Hindawi
Computational Intelligence and Neuroscience
Volume 2019, Article ID 5426986, 10 pages
https://doi.org/10.1155/2019/5426986

Research Article

A Repeatable Motion Scheme for Kinematic Control of Redundant Manipulators

Kong Ying, Tang Qingqing, Zhang Ruiyang, and Ye Lv

Department of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou, China

Correspondence should be addressed to Kong Ying; kongying-888@163.com

Received 29 May 2019; Accepted 28 July 2019; Published 18 September 2019

Academic Editor: Daniele Bibbo

To achieve closed trajectory motion planning of redundant manipulators, each joint angle has to be returned to its initial position. Most of the repeatable motion schemes have been proposed to solve kinematic problems considering only the initial desired position of each joint at first. Actually, it is very difficult for various joint angles of the robot arms to be positioned in the expected trajectory before moving. To construct an effective kinematic model, a novel optimal programming index based on a recurrent neural network is designed and analyzed in this paper. The repetitiveness and timeliness are presented and analyzed. Combining the kinematic equation constraint of manipulators, a repeatable motion scheme is formulated. In addition, the Lagrange multiplier theorem is introduced to prove that such a repeatable motion scheme can be converted into a time-varying linear equation. A finite-time neural network solver is constructed for the solution of the motion scheme. Simulation results for two different trajectories illustrate the accuracy and timeliness of the proposed motion scheme. Finally, two different repetitive schemes are compared and verified the optimal time for the novelty of the proposed kinematic scheme.

1. Introduction

Robot manipulators have been playing an important role in various kinds of engineering fields. They have been widely used to perform effective and high-intensive repetitive work, such as car assembling, logistics handling, and sculpturing [1–3]. Robot arms usually have two types in needs of special operators. One is the redundant manipulators which have more degrees of freedom (DOF) than the given tasks required while the other one is the ordinary nonredundant manipulators which can complete the objective directly. Correspondingly, nonredundant manipulators refer to the one that has no additional DOF when performing a given task. Redundant manipulators are more flexible and advantageous for its redundant DOF. That is the reason why it is increasingly important in practical engineering applications [4, 5].

One of the fundamental problems in redundant manipulators motion controlling is redundancy solutions, known as inverse kinematics, which attracts many researchers' interests [6, 7]. Given the position and pose of the end-effector, calculating the homologous trajectories of joint angles with the redundant manipulators is named inverse kinematics. Liegeios-Chauvel et al. put forward a gradient projection method based on inverse kinematics solution to divide the particular motion controlling into zero space by solving the optimization goal to regulate the solutions for redundancy [8]. From then on, many researchers study pseudo inverse approaches for kinematic equation of redundant manipulators [9, 10]. However, these approaches do not only take a heavy burden of calculations, but also require the Jacobian of manipulators to be full rank, which is hard to realize.

With the development of neural networks, recurrent neural networks based on negative gradient directions are emerged. Due to the high efficient ability for computing, gradient neural networks (GNNs) are widely used in identifications and matrix equations [10]. GNN is set up by establishing non-negative direction function and to obtain a scheme for kinematic controlling. When applied to motion planning of redundant manipulators, the convergent errors generated by GNN are always lagged behind the ideal one.

That is, every joint angle of manipulators cannot return to their initial position in the end, which may cause non-repetition phenomenon in trajectory planning of the robot manipulators. With the deepening of research studies in repeatable controlling of redundant manipulators, various velocity schemes based on online quadratic optimization have been developed. Such optimal schemes incorporate the equality and inequality constraints and avoid the limitation of joint angles and joint velocities. For efficient repetitive tasks, Zhang et al. firstly introduced a repetitive motion index as the optimization criterion, using Zhang neural network (ZNN) to solve the redundancy problems [11–13]. Then, various motion schemes combining the physical joint constraints are formulated as an optimal programming index subjected to the kinematic equation of manipulators. In addition, theoretical analyses prove that the motion schemes can be converted into time-varying equations by using the Lagrange multiplier. When considering necessary conditions of Karush-Kuhn-Tucker (KKT) for nonlinear optimization problems, such an optimal programming index also can be converted into linear variational inequality (LVI) and a piecewise linear projection equation (PLPE) [14, 15]. Various neural networks have spung up to solve the LVI and PLPE. Simulations on different types of redundant manipulators are studied and different shapes of the trajectory tasks are given, which verified the effectiveness and superiority of the proposed optimal programming index for repeatable motion planning as well as the corresponding neural solvers [16, 17].

Although, most of the aforementioned approaches for repeatable motion planning of redundant manipulators are effective, the convergent time of the dynamical equations has not been ensured. That is the optimal programming index for motion controlling using neural solvers can make the joint angles of the manipulators back to initial desired position as long as time goes infinity. For the perspective of finite-time convergence, Li et al. first proposed a finite-time neural network (FTNN) model to solve the repetitive motion scheme based on ZNN in order to ensure that the convergent time is finite [18]. Then, around this FTNN, various revised FTNN models are constructed to accelerate the convergent rate of ZNN [19–21]. In the literature [22], a motion scheme of mobile robot arms based on finite-time convergence property has been set up to apply in the grasping work of the manipulators.

A redundant manipulator is a part of a robot. It is hard to locate every joint angle in the desired state at first. It is not efficient for adjusting the positions of joint angles through self-movement. However, most of the above repetitive motion plans and different neural solvers, which are mainly for ideal initial state, do not consider the deviations of manipulators. These models of repetitive motion planning based on pseudo inverse and asymptotic convergent dynamic recurrent neural networks have been studied by many researchers. Few studies are reported for finite-time repeatable motion controlling when the joint angles are deviated from the initial desired position at first [23]. In the literature [24], only a new type of terminal neural network is researched for solving motion scheme of ZNN, which has been designed from the perspective of controlling. In [25], initial deviations of joint angles are considered. The optimization performance indices is still based on infinite-time convergence, and only the neural solver for motion scheme is designed for finite-time convergence.

