

Mars Sample-Return Rover

ENPM 662 Final Project

Robert Vandemark Diane Ngo

December 2020

Contents

1	Abstract	2
2	Introduction	3
2.1	Motivation	4
2.2	Assumptions	4
3	Robot Design	5
4	Forward Kinematics	7
4.1	Vehicle	7
4.2	Arm	7
5	Inverse Kinematics	12
5.1	Arm	12
6	Velocity Kinematics	14
7	Scope of Achievement	15
8	Model Validation and Testing	16
	Bibliography	17

1 Abstract

Space exploration is still a topic of interest because technology is continuously developing. Exploring and researching Mars is the most convenient way to trial and test new space technologies since it is the closest planet to Earth that is able to conduct research on. NASA approached Mars exploration through the Sample-Return Rover, a robot that can explore unknown terrain autonomously and be able to collect samples. The earliest primitive model was launched in 1997, and the latest one being 2020 with the Curiosity Rover. This project focuses on recreating the earlier version of the rover in SolidWorks and being able to simulate the exploration in a 3D environment with Gazebo.

2 Introduction

This project focuses on the Mars *Sample-Return Rover* (SRR), as seen in Figure 2.1, developed by NASA's Jet Propulsion Laboratory (JPL). Exploring the unknown is fascinating with the challenges that are present. Robotics is a field suitable to approach these challenges because they have the capability to operate autonomously. The model will be created in Solidworks and then simulated in Gazebo. The report explains the elements of the project, starting with the rover's geometric aspects such as its mechanical design, joints, links, and arm. The rover consists of the vehicle base with the chassis and wheels, and an attached *Four Degrees of Freedom* (4DOF) arm.

The rover's chassis has an interesting structure with the front and rear links are connected with one joint, but have independent movement. With a total range of 180 degrees, the rover can scale steep cliffs because of the design. The front wheels can also rotate side-to-side so that the rover can turn. Attached to the chassis is a 4DOF arm. It rotates at its base, an elbow joint, and the wrist joint can rotate sideways and and up and down. The end-effector on the arm allows for sample-collection.

With the challenges of unknown terrain and inability to remote control the robot, extensive research and testing has to be made before launching the robot. The scope of this project focuses on the Sample-Return Rover. Model assumptions will be discussed. Afterwards, the design of the robot will be explained with analysis of the kinematics of the vehicle and the arm.

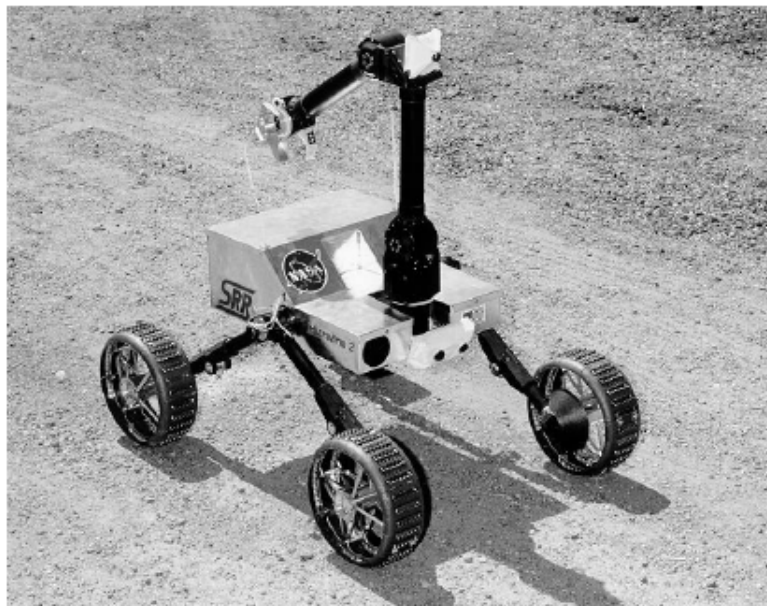


Figure 2.1: Early Model of the Sample Return Rover

Motivation

Space robotics is a big area of interest. Not only has there been the Sample-Return Rover, but there is also the CubeSat, BioSentinel, and many others. The Sample-Return Rover is a great choice for this project as the information gained from the class will be used on the rover. There are many possible usages for space robotics. They can serve the purpose of exploring unknown territory, surveillance of the Earth, or maintenance on existing robots/satellites.

