

# Research and Improvement of Kinematics Algorithm of Industrial Robot based on the D-H Coordinate System

<sup>1,2</sup>Mingyu Gao, <sup>1</sup>Jiaxiang Lou, <sup>1</sup>Yu Zeng and <sup>1</sup>Zhanxiong Wu

(HangZhou DianZi University, HangZhou, China)

<sup>2</sup>E-mail: mackgao@hdu.edu.cn

## Abstract

*Owing to the intensive computation of kinematics module in the real-time control of the industrial robot, this paper sets the six-joint industrial robot as research object and establishes forward and inverse kinematics equations based on the D-H coordinate system. Then, an algorithm of searching the proper inverse kinematic solutions was presented based on the characteristics of the solutions and the geometric structure of the robot. The proposed algorithm control position and orientation comprehensively in the Cartesian coordinate space and gives speed-related Jacobian matrix. The algorithm is simulated on the MATLAB and applied to a 6R manipulator in the laboratory to verify the effectiveness. The results indicate that the algorithm can make the robotic joints move smoothly, avoid mutation and have a higher accuracy and running stability.*

**Keywords:** robot kinematics; inverse optimization; interpolation algorithm

## 1. Introduction

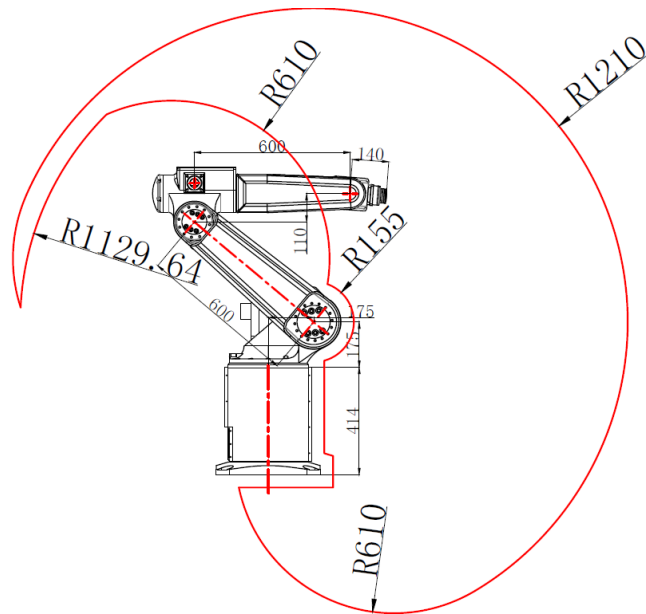
Robot kinematics plays an important role in the robot motion control. Industrial robots are mainly used to complete a certain job tasks, where it is inevitably to have a mutual conversion between joint coordinate space and Cartesian coordinate space and inevitably to achieve forward kinematics, inverse kinematics and interpolation algorithm for solving kinematic methods. The role of robot kinematics is self-evident.

Because robot kinematics have a wide coverage, this paper only research one of the more significant, including inverse kinematics optimization interpolation algorithm and Jacobian derivation. Currently, the methods of inverse kinematics include numerical iterative methods [1], geometric methods [2], neural networks [3] and genetic algorithms [4], etc. The numerical iterative methods depend on the initial value, which lead to a slow convergence rate. The geometric methods depend on the robot's special mechanical structure. The neural networks and genetic algorithms involve a large number of individual calculations, which is difficult to meet real-time requirements. The traditional method of selecting proper inverse solution generally use "shortest path", which ignore the current state of the robot and need to find all the inverse solutions of the robot.

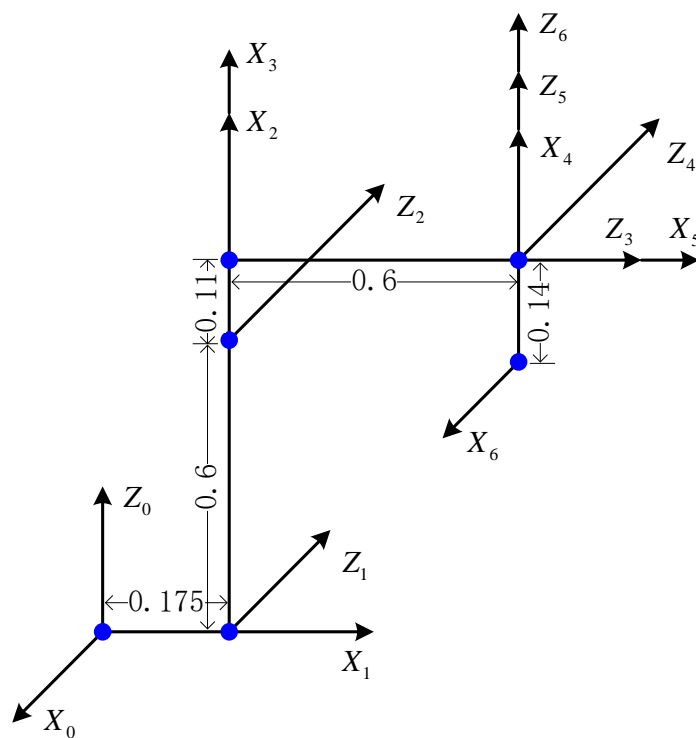
By using separation of variables to solve inverse solutions, an algorithm of searching the proper inverse kinematic solutions was presented based on the characteristics of the solutions and the geometric structure of the robot. This algorithm can get the required interpolation solution without all the inverse solution of the robot, which greatly reduce the amount of computation and improve real-time.

