

モデルベースデザインではじめる自律型アームロボットの開発

MathWorks Japan
アプリケーションエンジニアリング部
小林 昇洋

本セミナーでお伝えしたいこと

自律ロボットシステムの開発のキーポイント:

1. マルチドメインシミュレーション
2. 複雑な開発要素をトータルでサポートする開発環境
3. モデルベースデザイン

自律ロボットシステムの開発における課題

複数専門領域にまたがる開発

アルゴリズムの複雑さ

最新技術の製品への適用

システムの安定性と高度技術の両立

IPの保護

具体的には・・・



モデルベースデザインによる人型の自律ロボットの開発

German Aerospace Center (DLR) Robotics and Mechatronics Center

課題

2アーム53自由度のモバイル人型ロボットの制御システムの開発

解決策

MATLAB/Simulinkを用いたモデルベースデザイン

- ・机上シミュレーションの実現
- ・コード生成機能を以下用途で活用
 - リアルタイムテスト: Hardware in the Loop
 - アルゴリズム開発: リアルタイム制御、軌道最適化、センサの自動キャリブレーション

結果

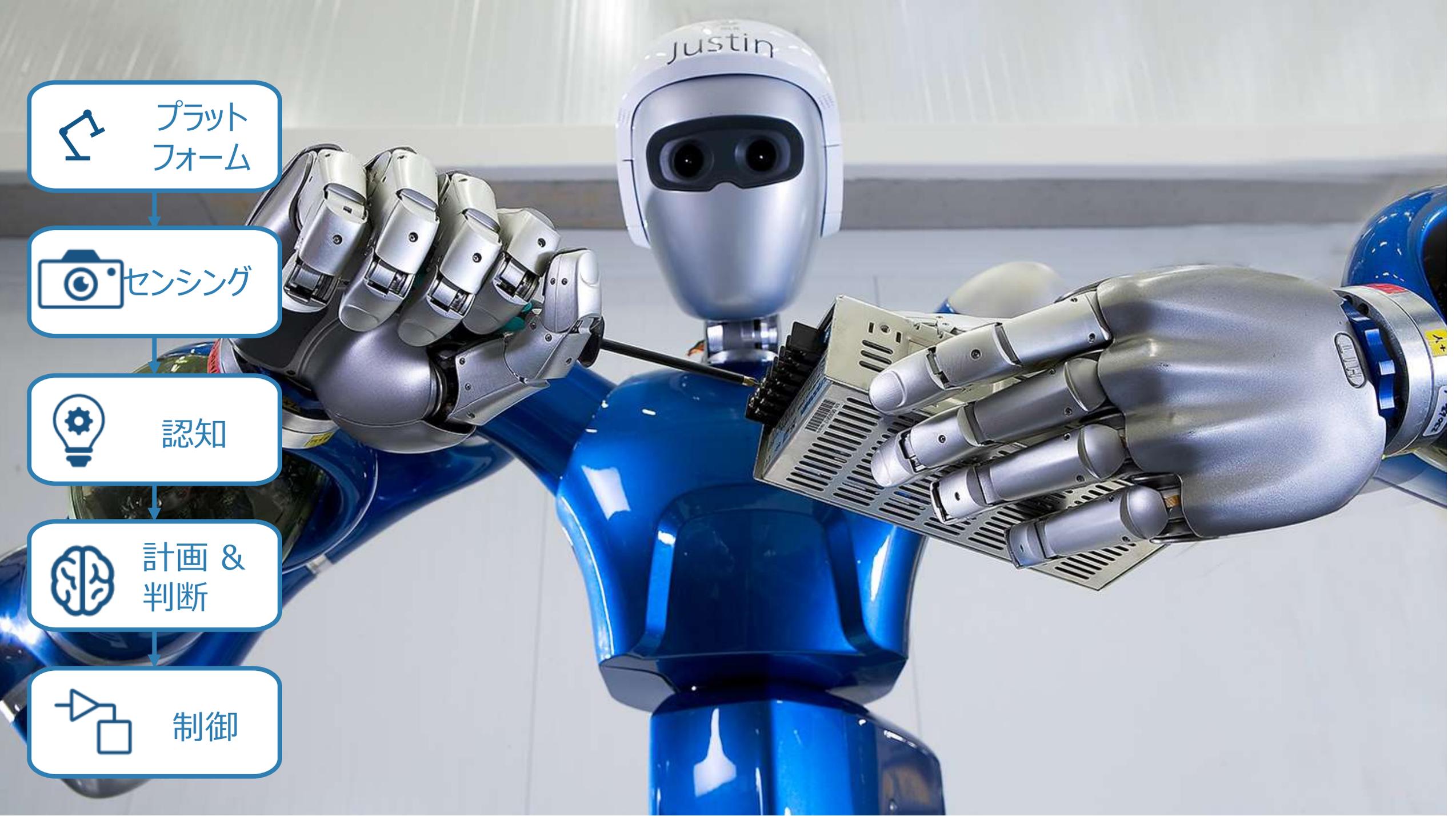
- ・ソフトウェアバグの排除
- ・複雑なアルゴリズムの実装を数時間で実現
- ・アドバンスな制御開発を学生でも開発可能にした



DLR's humanoid robot Agile Justin autonomously performing a complex construction task.

“Model-Based Design and automatic code generation enable us to cope with the complexity of Agile Justin’s 53 degrees of freedom. Without Model-Based Design it would have been impossible to build the controllers for such a complex robotic system with hard real-time performance.”

Berthold Bäuml
DLR



プラット
フォーム



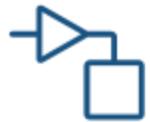
センシング



認知

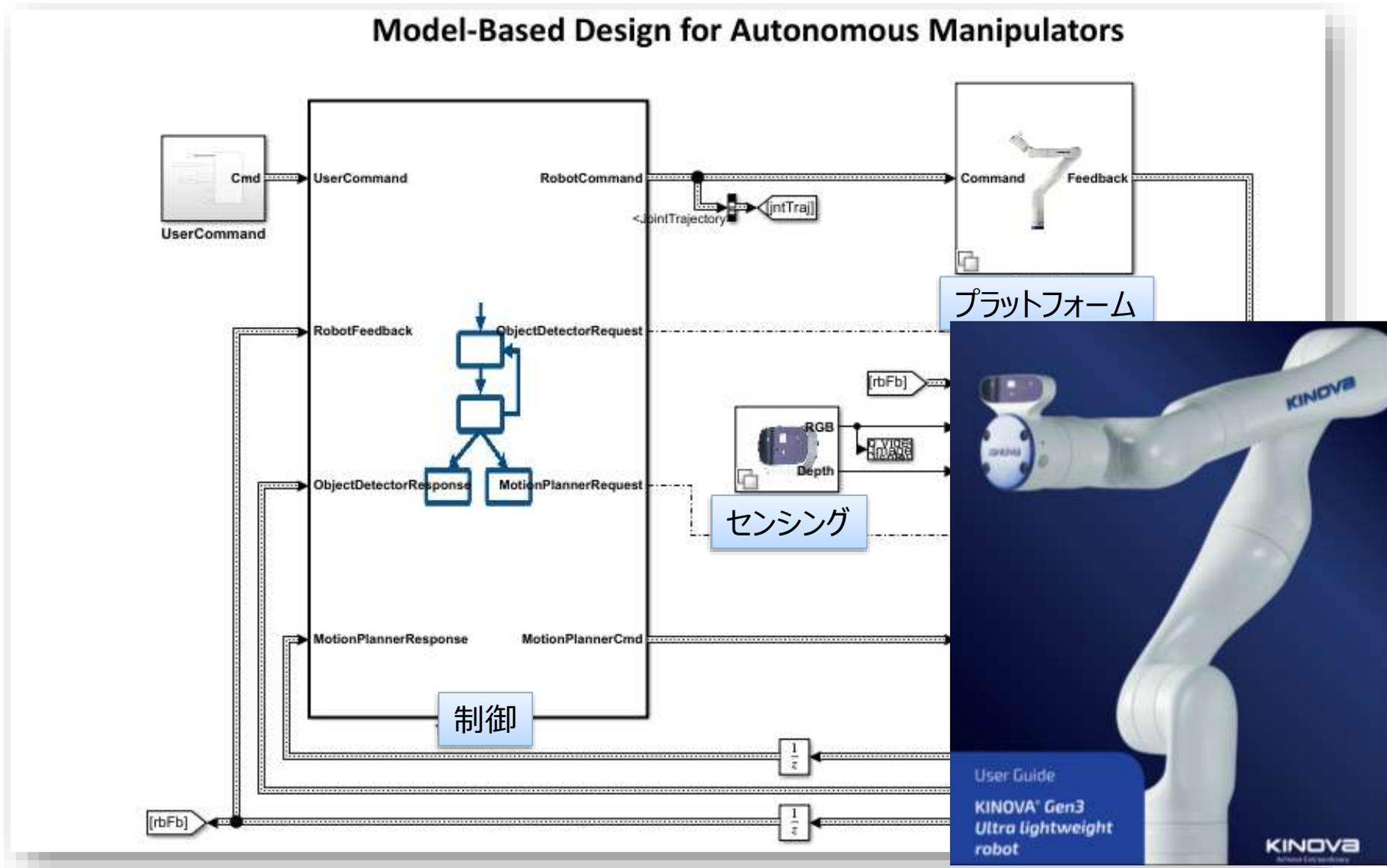
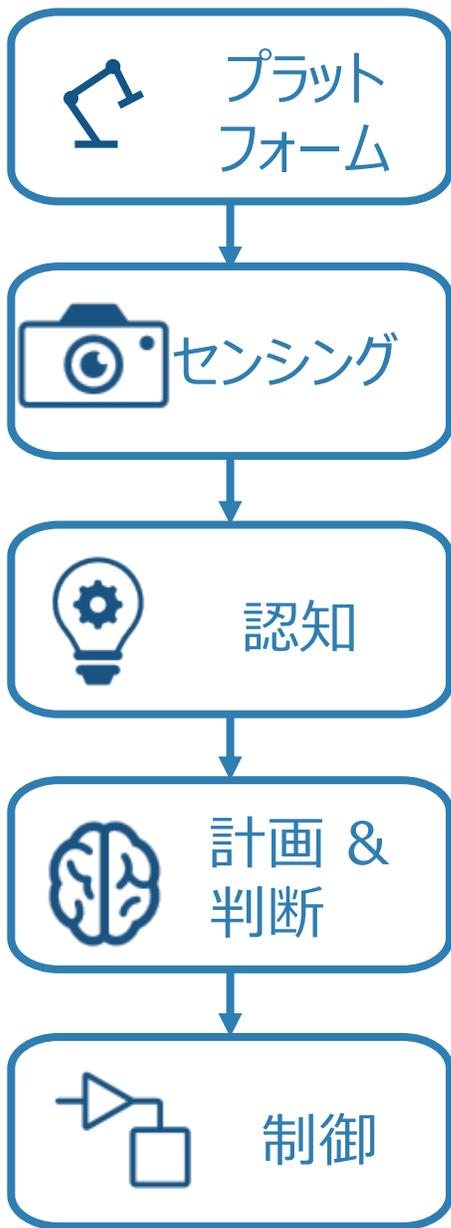


