SI4

Inverse Kinematics for an Igus Mover5 robot

By Dimitar Dyulgerov

Date: 15 Jun 2023

Abstract
The purpose of this report is to describe the process of developing a basic inverse kinematics algorithm, suited to Igus Mover5 robot arms.

Table of contents

Introduction

One of the main parts of the project is getting the robot to move to a position received from the outside, be that a simulation or hardware. Therefore, the PLC controlling the robot has to know how much to rotate each joint to get to the required position. The position is taken in as cartesian values. Converting this position to degrees of rotation for the joints of the robot is done via inverse kinematics.

Procedure

First, the pickup area has to be defined. The length (Z) and depth (X) are the most important parts. In the project, a 600x600 mm pickup area is used. Figure 1.1 shows a possible scenario. It is worth mentioning that the robot arm has to start in its true 0 position. All calculations are relative to that position. Movements towards the base/down/left are in negative degrees, the opposite is true for the rest.

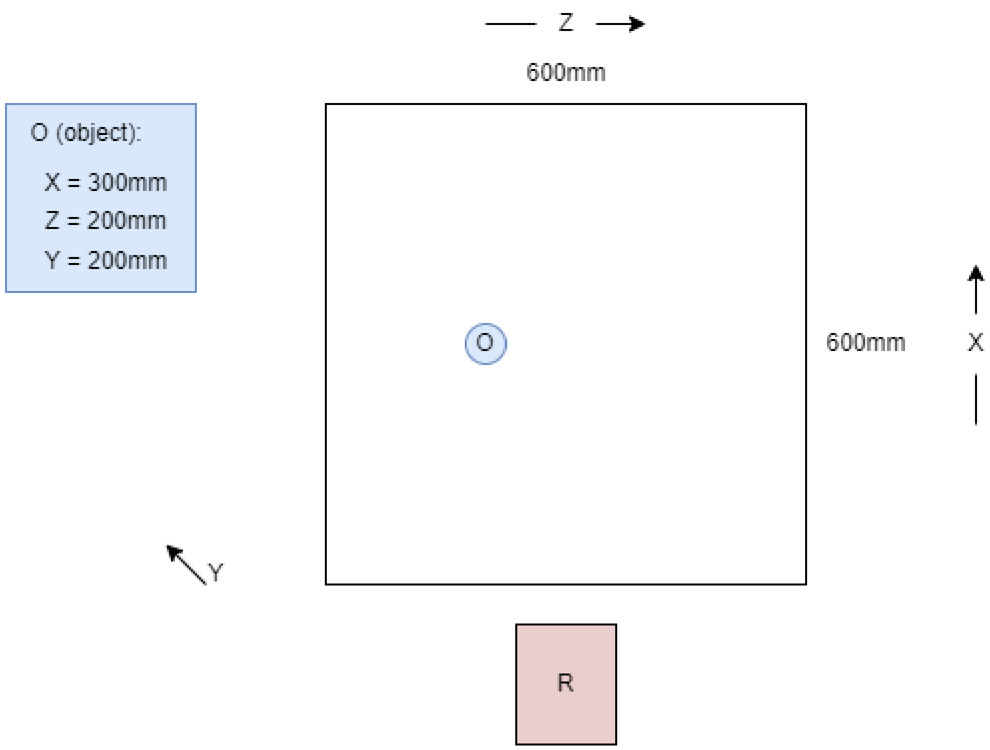

Figure 1.1

Step 1

Calculate the angle that the base of the robot has to move. See Figure 1.2

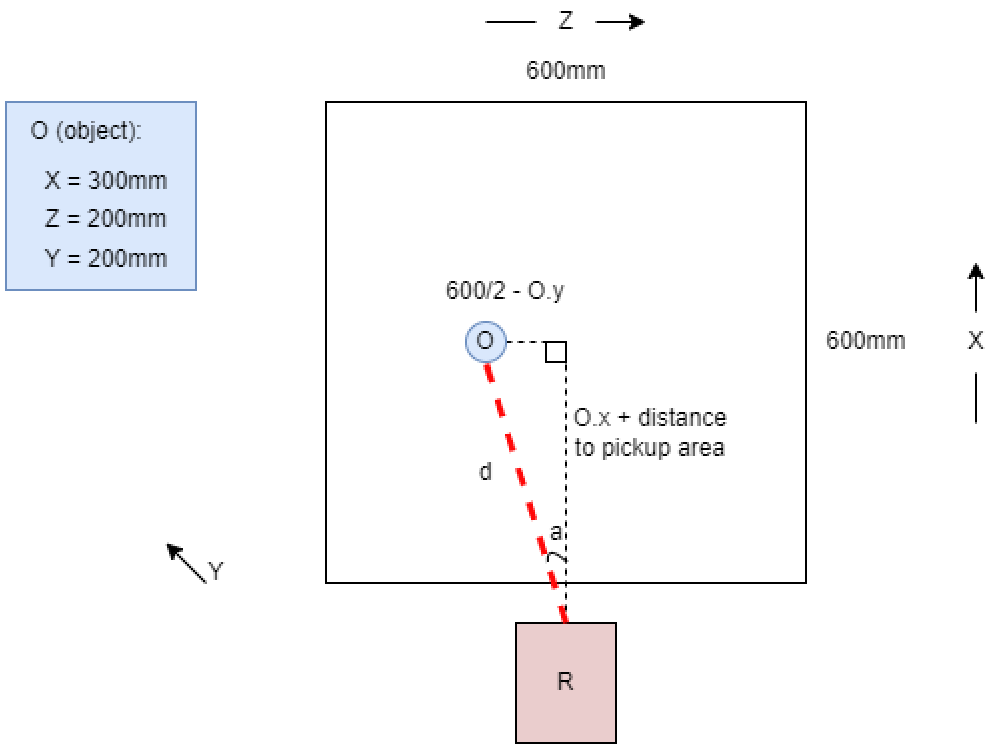

Figure 1.2

It is worth noting that, the way the algorithm is designed, the robot is always in the centre of the pickup area's Z. Movements to the left are in negative degrees, movements to the right are in positive degrees. The calculations are the same for both sides, but we must ensure we provide the proper value at the end of the algorithm. All trigonometry is done with positive values, however. The result is simply multiplied by -1.0 if it is in the left half.

