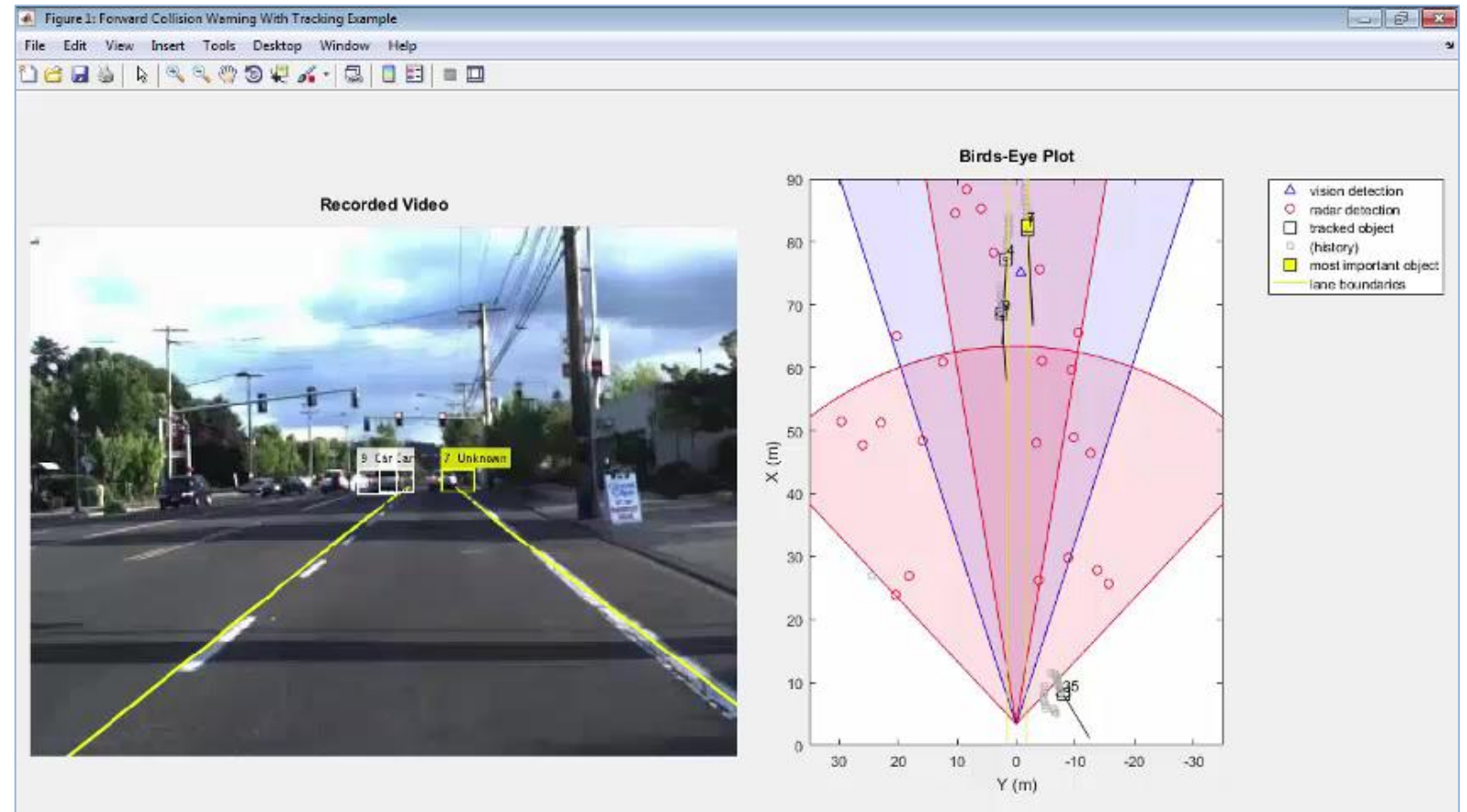
```

Timestamp: 1461634696379742
Velocity:  9.2795
YawRate:   0.0040
  
```

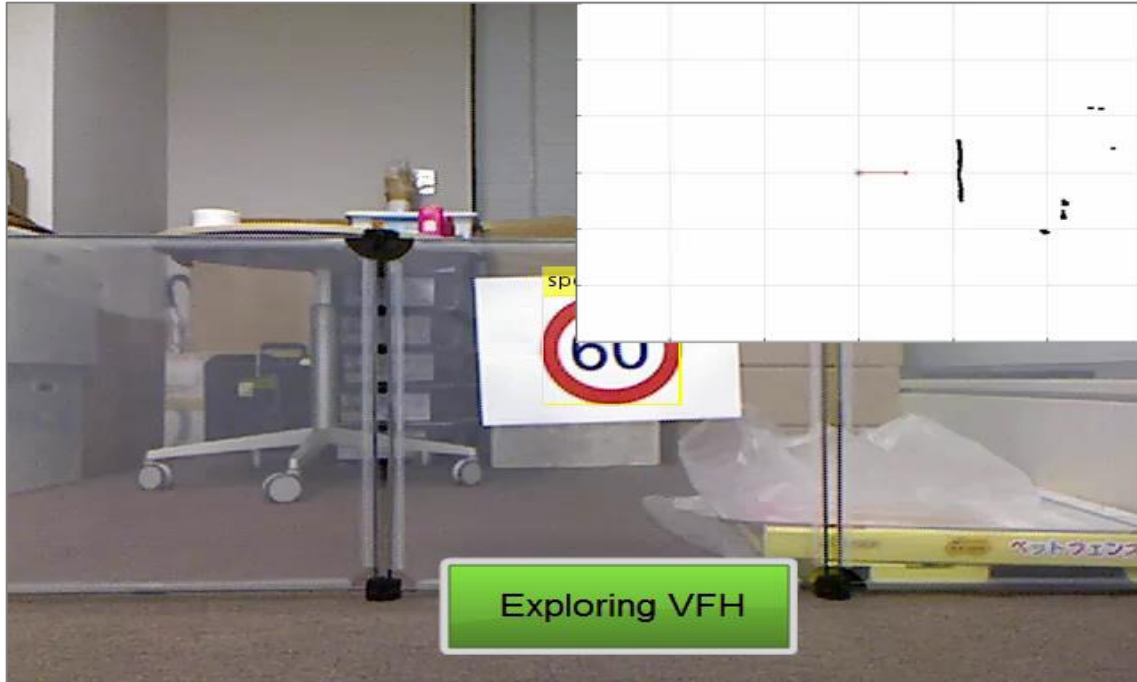
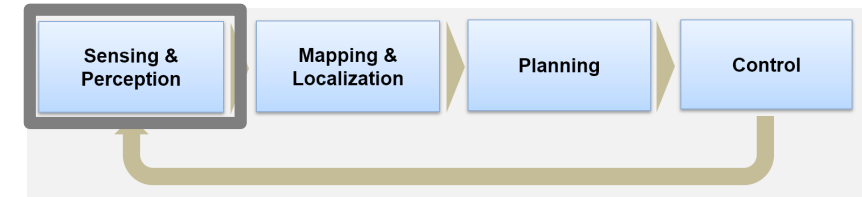
Data visualisation and sensor fusion



- Fusion of data from vision and radar sensors
- Outcome: representation of objects and obstacles around the vehicle
- Basis for a Forward Collision Warning system



