Human simulated robotics

Zhongyu Zhang - [zhangz28@rpi.edu](mailto:zhangz28@rpi.edu)

Luobin Ni - [nil2@rpi.edu](mailto:nil2@rpi.edu)

Jinhan Ren - [renj2@rpi.edu](mailto:renj2@rpi.edu)

Tiancheng Bao - [baot@rpi.edu](mailto:baot@rpi.edu)

December 09, 2022

# 1 Abstract

The overall goal of the project is to program the Dofbot to be manipulated by different speeds. The main frame is to build a user interface to allow users to choose different models, whether the speed is manipulated or automatically changed by seeing different colors. Therefore, it is able to make the whole robot more humanoid and simulate people’s emotions. The forward kinematics uses rotation matrices to calculate the final position. The inverse kinematics uses subproblems to get angles given by the rotation matrix and end effector position. The path following generates a path that allows the robotic arms to move along the way. The feedback control is based on the path following with PID control to decrease the errors. The color recognition will have slight differences on different Dofbot, which causes some slow or unprecise recognition. A detailed discussion of algorithms, results, and conclusions is presented.

[**1 Abstract**](#_r3swrr9bcdn7) **1**

[**2 Background and Motivations**](#_sd5h9k3663fh) **3**

[**3 Project Objectives**](#_ffvjpprdany1) **4**

[**4 Methodology**](#_w12m5osxlx7w) **5**

[4.1 Forward Kinematics](#_m940s87s61kb) 5

[4.2 Inverse Kinematics](#_c3nimjqh8f0v) 7

[4.3 Path Following](#_b4o8bo75sxqw) 8

[4.4 Feedback Control](#_c3buxbru30mo) 10

[4.5 New Skills](#_4funitqlpam5) 12

[**5 Results**](#_d2cxvd6yislu) **13**

[5.1 Forward Kinematics](#_mr9lhemsyek5) 13

[5.2 Inverse Kinematics](#_t0uv4op20ahg) 15

[5.3 Path Following](#_nj38tsw6hoaq) 16

[5.4 Feedback Control](#_lt0okz30cvc3) 17

[5.5 New task](#_p79mxe80tjoh) 19

[**6 Conclusion**](#_v1430fcbf744) **20**

[**7 Future Work**](#_91u417efddqs) **22**

[**8 Reference**](#_rc5kktt9v0bd) **23**

[**Appendix A: Professional Considerations**](#_fyr1fj7lwwum) **24**

[**Appendix B: Division of work**](#_inoh8py8mx79) **26**

[**Appendix C: Link to GitHub Repository**](#_34mph8ay3jip) **27**

# 

# 2 Background and Motivations

Because humanoid robotics is increasingly popular nowadays, people want to build more real robots worldwide. The main difference between humans and robots is that humans have emotions but robots do not. The emotions can be represented by different speeds of movement. For example, people may punch or slap when they are angry, and the speed of action at this time is very fast. However, people in peace or relaxed doing the same action may represent touch. Therefore, the same action can have totally different meanings at different speeds. The project is to make the robotic act at a different speed when it sees different colors.

In this project, Dofbot acts as the control object of the course. Dofbot is a robot that simulates a human arm with five degrees of freedom. Dofbot also has a mechanical claw that can grab objects. The team's goal is to be able to control the Dofbot and simulate human emotions in it to control the arm's joints at different speeds.

# 3 Project Objectives

The fundamental goal of this project is to achieve basic tasks: forward kinematics, inverse kinematics, path following, and feedback control. For forward kinematics, the goal is to generate numbers for angles to make each joint reach desired angles. For inverse kinematics, the goal is that given the rotation matrix and end effector position, the robotic arm can return each joint angle. For path following, the goal is to create a path between two points and make the robotic arm move along the path. For feedback control, the goal is to add PID (proportional–integral–derivative) control based on Path Following to decrease the error. The goal of New Skills is to create a user interface that can adjust each joint speed.

# 4 Methodology

## 4.1 Forward Kinematics

Forward kinematics is a transformation matrix to calculate the relationship

between position and orientation (pose) of task (end effector) frame and joint variables. (Piltan et al., 2015, #) With forward kinematics, it is easier to build a model to simulate and decrease potential errors.

The Dofbot is a five-degree-of-freedom robotic arm. The free-body diagram is listed below.