As can be seen in Figure 1.2, we get a right triangle with 2 known sides - the 2 shoulders - therefore we can solve any other part.

We are going to need the hypotenuse and the angle marked ‘a’. a is the angle that the base has to rotate to. (multiplied by -1.0, since it is to the left). That angle can be found by using the inverse tangent of a: arctan(opposite side / adjacent). The distance between the robot and the object O (the hypotenuse, marked ‘d’) can be found via the Pythagorean theorem. Now we have everything that we need from this triangle. We move on to the robot arm joints

Note: distance to pickup area is the distance between the base joint of the robot and the beginning of the conveyor belt.

Next, we need to calculate the angles of the actual arms. Figure 1.3 shows the next step

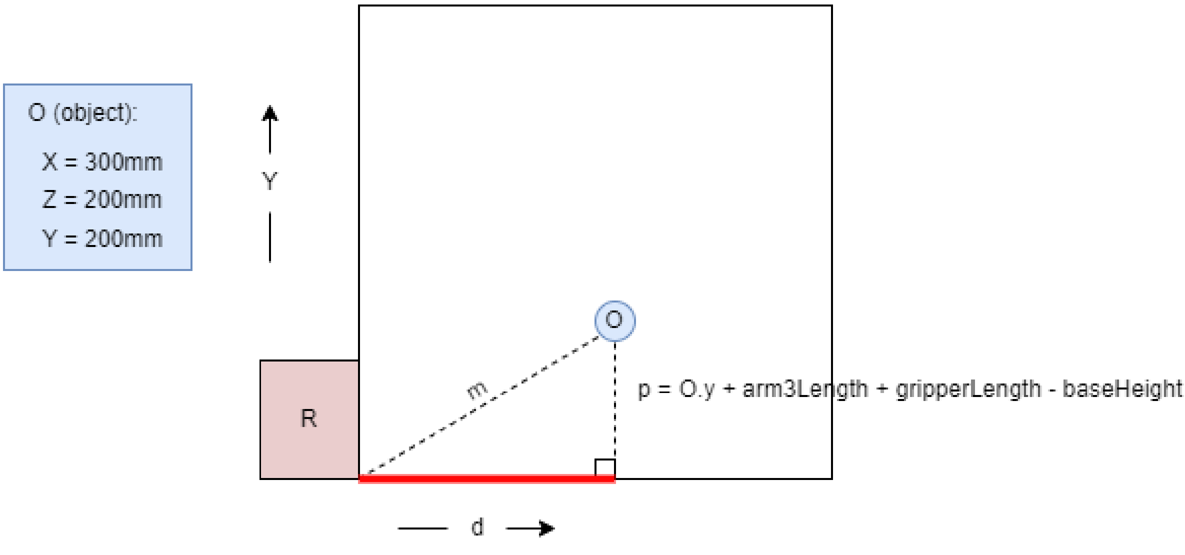

Figure 1.3

Step 2

Since we are doing 3d calculations on a 2d surface, we can only show 2 dimensions at a time. Whereas figures 1.1 and 1.2 show the X and Z planes (depth and length), Figure 1.3 shows side d as one plane (notice the same red colour) and y as the second plane. It is as if we were to put a piece of paper perpendicularly to Figure 1.2 and align it with side d so that they are parallel and the piece of paper is right on top of side d. That way, we get Figure 1.3.