Machine Learning & Deep Learning



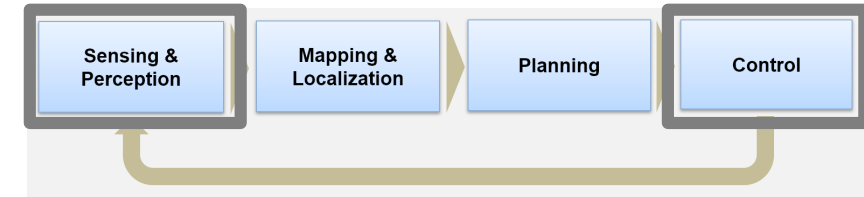
Object detection and recognition



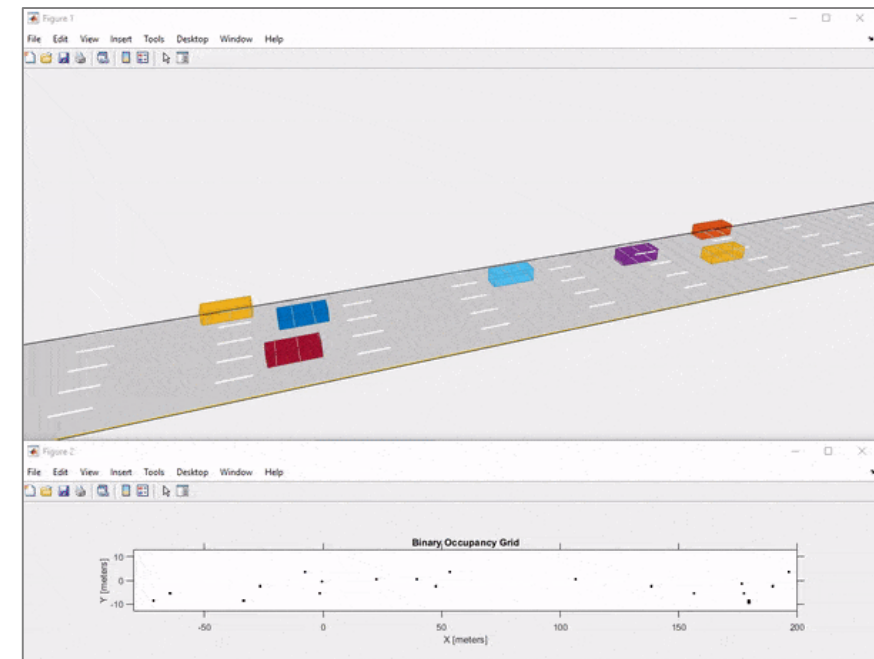
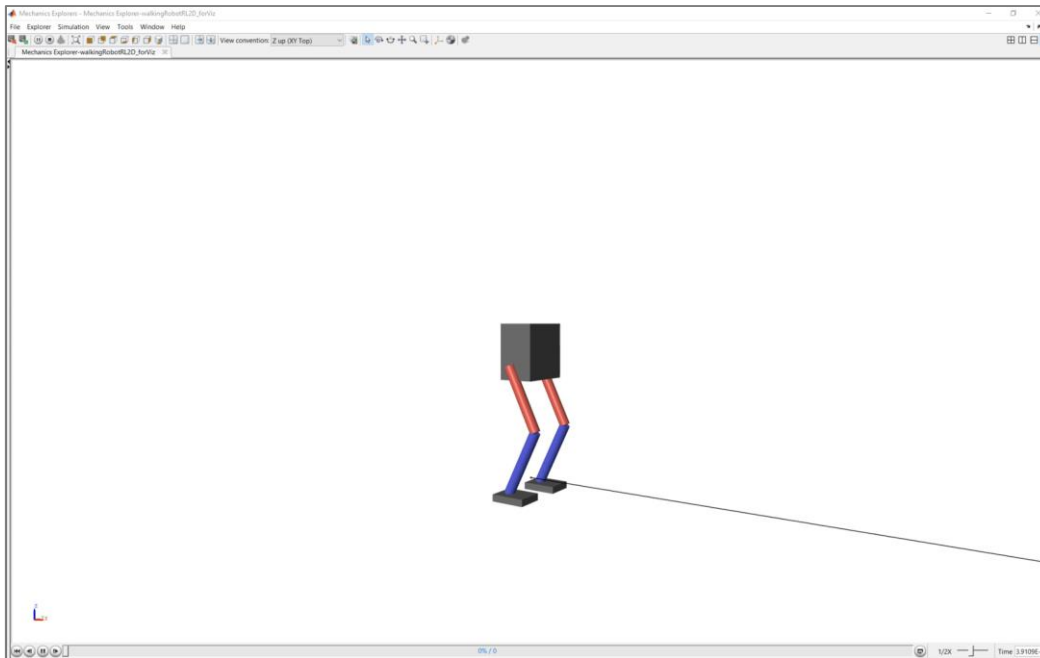
CamVid Dataset

1. Segmentation and Recognition Using Structure from Motion Point Clouds, ECCV 2008
2. Semantic Object Classes in Video: A High-Definition Ground Truth Database, Pattern Recognition Letters

Reinforcement Learning



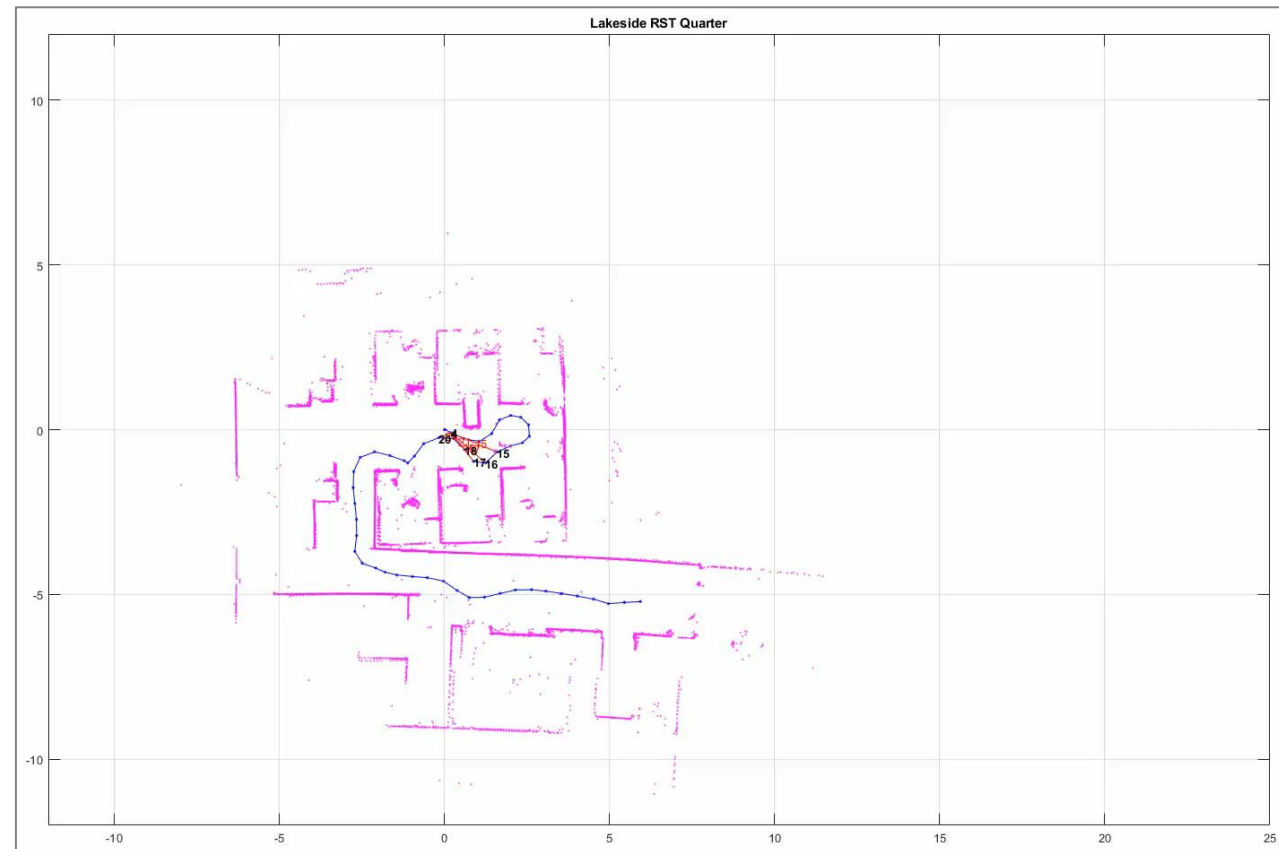
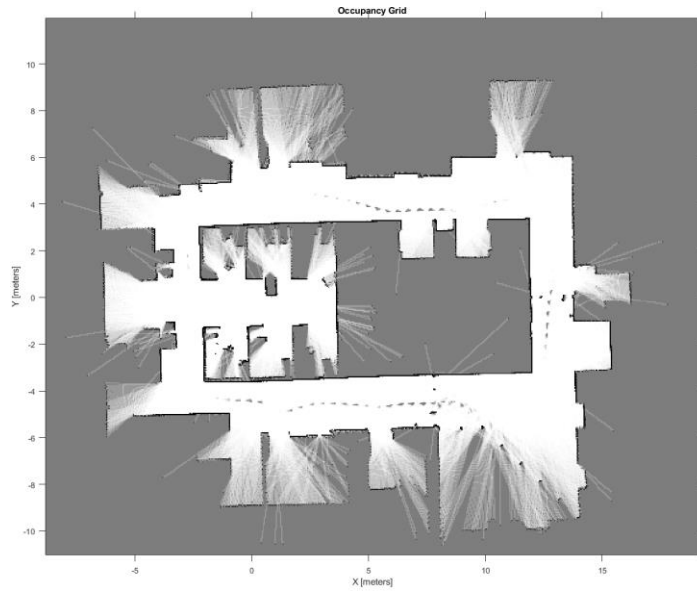
- What is Reinforcement Learning?
 - Type of machine learning that trains an **‘agent’** through repeated interactions with an environment
- How does it work?
 - Through a trial & error process that uses a reward system to maximize success



