Quadruped-Gait Analysis

Project Report

Eklavya Mentorship Programme
At
Society of Robotics and Automation,
Veermata Jijabai Technological Institute
Mumbai.
October 2021

Acknowledgement

It was a great experience working on this project with amazing seniors. Constant guidance of seniors was really helpful for us on this project. We are grateful towards you all and thank SRA for giving us this opportunity to learn something new and explore the vast knowledge of technology.

Thanks to our mentors: -

- Anushree Subnis.
- Chaitravi Chalke

...Team Exterminator

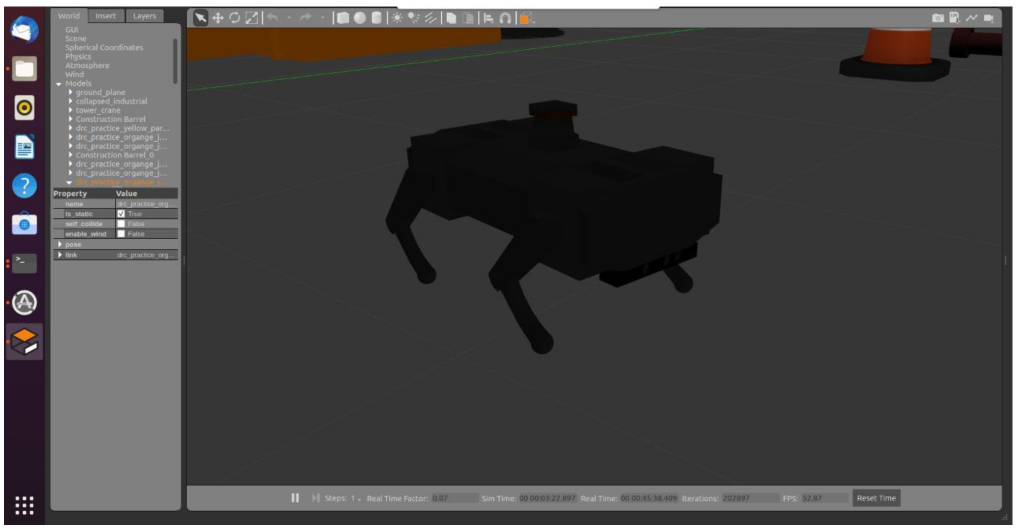
Chaitanya Deshpande

chaitanyad192002@gmail.com

Aniruddha Thakre

1. Overview:

1.1 Description of use cases of this project are as follows:



- Quadruped robots can be made to walk on planes as well as rough surfaces. They have the most stable configuration among the multi legged robots.

- They can be used for transportation on a small scale like for household purposes as well as in factories.

1.2 Technology used:

- **Ros:** The Robot Operating System, or ROS, is a framework for writing robot software. It is a collection of tools, libraries and conventions that aim to specify the task of creating complex and robust robot behaviour across a wide variety of robotic platforms. Ros is used here to simulate the robot.

- **Gazebo**: -Gazebo is an open source 3D robotics simulator. It provides a realistic 3D environment to simulate a robot. Here it is used to provide a world and to simulate the quadruped robot in it.

1.3 Brief Idea: Quadruped Robots, aptly named, these robots have 4 limbs and follow the gait patterns of quadruped animals. The aim of this project is to perform teleoperation(keyboard), forward simulation, obstacle avoidance, turning and reverse motion of a quadruped in ROS/ROS2 using Gazebo simulator.

2. Introduction:

2.1 General:

Our main idea is to find a suitable urdf file for a quadruped robot, set up the urdf by adding plugins and xacro commands into it, find the forward and inverse kinematics equations for the robot and build the code, plan gait pattern for its movements and perform the movements in 3D world in the Gazebo simulator with the help of teleop_twist_keyboard /cmd_vel.