The remainder of this paper is organized as follows. The kinematic equations of redundant manipulators are established in Section 2. Section 3 gives out the optimal programming index, and a repeatable motion scheme of manipulators is formulated and analyzed. In Section 4, a terminal recurrent neural network algorithm is proposed to solve the mentioned motion scheme. In Section 5, simulation results on two different path trajectories with Katana6M180 manipulators verified the effectiveness and superiority of the repeatable motion scheme and the terminal neural solver. Finally, remarks and conclusions are presented in Section 6. The main contributions of this paper are summarized in the following aspects:

(1) A new optimal programming index for repeatable motion planning of redundant manipulators is exploited. It is the first time to use this performance index, which can ensure the joint angles back to their initial desired positions in finite time no matter considering the initial state of each joint of the manipulators.

(2) A special kind of neural model based on recurrent neural networks is presented to solve the repeatable motion scheme. The activation of the neural solvers has adopted limited-value function, which is applicable in practical application problems.

(3) Two different tracking tasks with redundant manipulator Katana6M180 are introduced to illustrate the superiority and effectiveness of the proposed repeatable motion scheme. Comparison results of various repetitive motion schemes are visualized in the end.

2. Kinematic Structure of Katana6M180 Robot Arm

In this section, a redundant manipulator Katana6M180 model has been set up for illustrating the effectiveness of the proposed repeatable motion scheme. The mechanical structure of the Katana6M180 robot arm is shown in Figure 1. The Katana6M180 robot arm is composed of five degrees of freedom (DOF) with three-DOF elbow for revolute joints, two-DOF wrist for revolute joints, and a gripper connected to the end-effector. The range of angular motion of Katana6M180 is displayed in Table 1. From the Table 1, joint θ_1 is the angle between horizontal line and link 1 (l_1), joint θ_2 is the angle between link 2 and link 3, joint θ_3 is the angle between link 3 and link 4, and joint θ_4 is the angle between link 4 and link 5. Joint θ_5 is between motor 5 and motor 6. In Table 1, it is shown that the rotation and extension of the joint angles are limited by the mechanical arm itself.

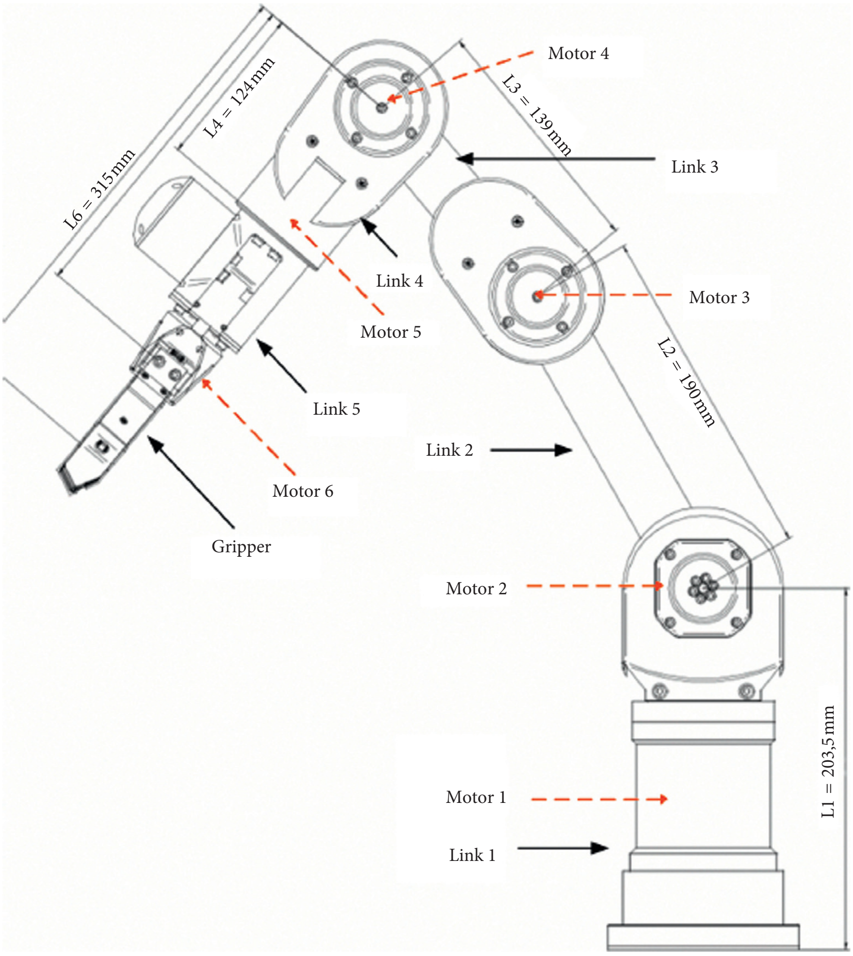


Figure 1: Design structure of redundant manipulator Katana6M180.

Table 1: Joints motion range of Katana6M180 redundant manipulator.

Joint angle	Offset angle	Motion range	Unit
θ_1	0	345.5	Degree
θ_2	124.25	140	Degree
θ_3	52.7	241.5	Degree
θ_4	63.5	232	Degree
θ_5	8.5	332.2	Degree

2.1. Kinematic Foundation of Katana6M180 Robot Arm. In this section, a kinematic frame for Katana6M180 is formulated with the DH parameters, homogeneous matrix, transformation matrix, and Jacobian matrix. The kinematic redundant arm of Katana6M180 is shown in Figure 2. From Figure 2, z_4 and z_5 are parallel to each other and are vertical to z_3. z_4 is orthogonal to x_4 and x_5. The motion point o_4 in the frame of $o_4x_4y_4z_4$ is chosen to work in coordination with o_3, which is the original point of $o_3x_3y_3z_3$. The length of a_4 is zero. From the end of the end gripper, the turning point o_5 of the frame $o_5x_5y_5z_5$ is fixed at the end of link 5. y_4 is supposed to be the center of the right-handed frames. The D-H parameters are shown in Table 2.