Each different type of robot will deal with their own challenges, whether it is dealing with low or zero gravity, extreme temperature changes, or rocky terrain. It is amazing how diverse space robotics can be. The area of research is vast and takes much effort to be in a robotics field. The intention of this project is to get an introductory hands-on experience with both mobile robots and a manipulator simulation. Calculating forward, inverse, and velocity kinematics are a big topic for this project, as well as implementation of those calculations into algorithms.

Assumptions

Since the scope of this project deals with simulating a robot in outer space, there are assumptions to be made because certain things cannot be measured easily. To deal with this issue and other design factors, there are assumptions to follow.

1. The robot is a rigid body, any torque or force acted upon any link will not be deformed.
2. The steering design is based off the Ackermann-Steering model.
3. All motion or revolution will be around the z-axis.
4. The simulations will be in Gazebo and RVIZ.

3 Robot Design

The robot base comprises of four legs, with two on each side that are concentric. On either side, two of the legs are connected via a revolute joint and is concentric. The front wheels have a revolute joint connected to the wheels so that the robot can steer left or right. To have the robot at the same height, the rear wheels have the same joint however it is fixed. The robot arm that is attached has four degrees of freedom.

Modeling the robot is split into the vehicle and arm separately. With this method, both models can be configured and tested properly in Gazebo in order to use tele-op on individual joints. Figure 3 shows the rover modeled in SolidWorks. Table 3.1 lists the dimensions of the robot and links. Figure 3 and Figure 3 are images of the arm for reference.

<i>Link</i>	<i>Length</i>
Chassis Length	18in
Chassis Width	12in
Chassis Max Height	7in
Leg from Center Joint	8.6in
Length of Steering Link	5.25in
Wheel Diameter	7.5in
Wheel Track	19.3in
Wheelbase	13.9in

