Novel Joint-Drift-Free Scheme at Acceleration Level for Robotic Redundancy Resolution with Tracking Error Theoretically Eliminated

Long Jin, *Member, IEEE*, Zhengtai Xie, Mei Liu, Ke Chen, Chunxu Li and Chenguang Yang

***Abstract*—In this paper, three acceleration-level joint-drift-free (ALJDF) schemes for kinematic control of redundant manipulators are proposed and analyzed from perspectives of dynamics and kinematics with the corresponding tracking error analyses. Firstly, the existing ALJDF schemes for kinematic control of redundant manipulators are systematized into a generalized acceleration-level joint-drift-free (GALJDF) scheme with a paradox pointing out the theoretical existence of the velocity error related to joint drift. Secondly, to remedy the deficiency of the existing solutions, a novel acceleration-level joint-drift-free (NALJDF) scheme is proposed to decouple Cartesian space error from joint space with the tracking error theoretically eliminated. Thirdly, in consideration of the uncertainty at the dynamics level, a multi-index optimization acceleration-level joint-drift-free (MOALJDF) scheme is presented to reveal the influence of dynamics factors on the redundant manipulator control. Afterwards, theoretical analyses are provided to prove the stability and feasibility of the corresponding dynamic neural network (DNN) with the tracking error deduced. Then, computer simulations, performance comparisons and physical experiments on different redundant manipulators synthesized by the proposed schemes are conducted to demonstrate the high performance and superiority of the NALJDF scheme and the influence of dynamics parameters on robot control. This work is of great significance to enhance the product quality and production efficiency in industrial production.**

***Index Terms*—Acceleration-level joint-drift-free (ALJDF) scheme, dynamic neural network (DNN), dynamics level, redundant manipulator.**

I. Introduction

REDUNDANT manipulators, to some extent, perform a greatly significant role in industrial manufacture, on account of more degrees of freedom (DOFs) than nonredundant manipulators in carrying out more complex and difficult tasks with better performance [1]–[3]. Therefore, in order to satisfy the demands of different tasks, various redundant manipulators with special structures have been investigated and manufactured for decades in different fields, such as mobile manipulators [4], parallel manipulators [5] and soft robot arms [6]. In order for redundant manipulators to execute tasks better in practice, a great deal of research is conducted for high performance, accuracy and multi-function. Hence, various properties of the redundant manipulator have been studied and developed, such as robot-environment interaction [7], learning ability [8] and obstacle avoidance [9]. At present, the perfect control of the manipulator still attracts the attention of many researchers. The kinematic control of redundant manipulators at different levels has different characteristics and superiorities. In addition, most of the investigations are mainly conducted at the velocity level, which are not applicable to acceleration and/or torque control for the redundant manipulator in practice. Especially, the majority of velocity-level investigations are not able to deal with the acceleration limit and cannot avoid effectively the instability and divergence states of accelerations, which may cause some damage to the manipulator. Thus, acceleration-level schemes [10]–[15] for kinematic control of redundant manipulators (for example, the minimum-acceleration-norm (MAN) scheme [10], [11]) are very necessary to deal with different tasks.

In recent years, abundant intelligent algorithms and design formulas have been increasingly constructed and applied to various fields, such as image processing [16], robotics [17] and localization [18]. Due to the superior characteristics different from conventional approaches, neural networks are acknowledged as effective models with high performance and have obtained great achievements in electronic and control domains [19]–[22]. As one of the effective networks, the dynamic neural network (DNN) is equipped with the ability to store recursive information thereby reducing the computation complexity, especially for controlling manipulators. Li *et al.* dispose of the redundancy problem of manipulators with high robustness and accuracy by using DNN method, which can deal with the problem of position error accumulation [23]. Moreover, DNNs are employed to improve noise tolerance for redundant manipulators, which can maintain accurate performance when polluted by various noises as reported in [24].

In a large number of mechanical assembly lines, redundant manipulators are required to perform repetitive motions that

This work is supported by the National Natural Science Foundation of China (No. 61703189 and No. 61902131), by the National Key Research and Development Program of China (No. 2017YFE0118900), by the Team Project of Natural Science Foundation of Qinghai Province, China (No. 2020-ZJ-903), by the Key Laboratory of IoT of Qinghai (No. 2020-ZJ-Y16), by the Natural Science Foundation of Gansu Province, China (No. 18JR3RA264), by the Program for Guangdong Introducing Innovative and Entrepreneurial Teams (No. 2017ZT07X183), and also by the Fundamental Research Funds for the Central Universities (No. lzujbky-2019-89, No. lzujbky-2020-it09, and No. 2019MS022). (Corresponding author: Long Jin.)

