# Motion planning for a six degrees of freedom Puma 560 manipulator using trapezoidal, polynomial and screw linear interpolation ⋆

**Daniel Moran-Garcia** * **Lex Brunett-Chavez** **
**Adrian Delgado-Zambrano** *** **Hector Trivino-Gonzalez** ****

* *Escuela Superior Politécnica del Litoral, ESPOL, Guayaquil-Ecuador (e-mail: damogarc@espol.edu.ec ).*
** *(e-mail: lbrunett@espol.edu.ec)*
*** *(e-mail: jodezamb@espol.edu.ec)*
**** *(e-mail: htrivino@espol.edu.ec)*

**Abstract:** Puma 560 Manipulator for a six-degrees-of-Freedom or Dof to minimise using a end effector for welding, our group designed two algorithms for the motion planning in a square trajectory to compare these methods and determine the differences. Our end effector moved by four different coordinates representing the vertices of the square, our manipulator is subjected to some constraints for welding explained in the document. The algorithms used are the trapezoidal and the polynomial interpolation based in the coordinates for the path A-B-C-D coordinates in the square and using inverse kinematics for the path and plotting a animation of the movement and some graphics for the analysis.

*Keywords:* Puma 560, Motion Planing algorithm, welding robot, trajectory, Trapezoidal Interpolation, Polynomial Interpolation, Joint Displacement, Inverse Kinematics.

## 1. INTRODUCTION

One of the main problems in robotics, is to bring accurately and naturally the positioning of the final effector of the robot, this due to errors and disturbance that affect the result of the final position of the manipulator, the control is complicated and Expensive, in a world where manipulative robots are in different industries, position errors should be reduced when controlling the manipulator. Simó Villanueva (2018)

In the following study, it is based on modeling a welding trajectory using the Six-degree Puma 560 robot model, this robot model was chosen because it has been in 1980 in the industry, it has different applications, from painting or Welding, even in the assembly area. One of the most important parts of this study the approaches of the solution of the problem through the kinematics inverse of it. Piltan et al. (2012)

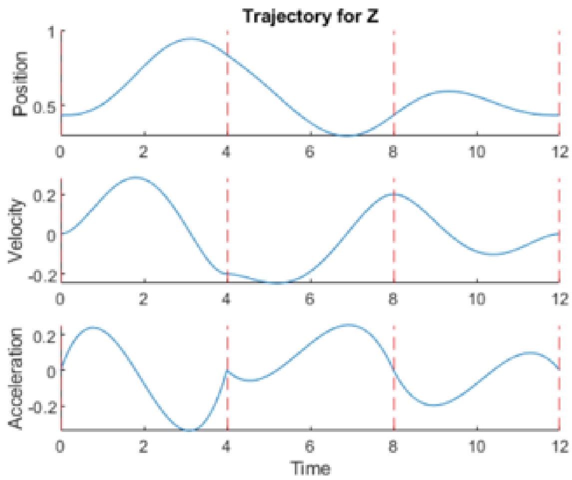
This way, the geometric figure to generate is a square on a horizontal plane, and to solve this problem it will be required to implement the trapezoidal profile to get the functions that describes the relations between displacement, velocity, and acceleration over time. The algorithm to find the joint positions is, first the points of the square should be defined, then we will use the trapezoidal profile approach to get the trajectory of the end effector this way inside a loop inverse kinematics will be used to find the joint positions. Zha (2002)

The Puma Robot comes from the acronym "Programmable Universal Machine for Assembly" developed by General Motors, The PUMA 560 robot is a general-purpose robot, to what is referred by volume of work of an articulated arm, in which the six axes are rotating, the first three form the movements of the upper arm, and the last three correspond To the movement of the wrist, so you can reach any point within your envelope in any direction. Olivares Fong (2016)