Simultaneous Localisation and Mapping (SLAM)

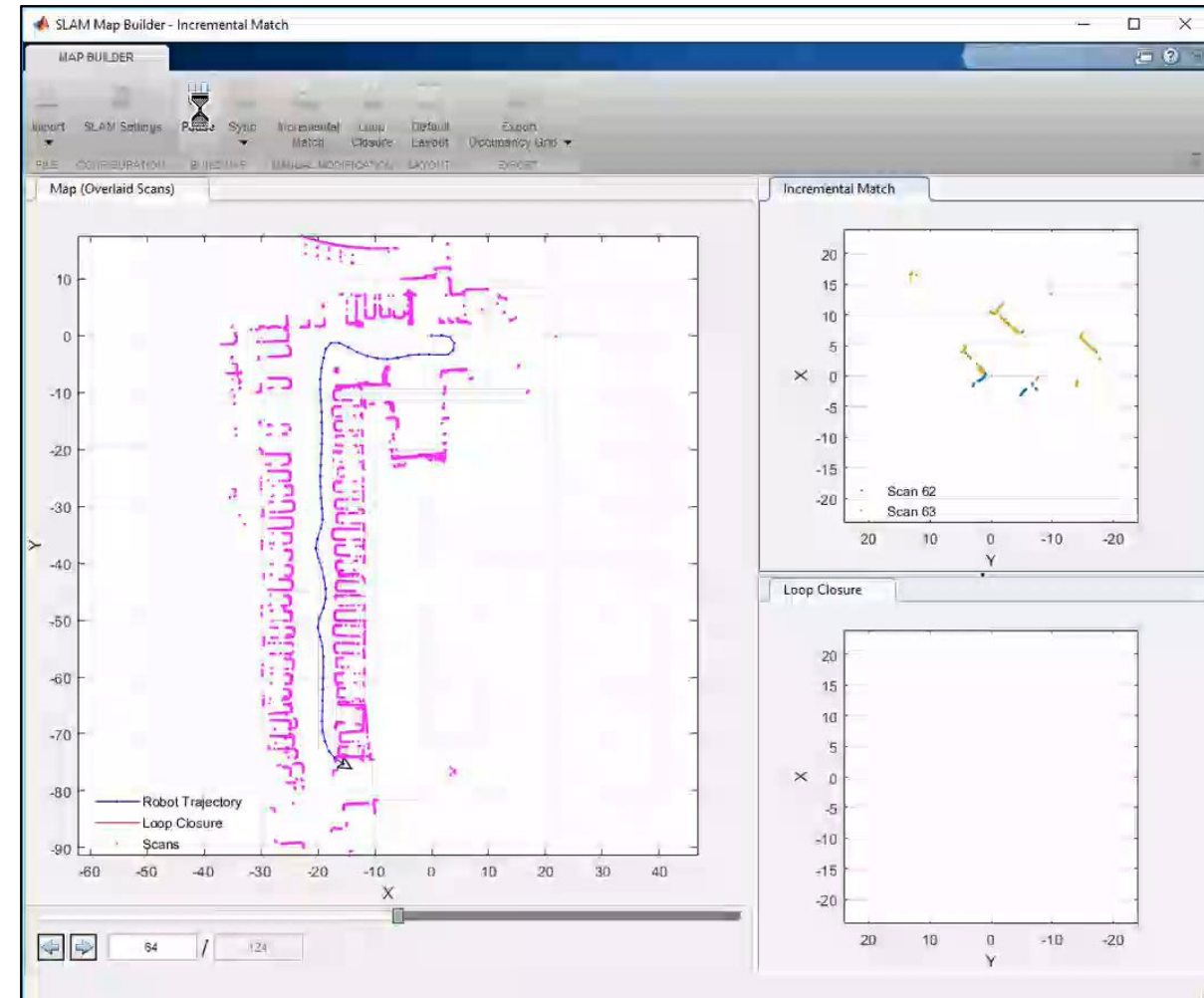


Expected Map

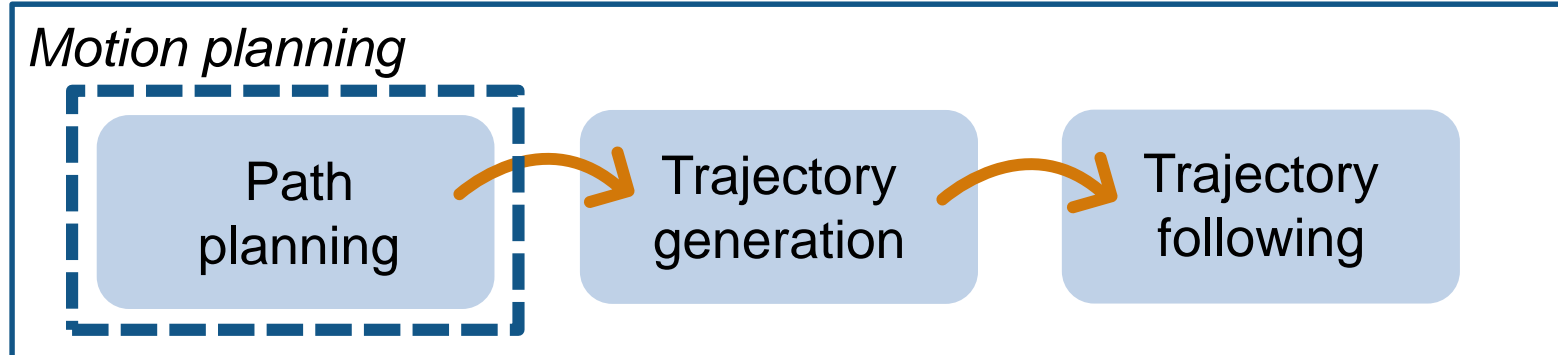
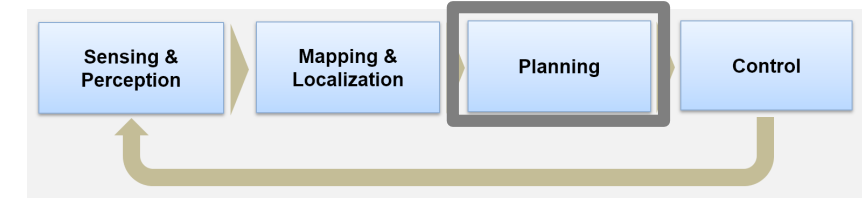