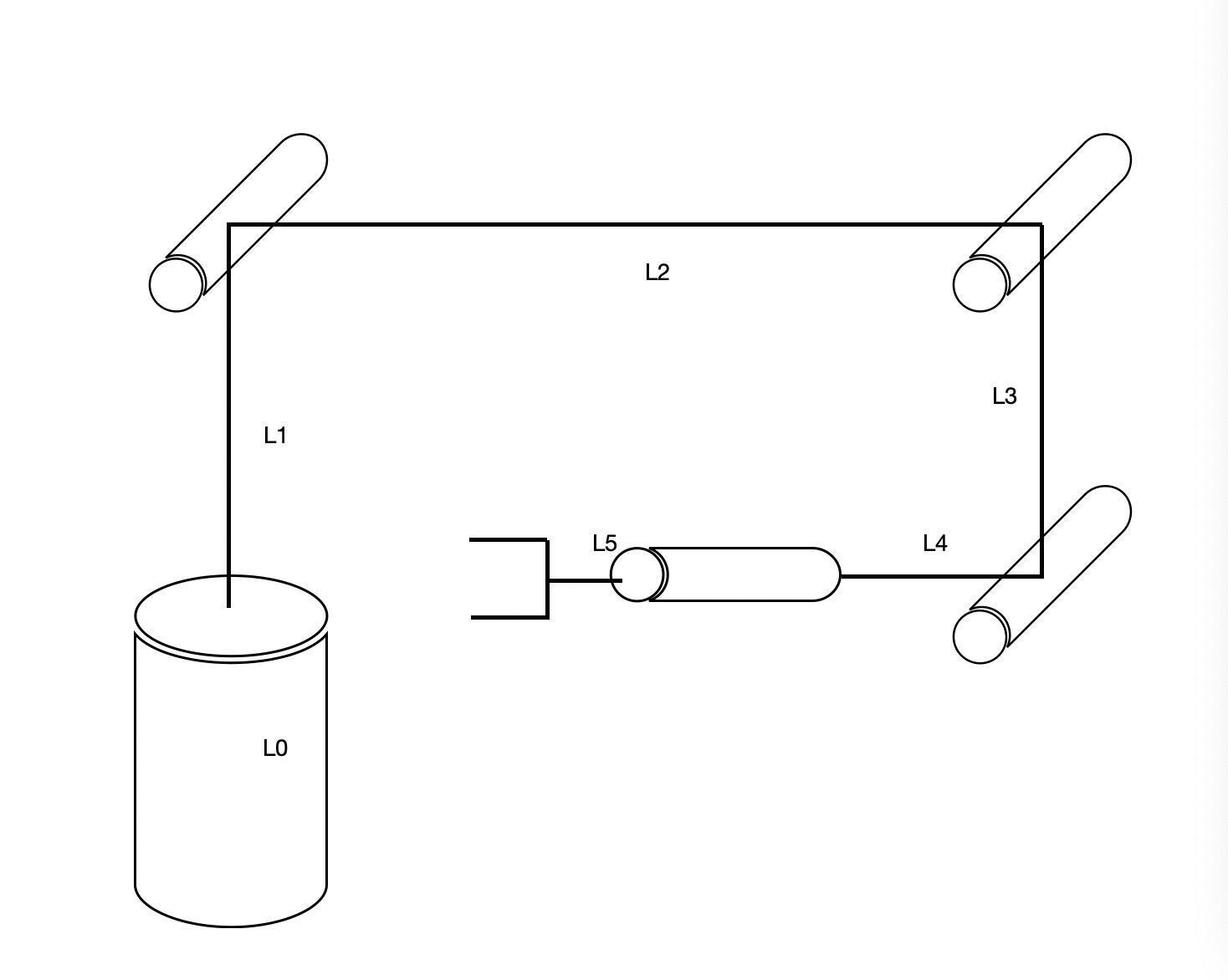
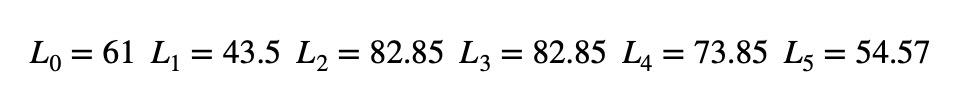
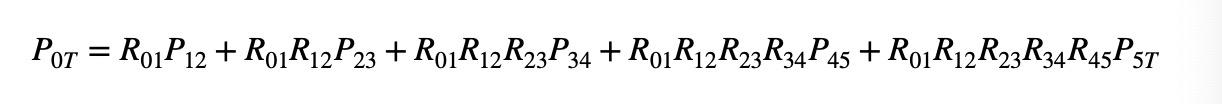


Figure 1: Free-Body Diagram of Dofbot

The number below is the length of each joint in millimeters.

