NVEO
Natural Volatiles &
Essential Oils

Robot Arm Visual Environment Development Using Ros, Solidworks, Rviz & Urdf Generation

¹Barath Krishna G, ²Krithik Rajavelu , ³Dr. Bindu B, ⁴Harsha Gopinath

¹Dept. of Robototics and Automation Engineering
PSG College of Technology
Coimbatore, India
barathgvbd@gmail.com

²Dept. of Robototics and Automation Engineering
PSG College of Technology
Coimbatore, India
krithikrajavelu@gmail.com

³Dept. of Robototics and Automation Engineering
PSG College of Technology
Coimbatore, India
bdu.rae@psgtech.ac.in

⁴Dept. of Robototics and Automation Engineering
PSG College of Technology
Coimbatore, India
har.rae@psgtech.ac.in

Abstract

Automation is the key element of every industrial unit across the world. Robotic Arms play a significant role in automatized industrial processes like assembling, packaging, etc. Development of a perfect robot arm needs an effective and error-free visualization of the simulated arm. This project focusses on designing a robot arm from scratch and visualizing it for further analysis. This reduces mistakes caused due to various factors including environment and operator considerations.

Keywords— ROS, RViz, Gazebo, URDF, Motion Planning

Introduction

INDUSTRIES ACROSS THE WORLD ARE ADVANCING TOWARDS COMPLETE AUTOMATION IN ORDER TO INCREASE PRODUCTIVITY, QUALITY AND ACCURACY. ROBOT ARMS ARE VITAL PARTS OF ANY AUTOMOTIVE MASS PRODUCTION INDUSTRY WHERE THEY SERVE A VARIETY OF PURPOSES UNDER VARIOUS SECTORS. ROBOT ARMS ALSO PROVED TO BE EFFECTIVE AND ACCURATE IN CRUCIAL APPLICATIONS THAT COULD NOT BE PERFORMED BY HUMAN LABORERS. THE

DEVELOPMENT OF A ROBOTIC ARM IS A STRUCTURED PROCESS THAT INVOLVES DESIGNING, MODELLING, PERFORMANCE ANALYSIS, TESTING AND OTHER INTERMEDIATE PROCESSES. DEVELOPING A ROBOT ARM SIMULATION AND MONITORING ITS PERFORMANCE HELPS THE ARM TO OPERATE WITHOUT ANY ERRORS AND DEFECTS IN THE FUTURE.

A. *Problem Statement*

Manual industrial processes require hard labor and has lesser accuracy and repeatability in performance which leads to errors in manufacturing and causes both financial and economic losses to the industries, especially in processes which has to be done quicker and yet in a precise way.

B. *Literature Survey*

The robotic arm is an integration of several links and joints each having their own movements and impacts on the overall movement of the arm. Visualization of the developed or the arm that is in development stage helps to understand the performance of the robot arm in a precise way. Robot Operating System (ROS) plays a significant role in performance analysis of the arm as well as other automatic systems which in turn proves to be useful as it supports other visualization tools like RViz and Gazebo.

II. PROPOSED METHODOLOGY

The robot arm simulation and visualization can be done in many ways. This project focuses on simulating it based on URDF generation and Moveit tool visualization.

A. *URDF Generation*

The URDF is a unique feature for analysing the developed robot arm in order to know the kinematic features of the arm as well the position, velocity, acceleration and inertial movement of each and every joint of the arm. The URDF also provides facilities to develop a robot arm from scratch without having to import the links developed in other design software. With this feature, one can develop an arm based on the requirements and configure the physical as well as the kinematic properties of the arm.
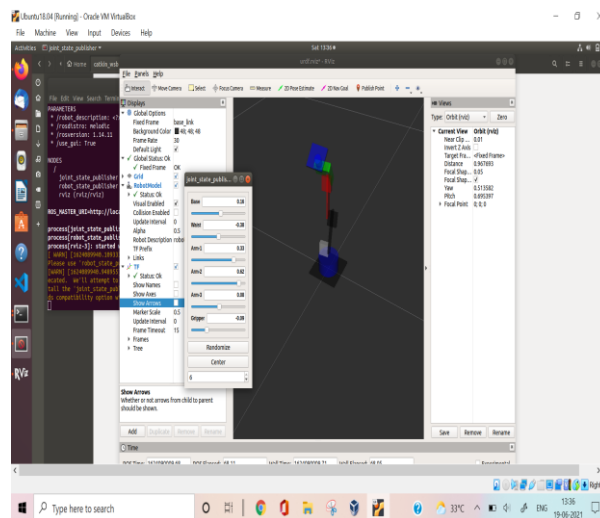