Table 3.1: Table of Dimensions of the Rover

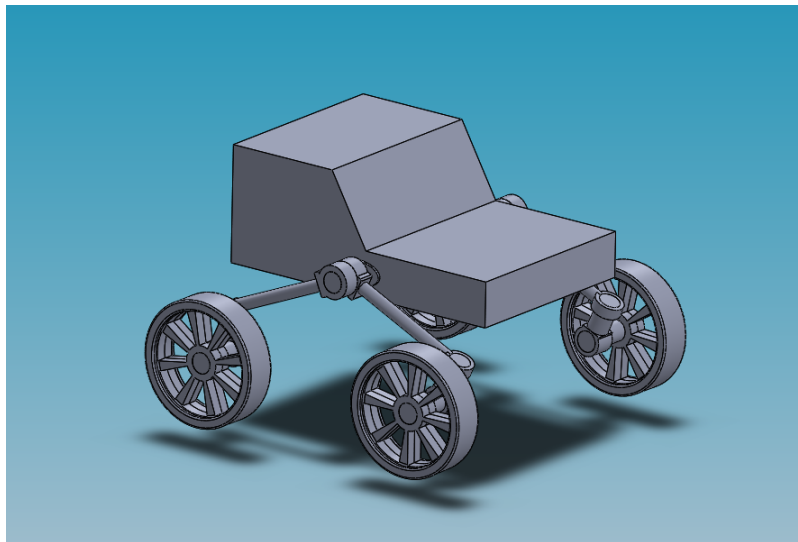


Figure 3.1: Sample Return Rover in SolidWorks

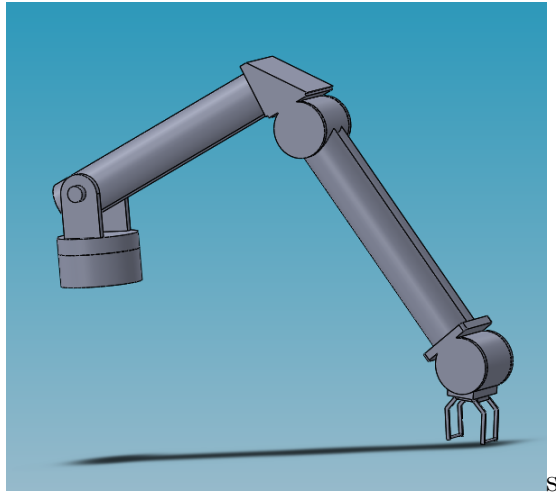


Figure 3.2: Arm Model in SolidWorks

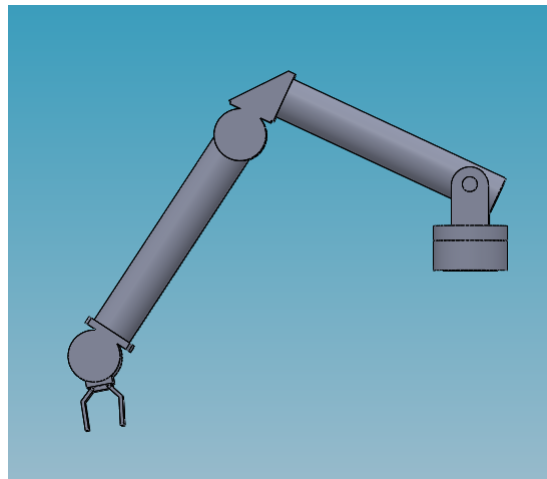


Figure 3.3: Side View of Robot Arm

4 Forward Kinematics

Vehicle

The rover uses an Ackermann steering mechanism to be able to navigate, like that seen in Figure 4.1. Unlike the average robotic arm, it is not a serial connection. Forward kinematics cannot be used on a mobile robot. Ackermann steering allows the front wheels to rotate independently, about a common center point. This center point is called the *Instantaneous Center of Curvature* (ICC). In general, it uses a four bar linkage in a trapezoidal shape. However the Curiosity Mars Rover and the Sample-Return Rover does not use this linkage, but the model still holds.

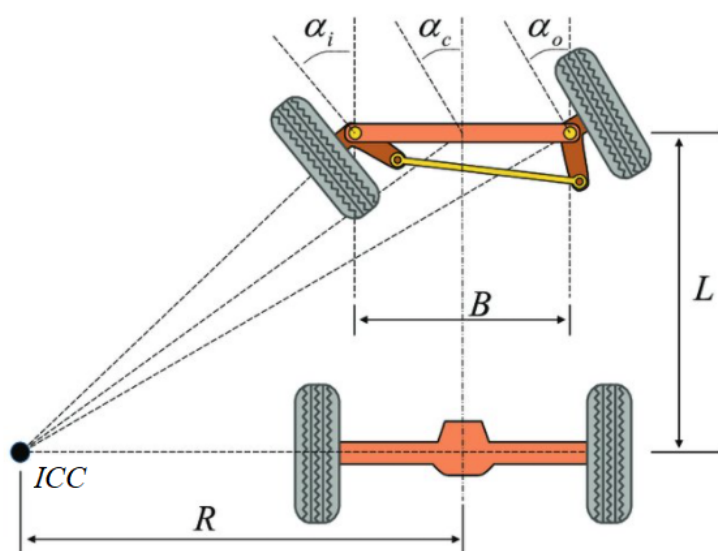


Figure 4.1: Diagram for Ackermann Steering

Arm

In robotics, forward kinematics is used to find the position and orientation of the robot's end-effector (or gripper) given the joint angles. In this course, joints can have two different types: *revolute* and *prismatic*. A revolute joint rotates about an axis, while a prismatic joint translates linearly along an axis. In serial kinematic chains, such as the arm seen in Figure 3, each coordinate frame assigned to the distal end of a link i is dependent on the position and geometry of the $(i - 1)^{th}$ joint and link, respectively. The frame i resolved in frame $i - 1$ represents a combination of a rotation performed by the $(i - 1)^{th}$ joint with a successive translation down the length of link $i - 1$. Chaining these homogeneous transformation matrices together results in the transformation of the end-effector in the

robot's base frame.