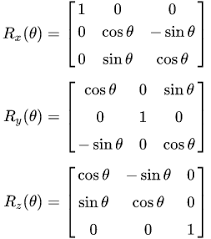


Equation 1: Length of Each Joint

The equation below describes the end effector of the Dofbot robotic arm. The equation here assumes that each joint is independent from each other.

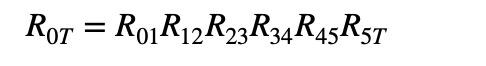
Equation 2: Position Equation of The End Effector

The following matrices are rotation matrices around the x, y, and z-axis. With the basic three rotation matrices, it is able to calculate the rotation matrices for each step.



Equation 3: Rotation Matrix around X, Y, and Z Axis

Based on the rotation equations, the overall rotation matrix can be calculated by multiplying each rotation matrix together.



Equation 4: Rotation Matrix of The End Effector

In the coding part, the Dofbot can be set by entering each joint’s length and maximum angle rotation. Then the end effector matrix can be calculated easily. With the desired position, it is able to calculate the error between the desired position and the real position.



Equation 5: Equation of Error

## 4.2 Inverse Kinematics

Inverse Kinematics can be used to calculate the servo angles for a certain input path. Since Dofbot has only 5 degrees-of-freedom, the Jacobian-based inverse kinematics method is used in this project.

The Jacobian-based inverse kinematics method uses the least square problem to reduce the error of desired angles and real angles of each iteration. The update law equation to decrease the cost is the following equation 6.

Equation 6: Cost decreasing function

In the coding part, the algorithm is designed by python, requiring the desired path as the input, desired angles as a reference to update the error decrease function, also setting the error tolerance to limit the iteration, then get the real output angles of each joint. Besides, the manual measurement between the real final position of Dofbot and desired path by the ruler is also supposed to prove the accuracy of the test.

## 4.3 Path Following

For the path following part, the robotic arm is going to follow a joint space path, q\_desired(λ). To facilitate measurement and observation, the path will be set as a straight line in space. The line is from (0.0558, 0.2058, 0.1464) to (0.2058, 0.1188, 0.1464). (Units: Meters)

