Implementation verification of picking system for industrial robot using ROS and MATLAB®

Masaru Ken Morita
Development Dept.
Environmental & Social Systems Div.
YASKAWA ELECTRIC CORPORATION

© 2019 YASKAWA Electric Corporation
Speaker Introduction

My job
- Controller development for plants
- Especially on HMI for operators
- Specializes in software development

My authored book on ROS
- I mainly wrote application sections such as OpenCV, PCL, Pluginlib, rostest, industrial_ci and so on...

MATLAB integration

1st prize seller on Amazon JP (Robotics category)
ROS commits experiences and contributions (Hobby)

**Autonomous drive**

- ackermann_steering_controller
- First author
- Sent PR to ros_controllers repo & already merged!
- stepback_and_rotate_recovery (plugin)

**steer_bot_hardware_sim (plugin)**

Remote monitoring with integrating ROS & OpenVPN

Source: [http://wiki.ros.org/Robots/CIR-KIT-Unit03](http://wiki.ros.org/Robots/CIR-KIT-Unit03)
Easy to use industrial robots

- Construction of industrial robot (teach-less) interactive UI with Pepper (voice+tablet)
- Industrial robot operation from a remote place (teach-less)
- Industrial robot operation from demonstration
- Improving the layout of industrial robots using 5G network
Outline

1. Motivation
- Robotics System Toolbox™
- What’s ROS-Industrial?
- Pros for ROS+MATLAB

2. Sample Apps
- Pick & Place App Overview
- Detail on each functions

3. Impression as a User
- Pros
- Cons
- Future work

© 2019 YASKAWA Electric Corporation
Motivation

YASKAWA

- Unique technologies with advantages
- Limited resources for new technology introduction and verification

In-house tech.

Open Innovation

ROS

- Easy combination of advanced technologies
- Unevenness of the available functions

OSS

Secure quality that could be difficult to be provided by only installing OSS
Advanced technology utilization not provided by ROS alone

MATLAB/Simulink

Possibility of innovation through integration of

In-house tech. × OSS × Commercial Tool

I introduce the combination verification of

OSS × Commercial Tool
High potential of Robotics System Toolbox

MATLAB/Simulink

Workspace

rosinit('master_host')
sub = rossubscriber('/chat', @callback)
pub = rospublisher('/chat', std_msgs/String)

ROS Network

Various Toolboxes

- Data Analytics
  Demonstrates how to use MATLAB for large-scale data, machine learning, and analysis in an enterprise environment.

- Wireless communication
  See how MATLAB can help you develop algorithms and perform wireless system simulations.

- Deep Learning
- Computer Vision
- Signal Processing
- Econometrics and Risk Management
- Robotics
- Control System

Source: https://www.mathworks.com/products/matlab.html

Source: https://projects.preferred.jp/tidying-up-robot/
Difficulty of adopting ROS at the manufacturer 1) Technical Issue

Technical issue:
- It is difficult to employ engineers who can handle the latest OSS libraries and various programming languages

Customers to deliver value to...
Difficulty of adopting ROS at the manufacturer 2) Strategic Issue

- Conflicts between the use of OSS and intellectual property protection.
- No precedent using OSS in development.

Customers to deliver value to

But not easy...
Difficulty of adopting ROS at the manufacturer

2) Strategic Issue

Customers to deliver value to

MATLAB/Simulink

• Software installation experience: more than 100,000 companies, governments, universities
• Customer base: More than 185 countries
• MATLAB Users: Over 4 Million Worldwide
• MATLAB Central File Exchange Downloads: Over 3 Million Files
• Number of contributors to the MATLAB Central app: more than 525,000 worldwide
• Number of third-party solutions created with MATLAB / Simulink: 500+
• MATLAB Number of Books: More than 2,000 in 27 languages

Source: https://pictarts.com/01-illustration/00011-free-art.html

Source: https://youtu.be/1Zpw2288VMQ
Source: https://projects.preferred.jp/tidying-up-robot/
Difficulty of adopting ROS at the manufacturer

1) Solution

MATLAB/Simulink

Solution:
1. Many MATLAB engineers exist in manufacturers
2. MATLAB resources are also accumulated a lot there

Technical issue:
- It is difficult to employ engineers who can handle the latest OSS libraries and various programming languages

Customers to deliver value to
Difficulty of adopting ROS at the manufacturer

2) Solution

Customers to deliver value to

That’s why I started to watch on MATLAB & ROS integration!

Solution:
1. Implementing algorithm with MATLAB avoids OSS argument
2. MATLAB-based product development is often experienced

Strategic issue:
• Conflicts between the use of OSS and intellectual property protection.
• No precedent using OSS in development.