The object height is known, so are the measurements of the robot arm. The arm 3 length and the gripper length are added to the height, since we need arm3 (and consequently, the gripper) to be perpendicular to the ground plane. The height of the base is subtracted, as the actual first joint of the arm is elevated and not on the same height as the conveyor belt. Having that information, we can calculate p. We know d from the previous step. We have a right triangle once more, therefore we can solve it in much the same way. We need the same 2 parts as we did in Step 1.
We get the angle between m and d via the arctan and we get the hypotenuse m via the Pythogorean theorem.

Step 3

Now we can calculate the angles of the arms. Figure 1.4 shows the basic idea.

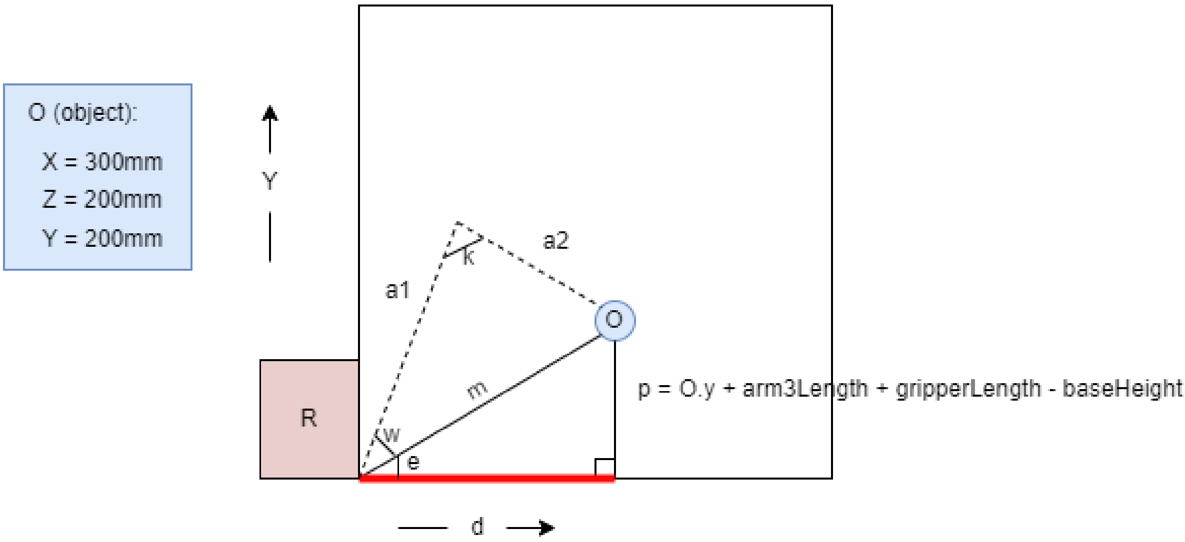

Figure 1.4

In order to find the actual angles of the arms, we need to solve one more triangle. a1 and a2 are the lengths of arm 1 and arm 2 respectively. We found m and e in Step 2. Having 3 sides, we can solve for any of the angles. We need angle w and angle k. We can get them by using the cosine formula. After we know the cos of the angle, we can simply use the inverse cos (arccos) to find its value in radians, which we can easily convert to degrees (radValue*180/pi). Finally, we sum e and w to get arm 1's angle and we have arm 2's angle - k.

Step 4

Lastly, we need to find the angle of arm 3 such that arm 3 is perpendicular to the ground plane. We can do that by summing arm1's and arm2's degrees of rotation (relative to their 0s).

Conclusion

In conclusion, the inverse kinematics algorithm developed for the mover5 robot is designed with the project's requirements and details in mind. The calculations result in proper results when plugged into the actual robot arm. The testing was done by manual measurements whose accuracy may vary. The arm moves to the proper position with a negligible margin of error, however. Therefore, it is concluded that it is sufficient. For the purposes of this project

References

[Basic Inverse kinematics tutorial video]