Home-Use Automatic Watering Robot Kinematic Analysis

Kehan Zhong

Faculty of Science and Engineering, University of Nottingham, Ningbo, China

ssykz8@nottingham.edu.cn

Abstract. Watering potted plants on time and correctly can be difficult for the city staff and students. To solve that issue, employing a robot to do the watering job may be a potential solution to decrease the mortality rate of potted plants. This paper describes a robot designed for watering potted plants at home. The robot is capable of determining the location of potted plants and deciding whether to water or not through the soil moisture sensor. After receiving the signal of watering, the watering process is mainly completed by the robot arm. In the method part, the modified forward kinematic calculation method and iterative approximation inverse kinematic calculation method are informed, and the feasibility of that robot arm is discussed in the result part. Through calculation, the robot arm's forward kinematic is represented in a transformation matrix and the inverse kinematic is calculated through MATLAB to prove the correctness of the forward kinematic transformation matrix, which illustrates that this robot has feasibility in kinematics.

Keywords: Watering robot, manipulator kinematic analysis, MATLAB, iterative approximation.

1. Introduction

With the development of human beings, potted plants are increasingly common in citizens' daily lives. It is noticed that keeping some plants at home and workplace benefits not only physically but also mentally. As the early research revealed that human beings whose average age is 21 and no matter the gender, it is easier for them to relax in the company of plants [1]. However, staff and students in the city who are most in need of mental health seldom have enough time to look after their plants, especially watering them in time and watering the plant with a suitable amount of water. This phenomenon also looms in the workplaces of enterprises. This results in numerous potted plants remaining soaked and dry for most of the day. There is no doubt that the life expectancy of plants would be drastically reduced in such a situation and the mortality rate would rise [2]. This would lead to increased expenditure on replacing the plants with new ones, as well as a negative impact on people's emotions. To solve these issues, some may hire extra gardeners or plant plants that are easier to maintain, which is obviously not an efficient solution.

To not only reduce the expenditure of extra cost but also expand the diversity of potted plants to be raised, the automatic home-use watering robot is designed. The prospective use of this robot is aimed at providing a stable and reliable device to automatically take care of potted plants for individuals or companies who have no time to water plants in the morning and afternoon. In fact, the robot watering process is credited with controlling the amount of watering precisely since the robot waters based on the soil moisture level instead of subjective judgment.

In this paper, the feasibility of this robot in the Cartesian coordinate space, where it is expected that rigid links can control the watering device at the end of the robot. Therefore, forward and inverse kinematics analyses are essential to clarify the conversion between Cartesian space and joint space. Forward kinematics is used to derive the position of the end device from a given joint angle. Inverse kinematics is used to figure out the joint angles with data of the position of end device. In the next section, the design overview of the robot is displaced with some simple calculations of working area and the motor decision. Then, the calculation methods of forward and inverse kinematics of the manipulator are introduced in the method section, and the result will be represented in the result section.

2. Design Overview

To tackle the issue, a home-use automatic watering robot is designed. The robot is working on a straight rail with a robot arm on the top. The working process of the robot could be summarized as receiving soil moisture signals, system judging whether to water, moving, completing watering instructions and resetting.

At the beginning, the pots are placed in several blocks of equal size, which are arranged linearly. Next, the soil moisture sensor in the pot sends the signal of the soil moisture data to the control centre of the robot. Then the control centre compares the data obtained with a predetermined minimum value of soil moisture. When the moisture level is lower than the predetermined value, the control centre sends a watering command, and the trolley is instructed to move along the track to the block where the pot is located. The infrared distance measurement sensor on the side of the trolley will determine the distance from the robot to the pot, and the robotic arm will gradually extend to complete the watering process.