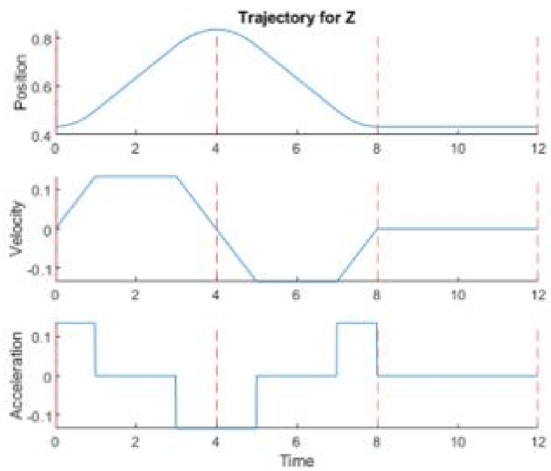
It is important to consider the main differences between the curves of the trapezoidal and polynomial interpolation as shown in the Fig. 1, the main difference is that trapezoidal velocity are piece-wise trajectories of constant acceleration, zero acceleration and constant deceleration, that is the reason why the velocity profile is a trapezoid, meanwhile polynomial trajectories are very useful for continuously stitching together segments with zero or nonzero velocity and acceleration due to its smooth curves.

However, the screw interpolation will also be used. Linear interpolation (sclerp). In this document we present a workspace. that has a focus on route planning based on the interpolation of linear screws, this can be calculated for a path in the workspace and use pseudoinverse of the Jacobian to calculate the corresponding. Path in the joint space. Sarker et al. (2020)

⋆ This is a paper showing different methods of planning the motion for a Puma manipulator and their comparisons.




Characteristic curve of polynomial interpolation




Characteristic curve of trapezoidal interpolation

Fig. 1. Motion curves of trapezoidal and polynomial path over time

## 2. METHODOLOGY

### *2.1 Problem Statement*

Our problem is the current efficiency in car manufacturing in Ecuador, a great example is CIAUTO in Ambato, Ecuador. They build a car in a hour, assembling many Great Wall Motors cars such as SUVs, Sedans and much other. One of the most important task in assembling a car is the welding of the chassis which use Human welders for all the cars.

We needed to design a manipulator capable of weld any car parts which CIAUTO needs to weld. For reason of the project we began planning the trajectory of a square, the algorithms are capable of make the trajectory in four coordinates to weld precisely in each vertex of the square. This project can help to design different moving planning for a real car part to optimise the manufacturing of these cars.

### *2.2 Problem solving and tools used*

For the beginning of our code we implemented Robotics Tool Box in python to describe our Puma560 and add the configuration of the six-degrees-of-freedom and proceed with two-coordinate-moving-planing example and plot the six-joints-coordinate in function of time. The interpolation methods to use in the project are described below.

*Trapezoidal Interpolation:* Trapezoidal velocity trajectories are basically trajectories where the acceleration can be a Piecewise function, but it always has a constant value, that is, it can be zero, or take positive or negative values. From this assertion we can say that the velocity will be composed of linear functions, and the position of the effector, which would pass through the points of interest of our problem, would be a Piecewise function of second degree polynomials. Castro (2019)

*Polynomial interpolation:* You can interpolate between two reference points using multi-order polynomials, the most common are third and fifth order but higher order trajectories can be used. Polynomial paths are useful for continuously joining segments with different values of velocity and acceleration. Because, unlike trapezoidal trajectories, you use smooth and Absolutely integrable function steel profiles with relative ease. However, validating them is more difficult due to the continuity of the curve.

*Screw linear interpolation:* The methodology used to obtain the route in the task space as follows:

(1) fix the initial configuration and that of each end effector, in such a way that the relative configuration between them at the beginning and in the end is the same and also for each end effector, the orientation at the beginning and at the end must be equal.
(2) Plan the route for each end effector with the same interpolation parameter. The resulting road will satisfy the desired movement restrictions, that is, The final effects will have a constant relative configuration and Maintain the constant orientation of the tray during everything. path. Sarker et al. (2020)

To generate the trajectory to be followed by our Puma robot, we first evaluated the different options at our disposal. To generate the trajectories, we had two different methods, task space and joint space. Task space trajectory is based on the following algorithm. Castro (2019)

(1) Task space way-points between the two coordinates.
(2) Find the task space trajectory with an interpolation method.
(3) Inverse kinematics.
(4) Joint position commands.

The joint space trajectory algorithm is shown below.

(1) Task space way-points between the two coordinates.
(2) Inverse kinematics between the coordinates.
(3) Find the joint space trajectory.
(4) Joint position commands.

The main difference between both methods is that on task space trajectory step (3) and (4) are periodic, meanwhile using joint space trajectory only step (4) is periodic. Polynomial and trapezoidal interpolation methods are used in this project, we decided to use the task space trajectory method because of the path we expect to generate with the mentioned interpolation method (an square on a plane).

