Documentation liens Simulink-IHM :

https://www.mathworks.com/matlabcentral/answers/481469-app-designer-linked-to-simulink

Généralités Digital Twins :

https://www.mathworks.com/discovery/digital-twin.html#why-digital-twins-matter

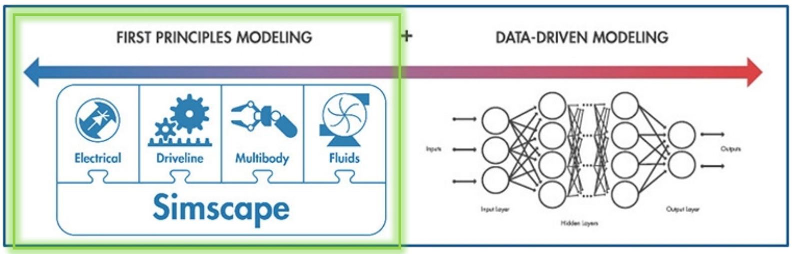
Jumeaux numériques

Rechercher sur MathWorks.com

Comment fonctionnent les jumeaux numériques ?

L'application IoT va dicter ce que vous devez modéliser dans le jumeau numérique. Un modèle de jumeau numérique comprend les composants requis, les comportements et la dynamique de la ressource IoT.

De manière générale, les méthodes de modélisation peuvent être regroupées en deux catégories : les méthodes basées sur les premiers principes ou sur la physique (par exemple, la modélisation mécanique) et les méthodes guidées par les données (par exemple, le Deep Learning). Un jumeau numérique peut aussi être un composite de plusieurs comportements modélisés et méthodes de modélisation. Il sera probablement élaboré au fil du temps, au fur et à mesure de l'identification de nouvelles d'utilisations.



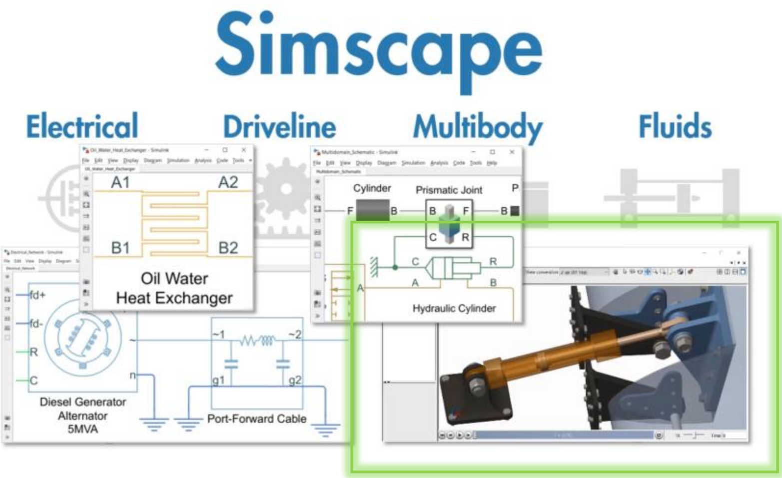
Méthodes de modélisation des jumeaux numériques basées sur les premiers principes et guidées par les données.