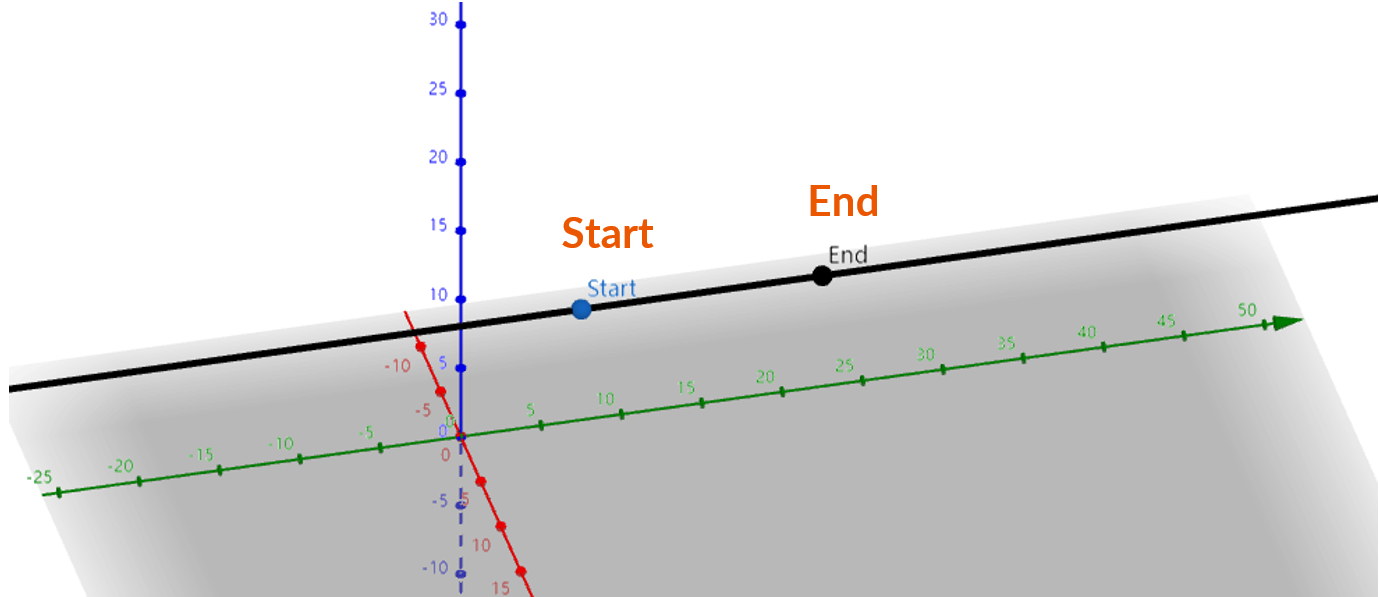


Figure 2: Schematic diagram for desired joint space path

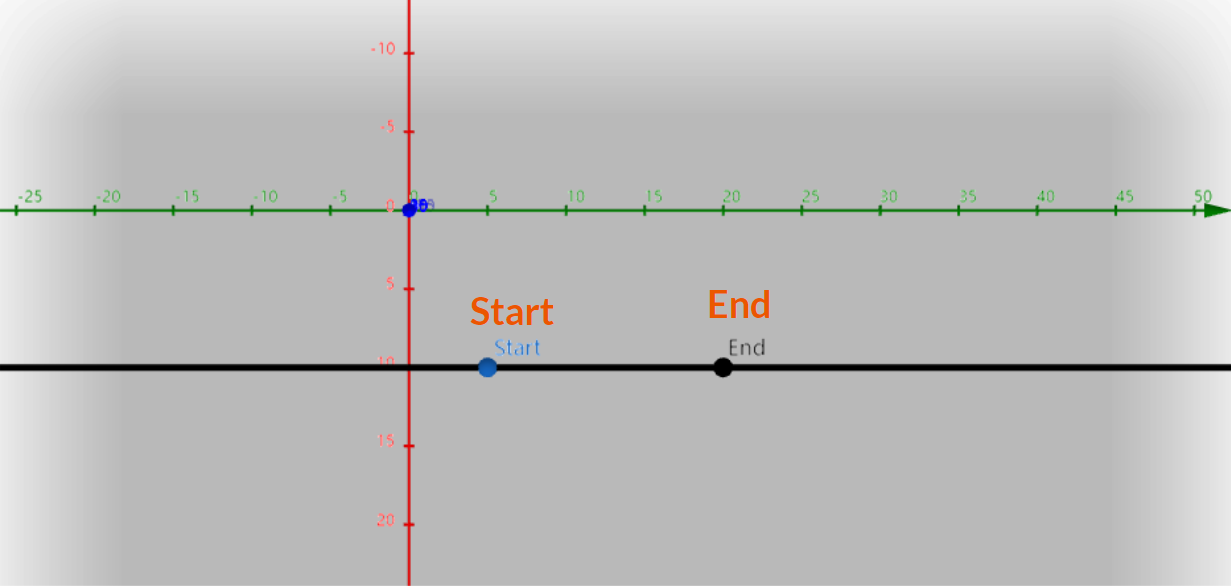


Figure 3: Top view of desired joint space path

Figure 3 shows the position of the desired path in space. The principle of making the robotic arm follow the desired joint space path is to set several equidistant points on the desired joint space path. Then, using inverse kinematics algorithm to control the robotic arm to reach the set points sequentially. The actual movement in space tends to be closer to the desired joint space path by increasing the number of set points.

The forward kinematics algorithm is used to calculate the error between the actual joint space path and desired joint space path. After commanding the robotic arm to follow the desired joint space path, the forward kinematics is used to calculate the end effector position by inputting the angle of each servo. The calculated position is the actual end effector position. The error is calculated by the following equation.

Equation 7: Path following error function

A different number of equidistant points will be set on the space path. The error is expected to decrease as the number of equidistant points set on the spatial path increases.

## 4.4 Feedback Control

In the feedback control section, the robotic arm follows the same space path as in the path following section. And a number of points are set on the desired space path. Then, the PID control is introduced as the feedback control algorithm to reduce the error of desired space path and actual space path. PID control stands for Proportional-Integral-Derivative control. In 1922, Minorsky first analyzed and discussed the PID controller and used the PID controller for automatic ship steering (Minorsky, 1922). The PID control algorithm is one of the most widely used algorithms in the industry to automatically correct the control system accurately and quickly in closed-loop system control.