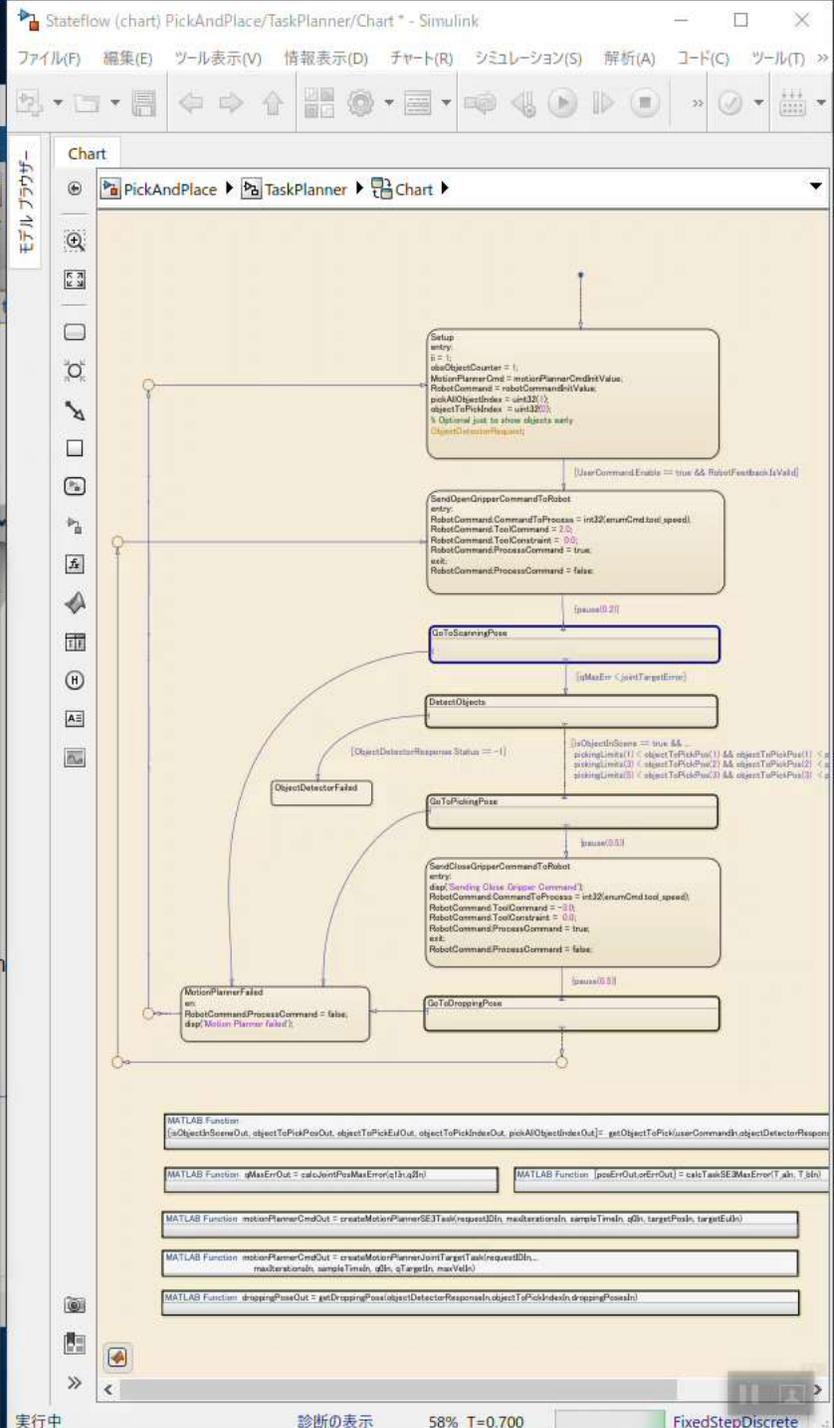
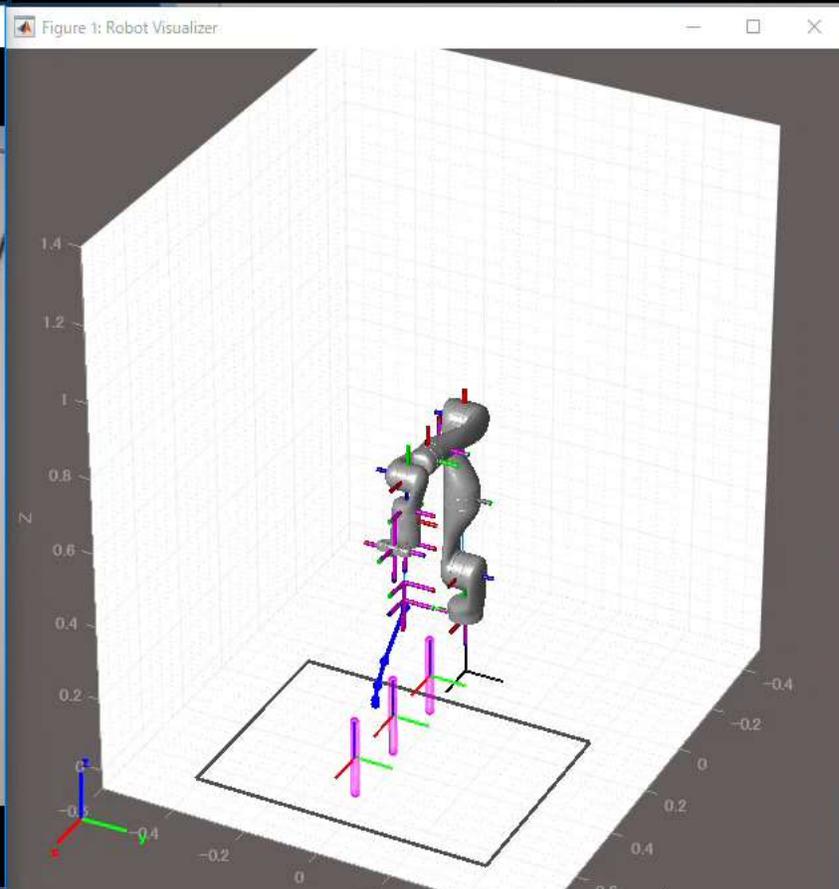
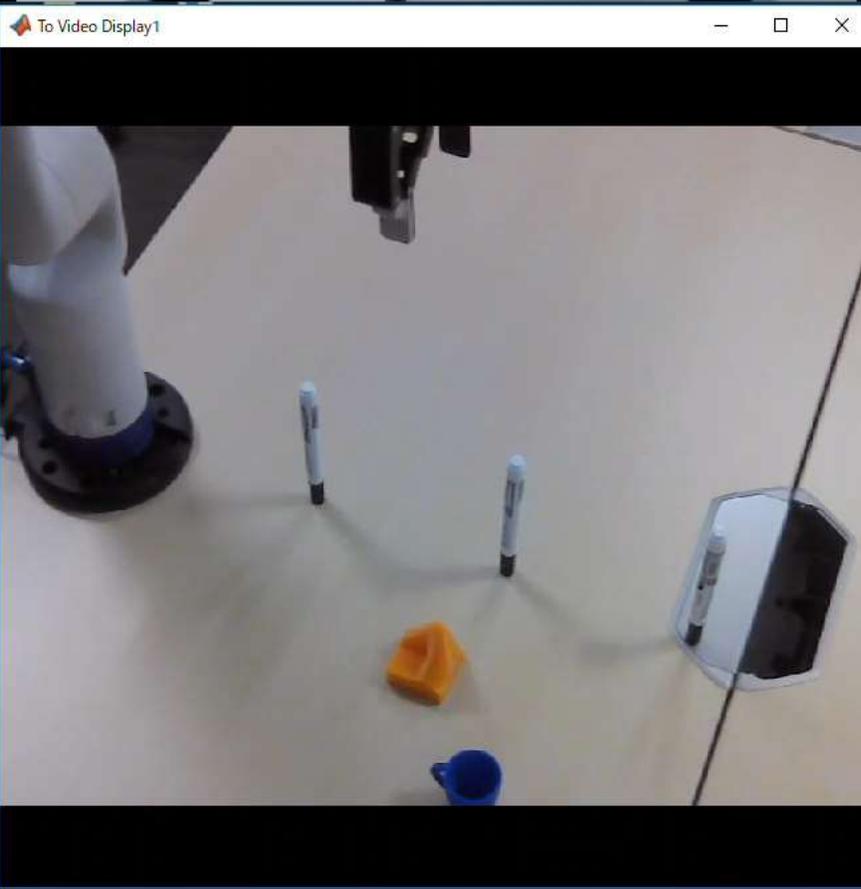
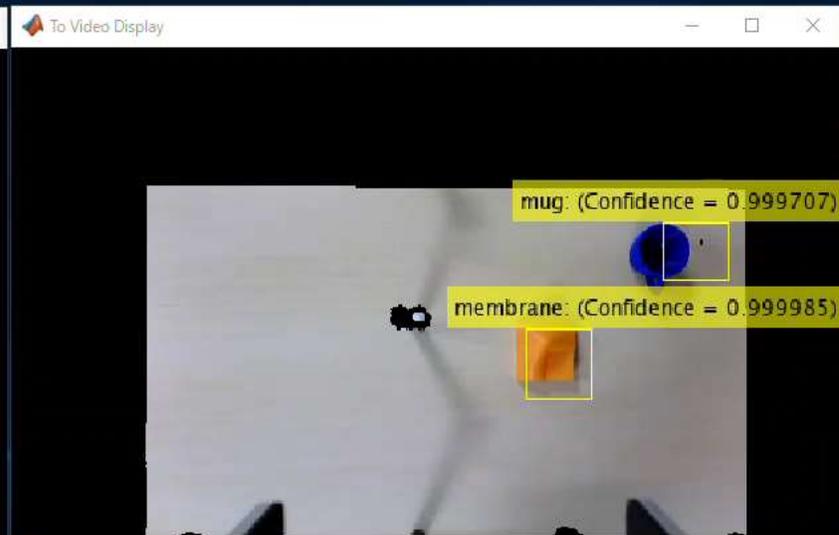
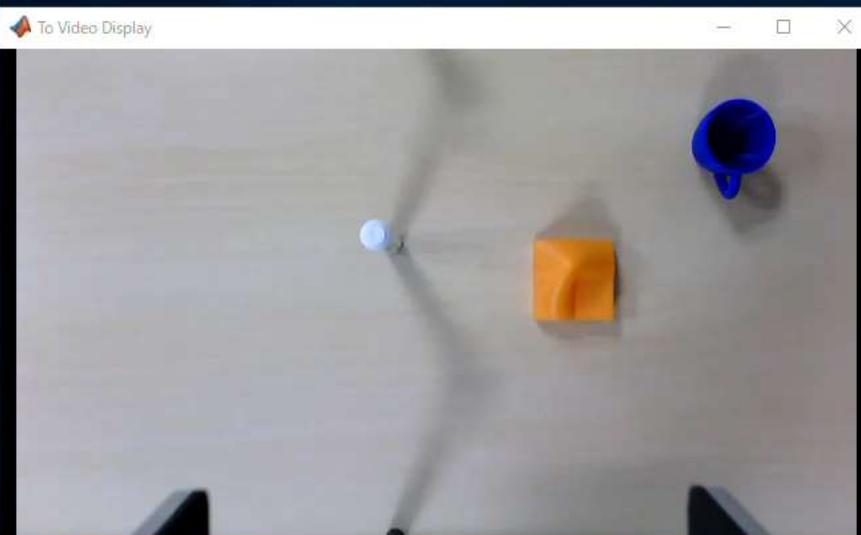
計画 &
判断



制御

本日はPick and Placeシステムを取り上げます





プラットフォームの設計

対象とするシステムの設計に使いやすいモデルの構築

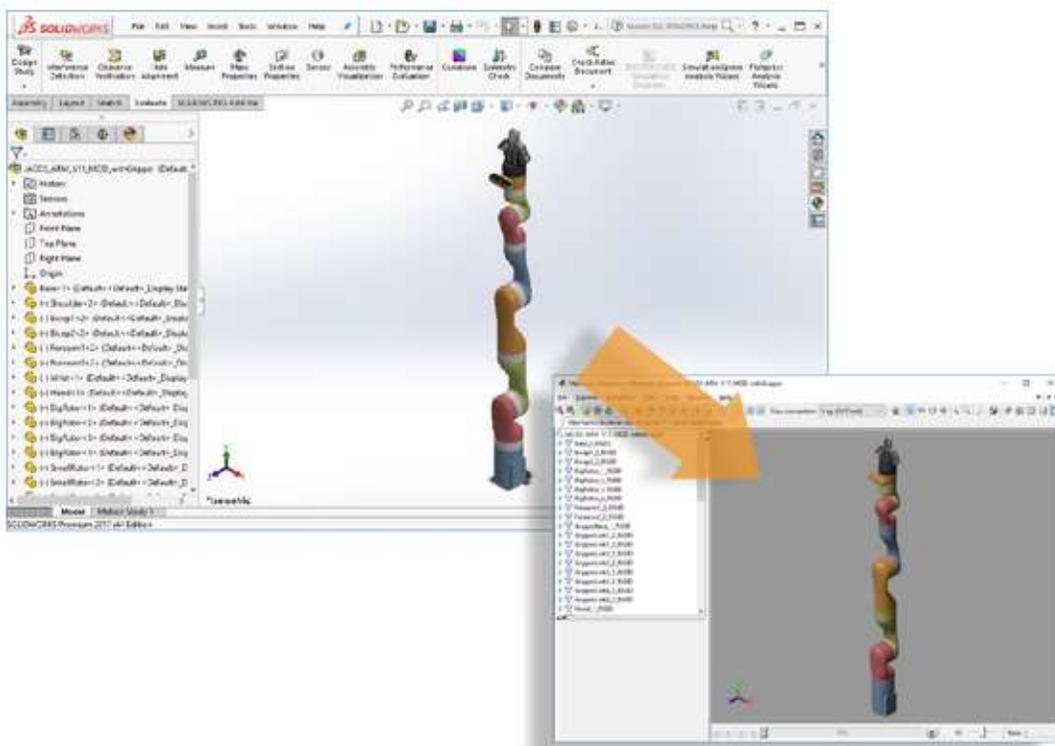
メカニクス

アクチュエータ

環境

メカニクス: CADからのインポートによるモデル構築

SolidWorks



Simscape Multibody™

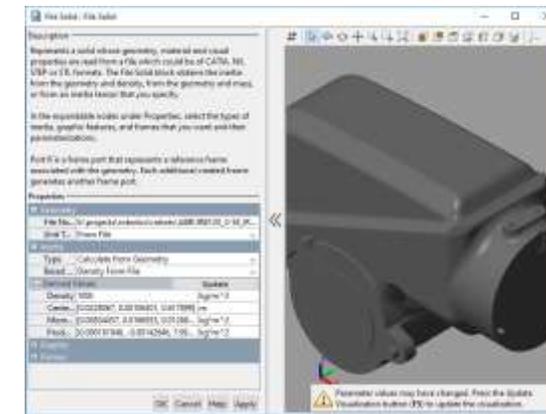
運動学、動力学など
マルチボディシミュレーションを実現

拘束を含むアセンブリ単位でのインポート

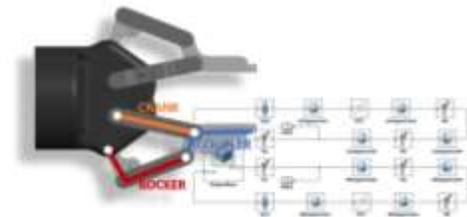
- SolidWorks
- Autodesk Inventor®
- PTC® Creo™(Pro/ENGINEER®)
- Onshape®

partファイル単位でのインポート

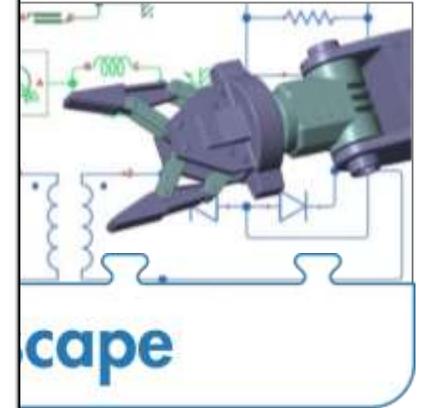
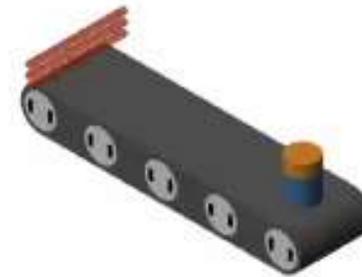
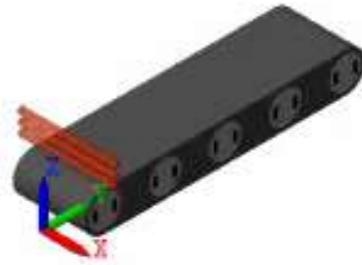
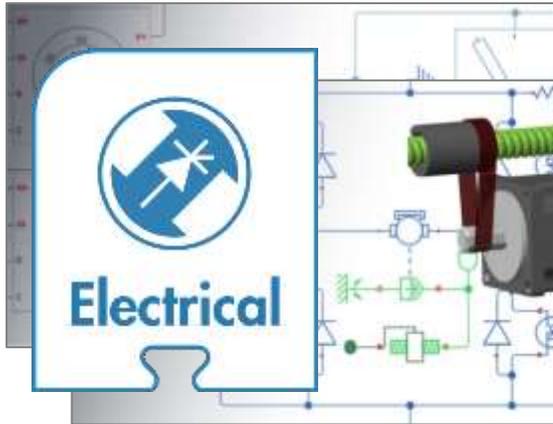
- CATIA V4,V5,V6
- Siemens NX
- SolidEdge
- SolidWorks
- JT
- Inventor
- ProE/Creo
- Parasolid
- ACIS