Screw linear interpolation was also used with the following algorithm, it is important to consider that in the case below, $t$ is at any step of the iteration.

(1) The algorithm requires the pose of the end effector $\mathbf{g}(0)$ and the joints $\Theta(0)$.
(2) Conversion of $4 \times 4$ homogeneous matrix $\mathbf{g}(t)$ to $7 \times 1$ vector of concatenation of position and unit quaternion $\gamma(t)$, and unit dual quaternion representation $\mathbf{A}(t)$.
(3) Coputing of $\mathbf{A}(t+1)$ using unit dual quaternion operations. In this project we considered $\tau = 1$.
(4) Conversion of unit dual quaternion representation $\mathbf{A}(t+1)$ to a vector of concatenation of position and unit quaternion $\gamma(t+1)$.
(5) Then, it is required to compute $\Theta(t+1)$.
(6) Finally forward kinematics is solved with $\Theta(t+1)$ computed.
(7) We defined $\varepsilon = 0.001$, once the error es smaller than $\varepsilon$, the iteration stopped.

The code is available on a public repository on **GitHub**.

## 3. VALIDATION, RESULTS AND COMPARISON

The coordinates to complete the square are the following:

$$A = (0.30, 0.15, 0)$$
$$B = (0.15, 0.15, 0)$$
$$C = (0.15, 0.30, 0)$$
$$D = (0.30, 0.30, 0)$$

It is important to consider that $z$-coordinate is zero, which means than the end effector will not change its position in this plane. After developing the code with the corresponding algorithms, the resulting curves of the end effector analysis of motion are shown below, the scripts and the animation are available on a repository on **GitHub**. The motion analysis of the end effector are shown on Fig. 2, 4, 6 with the analysis of position, velocity and acceleration vs steps, it is important to consider that *steps* was a constant value defined as 20.

Considering that there are four paths that the end effector has to follow, the must be 80 steps to follow as shown on the $x$-axis of the figures while using the trapezoidal and polynomial interpolation, according to the algorithm applied with screw linear interpolation, it stopped when the error $\varepsilon$ was smaller than 0.001, which is a constant defined.

Fig. 2 shows the position of the end effector by using the trapezoidal, polynomial and screw linear interpolation. Polynomial and trapezoidal have a very similar behaviour although the trapezoidal approximation involves a quadratic and a linear section, it is complex to find a difference between both curves, as mentioned before, there is no displacement on the $z$-plane, that is the reason why this curve does not have any change, meanwhile screw linear interpolation curve is considerably different