SLAM Map Builder App

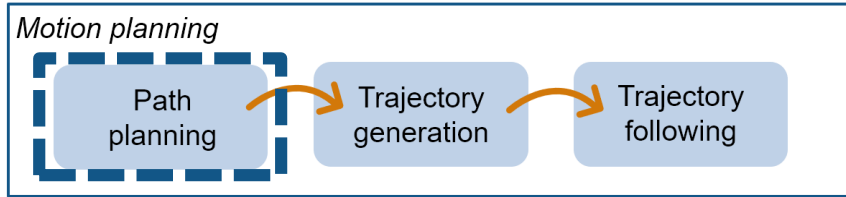
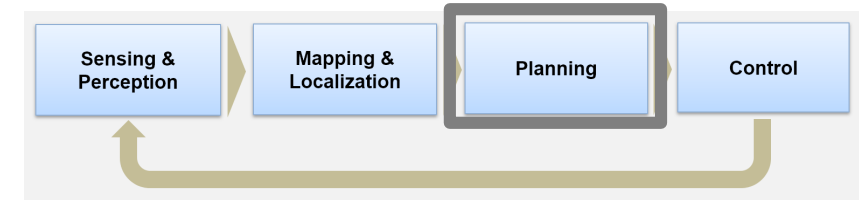
- **Build 2D map of environment** based on Lidar and odometry data
- Modify **loop closures** and **incremental scan matches** to improve map quality
- Export the resulting **occupancy grid** and **use for path planning**
- Loading and saving of sessions to allow saving of work and resuming SLAM map building



Motion planning



Motion planning: path planning



Trajectory Optimization

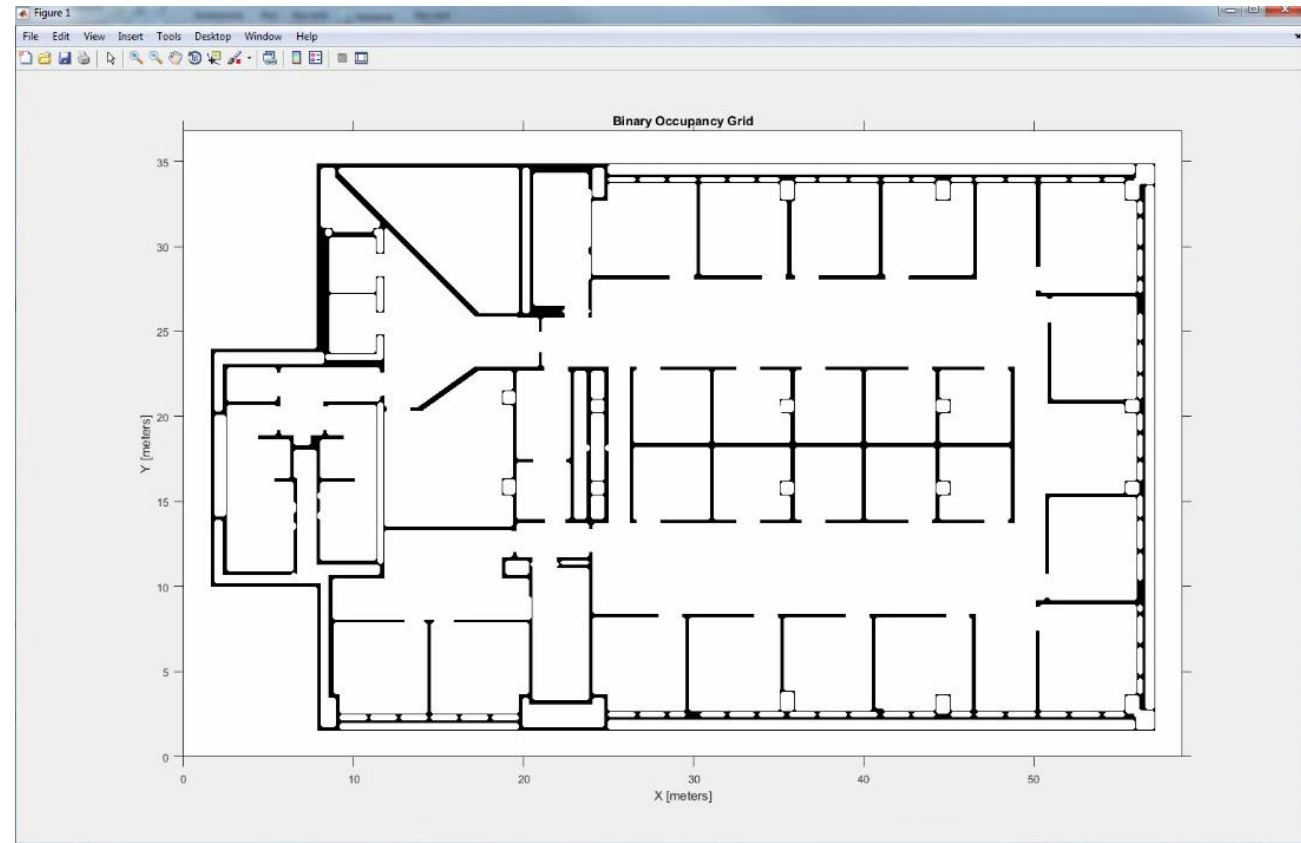
e.g. splines

Grid-based

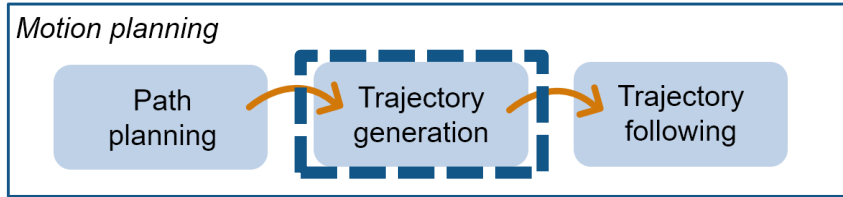
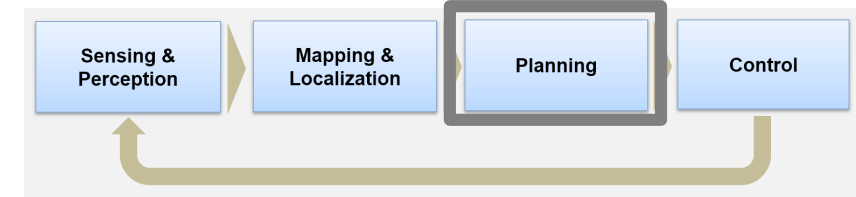
e.g. Dijkstra's, A*