The area in which the pots are placed is 1m*0.4m. The distance between the rail and the area is 0.3m because there is a certain area that the robot arm cannot reach. The area for pot to be set is divided into 5 parts with the same width. And one pot could be put in only one area. To calculate the water needed to determine the water tank size, the pot size is supposed to have a diameter of 10 to 20cm and a height of 10-25cm. The water needed for a 25cm-high pot is 0.7 gallons = 2650cm^2one day. The watering process for each pot is quantitative, watering soil moisture from 30% to 60%. Since the division of the area has determined the number of pots to be 5, the water needed could be calculated to be $2650\text{cm}^2 * 30\% * 5 = 3975\text{cm}^2$. So the water tank size is set to be 20*20*15cm. The water pump used Is ANGTE DC23A with a low rate of $72\text{L/h} = 20\ \text{cm}^3/\text{s}$. As a result, the watering time is $2650 * 30/20 = 40\text{s}$.

3. Methods

3.1. Forward Kinematic

To calculate the forward kinematic and inverse kinematic equation, the method called Denavit-Hartenberg (DH) parameter is used. In DH parameter, four parameters, α, a, d, and θ, are usually used to represent the state of motion of the robotic arm to parameterize the kinematics of the robotic arm for analysis. The four parameters denote the link twist, link length, link offset and joint angle respectively [3].

After defining 4 parameters, it is noticed that a is the distance between axis z_0 and axis z_1 along axis x_1. α is the angle between the axis z_0 and z_1, measured in the plane perpendicular to x_1. d is the distance from the origin o to the intersection of the axis $x_1$and z_0, measured along z_0. θ is the angle from $x_0$to x_1 measured in the plane perpendicular to z_1.

The D-H parameter method is broadly divided into the standard D-H parameter method and the modified D-H parameter method. The standard D-H Parameter is the parameter definition method proposed by Denavit and Hartenberg in 1955. For this method, coordinate system fixed relative to the connecting rod chosen is the coordinate system of the latter joint coordinate system. Furthermore, the x-axis direction of the coordinate system is determined by the cross product of the current z-axis and the z-axis of the previous coordinate system. Meanwhile, the parameter variation order of this definition method is θ d α a [4]. As for the modified D-H parameter method, it is a definition method for describing open-loop, tree and closed-loop robots based on the standard D-H parameter method, which was proposed by Khalil, and Kleinfinger in 1986. This definition method selects the former joint coordinate system for the fixed coordinate system respects of the linkage. In terms of the x-axis direction determination, the latter coordinate system z-axis is cross product with the previous coordinate system z-axis to identify the x-axis. With regard to the order of parameter transformation, it transfers into: α a θ d [5].

Table 1. Modified D-H parameter

α	a (mm)	θ	d (mm)
α_1	a_1	θ_1	d_1
α_2	a_2	θ_2	d_2
α_3	a_3	θ_3	d_3
α_4	a_4	θ_4	d_4
α_5	a_5	θ_5	d_5

The four parameters are substituted into a transformation matrix to represent the individual joints of this robotic arm. For standard D-H parameters, the transformation matrix is [5]:

$$^{i-1}T_i = Rot(z_{i-1},\theta_i) \times Trans(z_{i-1},d_i) \times Trans(x_i,a_i) \times Rot(x_i,\alpha_i) \tag{1}$$

$$Rot(z_{i-1},\theta_i) = \begin{bmatrix} cos\theta_i & -sin\theta_i & 0 & 0 \\ sin\theta_i & cos\theta_i & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

$$Trans(z_{i-1},d_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$Trans(x_i,a_i) = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

$$Rot(x_i,\alpha_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\alpha_i & -sin\alpha_i & 0 \\ 0 & sin\alpha_i & cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

Substitute in and the transformation matrix becomes:

$$^{i-1}T_i = \begin{bmatrix} cos\theta_i & -sin\theta_i sin\alpha_i & sin\theta_i sin\alpha_i & a_i cos\theta_i \\ sin\theta_i & cos\theta_i cos\alpha_i & -cos\theta_i cos\alpha_i & a_i sin\theta_i \\ 0 & sin\alpha_i & cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