Fig.1 URDF Generation and Joint state publisher analysis of a Robot Arm

B. *Robot Arm Design and Development*

A robot arm contains different links connected together by means of joints that facilitate the motion of the link and hence the motion of the entire arm. The robot arm developed here is articulated with 2 kg payload. The links are rigid elements connected together by joints. The robot arm dealt here has six links which are created using Solid works software. Fig. 2. The joints are connectors that bridge together different links and acts as the key to motion. Fig. 3
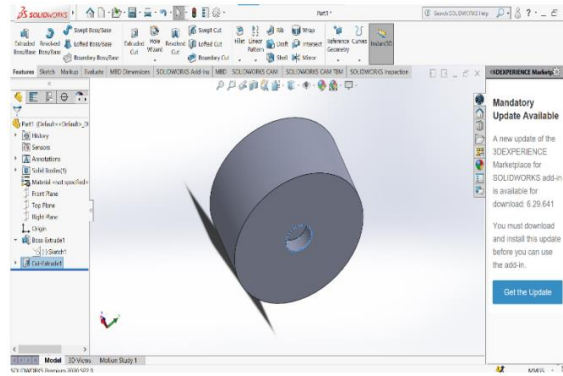

Fig.2 Base Link of Robot Arm created in Solid Works

Fig.3 Assembled Robot Arm created in Solid Works

C. *Visualization in Gazebo*

Gazebo is an open-source 3D robotics simulator. Gazebo integrated the ODE physics engine, OpenGL rendering, and support code for sensor simulation and actuator control. Gazebo can use multiple high-performance physics engines, such as ODE, Bullet, etc (the default is ODE). It provides realistic rendering of environments including high-quality lighting, shadows, and textures [7]. It can model sensors that "see" the simulated environment, such as laser range finders, cameras (including wide-angle), Kinect style sensors, etc. Robot environment development and visualization is one of the major applications of Gazebo. Fig. 4

D. *Virtual Development*

A perfect operation of the robot arm can be achieved by monitoring its performances priorly. Generating a virtual robot arm using URDF in collaboration with Joint state publisher helps to understand robot arm movements clearly [3].

In URDF coding, the initial step after providing section headers is to assign physical properties to the links that are to be created. The physical properties include color, size, dimensions of the links. Then it is followed by colliding the links and joints together. The two major functional variables involved in describing the links are <visual> and <collision> Fig. 5. Each of these functions holds the geometrical properties of links attached and the joint description is followed after that.
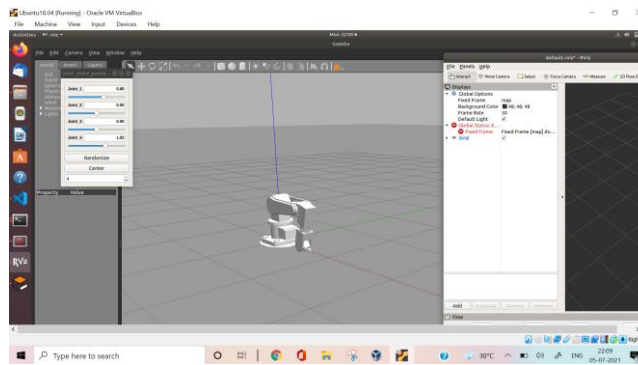

Fig.4 Gazebo Visualization of the Arm

Fig. 5 Code Snippet showing URDF Generation

E. *Orientation of Links and Joints*

The first description has to be about the base link which in turn holds the entire robot arm in correct orientation with the space. The links are described one after the other intervened by their corresponding joints in between them. The link description consists of the mesh part where the link designed and developed in solid works software could be imported [9]. The joint description part contains the parent link and the child link which it connects and also the origin and axes orientations are mentioned. Fig. 6. The reach limit values of the joints along with their velocities are also given as per the requirement.