## 2. Forward Kinematics Equation

The dimension of six-joint industrial robot is shown in Figure 1 and Figure 2 is the robot coordinate system, which is established according to the D-H [5] method.



**Figure 1. Robot Dimension Figure**



**Figure 2. Robot D-H Reference Coordinate**

According to figure 2, we can find the certain characteristics of the robot geometric structure: no matter what position and orientation the robot is, the axis  $Z_0$  of the 1st axis, the axis  $Z_3$  of the 4th axis and the axis  $Z_5$  of the 6th axis are always in the same plane and the axis  $Z_1$  of the 2nd axis, the axis  $Z_2$  of the 3rd axis and the axis  $Z_4$  of the 5th axis are always parallel. Using D-H method, the parameters table of joint link is shown in Table 1.

**Table 1. Robot D-H parameters**

	$\theta$	d	a	$\alpha$
0 -1	$\theta_1 + 90$	0	0. 175	- 90
1 -2	$\theta_2 - 90$	0	0. 6	0
2 -3	$\theta_3$	0	0. 11	- 90
3 -4	$\theta_4$	0.6	0	9 0
4 -5	$\theta_5 + 90$	0	0	9 0
5 -6	$\theta_6 - 90$	- 0 .14	0	0

According to the D-H method, we can get the relation matrix between neighboring coordinates, recorded as:  $A_1, A_2, A_3, A_4, A_5, A_6$ . In order to simplify writing, we use the symbol  $C\theta$  to represent the symbol  $\cos\theta$ , and the same to the symbol  $S\theta$ .

$$\begin{aligned}
 A_1 &= \begin{bmatrix} -S_1 & 0 & -C_1 & -0.175S_1 \\ C_1 & 0 & -S_1 & 0.175C_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} S_2 & C_2 & 0 & 0.6S_2 \\ -C_2 & S_2 & 0 & -0.6C_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 A_3 &= \begin{bmatrix} C_3 & 0 & -S_3 & 0.11C_3 \\ S_3 & 0 & C_3 & 0.11S_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_4 = \begin{bmatrix} C_4 & 0 & S_4 & 0 \\ S_4 & 0 & -C_4 & 0 \\ 0 & 1 & 0 & 0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_5 = \begin{bmatrix} -S_5 & 0 & C_5 & 0 \\ C_5 & 0 & S_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 A_6 &= \begin{bmatrix} S_6 & C_6 & 0 & 0 \\ -C_6 & S_6 & 0 & 0 \\ 0 & 0 & 1 & -0.14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-1)
 \end{aligned}$$

And its inverse matrix is recorded as  $A_1^{-1}, A_2^{-1}, A_3^{-1}, A_4^{-1}, A_5^{-1}, A_6^{-1}$ .

$$\begin{aligned}
 A_1^{-1} &= \begin{bmatrix} -S_1 & C_1 & 0 & -0.175 \\ 0 & 0 & -1 & 0 \\ -C_1 & -S_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2^{-1} = \begin{bmatrix} S_2 & -C_2 & 0 & -0.6 \\ C_2 & S_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_3^{-1} = \begin{bmatrix} C_3 & S_3 & 0 & -0.11 \\ 0 & 0 & -1 & 0 \\ -S_3 & C_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 A_4^{-1} &= \begin{bmatrix} C_4 & S_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ S_4 & -C_4 & 0 & -0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_5^{-1} = \begin{bmatrix} -S_5 & C_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ C_5 & S_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_6^{-1} = \begin{bmatrix} S_6 & -C_6 & 0 & 0 \\ C_6 & S_6 & 0 & 0 \\ 0 & 0 & 1 & 0.14 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 & \quad (2-2)
 \end{aligned}$$

So obtain the forward kinematics equation

$${}^0T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = A_1 A_2 A_3 A_4 A_5 A_6 \quad (2-3)$$