2.2 Basic Domains:

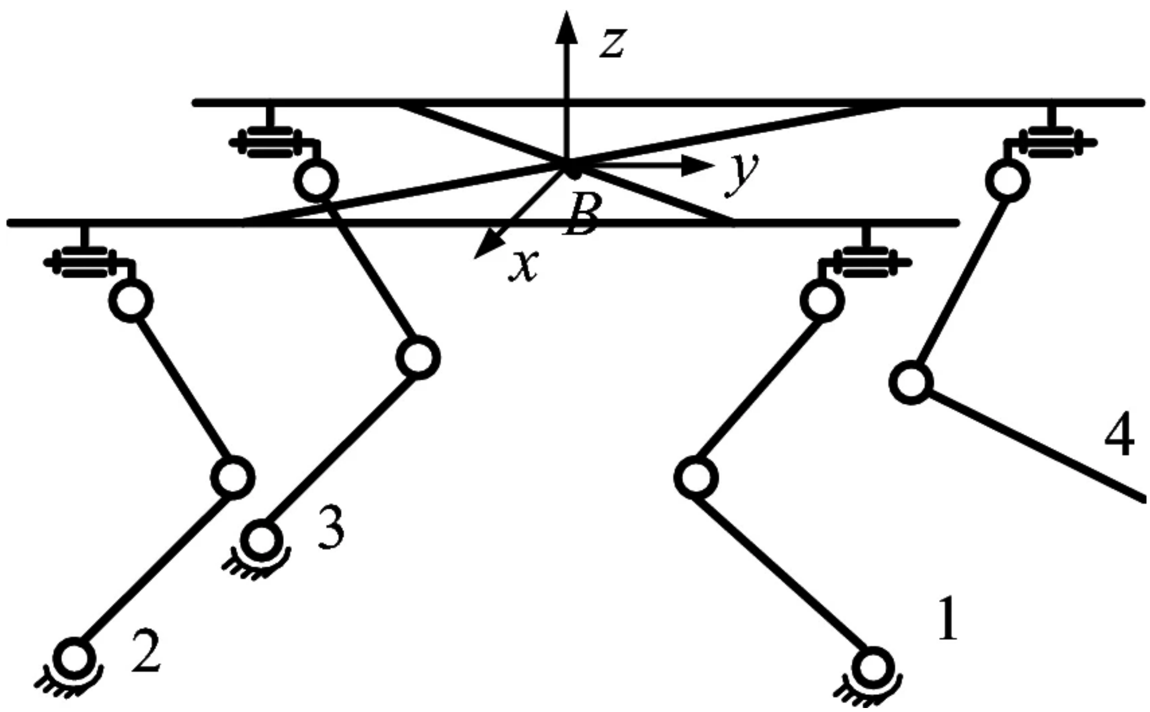
- Forward and inverse kinematics.
- Gait analysis of Quadruped robots.
- Mapping in Ros
- Configurations of planners and controllers.
- Ros.
- Simulation (Gazebo)

2.3 Theory: A Quadruped robot is a 4 legged robot which basically consists of a plane torso and 4 legs which have a revolute joint between them. In our case, the degree of freedom of the robot is 8. Quadruped robot simulation includes the detailed study of gaits and gait patterns. To make a quadruped robot move, we have to study about the gaits and plan the gaits for our tobot. It also includes the study of forward and inverse kinematics. Equations of forward and inverse kinematics are to be found for determination of position and configuration of legs. After the gait planning and implementation of forward and inverse kinematics, the robot is made to simulate in Gazebo simulator with the help of Ros.