The homogeneous transformation matrix for K6M180 is

$$T_5^0 = \begin{pmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ T_{41} & T_{42} & T_{43} & T_{44} \end{pmatrix}, \tag{1}$$

where $T_{11} = C_1C_{23}C_5 + S_1S_5$, $T_{21} = C_{234}C_5S_1 - C_1S_5$, $T_{12} = C_5S_1 - C_1C_{234}S_5$, $T_{22} = -C_1C_5 - C_{234}S_1S_5$, $T_{13} = C_1S_{234}$, $T_{23} = S_1S_{234}$, $T_{14} = C_1(l_2C_2 + l_3C_{23} + (l_4 + l_5)S_{234})$, $T_{24} = S_1(l_2C_2 + l_3C_{23} + (l_4 + l_5)S_{234})$, $T_{31} = C_5S_{234}, T_{41} = 0$, $T_{32} = -S_{234}S_5, T_{42} = 0, T_{33} = -C_{234}$, and $T_{34} = l_1 - (l_4 + l_5)\ C_{234} + l_2S_2\ + l_3S_{23}$. Moreover, C_{ijk} is the simplified notation for $\cos(\theta_i + \theta_j + \theta_k)$. S_{ijk} is the simplified notation for $\sin(\theta_i + \theta_j + \theta_k)$. Considering the position of the end-effector, the x, y, z compose the three dimensions of the following position vector r_5^0 of the K6M180 frame:

$$r_5^0 = \begin{bmatrix} C_1(l_2C_2 + l_3C_{23} + (l_4 + l_5)S_{234}) \\ S_1(l_2C_2 + l_3C_{23} + (l_4 + l_5)S_{234}) \\ l_1 - (l_4 + l_5)C_{234} + l_2S_2 + l_3S_{23} \end{bmatrix}. \tag{2}$$

The solutions for inverse kinematic problems are not suitable for using forward kinematic formations. For the existence of singularity with the robotic arms, it is necessary

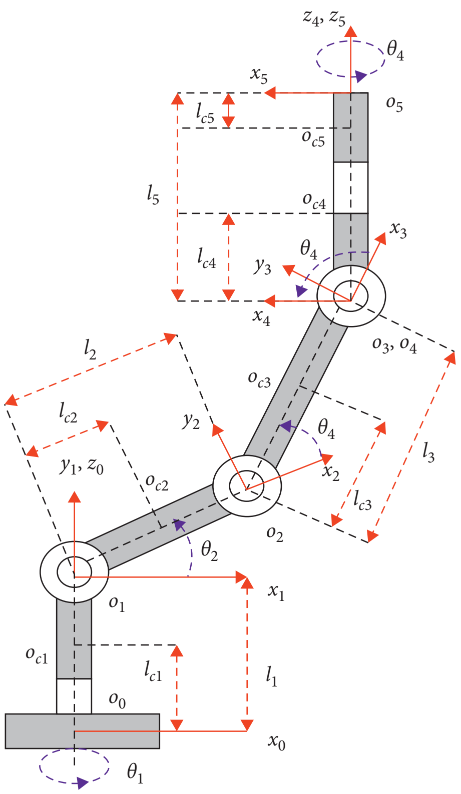


Figure 2: Mechanical assignment for Katana6M180 manipulator.

Table 2: D-H parameters of Katana6M180 redundant manipulator.

Link_i	a_i	α_i	d_i	θ_i
1	90°	0	l_1	θ_1
2	0	l_2	0	θ_2
3	0	l_3	0	θ_3
4	90°	0	0	θ_4
5	0	0	$l_4 + l_5$	θ_5

to find out an appropriate kinematic solution to solve the inverse kinematics.

2.2. Traditional Solution for Inverse Kinematic of Katana6M180 Robot Arm. Given a special trajectory with the end-effector of the Katana6M180 manipulator, the inverse problem is to solve each joint angle corresponding to the moving trail of the end-effector in the dimensional space. For the solving method for redundant manipulators, algebraic algorithms, iterative algorithms, and geometric algorithms are the primary solutions for trajectory planning problems. Algebraic algorithms do not propose a systematic method for choosing a special solution of the possible ways for motion planning of redundant arms. Iterative algorithms take a burden of the computational time, and in the singularity situations, experimental error may not reach to zero in a long time.

3. Repeatable Motion Scheme for Redundant Manipulators

As mentioned above, given a desired trajectory for redundant manipulators, each joint angle of the robot arm has to be returned to its desired initial position in the end. Traditional algorithms for inverse motion designing are not applicable for repetitive control especially in the situation that the initial position of the manipulators may not be in their desired places at first. Furthermore, the joint constraints of redundant manipulators should be taken into account. Motivated by these practical ideas, a repeatable inverse motion scheme for redundant manipulators is formulated as follows:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\left(\dot{\theta}(t)+\beta_{\theta}c\right)^{T}\left(\dot{\theta}(t)+\beta_{\theta}c\right) \\ \text{subject to} \quad & J(\theta)\dot{\theta} = \dot{r}^{*} + k_{p}\left(r^{*} - f(\theta)\right), \end{aligned} \tag{3}$$

where $c = |\theta(t) - \theta^{*}(0)|^{\alpha}\operatorname{sgn}(\theta(t) - \theta^{*}(0))$. $0 < \alpha < 1$, $\theta^{*}(0)$ is the desired initial position of the joint variable vector. The design parameter $\beta_{\theta} > 0$ is used to form the joint displacement of the manipulators. $k_p > 0 \in R$ represents the magnitude parameter to control the convergent speed of the end-effector. r^{*} is the desired motion trajectory of the end-effector. $\dot{r}^{*}$ is the speed vector of the end-effector. Considering that the initial position of the end-effector may not be at the desired initial point, the position error between the actual trajectory $f(\theta)$ and the desired motion trajectory r^{*} are needed to be reduced for changing the motion direction.

