



# 垂直多関節ロボットを外部接続PCから ROS I/F でインピーダンス制御した話

2019年09月25日  
株式会社 豆蔵 三部辰一

# 1. 自己紹介

- 豆蔵は、オブジェクト指向技術で先駆的メンバーが集結して誕生し、「一粒の知性がソフトウェアエンジニアリングを変える」を合言葉に、お客様のソフトウェア開発の開発スタイルの革新を実践的に支援することを目的としたベンチャー企業です

## 会社概要

社名	株式会社豆蔵
設立	2006年10月（旧豆蔵は 2000年5月 よりビジネス開始）
本社	東京都新宿区西新宿2-1-1 新宿三井ビル34F
代表者	代表取締役社長 中原 徹也
資本金	3億1000万円
従業員	151名（2019年4月1日現在）
事業内容	<ul style="list-style-type: none"> <li>■ コンサルティングビジネス           <ul style="list-style-type: none"> <li>□ 開発業務改革サービス</li> <li>□ 組込みシステムのコンサルティング</li> <li>□ 業務システム開発のコンサルティング</li> </ul> </li> <li>■ ソフトウェア開発ビジネス           <ul style="list-style-type: none"> <li>□ 要求開発</li> <li>□ 業務システム開発</li> </ul> </li> <li>■ 教育ビジネス</li> </ul>

## 自己紹介(専門分野)

- 制御工学
  - 古典制御から現代制御まで
- モデルベース開発
  - シミュレーション
  - コントローラ設計
  - MATLAB/Simulink
  - Modelica
- ロボット
  - ROS, ROS2



豆蔵

株式会社豆蔵



株式会社オープンストリーム

NEXTSCAPE

株式会社ネクストスケープ



株式会社フォスターネット

JAT

ジェイエムテクノロジー株式会社

neutral

ニュートラル株式会社

MEX

株式会社コーワメックス

NT SOLUTIONS

株式会社エヌティ・ソリューションズ



Sense Things Japan

センスシングスジャパン株式会社

- 1. 自己紹介
- 2. 背景
- 3. システム構成
- 4. 課題
- 5. 解決策
- 6. まとめ

## 2. 背景

### ■ 開発の動機

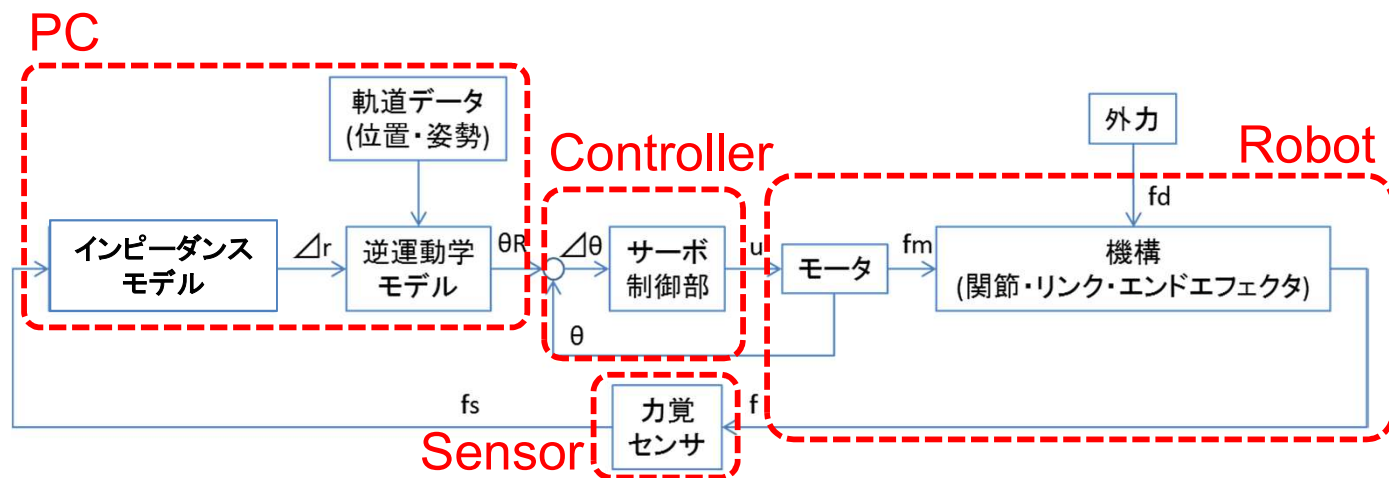
- 色々なロボットを使って部品の嵌合をしたい
- ROSを使って外部からロボットを制御したい