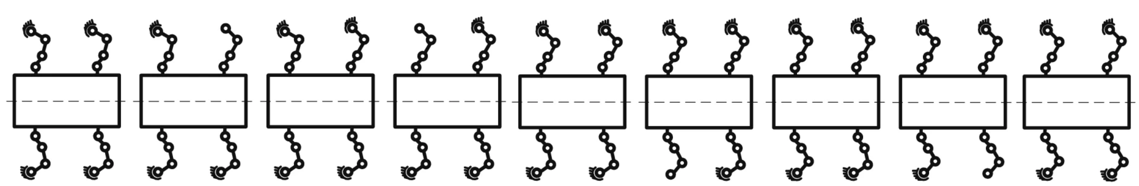
3. Methods and Stages of progress:

3.1 Gaits:

A quadruped robot can walk with statically and dynamically stable gaits. In the statically stable gait, each leg of the robot is lifted up and down sequentially, and there are three stance legs at the least at any moment. This type of gait is named creeping gaits. One gait cycle can be divided into eight different periods of movement. At the beginning of walking, the initial equivalent where four legs are in the stance phase. When one of the legs is lifted, it is transferred to the swing phase; we call this period the step forward stage. In this stage, the equivalent mechanism is. From this, the leg falls and is in contact with the ground until the next leg lifts off; this period is called the switching stage. The equivalent mechanism at this moment exhibits the same configuration as that of the initial period; however, the legs contain different position parameters with the initial period. Four legs of the quadruped robot repeat the motion individually in a certain order from the stance phase to the swing phase, to achieve walking using creeping gait. The step forward stage and switching stage occur alternately while the robot walks. The locomotion of the quadruped robot can be regarded as the movement of these series equivalent mechanisms. The figure below shows the sequence of equivalent mechanisms in one gait cycle. The efficiency of this gait is low because its minimum duty factor is 3/4.



Sequence of equivalent mechanisms in one cycle of creeping gait:



Statically stable gaits, or creeping gaits, utilize the idea of a support pattern formed by the feet in contact with the ground. For a quadruped to remain stable, the projection of its center of

mass must stay within the planar support pattern at all times during the gait sequence. To satisfy
this condition for a quadruped, a creeping gait must always have at least 3 feet in contact with the supporting surface. It is this condition that allows stability to be maintained for arbitrarily slow motion, including complete stops at any time. Support patterns for a statically unstable and Statically stable phases are shown in Figure 2-2. Simply satisfying the condition is not necessarily sufficient for real world systems, however, as dynamic effects and other disturbances are not considered. To provide some quantifiable metric with regards to the amount of stability, a stability margin is calculated based on the distance away from instability

3.2 Kinematics:

The Forward and Inverse Kinematics Solutions for the designed 2 DOF leg of
A quadruped robot is derived by using the Denavit-Hartenberg Convention. These so-

lutions will be applied to simulate and control the robot's movement. Four DH parameters (Denavit-Hartenberg) are used to determine the Kinematics and Inverse Kinematics Solution for each leg. In order to define a relative position and orientation of two fixed axes (axes which do not move), link length (also link distance or common normal) (a) and link twist (a) are required. If there are more than two fixed axes, the neighboring common normals in general case will not intersect the common axis at the same point. Hence, a new parameter called link offset (d) is necessary. The angles of the joint (T) will be determined using the Inverse Kinematics Solutions. Only one of these four parameters will be variable for a single link and others will remain constant. Since the structure uses only revolute joints, in this

case only the joint angles (T) are variable. As mentioned earlier, the joint angles are defined as t1 for hip joint rotation and t2 for knee joint rotation and, a1 and a2 are the length of thigh and leg respectively. The parameters for each leg are given in Table 1.

Since the structure has no prismatic joint, link offset (d) for both the links are zero.

Also, the Z-axis of the second link is 90 degree twisted with respect to the Z-axis of the first link which is the value of link twist (a) for the first link. To represent the equations in an easier way, cosðt1Þand cosðt2Þare defined as c1 and c2, also s1 and s2 represent sinðt1Þand sinðt2Þrespectively. uations, x, y, and z are the final coordinates of the corresponding

Homogeneous transformation matrix H_1^0 for hip joint and H_2^1 for knee joint are given in Eqs. (1) and (2):

$$H_1^0 = \begin{bmatrix} c_1 & 0 & s_1 & a_1 \times c_1 \\ s_1 & 0 & -c_1 & a_1 \times s1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$H_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 \times c_2 \\ s_2 & c_2 & 0 & a_2 \times s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Table 1. The Denavit-Hartenberg parameters (* represents variable).