Source: https://youtu.be/1Zpw2288VMQ
Source: https://projects.preferred.jp/tidying-up-robot/

Source: https://www.irasutoya.com/2013/07/blog-post_5717.html
Robotics System Toolbox

- Includes various algorithms and functions essential for robotics
- Added interface for linking MATLAB® / Simulink® and ROS

Usage in the demo

Robotics System Toolbox

ROS IO

Other Toolboxes

Source: https://www.vexels.com/png-svg/preview/140692/linux-logo

MATLAB/Simulink

Windows 10
Development of ROS for industrial application

- Consortium to promote industrial application of ROS
- Over 60 global companies such as manufacturers, users and plants participate

Source: https://rosindustrial.org/ric/current-members
Motoman's repository of ROS-Industrial

Free published in GitHub repository

- Driver, 3D CAD model, visualization tool correspondence, etc.

GP12/7/8
MH5/12/50
MotoMINI
SDA10F/10D
SIA5D/10D/20D

Reference: GitHub

Source: https://github.com/ros-industrial/motoman
Pros of ROS implementation at the manufacturer

**ROS features**

- Communication library → Focus on application development
- Development and operation tools → Graph, 3D Viewer, Simulator, Compiler
- High-performance library → Also functions at academic level
- Ecosystem → Easy to share and install apps

---

**Easy benchmarking**

- Free access to advanced technologies available with ROS

---

**Easy combination verification**

- Sensor / actuator compatible with ROS can be easily introduced
- Interworking can be expected with simple settings by using the ROS network

---

**Reduction of development man-hours Efficiency of advanced function benchmark etc.**
Advantages of Introducing Robotics System Toolbox

Robotics System Toolbox Features
- Maternal MATLAB is multi-platform
- Can interact with other toolboxes and scripts

Easy combination verification with existing models
- Works with MATLAB / Simulink assets accumulated in-house
- Mainstream OS among manufacturers: ROS verification is possible based on Windows

Easy to build various applications
- By combining various toolboxes of MATLAB, it is easy to develop advanced technology-based applications that are difficult to build with ROS alone

Perform verification with sample application
Outline

1. Motivation
   - Robotics System Toolbox™
   - What’s ROS-Industrial?
   - Pros for ROS+MATLAB

2. Sample Apps
   - Pick & Place App Overview
   - Detail on each functions

3. Impression as a User
   - Pros
   - Cons
   - Future work

© 2019 YASKAWA Electric Corporation
App demo video

Example based system construction

- Build a system based on Example provided by Mathworks
- KUKA youBot → Yaskawa MotoMINI, SVM → YOLOv2

Object recognition
State control
Block Diagram
Simulator
App demo video
## Sample application summary

### Object recognition + position estimation
- Object recognition by deep learning based on RGBD sensor information
- Estimate the position of a 3D object

### Path plan + pick & place
- Plan the trajectory from the robot's current posture to the recognized object position
- Control the robot based on the planned trajectory and carry out pick and place

### Voice input
- Voice input with microphone
- Control the robot according to the instructions

### MotoMINI simulation
- Utilize MotoMINI model provided by ROS-I
- Use simulator (Gazebo)

### Build sample application only with MATLAB function + open source
Sample app overview

Block diagram (Simulink model)

Flow of explanation

1. Object detection
   Position estimation

2. Path planning

3. Speech recognition

4. State machine

Path planning

State control

Simulation

Object detection
Position estimation

Acquisition of sensor value
Sample app overview:

1. Object detection
   - Position estimation

2. Path planning

3. Speech recognition

4. State machine

Block diagram (Simulink model):

- Object detection
- Position estimation
- Path planning
- State control
- Speech recognition
- Acquisition of sensor value
Flow of object detection + position estimation

**Sensor info.**
- Real or simulator

**Object detection**
- Use RGB image
- Detect by YOLOv2

**Position estimation**
- Use depth image
- Get 3D position

**Flow**
- ROS -> MATLAB
- Object detection
- Object position
- Path planning

**Toolboxes**
- Robotics System Toolbox™
- Deep Learning Toolbox™
- Computer Vision Toolbox™

**MATLAB**
- Simulink
Object recognition with deep learning (YOLOv2)

Flow of object recognition by YOLOv2

- **Learning**
  - Image Acquisition
  - Labeling
  - Feature extraction + learning
  - Input image
  - Anchor
  - YOLOv2
  - Learned model

- **Recognition**
  - Image Acquisition
  - End to End Network
  - Input image
  - Bounding box estimation
  - Extract work
  - To position estimation
Position estimation with point cloud

Use of both RGB image and depth image data by RGBD sensor

Object detection with RGB image

Get the depth of the corresponding position in the depth image
Sample app overview: 2. Path planning
Path planning flow

Setting
- Robot config (URDF, CAD)

Tree
- Easy access to ROS config data

IK
- Hand start / end point specification
- Calculate joint angle

Initial posture / final posture (joint angle expression)

