



Error Analysis and Compensation Method Of 6-axis Industrial

Robot

Jianhao Zhang, Jinda Cai

Numerical Control And Display Lab, USST, 200093

ShangHai,China

Emails:378666265@qq.com

Abstract: A method of compensation is proposed based on the error model with the robot's parameters of kinematic structure and the joint angle. Using the robot kinematics equation depending on D-H algorithm, a kinematic error model is deduced relative to the end actuator of the robot, a comprehensive compensation method of kinematic parameters' error by mapping structural parameters to the joint angular parameter is proposed. In order to solve the angular error problem in the compensation process of each joint, designs a set of robot's kinematic parameter compensation experiment based on the joint angle's correction. The experimental results verify the effectiveness of the presented method. Besides, puts forward that the main factor of the dynamic error is the deformation of the connecting rods. Software compensation way is analyzed in the paper.

Index terms: industrial robot error model error compensation joint angle correction

I. INTRODUCTION

The deviation between the real robot's pose and theoretical pose, which is called the robot's pose error, directly affects the precision of localization. In many applications, robot can be used as work platform carrying instruments, so its positioning accuracy affects the working quality of the robot directly. At present, there are many scholars studying robot's error at home and abroad, based on the parameters of the model calibration, which is common in

establishing the D-H kinematic model. Based on the identified geometric parameters of the robot, error compensation is given [1-3]. Due to the complicated factors, confirming the multiple parameters' error often makes the solving of the robot's error model difficult, and the calibrated parameters are not always accurate. Therefore, the using of wider and easier compensation method is very beneficial to the improvement of the precision of the robot and the simplified process of calculation.

This paper presents a simple compensation method, using a way of mapping the structural parameters' error to the joints' angular parameter of the kinematic parameters. Such a compensation method is made on the robot error compensation, so as to avoid the accuracy of the parameters' measurement affecting the precision of the robot, which makes the error compensation effect further improve. Through the experimental comparison, the robot's position accuracy has been significantly improved. In addition, besides the compensated parameters' error, puts forward that the main factor of the dynamic error is the deformation of the connecting rods. And dynamic error compensation method is discussed and analyzed in the paper, which can be designed through adding control algorithm to the software of compensation.

II. THE ESTABLISHMENT OF THE ROBOT ERROR MODEL

Using the NCD - 1001 robot, which is designed independently, as an example to six shafts industrial robot, kinematic analysis is carried out. Assume that the robot is consisted of a series of joints and connecting links. The base of the robot is considered to be link 0. The first moving link is numbered as link 1 and so on out to the last moving link which is numbered as link n. Kinematically, a link is a rigid body which defines the relationship between two neighboring joint axes of a robot. In order to use the D-H notation for robot modeling, we must first specify a local reference frame for each joint. Therefore, for each joint, a z axis and x axis must be specified, the y axis usually does not need to be specified, because the y axis is perpendicular to the x and z axis. In figure (1), as shown in the rotary joints, θ angle is the rotating angle of the joint around the z axis, d is the distance between two adjacent common normal length (also called joint offset) in the z axis, α angle is the angle between two adjacent

z axis (also called joint torsion), β angle is the angle between two adjacent y axis (β is zero if z axis not parallel). Usually, if does not consider the deformation of the robot, only θ is the joint variable.

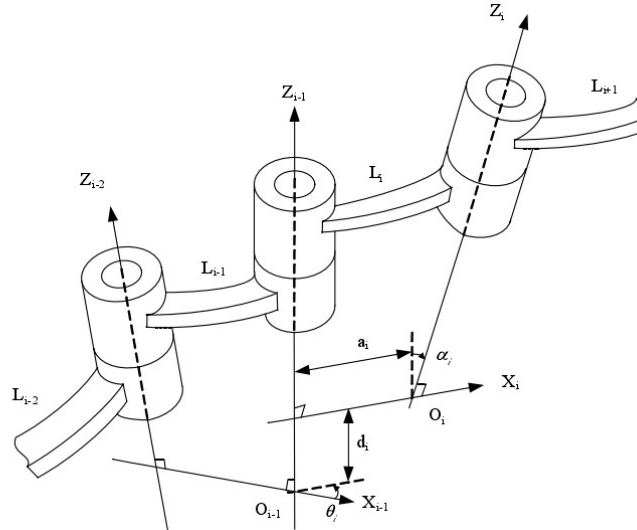


Figure 1. Parameters of D - H model

According to the classical D-H algorithm [4], the theoretical transformation matrix from coordinate system $i - 1$ to next coordinate system i is:

$$T_i^{th} = \begin{bmatrix} C\theta_i C\beta_i - S\theta_i S\alpha_i S\beta_i & -S\theta_i C\alpha_i & C\theta_i S\beta_i + S\theta_i S\alpha_i C\beta_i & a_i C\theta_i \\ S\theta_i C\beta_i + C\theta_i S\alpha_i S\beta_i & C\theta_i C\alpha_i & S\theta_i S\beta_i - C\theta_i S\alpha_i C\beta_i & a_i S\theta_i \\ -C\alpha_i S\beta_i & S\alpha_i & C\alpha_i C\beta_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Influence factors of robot's pose accuracy vary [5], mainly consisted of the robot structure parameter error and movement variable error. For rotary joints, d , a , α , β is structural parameters of the robot, which represent component size and assembling position, θ is a moving variable, represent the joint angle of the rotation joint. Because the robot manufacturing and installing process has generated structure error, directly eliminate this kind of error is very difficult. But eliminating this kind of error can be done through the indirect way of finding mapping relationship between structural parameter error and the motion variable error. And reducing the error of the movement variables is relatively simple, as long as the joint angle error is obtained, using certain compensation method can be very good to improve the precision of the robot. Therefore, establishing the error analysis model of the

robot is very important. Next, a complete error model will be established to the 6 axis industrial robot.

Robot's error model is built based on the vector differential method [5]. $\delta d_i, \delta a_i, \delta \alpha_i, \delta \beta_i$ and $\delta \theta_i$ represent each link's parameter error respectively. Because the robot has been fabricated, $\delta d_i, \delta a_i, \delta \alpha_i, \delta \beta_i$ are constant deviations, only $\delta \theta_i$ changes with motor's controlling parameters, so we can change the motor's controlling mode to reduce error.

Link i's actual transformation matrix is T_i^R , the link i's error model is:

$$d(T_i^{th}) = T_i^R - T_i^{th} = T_i^{th} \Delta_i \tag{2}$$

In the formula(2), Δ_i is the differential transformation of each parameter error relative to coordinate system of the link i.

$$d(T_i^{th}) = \frac{\partial T_i^{th}}{\partial \alpha_i} \delta \alpha_i + \frac{\partial T_i^{th}}{\partial a_i} \delta a_i + \frac{\partial T_i^{th}}{\partial d_i} \delta d_i + \frac{\partial T_i^{th}}{\partial \beta_i} \delta \beta_i + \frac{\partial T_i^{th}}{\partial \theta_i} \delta \theta_i \tag{3}$$

By formula (2) and (3), Δ_i is obtained:

$$\Delta_i = (T_i^{th})^{-1} d(T_i^{th}) = \begin{bmatrix} 0 & -S\beta_i \delta \alpha_i - C\alpha_i C\beta_i \delta \theta_i & S\alpha_i \delta \theta_i + \delta \theta_i & C\beta_i \delta \alpha_i + a_i S\alpha_i S\beta_i \delta \theta_i - C\alpha_i S\beta_i \delta d_i \\ S\beta_i \delta \alpha_i + C\alpha_i C\beta_i \delta \theta_i & 0 & -C\beta_i \delta \alpha_i + C\alpha_i S\beta_i \delta \theta_i & a_i C d_i \delta \theta_i + S\alpha_i \delta d_i \\ -(S\alpha_i \delta \theta_i + \delta \theta_i) & C\beta_i \delta \alpha_i - C\alpha_i S\beta_i \delta \theta_i & 0 & S\beta_i \delta a_i - a_i S\alpha_i C\beta_i \delta \theta_i + C\alpha_i C\beta_i \delta d_i \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{4}$$

According to the robot's differential motion principle [4]

$$\Delta_i = \begin{bmatrix} 0 & -\delta_z^i & \delta_y^i & d_x^i \\ \delta_z^i & 0 & -\delta_x^i & d_y^i \\ -\delta_y^i & \delta_x^i & 0 & d_z^i \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

Differential vector e_i is consisted of position's error and posture's error, that is:

$$e_i = \begin{bmatrix} d_x^i \\ d_y^i \\ d_z^i \\ \delta_x^i \\ \delta_y^i \\ \delta_z^i \end{bmatrix} \quad (6)$$

From the formula(4),(5),(6),we can get

$$e_i = \begin{bmatrix} C\beta_i\delta\alpha_i + a_iS\alpha_iS\beta_i\delta\theta_i - C\alpha_iS\beta_i\delta d_i \\ a_iC\alpha_i\delta\theta_i + S\alpha_i\delta d_i \\ S\beta_i\delta a_i - a_iS\alpha_iC\beta_i\delta\theta_i + C\alpha_iC\beta_i\delta d_i \\ C\beta_i\delta\alpha_i - C\alpha_iS\beta_i\delta\theta_i \\ S\alpha_i\delta\theta_i + \delta\beta_i \\ S\beta_i\delta\alpha_i + C\alpha_iC\beta_i\delta\theta_i \end{bmatrix} = \begin{bmatrix} C\beta_i & 0 & 0 & a_iS\alpha_iS\beta_i & -C\alpha_iS\beta_i \\ 0 & 0 & 0 & a_iC\alpha_i & S\alpha_i \\ 0 & S\beta_i & 0 & -a_iS\alpha_iC\beta_i & C\alpha_iC\beta_i \\ C\beta_i & 0 & 0 & -C\alpha_iS\beta_i & 0 \\ 0 & 0 & 1 & S\alpha_i & 0 \\ S\beta_i & 0 & 0 & C\alpha_iC\beta_i & 0 \end{bmatrix} \begin{bmatrix} \delta\alpha_i \\ \delta a_i \\ \delta\beta_i \\ \delta\theta_i \\ \delta d_i \end{bmatrix} = G_i\Delta x_i \quad (7)$$