L. Jin, Z. Xie, and M. Liu are with the School of Information Science and Engineering, Lanzhou University, Lanzhou, China (e-mails: jinlong@lzu.edu.cn; longjin@ieee.org).

K. Chen is with the School of Electronics and Information, South China University of Technology, Guangzhou 510640, China (e-mail: chenk@scut.edu.cn).

C. Li is with Centre for Robotics and Neural Systems, University of Plymouth, Plymouth, PL4 8AA, UK (chunxu.li@plymouth.ac.uk).

C. Yang is with Bristol Robotics Laboratory, University of the West of England, Bristol, BS16 1QY, UK (e-mail: cyang@ieee.org).

may affect the quality and speed of the task execution, to a large extent. In detail, a variety of features (including joint angles, joint velocities, joint accelerations and so on) are supposed to accurately return to their original values [13]. In addition, non-repetitive problems are deemed to generate the joint drift phenomenon, which influences the execution of the task and even can be considered as a failure of the task. However, there are always fluctuant tracking errors whether in simulations or real robotic arms performing repetitive motion tasks [12]–[15], [25]–[28]. Most investigators may have the misconception that these tracking errors in the repetitive motion control of redundant manipulators are invariably attributed to the inevitable calculation errors, external interference and so on. Nevertheless, the coupling relationship between the joint error and Cartesian space error in the existing velocity-level joint-drift-free scheme is first demonstrated and observed in [28]. Therefore, in order to reduce the errors of the existing solutions, accurate control methods for the redundant manipulator need to be researched urgently. To this end, this paper theoretically gives an answer to the existence of the velocity error for the existing acceleration-level joint-drift-free (ALJDF) schemes [12]–[15], which are systematized by the generalized acceleration-level joint-drift-free (GALJDF) scheme. Then, in order to overcome the existing problems, a novel acceleration-level joint-drift-free (NALJDF) scheme is proposed with theoretical analyses provided and tracking error theoretically eliminated. As an extended investigation, considering the robot dynamics level, a multi-index optimization acceleration-level joint-drift-free (MOALJDF) scheme is investigated to show the influence of dynamics parameters on robot control.

The remainder of this paper is constructed as follows. Section II provides the preliminaries and formulates the problem at the acceleration level. Three ALJDF schemes are put forward from different standpoints with the corresponding DNNs derived and theoretical error analyses given in Section III. Moreover, illustrative simulations on different redundant manipulators are conducted synthesized by the proposed ALJDF schemes in Section IV to reveal the feasibility of the three proposed schemes, the disadvantage of the existing ALJDF schemes and the influence of dynamic layer factors on robot control. Then, performance comparisons and physical experiments on different redundant manipulators synthesized by the GALJDF scheme and NALJDF scheme are performed in Section V to clearly demonstrate the accuracy and superior performance of the proposed NALJDF scheme. Conclusions are presented in Section VI.

II. Preliminaries

In this section, the forward kinematics of redundant manipulators, the joint-drift-free (JDF) index and the position error feedback are explained as preliminaries, which formulate the problem at the acceleration level.

A. Forward Kinematics

To lay a foundation for the following analyses, the forward kinematics of a redundant manipulator is given as follows [4], [23], [33]:

$$f(\mathbf{q}) = \mathbf{r}, \tag{1}$$

where $f(\cdot)$ contains the relationship between the trajectory of the end-effector $\mathbf{r} \in \mathbb{R}^n$ in Cartesian space and the joint angle $\mathbf{q} \in \mathbb{R}^m$ in joint space with $n < m$. Generally, it is difficult to directly solve nonlinear system (1) to obtain the redundancy resolution for the kinematic control of redundant manipulator. In contrast, taking the time derivative to both sides of (1) leads to an affine system, which can readily solve the redundancy resolution:

$$J(\mathbf{q})\dot{\mathbf{q}} = \dot{\mathbf{r}}, \tag{2}$$

where $J(\mathbf{q}) = \partial f(\mathbf{q})/\partial \mathbf{q} \in \mathbb{R}^{n\times m}$ stands for the Jacobian matrix of $f(\cdot)$, which depends on the structure of the redundant manipulator; $\dot{\mathbf{q}}$ and $\dot{\mathbf{r}}$ represent the joint velocity and the velocity of the end-effector, respectively. In order to get the relation at the acceleration level, calculating the time derivative of (2) generates

$$J(\mathbf{q})\ddot{\mathbf{q}} = \ddot{\mathbf{r}} - \dot{J}(\mathbf{q})\dot{\mathbf{q}}, \tag{3}$$

where $\ddot{\mathbf{q}}$ and $\ddot{\mathbf{r}}$ stand for the joint acceleration and the end-effector acceleration, respectively; $\dot{J}(\mathbf{q})$ denotes the time derivative of $J(\mathbf{q})$ with its abbreviation $\dot{J}$.

B. JDF Index and Position Error Feedback

In order to figure out the representation of quadratic programming index of JDF at the acceleration level, we devise a joint drift function as follows:

$$\xi_1 = \mathbf{q} - \mathbf{q}_0, \tag{4}$$

where $\mathbf{q}_0 \in \mathbb{R}^m$ stands for initial angle states. Generally, joint drift ξ_1 demonstrates the performance accuracy of the task, to some extent. Moreover, $\mathbf{q}(T)$ is supposed to be equal to $\mathbf{q}_0$ with T being the task execution time. Therefore, ξ_1 should be equal to zero when a task loop ends. According to the neural dynamics method [29], i.e., $\dot{\xi}_1 = -\beta_1\xi_1$ with the design parameter $\beta_1 > 0$, we can simply set

$$\dot{\xi}_1 = -\beta_1\xi_1 = -\beta_1(\mathbf{q} - \mathbf{q}_0). \tag{5}$$

Considering that $\dot{\xi}_1$ equals to $\dot{\mathbf{q}}$, rearranging (5) obtains

$$\dot{\mathbf{q}} + \beta_1(\mathbf{q} - \mathbf{q}_0) = 0. \tag{6}$$

Designing an error equation $\xi_2 = \dot{\mathbf{q}} + \beta_1(\mathbf{q} - \mathbf{q}_0)$, we exploit the neural dynamics method again, i.e., $\dot{\xi}_2 = -\gamma_1\xi_2$ with the design parameter $\gamma_1 > 0$, and obtain

$$\ddot{\mathbf{q}} + \beta_1\dot{\mathbf{q}} = -\gamma_1(\dot{\mathbf{q}} + \beta_1(\mathbf{q} - \mathbf{q}_0)), \tag{7}$$

Then, readjusting (7), one can have

$$\ddot{\mathbf{q}} + (\beta_1 + \gamma_1)\dot{\mathbf{q}} + \gamma_1\beta_1(\mathbf{q} - \mathbf{q}_0) = 0. \tag{8}$$

Based on the above analyses, the representation of the quadratic programming index of repetitive motion at the acceleration level can be designed as $\mathbf{c} = (\beta_1 + \gamma_1)\dot{\mathbf{q}} + \beta_1\gamma_1(\mathbf{q} - \mathbf{q}_0)$ [13], [14].

Similarly, we design $\varepsilon = \mathbf{r} - \mathbf{r}_\mathrm{d}$ with $\mathbf{r}_\mathrm{d}$ being the desired end-effector trajectory. The position error feedback at the acceleration level can be easily derived as $\mathbf{d} = (\beta_2 + \gamma_2)(\dot{\mathbf{r}} -$

$\dot{\mathbf{r}}_{\mathrm{d}}) + \beta_2\gamma_2(\mathbf{r} - \mathbf{r}_{\mathrm{d}})$, with $\beta_2 > 0$ and $\gamma_2 > 0$ standing for the setting parameters.