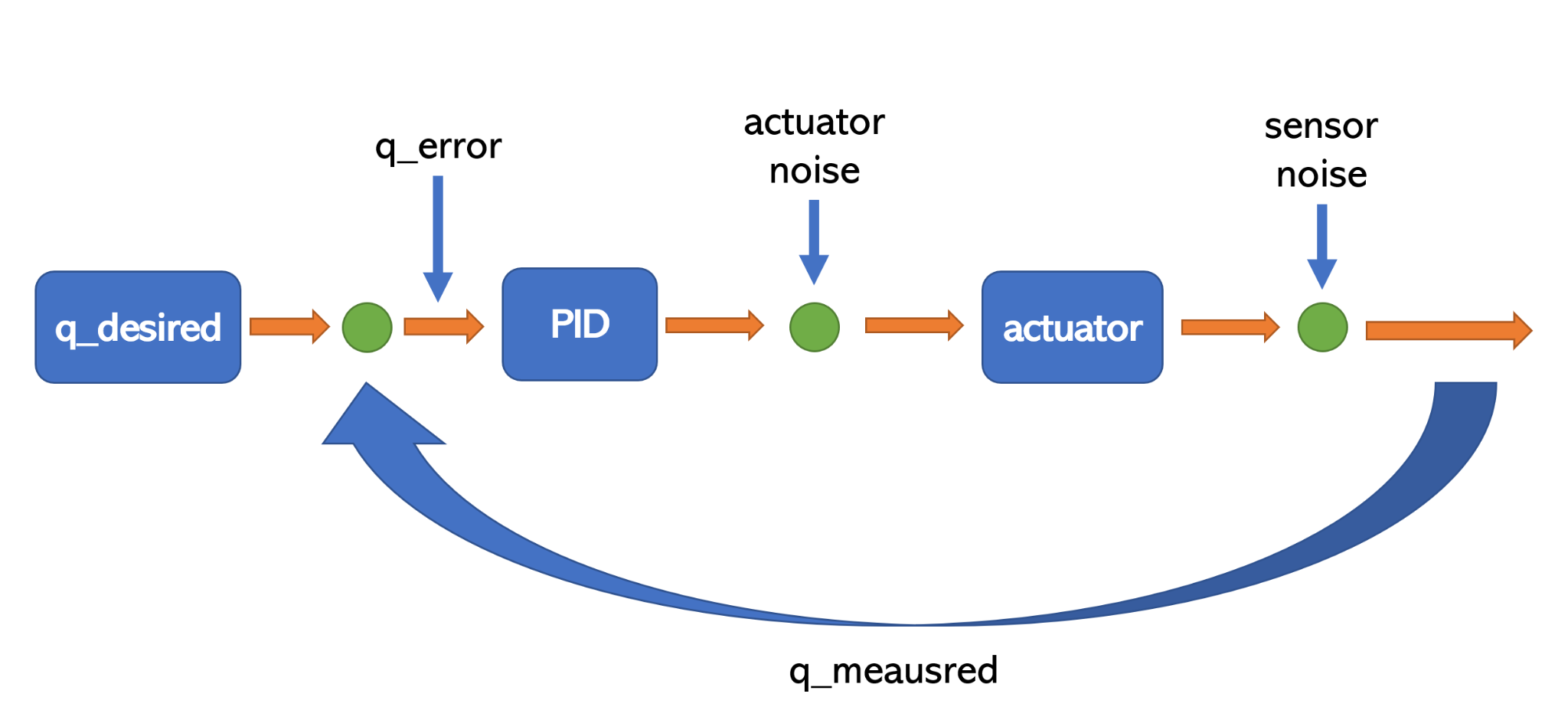
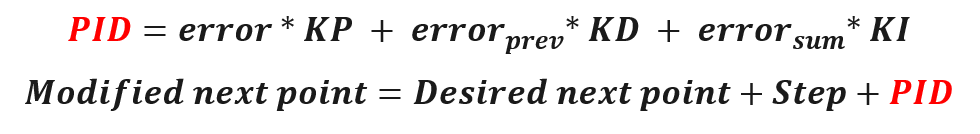


Figure 4: Logic diagram for PID control

The Figure 4 shows the basic logic for PID control. Throughout the closed loop system, the system detects the error from outputs. Then, the system calculates the proportional, integral, and derivative responses to get the value for the next step.

In 2019, Bârsan introduced the position-based PID control (Bârsan, 2019). However, the position-based PID control has never been used for Dofbot. To test whether position-based PID control works on Dofbot, the position-based PID control is decided to be used in Dofbot to reduce the error in path following.