Where,

$$\begin{aligned} n_x &= (S_1 S_{23} C_4 - C_1 S_4) S_5 S_6 - S_1 C_{23} C_5 S_6 + (S_1 S_{23} S_4 + C_1 C_4) C_6 ; \\ n_y &= -(C_1 S_{23} C_4 + S_1 S_4) S_5 S_6 + C_1 C_{23} C_5 S_6 - (C_1 S_{23} S_4 - S_1 C_4) C_6 ; \\ n_z &= -C_{23} C_4 S_5 S_6 - S_{23} C_5 S_6 - C_{23} S_4 C_6 ; \\ o_x &= (S_1 S_{23} C_4 - C_1 S_4) S_5 C_6 - S_1 C_{23} C_5 C_6 - (S_1 S_{23} S_4 + C_1 C_4) S_6 ; \\ o_y &= -(C_1 S_{23} C_4 + S_1 S_4) S_5 C_6 + C_1 C_{23} C_5 C_6 + (C_1 S_{23} S_4 - S_1 C_4) S_6 ; \\ o_z &= -C_{23} C_4 S_5 C_6 - S_{23} C_5 C_6 + C_{23} S_4 S_6 ; \\ a_x &= -(S_1 S_{23} C_4 - C_1 S_4) C_5 - S_1 C_{23} S_5 ; \\ a_y &= (C_1 S_{23} C_4 + S_1 S_4) C_5 + C_1 C_{23} S_5 ; \\ a_z &= C_{23} C_4 C_5 - S_{23} S_5 ; \\ p_x &= 0.14(S_1 S_{23} C_4 - C_1 S_4) C_5 + 0.14 S_1 C_{23} S_5 - 0.6 S_1 C_{23} - 0.11 S_1 S_{23} - 0.6 S_1 S_2 - 0.175 S_1 ; \\ p_y &= -0.14(C_1 S_{23} C_4 + S_1 S_4) C_5 - 0.14 C_1 C_{23} S_5 + 0.6 C_1 C_{23} + 0.11 C_1 S_{23} + 0.6 C_1 S_2 + 0.175 C_1 ; \\ p_z &= -0.14 C_{23} C_4 C_5 + 0.14 S_{23} S_5 - 0.6 S_{23} + 0.11 C_{23} + 0.6 C_2 . \end{aligned}$$

### 3. Inverse Kinematics Solutions

#### 3.1 Calculation Process of Inverse Solutions

Because there are many sine and cosine values in front of the coupling angle, such as  $C_{23}$ , it is impossible to extract enough elements from the matrix to solve a single sine and cosine to calculate the angle. In order to make the angle decoupling and never include a particular angle, we use a single matrix  $A_n^{-1}$  to multiply matrix  ${}^R T_H$ . Thus, we can find the element of sine and cosine values about this angle, and obtain the corresponding angle.

By the (1, 3) element  $a_x$  and (2,3) element  $a_y$  in the formula (2-3), obtain

$$\begin{cases} a_x = -(S_1 S_{23} C_4 - C_1 S_4) C_5 - S_1 C_{23} S_5 \\ a_y = (C_1 S_{23} C_4 + S_1 S_4) C_5 + C_1 C_{23} S_5 \end{cases}$$

By the (1, 4) element  $p_x$  and (2, 4) element  $p_y$  in the formula (2-3), obtain

$$\begin{aligned} p_x &= 0.14(S_1 S_{23} C_4 - C_1 S_4) C_5 + 0.14 S_1 C_{23} S_5 - 0.6 S_1 C_{23} - 0.11 S_1 S_{23} - 0.6 S_1 S_2 - 0.175 S_1 \\ &= -0.14(a_x + S_1 C_{23} S_5) + 0.14 S_1 C_{23} S_5 - 0.6 S_1 C_{23} - 0.11 S_1 S_{23} - 0.6 S_1 S_2 - 0.175 S_1 \end{aligned}$$

$$\frac{p_x + 0.14 a_x}{S_1} = -0.14 C_{23} S_5 + 0.14 C_{23} S_5 - 0.6 C_{23} - 0.11 S_{23} - 0.6 S_2 - 0.175$$

$$\begin{aligned} p_y &= -0.14(C_1 S_{23} C_4 + S_1 S_4) C_5 - 0.14 C_1 C_{23} S_5 + 0.6 C_1 C_{23} + 0.11 C_1 S_{23} + 0.6 C_1 S_2 + 0.175 C_1 \\ &= -0.14(a_y - C_1 C_{23} S_5) - 0.14 C_1 C_{23} S_5 + 0.6 C_1 C_{23} + 0.11 C_1 S_{23} + 0.6 C_1 S_2 + 0.175 C_1 \end{aligned}$$

$$\frac{p_y + 0.14 a_y}{C_1} = 0.14 C_{23} S_5 - 0.14 C_{23} S_5 + 0.6 C_{23} + 0.11 S_{23} + 0.6 S_2 + 0.175$$

So

$$\frac{p_x + 0.14a_x}{S_1} + \frac{p_y + 0.14a_y}{C_1} = 0$$

$$\frac{S_1}{C_1} = \frac{-(p_x + 0.14a_x)}{p_y + 0.14a_y} \Rightarrow \theta_1 = \arctan\left(\frac{-p_x - 0.14a_x}{p_y + 0.14a_y}\right), \theta_1 = \theta_1 + 180^\circ$$

(3-1)