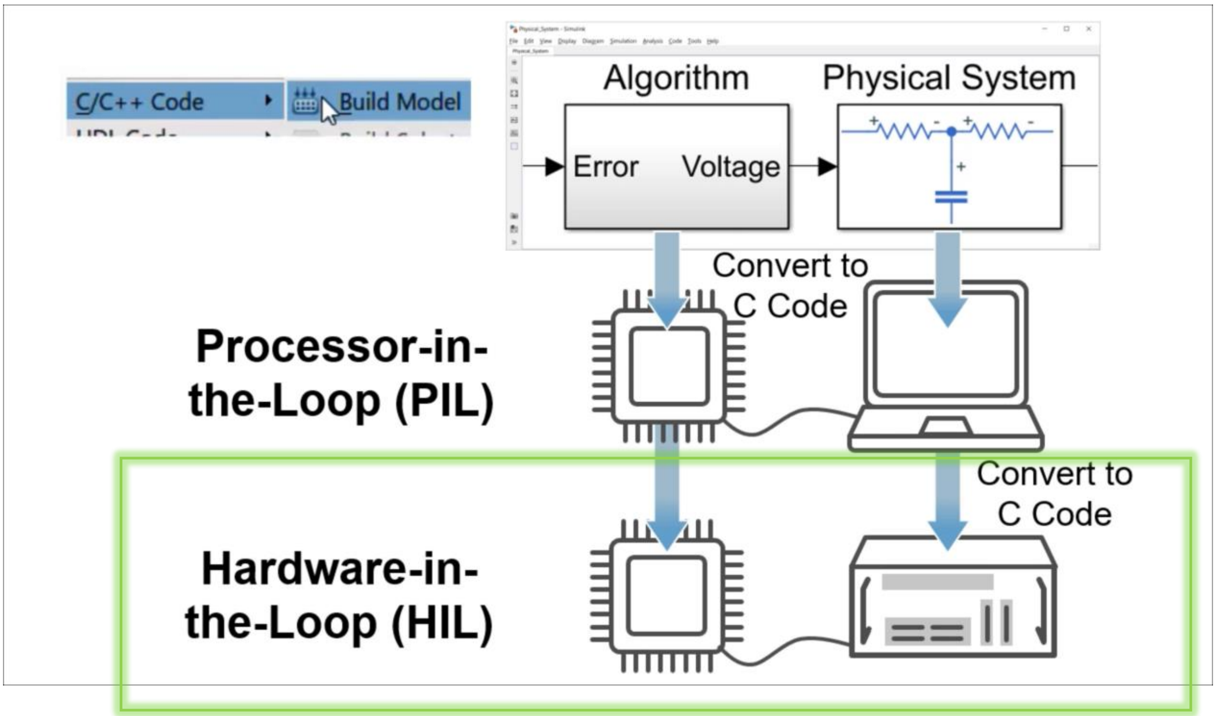
Les modèles doivent rester à jour et en phase avec les ressources en fonctionnement, ce qui implique un échange direct des données entre les ressources et les algorithmes d'ajustement du jumeau numérique. Cela vous permet de prendre en compte des aspects tels que l'environnement, l'âge et la configuration de la ressource.

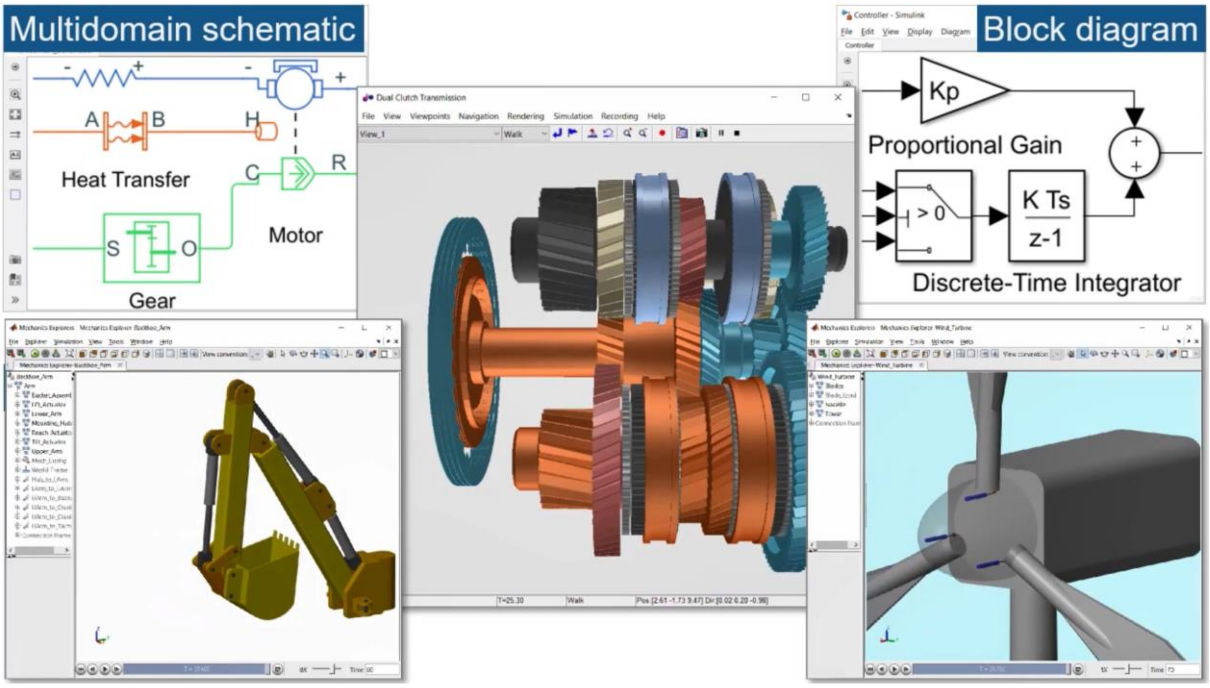
Une fois le jumeau numérique disponible et à jour, vous pouvez l'utiliser de nombreuses manières afin de prévoir un comportement futur, améliorer le contrôle ou optimiser le fonctionnement de la ressource. Par exemple, il peut être utilisé pour la simulation de capteurs non présents sur la ressource en question, la simulation de futurs scénarios pour déterminer les opérations actuelles et à venir, ou encore l'utilisation d'un jumeau numérique pour extraire l'état de fonctionnement actuel par l'envoi de données actuelles et réelles.

