Since the scenario requires trajectory planning in order for the robot to avoid the obstacle. In this chapter trajectories that are polynomial functions are investigated. To simplify the problem a robot with a single rotary joint is used.

Firstly, trajectories without obstacles are considered.

As the robot has to be at zero velocity a cubic polynomial is need to fulfil all requirements of the trajectory. The joint angle and velocity can be described as a function of time with the coefficients .

Furthermore, there are four boundary conditions, where the joint angles and the end time are known values.

With that a system of equations can be put into a matrix to deduce the coefficients.

When using a cubic polynomial the start and end acceleration does not equal zero, which is a problem in reality, because a sudden acceleration or deceleration is impossible. To meet this a quintic polynomial function is required.

Two more boundary conditions can be formulated. The other four are still valid.

This leads to another system of equations which can be solved for the coefficients.

When there are obstacles in the workspace of a robot it is often required to define via points for the trajectory in order to avoid collisions. Assume that the via point is crossed at then there is one more boundary.

With that the cubic polynomial has to be expanded to a quartic polynomial.

The coefficients can be derived from the following equation system.

The same logic applies for the quintic polynomial, which becomes a hexic polynomial.

Solving for the coefficients.

Another way of connecting the via point to the trajectory is by using two separate polynomials for the first and second part of the movement. It is required that the speed and acceleration at the via point are constant.

The following conditions apply for this case. Assume

This leads to the following system of equations.

The last method to find the trajectory is by connecting two quintic polynomials.

There are four more conditions than in the previous method.

The following system of equations can then be used to find the coefficients.

A MATLAB script is developed to find and visualize the trajectories. In order to deduce the coefficients for the polynomial functions, MATLAB functions are written where the previously shown equations are implemented.