

ROS Based Pick-and-Place Robot Simulation Using MOVEit

S. Karthikeyan^{1*}, S. Karthik Raja², R. Hariharan³, M. Premkumar⁴

^{1,2,3}Student, Department of Electronics and Communication Engineering, Panimalar Institute of Technology,
Chennai, India

⁴Professor, Department of Electronics and Communication Engineering, Panimalar Institute of Technology,
Chennai, India

*Corresponding author: karthikshan1521999@gmail.com

Abstract: A conventional robot is a system which receives data from the sensor, process those based on predefined algorithms and makes the actions through the actuators. Moreover, different sections of the robot such as the controller and the sensors needed to be coded out in different environment and tools. Which leads to miscommunication among the various nodes. In order to resolve these issues, the Robotic Operating System is required, as it is capable of communicating among various nodes through a common master.

Keywords: Robotic operating system, Ubuntu, Melodic, Pick and Place, rViz, Move it, Python, Inverse kinematics, Forward kinematics, Unified robot description format, Panda robot, End-Effector.

1. Introduction

The humans have been fascinating about robots over long ages and have built in numbers. The humans have taken several actions as granted such as open a door knob, climbing the steps and so on. But those in case of robotics it's not that easy, as they had to go through a number of coded lines to carry out even simple tasks.

This makes building up of robots an extreme challenge. Robot Operating System (ROS or ros) is robotics middleware (i.e. collection of software frameworks for robotic software development). Packages containing application-related code which uses one or more ROS client libraries.

This robot can simply pick the object for a particular position and place it in the specified position through the specified trajectory path without the arm colliding with the surrounding environment. For which it involves the accurate forward and inverse kinematics calculations.

2. Project Description

For doing many personal physical works human needs to depend on man power. Our creativity is what differentiates us as a superior species. When man power is used for non-creative tasks too leads to various problems. One of the common problem include the shortage of man power for creative tasks. The next major problem will be the lack of interest in the people

who are employed to do non creative tasks.

As non-creative tasks are somewhat root involving a set of procedures, humans who are employed to do them are not inclined to enjoy it. To overcome this Robots are being used up, to do those non creative tasks. One of which is the Pick-and-Place action. Hence for that purpose the pick-and-place robots are being created.

But still the construction of the robot in the conventional way is tedious and time consuming. Hence we have now introduced the field of Robotic operating system to pick and place robot.

ROS:

The Robotic Operating System is a middle-ware, Meta operating system. Although the ROS is not an operating system, it provides the services designed for a heterogeneous computer cluster such as hardware abstraction low level device control, implementation of commonly used functionality message-passing between processes and package management.

The ROS is a flexible framework for writing robot software. It is a collection of tools, libraries, and convention that aim to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platform.

rViz:

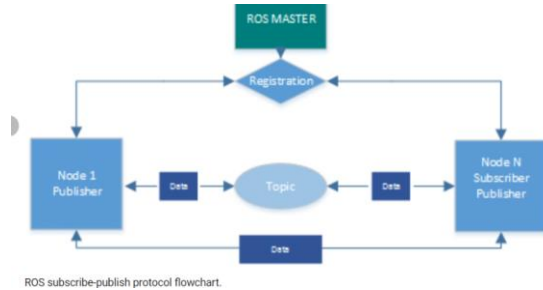
It is a Three dimensional visualization tool for ROS application. It provides view of robot models, capture sensor information's from robot sensor , and replay captured data.rviz is ROS graphical interface that allows you to visualize a lot of information's ,using plug-ins for many kinds of available topics.

MOVEIT:

It is a primary source a lot of the functionality for manipulation (and mobile manipulation) in ROS. It is used for motion planning in industrial robots like ARM navigation, Ros control, 3-D navigation. Moveit transfer support the exchange of file and data between servers, System and application within and between organization, as well as group and individuals using a common shared folder with simple browser access for users.

3. Working of ROS

Initially the ROSCORE needed to be executed in a separate terminal, as this is highly important for the execution of the entire code. This unites the output of various other nodes of the project as they are constructed in different languages.



Then the execution coded needed to be given and run.

The communication model requires that both A and B communicates via a single topic. A topic is fully defined by its name, so as long as the resolved topic name is the same and they are connected to the same ROS Master, the nodes will be able to communicate with each other.

The ROSCORE combines the various ROSTOPIC and the Robot begins to execute its movement over the prescribed trajectory path. For which the motion planning need to be done through forward and inverse kinematics.

The forward kinematics refers to the calculations need over the various parts of the robot for its movement and the inverse kinematics refers to the calculations needed for the reflexes needed for the others.

The various libraries constructed in various languages are run simultaneously over the same environment. The panda robot that is being executed here has the inbuilt libraries that are stored in CATKIN folder are fetched.

The 3-D model movement of the panda ROBOT is being run on rViz and on the same time the simulation of the arm in the environment setup runs on moveit for collision free testing.