Prise en main de Simscape pour la simulation mécatronique du jumeau numérique :

https://www.mathworks.com/videos/simscape-overview-61215.html







EXEMPLE Hardware In the Loop VOLVO:

https://www.mathworks.com/company/user_stories/volvo-construction-equipment-streamlines-product-development-with-a-real-time-human-in-the-loop-simulator.html

- Modules nécessaires :
  - Simscape ?
  - Simscape multibody ?

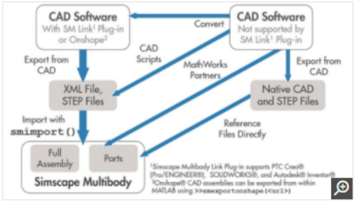
Import from CAD Software

Automatically convert CAD designs to create a digital twin of your system.

Import Assemblies with Joints

Entire CAD assemblies, including all parts with mass, inertia, and color, along with mate and joint connections, are automatically converted into a Simscape model. Updates to existing CAD parts can be merged into the Simscape model..

Johns Hopkins APL Develops Prosthetic Arm with Simscape



Options for reusing CAD parts and assemblies in Simscape.

Model-Based Design :

https://www.mathworks.com/solutions/model-based-design.html

& Embedded Code Generation :

https://www.mathworks.com/solutions/embedded-code-generation.html

➔ Exemple d'un modèle remplaçant la fonction de la boucle de retour avec capteur en modélisant mathématiquement/physiquement le comportant censé être obtenu :

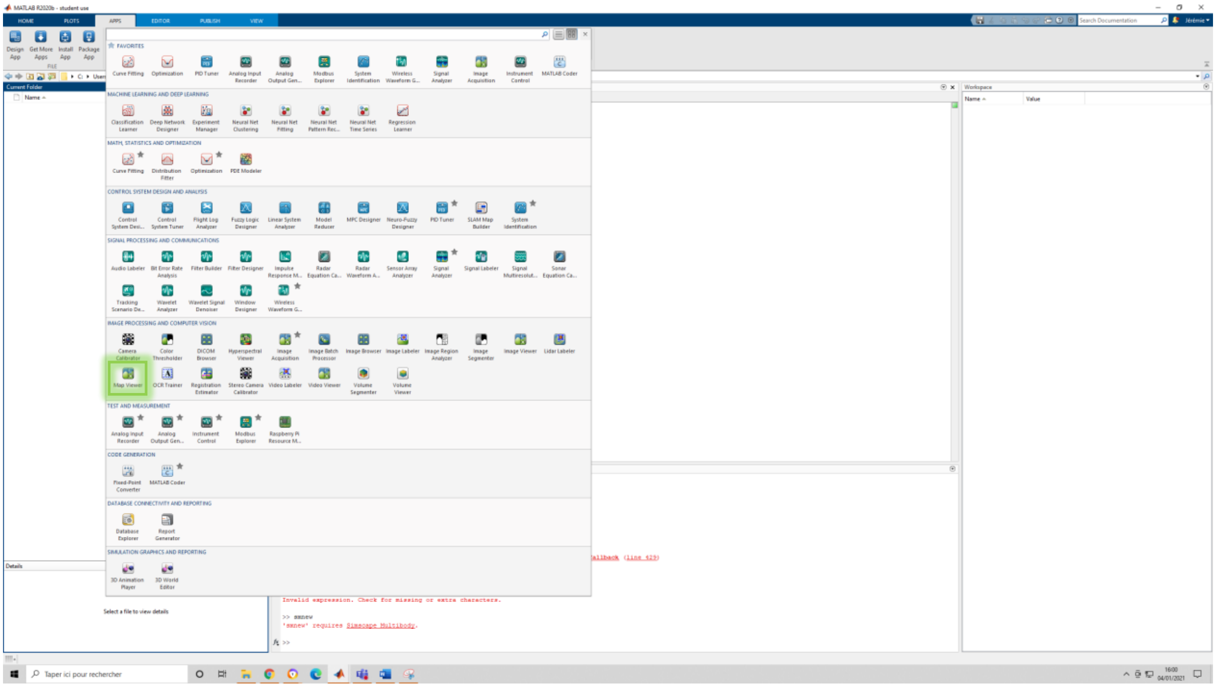
https://www.mathworks.com/videos/b-r-develops-virtual-sensor-technology-to-improve-servo-drive-performance-1490385615790.html

avec : https://www.mathworks.com/company/user_stories/br-industrial-automation-improves-servo-drive-performance-with-virtual-sensor-algorithms-developed-using-model-based-design.html