G_i is the coefficient matrix of error, Δx_i is the actual error of link i .

Because formula(7) is the link i 's differential change caused by parameters' error, when measured in the coordinate of the robot's end actuator, all the errors are the ones of end actuator. Therefore, we should transform these errors to end actuator, based on the robot's differential transformation principle [5], the error of the end actuator can be represented like this:

$$e = \sum_{i=1}^n J_i e_i \quad (8)$$

$$J_i = \begin{bmatrix} n_x & n_y & n_z & (p \times n)_x & (p \times n)_y & (p \times n)_z \\ o_x & o_y & o_z & (p \times o)_x & (p \times o)_y & (p \times o)_z \\ a_x & a_y & a_z & (p \times a)_x & (p \times a)_y & (p \times a)_z \\ 0 & 0 & 0 & n_x & n_y & n_z \\ 0 & 0 & 0 & o_x & o_y & o_z \\ 0 & 0 & 0 & a_x & a_y & a_z \end{bmatrix} \quad (9)$$

J_i is differential transformation matrix [5] of link i 's coordinate system to the end coordinate system. Among them,

$$\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_i^{th} T_{i+1}^{th} \cdots T_n^{th} \quad (10)$$

Robot pose error compensation can be divided into two forms, online compensation and offline compensation [6-7]. In a certain period of time, some factors don't change with the time, such as length, joint's initial position, inertia parameters, gap of the transmission. Errors caused by these factors can be compensated in an offline way. In this paper all the parameters' errors are mapped to $\delta\theta$, through the θ compensation, the errors of the end actuator can be reduced, thus an ideal effect is achieved. But some factors changing with environmental or some other factors, such as deviations caused by the controller, temperature change and so on, if we want to control the robot perfectly, these factors must be compensated in online way.

III. EXPERIMENTAL PLATFORM AND METHOD

This experiment is consisted of the NCD-1001 robot (figure 2 right), which is independently researched and made, API T3 laser tracker (figure 2 left) and a computer.

The laser tracker is mainly composed of a tracing head, a control box, a tracker/controller cable, etc, the measurement precision can reach 1 μm , by tracking the end of the robot in a real-time measurement, the measured data (mainly points' coordinates) will be disposed in a certain way to get the errors we need. Using the laser tracking to track the end of the robot (where there is a target ball), this process is real-time, which can test repeating precision and absolute positioning accuracy when the end is going round (as shown in figure 3). Repeat accuracy is a certain position accuracy which is reached when the same group of joint angles repeat controlling order, the end actuator will move in a certain pose. According to ISO standard description, when make multiple measurement on each target point, there is a practical distribution of the measurement points. Based on the distribution of the standard deviation calculation (many times, cumulative Σ), you can define the distribution. A ± 3 times of the standard deviation ($\pm 3\sigma$ - that is 6σ) can cover 99.74% of an infinite actual position distribution. This divergence is called the repeat accuracy, it is a specified target point

repeating precision [5]. Absolute accuracy refers to the deviation of the actual end actuator trajectory and theoretical trajectory, after the robot trajectory is set in advance. Generally, the improvement of absolute positioning accuracy can ensure repeat accuracy to some satisfied degree.



Figure 2. Experimental platform

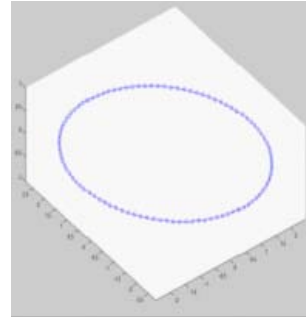


Figure 3. The motion trail

The measured errors in the experiment is mainly position errors, and position errors are mainly caused by 1, 2, 3 axis, so offline compensation are mainly used in 1, 2, 3 axis, thus can reduce the uncertain factors, bringing convenience to calculation. The relationships between $\delta d_i, \delta \alpha_i, \delta \beta_i$ and $\delta \theta_i$ can be obtained by formula (7), (8), (9) and (10):

$$k_1 \delta \alpha_i + k_2 \delta \theta_i + k_3 \delta d_i = e_{ix} \quad (11)$$

$$k_4 \delta \theta_i + k_5 \delta d_i = e_{iy} \quad (12)$$

$$k_6 \delta \alpha_i + k_7 \delta \theta_i + k_8 \delta d_i = e_{iz} \quad (13)$$

Among them, $k_1, k_2 \dots k_8$ is constant coefficient, e_{ix}, e_{iy}, e_{iz} is measured by the experiment.