Sampling-based

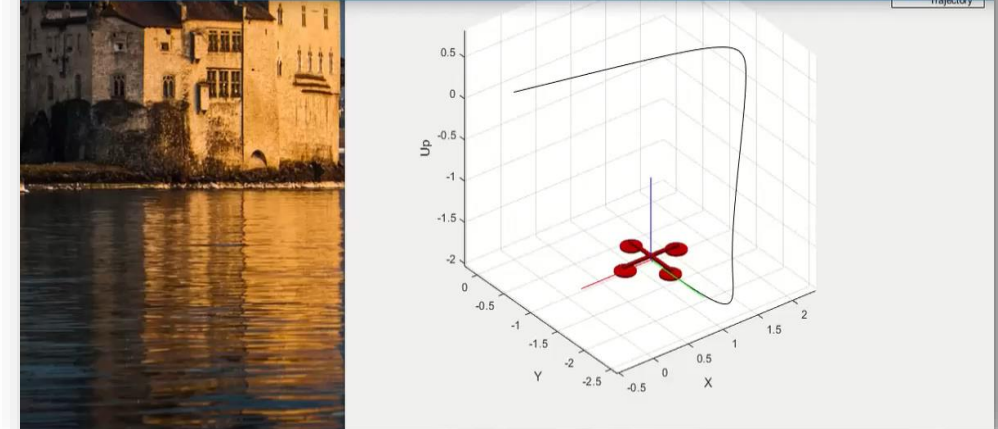
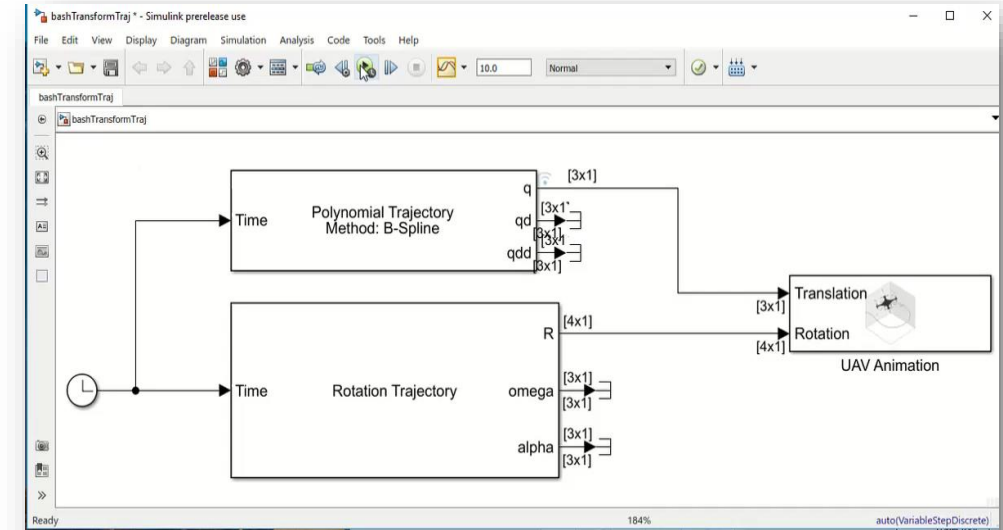
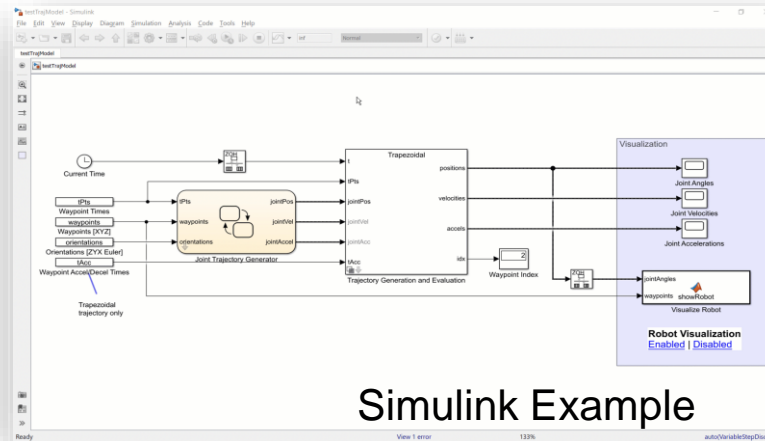
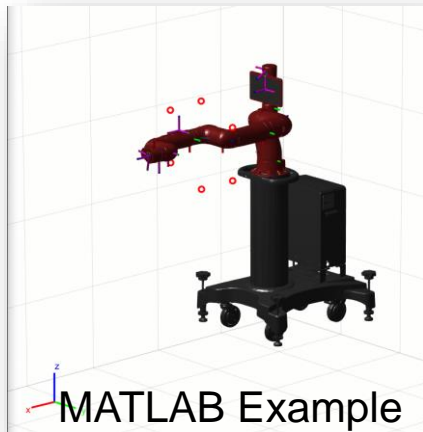
e.g. PRM



Motion planning: Trajectory Generation

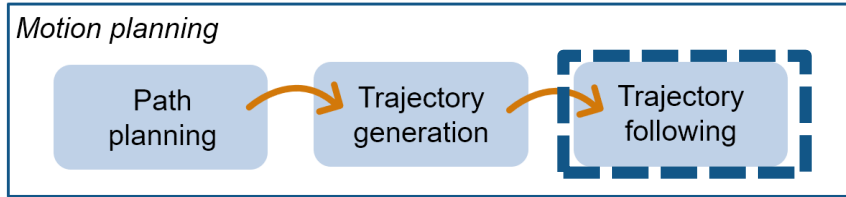


Use polynomials and splines to generate manipulator trajectories



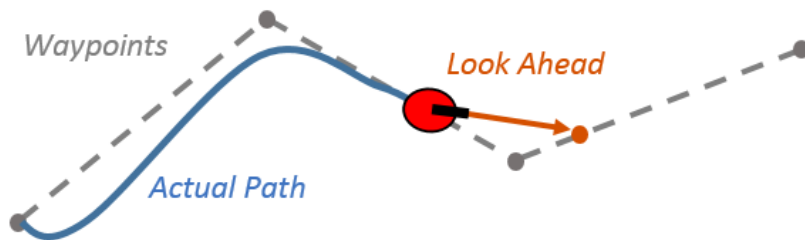
Extensible to other autonomous systems, e.g. UAVs

Motion planning: trajectory following

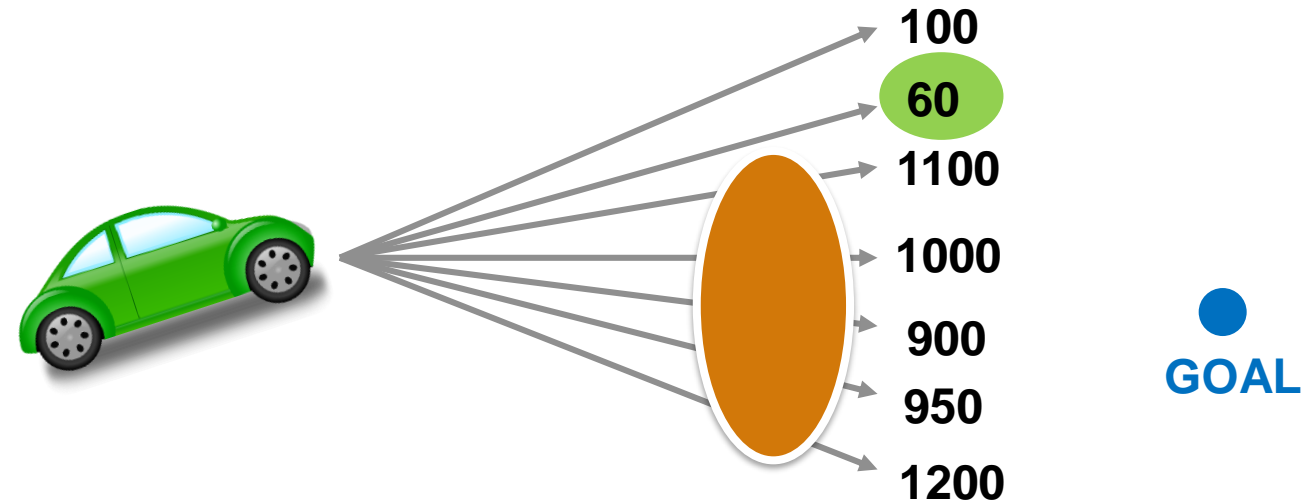


Pure Pursuit

- No collision avoidance!
- Mobile robots or latitude controller for vehicles operating at low speeds



Predictive Control



- Rollout all the trajectories
 - Actuator limits / State limits / Motion constraints
- Assign cost to each trajectory
- Select best one and execute

Complete Workflow

SLAM, Planning, Following



Two ways of interfacing MATLAB/Simulink with ROS

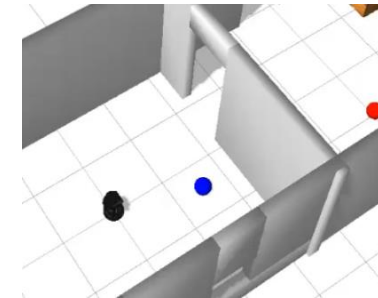
Mode 1



- Develop algorithm in MATLAB / Simulink
- Algorithm runs on PC



ROS-enabled simulator or robot



ROS

Mode 2



Develop algorithm in Simulink



Generate independent ROS node (C++)



Deploy on robot / ROS network

Robot is autonomous

Design and Test Robotics Algorithms

Test with a physical robot running ROS



Mode 1: Algorithm on PC, communicate with ROS-enabled robot through interface

Mode 2: Independent ROS node on robot



Key Components for Enabling Full Autonomy

Physical Systems & Environment



Algorithms





Manipulators



Ground vehicles



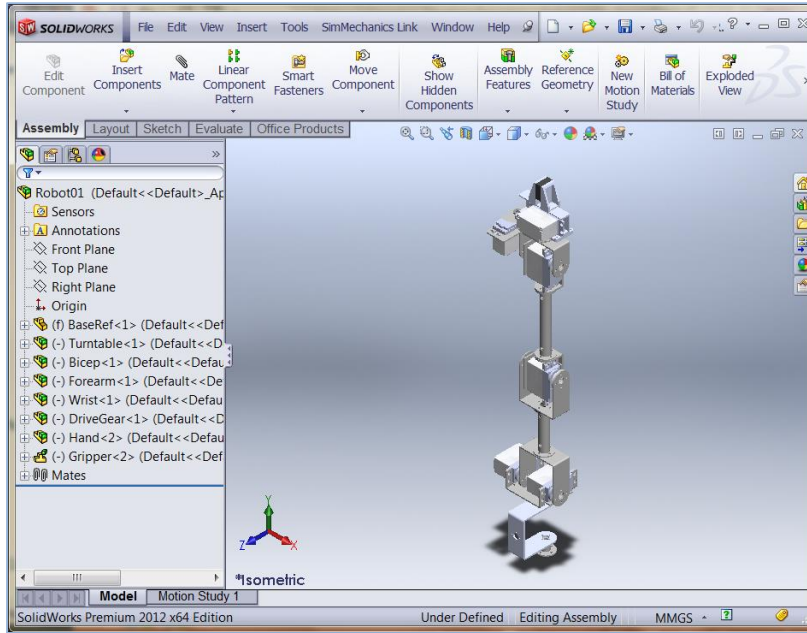
UAVs

The image displays a robotic system in three different views:

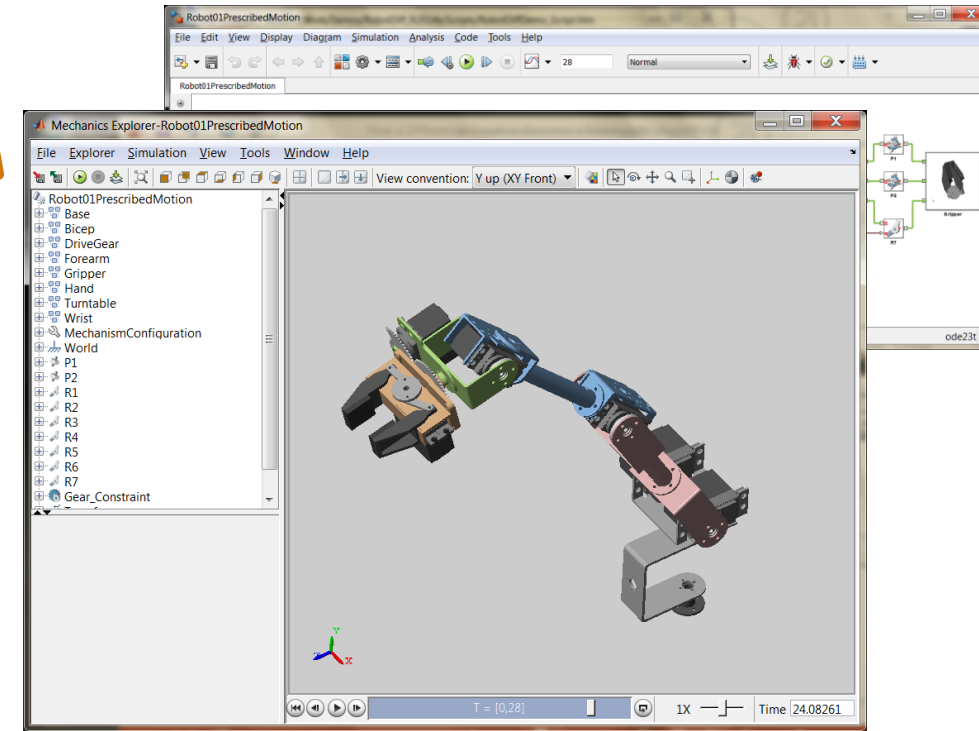
- Top Left:** A photograph of a physical setup. A person is operating a computer with a mouse. In the foreground, a robotic arm is positioned over a table with a grid. Several objects are on the table, including a Snickers candy bar and a small white card with a black and white pattern. The robotic arm is labeled 'KINOVIA'.
- Top Right:** A screenshot of the 'Mechanics Explorers' software. It shows a 3D simulation of the robotic arm in a green environment. The interface includes a menu bar (File, Explorer, Simulation, View, Tools, Window, Help), a toolbar, and a playback control bar at the bottom showing '0% / 15.7' and 'Time 15.9'.
- Bottom:** A screenshot of the 'jacoLogicAndVision' Simulink model. The diagram shows the control architecture:
 - User_Command** block sends a **userCommand** signal to the **Motion_Planner** block.
 - Kinect 2** block sends an **image** signal to the **Vision_System** block.
 - Vision_System** block outputs **visionObjects** to the **Motion_Planner** block.
 - Motion_Planner** block outputs a **Command** signal to the **robotState** block.
 - robotState** block outputs a **robotState** signal to the **Motion_Planner** block.
 - Motion_Planner** block outputs a **ROBOT_CMD_BUS D1** signal to the **robotState** block.
 - robotState** block outputs a **3(27) D1** signal to the **Motion_Planner** block.
 - Vision_System_Test** block outputs a **visionObjects** signal to the **Motion_Planner** block.

**Mechanics:
import models
from common
CAD tools**

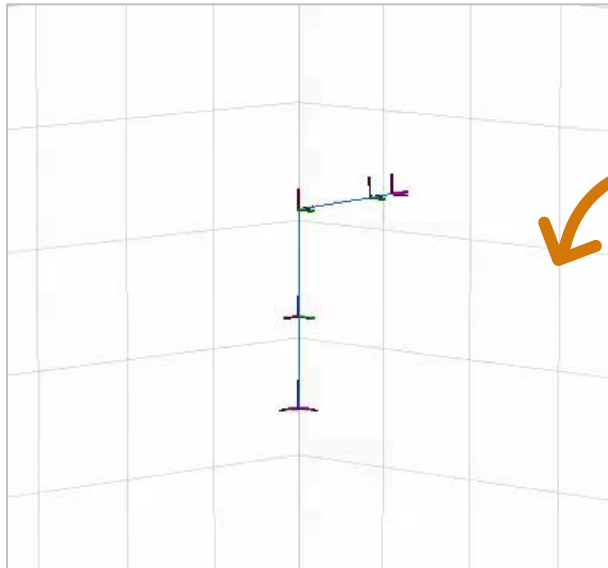
SolidWorks Model



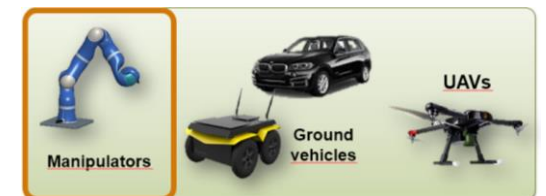
Simscape Multibody Model

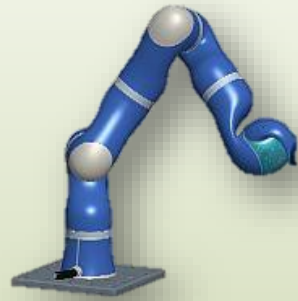


**Define rigid
body tree and
compute
dynamics**



**Generate rigid
body tree from
Simscape
Multibody
models**





Manipulators



Ground vehicles



UAVs

UAV library

- Support Package for UAVs
- Supports small multi-rotor and fixed-wing aircraft models
- MATLAB functions and Simulink blocks for autonomous flight



The screenshot shows the MathWorks File Exchange interface. At the top, there are navigation links for Products, Solutions, Academia, Support, Community, and Events. The main header is "File Exchange" with a search bar. Below this, there are tabs for MATLAB Central, Files, Authors, My File Exchange, Contribute, and About. The featured submission is "Robotics System Toolbox UAV Library" by the MathWorks Robotics Team (STAFF). The description states it contains algorithms for simulating UAVs. The "Overview" tab is selected, showing a detailed description of the library's capabilities and reference examples. There are also sections for "Comments and Ratings (4)" and a "Rate this submission" section with a five-star rating and a comment box. The right sidebar shows a "Requirements" section and a "Tags" section with labels like "aerial", "fixed-wing", "guidance", "multi-rotor", and "uav".

MathWorks® Products Solutions Academia Support Community Events

File Exchange Search File Exchange

MATLAB Central Files Authors My File Exchange Contribute About

Robotics System Toolbox UAV Library
by MathWorks Robotics Team **STAFF**
Algorithms to simulate Unmanned Aerial Vehicles (UAVs)

Overview

Unmanned aerial vehicles (UAVs) can be modeled and controlled using UAV Library for Robotics System Toolbox™ functions, objects, and blocks. You can simulate a reduced-order guidance model for fixed-wing and multi-rotor UAVs that approximates a closed-loop autopilot controller with a kinematic model. Generate control commands, UAV states, and environmental inputs using the given functions. A waypoint follower is also provided to execute flight missions based on pre-defined waypoints.