パラレルリンク機構の解析に対応



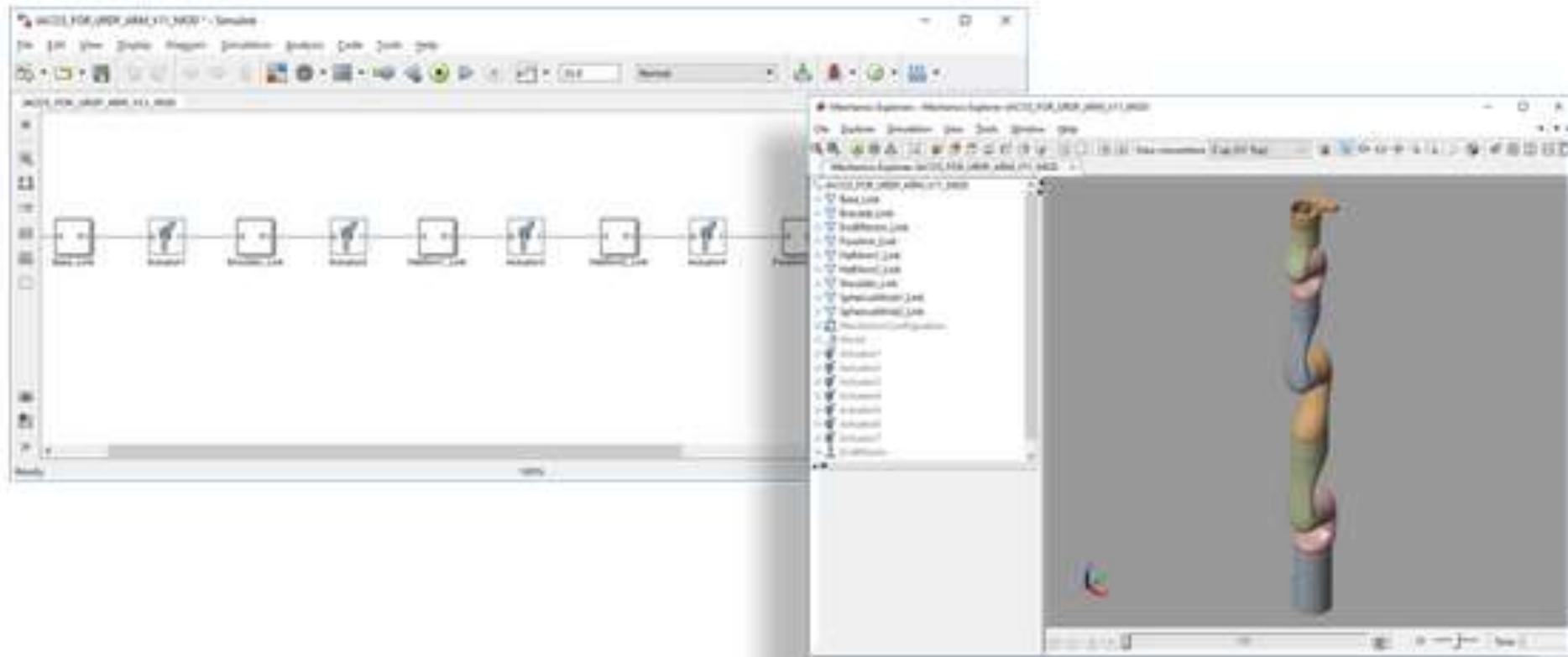
アクチュエータ: メカ以外のドメインを含むシステムシミュレーション



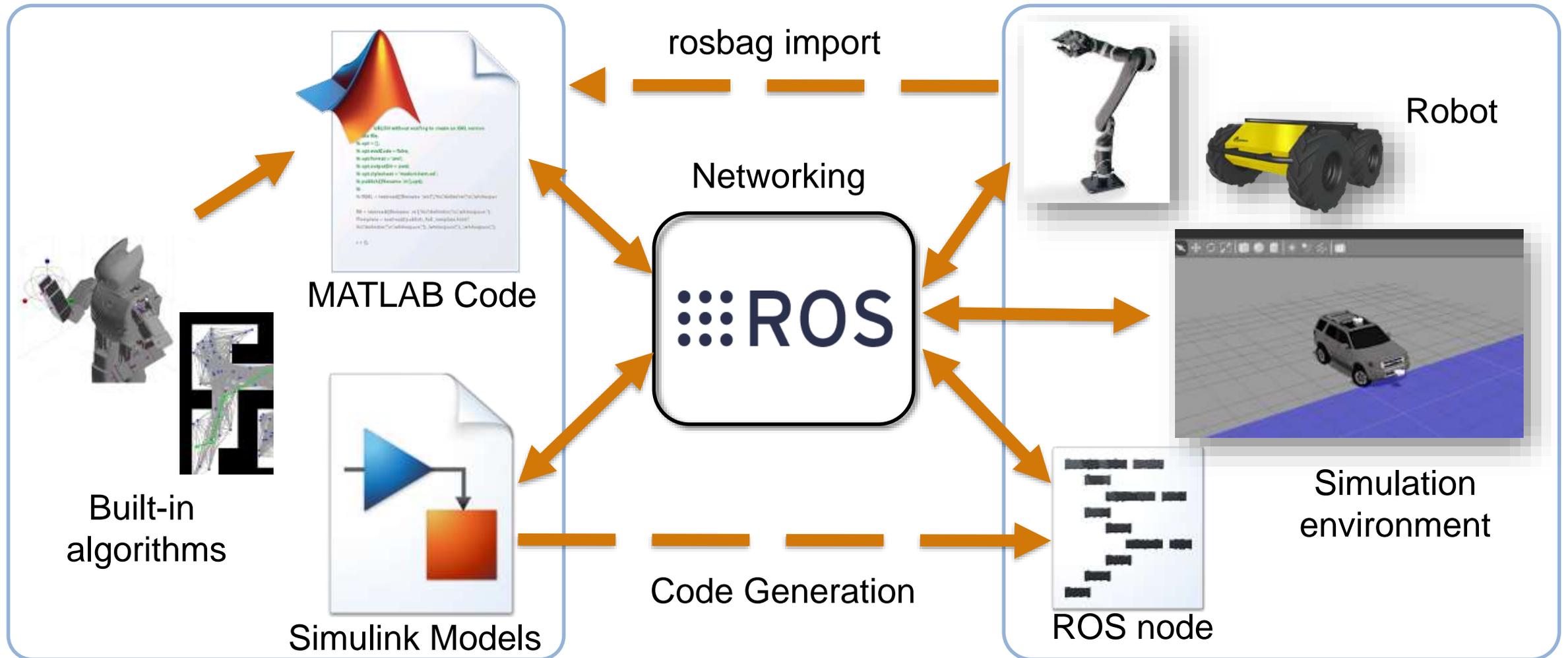
メカニクス: URDFからのインポート

```
%% Import robot from URDF
```

```
smimport('GEN3_FOR_URDF_ARM_V11_MOD.urdf');
```



環境モデル: MATLAB と Simulink を ROS に接続

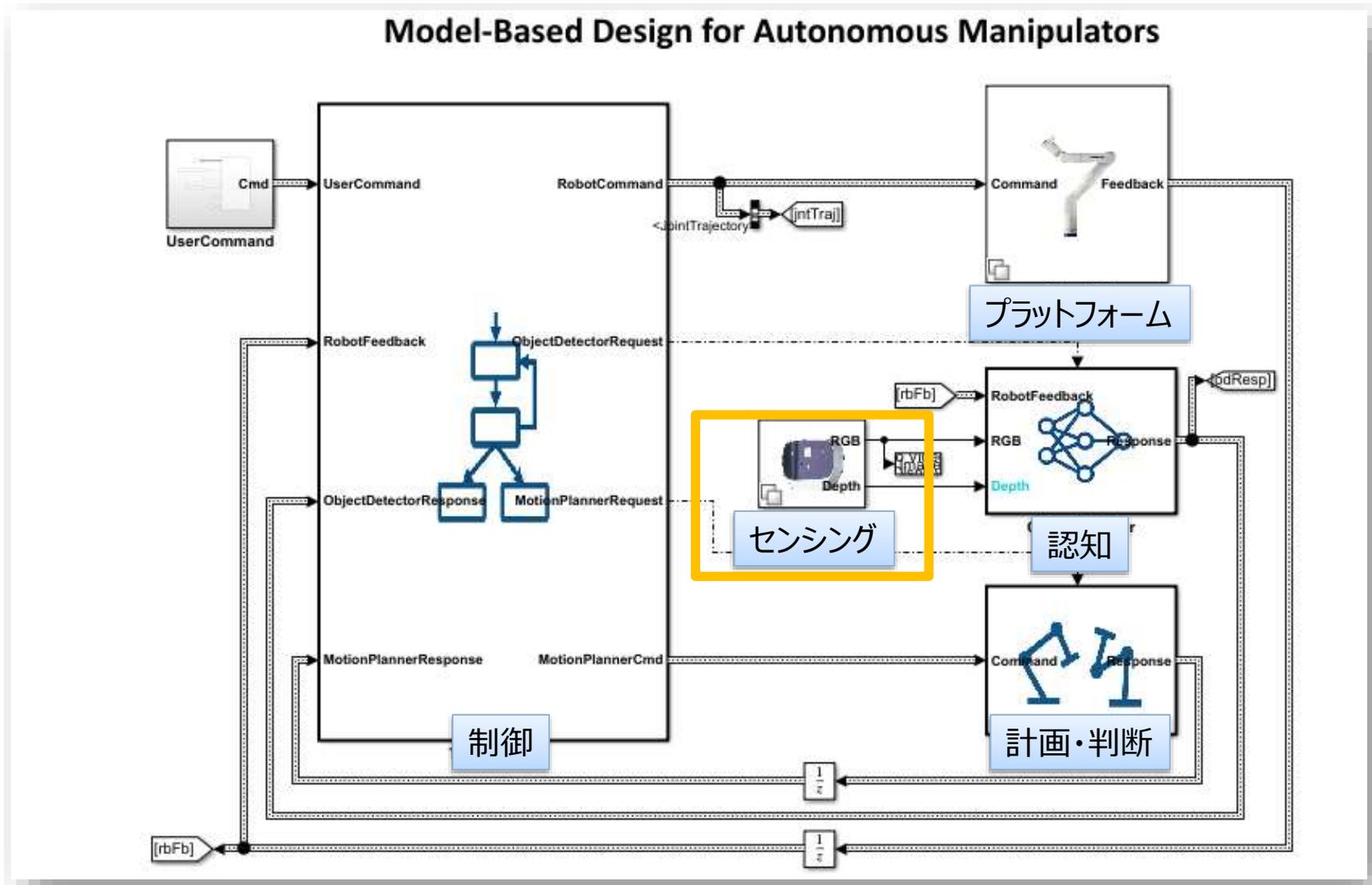
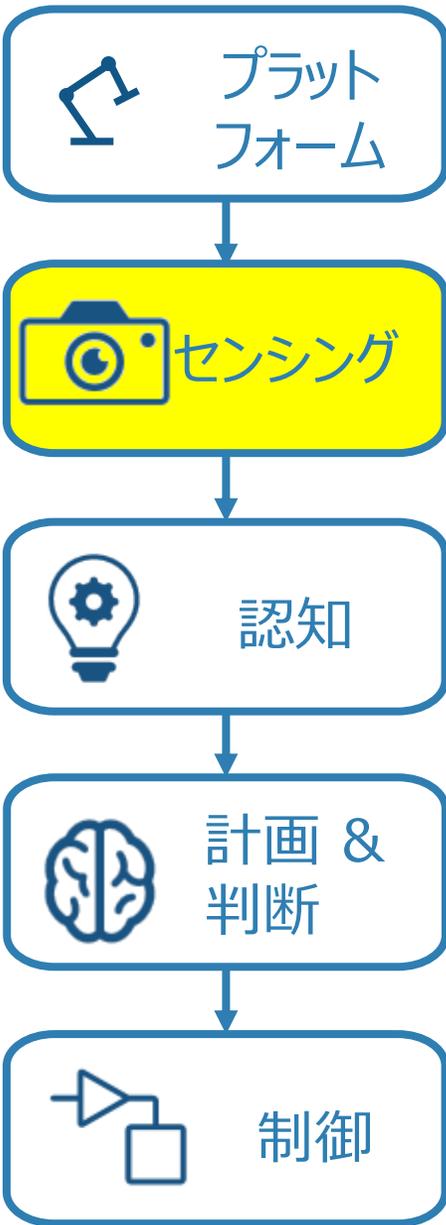


環境モデル: 外部のロボティクスシミュレータとの統合

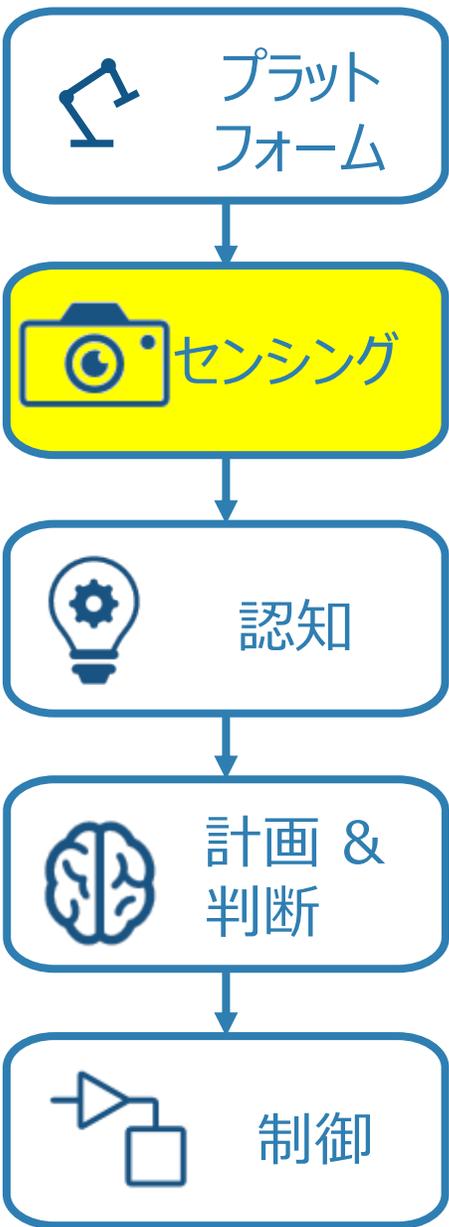
The image displays a Simulink environment integrated with Gazebo 7. The Simulink window, titled "mainController * - Simulink", shows a control loop between a Motion Planner and a State Controller. The Motion Planner block receives a "request" (containing $\min f(x)$ and $qAct$) and outputs a "response" (containing "plannerResponse" and "plannerCommand"). The State Controller block receives "userCommands" and "objectDetected1" and outputs an "armCommand" to the Robot ROS Interface. The State Controller also outputs "goalReached" and "goalSpline" (double [5x1]). The Robot ROS Interface block receives "qAct" and "nStateValid" and outputs "armCommand" and "goalReached" back to the State Controller. A "Cloud prediction" block outputs "objectDetected1" to the State Controller. A "User Command" block outputs "userCommand" to the Motion Planner. An "Object To Grab" block outputs "objectToGrab" to the Motion Planner. A "To Video Display1" window shows a 3D view of the robot arm with a table and objects. The Gazebo window shows the 3D environment with the robot arm and a table. The status bar at the bottom of the Simulink window indicates "50% T=23.000" and "FixedStepDiscrete".