Applications:

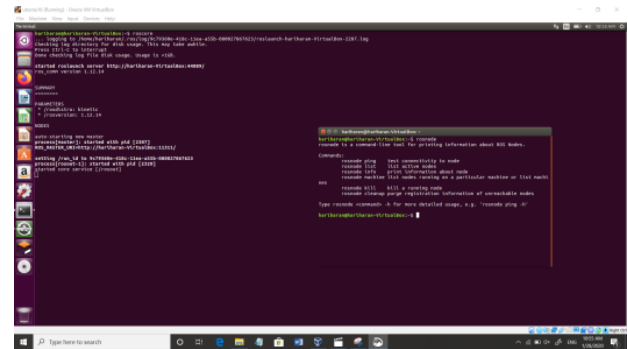
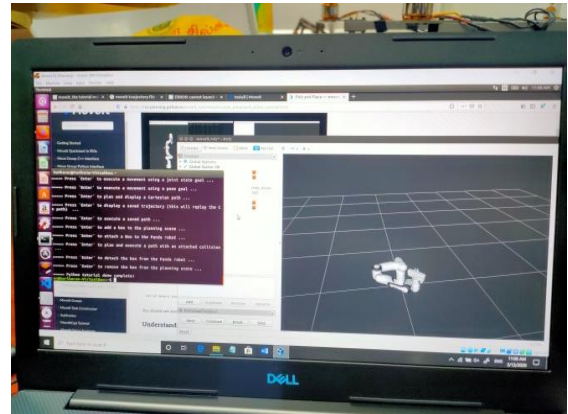
In robots that are used in space research needed to be modified often, as they are working in various environment involving several experiments. But practically it is difficult for an Astronaut to code those and also to take a coder to space. In order to eliminate this difficulty, the ROS robots are used.

As in ROS the programmer can code the robot in the earth and can check all the conditional motion planning and collision avoidance testing, after which it can be sent to space.

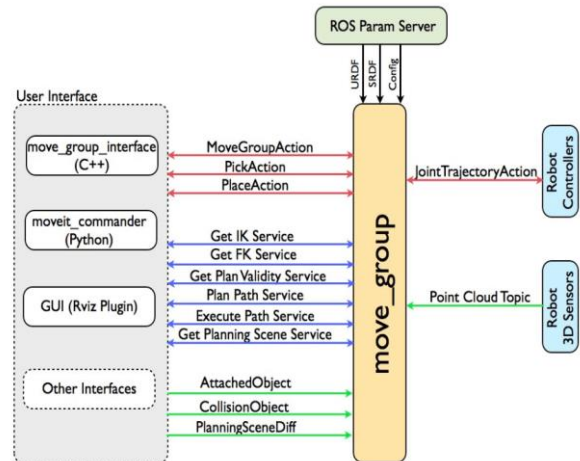
The other main application of ROS is that, in industries where several robots of same operation need not be coded separately every time, as it is a tedious work. But by using ROS this can be eliminated as code reuse is possible here.

And also in some cases the sensors and motors needed to be coded in python, whereas the microprocessors needed to be done in c language. ROS connects all those under one domain.

Output:



Flowchart:



4. Conclusion

This is an innovation that takes the benefit of many available inventions. This is kind of assembling many available works in technology to execute the intended task. By this the man kind would be benefited. This technology does not just be limited to pick and place but also in future could be used up to rescue and high end industrial application.

References

- [1] Pedro Tavares, Armando Sousa, "Flexible Pick and Place Architecture using ROS Framework," IEEE systems journal, vol. 9, no. 1, March 2015.
- [2] Wei Qian, Zeyang Xia, Jing Xiong, Yangzhou Gan, Yangchao Guo, Shaokui Weng, Hao Deng, Ying Hu, Jianwei Zhang, "Manipulation Task Simulation using ROS and Gazebo," 2014 IEEE International Conference on Robotics and Biomimetics December 5-10, 2014, Bali, Indonesia.
- [3] Kenichi Ohara, Kyohei Iwane, Tomohito Takubo, Yasushi Mae, Tatsuo Arai, "Component-based Robot Software Design for Pick-and-Place Task Described by SysML," 2017 First IEEE International Conference on Robotic Computing.
- [4] Mahidzal Dahari, Jian-Ding Tan, "Forward and Inverse Kinematics Model for Robotic Welding Process Using KR-16KS KUKA Robot."
- [5] Chi-Yi Tsai, Ching-Chang Wong, Chia-Jun Yu, Chih-Cheng Liu, and Tsung-Yen Liu, "A Hybrid Switched Reactive-Based Visual Servo Control of 5-DOF Robot Manipulators for Pick-and-Place Tasks."
- [6] Kai Weng Wong and Hadas Kress-Gazit, "High-level Task Specification to Robot Operating System (ROS) Implementation."
- [7] Rodney A. Brooks, "Planning collision free manipulation for pick and place operation."