### ■ 技術的課題

- ビジョンでは0.5～1.0mm程度ズれるため、クリアランスの厳しい嵌合は困難
  - ビジョン自身の性能に加え、カメラ座標系⇔ロボット座標系の誤差も加わる

### ■ 解決策

- 力覚センサを使用してインピーダンス制御する
  - インピーダンス制御：外力をフィードバックして位置や速度を調整しながら制御する
- センサを外部PCに接続して、外部PCからROSインタフェースでロボットを制御
  - この構成を取ることで、MoveIt!で制御できるロボットならどれでも使えるようになる



インピーダンスモデル

$$\Delta F = M\Delta\ddot{x} + D\Delta\dot{x} + K\Delta x$$

$\Delta F$  : 外力(センサ値から自重を引いたもの)

$\Delta x$  : 目標軌道からの微小移動量

# 3. システム構成

## ■ ロボット

### ■ UR3 (UNIVERSAL ROBOTS社)

本発表用に使用したロボット

■ ROSパッケージ : [http://wiki.ros.org/universal\\_robots](http://wiki.ros.org/universal_robots)

## ■ 力覚センサ

### ■ DynPick (ワコーテック社)



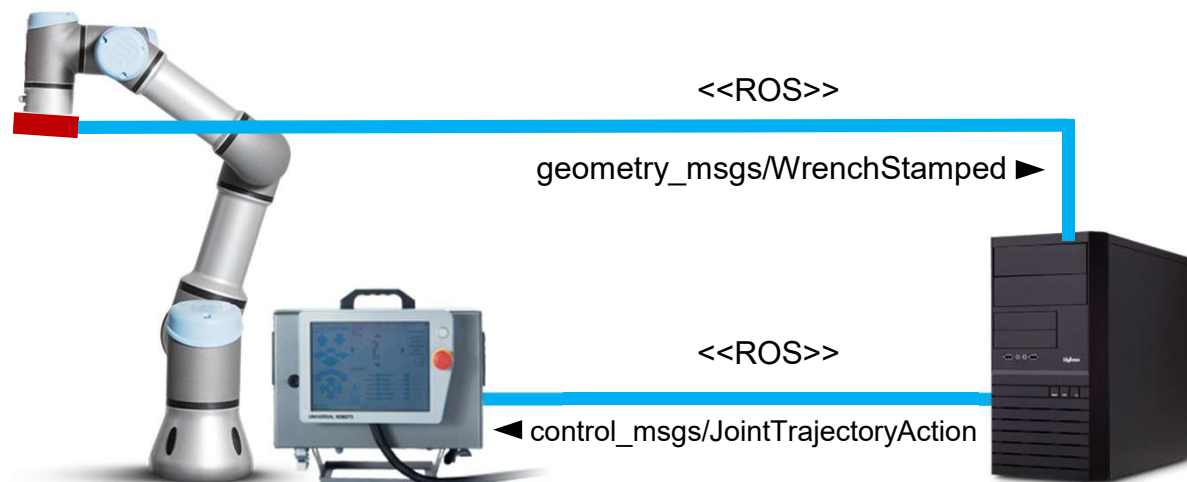
■ エンドエフェクタにかかる力を6軸 ( $F_x$ ,  $F_y$ ,  $F_z$ ,  $M_x$ ,  $M_y$ ,  $M_z$ ) で計測

■ ROSパッケージ : [http://wiki.ros.org/dynpick\\_driver](http://wiki.ros.org/dynpick_driver)