Remark 1: With regard to the JDF index $\mathbf{c}$ and position error feedback $\mathbf{d}$, necessary descriptions are presented as follows. Similar to equation (8), $\mathbf{d}$ originates from the following relationship

$$\ddot{\mathbf{r}} - \ddot{\mathbf{r}}_{\mathrm{d}} + (\gamma_2 + \beta_2)(\dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}) + \gamma_2\beta_2(\mathbf{r} - \mathbf{r}_{\mathrm{d}}) = 0,$$

which can be regarded as a quadratic constant coefficient differential equation with variable $\varepsilon = \mathbf{r} - \mathbf{r}_{\mathrm{d}}$. In this sense, one has

$$\ddot{\varepsilon} + (\gamma_2 + \beta_2)\dot{\varepsilon} + \gamma_2\beta_2\varepsilon = 0,$$

whose characteristic roots are $\mathcal{R}_1 = \gamma_2 > 0$ and $\mathcal{R}_2 = \beta_2 > 0$. According to Routh-Hurwitz criterion [30], the characteristic roots are in the left of the s plane and thus ε is of global convergence to zero. Analogously, it can be readily obtained $\lim_{t\to\infty} \mathbf{q} = \mathbf{q}_0$ for (8). Therefore, the JDF index $\mathbf{c}$ and position error feedback $\mathbf{d}$ are leveraged to drive errors to converge exponentially [13], [14].

III. Schemes, DNNs and Theoretical Analyses

This section presents three ALJDF schemes from different standpoints with the corresponding DNNs and theoretical error analyses provided.

A. GALJDF Scheme and NALJDF Scheme

In the form of quadratic programming, the GALJDF scheme, which generalizes the existing ALJDF schemes [12]–[15] from a kinematic control point of view, can be written as follows:

$$\begin{aligned}
\text{min.} \quad & \frac{1}{2}\ddot{\mathbf{q}}^{\mathrm{T}}\ddot{\mathbf{q}} + \lambda\mathbf{c}^{\mathrm{T}}\ddot{\mathbf{q}} && \text{(9a)}\\
\text{s.t.} \quad & \ddot{\mathbf{r}}_{\mathrm{a}} = J\ddot{\mathbf{q}} + \delta\mathbf{d} && \text{(9b)}\\
& \ddot{\mathbf{q}} \in \nu && \text{(9c)}\\
\text{with} \quad & \ddot{\mathbf{r}}_{\mathrm{a}} = \ddot{\mathbf{r}}_{\mathrm{d}} - \dot{J}(\mathbf{q})\dot{\mathbf{q}}\\
& \mathbf{c} = (\beta_1 + \gamma_1)\dot{\mathbf{q}} + \beta_1\gamma_1(\mathbf{q} - \mathbf{q}_0)\\
& \mathbf{d} = (\beta_2 + \gamma_2)(\dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}) + \beta_2\gamma_2(\mathbf{r} - \mathbf{r}_{\mathrm{d}}),
\end{aligned}$$

where $\lambda > 0$ denotes the feedback coefficient of the JDF index; $\ddot{\mathbf{r}}_{\mathrm{d}}$ represents the desired acceleration of the end-effector; the superscript $^{\mathrm{T}}$ is the transpose operation of a vector or a matrix. For simplifying the expression, $\ddot{\mathbf{r}}_{\mathrm{d}} - \dot{J}(\mathbf{q})\dot{\mathbf{q}}$ is replaced by an auxiliary variable $\ddot{\mathbf{r}}_{\mathrm{a}}$ in the following derivations. Besides, joint limit (9c) is able to control the joint acceleration in a suitable range to protect redundant manipulators, which is expressed as $\nu = \{a \in \mathbb{R}^n, \nu^- \le a \le \nu^+\}$, where ν^- and ν^+ denote the lower and upper bounds of the acceleration, respectively. In addition, the position error feedback $\delta\mathbf{d}$ is added with $\delta > 0$ being the feedback coefficient.

It is worth noting that the emergence of the repetitive motion index $\mathbf{c}$ couples the Cartesian space error with the joint space error and reduces the accuracy of task execution, which is proved quantitatively below. This deficiency exists in all the existing ALJDF schemes [12]–[15]. In this regard, simply adding an important coefficient $(I - J^{\dagger}J)$ to the GALJDF scheme (9) generates the NALJDF scheme as follows:

$$\begin{aligned}
\text{min.} \quad & \frac{1}{2}\ddot{\mathbf{q}}^{\mathrm{T}}\ddot{\mathbf{q}} + \lambda\mathbf{c}^{\mathrm{T}}(I - J^{\dagger}J)^{\mathrm{T}}\ddot{\mathbf{q}} && \text{(10a)}\\
\text{s.t.} \quad & \ddot{\mathbf{r}}_{\mathrm{a}} = J\ddot{\mathbf{q}} + \delta\mathbf{d} && \text{(10b)}\\
& \ddot{\mathbf{q}} \in \nu && \text{(10c)}\\
\text{with} \quad & \ddot{\mathbf{r}}_{\mathrm{a}} = \ddot{\mathbf{r}}_{\mathrm{d}} - \dot{J}(\mathbf{q})\dot{\mathbf{q}}\\
& \mathbf{c} = (\beta_1 + \gamma_1)\dot{\mathbf{q}} + \beta_1\gamma_1(\mathbf{q} - \mathbf{q}_0)\\
& \mathbf{d} = (\beta_2 + \gamma_2)(\dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}) + \beta_2\gamma_2(\mathbf{r} - \mathbf{r}_{\mathrm{d}}),
\end{aligned}$$

where $(I - J^{\dagger}J)$ is a projection matrix with the superscript † being the pseudo-inverse of a matrix. For further discussion, Remark 2 is supplied to introduce projection matrix $(I - J^{\dagger}J)$. In addition, choosing $\lambda = 0$, GALJDF scheme (9) and NALJDF (10) directly degrade into the MAN scheme as reported in [10]. For convenience, a symbol definition is given: $\mathcal{A} \succeq \mathcal{B}$ represents that each element of matrix $\mathcal{A}$ is greater than or equal to the corresponding element of matrix $\mathcal{B}$.

Remark 2: For projection matrix $(I - J^{\dagger}J)$, some necessary and useful characteristics are given here.

Firstly, considering $JJ^{\dagger} = I$, it can be readily deduced that $J(I - J^{\dagger}J) = J - JJ^{\dagger}J = J - J = 0$. In this regard, projection matrix $(I - J^{\dagger}J)$ has the ability to project Jacobian matrix J into a zero matrix.

Secondly, with regard to the further structural analyses of projection matrix $(I - J^{\dagger}J)$, taking advantage of the singular value decomposition (SVD) method, J can be rewritten as

$$J = \mathcal{S}\Upsilon\mathcal{D}, \tag{11}$$

where $\mathcal{S} \in \mathbb{R}^{n\times n}$ with $\mathcal{S}^{\mathrm{T}}\mathcal{S} = I$; $\Upsilon = [\Upsilon_0 \quad 0] \in \mathbb{R}^{n\times m}$ with $\Upsilon_0 \in \mathbb{R}^{n\times n}$ standing for a diagonal matrix; $\mathcal{D} \in \mathbb{R}^{m\times m}$ with $\mathcal{D}^{\mathrm{T}}\mathcal{D} = I$. In view of $J^{\mathrm{T}} = \mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\mathcal{S}^{\mathrm{T}}$, JJ^{T} can be rearranged as

$$JJ^{\mathrm{T}} = \mathcal{S}\Upsilon\mathcal{D}\mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\mathcal{S}^{\mathrm{T}} = \mathcal{S}\Upsilon\Upsilon^{\mathrm{T}}\mathcal{S}^{\mathrm{T}} = \mathcal{S}(\Upsilon_0^2)\mathcal{S}^{\mathrm{T}}, \tag{12}$$

which is a full rank matrix with all eigenvalues greater than zero. After the matrix inverse operation, one can obtain

$$(JJ^{\mathrm{T}})^{-1} = (\mathcal{S}^{\mathrm{T}})^{-1}(\Upsilon_0^{-2})\mathcal{S}^{-1} = \mathcal{S}\Upsilon_0^{-2}\mathcal{S}^{\mathrm{T}}. \tag{13}$$

Then, $J^{\dagger}$ can be rewritten as

$$J^{\dagger} = J^{\mathrm{T}}(JJ^{\mathrm{T}})^{-1} = \mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\mathcal{S}^{\mathrm{T}}(\mathcal{S}\Upsilon_0^{-2}\mathcal{S}^{\mathrm{T}}) = \mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\Upsilon_0^{-2}\mathcal{S}^{\mathrm{T}}. \tag{14}$$