In order to solve the other angles, left-multiply both sides of the matrix with  $A_1^{-1}$ , obtain  $A_1^{-1} \times {}^0T_6 = A_2 A_3 A_4 A_5 A_6$

$$\begin{bmatrix} n_{x1} & o_{x1} & a_{x1} & p_{x1} \\ n_{y1} & o_{y1} & a_{y1} & p_{y1} \\ n_{z1} & o_{z1} & a_{z1} & p_{z1} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -S_1 & C_1 & 0 & -0.175 \\ 0 & 0 & -1 & 0 \\ -C_1 & -S_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3-2)$$

$$= \begin{bmatrix} -n_x S_1 + n_y C_1 & -o_x S_1 + o_y C_1 & -a_x S_1 + a_y C_1 & -p_x S_1 + p_y C_1 - 0.175 \\ -n_z & -o_z & -a_z & -p_z \\ -n_x C_1 - n_y S_1 & -o_x C_1 - o_y S_1 & -a_x C_1 - a_y S_1 & -p_x C_1 - p_y S_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where,

$$\begin{aligned} n_{x1} &= -S_{23}C_4S_5S_6 + C_{23}C_5S_6 - S_{23}S_4C_6; \\ n_{y1} &= C_{23}C_4S_5S_6 + S_{23}C_5S_6 + C_{23}S_4C_6; \\ n_{z1} &= S_4S_5S_6 - C_4C_6; \\ o_{x1} &= -S_{23}C_4S_5C_6 + C_{23}C_5C_6 + S_{23}S_4S_6; \\ o_{y1} &= C_{23}C_4S_5C_6 + S_{23}C_5C_6 - C_{23}S_4S_6; \\ o_{z1} &= S_4S_5C_6 + C_4S_6; \\ a_{x1} &= S_{23}C_4C_5 + C_{23}S_5; \\ a_{y1} &= -C_{23}C_4C_5 + S_{23}S_5; \\ a_{z1} &= -S_4C_6; \\ p_{x1} &= -0.14S_{23}C_4C_5 - 0.14C_{23}S_5 + 0.6C_{23} + 0.11S_{23} + 0.6S_2; \\ p_{y1} &= 0.14C_{23}C_4C_5 - 0.14S_{23}S_5 + 0.6S_{23} + 0.11C_{23} + 0.6C_2; \\ p_{z1} &= 0.14S_4C_5. \end{aligned}$$

By the (1, 3) element  $a_{x1}$  and (2, 3) element  $a_{y1}$  in the formula (3-2), obtain

$$\begin{cases} -a_x S_1 + a_y C_1 = S_{23}C_4C_5 + C_{23}S_5 \\ -a_z = -C_{23}C_4C_5 + S_{23}S_5 \end{cases}$$

By the (1,4) element  $p_{x1}$  and (2,4) element  $p_{y1}$  in the formula (3-2), obtain

$$\begin{cases} -p_x S_1 + p_y C_1 - 0.175 = -0.14S_{23}C_4C_5 - 0.14C_{23}S_5 + 0.6C_{23} + 0.11S_{23} + 0.6S_2 \\ -p_z = 0.14C_{23}C_4C_5 - 0.14S_{23}S_5 + 0.6S_{23} + 0.11C_{23} + 0.6C_2 \end{cases}$$

$$\text{Set } \begin{cases} a = -p_x S_1 + p_y C_1 - 0.175 + 0.14(-a_x S_1 + a_y C_1) \\ b = -p_z + 0.14(-a_z) \end{cases}$$

$$\text{Obtain } \begin{cases} a - 0.6C_{23} - 0.11S_{23} = 0.6S_2 \\ b - 0.6S_{23} + 0.11C_{23} = -0.6C_2 \end{cases}$$

Sum of squares, obtain

$$a^2 + b^2 + 0.0121 + (0.22b - 1.2a)C_{23} = (0.22a + 1.2b)S_{23}$$

$$S_{23} = \frac{-1.2a + 0.22b}{0.22a + 1.2b} C_{23} + \frac{a^2 + b^2 + 0.0121}{0.22a + 1.2b} = cC_{23} + d$$

$$\text{Set } \begin{cases} c = \frac{-1.2a + 0.22b}{0.22a + 1.2b} \\ d = \frac{a^2 + b^2 + 0.0121}{0.22a + 1.2b} \end{cases}$$

Due to  $S_{23}^2 + C_{23}^2 = 1$

$$\text{So } C_{23} = \frac{-2cd \pm \sqrt{4c^2d^2 - 4(d^2 - 1)(c^2 + 1)}}{2(c^2 + 1)}$$

$$S_{23} = cC_{23} + d \Rightarrow \theta_{23} = \arctan\left(\frac{S_{23}}{C_{23}}\right)$$

$$\text{Due to } \begin{cases} S_2 = (a - 0.6C_{23} - 0.11S_{23})/0.6 \\ C_2 = (b - 0.6S_{23} + 0.11C_{23})/(-0.6) \end{cases}$$

$$\Rightarrow \theta_2 = \arctan\left(\frac{-a + 0.6C_{23} + 0.11S_{23}}{b - 0.6S_{23} + 0.11C_{23}}\right) \quad (3-3)$$

$$\Rightarrow \theta_3 = \theta_{23} - \theta_2 \quad (3-4)$$

Because the second and third joints are parallel, left-multiply both sides of the matrix with  $A_2^{-1}$  and  $A_3^{-1}$ , obtain