ハードウェアから外部シミュレータまでの複合領域の連携サポート

Pick and Place システムの設計



Pick and Place システムの設計



- 標準的なセンサーのサポート
- 画像解析
- Apps
- 画像の強調
- ポイントクラウドの可視化

サードパーティ製品/サービス

Kinova
Versatile and ultra-lightweight innovation robotics for enterprises, researchers, and end users

Highlights

- Easily integrated and highly adaptable robotic arm manipulators, with up to 7 DOF
- Sensors for torque, position, current, temperature, and acceleration in each actuator
- Torque, position, or velocity control through embedded controllers
- Intuitive Cartesian control software solution provided
- Aesthetic design in either carbon fiber or reinforced plastic structure
- Open architecture with the Kinova[®] SDK (on Windows[®], Linux[®]) and ROS enabled

Description

Kinova's operator-centric service robotics are built to assist and complement their operators at all times, rather than marginalizing human capability. The KINOVA[®] ultra lightweight series includes easy-to-use, power-efficient robot arms with an open architecture, ROS compatibility, and software tools for maximum performance and flexibility. The company is rooted in augmentation of human capabilities; their robots are designed to be controlled by humans and to adapt to their specific working environment and conditions, not the other way around. Their robotic arm manipulators enable new robotic applications to optimize research capabilities across multiple industries. Researchers at more than 300 universities around the world rely on Kinova solutions for a wide range of tasks, from SCI control to humanoid manipulators.

A MATLAB[®] wrapper on the Kinova API allows users to connect to the robots from within MATLAB, and query the status of robot components (fingers, arm joints, etc.), including properties such as position and velocity. Users can also send commands to control and move the robots. The KINOVA Gen3 Ultra lightweight robot is supported with MATLAB. To learn about MathWorks using Kinova product case study, click here.

Stay informed by monitoring the GitHub repository, and our website.

KINOVA

Kinova
4333, Boulevard de la Grande-Allée
Boisbriand, Quebec, J7H 1H7
CANADA
Tel: 514-277-3777
support@kinova.ca
http://www.kinova-robotics.com

Required Products

- MATLAB

Platforms

- Linux
- Windows

Support

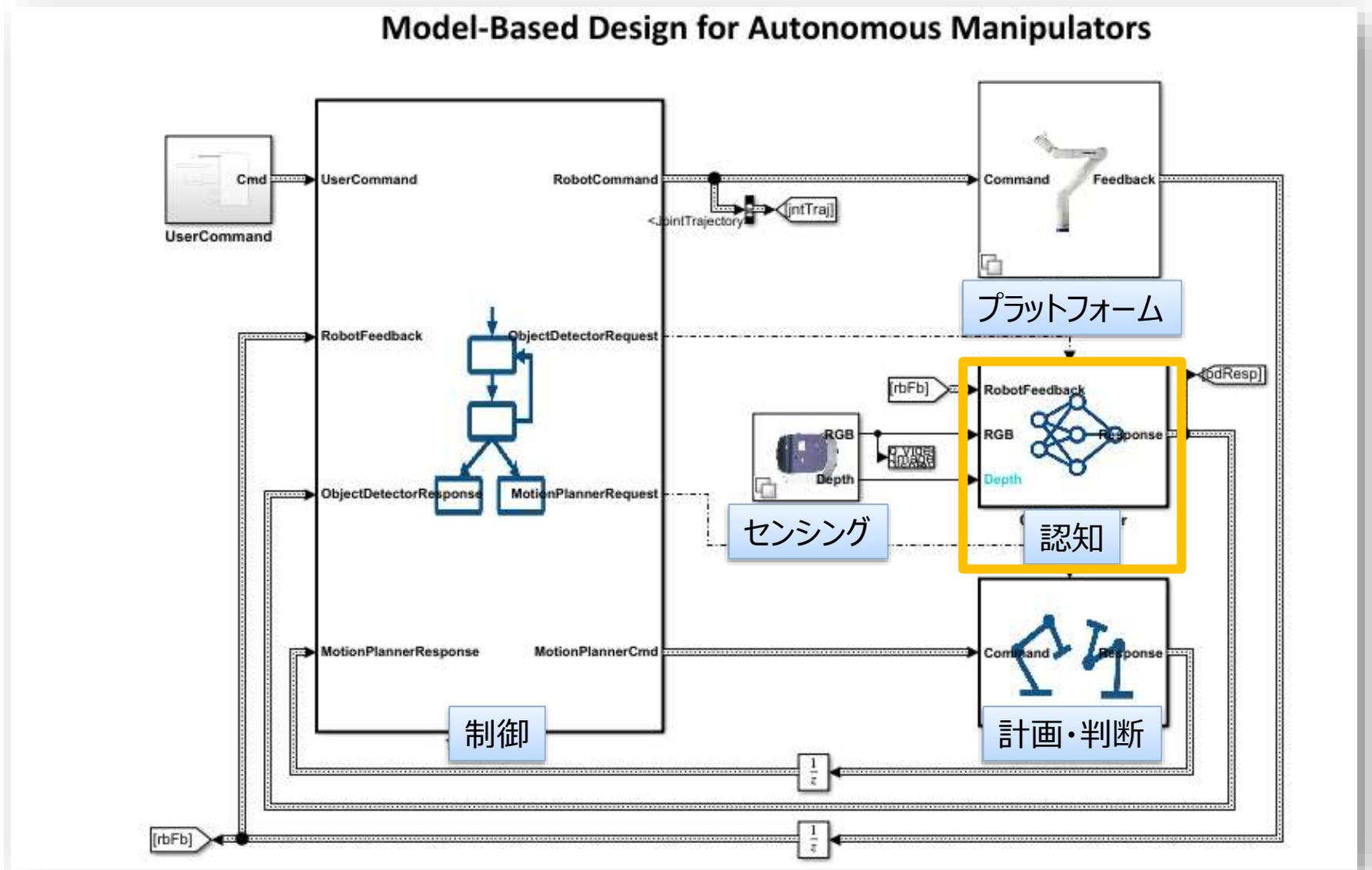
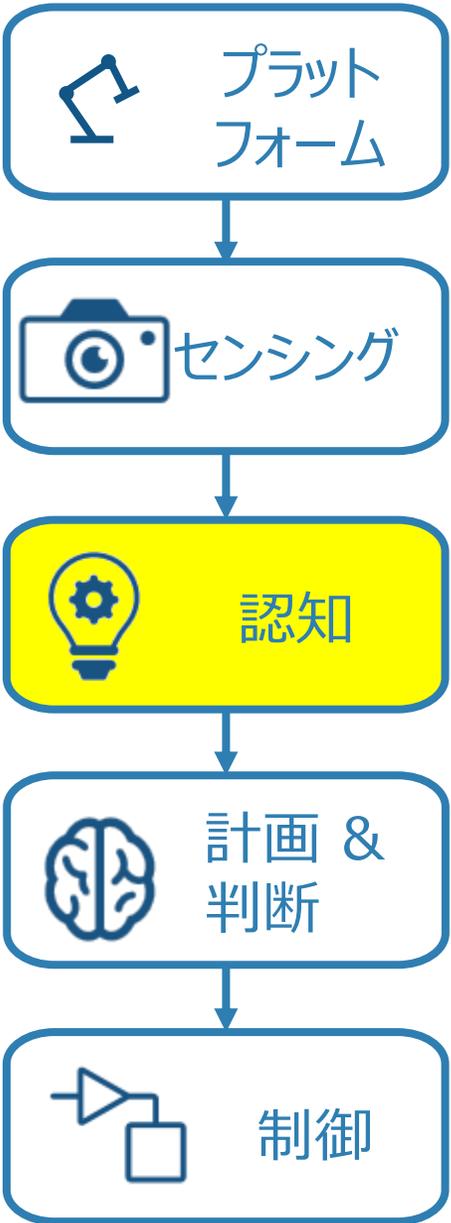
- E-mail
- System integration
- Telephone
- Training

Product Type

- Lab Experiments
- Robotics Systems

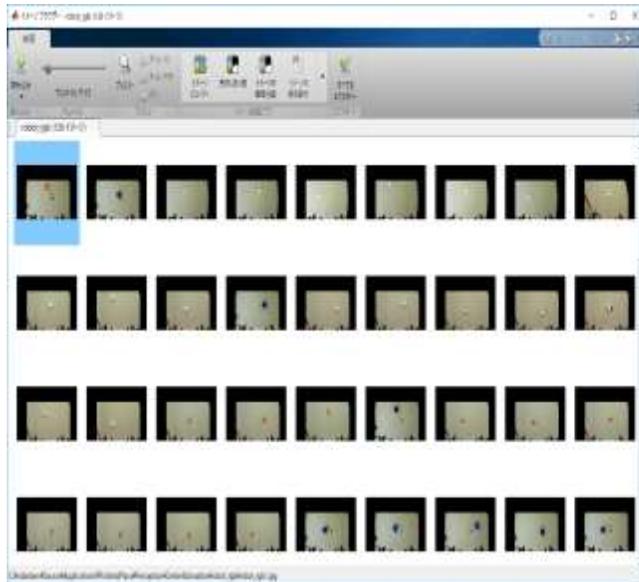
User Guide
KINOVA[®] Gen3 Ultra lightweight robot

Pick and Place システムの設計



オブジェクトの分類と姿勢の推定

画像



分類

分類

Object 1



Object 2



Object 3



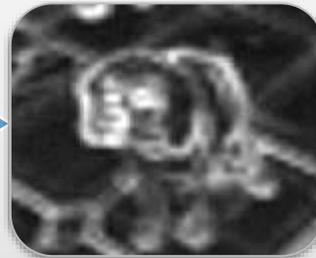
Object 4

MATLAB を使って機械学習のアルゴリズムも簡単に導入

従来型の機械学習のアプローチ



特徴抽出



分類

機械学習

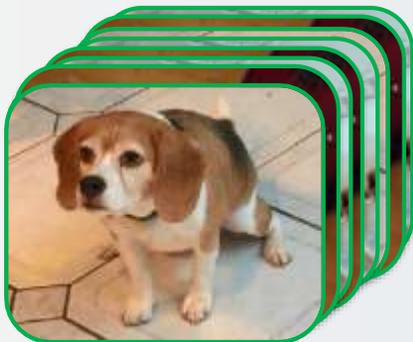
犬 ✓

男の子 ✗

●
●

自転車 ✗

ディープラーニングのアプローチ



畳み込み Neural Network (CNN)

Learned features

[95%]

End-to-end learning

特徴の学習 + 分類

犬 ✓

男の子 ✗

●
●

自転車 ✗



使いやすさを追求したMATLABでのディープラーニング

ネットワーク作成、ラベリング用アプリ

Deep NetworkDesigner



Image Labeler



動画や三次元に対応

ボリューム画像



分類やセグメンテーション

ビデオの分類



“腕立て”

他のフレームワークとのインターフェイス

Keras

Caffe MODELS

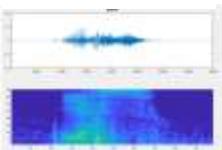
TensorFlow

ONNX

- モデルのインポート/エクスポート機能での連携
- 10種類以上の事前学習済みモデルを提供

様々なアプリケーションへ適用

音声



文章



Yolo v2による物体検出

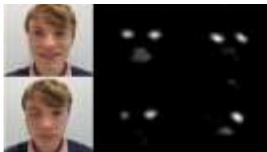


セマンティックセグメンテーション



豊富な可視化機能で脱ブラックボックス

中間層の可視化



クラスアクティベーションマッピング



Deep dream



学習曲線



システムへの展開
エッジ&クラウドへ

ARM® Compute library



Intel® MKL-DNN



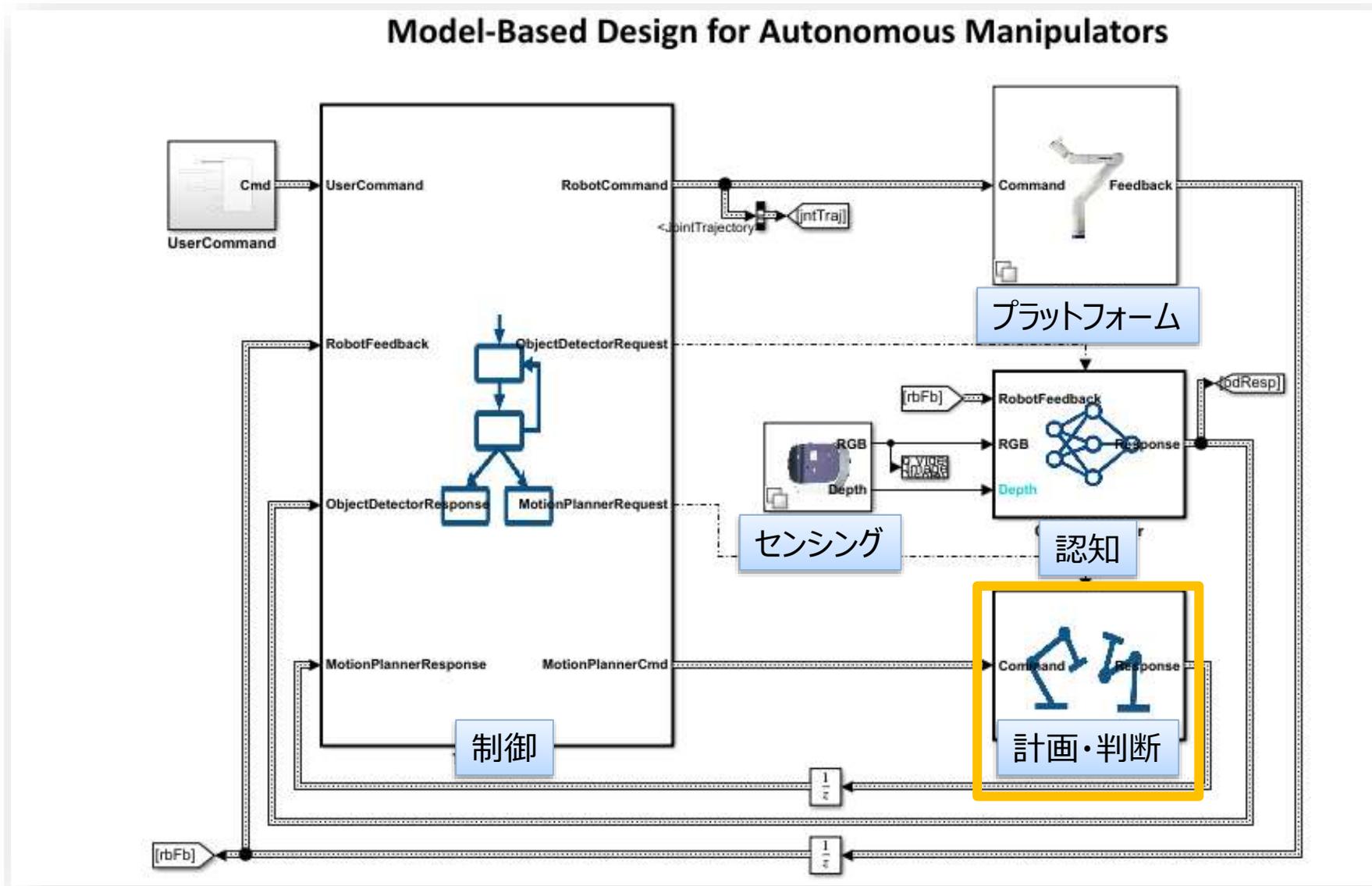
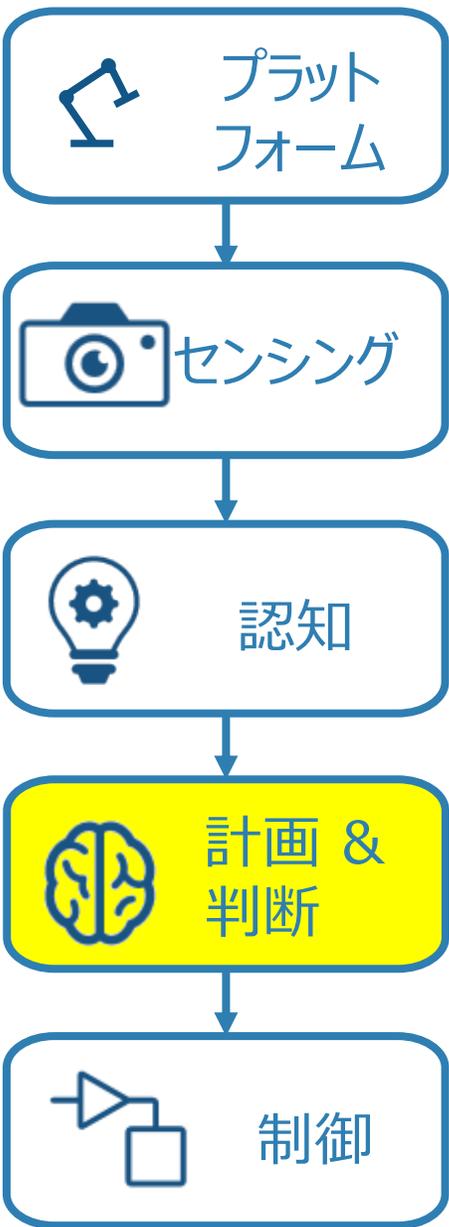
NVIDIA® GPU cuDNN / Tensorf