The robot arm itself is 4DOF where each joint is revolute. However, as seen in Figure 4.2, there are eight frames in total, because four dummy frames are for Gazebo to properly render in the arm. Its corresponding DH table is shown in Table 4.2, and the values for the nonzero static lengths can be seen in Table 4.2. A list of the links are shown in Figure 4.2. The links *LeftFinger* and *RightFinger* are just there for Gazebo to be able to rotate the end-effector fingers, but they are not used in the DH parameters and table.

Note that the arm's home position starts with all angles being at 0, so the arm is fully upward. Each joint has a limit for its range of motion so that the robot does not crash into itself. Equation 4.1 to 4.4 offer reasonable ranges of allowable motion.

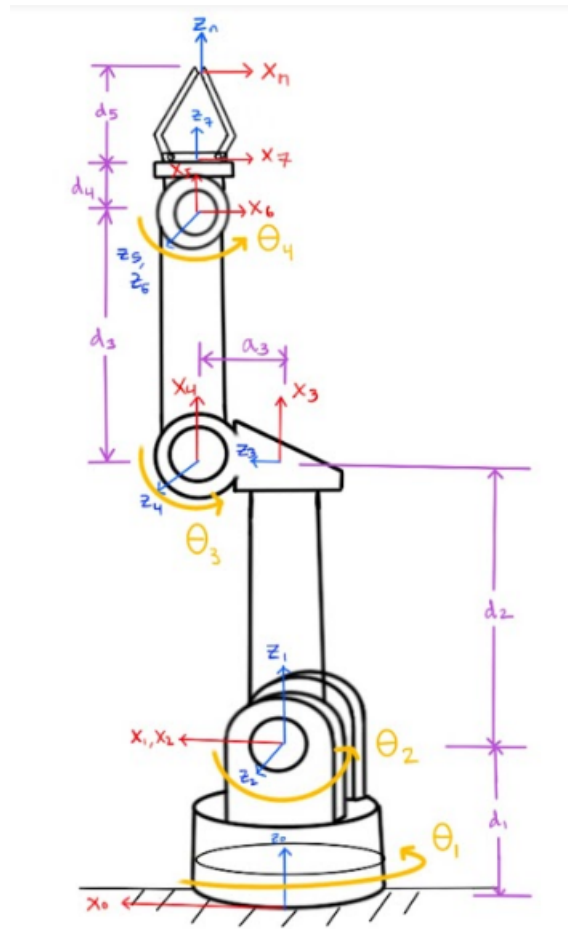


Figure 4.2: DH Frames of the Arm

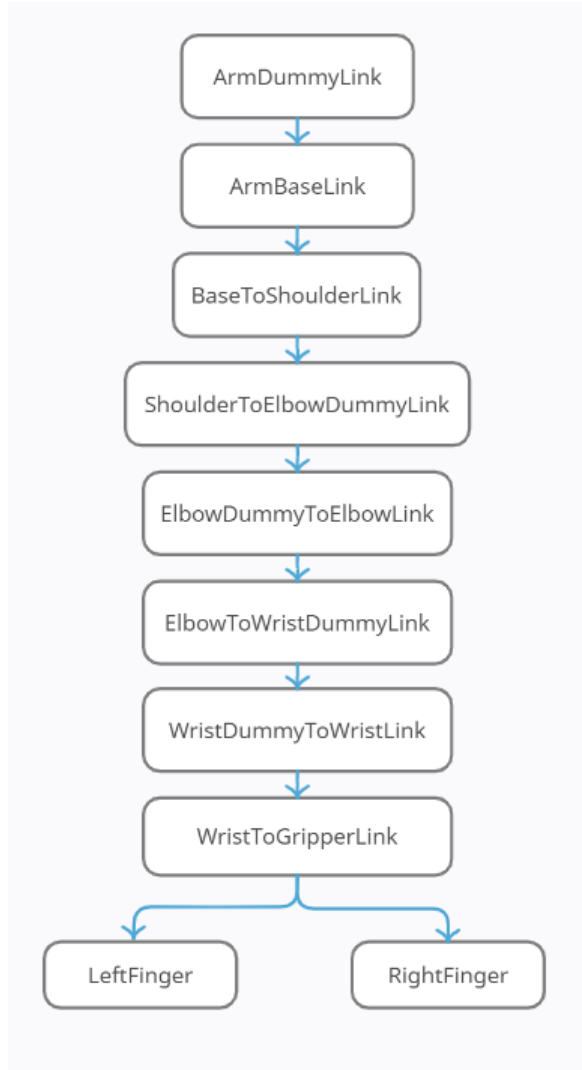


Figure 4.3: Flowchart of the Arm Links

<i>Link i</i>	θ	d	α	a
1	θ_1	d_1	0	0
2	0	0	-90	0
3	$\theta_2 - 90$	0	-90	d_2
4	0	a_3	90	0
5	θ_3	0	0	d_3
6	$\theta_4 - 90$	0	-90	0
7	0	d_4	0	0
n	0	d_5	0	0

Table 4.1: DH Table for the Arm

<i>Distance between Frames</i>	<i>Length</i>
Base to Frame 1 (d_1)	2.9in
Frame 2 to Frame 3 (d_2)	7.73in
Frame 3 to Frame 4 (a_3)	1.76in
Frame 4 to Frame 5 (d_3)	8.93in
Frame 6 to Frame 7 (d_4)	1in
Frame 7 to Frame n (d_5)	1.41in

Table 4.2: Table of Link Lengths of the Arm

$$\theta_1 = [-\pi, +\pi] \quad (4.1)$$

$$\theta_2 = \left[0, +\frac{2\pi}{3}\right] \quad (4.2)$$

$$\theta_3 = [0, +\pi] \quad (4.3)$$

$$\theta_4 = \left[-\frac{2\pi}{3}, +\frac{2\pi}{3}\right] \quad (4.4)$$

One method to start calculating the forward kinematics is to create DH matrices T_n^{n-1} , which are composed of rotations about the z-axis, translations in the z-axis, rotations about the x-axis, and finally translations in the x-axis. The basic rotation matrix for a rotation about the z-axis with an angle θ can be shown in Equation 4.5, and Equation 4.6 for rotation about the x-axis by angle α . The translation vectors can be seen in Equations 4.8 and 4.7 for along the z and x axes by a displacement d and a , respectively. Using the DH Table, the homogeneous transformation matrices can now be written. The setup for this matrix is seen in Equation 4.9.