$$\begin{aligned} A_3^{-1}A_2^{-1}A_1^{-1} \times {}^0T_6 &= A_4A_5A_6 \\ \begin{bmatrix} n_{x3} & o_{x3} & a_{x3} & p_{x3} \\ n_{y3} & o_{y3} & a_{y3} & p_{y3} \\ n_{z3} & o_{z3} & a_{z3} & p_{z3} \\ 0 & 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} S_{23} & -C_{23} & 0 & -0.6C_3 - 0.11 \\ 0 & 0 & -1 & 0 \\ C_{23} & S_{23} & 0 & 0.6S_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_{x1} & o_{x1} & a_{x1} & p_{x1} \\ n_{y1} & o_{y1} & a_{y1} & p_{y1} \\ n_{z1} & o_{z1} & a_{z1} & p_{z1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} n_{x1}S_{23} - n_{y1}C_{23} & o_{x1}S_{23} - o_{y1}C_{23} & a_{x1}S_{23} - a_{y1}C_{23} & p_{x1}S_{23} - p_{y1}C_{23} - 0.6C_3 - 0.11 \\ -n_{z1} & -o_{z1} & -a_{z1} & -p_{z1} \\ n_{x1}C_{23} + n_{y1}S_{23} & o_{x1}C_{23} + o_{y1}S_{23} & a_{x1}C_{23} + a_{y1}S_{23} & p_{x1}C_{23} + p_{y1}S_{23} + 0.6S_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} -C_4S_5S_6 - S_4C_6 & -C_4S_5C_6 + S_4S_6 & C_4C_5 & -0.14C_4C_5 \\ -S_4S_5S_6 + C_4C_6 & -S_4S_5C_6 - C_4S_6 & S_4C_5 & -0.14S_4C_5 \\ C_5S_6 & C_5C_6 & S_5 & -0.14S_5 + 0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (3-5)$$

By the (1,3) element  $a_{x3}$  and (2,3) element  $a_{y3}$  in the formula (3-5), obtain

$$\frac{S_4}{C_4} = \frac{S_4C_5}{C_4C_5} = \frac{-a_{z1}}{a_{x1}S_{23} - a_{y1}C_{23}} \Rightarrow \theta_4 = \arctan\left(\frac{-a_{z1}}{a_{x1}S_{23} - a_{y1}C_{23}}\right), \quad \theta_4 = \theta_4 + 180^\circ \quad (3-6)$$

Then, left-multiply both sides of the matrix with  $A_4^{-1}$ , obtain

$$\begin{aligned}
 & A_4^{-1}A_3^{-1}A_2^{-1}A_1^{-1}\times^0T_6 = A_5A_6 \\
 & \begin{bmatrix} n_{x4} & o_{x4} & a_{x4} & p_{x4} \\ n_{y4} & o_{y4} & a_{y4} & p_{y4} \\ n_{z4} & o_{z4} & a_{z4} & p_{z4} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C_4 & S_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ S_4 & -C_4 & 0 & -0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} n_{x3} & o_{x3} & a_{x3} & p_{x3} \\ n_{y3} & o_{y3} & a_{y3} & p_{y3} \\ n_{z3} & o_{z3} & a_{z3} & p_{z3} \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 & = \begin{bmatrix} n_{x3}C_4 + n_{y3}S_4 & o_{x3}C_4 + o_{y3}S_4 & a_{x3}C_4 + a_{y3}S_4 & p_{x3}C_4 + p_{y3}S_4 \\ n_{z3} & o_{z3} & a_{z3} & p_{z3} \\ n_{x3}S_4 - n_{y3}C_4 & o_{x3}S_4 - o_{y3}C_4 & a_{x3}S_4 - a_{y3}C_4 & p_{x3}S_4 - p_{y3}C_4 - 0.6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 & = \begin{bmatrix} -S_5S_6 & -S_5C_6 & C_5 & -0.14C_5 \\ C_5S_6 & C_5C_6 & S_5 & -0.14S_5 \\ -C_6 & S_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3-7)
 \end{aligned}$$

By the (1,3) element  $a_{x4}$  and (2,3) element  $a_{y4}$  in the formula (3-7), obtain

$$\begin{cases} S_5 = a_{z3} \\ C_5 = a_{x3}C_4 + a_{y3}S_4 \end{cases} \Rightarrow \theta_5 = \arctan\left(\frac{a_{z3}}{a_{x3}C_4 + a_{y3}S_4}\right) \quad (3-8)$$

By the (3,1) element  $n_{z4}$  and (3,2) element  $o_{z4}$  in the formula (3-7), obtain

$$\begin{cases} S_6 = o_{x3}S_4 - o_{y3}C_4 \\ C_6 = n_{y3}C_4 - n_{x3}S_4 \end{cases} \Rightarrow \theta_6 = \arctan\left(\frac{o_{x3}S_4 - o_{y3}C_4}{n_{y3}C_4 - n_{x3}S_4}\right) \quad (3-9)$$

So far, there are six equations, which mean we can the required joint angles of any desired position and orientation.