PC・クラウド・サーバー



Pick and Place システムの設計

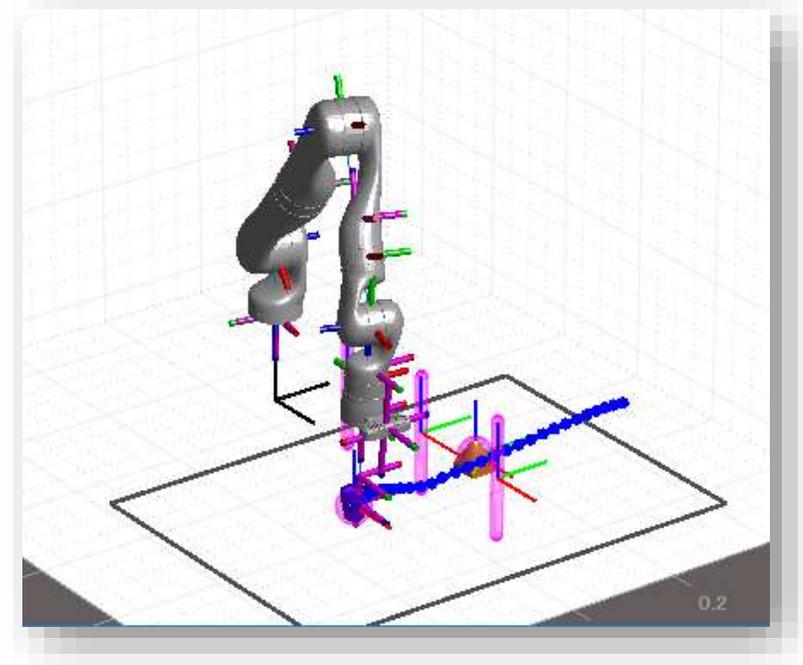
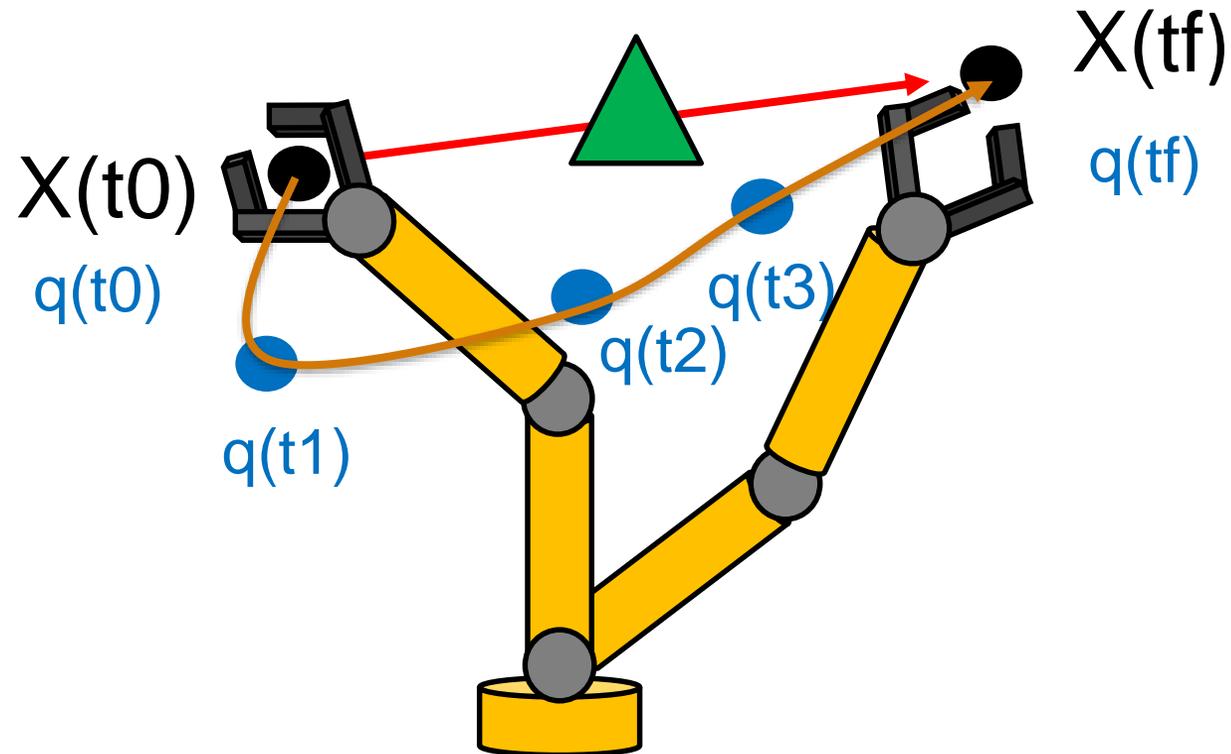


計画問題: 経路探索

Map(障害物)など各種制約
初期姿勢
最終姿勢

パス
プランニング

先端部の軌道
各関節の軌道

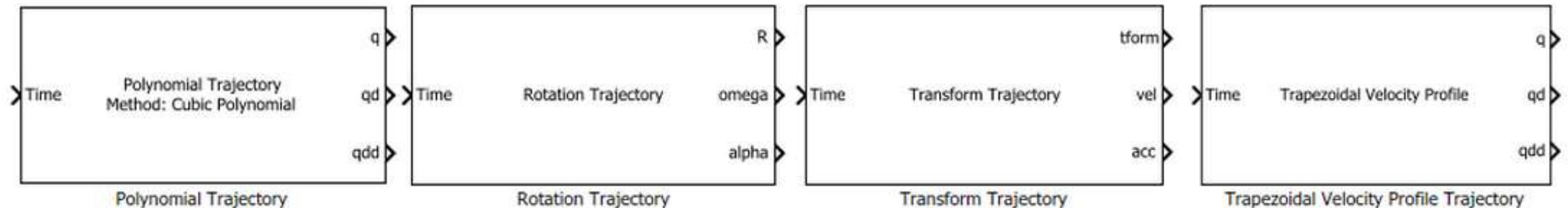
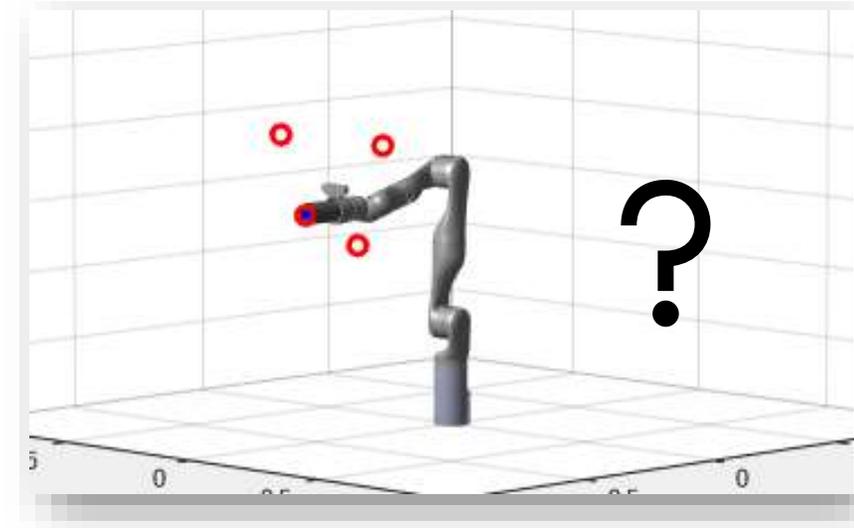


Robotics System Toolbox の 軌道計画アルゴリズム

経由したい座標は決まっているが...