3.1. Convergent Analysis. According to the ZNN theories, the following equation has been formulated:

$$\dot{E}(t) = -\mu E(t), \tag{4}$$

where $E(t) = \theta_t - \theta_0$, $\mu > 0$. It follows the fact that the convergent error $E(t)$ reduces to zero as time goes by. By applying the ZNN theory, we get the scheme $\dot{\theta}(t) = -\mu(\theta(t) - \theta(0))$, then, $\dot{\theta}(t) + \mu(\theta(t) - \theta(0)) = 0$. Therefore, we obtain the ZNN repetitive index:

$$\frac{1}{2}(\dot{\theta}(t) + \mu(\theta(t) - \theta(0)))^{T}(\dot{\theta}(t) + \mu(\theta(t) - \theta(0))). \tag{5}$$

Motivated by the ZNN theory, a finite-time convergent neural network model has been proposed, which greatly reduced the convergent time. The error dynamics of the finite-time convergence neural model is described as follows:

$$\dot{E}(t) = -\beta_{\theta}|E(t)|^{\alpha}\operatorname{sgn}(E(t)). \tag{6}$$

Setting the following Lyapunov function,

$$V(t) = \frac{1}{2}E^{2}(t). \tag{7}$$

The derivation of equation (7) with respect to time is as follows:

$$\dot{V}(t) = -\beta_\theta |E(t)|^\alpha |E(t)| = -\beta_\theta\left(\sqrt{2V(t)}^{\,1+\alpha}\right) < 0. \quad (8)$$

For $\dot{V}(t) < 0$, the finite-time convergent model (6) is asymptotically stable in the end. In addition, the above equation can be rewritten as

$$\frac{1}{V^{1+\alpha/2}(t)} dV(t) = -\beta_\theta 2^{(1+\alpha)/2} dt\ . \quad (9)$$

Integrating both sides produces

$$\frac{2}{1-\alpha} V^{(1-\alpha)/2}(t) - \frac{2}{1-\alpha} V^{(1-\alpha)/2}(0) = -\beta_\theta 2^{(1+\alpha)/2} t. \quad (10)$$

When $E(t)$ has been converged to zero, $V(t) = 0$, $V^{(1-\alpha)/2}(t) = 0$, equation (8) is rewritten as follows:

$$t_s = \frac{2^{(1-\alpha)/2}}{\beta_\theta (1-\alpha)} V^{(1-\alpha)/2}(0). \quad (11)$$

For repetitive motion planning of redundant manipulators, we set $E(t) = \theta(t) - \theta^*(0)$, and the dynamical equation (6) can be changed into (11).

$$\dot{\theta}(t) + \beta_\theta \left|\theta(t) - \theta^*(0)\right|^\alpha \mathrm{sgn}\left(\theta(t) - \theta^*(0)\right) = 0. \quad (12)$$

That is,

$$\left\|\dot{\theta}(t) + \beta_\theta \left|\theta(t) - \theta^*(0)\right|^\alpha \mathrm{sgn}\left(\theta(t) - \theta^*(0)\right)\right\|_2^2 = 0. \quad (13)$$

According to the solution of norm problem (11), we set the repeatable motion index in equation (3).

3.2. Scheme Formulation. For the repeatable motion scheme (3), the performance kinematic index is reformulated as $(\dot{\theta}(t) + \beta_\theta c)(\dot{\theta}(t) + \beta_\theta c) = \dot{\theta}(t)^T \dot{\theta}(t) + 2\beta_\theta c^T \dot{\theta}(t) + \beta_\theta c^T \beta_\theta c$. Since the motion scheme is only considered the velocity level and $\dot{\theta}(t)$ is the variable vector, the term $\beta_\theta c^T \beta_\theta c$ is visualized as a positive constant. Although the limits of joint velocity and joint angle are not emerged into the scheme index, the range of rotated joints is still reflected in the program. Due to the deviations among initial positions of joint angles, a feedback control $r - f(\theta)$ is added into the motion equation of redundant manipulators to guarantee that the end-effector will back to the desired initial trajectory at last.

By analyzing and verifying the above deductions, the repeatable motion scheme (3) can be simplified as the following index:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\dot{\theta}^T \dot{\theta} + \beta_\theta c^T \dot{\theta} \\ &\text{subject to} \quad J(\theta)\dot{\theta} = \dot{r}^* + k_p\left(r^* - f(\theta)\right), \end{aligned} \quad (14)$$

where $k_p > 0 \in R$ represents a feedback gain parameter. $c = |\theta(t) - \theta^*(0)|^\alpha \mathrm{sgn}(\theta(t) - \theta^*(0))$. Besides, $0 < \alpha < 1$, $\theta^*(0)$ is the initial position of joint angle vectors. The motion scheme (12) visualizes the basic kinematics of the redundant manipulators. From the optimization equation (12), joint velocity level limitations for joint angle and joint velocity are difficult to be combined with the motion scheme. The situation of exceeding the joint limits has been considered in the simulation programs.

4. Neural Network Solving

Considering the scheme of repeatable motion planning, we change equation (12) by using the method of Lagrange multipliers.

$$\begin{aligned} L(\dot{\theta}(t), \lambda(t), t) = {} & \frac{\dot{\theta}^T(t)\dot{\theta}(t)}{2} + \beta_\theta c^T \dot{\theta}(t) \\ & + \lambda^T(t)\left(J(\theta)\dot{\theta}(t) - \dot{r}^* - k_p\left(r^* - f(\theta)\right)\right), \end{aligned} \quad (15)$$

where $\lambda^T(t) \in R^{1\times m}$ denotes the Lagrangian-multiplier vector.

$$\begin{aligned} \frac{\partial L(\dot{\theta}(t), \lambda(t), t)}{\partial \dot{\theta}(t)} &= \dot{\theta}(t) + \beta_\theta c + J^T(\theta)\lambda(t) = 0, \\ \frac{\partial L(\dot{\theta}(t), \lambda(t), t)}{\partial \lambda(t)} &= J(\theta)\dot{\theta}(t) - \dot{r}^* - k_p\left(r^* - f(\theta)\right) = 0. \end{aligned} \quad (16)$$