### 3.2 Inverse Optimization

From above calculation process of inverse solutions, there are up to eight different solutions corresponding to a position and orientation. However, the actual robot motion only selects one of the inverse solutions, and which is also limited by the angle range of the joint. Meanwhile, how to make each joint gentle movement and reduce mutant angle of the robots is also a factor in selecting inverse solutions.

The traditional method is to find a minimum angle error accumulation of all angles [6-7], as following formula

$$D = \sum_{i=1}^6 |\theta_m - \theta_{ic}| \quad (3-10)$$

Where  $\theta_m$  represents the inverse solution of the i-th joint and  $\theta_{ic}$  represents the current angle of the i-th joint.

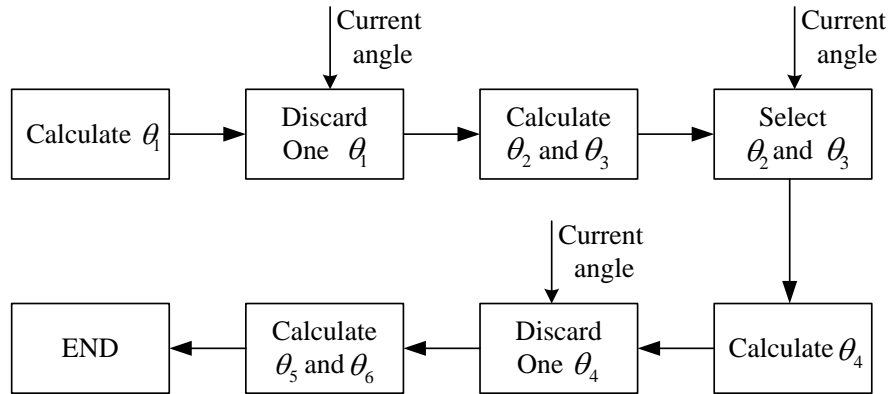
This method need to find 8 groups of inverse solutions, calculate the value  $D$  and select the minimum value  $D$ , according to formula (3-10), which speed unnecessary calculating time and lead to low efficiency.

We can simplify algorithm based on the characteristics of the solutions and the geometric structure of the robot. In order to make the robot have a high accuracy, the interpolation angles of each joint are very small. Shown in the above formula [8], the two sets of solutions  $\theta_1$  and  $\theta_4$  have a difference of  $180^\circ$ . So, we can select one closer solution and discard another solution, according to current angle of  $\theta_1$  and  $\theta_4$ . However,

the two sets of solution  $\theta_2$  and  $\theta_3$  are dependent on  $\theta_{23}$ , which is related to a particular position and orientation. So we can use following formula to find a proper one.

$$D = \sum_{i=2}^3 |\theta_{in} - \theta_{ic}| \quad (3-11)$$

$\theta_5$  and  $\theta_6$  are entirely dependent on the front solutions. Thus, we have selected a preferred group of inverse solutions. This method avoids unnecessary computing solutions to improve the real-time interpolation. The specific optimization process is shown in Figure 3.



**Figure 3. Inverse Optimization Process**

#### 4. Jacobian

The Jacobian [9-10] not only shows the speed mapping relation between operating space and joint space, but also shows the force transmitting relationship of them, which provides a convenient method to determine the static joint torque between the robot and the different coordinate systems velocity, acceleration and static transformation. In robotics, the Jacobian is a transformation matrix, which transforms the joint velocity vector to the generalized velocity vector of robot base coordinate. The simple linear transformation between the robot operating speed and the joint speed is defined as the robot Jacobian.

Transformation matrix of the robot end effector

$${}^0T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0_6R & P \\ 0 & 1 \end{bmatrix} \quad (4-1)$$

According to the differential transformation method, we can obtain a Jacobian matrix relative to the last coordinate system  $T^6$ .

$${}_{T^6}J = \begin{bmatrix} T_6 J_{11} & T_6 J_{12} & \cdot & \cdot & \cdot & T_6 J_{16} \\ T_6 J_{21} & T_6 J_{22} & \cdot & \cdot & \cdot & T_6 J_{26} \\ T_6 J_{31} & \cdot & \cdot & \cdot & \cdot & T_6 J_{36} \\ T_6 J_{41} & \cdot & \cdot & \cdot & \cdot & T_6 J_{46} \\ T_6 J_{51} & \cdot & \cdot & \cdot & \cdot & T_6 J_{56} \\ T_6 J_{61} & \cdot & \cdot & \cdot & \cdot & T_6 J_{66} \end{bmatrix} \quad (4-2)$$



Where  ${}^T_6J_{1i} = (-n_x p_y + n_y p_x)$ ,  ${}^T_6J_{2i} = (-o_x p_y + o_y p_x)$ ,  ${}^T_6J_{3i} = (-a_x p_y + a_y p_x)$ ,  ${}^T_6J_{4i} = n_z$ ,  ${}^T_6J_{5i} = o_z$ ,  ${}^T_6J_{6i} = a_z$ .