The UAV Library for Robotics System Toolbox contains the following reference examples:

- Approximate High-Fidelity UAV model with UAV Guidance Model block
- Tuning Waypoint Follower for Fixed-Wing UAV

Comments and Ratings (4)

Rate this submission ★★★★★ (Rating not required)

Comment on this submission

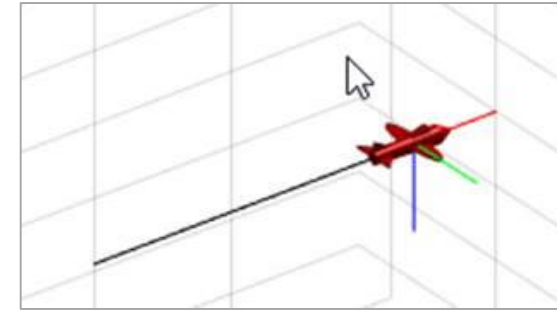
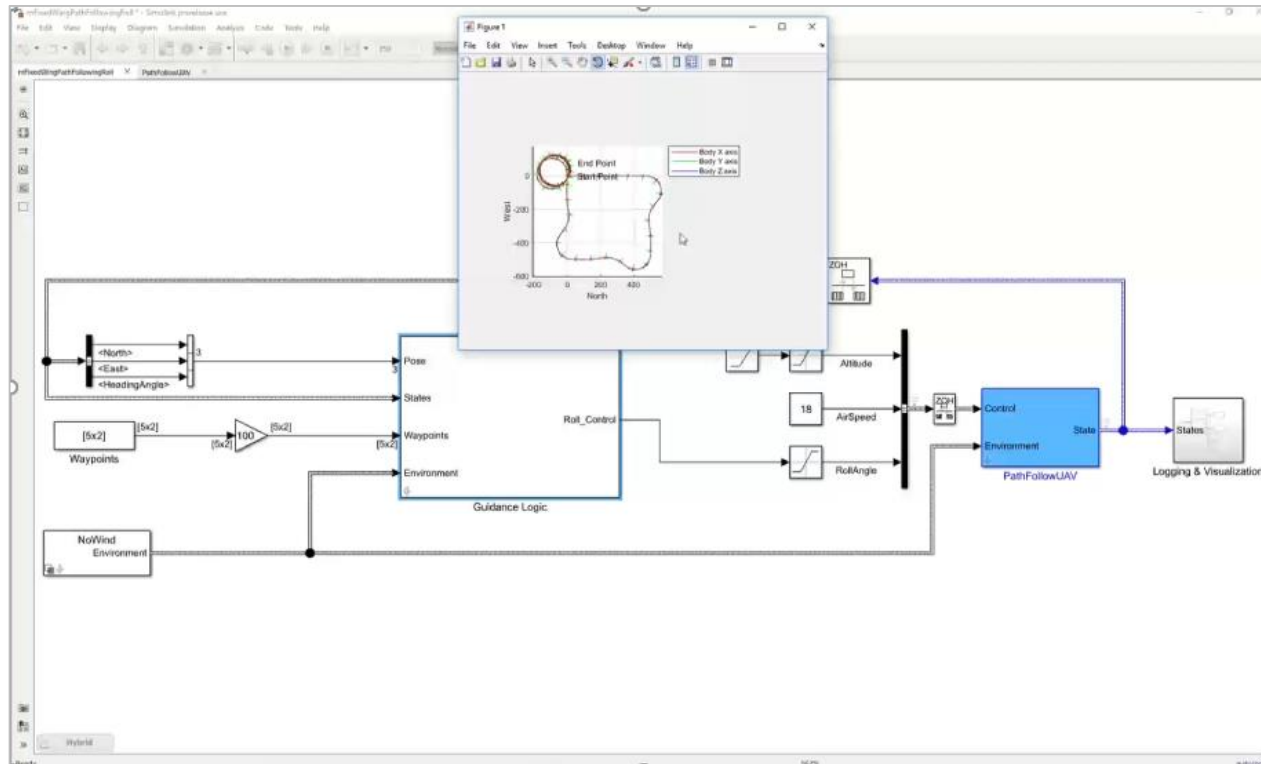
anant garg ★★★★★
14 Jan 2019

Mihir Acharya ★★★★★

Requirements
MATLAB
Robotics System Toolbox
MATLAB
Created by
Company
Platform
 Windows
Tags
aerial
fixed-wing
guidance
multi-rotor
uav

UAV Guidance Block & Waypoint Following

- **Low-fidelity simulation model** a small fixed-wing or multi-rotor UAV with autopilot
- Follow a set of user-defined **waypoints**
- **Deploy algorithm** to UAV hardware



Key Components for Enabling Full Autonomy



ROS Interface  ROS

Deployment to embedded platforms with automatic code generation

C/C++



CUDA



FPGA



Nara Institute of Science and Technology Researchers Develop Tactile Object Recognition Algorithms for a Robotic Hand



NAIST's ROS-enabled Shadow robot performing tactile object recognition.

Challenge

Enable a robotic hand to identify objects based on tactile sensor input

Solution

Use MATLAB to develop machine learning and object recognition algorithms and use Robotics System Toolbox to establish connectivity between the algorithms and the ROS-enabled robot

Results

- Hundreds of manual steps eliminated
- Opportunities to try new algorithm ideas increased
- Expertise shared with other students and researchers

[Link to user story](#)

“With Robotics System Toolbox, we seamlessly connected to and controlled our robot directly from the algorithms we had developed in MATLAB, enabling us to minimize development time. We used the time we gained to further our research into new tactile object recognition algorithms.”

Takamitsu Matsubara

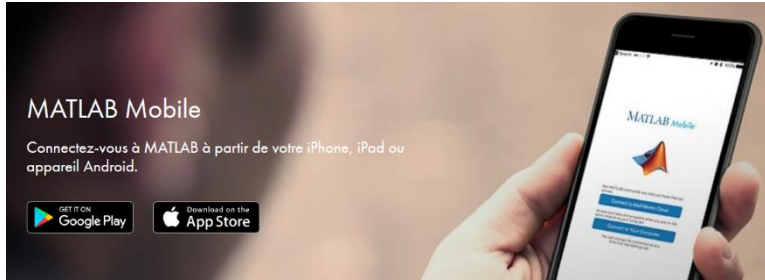
Nara Institute of Science and Technology

Supports Pédagogiques

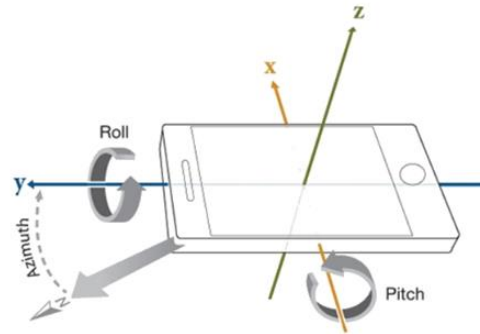
Ninon Candalh-Touta
Ingénieure Pédagogique

Outils Pédagogique

➤ Projets interactifs à bas coût



- Connexion au Cloud
- Connexion à votre ordinateur
- Collecter des données de capteurs



- Accélération sur 3 axes
- Vitesse angulaire sur 3 axes
- Champ magnétique sur 3 axes
- Orientation (azimut, tangage et roulis)
- Position (latitude, longitude, altitude, précision horizontale, vitesse, course)

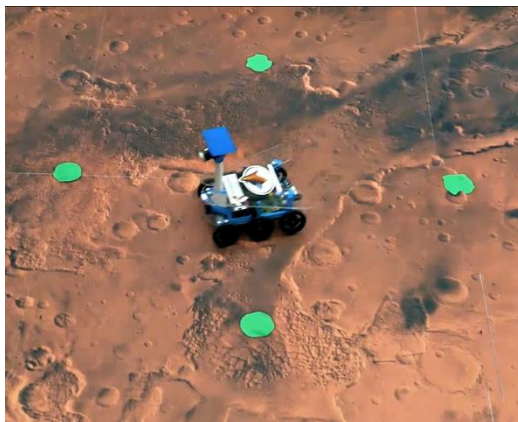
[See examples](#)