The following time-varying equation (14) can be obtained:

$$W(t)y(t) = v(t). \quad (17)$$

With

$$\begin{aligned} W(t) &= \begin{bmatrix} I & J^T(t) \\ J(t) & 0 \end{bmatrix} \in R^{(n+m)\times(n+m)}, \\ y(t) &= \begin{bmatrix} \dot{\theta}(t) \\ \lambda(t) \end{bmatrix} \in R^{n+m}, \\ v(t) &= \left[-\beta_\theta c \dot{r}^* + k_p\left(r^* - f(\theta)\right)\right]^T \in R^{n+m}, \end{aligned} \quad (18)$$

where vector I represents the identity matrix. We may calculate the convergent error by setting

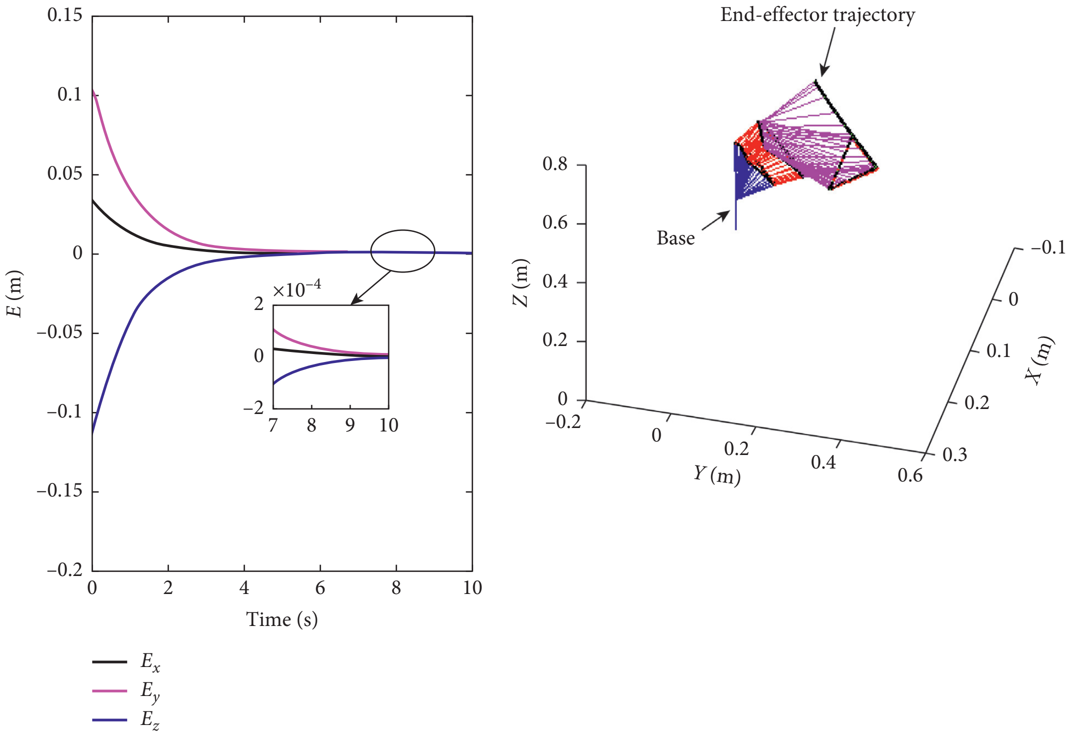
$$E(t) = W(t)y(t) - v(t). \quad (19)$$

According to the neural solver (10), we get the following dynamic system equation form for trajectory planning:

$$\dot{y} = \dot{W}y + (W + I)\dot{y} - \dot{v} + \beta_\theta |Wy - v|^\alpha \mathrm{sgn}(Wy - v). \quad (20)$$

5. Applications to Redundant Manipulators

In this section, two experimental examples are displayed and analyzed to illustrate the repetitiveness and finite-time convergence of the proposed motion scheme (12) for neural model solving. Comparisons on different repetitive motion schemes based on Katana6M180 manipulator are introduced to substantiate the



(a)

(b)

FIGURE 3: Simulation profiles for Katana6M180 manipulator synthesized with motion scheme (12).

superiority and timeliness of the proposed index for task controlling.

5.1. Triangle-Path Tracking Task. For this experimental simulation, the end-effector of the Katana6M180 manipulator is required to track a triangle path. During the simulation, $t \in [0, T]$, the desired trajectory of the manipulator is defined as follows:

$$
\begin{gathered}
r_x^* = \begin{cases} 0.05\sin^2\left(\dfrac{0.5\pi t}{3}\right), & 0 \le t \le 3, \\[2ex] 0.05\sin^2\left(\dfrac{0.5\pi(t-3)}{3}\right), & 3 < t \le 6, \\[2ex] -0.1\sin^2\left(\dfrac{0.5\pi(t-6)}{3}\right), & 6 < t \le 9, \end{cases} \\[2ex]
r_y^* = \begin{cases} 0.05\sqrt{3}\sin^2\left(\dfrac{0.5\pi t}{3}\right), & 0 \le t \le 3, \\[2ex] -0.05\sqrt{3}\sin^2\left(\dfrac{0.5\pi(t-3)}{3}\right), & 3 < t \le 6, \\[2ex] 0, & 6 < t \le 9, \end{cases} \\[2ex]
r_z^* = 0.
\end{gathered}
\tag{21}
$$