F. *Joint State Publisher Analysis*

Joint state publisher is a tool that facilitates in publishing and setting joint state values to the provided URDF function [10]. It can be called in a ROS program and a window portraying all the given joints pops up. This would have the minimum and maximum limit values of each and every link starting from the base to the end effector. By dragging the limit values of the corresponding links, the joint movements can be noted. There are also options for random customization as well as

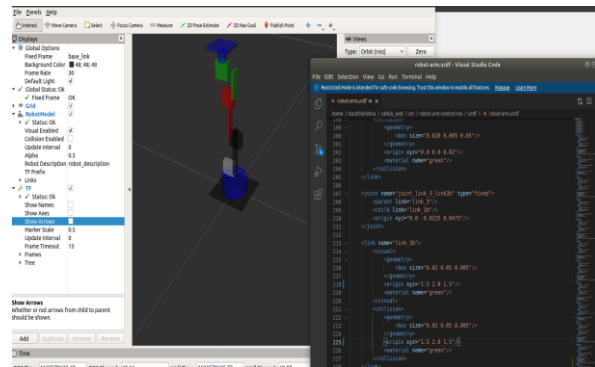center orientation where the arm locates itself to the default position allocated for the particular feature.

Fig. 6 A Robot Arm Generated using URDF

III.   MOVEIT VISUALIZATION

Moveit is a tool for performing various operations on a designed robot arm to help study the properties of the arm such as its position, velocities and acceleration [12]. The designed robot arm has to be configured and exported to URDF so that it can be used with ROS simulation and RViz Visualization and this configuration is facilitated with Moveit.

Moveit setup assistant is a graphical user interfacing tool which aids in configuring robots to be used with moveit for further analysis and inspection [13]. This tool has various configuration functions that helps to format the robot arm as per requirements. After assembling the robot arm in the designing software, in order to export it as URDF, some constraints have to be mentioned and should be followed by moveit setup assistant after getting the generated URDF. The initial step is to import the URDF file generated into the setup assistant by browsing it. Then the files are loaded and the robot arm could be viewed inside the setup assistant. Fig. 7 is depicted below.

Fig. 7 Moveit Setup Assistant

A. *Virtual Joints*

After generating a collision matrix in self collisions feature, the virtual joints have to be defined. Virtual joints are defined because the links of the robotic arm has constraints only with each other and not with the 'world' that it is present. Thus, the base has to be attached to the world in which the robot arm operates. Then, by clicking the 'add virtual joint' option, those joints can be defined. The virtual joint is named and then the link which is to be grounded is given as the child link. The parent frame could be named and the joint type can also be configured here Fig. 8.
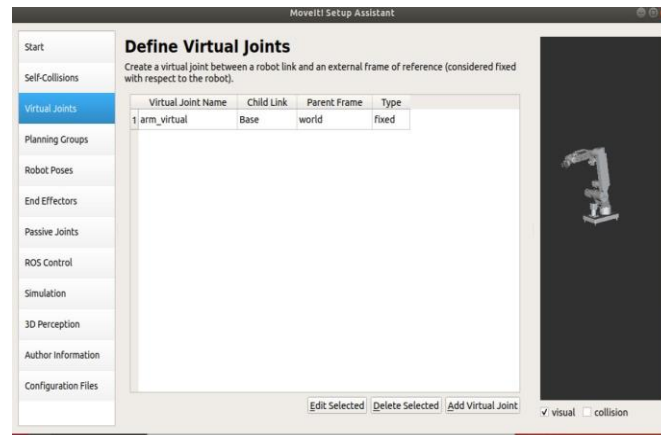
Fig. 8 Virtual Joints Configuration

B. *Planning Groups*

The defining of virtual joints is followed by defining planning groups. A planning group states the set of joint-link pairs considered for planning and checking collisions. Individual groups for every subset of the robot have to be defined and when adding the link to the group, its corresponding parent joint should also be added and vice versa. The group name can be provided and the kinematic solver in which the robot arm is needed to operate can be chosen. Kinematic search resolution and search timeout can also be given. The selected joints are indicated in red colour Fig. 9.

Fig. 9 Defining of Planning Groups

C. *Robot Poses*

These are sets of values of joints for the given planning groups. The first pose of the robot will be the starting position in the simulation. The rest pose can be defined where all the joints are in zeroth limit value. Fig.10.