The  $i$ -th column is represented by  ${}^{i-1}T_6$ , which means the first one uses  ${}^0T_6 = A_1A_2A_3A_4A_5A_6$ , the second one uses  ${}^1T_6 = A_2A_3A_4A_5A_6$ , the third one uses  ${}^2T_6 = A_3A_4A_5A_6$ , the fourth one uses  ${}^3T_6 = A_4A_5A_6$ , the fifth one uses  ${}^4T_6 = A_5A_6$ , the sixth one uses  ${}^5T_6 = A_6$ .

The relationship between Jacobian  $J$  and relative Jacobian  ${}^T_6J$  is

$$J = \begin{bmatrix} {}^0_nR^T & 0 \\ 0 & {}^0_nR^T \end{bmatrix}^{-1} {}^T_6J \quad (4-3)$$

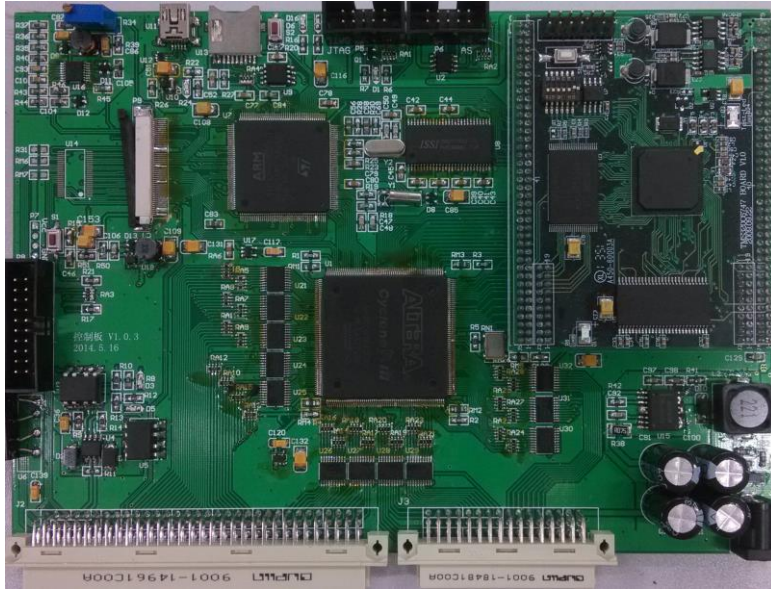
Where  ${}^0_nR^T$  is the transpose matrix of  ${}^0_nR$ , which can be obtained by the formula (4-1). We can get the Jacobian  $J$ , by put  ${}^T_6J$  and  ${}^0_nR^T$  into the formula (4-3).

## 5. Experiments and Results

Experiment With a six joint industrial robots, we send the robot real-time data to the MATLAB program for analysis, including spatial location, angle value and step value of each joint. Figure 4 shows the 6R robot in the laboratory. Figure 5 shows the main control board of the robot.