For modified D-H parameters, the transformation matrix is:

$$^{i-1}T_i = Rot(x_i,\alpha_{i-1}) \times Trans(x_i,a_{i-1}) \times Rot(z,\theta_i) \times Trans(z,d_i) \tag{7}$$

$$^{i-1}T_i = \begin{bmatrix} cos\theta_i & -sin\theta_i & 0 & a_i \\ sin\theta_i cos\alpha_i & cos\theta_i cos\alpha_i & -sin\alpha_i & -sin\alpha_i d_i \\ sin\theta_i sin\alpha_i & cos\theta_i sin\alpha_i & cos\alpha_i & cos\alpha_i d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

The method of analysis chosen in this paper is the modified D-H analysis, so after substituting each parameter, a total of five matrices can be obtained as follows:

$$^{0}T_1 = \begin{bmatrix} cos\theta_1 & -sin\theta_1 & 0 & a_1 \\ sin\theta_1 cos\alpha_1 & cos\theta_1 cos\alpha_1 & -sin\alpha_1 & -sin\alpha_1 d_1 \\ sin\theta_1 sin\alpha_1 & cos\theta_1 sin\alpha_1 & cos\alpha_1 & cos\alpha_1 d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

$$ {}^{1}T_{2} = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 & a_2 \\ sin\theta_2 cos\alpha_2 & cos\theta_2 cos\alpha_2 & -sin\alpha_2 & -sin\alpha_2 d_2 \\ sin\theta_2 sin\alpha_2 & cos\theta_2 sin\alpha_2 & cos\alpha_2 & cos\alpha_2 d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10} $$

$$ {}^{2}T_{3} = \begin{bmatrix} cos\theta_3 & -sin\theta_3 & 0 & a_3 \\ sin\theta_3 cos\alpha_3 & cos\theta_3 cos\alpha_3 & -sin\alpha_3 & -sin\alpha_3 d_3 \\ sin\theta_3 sin\alpha_3 & cos\theta_3 sin\alpha_3 & cos\alpha_3 & cos\alpha_3 d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11} $$

$$ {}^{3}T_{4} = \begin{bmatrix} cos\theta_4 & -sin\theta_4 & 0 & a_4 \\ sin\theta_4 cos\alpha_4 & cos\theta_4 cos\alpha_4 & -sin\alpha_4 & -sin\alpha_4 d_4 \\ sin\theta_4 sin\alpha_4 & cos\theta_4 sin\alpha_4 & cos\alpha_4 & cos\alpha_4 d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12} $$

$$ {}^{4}T_{5} = \begin{bmatrix} cos\theta_5 & -sin\theta_5 & 0 & a_5 \\ sin\theta_5 cos\alpha_5 & cos\theta_5 cos\alpha_5 & -sin\alpha_5 & -sin\alpha_5 d_5 \\ sin\theta_5 sin\alpha_5 & cos\theta_5 sin\alpha_5 & cos\alpha_5 & cos\alpha_5 d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13} $$

Multiplying the above five matrices results in the total forward kinematic transformation matrix of the robot arm:

$$ {}^{0}T_{5} = {}^{0}T_{1} \times {}^{1}T_{2} \times {}^{2}T_{3} \times {}^{3}T_{4} \times {}^{4}T_{5} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{14} $$

To facilitate the processing of the data, MATLAB is used to assist in simulation modelling to calculate the Forward Kinematic. With the assistance of MATLAB robotics toolbox, the D-H parameter table is modelled by using the "Link" function, and then the five links are connected through the "SerialLink" function in order to simulate the robotic arm. Upon completion of the simulation, the parameters were grouped in a matrix and θ were set as unknown quantities. Then the values of the parameter matrices are introduced in the transformation matrix to define the joints. Finally, the transformation matrices are multiplied together to give the total forward kinematic transformation matrix.