Link	*a*	*A*	*d*	*T*
L_1	a_1	90	0	t_1^*
L_2	a_2	0	0	t_2^*

axes of the endpoint for Quadruped’s leg. The base motor situated on the origin of Fig. 1 rotates the leg on the Yaw axis/Z axis. Transformation matrix T is used to determine the Inverse Kinematics Solutions. The first three values of the fourth column represents the value of x, y, and z respectively. The Kinematics solutions for the

The above matrices are to be multiplied to determine the total transformation matrix (H_2^0) which will represent the complete system. The total transformation matrix T is given in the Eq. (3) where, $T = H_2^0 = H_1^0 \times H_2^1$:

$$T = \begin{bmatrix} c_1 \times c_2 & -c_1 \times s_2 & s_1 & c_1(a_1 + a_2 \times c_2) \\ s_1 \times c_2 & -s_1 \times s_2 & -c_1 & s_1(a_1 + a_2 \times c_2) \\ s_2 & c2 & 0 & a_2 s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

In the following equations, x, y, and z are the final coordinates of the corresponding axes of the endpoint for Quadruped's leg. The base motor situated on the origin of Fig. 1 rotates the leg on the Yaw axis/Z axis. Transformation matrix T is used to determine the Inverse Kinematics Solutions. The first three values of the fourth column represent the value of x, y, and z respectively. The Kinematics solutions for the following structure is given in Eqs. (4)–(6):

$$x = (a_1 + a_2 * \cos(t_2)) * \cos(t_1) \tag{4}$$

$$y = (a_1 + a_2 * \cos(t_2)) * \sin(t_1) \tag{5}$$

$$z = a_2 * \sin(t_2) \tag{6}$$

The Inverse Kinematics Solutions are given in Eqs. (7) and (8):

$$t_1 = atan\,2(y, x) \tag{7}$$

$$t_2 = atan\,2\left(z, \left(\sqrt{(x^2 + y^2)} - a_1\right)\right) \tag{8}$$

Inverse Kinematics equations (Eqs. (7)e(8)) are necessary because they describe the necessary angles for the motors to change the position of the end of the leg to any desired point.

4. Conclusion:

4.1 What do we learnt:

We learnt about Gaits, gait pattern, creeping gaits, trotting gates, straight gaits, turning gaits, gait tuning. We also learnt the basics of forward and inverse kinematics and how to find DH parameters for a robotic arm. We also learnt about teleoperations of a robot in the gazebo, concept of xacro and working on a urdf file by adding plugins, materials and giving velocities to joints.

4.2 Future works:

1. In this project, we have used pre-made urdf. But we haven't worked on our own made urdf. So we will work on our own urdf in future.
2. Quadruped in this project only moves forward and backward according to the coordinate input. We will work on making the robot turn left and right according to the command.
3. We will work on obstacle avoidance with this robot.

4.3 References:

1. URDF exporting: https://youtu.be/T7X_p_KMwus
2. Champ setup: https://github.com/chvmp/champ_teleop
3. Ros: https://www.ros.org/
4. Gazebo: http://gazebosim.org/
5. Teleoperations in Gazebo: https://www.youtube.com/watch?v=ufYxkNnEFYw

   https://www.generationrobots.com/blog/en/robotic-simulation-scenarios-with-gazebo-and-ros/

   http://gazebosim.org/tutorials?tut=set_velocity

   https://youtube.com/playlist?list=PLK0b4e05Lnzah3QAIsdb0JxAY21YypdZl

6. Gait analysis: https://cjme.springeropen.com/articles/10.1186/s10033-019-0321-2

https://github.com/chaitravi-ce/Eklavya-QuadrupedMotionSimulation