MathWorks Supported Hardware

 Arduino	 Raspberry Pi	 Lego EV3	 Kinect for Windows
 BeagleBone	 STM Electronics	 Zynq SDR	 Texas Instruments
 Arduino Kit			 Parrot Mini-Drone

Outils Pédagogique

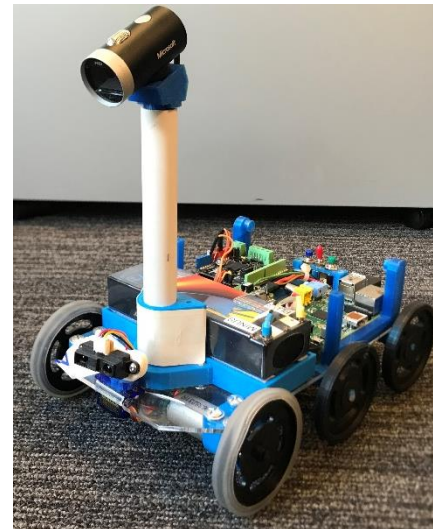
- MathWorks Rover Robot: a platform for multidisciplinary project



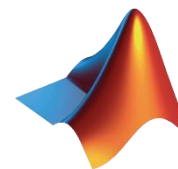
Véhicule Autonome



CentraleSupélec



RENAULT



Student's point of view

“The MathWorks rover is an **inspiring** and **exciting** platform to work on. It is a concrete project with real constraints. It enabled us to

- see the **difference between simulation** and **implementation** on a real system
- show the interest of all the knowledge we learn in class and **allow us to mix them up**.

It was for us a **fun way** to practice with the MATLAB/Simulink software suite.”

Outils Pédagogique

➤ Cours et Exemples en ligne

Vidéos et Webinars

[Vidéos](#)

Vidéos | Recherche

Contactez-nous
 Version d'évaluation

Recherche par Produit

MATLAB	92
Simulink	188
Aerospace Blockset	1
Antenna Toolbox	1
Audio Toolbox	1
Automated Driving Toolbox	1
Communications Toolbox	1

FILTERED BY Mécatronique x Remove All x

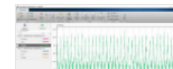
← Résultats 1 - 25 sur 332 →



Le Model-Based Design pour la conception de produits de protection électrique
 A travers l'exemple de l'intégration de fonctions avancées dans un disjoncteur de puissance, découvrez comment les approches à base de modèles permettent à Schneider Electric d'appréhender cette complexité...
Date: 19 juin 2018



Concevez vos algorithmes de maintenance prédictive en toute simplicité avec MATLAB
 Découvrez comment MATLAB et Predictive Maintenance Toolbox simplifie le travail des ingénieurs élaborant des algorithmes de maintenance prédictive.
Date: 30 mai 2018



Maintenance Prédictive - Comment recourir à la simulation en l'absence de données ?
 Le développement d'algorithmes de maintenance prédictive présuppose l'existence de jeux de données exploitables.

Recherche par Type de vidéo

Conférence	53
Démo	76
Ressources	25
Témoignages des utilisateurs	2
Vue d'ensemble du produit	7
Webinar	147

Get MATLAB



File Exchange

File Exchange

Trial software

Filter by Source

- Community 35,346
- MathWorks 260

Filter by Category

- Using MATLAB
- Language 879
 - Fundamentals
 - Data Import and Analysis 1,056
 - Mathematics 1,392
 - Graphics 1,857
 - Programming 366
 - App Building 409
 - Software Development 144
 - Tools
 - External Language 431

Most Recent

points -2644000

Asteroids

short game

0 Downloads ★★★★★

Power Spectral Density of the noise signal

Noise Analysis with Matlab

Time and frequency analysis, measurement of the noise statistics, etc.

0 Downloads ★★★★★

Signal in the time domain

Noise Measurement with Matlab

Estimation of the Noise PSD and Noise Voltage referred to the output of a circuit.

22 Downloads ★★★★★

Efficient Primal-Dual Method for the Obstacle Problem

Solve 1D/2D non-linearized and linearized obstacle problems efficiently using primal-dual hybrid

0 Downloads ★★★★★

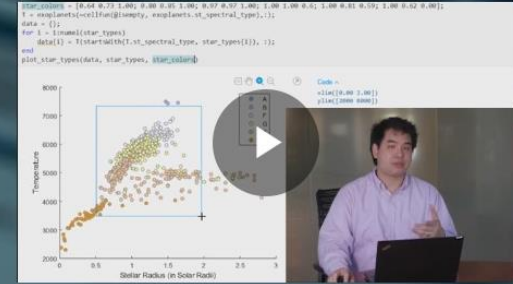
Show All

Outils Pédagogique

➤ Live Editor

MATLAB Live Editor

Explorez visuellement et analysez les problèmes dans un environnement unique et interactif, et transformez votre code en documents formatés et exécutables qui racontent votre histoire.



Estimating Sunrise and Sunset

We can calculate sunrise and sunset times from the following equations.

$$\text{sunrise} = 12 - \frac{\cos^{-1}(-\tan \phi \tan \delta) - \frac{SC}{60}}{15^\circ} \quad \text{sunset} = 12 + \frac{\cos^{-1}(-\tan \phi \tan \delta) - \frac{SC}{60}}{15^\circ}$$

```

long = -71 ;
lat = 42 ;
timeZone = 'Eastern (UTC-5)';
sc = solarT('Central (UTC-6)', timeZone);
delta = asin(sin(lat)*sin(sc/360*(days - 81)/365));
sunrise = 12 - (acosd(-tand(lat)*tand(delta))/15 - sc/60);
sunset = 12 + acosd(-tand(lat)*tand(delta))/15 - sc/60;

plot(days, sunrise, days, sunset, 'LineWidth', 4)
title('Sunrise and Sunset')
xlabel('Day of Year')
    
```


Outils Pédagogique

➤ Formations en ligne

Getting Started

FREE

MATLAB Onramp

Get started quickly with the basics of MATLAB.

Launch
Details

FREE

Simulink Onramp

Get started quickly with the basics of Simulink.

Details

FREE

Deep Learning Onramp

Get started quickly using deep learning methods to perform image recognition.

Launch
Details

Computational Mathematics

*Available only to users at universities that offer campus-wide online training access.

NEW

Solving Nonlinear Equations with MATLAB

NEW

Solving Ordinary Differential Equations with MATLAB

NEW

Introduction to Linear Algebra with MATLAB

NEW

Introduction to Statistical Methods with MATLAB

Core MATLAB Functionality

MATLAB Fundamentals

MATLAB Programming Techniques

MATLAB for Financial Applications

MATLAB for Data Processing and Visualization

Machine Learning with MATLAB

UserStory

@CentraleMarseille

Cours en autonomie

260 sur 263 étudiant(e)s ont validé

+ Retours positifs des autres enseignants, les élèves sont meilleurs en MATLAB et sont plus autonomes
 + Plus de temps « perdu » dans l'apprentissage des fondamentaux de MATLAB

View/Share Certificate

Course Certificate

View/print certificate

Share certificate:

in
f
shareable link

Progress Report (includes...)

View/print progress report

Share progress report:

shareable link

MathWorks | Training Services

Progress Report

Name: Nadia Hedjazi

Course: MATLAB Onramp

Progress: 84% complete (as of 08-Jan-2018)

Chapters

1. Course Overview 100%	12. Logical Arrays 0%
2. Commands 50%	13. Programming 0%
3. Vectors and Matrices 66%	14. Final Project 0%
4. Importing Data 100%	15. Survey 0%
5. Indexing into and Modifying Arrays 0%	
6. Array Calculations 0%	
7. Calling Functions 0%	
8. Obtaining Help 0%	
9. Plotting Data 0%	
10. Review Problems 100%	
11. MATLAB Scripts 100%	

Release: R2017a | Language: English

**Merci,
avez-vous des questions?**

Venez discutez au **Stand MathWorks**