3.2. Inverse Kinematic

Currently the inverse kinematic of robot arms are mainly calculated through two ways: analytical method and numerical method. The former one is solved by closed form of equation and the second one is solved by iterative approximation. Hence the analytical method is more complex than numerical method, due to the non-linear equation system with trigonometric functions [6], in this paper, the iterative approximation is used to obtain the inverse kinematic of this watering robot.

The position of the end link is given, which means that n, o, a, and p are known, then find the variables θ for each joint is called the inverse kinematic.

To calculate the inverse kinematic, Newton-Raphson Algorithm is used to build a Jocabi's matrix [7]. For the 5 degrees of free robot designed in the paper, set

$$ F(\theta) = (T_\theta - T_{desired}) = \begin{pmatrix} f_1 & f_2 & f_3 & f_4 \\ f_5 & f_6 & f_7 & f_8 \\ f_9 & f_{10} & f_{11} & f_{12} \\ 0 & 0 & 0 & 0 \end{pmatrix} = 0 \tag{15} $$

So the Jocabi's matrix is :

$$J_i(\theta_i) = \begin{pmatrix} \frac{\partial f_1}{\partial \theta_1^i} & \frac{\partial f_1}{\partial \theta_2^i} & \cdots & \frac{\partial f_1}{\partial \theta_5^i} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_{12}}{\partial \theta_1^i} & \frac{\partial f_{12}}{\partial \theta_2^i} & \cdots & \frac{\partial f_{12}}{\partial \theta_5^i} \end{pmatrix} \tag{16}$$

$$(i\ means\ the\ iterative\ times)$$

The iterative equation is:

$$\theta^{i+1} = \theta^i - J^{-1} f(\theta^i) \tag{17}$$

To calculate J 's pseudo-reverse, the equation could be transformed into:

$$\theta^{i+1} = \theta^i - (J_i^T J_i)^{-1} J_i^T f(\theta^i) \tag{18}$$

For f_k^1, since $T_{desired}$ is a constant, calculate the derivative of T_θ to calculate T_θ through $F(K)$, then partial derivate T_θ's components and substitute θ_i to find J. To simplify the calculation process, the inverse Kinematic is calculated through the "ikine" code in MATLAB robotics toolbox.

4. Results

4.1. Forward Kinematic Result

As for the robotic arm design the paper introduced, the data of four parameters are clearly set. They are placed in a matrix with the order of α, a, θ, d.

Table 2. Modified D-H parameter after substituting.

α	a (mm)	θ	d (mm)
0	0	θ_1	115
pi/2	126	θ_2	0
0	350	θ_3	0
0	250	θ_4	0
pi/2	0	θ_5	405

$$\begin{bmatrix} 0 & 0 & \theta_1 & 115 \\ pi/2 & 126 & \theta_2 & 0 \\ 0 & 350 & \theta_3 & 0 \\ 0 & 250 & \theta_4 & 0 \\ pi/2 & 0 & \theta_5 & 405 \end{bmatrix} \tag{19}$$

Firstly, the simulation is carried out using the above data, resulting in the following Figure 1:

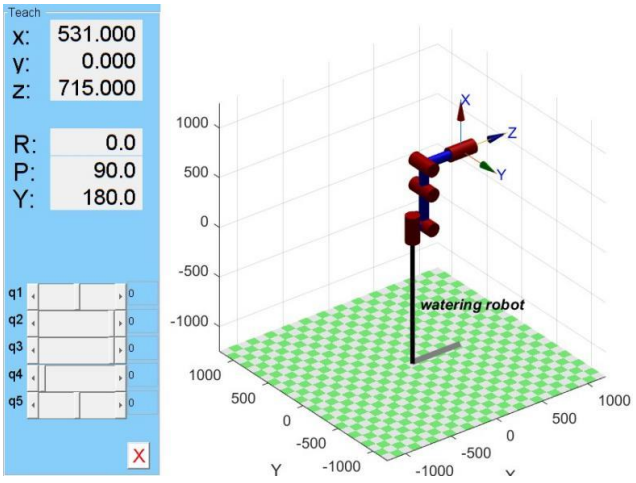

Fig. 1 The result of simulation (Photo credited: Original)