Further, it is gained

$$J^{\dagger}J = \mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\Upsilon_0^{-2}\mathcal{S}^{\mathrm{T}}\mathcal{S}\Upsilon\mathcal{D} = \mathcal{D}^{\mathrm{T}}\Upsilon^{\mathrm{T}}\Upsilon_0^{-2}\Upsilon\mathcal{D} = \begin{bmatrix} I_{n\times n} & 0 \\ 0 & 0 \end{bmatrix}.$$

Ultimately,

$$I - J^{\dagger}J = \begin{bmatrix} 0 & 0 \\ 0 & I_{(m-n)\times(m-n)} \end{bmatrix} \succeq 0. \tag{15}$$

According to the diagonal elements, $(I - J^{\dagger}J)$ is divided into the first n subsystems and the remaining $m - n$ subsystems. For the first n subsystems, i.e., zero diagonal elements, the NALJDF scheme (10) can be regarded as the MAN scheme for a non-redundant manipulator. Whether the first n joints return

to their original states depends on whether the trajectory of the nth joint is closed. For the remaining $m-n$ subsystems, repetitive motion index $\mathbf{c}$ directly acts on the redundant manipulator to drive the manipulator joints back to their original positions, which indirectly ensures the repeatability of the first n subsystems.

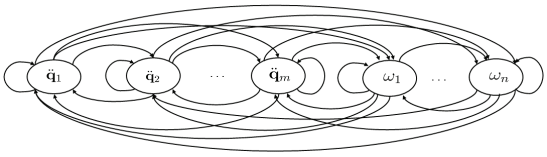


Fig. 1. Uniform structure of DNNs (19), (29) and (43).

B. DNN and Theoretical Analyses for GALJDF Scheme

In this part, with the assistant of the gradient method [31] as well as the acceleration compensation, the corresponding DNN is built with theoretical analyses provided to demonstrate the feasibility of the GALJDF scheme (9).

To start the derivations, we design a 2-norm equation of the velocity error of the end-effector as follows:

$$\dot{\epsilon} = ||\dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}||_2^2/2, \tag{16}$$

where $\dot{\mathbf{r}}_{\mathrm{d}}$ denotes the desired velocity of the end-effector. Taking advantage of the gradient method [31] to minimize the velocity error, equation (16) can be evolved into

$$\ddot{\mathbf{q}} = -\varrho\frac{\partial(\dot{\epsilon})}{\partial\dot{\mathbf{q}}} = \varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}}), \tag{17}$$

where $\varrho > 0$. In view of the joint limit in (9c), (17) can be adjusted as

$$\ddot{\mathbf{q}} = P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}})), \tag{18}$$

where $P_\nu(a) = \arg\min_{b\in\nu}||b-a||$. From the perspective of neural networks, the joint constraints are designed as a bound activation function in this paper as

$$P_\nu(a) = \begin{cases} \nu^-, & a < \nu^- \\ a, & \nu^- < a < \nu^+ \\ \nu^+, & a > \nu^+. \end{cases}$$

Evidently, there exist lagging errors in (18). Recalling the GALJDF scheme (9), it is necessary to present the scheme concretely through the neural network. To this end, an acceleration compensation is deliberately devised within a recursive process. Hence, the DNN corresponding to GALJDF scheme (9) is derived as

$$\ddot{\mathbf{q}} = P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}}) + J^{\mathrm{T}}\omega - \lambda\mathbf{c}), \tag{19a}$$
$$\dot{\omega} = \eta(\ddot{\mathbf{r}}_{\mathrm{a}} - JJ^{\mathrm{T}}\omega - \delta\mathbf{d}), \tag{19b}$$

where $J^{\mathrm{T}}\omega - \lambda\mathbf{c}$ denotes the acceleration compensation, which directly involves the MAN and JDF indices corresponding to (9a); $\eta > 0$ is a design parameter to control the convergence speed; ω stands for the parameter which promotes the recursive system operation and stores the recursive state with its time derivative being $\dot{\omega}$. Furthermore, the uniform structure of the proposed DNNs in this paper is depicted in Fig. 1.

For completing the kinematic control of the manipulator by employing the GALJDF scheme (9) with the assistance of DNN (19), the following theorem is given.

