

# Kinematics Analysis and Simulation of Foaming Mold Cleaning Robot by Matlab Robotics Toolbox

Zeng Qu<sup>1</sup>, Bo Yuan<sup>2</sup>

<sup>1</sup>School of Mechanical Engineering, Hubei University of Technology, Wuhan, 430068, China

<sup>2</sup>WuHan City Vocational College, Wuhan, 430064, China

**Keywords:** Cleaning Robot; Matlab; Kinematics; Trajectory Planning; Simulation.

**Abstract.** In order to better study the kinematics of the foaming mold cleaning robot, parameter design is conducted according to certain requirements for the robot, and its forward and inverse kinematics are discussed. The modified D-H Method is employed to build the kinematical equation of the corresponding structure. In addition, the Robotics Toolbox module of Matlab is utilized to carry out simulation of the trajectory planning. The simulation has observed the movement of each joint of the cleaning robot, and obtained the necessary data, and verified the designed parameters, thus achieving the predetermined goal.

## Introduction

Ever since the first robot was invented in the world, robot has gained particular favour among us. Robots are capable of playing a powerful role in the manufacturing industry through replacing the traditional manual operation with efficient automatic production. Moreover, it saves energy and reduces pollution during the production process. With the rapidly advancing technology, robots will come into play in more and more industries. For example, robotic visual system and bionic system can help to guarantee safety. Sensor technology makes robots serviceable in the exploitation and exploration fields. Robots are playing a growing number of roles in our daily lives, preventing us from danger as much as possible and bringing us a comfortable life.

Following the development of modern technology, an increasing number of automatic, intelligent technology products come into our production and life. Industrial robots are employed in harsh working environments, dangerous workplace, or special job sites of the automatic industries <sup>[1]</sup>.

There will be many impurities left on the foaming mold during the operation process, such as release agent, raw materials, and dust, so on. These residues will affect the normal use of the mold and its precision. Due to large production lots in operating the foaming mold, it is often intended to minimize the mold cleaning time as little as possible. However, traditional manual cleaning takes a lot of time. It is one of the necessary maintenance methods to prolong the mold's service life by cleaning the mold. For foaming mold, it is often maintained by mold cleaning agent. But, in most cases, the use efficiency of cleaning agent is not high, and cleaning effect not guaranteed. Therefore, this study aims to design a robot that is able to adjust its arm poses to make the air nozzles at the end of the arms reach every corner of the mold surface. And the nozzles will blow high-speed air flow to clear all the impurities like release agent left on the foaming mold, thus realizing cleaning the foaming mold.

## Kinematic Equation of Foaming Mold Cleaning Robot

In 1955, Denavit and Harteberg put forward a modeling approach of robot that basically considered robot as a link connected by a series of joints, and described the relationships between adjoining links by the link coordinate system (or "D-H Coordinate System"), while the relative positions and orientations between these coordinate systems were displayed by the homogenous transformation matrix (for short "A Matrix"). The overall transformation matrix of a robot can be obtained by collecting all the homogenous coordinates transformation starting from the base to the first joint, then to the second joint, and until to the last joint. D-H method has become the standard way to

describe and model a robot, applicable to any configuration, and also to the calculation and force analysis in Jacobian Matrix, etc [2].

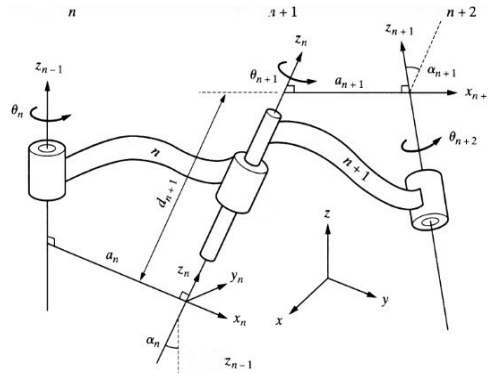


Fig. 1 parameter relationship between adjoining links

Fig. 1 shows the parameter relationship between adjoining links. And their homogenous transformation matrix is shown as follows:

$${}^nT_{n+1} = A_{n+1} = \text{Rot}(z, \theta_{n+1}) \times \text{Trans}(0, 0, d_{n+1}) \times \text{Trans}(a_{n+1}, 0, 0) \times \text{Rot}(x, \alpha_{n+1})$$

$$= \begin{bmatrix} c\theta_{n+1} & -s\theta_{n+1} & 0 & 0 \\ s\theta_{n+1} & c\theta_{n+1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_{n+1} & -s\alpha_{n+1} & 0 \\ 0 & s\alpha_{n+1} & c\alpha_{n+1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} c\theta_{n+1} & -s\theta_{n+1}c\alpha_{n+1} & s\theta_{n+1}s\alpha_{n+1} & a_{n+1}c\theta_{n+1} \\ s\theta_{n+1} & c\theta_{n+1}c\alpha_{n+1} & -c\theta_{n+1}s\alpha_{n+1} & a_{n+1}s\theta_{n+1} \\ 0 & s\alpha_{n+1} & c\alpha_{n+1} & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In the formula,  $c\theta_{n+1} = \cos\theta_{n+1}$ ,  $s\theta_{n+1} = \sin\theta_{n+1}$

$c\alpha_{n+1} = \cos\alpha_{n+1}$ ,  $s\alpha_{n+1} = \sin\alpha_{n+1}$

Thus the overall transformation from the robot base to the end-effector is:

$$T_n = {}^nT_B = {}^nT_1 {}^1T_2 {}^2T_3 \cdots {}^{n+1}T_n = A_1 A_2 A_3 \cdots A_n, \quad n \text{ refers to the number of joints.}$$

### Forward kinematics analysis of foaming mold cleaning robot

Foaming mold cleaning robot is a three degree-of-freedom robot. D-H coordinate system is set up as Fig. 2.

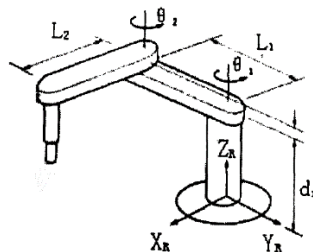


Fig. 2 D-H coordinate system

The corresponding link parameters for each joint are listed in Table 1.

Table 1 Link parameters of foaming mold cleaning robot

	$\alpha_{i-1}$	$a_{i-1}$	$\theta_i$	$d_i$	Remarks
Joint No.1	0	$l_1$	$\theta_{.1}$	0	$l_1=850\text{mm}$
Joint No.2	0	$l_2$	$\theta_{.2}$	0	$l_2=450\text{mm}$
Joint No.3	$180^\circ$	0	0	$d_3$	

Through the mathematical derivation of kinematics system, the following homogenous transformation formula is obtained as follows:

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_i - 1 \\ s\theta_i c\alpha_i - 1 & c\theta_i c\alpha_i - 1 & -s\alpha_i - 1 & -dis\alpha_i - 1 \\ s\theta_i s\alpha_i - 1 & c\theta_i s\alpha_i - 1 & c\alpha_i - 1 & di\alpha_i - 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Substitute the link parameters of each joint of the foaming mold cleaning robot into the homogenous transformation formula, results are as follows:

$$A_1 = \begin{bmatrix} c\theta_1 & s\theta_1 & 0 & l_1 c\theta_1 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c\theta_2 & s\theta_2 & 0 & l_2 c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Multiply the transformation matrixes  ${}^{i-1}_i T$  ( $i=1,2,3$ ) of each link and get the pose equation (forward kinematics equation) of the end-effector of the foaming mold cleaning robot as follows [3]:

$$\begin{aligned} {}^0_3 T &= {}^0_1 T {}^1_2 T {}^2_3 T = A_1 A_2 A_3 \\ &= \begin{bmatrix} \cos(\theta_1 - \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & l_2 \cos\theta_1 \cos\theta_2 + l_1 \cos\theta_1 \\ \sin(\theta_1 + \theta_2) & -\cos(\theta_1 - \theta_2) & 0 & l_2 \sin\theta_1 \cos\theta_2 \\ 0 & 0 & -1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