Path planning
- Initial joint angle path (Linear interpolation)
- Optimization
- Optimal joint angle path (Spline interpolation)
Path planning flow

Setting
- Robot config (URDF, CAD)

Tree
- Easy access to ROS config data

IK
- Hand start / end point specification
- Calculate joint angle

Initial posture / final posture (joint angle expression)

Path planning
- Initial joint angle path (Linear interpolation)
- Optimization
- Optimal joint angle path (Spline interpolation)

MATLAB Simulink
Robotics System Toolbox™
Simscape™ Multibody™ Optimization Toolbox™
ROS

© 2019 YASKAWA Electric Corporation
Optimization overview

Optimization calculation

- Initial path (linear)
- Physics sim
- Cost function
- Update Spline points
- Optimal path

Simscape™ Multibody™

Optimization Toolbox™

ROS→MATLAB import

Note: Separately from Robotics System Toolbox

ROS

MATLAB Simulink
Optimization overview: Main points

- Physics sim
- Cost function

Simscape™ Multibody™

Config

Robot config (URDF, CAD)
Optimization overview: Main points

- **IK simulation**: Visualization of 3D motion
- **Calculate torque by simulation**: Total torque cost optimization

**Physical sim**:
- Simscape™ Multibody™
- URDF

**Cost function**: Config
- Robot config (URDF, CAD)
- CAD import

**URDF import**
- Cooperation is easy by the import function of ROS information

- Cooperation is easy by the import function of ROS information
Sample app overview: 3. Speech recognition

Flow of explanation:
1. Object detection
   Position estimation
2. Path planning
3. Speech recognition
4. State machine

Block diagram (Simulink model):
- Path planning
- State control
- Speech recognition
- Object detection
  Position estimation
- Acquisition of sensor value
Speech recognition flow

**Voice input**
- Via mic.
- Acquire sound signal

**Voice recognition**
- Spectrogram img
- Word recognition w/t CNN

**Word recognition result**
“go” “right” ...

**Operation command**
- generate motion instruction to hold the specified work

**Spectrogram conversion**

**Operation instruction**

**State Control**

**Audio Toolbox™**
- MATLAB Simulink

**Deep Learning Toolbox™**
- Audio Toolbox™
- MATLAB Simulink

**Robotics System Toolbox™**
- MATLAB Simulink

**Explain from now**

© 2019 YASKAWA Electric Corporation
Speech recognition Using deep learning (CNN)

**Flow of speech recognition w/t CNN**

**Learning**
- Voice data set
- Spectrogram image
- Feature extraction + learning
- CNN
- Learned Model

**Recognition**
- Voice input
- Spectrogram image
- Feature extraction + classification
- Learned CNN

**Word recognition**
- Motion: go, stop
- Place: left, center, right

**Motion instruction generation**

**State Control**
Sample app overview:

1. Object detection
   Position estimation

2. Path planning

3. Speech recognition

4. State machine

Block diagram (Simulink model):

- Path planning
- State machine
- Speech recognition
- Object detection
  Position estimation
- Acquisition of sensor value
- Simulation ROS
State machine

Control modules by state transition

- State management of manipulator

Loop of object recognition & Pick & Place

Home Position

Wait

Pick & Place

Path planning based on object position

Object Recognition

State transition during object detection

Joint path Open/close

Pick and place Completion

Path planning

MATLAB Simulink

Stateflow®

© 2019 YASKAWA Electric Corporation
Outline

1. Motivation
   - Robotics System Toolbox™
   - What’s ROS-Industrial?
   - Pros for ROS+MATLAB

2. Sample Apps
   - Pick & Place App Overview
   - Detail on each functions

3. Impression as a User
   - Pros
   - Cons
   - Expecting Future work

MATLAB/Simulink
ROS
YASKAWA
Impression & Requests as a User of MATLAB

**Pros**
- Easy collaboration with advanced functions → ROS alone can not be handled as easily as MATLAB
- Highly compatible with ROS, such as URDF, TF, Gazebo I/F too.
- Easy to use sequencer and blocking GUI such as StateFlow and Simulink
- Development based on Example enables early startup of prototypes

**Cons**
- Processing takes time → Speeding up with Coder is also possible
- The parts requiring tuning are dispersed when changing the robot
- Some apps require to load the robot model separately for each toolbox

**Expecting Future work**
- Import MATLAB motor and other models as Gazebo plug-in
- Need a sample where ROS for Windows and MATLAB work together → Currently there is only the tutorials where ROS runs on Linux on VM
- Support for ROS 2 and V-REP too!
YASKAWA

MATLAB/Simulink

ROS

GAZEBO
Labeling for YOLOv2

Training data
- Place works with random sets of position and posture
- Automatically capture parts images with various poses
- Automatically estimate bounding boxes and labels

Gazebo
- Replace works from MATLAB
- Send images to ROS

MATLAB Simulink

ROS

10X Faster