In this way, all the parameter errors can be reduced by offsetting the errors to $\delta \theta_i$. The specific steps are as follows:

- 1) Use laser tracker to confirm robot's basic coordinate system, set the track r along which robot's end actuator will go, let it move several times repetitively and be real-time tracked.
- 2) Get the robot end error Δr by tracking the position of the target ball, as well as 6 angles' value through RS232, after MATLAB's calculation and analysis, put the various coordinate errors and the shaft angles' value into formula(11), (12), (13) for calculation, get the relationship between various parameters of the joint error.

- 3) Use the numerical approximation method [8] to map all other parameter error to $\delta \theta_i$, make e_{ix}, e_{iy}, e_{iz} minimum, thus the robot end's error in each direction will be minimum.
- 4) After the compensation of parameters' error, let the robot move in the same track, measure the track of the end actuator several times, use MATLAB to calculate and analyze the measured data before and after compensation, get the error of each coordinate and draw the error curve.
- 5) On the basis of the first compensation, repeat the work until reach the satisfied requirements.

Table 1 : The initial position parameters of the robot

link	$\theta(^{\circ})$	d(mm)	a(mm)	$\alpha(^{\circ})$	$\beta(^{\circ})$
1	60	0	100	90	0
2	30	0	705	0	0
3	30	0	135	90	0
4	60	750	0	-90	0
5	60	0	0	90	0
6	60	0	0	0	0

Table 2 : The $\delta \theta$ 'value of 1, 2, 3 axis in each compensation

#	$\delta\theta_1' (^{\circ})$	$\delta\theta_2' (^{\circ})$	$\delta\theta_3' (^{\circ})$
First time	0.027	0.004	0.083
Second time	0.029	0.005	0.084

IV. THE EXPERIMENTAL RESULTS AND SIMULATION ANALYSIS

a. The comparison of the results before and after each compensation

From the experiment data, the first group, which is not compensated, the error can achieve 2 mm or even greater. But after error compensation, the second group of data has obvious improvement, but still can not achieve the desired accuracy. The third group of data is on the basis of the first compensation, after calculation for each joint's compensation value, the experimental result is satisfactory.