To apply position-based PID control, the error between the desired position and the actual position for each point on the line is calculated by using the same method illustrated in the Path Following section. Then, the position for the next point is modified by adding the PID parameter.



Equation 8: Pseudocode for PID algorithm

Equation 8 shows how the PID control is applied. The error is the current error between the desired position and the actual position. The error\_prev is the error calculated from the previous loop. The error\_sum is the cumulative error.

The KP, KD and KI are tried out by increase and decrease the values. As a result, KP = 0.02, KI = 0.005 and KD = 0.01 can minimize the error between the result with PID control and without PID control.

After the end effector reached the end of the desired space path, the error is calculated by the following equation.

Equation 9: Feedback control error function

To compare the error between the result with PID control and without PID control, the following equation is used.

Equation 10: Error between results with PID and without PID

## 4.5 New Skills

This new task is designed to let robots simulate the human arm. To make an arm move like a human, the robot is designed to control its joints based on its “emotion”. To do so, the robot uses its camera to detect color blocks and behave accordingly. This requires a robot to calibrate the camera to detect colors.

The method used in the DOFBOT to calibrate color is based on changing HSV values, hue, saturation, and value. Hue defines the color type, saturation defines how faded the color appears, and Value defines the brightness of the color (Li, 2015). Since the robot is only interested in color type, the code only looks at the hue. For each different color, it has a different range of hues. As a result, small experiments are set up to determine the hue for different colors with a test code modified by a sample code from DOFBOT. During each task, 4 different color blocks are brought to the front of the camera to see if the robot can print the correct color name.

The code for measuring hues begins with getting images from the camera. As the robot reads the images, the code resizes the picture to 640\* 480 and obtains HSV values of the center rectangle (80\*80). From the HSV values, the code calculates the average values and compares them with the regular hue (found online). The error is the difference between the calculated value and the regular value.

# 5 Results

## 5.1 Forward Kinematics

Result for the FK is listed below. It generates random angles for q1 to q5 for 5 times.

| Times | q1(degree) | q2(degree) | q3(degree) | q4(degree) | q5(degree) |
| --- | --- | --- | --- | --- | --- |
| 1 | 87 | 64 | 67 | 57 | 63 |
| 2 | 51 | 53 | 89 | 59 | 45 |
| 3 | 60 | 89 | 78 | 87 | 62 |
| 4 | 87 | 85 | 48 | 37 | 52 |
| 5 | 46 | 69 | 67 | 77 | 37 |

Table 1: Angles for q1 to q5

| Times | Measured(centimeter) | Desired(centimeter) | Error(percentage) |
| --- | --- | --- | --- |
| 1 | [1.5 21.0 26.0] | [1.18 22.57 25.12] | 0.99% |
| 2 | [15.0 16.5 29] | [13.89 17.16 28.20] | 2.15% |
| 3 | [3.5 6.0 40.5] | [2.77 4.80 39.15] | 3.92% |
| 4 | [1.5 19.0 23.5] | [1.02 19.40 22.12] | 2.78% |
| 5 | [13.9 14.0 33.0] | [13.54 14.02 31.14] | 4.66% |

Table 2: Table for Measured and Desired Positions, and Errors

In Table 2, the error is within 5% in centimeters. The maximum is 4.66% and the minimum is 0.99%. Since the end effector is in a 3D place, it is hard to measure an accurate number. Moreover, this ruler is accurate only to millimeters. This makes it impossible to get more precise data.

Overall, the error is within the expectation. With a more accurate measurement tool, it is possible to get a more precise position and error. Therefore, a better improvement will be applied in the following work.

## 5.2 Inverse Kinematics

The value of the desired path is based on previous homework. The input values in the initial process show below.

| Rotation desired () | [[ -0.75 , -0.1047 , -0.6531 ],  [ -0.433 , 0.8241 , 0.3652 ],  [ 0.5 , 0.5567 , -0.6634 ]] |
| --- | --- |
| Path desired () |  |
| Angles in 0 configuration () |  |
| Alpha () | 0.3 |
| Tolerance (tol) |  |
| Maximum iteration () | 200 |

Table 3: Input Parameters in IK

In Table 3, the alpha is the value in cost decreasing function. Tolerance is set to limit the iteration to a shorter time. After the iteration finished, the result includes real path position and real angle.

| End effector position () |  |
| --- | --- |
| Real angles () |  |

Table 4: Outputs in IK

For result analysis, in this test, the error of the desired position and end effector position is around 0.0001m, which is reasonable and acceptable. Ruler measurement is also involved to double check the test and shows an accurate result.