### Inverse kinematics analysis of foaming mold cleaning robot

Suppose the end position  $[p_x, p_y, p_z]^T$  of the third articulated arm is given, solve each joint variable  $[\theta_1, \theta_2, d_3]^T$ :

As the last section obtained the kinematics equation of the foaming mold cleaning robot as follows:

$${}^0_3 T = \begin{bmatrix} \cos(\theta_1 - \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & l_2 \cos\theta_1 \cos\theta_2 + l_1 \cos\theta_1 \\ \sin(\theta_1 + \theta_2) & -\cos(\theta_1 - \theta_2) & 0 & l_2 \sin\theta_1 \cos\theta_2 \\ 0 & 0 & -1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Then, the fourth column in the above transformation matrix is the end position vector of the third articulated arm, so:

$$\begin{aligned} p_x &= l_2 \cos\theta_1 \cos\theta_2 + l_1 \cos\theta_1 \\ p_y &= l_2 \sin\theta_1 \cos\theta_2 \\ p_z &= -d_3 \end{aligned}$$

Combine the above equation set, so [4]:

$$\begin{aligned} \theta_1 &= \arctan \frac{l_1 + p_y}{p_x} \\ \theta_2 &= \arccos \frac{p_x p_y}{l_1 + p_y} \\ d_3 &= -p_z \end{aligned}$$

## Trajectory Planning and Simulation by Matlab

In order to conduct trajectory planning and simulation for the foaming mold cleaning robot, first of all, a corresponding robot object should be set up. In the module of Robotics Toolbox, it is mainly to set up each joint of the robot. During setting up the joints, Link Function in the toolbox will be used, whose basic form is:

$L=LINK ([\alpha \ A \ \theta \ D \ \sigma], CONVENTION)$

In the function, alpha refers to torsion angle; A refers to link length; theta refers to joint angle; D refers to link distance; CONVENTION can be “standard” or “modified”, among which, “standard” means the standard D-H parameters, while “modified” the modified D-H parameters [5].

Adopt the modified D-H parameters for the foaming mold cleaning robot. Corresponding commands are as follows:

```
L1=LINK ([0 850 0 0 0], 'modified');
L2=LINK ([0 450 0 0 0], 'modified');
L3=LINK ([pi 0 0 600 1], 'modified'); %prismatic joint
r=robot ({L1 L2 L3});
r.name='Clean';
drivebot (r)
```

Run the above program and generate the kinematics model of the foaming mold cleaning robot, as shown in Fig. 3. And the slider control diagram of the model is displayed in Fig. 4.

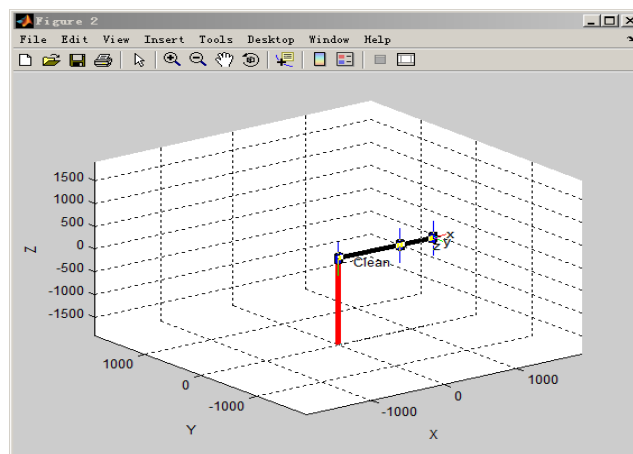


Fig. 3 Kinematics model of the foaming mold cleaning robot

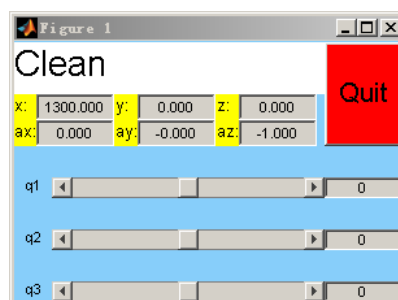


Fig. 4 Slider control diagram of the foaming mold cleaning robot

In order to verify the kinematics equation, we can randomly select three sets of joint variable and substitute them into the equation to get the set poses. Then, substitute these values into the robot's trajectory planner, and directly read the actual pose information. Compare the two sets of results.