D. *End Effector Configuration*

Every robot arm has its end effectors designed for its own applications. Defining robot end effectors means that planning groups corresponding to tools, grippers are created. Fig 11. The parent link of the end effector will be used as the reference frame for that particular link. The end effector name and its specific group are specified and then the parent link, that is the link before the end effector is mentioned.
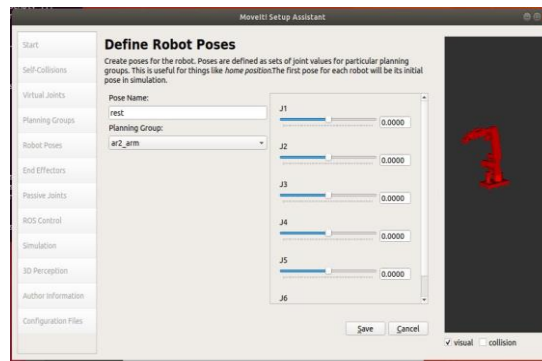
Fig.10 Definition of Robot Poses

Fig.11 End Effector Configuration

E. *ROS Controller Setup*

The Controller name and type have to be given. There are several controller types each having its specific functions. Depending on the purpose the specific type of controller can be chosen. Position, velocity and acceleration controls are available and are to be chosen if needed. Fig. 12.
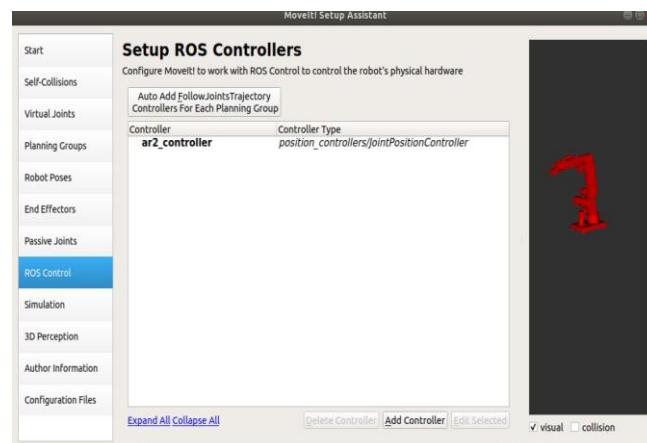

Fig.12 ROS Controller Setup

F. *Miscellaneous Functions*

Some of the other functions present in Moveit setup assistant are the addition of passive joints, setting up 3D perception sensors, providing author information followed by the generation of the configured files [15]. In that last stage, the location of the file that is to be generated has to be

determined and by clicking on the 'Generate Package' button, the package can be generated and the corresponding URDF configuration folder will be created in the given location.

IV.   SIMULATION OF ROBOTIC ARM

The final step after successfully developing the robot arm is to simulate and observe the operations of the arm and gather the inference to develop the hardware model, so that it would operate without any errors and perform the assigned tasks in an effective manner.

A. *Launching Moveit Package*

The terminal is opened in the ROS workspace after ensuring that the moveit package has been created in the desired location. Then the sourcing has to be done. Fig. 13. The launch file created inside the generated URDF configuration folder is launched. It opens up the RViz simulation window.
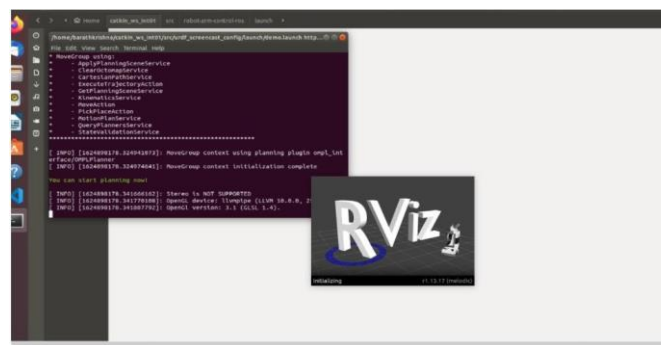