その他の制約条件の例

- 経由したい点で停止する
- 座標間を直線的に移動しはみ出さない
- はみ出しを許容するが振動を引き起こさない
- 先端アームの姿勢に対する制約



多項式補間

姿勢の補間

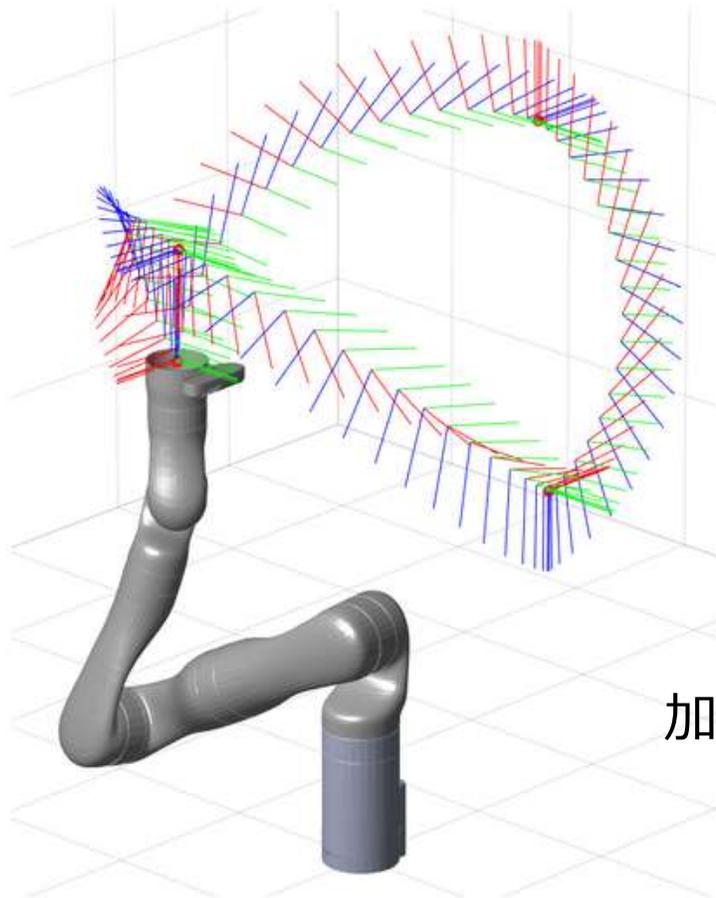
同次変換行列間の補間

台形プロファイルの生成

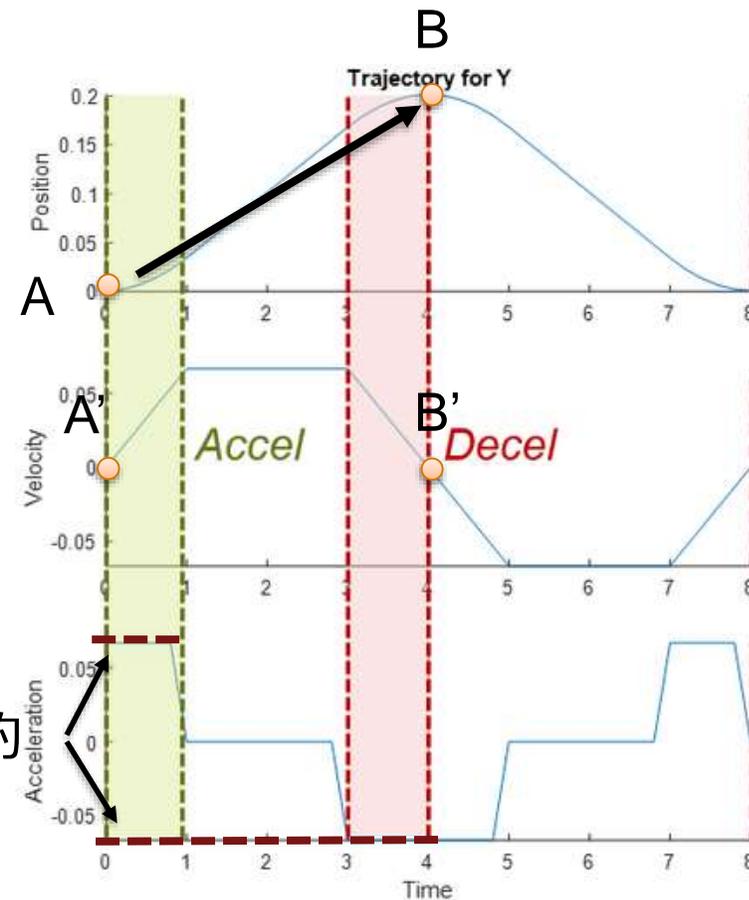
Robotics System Toolbox の軌道計画アルゴリズム

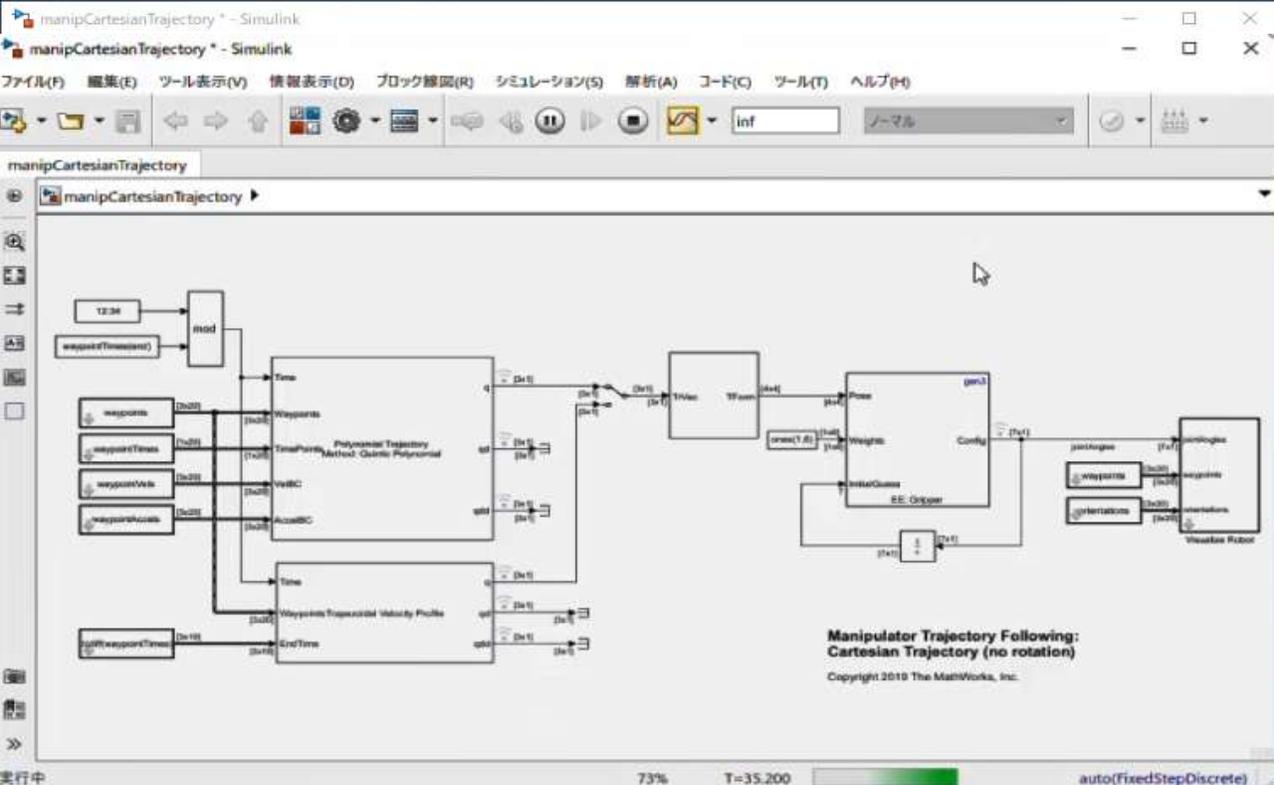
R2019a

例：台形プロファイルの生成による加速度制約

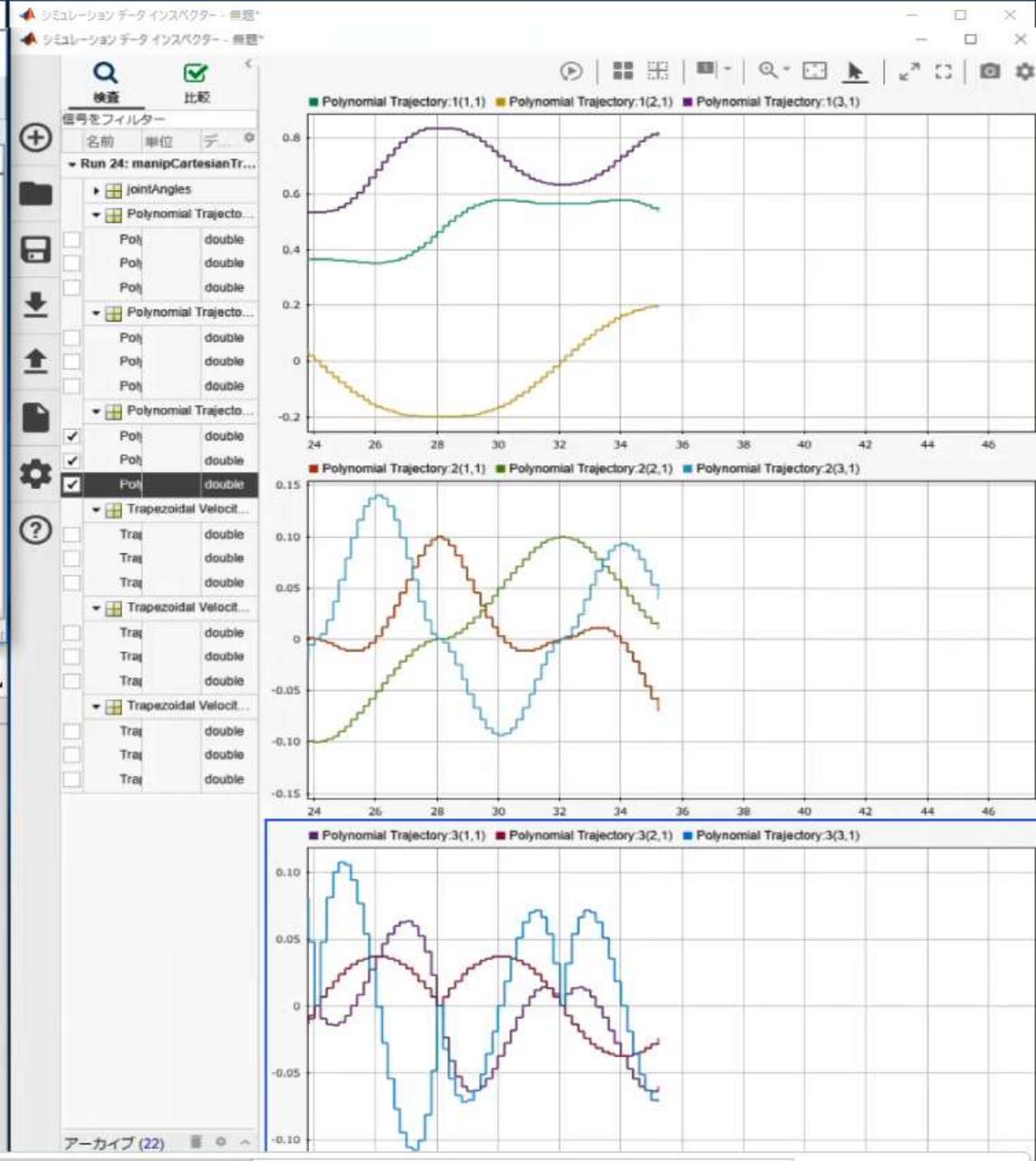
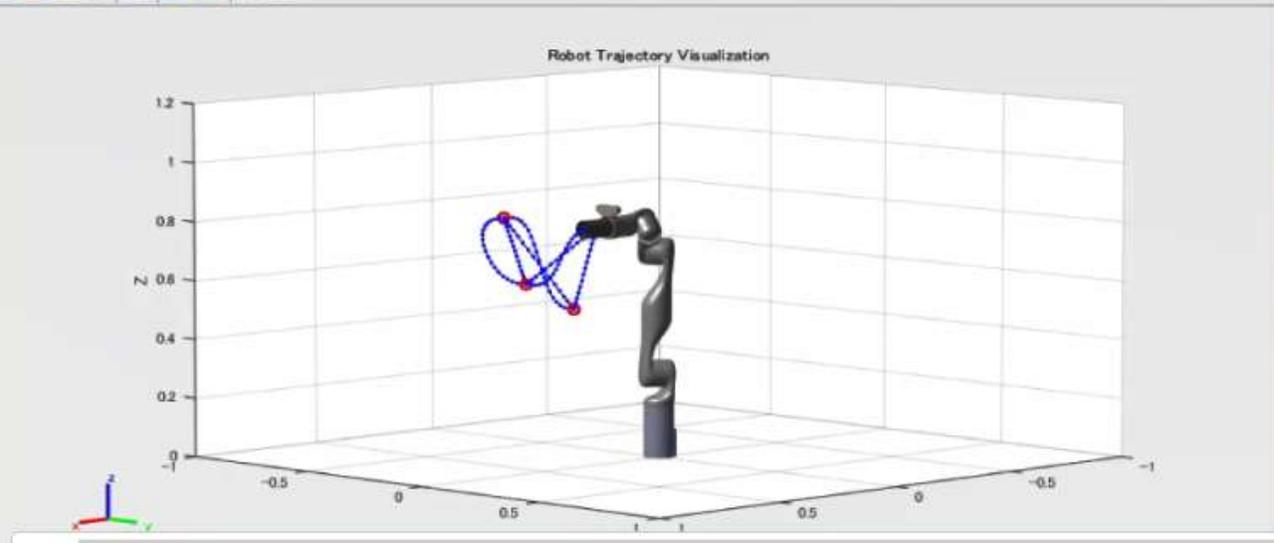


加速度制約





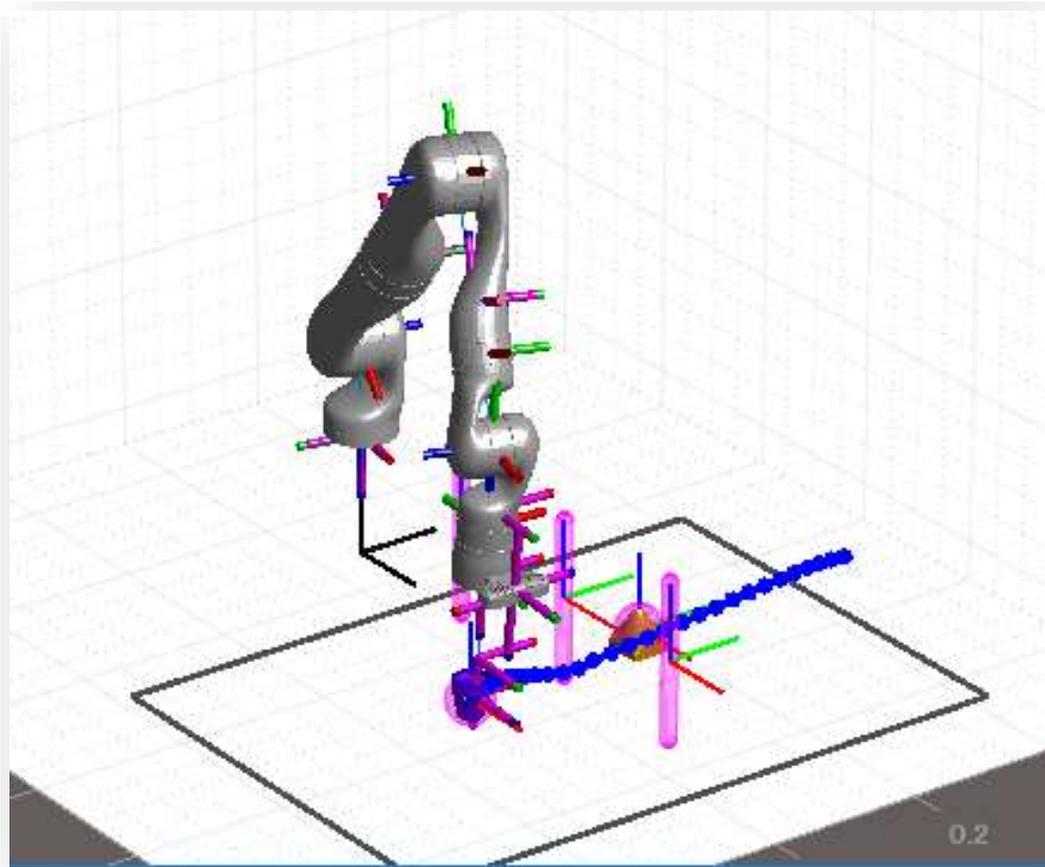
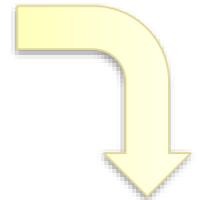
実行中 73% T=35.200 auto(fixedStepDiscrete)



二次計画法に基づく最適軌道の探索

制約条件を二次計画問題へ

- できる限り早く目標地点まで移動
- 角速度制約の範囲を超えない
- 障害物にぶつからない
- アームの各関節が可動範囲を超えない
- 速度が連続的に変化すること



quadprog

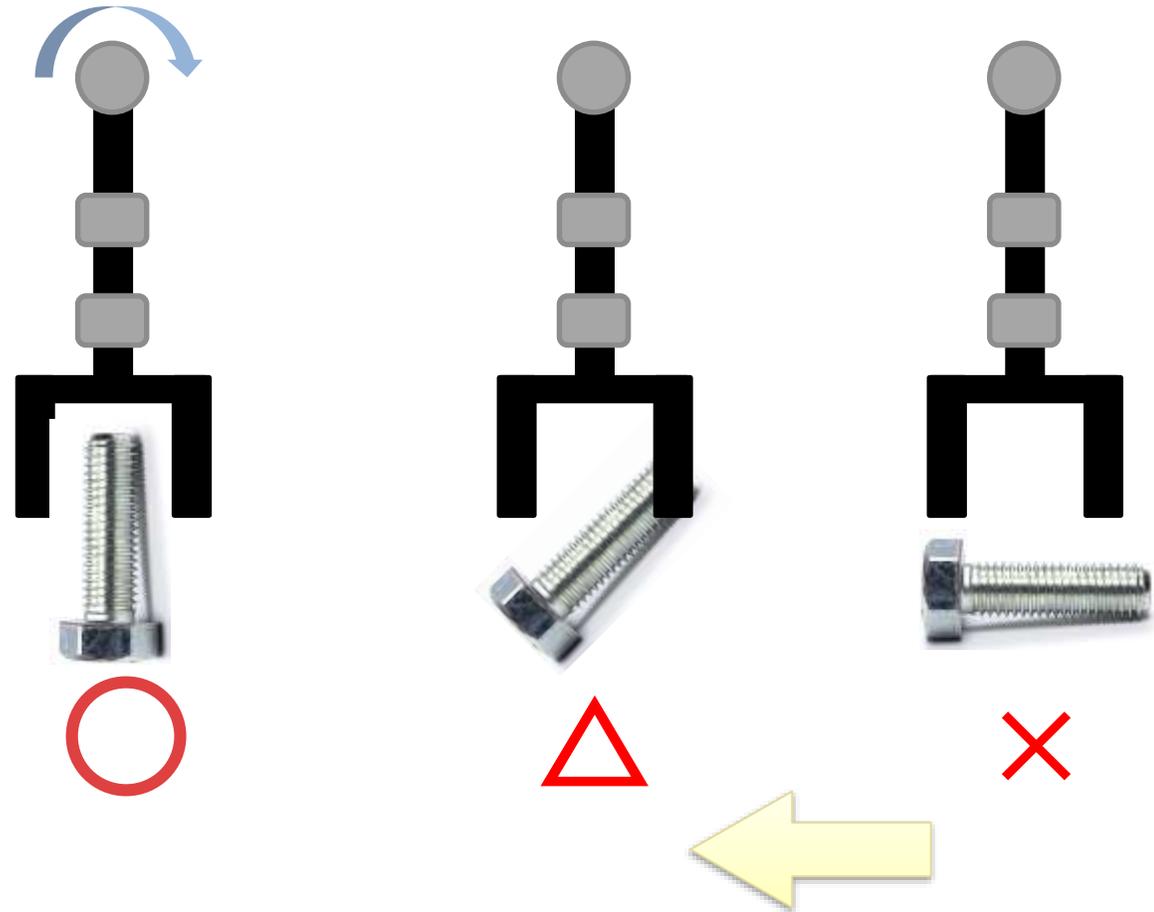
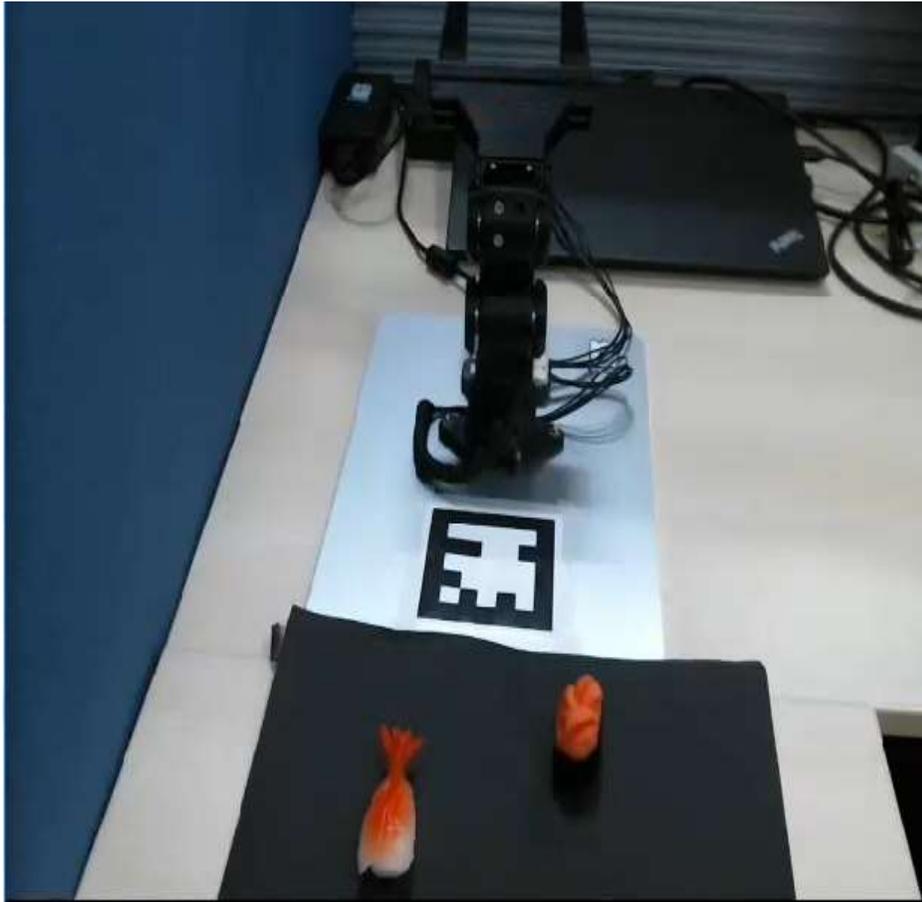
二次計画法