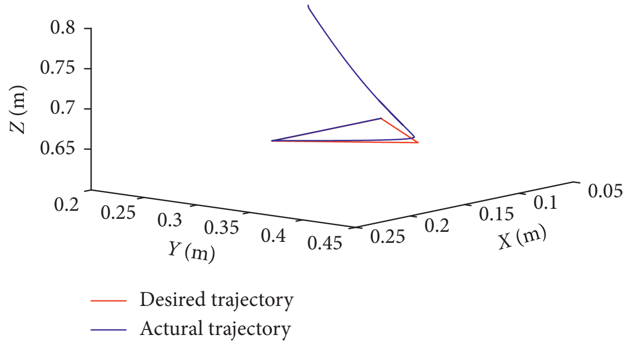
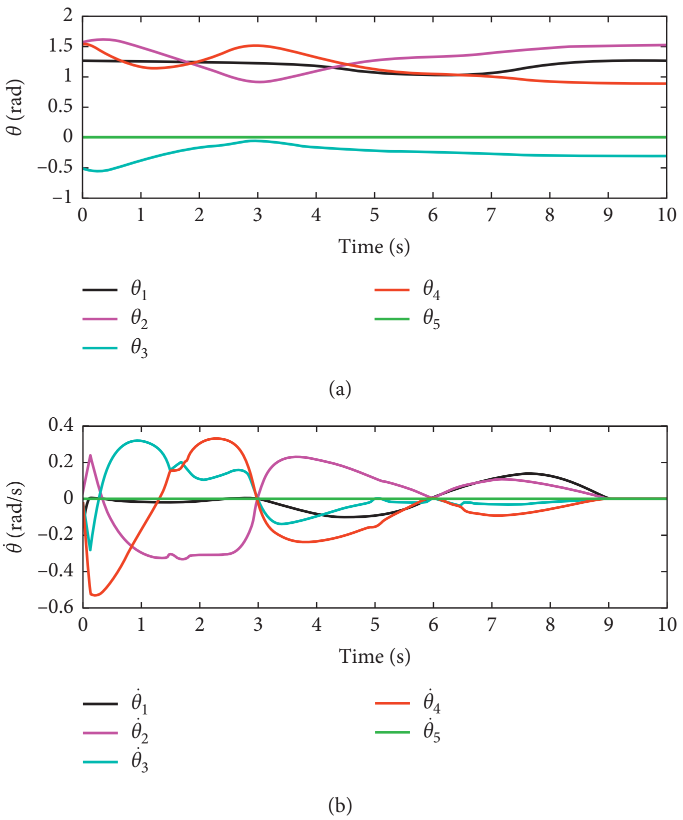


FIGURE 4: Trajectory profiles with desired path and actual path of end-effector.

The side length is assumed to be 0.18 m. The task completing time is set $T = 9$ s. The desired initial position of each joint angle is set to $\theta^*(0) = [2\pi/5, \pi/2, 0, -\pi/6, \pi/3, 0]^T$ rad. Considering the deviation of joint angles before tracking, the initial position of each joint angle is set to be $\theta(0) = [2\pi/5, \pi/2, 0, -\pi/6, \pi/3 + 0.5, 0]^T$ rad. Furthermore, design parameter $\beta_\theta = 1, k_p = 1, \alpha = 0.1$. The simulation profiles are shown in Figures 3–5. In Figure 3(b), the end-effector of K6M180 is tracking a triangle path. The end-effector comes back to its desired initial position though the initial positions are not correctly settled. In Figure 3(a), the error position of XYZ has converged to zero around 6 s and the convergent precision is less than 2.4×10^{-4} at the end of the tasks. From Figure 4 the end-effector rapidly moves to the desired



(a)

(b)

Figure 5: Trajectory profiles with joint angles and joint velocities of the manipulator.

motion trajectory under the repeatable motion scheme (12). That is the joint-drifting phenomenon can be remedied under the neural network solving control. With the Figure 5, the corresponding trajectory profiles of joint angles and joint velocities demonstrated the final statement of the manipulator. It is shown that all the joint angles have been returned to the desired position, and the motion velocity of each joint angle has been reduced to zero in the end. For obvious illustration, Table 3 shows the convergent errors of six joints. From Table 3, the maximum error in joint position is 3×10^{-4}.

5.2. Circular-Path Tracking Task. In this part, the end-effector of the manipulator what we provide has been planned to rack a circular path. The radius of the tracking circle is set to be 0.05 m. The corresponding profiles synthesized with the motion scheme (12) under the neural solver are shown in Figures 6–8. Figure 6 shows the trajectory of each joint angle along the motion procedure. From Figure 6, we can find each joint has performed a closed path at last. The end-effector of the manipulators has been returned to the desired initial position although the deviations existed at first. End-effector path and position error XYZ are shown in Figure 7. From Figure 7(a), we can obviously discover the end-effector has chased the desired task in several time and has performed a precise circular path with time. In Figure 7(b) the convergent precision in three direction XYZ is about 2×10^{-3}, which demonstrates that the task is completed well.

Table 3: Error deviations of the Katana6M180 manipulator under the different scheme (12) and scheme (17), which are solved by neural networks tracking a circular path.

Joint	Finite-time solving scheme	ZNN solving scheme
$\theta_1(9) - \theta_1(0)$	−0.00001982	−0.0001762
$\theta_2(9) - \theta_2(0)$	−0.00002644	−0.0002344
$\theta_3(9) - \theta_3(0)$	−0.000009387	−0.00006367
$\theta_4(9) - \theta_4(0)$	−0.000009656	−0.00007636
$\theta_5(9) - \theta_5(0)$	-8.022×10^{-9}	-7.012×10^{-8}