## 5.3 Path Following

The following table shows errors with different numbers of points set on the desired space path. (Units: Meters).

| Number of Points | Errors (x-axis) | Errors (y-axis) | Errors (z-axis) |
| --- | --- | --- | --- |
| 10 | 0.0063 | 0.0012 | -0.0219 |
| 20 | 0.0063 | 0.0013 | -0.0215 |
| 30 | 0.0062 | 0.0011 | -0.0214 |
| 40 | 0.0062 | 0.0010 | -0.0211 |
| 50 | 0.0062 | 0.0011 | -0.0209 |

Table 5: Path Following Errors for x, y, z axes

The table 5 shows the error decrease with the increasing number of points. Although the reduction is not very significant, the error is acceptable. The error for the x-axis is less than 0.6 centimeters. The error for the y-axis is less than 0.1 centimeters. The error for the z-axis is less than 2 centimeters. The z-axis has significantly more errors than the other axes. The errors can be further reduced by applying feedback control.

## 5.4 Feedback Control

The following table shows both errors with PID control and without PID control for different number of points set on the desired space path. (Units: Meters).

|  | x-axis | | y-axis | | z-axis | |
| --- | --- | --- | --- | --- | --- | --- |
| Number of Points | Errors  Without PID | Errors  With  PID | Errors  Without PID | Errors  With  PID | Errors Without PID | Errors  With  PID |
| 10 | 0.0063 | 0.0060 | 0.0012 | 0.0003 | -0.0219 | -0.0037 |
| 20 | 0.0063 | 0.0060 | 0.0013 | 0.0003 | -0.0215 | -0.0035 |
| 30 | 0.0062 | 0.0058 | 0.0011 | 0.0003 | -0.0214 | -0.0036 |
| 40 | 0.0062 | 0.0057 | 0.0010 | 0.0003 | -0.0211 | -0.0036 |
| 50 | 0.0062 | 0.0057 | 0.0011 | 0.0003 | -0.0209 | -0.0035 |

Table 6: PID control vs. without PID control

The following table shows errors between an actual final position with PID control and without PID control. (Units: Meters).

| Number of Points | x-axis | y-axis | z-axis |
| --- | --- | --- | --- |
| 10 | 0.0003 | 0.0009 | 0.0182 |
| 20 | 0.0003 | 0.0010 | 0.018 |
| 30 | 0.0004 | 0.0008 | 0.018 |
| 40 | 0.0005 | 0.0007 | 0.0178 |
| 50 | 0.0005 | 0.0003 | 0.0174 |
| Average Error | 0.0004 | 0.00074 | 0.01788 |

Table 7: Errors between an actual final position with PID vs. without PID

The table 7 shows the final position of end effector becomes closer to the desired position. The error with PID control was reduced a lot compared to the error without reduced control.

## 

## 5.5 New task

The following tables show the hue we measured with DOFBOT and the data (regular) we found online. The error is equal to the measured hue minus the regular hue. Based on the table, the measured hue has a small difference from the regular hue but it is acceptable since the block used in the tasks is not standard. In order to improve the measured result, color blocks with standard color and more precise cameras are needed for robots.

| Color | Red(degree) | Yellow(degree) | Green(degree) | Blue(degree) |
| --- | --- | --- | --- | --- |
| Measured Hue | 10 | 74 | 100 | 259 |
| Regular Hue | 0 | 60 | 120 | 240 |
| Error | 10 | 14 | -20 | 19 |

Table 8: Hue Range for Different Colors

# 6 Conclusion

Robot is a sensitive and accurate device for multiple functions and motions. This project uses the forward kinematic method and the inverse kinematic method to achieve different kinds of motion, which is roughly accurate, but still makes small errors. In order to enhance the maneuverability and practicability of the robot, path planning is designed based on the realization of motion control. Path planning is designed by dividing the path into a series of points and controlling the robot to follow the interval points to follow a continuous path. However, the experimental results show that direct path planning will cause errors in the results. In order to reduce the experimental error, PID control is considered to be an available way. After the specific PID controller value is determined through experiments and added to the path planning, the robot's movement error on the specific path is greatly reduced.