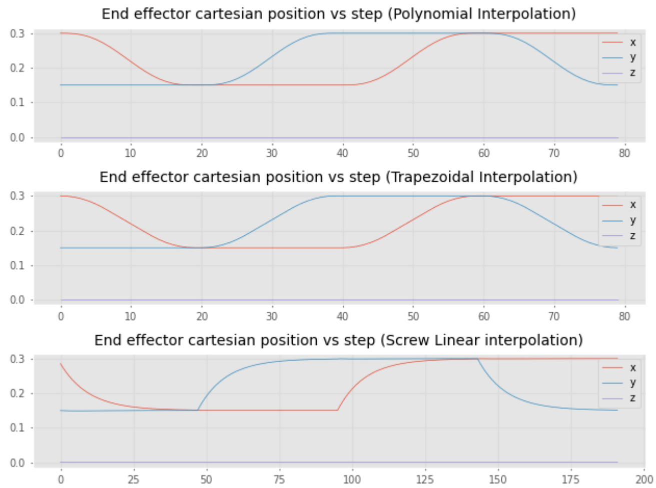



Fig. 2. End effector displacement vs steps

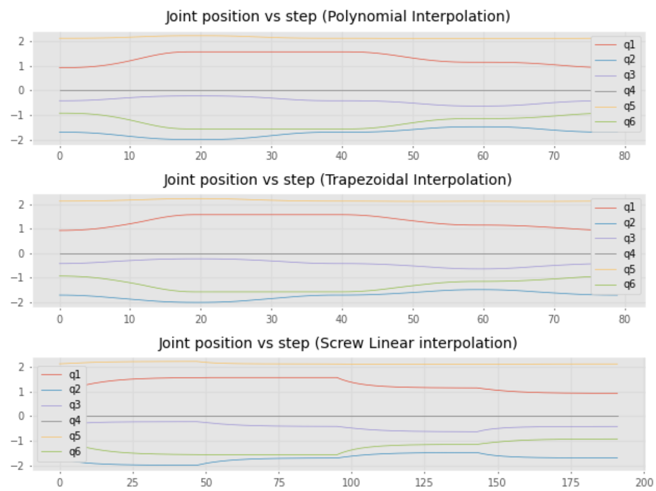



Fig. 3. Joint position vs step

than the other methods applied, this method lets the end effector reach the goal coordinate in apparently less steps, it is important to consider that the steps changes if the admissible error is smaller. It is also important to consider the motion of the joints according to the trajectory, Fig. 3 shows the joint position curves using the three methods, as expected trapezoidal and polynomial curves are very similar, but screw linear interpolation is different than the other methods.

The figure shown before Fig. 4 shows the velocity of the end effector, this figure shows one of the main differences, this figure shows linearity on the velocity of the trapezoidal interpolation which are basically trapezoids forms on the curve, this is the reason for the name of the method, meanwhile the polynomial interpolation is characterized by a smooth curve. This figure definitely demonstrate the main difference between both methods, the polynomial interpolation has smooth curves, meanwhile the trapezoidal interpolation does not, finally screw linear interpolation has very aggressive changes on velocity, that means that the acceleration it needs to reach is pretty high in comparison to the other methods, this is the reason why this method lets the end effector be very close to the

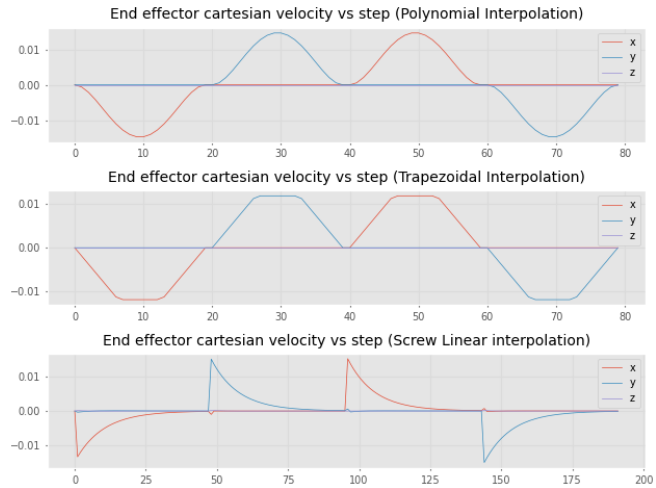



Fig. 4. End Effector velocity vs steps

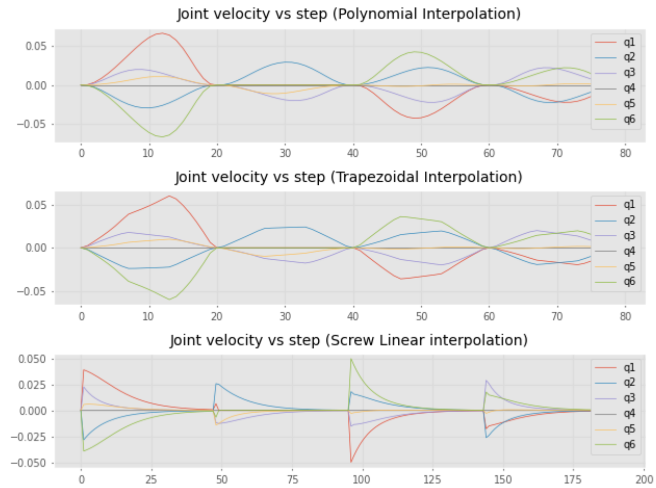



Fig. 5. Joint velocity vs step

goal coordinate.Very similar results were gotten with the joint velocity curves, using polynomial interpolation shows smoothness on trajectory, meanwhile trapezoidal interpolation shows continuity but not smoothness on trajectory, finally screw linear interpolation shows some discontinuities and several peaks on the domain, which means that the joints must reach a high velocity in a very short interval of time.