線形制約がある二次目的関数のソルバーです。

quadprog は、次によって指定される問題の最小値を求めます。

$$\min_x \frac{1}{2} x^T H x + f^T x \text{ such that } \begin{cases} A \cdot x \leq b, \\ A_{eq} \cdot x = beq, \\ lb \leq x \leq ub. \end{cases}$$

軌道計画への強化学学習の適用



転がってしまうなどの複雑な制約条件を考慮した
軌道計画が必要なケースも・・・

強化学習による軌道生成

<報酬>

棒を縦向き&アームの中心に移動させること

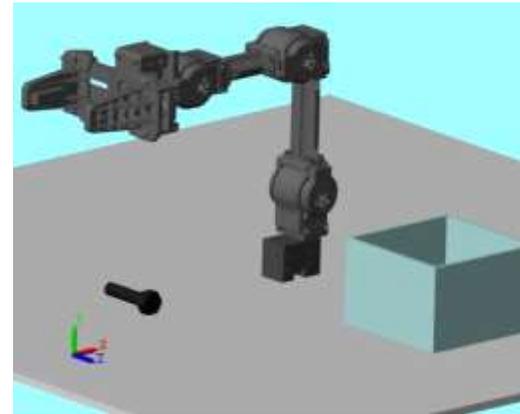
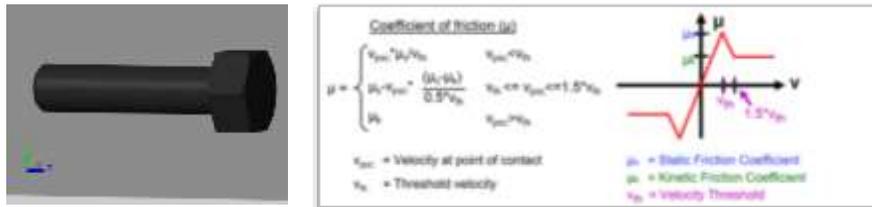
<状態>

棒の座標と傾き

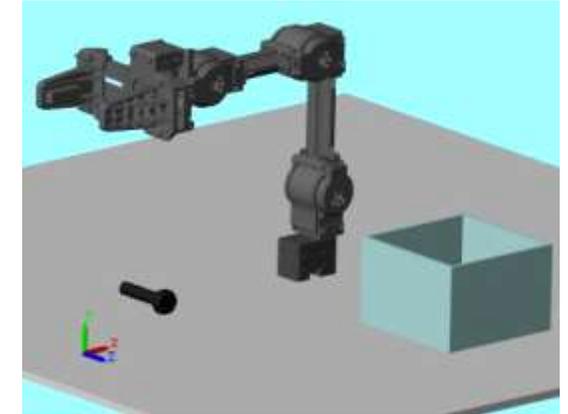
<アクション>

ロボットの先端位置の制御

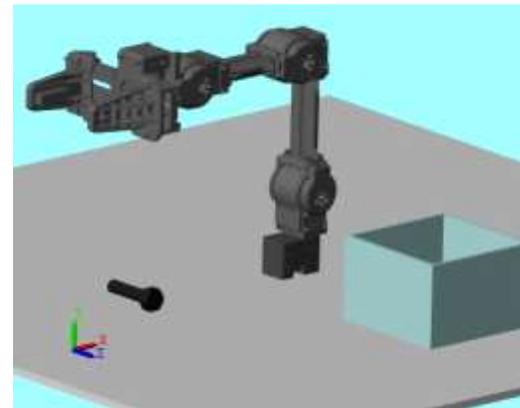
接触部は静摩擦と動摩擦を考慮しモデリング



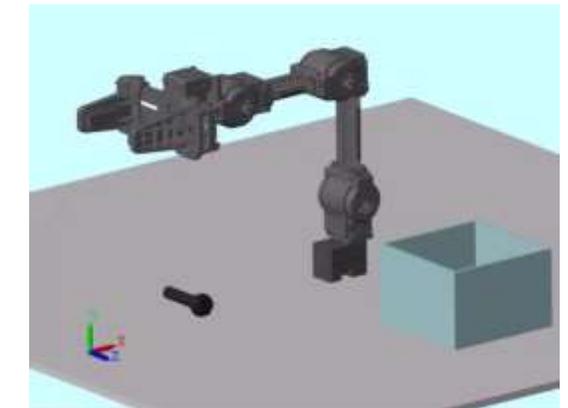
学習初期



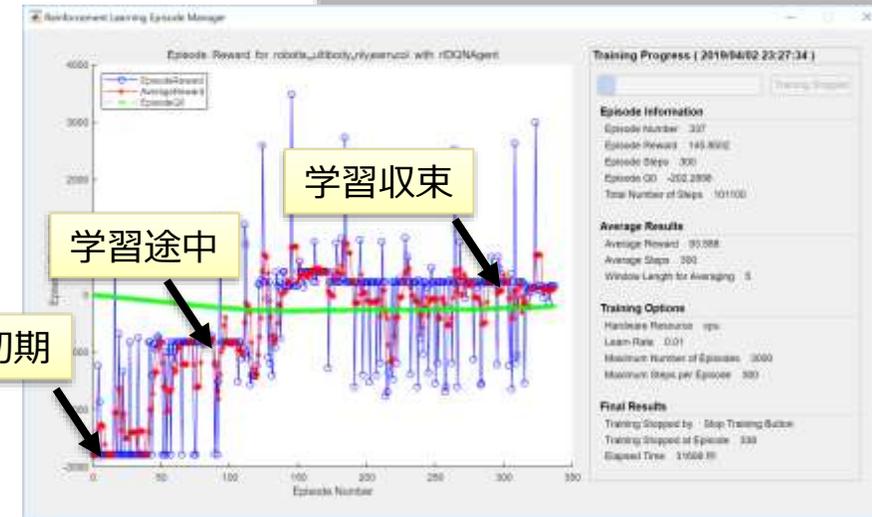
学習途中



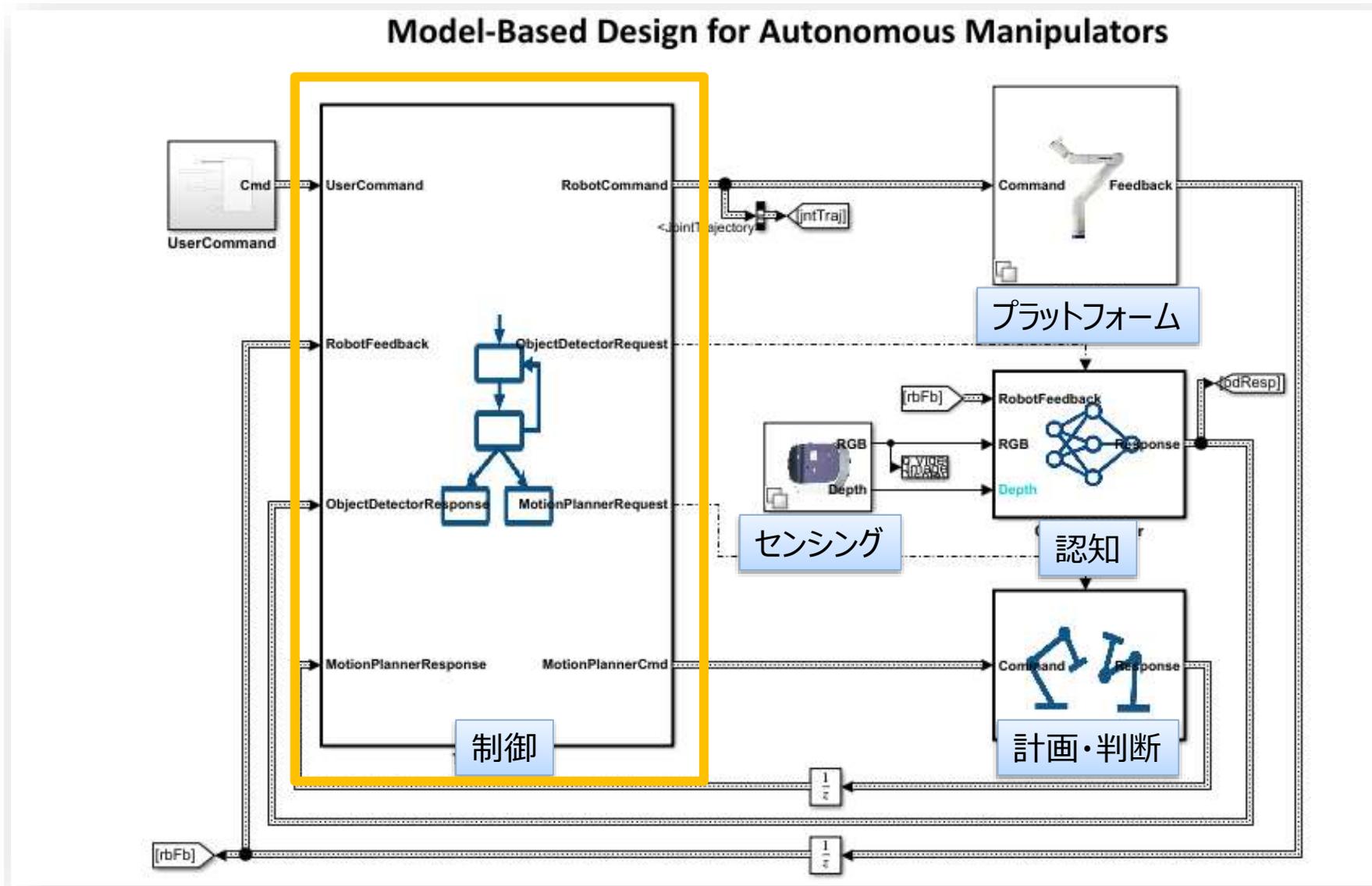
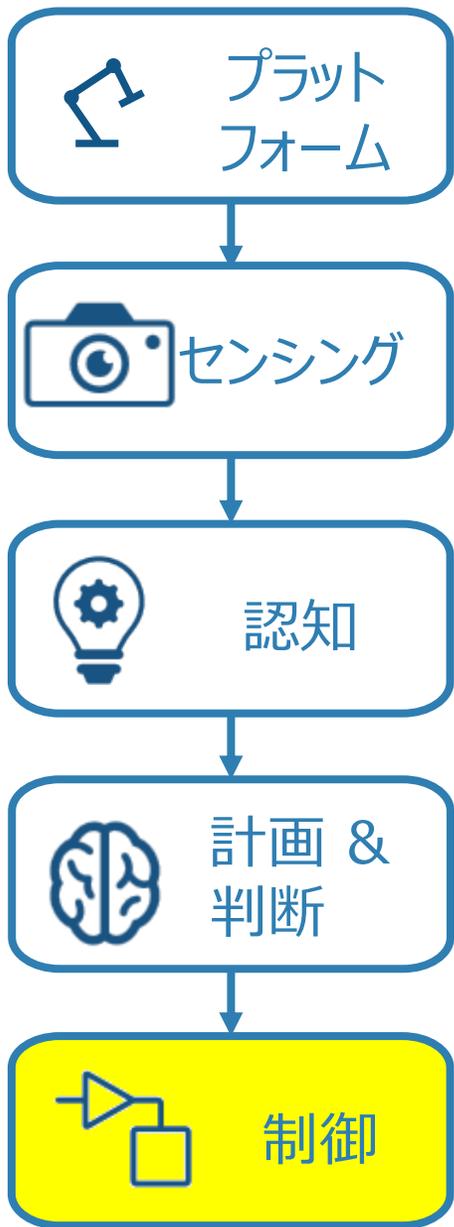
学習収束