Theorem 1: The optimal solution to the velocity error $\dot{\varepsilon} = \dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}$ generated by the GALJDF scheme (9) assisted with DNN (19) for the redundant manipulator completing the given JDF task globally converges to $-(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c} + J^\dagger\delta\mathbf{d})/\varrho$, given that $J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} \in \mathrm{in}(\nu)$ with $\mathrm{in}(\nu)$ standing for the interior of ν.

Proof: Due to the construction of DNN (19), there are two steps to prove the convergence of DNN (19), that is, the stability of ω and the stability of velocity error $\dot{\varepsilon} = \dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}$.

First step: The stability of ω. It is worth noting that the decisive parameter ω stores recursive states with $\eta > 0$. Therefore, from a mathematical point of view, proving convergence of ω can be translated into proving the positive definiteness of JJ^{T}. Recalling Remark 2, it is evident that full rank matrix JJ^{T} is positive definite. Therefore, $\dot{\omega}$ globally converges to 0 and ω is of convergence to $(JJ^{\mathrm{T}})^{-1}(\ddot{\mathbf{r}}_{\mathrm{a}} - \delta\mathbf{d})$.

Second step: The stability of velocity error $\dot{\varepsilon} = \dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}$. Because of the convergence of ω, considering LaSalle's invariance principle [32], we can obtain $\omega = (JJ^{\mathrm{T}})^{-1}(\ddot{\mathbf{r}}_{\mathrm{a}} - \delta\mathbf{d})$, which can be substituted into (19a), thus obtaining

$$\begin{aligned}\ddot{\mathbf{q}} =& P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}}) + J^{\mathrm{T}}(JJ^{\mathrm{T}})^{-1}(\ddot{\mathbf{r}}_{\mathrm{a}} - \delta\mathbf{d}) - \lambda\mathbf{c}) \\ =& P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}}) + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}).\end{aligned} \tag{20}$$

In consideration of $J\ddot{\mathbf{q}} = \ddot{\mathbf{r}} - \dot{J}\dot{\mathbf{q}}$ and $\ddot{\mathbf{r}}_{\mathrm{a}} = \ddot{\mathbf{r}}_{\mathrm{d}} - \dot{J}\dot{\mathbf{q}}$ in the GALJDF scheme (9) and forward kinematics (3), $\dot{\varepsilon} = \dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}$ can be readjusted as

$$\ddot{\varepsilon} = J(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}). \tag{21}$$

Then, devising the Lyapunov function $V_g = \dot{\varepsilon}^{\mathrm{T}}\dot{\varepsilon}/2$ and its time derivative generates

$$\begin{aligned}\dot{V}_g =& \dot{\varepsilon}^{\mathrm{T}}J(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}) \\ =& -\frac{1}{\varrho}((-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}})^{\mathrm{T}} \\ & \times (P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}).\end{aligned} \tag{22}$$

Given $P_\nu(a) = \arg\min_{b\in\nu}||b-a||$ as well as $||P_\nu(a)-a||^2 \le ||a-b||^2$, $\forall b \in \nu$, designing $a = -\varrho J^{\mathrm{T}}\varepsilon + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}$ and $b = J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}$ generates

$$\begin{aligned}&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c} - P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} \\ &- \lambda\mathbf{c})||^2 \le ||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - \lambda\mathbf{c} - J^\dagger\delta\mathbf{d} - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}||^2,\end{aligned} \tag{23}$$

whose left side can be expanded as

$$\begin{aligned}&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c} - \\ &P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c})||^2 \\ =&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c} - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}||^2 \\ &+||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}||^2 \\ &-2((-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}})^{\mathrm{T}} \\ &\times(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}).\end{aligned} \tag{24}$$

In combination with (23) and (24), we can get

$$\begin{aligned}&||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}||^2 \\ \le&2((-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}})^{\mathrm{T}} \\ &\times(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}).\end{aligned} \tag{25}$$

Putting (22) and (25) together obtains

$$\begin{aligned}\dot{V}_g + \frac{(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})^{\mathrm{T}}}{\varrho}(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda\mathbf{c})\\ -J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}) \le -\frac{1}{2\varrho}||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda\mathbf{c})\\ -J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}||^2 \le 0.\end{aligned} \tag{26}$$

Then, the following proof falls into two cases.