$q1 = [1.5708 \ 0 \ 0.0195]$   $q2 = [0 \ 3.1416 \ 0.020]$   $q3 = [1.5708 \ -1.5708 \ 1.7453]$

The comparison results of the set pose and actual pose information of the wrist are shown in Table 2.

Table 2 Comparison results of the set values and actual values of the wrist position of the foaming mold cleaning robot

Group	Set value			Actual value		
	Px	Py	Pz	Px	Py	Pz
1	0.7854	1.0472	0.010	0.785	1.047	0.010
2	1.0472	0.7854	1.0472	1.047	0.785	1.047
3	0.0000	3.1416	0.020	0.000	3.142	0.020

After comparing the set values and actual values of the wrist position of the foaming mold cleaning robot in the table, it is seen that the deviation is very slight. So, this verifies that the kinematics equation and model are reliable.

Trajectory planning is to determine the path, speed at each waypoint, and acceleration of the robot's arm or joint from the starting point to the end point. Usually, trajectory planning is categorized into joint space trajectory planning and Cartesian space trajectory planning. The former plans the movement of the joints, while the latter plans the pose trajectory of the end-effector. Trajectory planning also has two types of motion, namely, point-to-point motion and continuous-path motion. This study adopts point-to-point planning in the joint space. The starting point is regulated as  $q_0 = [0 \ 0 \ 0]$  and the end point as  $q_1 = [1.5708 \ 0 \ 200]$ . And the robot's speed at the two points is zero. Motion duration is  $t=2s$ . In this planning, joint No.1 and No.3 are selected to carry out trajectory planning. Program is written as follows:

```

q0=[0 0 0];% Joint space variable at starting point
q1=[1.5708 0 200];% Joint space variable at end point
t=[0:0.2:2];% Simulation duration is 2s. Sample interval is 0.2s.
q=jtraj(q0,q1,t);% Create trajectory command
plot(r,q)
[q,qd,qdd]=jtraj(q0,q1,t);
plot(t,q(:,1))% Draw the displacement curve of joint No.1
plot(t,q(:,3))% Draw the displacement curve of joint No.3
plot(t,qd(:,1))% Draw the speed curve of joint No.1
plot(t,qd(:,3))% Draw the speed curve of joint No.3
plot(t,qdd(:,1))% Draw the acceleration curve of joint No.1
plot(t,qdd(:,3))% Draw the acceleration curve of joint No.3

```

The displacement curves, speed curves, and acceleration curves of the joint No.1 and No.3 are shown in Fig.5~10.

After observing the above curves, it is found that the displacement curves of the foaming mold cleaning robot are smooth, and the speed curves and acceleration curves are continuous. This indicates that the cleaning robot has operated smoothly and the entire structure has no evident vibration during the working process [6].

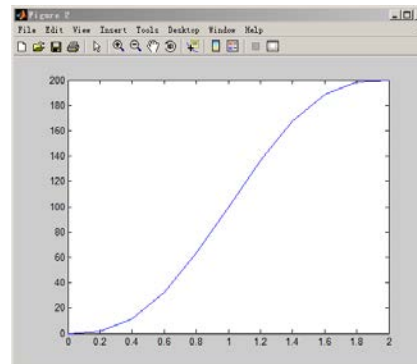
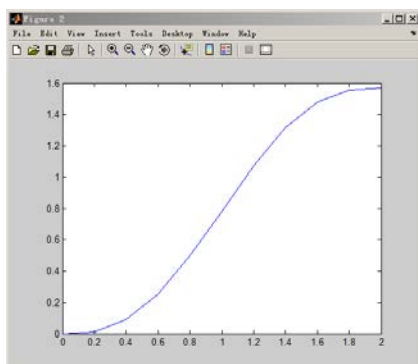