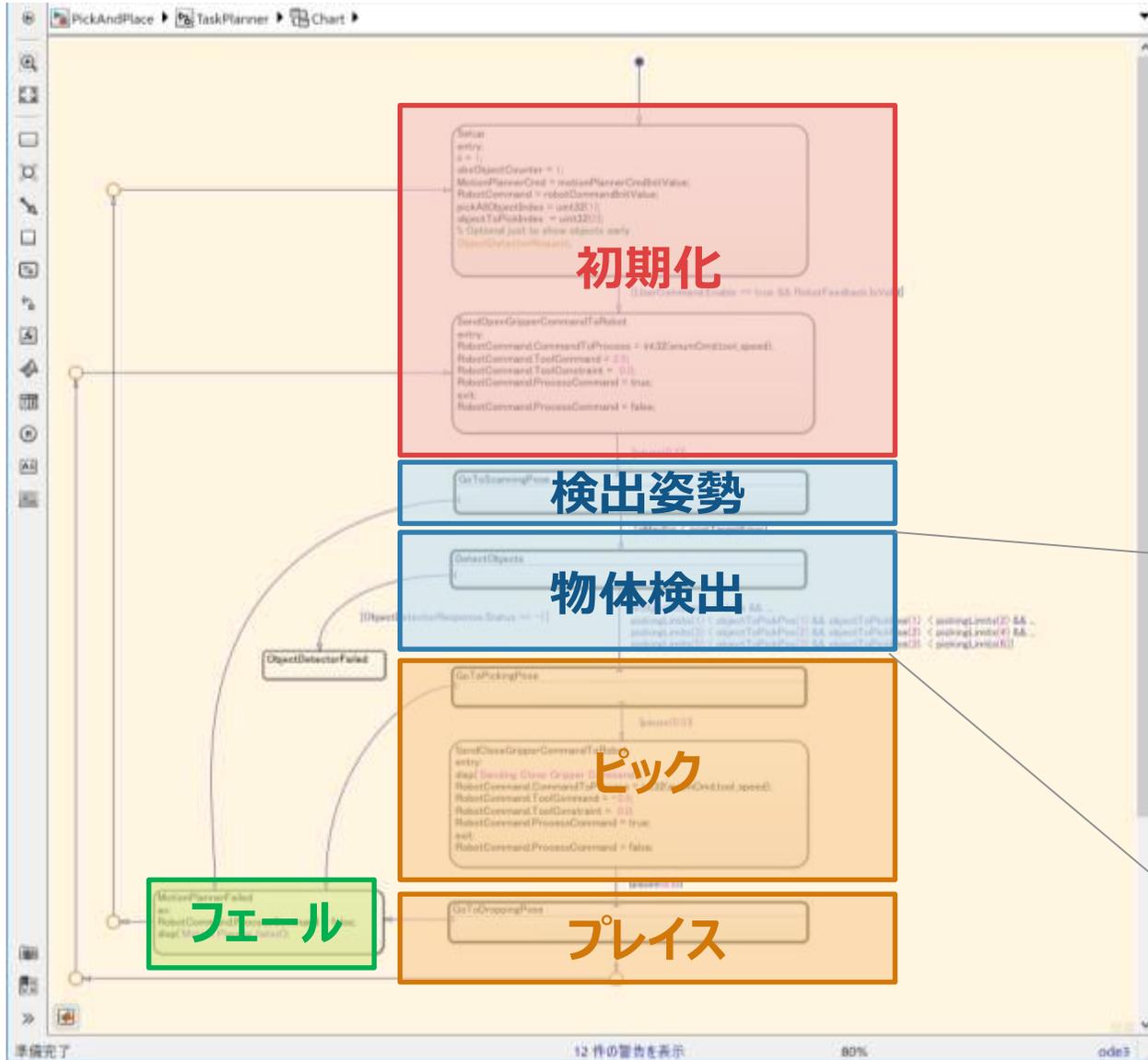


ピック&プレイス

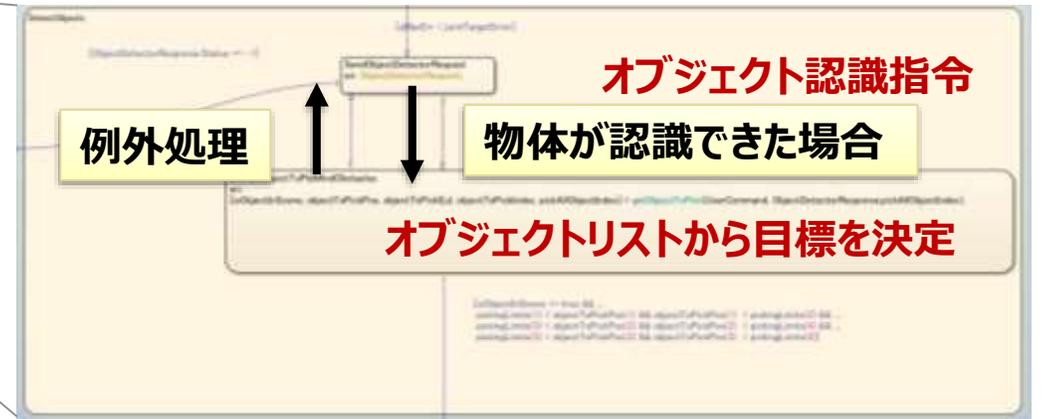


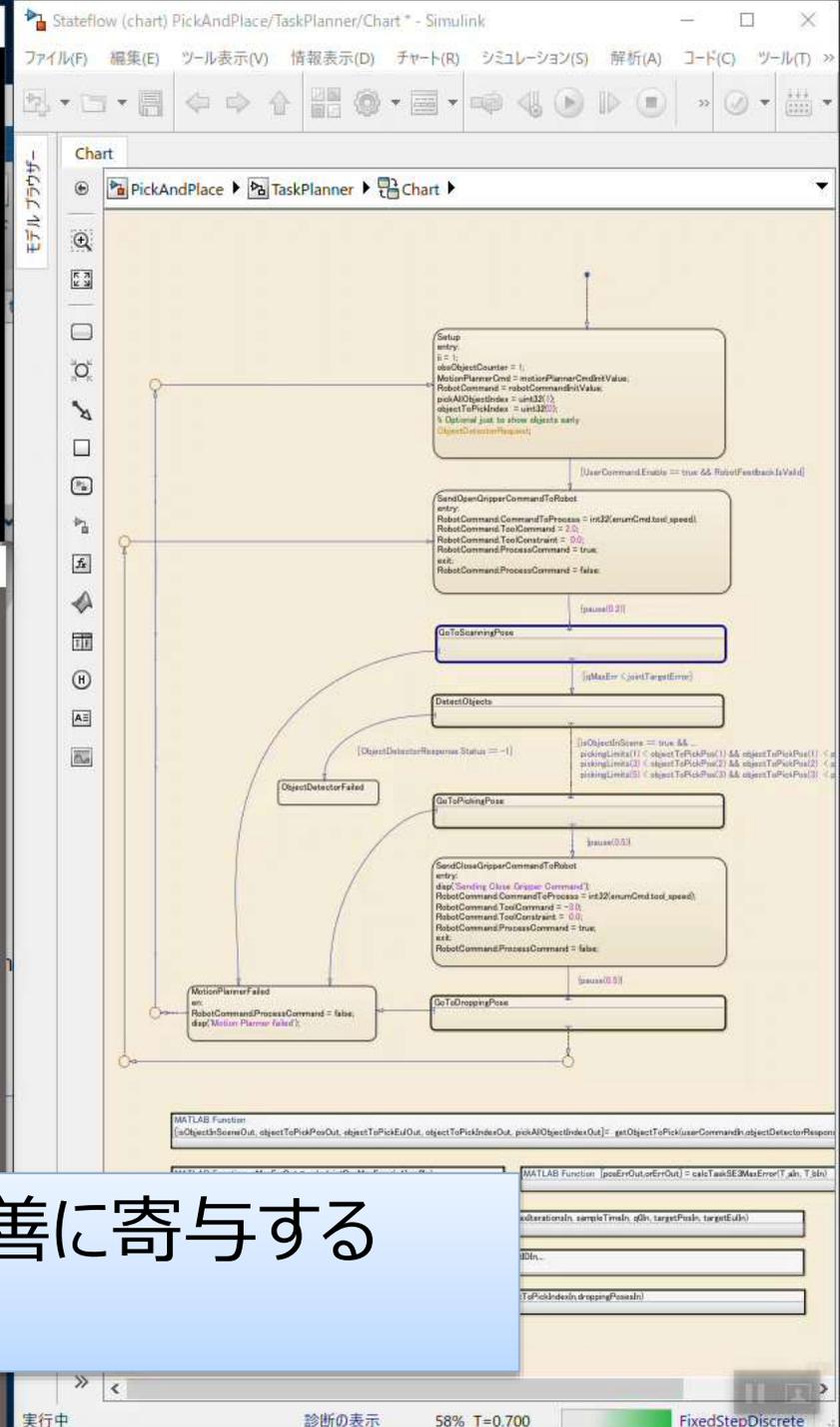
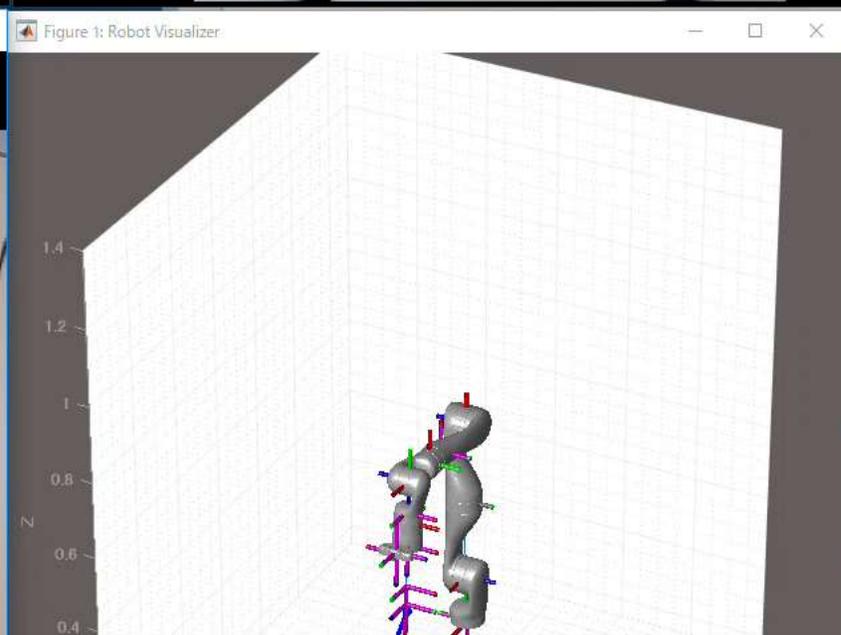
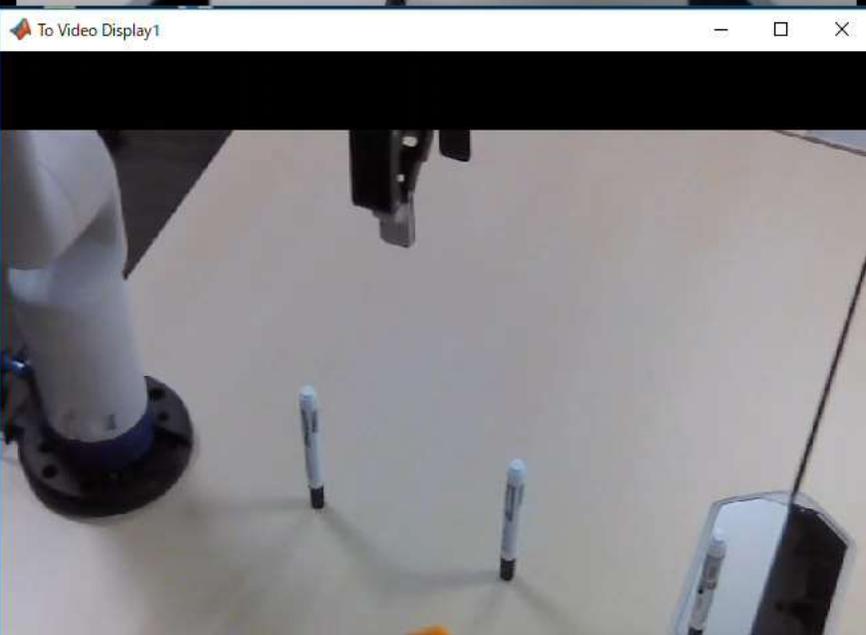
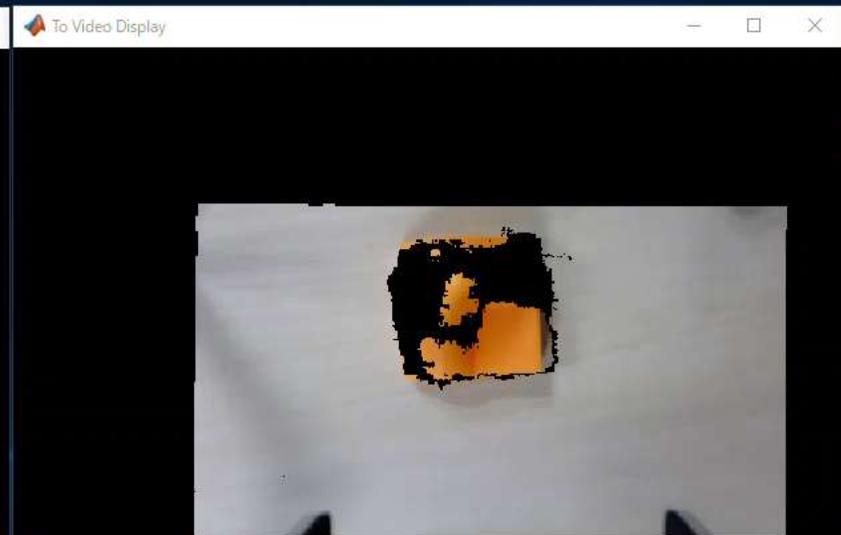
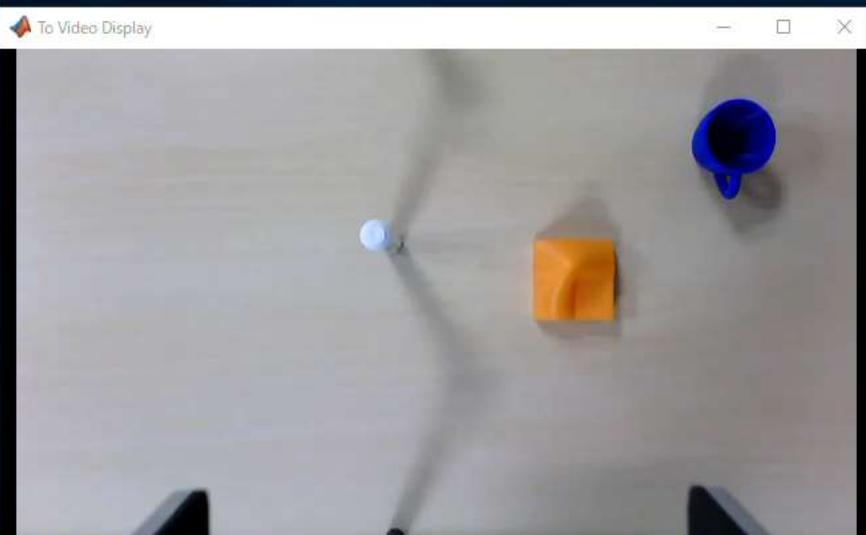
Pick and Place システムの設計





Stateflowによる 行動計画・異常処理の 視覚的な実装





開発プロセス全体を通して開発効率の改善に寄与する
モデルベースデザイン

まとめ

自律ロボットシステムの開発のキーポイント:

1. マルチドメインシミュレーション
2. 複雑な開発要素をトータルでサポートする開発環境
3. モデルベースデザイン

% Thank you