Mapping avec Matlab-Simulink :

https://www.google.com/search?client=avast&q=mapping+matlab+simulink

Toolboxes en lien :



- Mapping : https://fr.mathworks.com/help/releases/R2020b/map/ref/mapview.html
- Mobile robotics : https://fr.mathworks.com/matlabcentral/fileexchange/66586-mobile-robotics-simulation-toolbox
- Github repo : https://github.com/mathworks-robotics/mobile-robotics-simulation-toolbox

TUTORIEL :

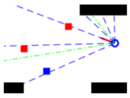
https://fr.mathworks.com/matlabcentral/fileexchange/66586-mobile-robotics-simulation-toolbox

MathWorks Products Solutions Academia Support Community Events

File Exchange

Search File Exchar

MATLAB Central | Files | Authors | My File Exchange | Contribute | About



Mobile Robotics Simulation Toolbox

version 2.2 (8.31 MB) by MathWorks Student Competitions Team STAFF

MATLAB and Simulink utilities for vehicle kinematics, visualization, and sensor simulation.

https://github.com/mathworks-robotics/mobile-robotics-simulation-toolbox

Overview | Functions | Models | Examples

doc

- GettingStarted.mlx
- mrsDocDifferentialDrive.mlx
- mrsDocFourWheelMecanum.mlx
- mrsDocFourWheelSteer.mlx
- mrsDocGenericOmniwheel.mlx
- mrsDocLidarSensor.mlx
- mrsDocMultiRobotEnv.mlx
- mrsDocMultiRobotLidarSensor.mlx
- mrsDocObjectDetector.mlx
- mrsDocRobotDetector.mlx
- mrsDocTripleOmniwheel.mlx
- mrsDocVisualizer2D.mlx

examples/matlab/environment

- mrsExampleVizBasics.mlx
- mrsExampleVizLidar.mlx
- mrsExampleVizObject.mlx

Getting Started with Mobile Robotics Simulation Toolbox

Description

This toolbox provides utilities for robot simulation and algorithm development. This includes:

- 2D kinematic models for robot geometries such as differential drive, three, and four-wheeled vehicles, including forward and inverse kinematics
- Configurable lidar and object detector simulators
- Visualization of robotic vehicles and sensors in occupancy grid maps
- MATLAB and Simulink examples and documentation

System Requirements

This version of the toolbox was developed and tested in MATLAB 9.7 (R2019b).

Product dependencies are:

- MATLAB
- Simulink
- Stateflow
- Robotics System Toolbox
- Navigation Toolbox
- Control System Toolbox + Model Predictive Control Toolbox (optional)
- Reinforcement Learning Toolbox + Deep Learning Toolbox (optional)
- Simscape + Simscape Multibody (optional)

Documentation

Kinematic Models

- Differential Drive
- Triple Omniwheel
- Generic Omniwheel
- Four-Wheel Steering
- Four-Wheel Mecanum

Environment Models

- Robot Visualizer
- Multi-Robot Environment

Sensor Models

- Lidar Sensor
- Multi-Robot Lidar Sensor
- Object Detector
- Robot Detector

Examples

For simple examples outlining usage of individual models, sensors, or visualizers, please refer to the documentation links above.

Below is a list of all the application examples, which can be found in the `examples` directory of the toolbox.

MATLAB Examples

- Waypoint following using the Pure Pursuit Algorithm (Differential Drive)
- Waypoint following using the Pure Pursuit Algorithm (Four-Wheel Steering)
- Path planning and following of a differential drive robot
- Path following and obstacle avoidance of a differential drive robot
- Multi-robot swarm behavior
- Model Predictive Control of three-wheel omniwheel robot
- Path planning and following for a car-like vehicle using RRT

Simulink Examples

- Waypoint following using the Pure Pursuit Algorithm (Differential Drive)
- Waypoint following using the Pure Pursuit Algorithm (Four-Wheel Steering)
- Waypoint following using supervisory logic
- Waypoint following and obstacle avoidance
- Object tracking logic
- Multi-robot obstacle avoidance
- Model Predictive Control of three-wheel omniwheel robot

Application Examples

- Robot Soccer Simulation
- Navigation Behavior with Vector Field Histogram (VFH) Obstacle Avoidance
- Navigation Behavior with Reinforcement Learning Based Obstacle Avoidance

Mise en œuvre Raspberry Pi ⇔ Matlab :