The versatility of the robot provides the basis for the new task in the later stage of the experiment. Since the basic motion functions have been established, the robot control system lacks specific interfaces for motion generation, such as speed adjustment of different joints. Therefore, the new task part introduces two functions of speed adjustment for the robot. The first function is manual speed adjustment. A new speed control interface is designed to enable users to freely input the desired speeds for different joints. The second function is the automatic speed control system. The camera is introduced into the system to identify different color blocks as commands to control the robot to move at different speeds. From the results, the experimental results of the new task part are accurate, which makes the speed controllable easily. In addition, the interface is friendly to users, and the color recognition of the camera is very sensitive by doing the color calibration.

In general, this project is of far-reaching meaning. During the project, the team members practiced the knowledge that was taught in class, flexibly applied the knowledge to the robot motion function, and successfully added some innovations. The skill What is learned through the project can be used in future research and work, which will greatly improve students' abilities.

# 7 Future Work

The goal of this project in the future is to improve the existing user interface. The robot can recognize not only the difference between color blocks but also various shapes and movements. Since the robot's goal is to capture the environment around it to express its emotions, enriching the robot's emotions will also be one of the future works.

For graphics and action recognition, developers can use OpenCV to recognize more complex situations, such as gesture recognition. This will improve the robot's response to the real world.

With the rapid development of AI, this project can use AI as an auxiliary condition for emotional learning. With the help of AI, robots will be able to make more accurate judgments and more intelligent choices.

# 

# 8 Reference

A. Bârsan, “Position control of a mobile robot through PID controller,” Acta Universitatis Cibiniensis. Technical Series, vol. 71, no. 1, pp. 14–20, 2019.

Li, D. (2015). Encyclopedia of Microfluidics and nanofluidics. Springer.

N. Minorsky, “Directional stability of automatically steered bodies.” J. Amer. SOC. of Naval Engineers, pp. 280-309, v. 34, 1922

Piltan, F., Taghizadegan, A., & Sulaiman, N. (2015, November). Modeling and Control of Four Degrees of Freedom Surgical Robot Manipulator Using MATLAB/SIMULINK. *International Journal of Hybrid Information Technology*, *8*(11), 47-78.10.14257/ijhit.2015.8.11.05

# 

# Appendix A: Professional Considerations

Since this project is about simulating an artificial intelligence manipulator, the input and output of the angle must be controlled during operation. In this way, the damage caused by each joint of the manipulator exceeding the maximum use limit can be avoided. In the New Task, the robot will autonomously learn color recognition to control the speed of each joint. Speed control must have an upper limit to ensure that the arm does not swing too fast and cause injury to personnel. During debugging, it is necessary to ensure that there is no person or object within the length of the manipulator to avoid unnecessary injury and property loss.

The project can be applied to future human simulations and simulations. Emotional robots could be used to accompany lonely elderly people in nursing homes. This can reduce the work of some staff and increase efficiency and save money. More importantly, it can provide spiritual companionship and care for lonely people, reducing the potential possibility of depression.

The mass use of sentient robots could replace many traditional jobs around the world. It provides round-the-clock companionship and support. This could lead to job losses in some occupations.

The robot has been partially adjusted to be able to send and take part in heavy and boring or dangerous work in the future. This will greatly improve productivity or avoid loss of life.

In conclusion, the future application of this project can help mankind save time and money. Depending on the project, it can provide emotional support or do some work for humans. Although large-scale use of the robot may lead to job losses for some employees, the overall advantages outweigh the disadvantages.

# 

# Appendix B: Division of work

Tiancheng Bao: Coded for Task 1 and Task 2. Designed and took the video. Took part in the data collecting and camera calibration for the whole project. Involved in the design of the new task and debugging for the whole project. Participated in report writing.

Jinhan Ren: Coded for Task 2 and color speed control of the new task. Calibrated the camera and took part in the New task design. Took the video and edited the video. Involved in the discussion and testing for the whole project. Participated in Task2 and conclusion of report writing.

Luobin Ni: Involved in discussing, testing and recording the whole project. Designed the new task and wrote codes for the new task. Modify the new task to let robots simulate human emotional reactions. Participated in writing the final report.

Zhongyu Zhang: Worte all the code for Task 3 and Task 4. Tested the code for Task 1, 2, 3, 4 and the new task. Wrote the section 4.3, 4.4, 5.3, 5.4 for the final report. Recorded video clips. Involved in the discussion for the whole project.

# 

# Appendix C: Link to GitHub Repository

<https://github.com/Salad5413/Robotics_Team16_2022>