Fig.13 RViz Simulation Window

B. *RViz Inspection*

RViz is a 3D visualization tool used for ROS applications. It provides a view of the robot model, captures sensor information from robot sensors, and replay the captured data [18]. It can display data from lasers, camera, 3D and 2D devices including point clouds and pictures. As soon as the launch file is launched, the RViz window opens with the robot arm available in its workspace. The robotic arm will be perfectly oriented on the surface of the workspace. Then a globe of arrows pops up with which the robot arm can be maneuvered. Simulation of the Robot Arm for path planning. Fig. 14. The RViz window also has the motion planning menu where the movement of the robot arm could be configured.
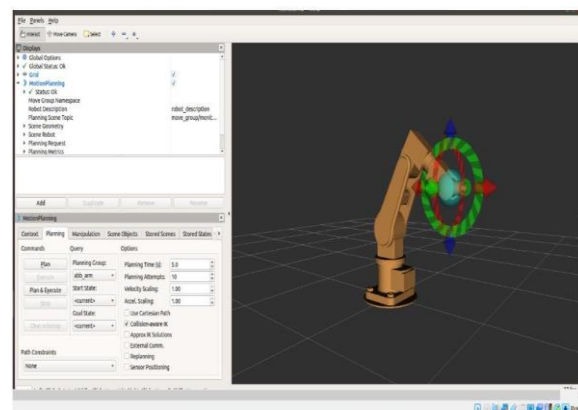

Fig.14 RViz Demonstration

C. *Motion Planning*

The motion planning tab has the features for planning, manipulating and storing the states of the robotic arm. Planning libraries could be added to help in path planning. The other way is to drag the arrow globe that has popped up in the required directions and then pressing 'Plan' option. Then, the robot arm starts to move to the target location in the shortest possible distance. Several path constraints can be provided and the planning time, planning attempts as well as the velocity scaling can be viewed and configured and the plan can be executed. There are options for separate execution and also for combined planning and execution. Fig. 15. An environment can be created around and the application of the arm around its environment can also be noted such as avoiding the obstacles around it or by picking and placing the specified objects.
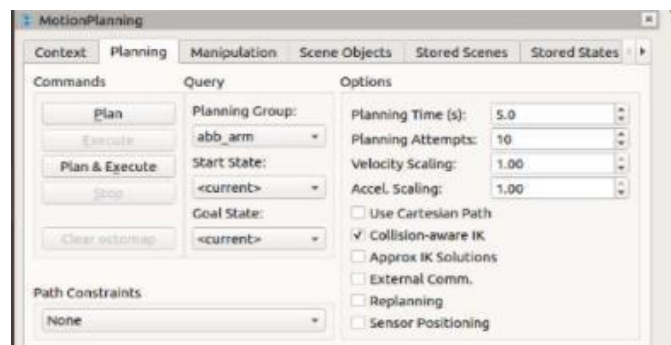

Fig. 15 Motion Planning Window

V.   RESULTS AND DISCUSSIONS

Thus, the arm was simulated and the path of it's motion was configured and executed so that the arm was able to reach exactly to its target location.
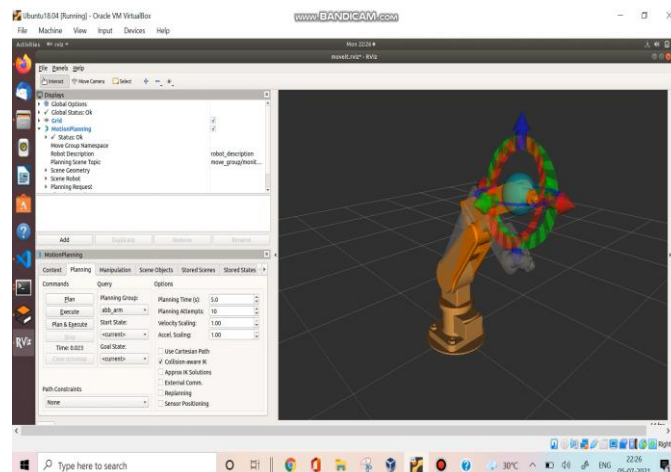

Fig. 16 Performance of the Arm

VI.   CONCLUSION

A perfect performance of the arm relies on the pre-development stage of the hardware model where the arm is virtually designed, developed, configured and simulated using various tools and simulation software. Thus, an articulated robot arm has been designed from scratch, followed by the

configuration of all the required specifications and simulated successfully and the simulation aided in an effective study of the robot arm, helped to understand its operations in an elaborate manner and therefore the details could be shared to the hardware development of the robot arm to guide in the building of an error-free robotic arm suited for the needed applications.