$$R_{z,\theta} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (4.6)$$

$$t_{n_z}^{n-1} = \begin{bmatrix} 0 \\ 0 \\ d \end{bmatrix} \quad (4.7)$$

$$t_{n_x}^{n-1} = \begin{bmatrix} a \\ 0 \\ 0 \end{bmatrix} \quad (4.8)$$

$$H_n^0 = \begin{bmatrix} R_n^0 & t_n^0 \\ 0 & 1 \end{bmatrix} \quad (4.9)$$

Another method is to observe the geometry of the workspace, which is the set of positions and orientations that the end effector can accomplish, and find the restrictions created by it. This type of approach was taken for this project to solve for the four dimensions of actuation, the 3D position $[x_n^0 \ y_n^0 \ z_n^0]^T$ and a rotation off the z-axis of the robot's shoulder joint γ .

The base joint is the only one that rotates about the base's z-axis, so the values for x_n^0 and y_n^0 can be thought of as a projection of the arm on to the (x_0, y_0) plane, where the length of this projection is a function of the shoulder, elbow, and wrist joints, as well as their child links. This length rotates about the z axis with the base joint. This geometry describes how to derive x_n^0 and y_n^0 , seen in Equations 4.10 and 4.11. The z coordinate by definition is the displacement of linkage in the z-axis. Considering the fact that the rest of the three joints act in the xz plane, the solution can be solved as a summation of the length of each link that traverses its local x-axis times the cosine of the net angle and the length of each link that traverses its local z-axis times the sine of the net angle. For this arm, that results in Equation 4.12. Finally, because γ is essentially the angle between the xy plane of the base and the surface that the vehicle is on, assuming that the vehicle is flat relative to the surface, then γ can be solved with Equation 4.13.