Fig. 5 Displacement curve of joint No.1 Fig. 6 Displacement curve of joint No.3

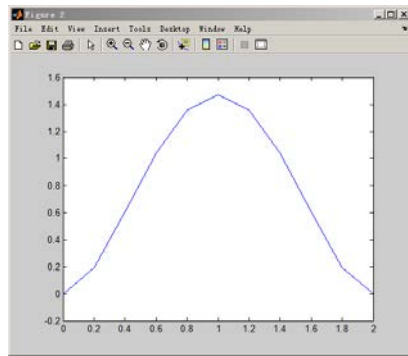


Fig.7 Speed curve of joint No.1

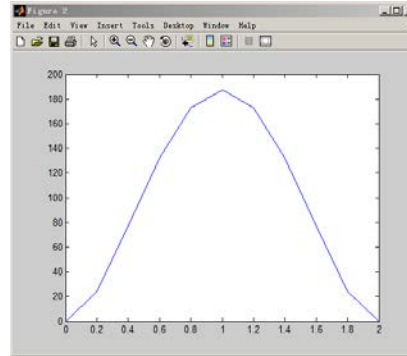


Fig.8 Speed curve of joint No.3

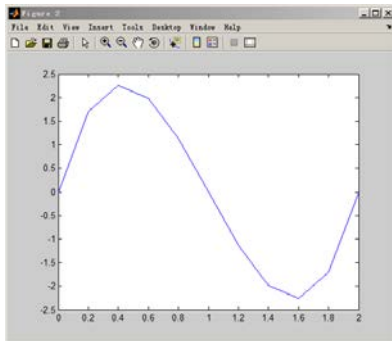


Fig.9 Acceleration curve of joint No.1

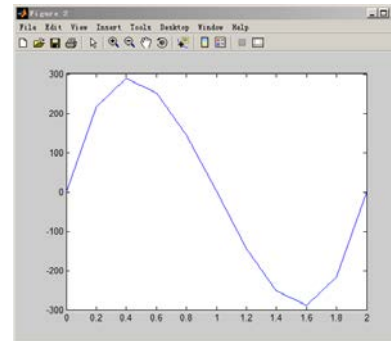


Fig.10 Acceleration curve of joint No.10

## Conclusions

This paper carries out parameter design for the foaming mold cleaning robot and analyses its forward kinematics and inverse kinematics. Moreover, trajectory planning and simulation are conducted for the robot based on Matlab.

With the assistance of the Robotics Toolbox module of Matlab, simple program is coded to conduct trajectory planning and simulation for the robot. Eventually, the parameter design is proved to be reasonable, and the kinematics equation and model of the robot are verified as reliable. In general, favorable results have been obtained.

## References

- [1] Liu Yingchun, Yu Yueqing, Jiang Chunfu. *Research Status of Flexible Robot* [J]. Machine Design, 2002, 20 (12): 4-7.
- [2] John J. Craig. *Introduction to Robotics* [M]. Translated by Yun Chao. Beijing: China Machine Press. 2005
- [3] Wang Zhixing, Fan Wenxin, Zhang Baocheng et al. Kinematics Analysis and Simulation of Industrial Robot based on Matlab [J]. Electromechanical Engineering. 2012, 29 (1): 34-37.
- [4] Luo Jiajia, Hu Guoqing. Robotic Kinematics Simulation based on Matlab [J]. Nature & Technology page, Journal of Xiamen University, 2005, 44 (5): 640-644.
- [5] Wang Yonglong, Zhang Zhaozhong, Zhang Guihong. *Matlab Language Basics and Application* [M]. Beijing: Publishing House of Electronics Industry, 2010.
- [6] Xie Bin, Cai Zixing. Robotics Simulation Experiment Teaching based on Matlab Robotics Toolbox [J]. Computer Education, 2010, 19 (10): 140-143.