Finally, the acceleration curve shows the discontinuities on trapezoidal interpolation, which is basically the reason why the velocity curve has a linear behaviour with a constant slope and a constant value on some intervals, the main difference with the polynomial is the smoothness of both curves, this aspect is evident on Fig. 6, finally as expected and mentioned before, the acceleration curve has few peaks, this explains the behaviour of the position of the end effector, the acceleration curves of joints through the path is shown on Fig. 7, this shows the main differences between the three interpolation methods, probably one of the most

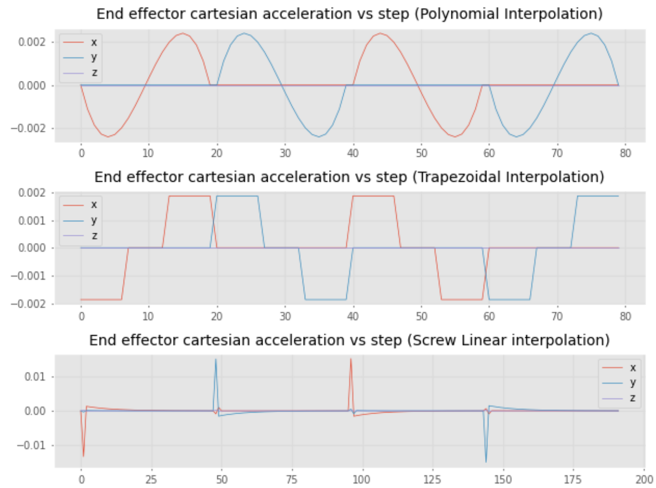



Fig. 6. End Effector acceleration vs steps

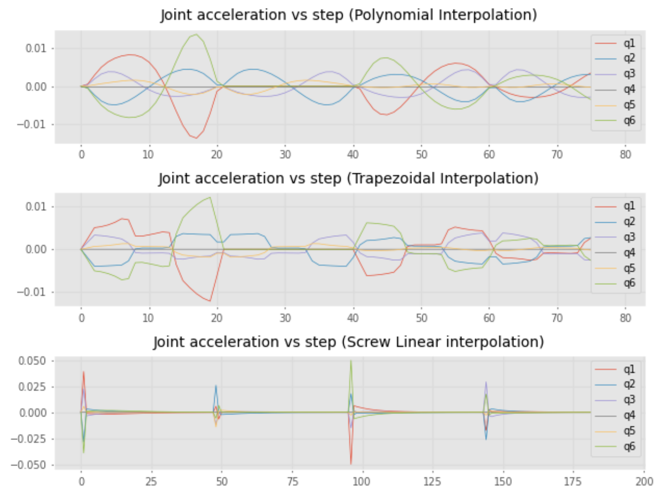



Fig. 7. Joint acceleration vs step

## 4. CONCLUSION

In conclusion, the main objective of the project was met due to the use of the Puma 560 manipulator with six-degrees-of-freedom to describe a square path using two different interpolation methods base on the polynomial and trapezoidal path. The steps to solve the problem is summarized on the following algorithm, first it is required to define the task space way-points, then find the joint space trajectory with the interpolation method, solve inverse kinematics and finally prepare the joint position commands.

The algorithm mentioned before was applied on the code developed, the project is available on a repository on **GitHub**. The first part, was defining the coordinates to analyze as mentioned before, then we used inverse kinematics to find the required displacement by interpolating with the step defined, then by iterating on every step by solving inverse kinematics we will find the displacement of the joints which are finally our solution to apply on the robot.

As a result, we obtained different figures to analyze the nature of the curves using trapezoidal and polynomial interpolation, the results are consistent with what was expected, to sum up, the polynomial interpolation is smooth meanwhile the trapezoidal solution has some discontinuities within the domain.

## REFERENCES


Castro, S. (2019). Trajectory planning for robot manipulatorss. *Mathworks*.

Olivares Fong, J.A. (2016). Interfaz de usuario para el robot puma 560 de unimation-edición única.

Piltan, F., Yarmahmoudi, M.H., Shamsodini, M., Mazlomian, E., and Hosainpour, A. (2012). Puma-560 robot manipulator position computed torque control methods using matlab/simulink and their integration into graduate nonlinear control and matlab courses. *International Journal of Robotics and Automation*, 3(3), 167–191.

Sarker, A., Sinha, A., and Chakraborty, N. (2020). On screw linear interpolation for point-to-point path planning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9480–9487. IEEE.

Simó Villanueva, M. (2018). *Modelado y simulación dinámica del robot industrial Staübli Unimation PUMA 560*. Ph.D. thesis, Universitat Politècnica de València.

Zha, X. (2002). Optimal pose trajectory planning for robot manipulators. *Mechanism and Machine Theory*, 37(10), 1063–1086.