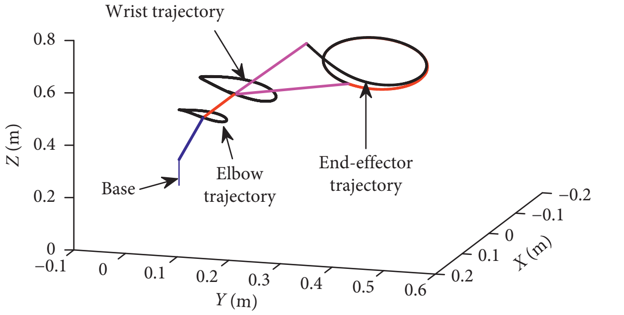
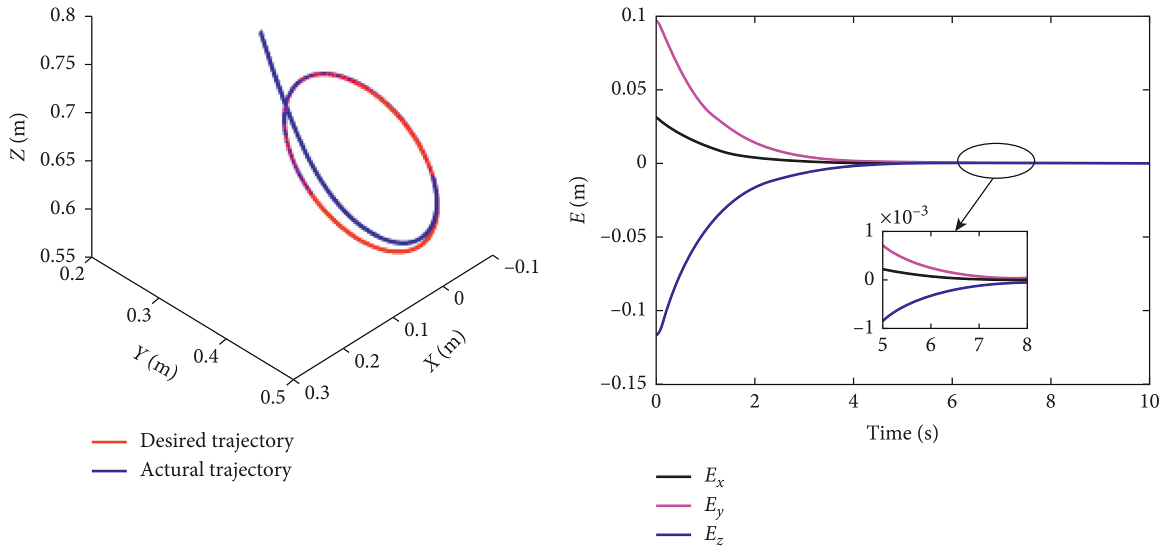


Figure 6: Trajectories of each joint angle to follow the circular-path tracking.

5.3. Comparisons. As we discuss the repeatable motion scheme, the ZNN method formulated with the repetitive motion plan has been provided in many literatures [24, 25]. It can be generalized in velocity level as follows:



(a)

(b)

Figure 7: Path profiles of the end-effector when it is tracking the given path under the scheme index (12).

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2}\dot{\theta}^T\dot{\theta} + \mu(\theta-\theta(0))^T\dot{\theta} \\ &\text{subject to} \quad J(\theta)\dot{\theta} = \dot{r}^* + k_p(r^* - f(\theta)), \end{aligned} \tag{22}$$

where $\mu > 0$ is the magnifying parameter to scale the convergent speed. In the motion index (17), the end-effector of the redundant manipulator can return to the initial position after executing the triangle trajectory as long as time is infinite. That means the convergent time is supposed to be infinity, which is not applicable in real-time processing. In addition, the initial position of the joint angles is assumed to be in the desired position at first, which may lead to the inaccurate convergence by manipulators. In order to illustrate the advantages of our finite-time model in repetitive motion, the comparison experiments are simulated on the same Katana6M180 manipulators. For simplifying the simulation results, we use the error norm $J_E = \|Wy - v\|_F$ to compare the convergent speed and the improved convergent precision. The corresponding results are shown in Figures 8 and 9. As seen in Figure 8, synthesized with motion scheme based on ZNN, the convergent rate is gradually catch up the red line, which is obviously slower than the finite-time solving scheme. For clear comparisons, we use J_E in Figure 9 and the convergent rate of finite-time solving scheme reaches to zero in 1 s. The convergent precision is about 2×10^{-4}. For comparisons, Table 3 lists the error of the joint angles by different repeatable motion schemes. The finite-time solving scheme remedies the initial position deviation, that is $\theta_i(9) - \theta_i(0) < 2.644 \times 10^{-5}$. The deviations of the joint angle based on ZNN is around 3.744×10^{-4}, which shows larger convergent precision in joint-drifting phenomenon.

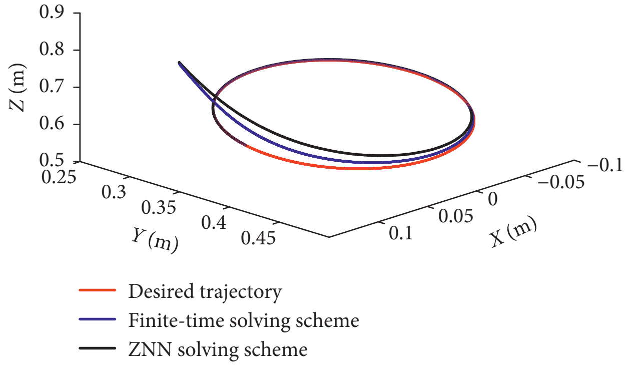


Figure 8: Comparison results of the end-effector when using different motion scheme index to follow a circular path.

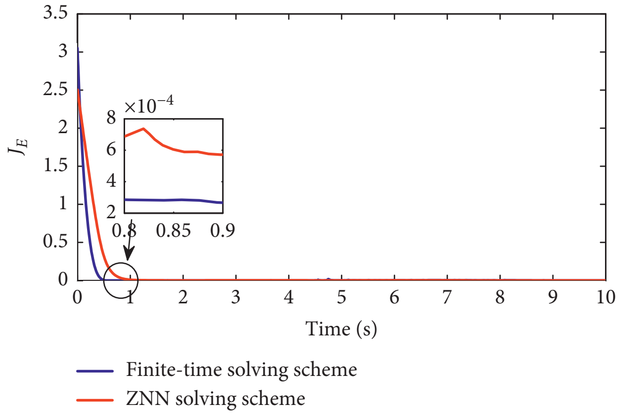


Figure 9: Convergent error produced by ZNN motion scheme (17) and finite-time scheme (12) for circular path.