$$x_n^0 = c_1 (d_2 s_2 + a_2 c_2 + d_3 s_{23} + (d_4 + d_5) s_{234}) \quad (4.10)$$

$$y_n^0 = s_1 (d_2 s_2 + a_2 c_2 + d_3 s_{23} + (d_4 + d_5) s_{234}) \quad (4.11)$$

$$z_n^0 = d_1 + d_2 c_2 - a_2 s_2 + d_3 c_{23} + (d_4 + d_5) c_{234} \quad (4.12)$$

$$\gamma = \theta_2 + \theta_3 + \theta_4 \quad (4.13)$$

Where:

$$c_{ij...} = \cos(\theta_i + \theta_j + ...) \quad (4.14)$$

$$s_{ij...} = \sin(\theta_i + \theta_j + ...) \quad (4.15)$$

5 Inverse Kinematics

Arm

Inverse Kinematics can be thought of as the opposite of Forward Kinematics, such that it solves for all of the possible sets of joint angles that resolves the end effector to a desired position relative to the robot's base frame. In this case H is the desired position and orientation. Equation 5.1 is the equation where one or more solutions needs to be solved in order to find the joint angles (q_i).

$$T_n^0(q_1, \dots, q_n) = H_1(q_1) \cdots H_n(q_n) = H \quad (5.1)$$

Inverse kinematics can be far more complex than solving forward kinematics. In some cases an unsolvable problem will be encountered. A typical approach to solving inverse kinematics problems is called **kinematic decoupling**. Essentially it approaches the problem by breaking it down into two subproblems, with the first is calculating the position of the wrist center (intersection between the wrist axes) and then finding the orientation of the wrist center.[2]

For the arm, the easiest start is to solve for θ_1 . With the desired x_n^0 and y_n^0 , the $\arctan(y/x)$ can be used.

$$\frac{y_n^0}{x_n^0} = \frac{s_1[d_2s_2 + a_2c_2 + d_3s_{23} + (d_4 + d_5)s_{234}]}{c_1[d_2s_2 + a_2c_2 + d_3s_{23} + (d_4 + d_5)s_{234}]} \quad (5.2)$$

The numerator and denominator cancels out. The term (s_1/c_1) can be written as $\tan(\theta_1)$. The equation is now

$$\frac{y_n^0}{x_n^0} = \tan(\theta_1) \quad (5.3)$$

Now take the $\arctan(\theta_1)$ to solve for (θ_1) .

$$\theta_1 = \text{atan}\left(\frac{y_n^0}{x_n^0}\right) = \text{atan2}(y_n^0, x_n^0) \quad (5.4)$$

θ_1 is the only revolution in the xy plane. Therefore after θ_1 , the remaining joints turn the robot arm to a planar manipulator, meaning all the movement is parallel. Now, the equations for link i are:

$$\begin{aligned} x_n^i &= d_2c_2 + a_2s_2 + d_3c_{23} + (d_4 + d_5)c_{234} \\ y_n^i &= d_2s_2 + a_2c_2 + d_3s_{23} + (d_4 + d_5)s_{234} \\ z_n^i &= 0 \end{aligned} \quad (5.5)$$

Let's substitute $\gamma = \theta_2 + \theta_3 + \theta_4$

$$x_n^i = d_2c_2 + a_2s_2 + d_3c_{23} + (d_4 + d_5)c_\gamma$$

$$y_n^i = d_2 s_2 + a_2 c_2 + d_3 s_{23} + (d_4 + d_5) s_\gamma$$

$$\Rightarrow x' = x_n^i - (d_4 + d_5) c_\gamma = d_2 c_2 + a_2 s_2 + d_3 c_{23} \quad (5.6)$$

$$\Rightarrow y' = y_n^i - (d_4 + d_5) s_\gamma = d_2 s_2 + a_2 c_2 + d_3 s_{23} \quad (5.7)$$

From here, square both equations to add them and simplify. Equation 5.8.