## ■ 外部PC

■ OS: Ubuntu16.04LTS

■ ROS: Kinetic Kame



# 4. 課題

- ROS1はリアルタイム性が保証されていない
  - センサのサンプリング周期( $\Delta t$ )が一定でない可能性がある
  - デジタル制御では離散化の際に用いた $\Delta t$ を計算に使用する為リアルタイム性が保証されないと制御性能に大きな影響を与える

インピーダンスモデル(連続系)

$$\Delta F = M\Delta\ddot{x} + D\Delta\dot{x} + K\Delta x$$

$\Delta F$ : 外力(センサ値から自重を引いたもの)

$\Delta x$ : 目標軌道からの微小移動量

インピーダンスモデル(離散系差分方程式)

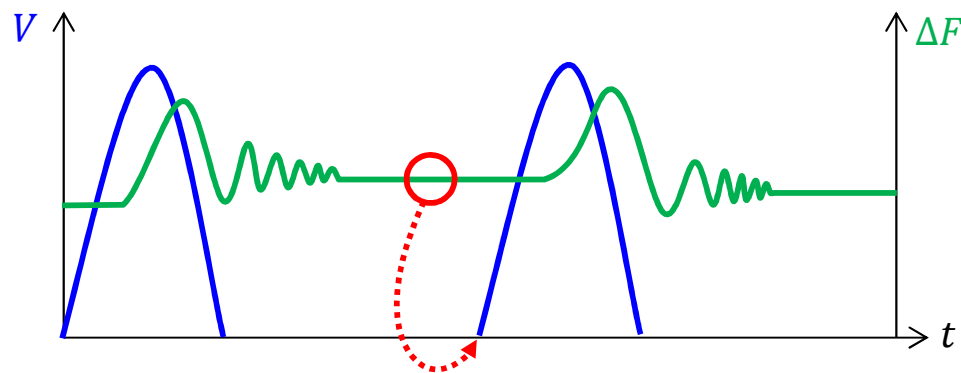
$\Delta t$ で離散化

$$\Delta F = \frac{M(\Delta x_{i+1} - 2\Delta x_i + \Delta x_{i-1})}{\Delta t^2} + \frac{D(\Delta x_{i+1} - \Delta x_i)}{\Delta t} + K\Delta x_{i+1}$$

サンプリング周期 $\Delta t$ は固定値の為  
リアルタイム性が保証されず $\Delta t$ が一定とならなければ  
一般的に制御性能に大きな影響を与える

# 5. 解決策

- ロボットに動作・停止を繰り返し制御して、微動させながら嵌合させていく
- 動作時間より停止時間を長めに取るのがコツ
  - 停止している間に外力が収束してセンサ値が安定する→サンプリング周期のばらつきを無視できる
  - ex. 制御周期=100msec、動作時間=30msec、停止時間=70msec



停止して収束したときの $\Delta F$ でフィードバックすることで、サンプリング周波数のばらつきを無視できる

- フィードバック制御ではMoveIt!を利用できない
  - MoveIt!はパスプランニングに時間がかかる(本環境では1sec以上)
  - PTPで移動(1周期で極短い距離しか移動しないのでPTPで十分)
    - 逆運動学計算については、パッケージで提供されているものを使用するか自前で準備する必要がある

# 5. 解決策

- デモ (インピーダンス制御による嵌合)



## 6. まとめ

- 力覚センサを外部PCに接続
  - 外部PCからROSインタフェースでロボットをインピーダンス制御
    - MoveIt!で制御できるロボットならどれでも使えるようになる
- インピーダンス制御ではMoveIt!を利用できない
  - MoveIt!はパスプランニングに時間がかかる(本環境では1sec以上)
  - PTPで移動(1周期で極短い距離しか移動しないのでPTPで十分)
    - 逆運動学計算については、パッケージで提供されているものを使用するか自前で準備する必要がある
- リアルタイム性が保証されていないROS1で嵌合するにはロボットに動作・停止を繰り返し制御して微動させながら嵌合させていく
  - 動作時間より停止時間を長めに取るのがコツ
    - 停止している間に外力が収束してセンサ値が安定する  
→ サンプル周期のばらつきを無視できる
    - ex. 制御周期=100msec、動作時間=30msec、停止時間=70msec

END