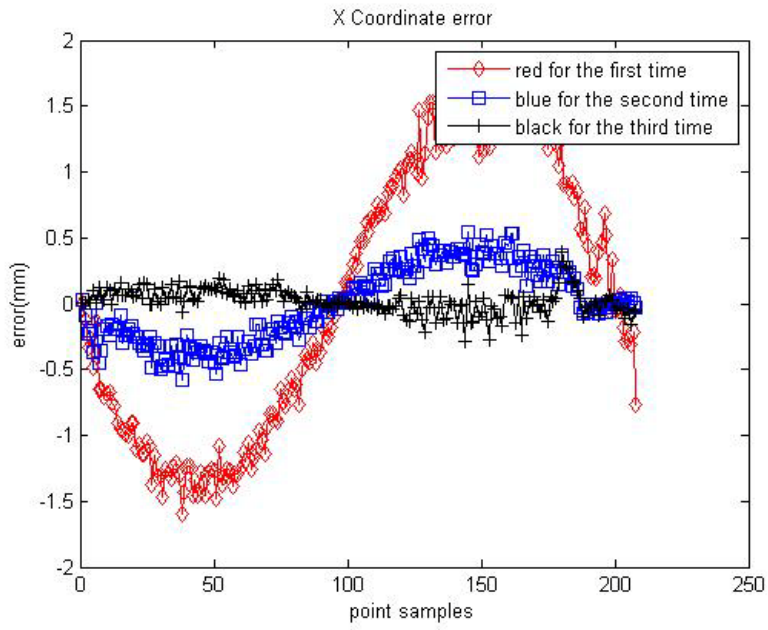


Figure 4.Errors in X axis

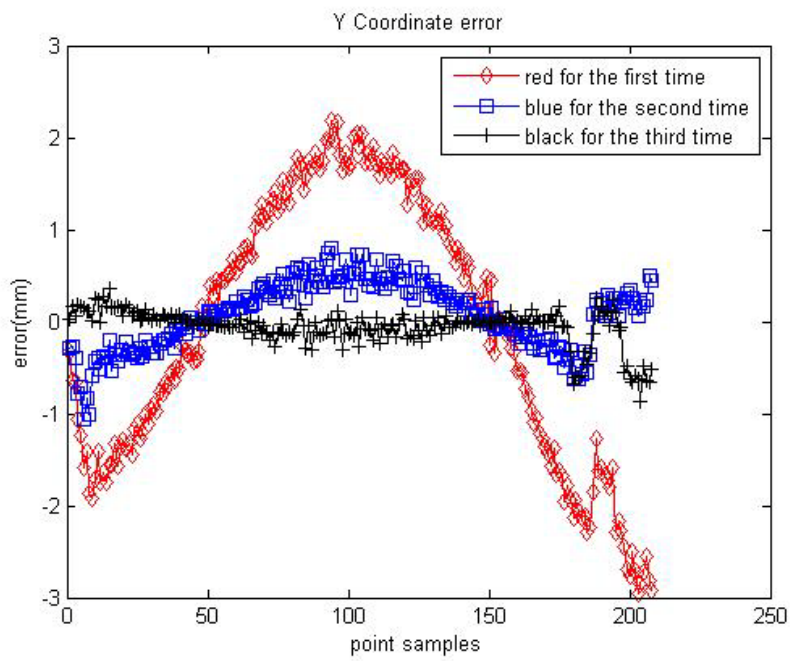


Figure 5. Errors in Y axis

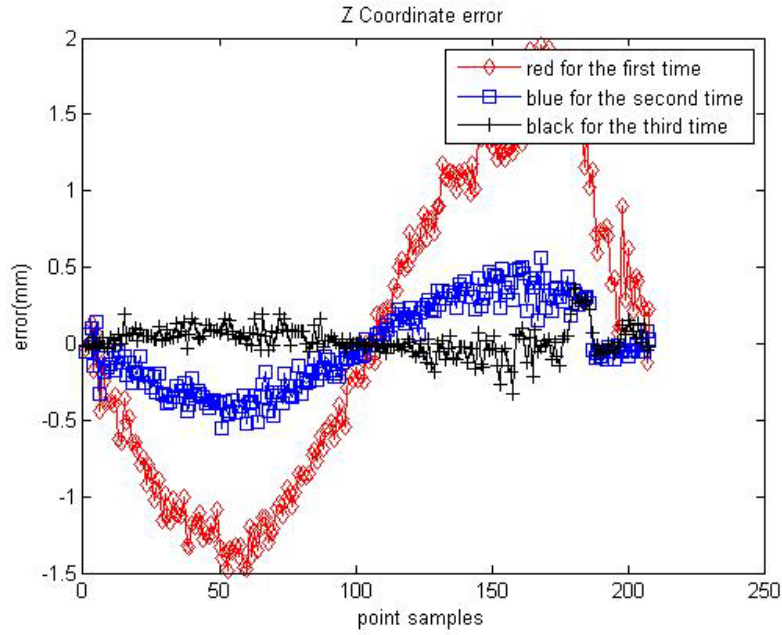


Figure 6. Errors in Z axis

Figure 4 is position error of the end actuator before and after compensation in X axis, figure 5 and figure 6 is position error of the end actuator before and after each compensation in Y and Z axis respectively. From these figures, we can say that using this offline compensation method for improving the positioning accuracy of the robot is very effective. Compared with r1, the coordinate errors in r2 and r3 are greatly reduced, and compensation effects are better as long as more efforts are made. After constant corrections, the robot’s positioning accuracy can reach 6σ standard.

Table 3 : The coordinate error variance value in three measurements.

#	D(x)	D(y)	D(z)
No Compensation	1.0809	2.0733	1.0368
First Compensation	0.0844	0.1329	0.0786
Second Compensation	0.0089	0.0379	0.0092

b. Error analysis based on the dynamics

From the experimental results, the significant improvement of the actuator positioning precision can be clearly seen, after each parameters’ error compensation. But on the other hand, due to the single offline compensation, some individual points in the experimental data have a large fluctuation, few points even to 1 mm. For the correction of these points, the

offline compensation is not enough. Therefore, we must analyze the reason behind these special points' fluctuation, and use online compensation to realize real time control, in order to prevent the generation of these mutations.

The above analysis, depending on a hypothesis, that is the robot's link is rigid, so it will not deform under pressures. In fact, because of the deflecting deformation in connecting rods and joints, effect caused by connecting rod's weight and applied load, any part of the robot will deform, and sometimes these deformation can not be ignored, even has an important influence to the robot's movement. The dynamic error caused by deformation is one of the important factors which influence the stability of robot's localization.

In the D - H model, each joint contains four parameters (θ , α , a , d), θ and α are the angle parameters of the link, the influence of deformation to these is small; But a and d are distance parameters, represent the physical connection length of certain links, the influences of deformation to these parameters can't always be ignored. In order to find out the relationship between the robot's deformation and the dynamic error, this article use ANSYS(transient structural) for the robot's dynamic analysis. Figure 7 is the total deformation of the robot, when the end actuator is moving along the path set in advance. Figure 8 is the deformation of shaft1 and shaft2, figure 9 is the deformation of shaft3 and shaft4.