6. Conclusion

In this paper, a method of solving redundant robot repetitive motion based on neural network has been proposed. The solution for manipulator motion planning not only improves the convergent precision but also accelerates the convergent rate. The motion scheme makes the end-effector of the manipulators return to the desired initial position in finite time, which is more applicable in the practical engineering fields. In addition, theoretical analysis and simulations verify the superiority and timeliness of the proposed method in solving time-varying problems, especially in motion planning of manipulators. However, robustness for the new repeatable motion scheme is not considered in the theoretical analysis, even if instable phenomena may affect the convergent rate. The effect of interference phenomenon for repetitive planning of the redundant manipulators will be considered in the future.

Data Availability

The source code and source data can be provided by contacting with the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was partly supported by the National Natural Science Foundation of China (No. 61803338) and Public Projects of Zhejiang Province (LGG18F020011).

References

[1] J. Zhu, R. Suzuki, T. Tanaka, and Y. Saito, "Automatic tool path generation for robot integrated surface sculpturing system," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 2, no. 4, pp. 812–823, 2008.

[2] W. He, W. Ge, Y. Li, Y.-J. Liu, C. Yang, and C. Sun, "Model identification and control design for a humanoid robot," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 45–57, 2017.

[3] Y. C. Sun and C. C. Cheah, "Region-reaching control for underwater vehicle with onboard manipulator," *IET Control Theory & Applications*, vol. 2, no. 9, pp. 819–828, 2007.

[4] Y. Zhang, P. Xu, and N. Tan, "Further studies on Zhang neural-dynamics and gradient dynamics for online nonlinear equations solving," in *Proceedings of the 2009 IEEE International Conference on Automation and Logistics*, pp. 566–571, Shenyang, China, August 2009.

[5] Y. Zhang, A. Beck, and N. Magnenat-Thalmann, "Human-like behavior generation based on head-arms model for robot tracking external targets and body parts," *IEEE Transactions on Cybernetics*, vol. 45, no. 8, pp. 1390–1400, 2015.

[6] Y. Zhang, C. Yi, D. Guo, and J. Zheng, "Comparison on Zhang neural dynamics and gradient-based neural dynamics for online solution of nonlinear time-varying equation," *Neural Computing and Applications*, vol. 20, no. 1, pp. 1–7, 2011.

[7] C. Yang, Y. Jiang, Z. Li, W. He, and C.-Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1162–1171, 2017.

[8] C. Liegeios-Chauvel, K. Giraud, J. M. Badier, P. Marquis, and P. Chauvel, "Intracerebral evoked potentials in pitch perception reveal a functional asymmetry of the human auditory cortex," *Annals of the New York Academy of Sciences*, vol. 930, pp. 117–132, 2010.

[9] H. Zhou and K.-L. Ting, "Path generation with singularity avoidance for five-bar slider-crank parallel manipulators," *Mechanism and Machine Theory*, vol. 40, no. 3, pp. 371–384, 2005.

[10] K. Tchon and J. Jakubiak, "A repeatable inverse kinematics algorithm with linear invariant subspaces for mobile manipulators," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 5, pp. 1051–1057, 2005.

[11] Y. Zhang, X. Lv, Z. Li, Z. Yang, and K. Chen, "Repetitive motion planning of PA10 robot arm subject to joint physical limits and using LVI-based primal–dual neural network," *Mechatronics*, vol. 18, no. 3, pp. 475–485, 2008.

[12] Y. Zhang and J. Wang, "A dual neural network for constrained joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 5, pp. 654–662, 2002.

[13] Z. Zheng and Y. Zhang, "Design and experimentation of acceleration-level drift-free scheme aided by two recurrent neural networks," *IET Control Theory & Applications*, vol. 7, no. 1, pp. 25–42, 2013.

[14] Y. Zhang, X. Yan, B. Liao, Y. Zhang, and Y. Ding, "Z-type control of populations for Lotka-Volterra model with exponential convergence," *Mathematical Biosciences*, vol. 272, no. 1, pp. 15–23, 2016.

[15] Y. Zhang, X. Yu, Y. Yin, C. Peng, and Z. Fan, "Singularity-conquering ZG controllers of z2g1 type for tracking control of the IPC system," *International Journal of Control*, vol. 87, no. 9, pp. 1729–1746, 2014.

[16] Z. Zhang and Y. Zhang, "Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 2, pp. 674–686, 2013.

[17] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4710–4720, 2017.

[18] S. Li, S. Chen, and B. Liu, "Accelerating a recurrent neural network to finite-time convergence for solving time-varying Sylvester equation by using a sign-Bi-power activation function," *Neural Processing Letters*, vol. 37, no. 2, pp. 189–205, 2013.

[19] D. Chen and Y. Zhang, "Minimum jerk norm scheme applied to obstacle avoidance of redundant robot arm with jerk bounded and feedback control," *IET Control Theory & Applications*, vol. 10, no. 15, pp. 1896–1903, 2016.

[20] D. Guo and Y. Zhang, "Li-function activated ZNN with finite-time convergence applied to redundant-manipulator kinematic control via time-varying Jacobian matrix pseudoinversion," *Applied Soft Computing*, vol. 24, no. 2, pp. 158–168, 2014.

[21] Y. Kong, H.-J. Lu, Y. Xue, and H.-X. Xia, "Terminal neural computing: finite-time convergence and its applications," *Neurocomputing*, vol. 217, no. 12, pp. 133–141, 2016.

[22] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2243–2254, 2017.

[23] L. Jin, Y. Zhang, and S. Li, "Integration-enhanced Zhang neural network for real-time-varying matrix inversion in the presence of various kinds of noises," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 12, pp. 2615–2627, 2016.

[24] L. Jin, S. Li, and B. Hu, "RNN models for dynamic matrix inversion: a control-theoretical perspective," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 189–199, 2018.

[25] Y.-J. Liu and S. Tong, "Optimal control-based adaptive NN design for a class of nonlinear discrete-time block-triangular systems," *IEEE Transactions on Cybernetics*, vol. 46, no. 11, pp. 2670–2680, 2017.