Figure 4. The 6R Robot

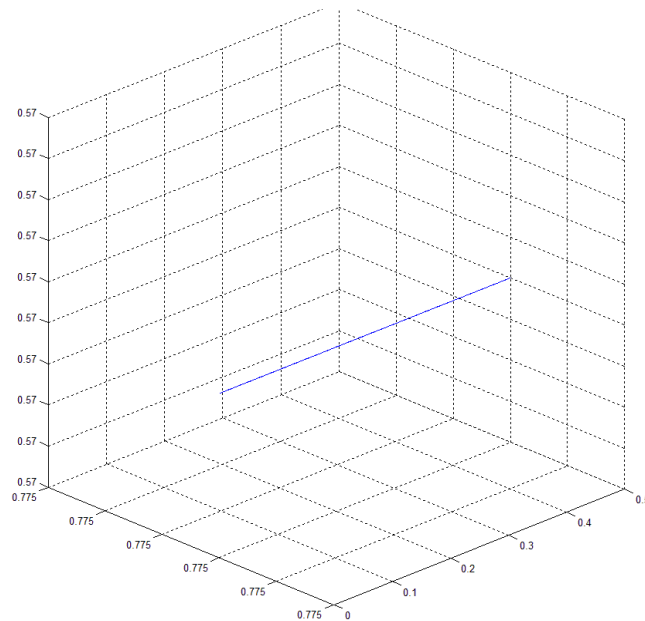


**Figure 5. The Main Control Board**

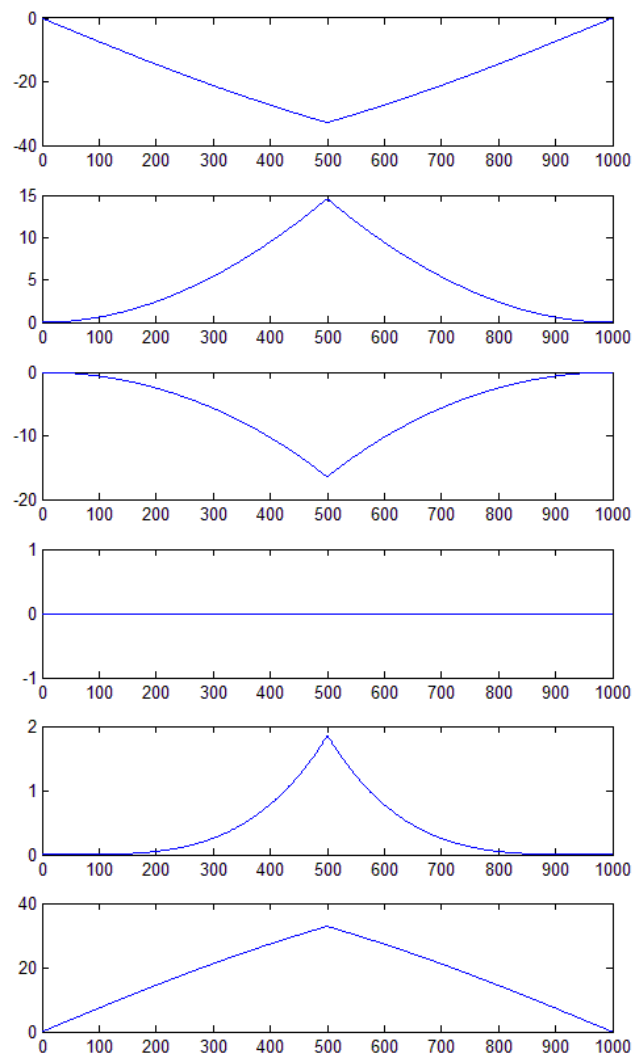
Define spatial position and orientation  $A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0.775 \\ 0 & 0 & 1 & 0.57 \\ 0 & 0 & 0 & 1 \end{bmatrix}$  and

$B = \begin{bmatrix} 1 & 0 & 0 & 0.5 \\ 0 & 1 & 0 & 0.775 \\ 0 & 0 & 1 & 0.57 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ . The step value is 0.001. The robot moves from the spatial position

and orientation A to B, and then back from B to A. Figure 6 shows the three-dimensional space robot trajectory. Figure 7 shows the real-time angle values of six-joint.



**Figure 6. Robot Trajectory**



**Figure 7. Angle Values of Six-Joint**

Shown in the figure, the interpolation angle of each joint changes evenly, which effectively reduce the possibility of angle mutations, achieve a smaller interpolation angle and help to improve the accuracy.

## 6. Conclusion

By using separation of variables to solve inverse solutions and the characteristics of the solutions and the geometric structure of the robot, we find a way to search the proper inverse kinematic solutions and convert the position and orientation interpolation into step value output of each joint. The algorithm is programmed on the MATLAB to simulate track graphical output and applied to a 6R real manipulator in the laboratory. Experimental results show that algorithm can make higher interpolation precision and avoid joint angle mutation. It can be widely used in industrial robots, and has a high application value and economic benefits.

## References

- [1] D. Manocha and J.F. Canny, "Efficient inverse kinematics for general 6R manipulator, Robotics and Automatio", (1994), pp. 648 - 657.
- [2] C. Wang, H. Wu, and Q. Miao, "Inverse kinematics computation in robotics using Conformal Geometric Algebra", Technology and Innovation Conference 2009 (ITIC 2009), International, (2009), pp. 1-5.
- [3] P. Karira, N.R. Prakash, "A neuro-genetic algorithm approach for solving the inverse kinematics of robotic manipulators", Systems, Man and Cybernetics, (2003), pp. 1979-1984.
- [4] Y. Yang, P. Guangzheng, Y. Wang and H. Zhang, "A New Solution for Inverse Kinematics of 7-DOF Manipulator Based on Neural Network", Automation and Logistics, (2007), pp. 1958-1962.
- [5] S.B. Niku, "Introduction to Robotics: Analysis, Control, Applications, Second Edition Beijing: Publishing House of Electronics Industry", (2013).
- [6] W. Wang, M.H. Xie and G.Y. Zhou, "Research on Kinematic Inverse Optimization and Its Working Space of 6-DOF Industry Robot", MACHINERY & ELECTRONICS, (2011).
- [7] J.X. Xu and W. Wang, "Two Optimization Algorithm for Solving Robotics Inverse Kinematics with Redundancy", Control and Automation, ICCA, (2007), pp. 3021-3028.
- [8] T. Wu, G.K. Wu and H.B. Wu, "Improvement of kinematics algorithm of 6R industrial robot", Journal of Mechanical & Electrical Engineering, v01. 30, no. 7, (2013).
- [9] K. Tchon, "Optimal Extended Jacobian Inverse Kinematics Algorithms for Robotic Manipulators", Robotics, (2008), pp. 1440-1445.
- [10] C.C. Chien, M. Hirano, S. Kawamura and S. Arimoto, "Approximate Jacobian control for robots with uncertain kinematics and dynamics", Robotics and Automation, (2003), pp. 692-702.

## Authors



**Mingyu Gao**, He received his MS degree in power electronics from Zhejiang University, China, in 1993, and received his PhD degree from Wuhan University of Technology, China, in 2013, in information and communication engineering. He joined Hangzhou Dianzi University, China, in 2001, where he is currently a professor in the Department of Electronic and Information. His current research interests are in the areas of NC interpolation, industrial electronics, and vehicle electronics.