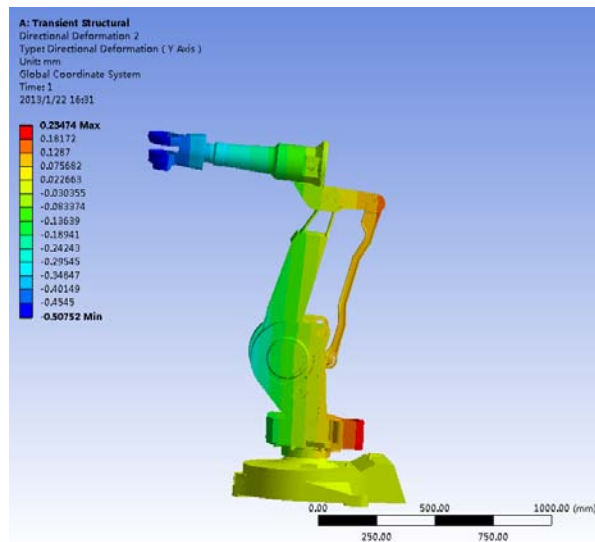


Figure 7. The total deformation of the robot

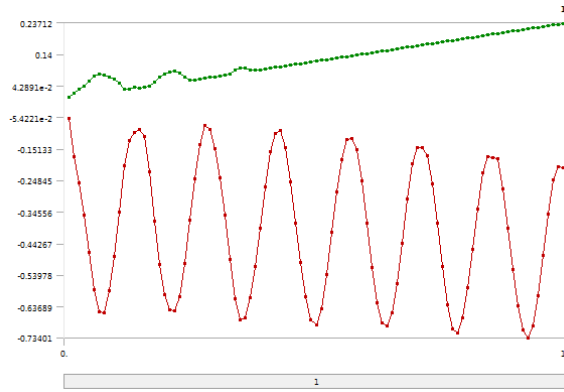


Figure 8. Deformation of shaft1 and shaft2

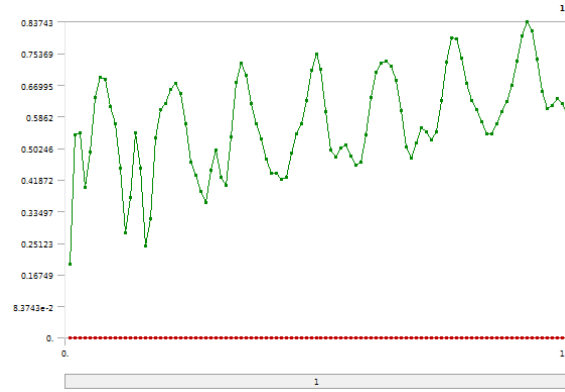


Figure 9. Deformation of shaft3 and shaft4

From the result of ANSYS we can see, the deformation of shaft2 and shaft3 is bigger than other shafts, because shaft3 is the connection of robot's waist and arm, the deformation is the biggest. What's more, larger fluctuation in the end actuator appears the same time when larger deformation of shaft2 and shaft3 appears in the process. From the simulation analysis and experimental data, we can preliminarily judge that the changes of a_2 , a_3 , d_2 , d_3 are bigger (sometimes the deformation even reach 0.1 mm in magnitude), and these parameters' changes have a direct impact on the end error fluctuations. Because the error caused by deformation is a dynamic error, which is real-time, using the normal calibration for these 6 joints(12 length parameters) is not feasible. Meanwhile, online compensation can solve this problem, but the mapping relations between shaft's deformation and end error must be set, use specific control algorithm (such as fuzzy PID, neural network, etc.) to reduce the dynamic error and control accuracy.

c. Feasibility of online compensation

Due to the fact that the dynamic error is generally nonlinear, in engineering practice, a variety of methods can be used to compensate dynamic error [9-11] . Among them, the BP network is widely used because of its strong nonlinear mapping ability in the error compensation, but the network also has disadvantages such as the learning process is long, low efficiency, easy to get local optima. Some scholars used genetic algorithm to improve the BP neural network to compensate the dynamic error, and made the control system have a global optimization ability, and also a good robustness [12], which achieved a good effect. However, the initial group in GA is selected by random distribution function, once the population become smaller or

evolution become less, the results will not have universality, fitness and the ability of deal with mutations tend to become its defects, and the real-time response is hard to meet the requirements.

Similar with neural network, fuzzy control is also a kind of control technology according to people's thinking method, it uses various sensors to detect the state of the controlled object in order to make the right decision, and realize the expected requirements. However, fuzzy control doesn't have learning ability and is unable to complete the adaptive process. If it is used as a robot's real-time dynamic error compensation, it must be combined with neural network, generate control rules and membership functions automatically through the sample's data. In this way, fuzzy control becomes more intelligent, can adapt to changes in the environment (temperature, noise, etc.).