REFERENCES

1. Dr.Ignatius Antony Herman, Mr.Nikhil Amala Jerrin J, Mr.Kishore S,Ms.Cecil Ignatius Hermina, Ms. Sneha N, "Robust Control Platform for Robotic Arm Using ROS", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)., Volume 9, Issue 5, September - October 2020.

2. Eric Rosen, "Communicating and controlling robot arm motion intent through mixedreality head-mounted displays", International Journal of Robotics Research., Volume: 38 issue: 12-13, page(s): 1513-1526, April 12, 2019.

3. S. Hernandez-Mendez, C. Maldonado-Mendez, A. Marin-Hernandez, H. V. Rios- Figueroa, H. Vazquez-Leal and E. R. Palacios-Hernandez, "Design and implementation of a robotic arm using ROS and MoveIt!," 2017 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), 2017, pp. 1-6, doi: 10.11.09

4. Md. Anisur Rahman,et al.,"Design Analysis and Implentation of a Robotic Arm", American Journal of Engineering Research(AJER)., Volume 2 issue:10 pp. 298-307, 2013

5. Sharath Surati,et al.,"Pick and Place Robotic Arm: A review paper, International Research Journal of Engineering and Technology(IRJET)., Volume 08 issue:02 Feb 2021

6. Mohd Ashiq Kamaril Yusoff,et al.,"Wireless Mobile Robotic Arm", International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)

7. K Rajeshekar,et al., "Robotic Arm Control Using Arduino", Journal of Emerging Technologies and Innovative Research., Volume 8 issue:10 October 2021

8. Gurudu Rishank Reddy,et al.,"Design and Structural Analysis of a Robotic Arm, International Scholarly Research Notices(ISRN).,2016

9. I.B. Alit Swamardika,et al.,"Design of Mobile Robot Arm Utilising Microcontroller and Wireless Communication", International Journal of Engineering and Technology(IRJET).,2017

10. B.Sundari,et al.,"Design and Implementation of Robotic Arm using IoT",International Journal of Electronics and Communication Engineering , Volume 5 issue:8 2018

11. J.Iqbal,et al.,"Modern Control Laws for an Articulated Robot Arm",Engineering Technology and Applied Science and Research, Volume 9 issue:2 pp.4057-4061,April 2019

12. Md. Tasnim Rana,et al.,"Design and Construction of a Robotic arm for Industrial Application",International Journal for Engineering Research and Technology, Volume 6 issue:5,May 2017

13. P.U. Chavan,et al.,"Computer Controlled Robotic Arm using Mechanical Reconfigurable Blocks", ",International Journal for Engineering Research and Technology, Volume 3 issue:4,April 2014

14. Surender Kumar,et al.,"Robotic Arm Movement Optimisation Using Soft Computing", International Journal of Robotics and Automation, Volume 6 issue:1

15. S.Pachaiyappan,et al.,"Design and Analysis of an Articulated Robot Arm for Various Industrial Applications",IOSR Journal of Mechanical and Civil Engineering(IOSRJMCE),pp.42-53

16. Kaustubh Ghadge,et al.,"Robotic Arm for Pick and Place Application",International Journal of Mechanical Engineering and Technology(IJMET),Volume 9 issue: 1 pp.125133,January 2018

17. P.Arunagiri,et al.,"Android Based Robotic Arm", International Journal of Future Generation Communication and Networking, Volume 13 issue:3 pp.895-901,2020

18. P.S.Aithal,e.,"A Custom Robotic Arm in Coppealism",International Journal of Applied Engineering and Management Letters(IJAEML), April 2020

19. Said Munzir,et al.,"Optimal Control of Robotic Arm Movement involving Coriolis Centrifugal and Gravitiational Forces, International Journal of Psychosocial Rehabilitation, Volume 24 issue:24 April 2020

20. Yalcin Bulut,et al.,"A gravitational torque-Compensated 2-DOF Planar Robotic Arm Design and its Active Control",Sage Journals" May 17 2021