$$\Rightarrow (-2x'd_2 - 2y'a_2)c_2 + (-2x'a_2 - 2y'd_2)s_2 + (x'^2 + y'^2 + d_2^2 + a_2^2 - d_3^2) \quad (5.8)$$

Equation 5.8 can be compared to $Pc_\beta + Qs_\beta + R = 0$ in which Equation 5.9 can be solved with γ :

$$c_\gamma = \frac{P}{\sqrt{P^2 + Q^2}}, \quad s_\gamma = \frac{Q}{\sqrt{P^2 + Q^2}} \quad (5.9)$$

$$\Rightarrow \delta = \text{atan2}\left(\frac{Q}{\sqrt{P^2 + Q^2}}, \frac{P}{\sqrt{P^2 + Q^2}}\right)$$

$$\Rightarrow c_\gamma c_\beta + s_\gamma s_\beta + \frac{R}{\sqrt{P^2 + Q^2}} = 0$$

$$\beta = \gamma \pm \cos^{-1}\left(\frac{-R}{\sqrt{P^2 + Q^2}}\right) \quad (5.10)$$

θ_2 can be set to

$$\theta_2 = \gamma \pm \cos^{-1} \quad (5.11)$$

Equation 5.10 and given γ , P , Q , and R then θ_2 and θ_3 can be solved for:

$$P = -2x'd_2 - 2y'a_2, \quad (5.12)$$

$$Q = -2x'a_2 - 2y'd_2, \quad (5.13)$$

$$R = x'^2 + y'^2 + d_2^2 + a_2^2 - d_3^2 \quad (5.14)$$

Then using Equation 5.6 and 5.7, cancel out d_3 and then divide y'/x' to get:

$$\tan(\theta_2 + \theta_3) = \text{atan2}\left(\frac{y' - d_2 s_2 - a_2 c_2}{x' - d_2 c_2 + a_2 s_2}\right) \quad (5.15)$$

$$\theta_3 = \text{atan2}(y' - d_2 s_2 - a_2 c_2, x' - d_2 c_2 + a_2 s_2) - \theta_2$$

Remember that $\gamma = \theta_2 + \theta_3 + \theta_4$, solve for θ_4 to get

$$\theta_4 = \gamma - \theta_2 - \theta_3 \quad (5.16)$$

Finally, resolving the (x,y) coordinates of the end-effector in the i^{th} frame yields

$$[x_n^i \ y_n^i \ z_n^i]^T = P_n^i, \quad [x_n^0 \ y_n^0 \ z_n^0]^T = P_n^0 \quad (5.17)$$

$$\Rightarrow P_n^i = T_0^i \cdot P_n^0 = (T_1^0 \cdot T_2^1 \cdot T_i^2)^{-1} \cdot P_n^0 \quad (5.18)$$

6 Velocity Kinematics

7 Scope of Achievement

For the scope of achievement and study, milestones and careful planning had to be made. Most of the concepts learned in this class were able to translate over to this project. A list of milestones are presented below, in order of planning and achievement.

1. Build the vehicle part of the rover
2. Create the ROS workspace environment
3. Create a terrain similar to Mars terrain in Gazebo
4. Test the vehicle URDF and tune the controls
5. Build the arm for the rover
6. Calculate and create scripts for the kinematics of the arm

8 Model Validation and Testing

Bibliography

- [1] John J. Craig. *Introduction to Robotics: Mechanics and Control*. 2nd ed. Reading, Massachusetts: Addison-Wesley, 1989. ISBN: 0201095289.
- [2] S. Hutchinson M. W. Spong and M. Vidyasagar. *Robot Modeling and Control*. 1st ed. John Wiley and Sons, Inc., 2001.
- [3] T. L. Huntsberger P. S. Schenker and G. T. McKee University of Reading (UK) P. Pirjanian Jet Propulsion Laboratory (USA). “Robotic Autonomy for Space: Cooperative and Reconfigurable Mobile Surface Systems”. In: i-SAIRAS 2001. Canadian Space Agency, St. Hubert, Quebec, Canada, 2001.