Methods to realize fuzzy control are mainly table method, hardware controller, software realization, etc. They all have advantages and disadvantages, table method is a fast method with simple structure, costs less resource, but the accuracy is not high; Hardware has a high speed, high control precision, but the price is more expensive; The method of software realization can perform well as the hardware, and bring convenience to the process, so this paper uses the software method to realize dynamic error compensation, the establishment of fuzzy control is shown in figure10,

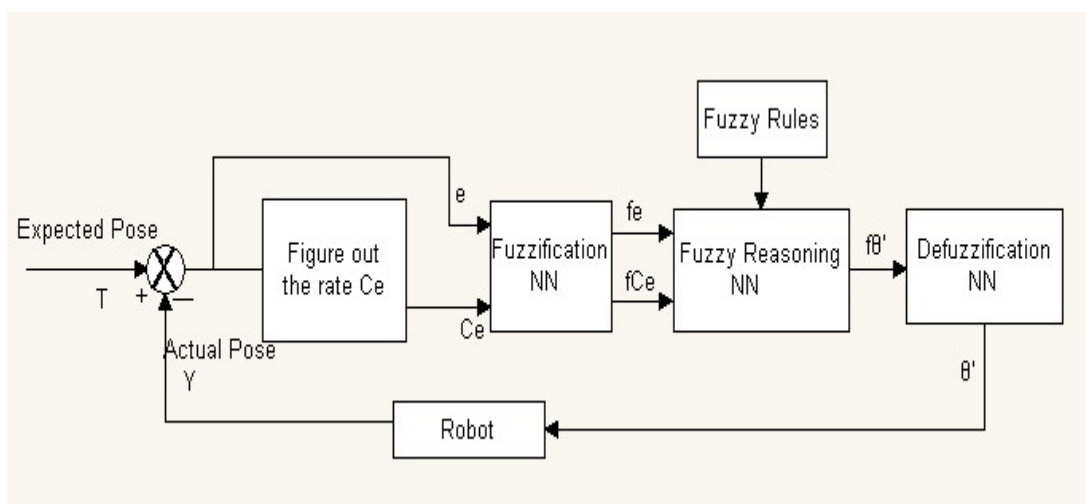


Figure 10. The model of NN fuzzy control

Among them, T is the expected pose of the robot's end actuator, Y is the actual pose of the robot's end actuator, e is the error between T and Y , Ce is e 's change rate, θ' is the

compensation value, $f_e, fC_e, f\theta'$ is the corresponding fuzzy value respectively and the fuzzy rules are got from the experiment and simulation.

Table 4 : The fuzzy rules used in this simulation.

$f\theta' \backslash fC_e$	NB	NM	NS	ZE	PS	PM	PB
f_e							
NB	NB	NB	NB	NB	NM	ZE	ZE
NM	NB	NB	NB	NB	NM	ZE	ZE
NS	NM	NM	NM	NM	ZE	PS	PS
ZE	NM	NM	NS	ZE	PS	PM	PM
PS	NM	NM	NS	ZE	PS	PM	PM
PM	NS	NS	ZE	PS	PM	PB	PB
PB	ZE	ZE	PS	PM	PB	PB	PB

{NB,NM,NS,ZE,PS,PM,PB}is the fuzzy sets.

Table 5 : The membership functions of fuzzy sets

MG \ EM	-4	-3	-2	-1	0	1	2	3	4
FS									
NB	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
NM	0.5	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0
NS	0.0	0.5	1.0	0.5	0.0	0.0	0.0	0.0	0.0
ZE	0.0	0.0	0.0	0.5	1.0	0.5	0.0	0.0	0.0
PS	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.5	0.0
PM	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0	0.5
PB	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	1.0

FS is short for Fuzzy Sets, MG is short for Membership Grade, EM is short for Element.

Using fuzzy control improved by the neural networks can have advantages of both sides [13], fuzzy rules can be used for training the neural network, and the neural network’s learning

ability can make fuzzy control intelligent enough to make response of the changing conditions, realize the adaptive requirement. In the above model, three neural networks are used, which represent the process of fuzzification, fuzzy reasoning and defuzzification, and also the input/output characteristics of three modules. In the first module, the controller's input becomes the first neural network's input, after transformed into a fuzzy quantity, its output is next network's input; in the fuzzy reasoning part, transform the experience rules into samples, train the neural network to form a mapping network; The third network will transform the previous output, which is a fuzzy quantity, into a specific control quantity (the compensation value of θ).