Then adjust θ through the simulation results, set θ range. Subsequently, use data above to figure out the total transformation matrix. Set theta to be:

$$\begin{bmatrix} 0 \\ -pi/6 \\ -pi/6 \\ pi/4 \\ 0 \end{bmatrix} \tag{20}$$

The final result is

$$\begin{bmatrix} 2.588 * 10^{-1} & -5.915 * 10^{-17} & 9.659 * 10^{-1} & 9.087 * 10^{2} \\ 5.915 * 10^{-17} & -1 & -7.708 * 10^{-17} & -5.003 * 10^{-15} \\ 9.659 * 10^{-1} & 7.708 * 10^{-17} & -2.588 * 10^{-1} & 4.383 * 10^{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}$$

4.2. Inverse Kinematic Results

After setting the end-position matrix equal to the matrix calculated by substituting θ in the forward kinematic, the required angle for θ can be derived using the function:

```
T_desired = T05;
q = robot.ikine(T_desired, 'mask', [1, 1, 1, 1, 1, 0]);
```

The inverse kinematic θ is calculated to be:

$$\begin{bmatrix} 0 \\ -0.5236 \\ -0.5236 \\ 0.7854 \\ 0 \end{bmatrix} \tag{22}$$

Convert radians to degrees, the result matrix transfers into:

$$\begin{bmatrix} 0 \\ -30.015 \\ -30.015 \\ 45.023 \\ 0 \end{bmatrix} \tag{23}$$

It is illustrated that the results of the forward kinematic and inverse kinematic are compatible with each other. The forward-inverse kinematic procedure is perfectly feasible.

5. Conclusion

After kinematic analysis, it can be concluded that the kinematics of this type of robot are verified. The practicality of this robot is to be tested, the practicality of this robot is still to be tested, but its forward and reverse kinematics has been possible to be derived by MATLAB assistance, which means that this robotic arm has some feasibility in terms of kinematics. The forward kinematic of this robot arm is displaced in the form of MATLAB and the inverse kinematic results from robotic toolbox code. Nevertheless, many modifications are desirable for the practical use of the robot. Some of the improvements anticipated are the extension of the arm to extend the watering range, the installation of sensors on the end of the arm to increase the number of pots that can be acted upon, and the installation of sensors on the bottom of the robot to allow for the location of pots not to be restricted to a certain area. With these modifications to this robot, the utility of the robot may be further increased, thus potentially providing some new products for the future home potted plant watering market.

Reference

[1] Yamane, K., et al. Effects of interior horticultural hctivities with potted plants on human physiological and emotional status. International Society for Horticultural Science (ISHS), 2004, 639, 37-43

[2] Asbjornsen, H., et al., Ecohydrological advances and applications in plant–water relations research: a review. Journal of Plant Ecology, 2011. 4(1-2): 3-22.

[3] Corke, P.I., A simple and systematic approach to assigning Denavit–Hartenberg parameters. IEEE Transactions on Robotics, 2007. 23(3): 590-594.

[4] Denavit, J. and R.S. Hartenberg, A kinematic notation for lower-pair mechanisms based on matrices. ASME. J. Appl. Mech, 1955. 22(2): 215-221

[5] Khalil, W. and J. Kleinfinger. A new geometric notation for open and closed-loop robots. in Proceedings. IEEE International Conference on Robotics and Automation, 1986. 3: 1174-1179

[6] Patil, A., M. Kulkarni, and A. Aswale. Analysis of the inverse kinematics for 5 DOF robot arm using D-H parameters. IEEE International Conference on Real-time Computing and Robotics (RCAR), 2017. 688-693,

[7] Stone, H.W., Inverse Kinematics, in Kinematic Modeling, Identification, and Control of Robotic Manipulators, H.W. Stone, Editor. 1987, Springer US: Boston, MA. 79-95.