This control system is simulated in MATLAB, the simulation results is shown in figure 11. Figure 11 is the robot's error change in Z-axis, from the figure we can see, using this fuzzy neural network on-line compensation method can effectively reduce the error, and ultimately make accuracy be less than 0.1 mm, achieve the requirement of the robot's control. On the basis of simulation, this method can be used as a robot's online error compensation for actual application.

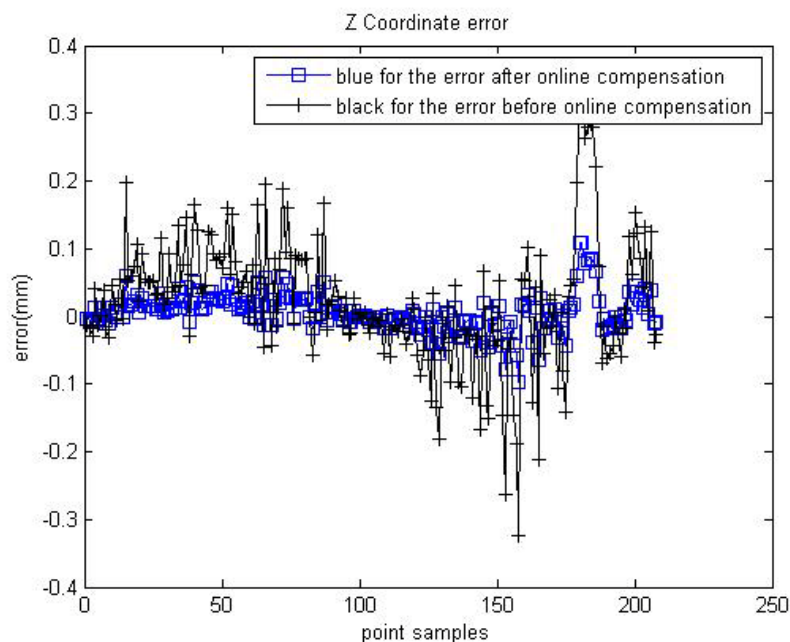


Figure 11. Errors in Z-axis

V. CONCLUSIONS

This paper is a research based on the industrial robot's motion control algorithm, which has found a suitable model for the robot's error. After getting the relationship between error and the parameters of each joint, an error compensation method has been designed to reduce error. Experimental results confirm that the error compensation method is correct and effective, but the disability of controlling dynamic error makes robot's error have fluctuation. Therefore, the analysis of the dynamic error has been done to find the cause and then a compensation method for dynamic error has been designed, the results of simulation has shown that the method is feasible. From the experiment and simulation, this improved comprehensive error compensation method has been proved that it can reduce the error and improve the precision effectively.

REFERENCES

- [1] Liu Zhenyu, Chen Yinglin, Qu Dao kui, et al. "Research on robot calibration"[J] . Robot , 2002, 24(5) :447-450.
- [2] Wang dong. "Research on industrial robot calibration technology" [D]. Shenyang: northeast university press, 2006.
- [3] RenYongJie, Zhu, Yang Xueyou etc. "Robot calibration method using laser tracker" [J]. Journal of mechanical engineering, 2007, 43 (9):195-200.
- [4] Cai Zixing. "Robotics". Tsinghua university press. 2009.
- [5] Xia Tian. "Industrial robot kinematics calibration and error analysis" [D]. Shanghai: Shanghai jiaotong university, 2009.
- [6] Jiao Guotai, LiQing, etc. "Comprehensive compensation of robot's pose error" [J]. Journal of north China, 2003, 24(2):104-107.
- [7] Gong Xingru, ShenJianXin, etc. "Industrial robot's positioning error model and its compensation algorithm" [J]. Journal of nanjing university of aeronautics and astronautics, 2012, 44(1):61-64.
- [8] Ziegert J, Datseris p. "Basic Consideration for Robot Calibration" [C]. IEEE Inter national Conference on Robotics and Automation, Philadelphia, 1988, 932-938.
- [9] Wang Yi, RenYongJie etc. "Compensation technology of measuring robot's online

dynamic temperature error” [J]. Journal of optoelectronic. Laser, 2009, 16 (4) : 492-494.

[10] Zhou Xuecai, Zheng Xuexiong. “The compensation algorithm of robot’s dynamic pose error in the work space.” Robots, 1989, 51(3):38-42

[11] Biqiang Du, Ning Xi, “Industrial Robot Calibration Using a Virtual Linear Constraint”, International Journal on Smart Sensing and Intelligent Systems,2012,5(4):987-1000.

[12] Wang Dongshu, Chi Jiannan, etc. “Genetic neural network and its application in robot error compensation” [J]. Journal of northeastern university (natural science edition), 2006, 27 (1):13-16.

[13] Zheng LI. “Neural Network Based Multisensor Fusion in a Novel Permanent Magnet Multi-DOF Actuator Orientation Detection System”. International Journal on Smart Sensing and Intelligent Systems,2012,5(4):911-926.