Case I: Devising $\dot{\varepsilon} = -(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho$ and given that $J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} \in \mathrm{in}(\nu)$, we obtain

$$P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} = 0. \tag{27}$$

Thus, it is evident that $\dot{V}_g \le 0$. According to LaSalle's invariance principle [32], we substitute $\dot{V}_g = 0$ into (22) thus generating two conditions.

Condition I: $P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda\mathbf{c}) = J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}$.

Condition II: $-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} = J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}$.

For Condition I, given $J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} \in \mathrm{in}(\nu)$, it is evident to obtain

$$\dot{\varepsilon} = -(J^{\mathrm{T}})^\dagger\frac{\lambda\mathbf{c}+J^\dagger\delta\mathbf{d}}{\varrho}. \tag{28}$$

For Condition II, we directly get $\dot{\varepsilon} = 0$.

In addition, when the given task loop ends, the joint limit and the position error feedback almost equal to zero. In this sense, the solution to Condition I becomes $\dot{\varepsilon} \to 0$. Therefore, Condition II can be generalized into Condition I. Both Condition I and Condition II are considered, which generates the sufficient and necessary condition for $\dot{V}_g = 0$. In summary, it can be easily obtained that $\dot{\varepsilon} = -(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho$.

Case II: As for the condition $\dot{\varepsilon} \neq -(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho$, simply ignoring the joint limit, equation (22) can be rewritten as

$$\dot{V}_g = -\dot{\varepsilon}^{\mathrm{T}}J(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d}+\varrho J^{\mathrm{T}}\dot{\varepsilon}),$$

which can be viewed as a quadratic function with $\dot{\varepsilon}$ being a variable. Evidently, due to the negative quadratic coefficient, we readily obtain that when $-(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho > 0$,

$$\forall\dot{\varepsilon} = \{x \in \mathbb{R}^n, 0 < x < -(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho\},\ \exists\dot{V} > 0,$$

and when $-(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho < 0$,

$$\forall\dot{\varepsilon} = \{x \in \mathbb{R}^n, -(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho < x < 0\},\ \exists\dot{V} > 0.$$

On account of Case II, $\dot{\varepsilon}$ may maintain divergence and increase in an imprecision state.

In conclusion, the optimal solution to the minimum velocity error generated by DNN (19) for the manipulator completing the given task globally converges to $-(J^{\mathrm{T}})^\dagger(\lambda\mathbf{c}+J^\dagger\delta\mathbf{d})/\varrho$, which also implies the existence of the position error and the acceleration error theoretically. The proof is thus completed.

We can generalize from the derivations above that both steady-state and transient-state share the optimal solution to the minimum velocity error (28). Generally, in order to obtain a small velocity error to stabilize the system, ϱ is supposed to be large enough with a small feedback coefficient λ and δ according to (28). Given that λ and $\mathbf{d}$ perform key roles in the completion of the task and a large value of ϱ may put a burden on the machinery, the parameter setting is strictly limited for the GALJDF scheme (9), to some extent.

Remark 3: For the existing solutions to ALJDF schemes [12]–[15], there exists a paradox about the velocity error and the joint drift. To be more specific, increasing the feedback coefficient of the joint drift to obtain the angle repetition is deemed to increase the velocity error of the end-effector. Due to the coupling relationship between the position error and the velocity error, the position error increases synchronously as the joint drift decreases. However, when the position error increases over a certain value and deviates from the desired path, the joint drift becomes severe in an imprecise state. Hence, the selection of parameters is relatively strict and limited for the existing ALJDF schemes [12]–[15].

C. DNN and Theoretical Analyses for NALJDF Scheme

Different from the GALJDF scheme (9), $(I - J^\dagger J)$ is deemed as an important component in the NALJDF scheme (10). Therefore, adding $(I - J^\dagger J)$ in front of the joint drift index generates the DNN corresponding to the NALJDF scheme (10) as

$$\ddot{\mathbf{q}} = P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}) + J^{\mathrm{T}}\omega - \lambda(I-J^\dagger J)\mathbf{c}), \tag{29a}$$

$$\dot{\omega} = \eta(\ddot{\mathbf{r}}_{\mathrm{a}} - JJ^{\mathrm{T}}\omega - \delta\mathbf{d}). \tag{29b}$$

In order to prove the stability and feasibility of DNN (29), we offer the following theorem.

Theorem 2: The velocity error, position error and acceleration error generated by the NALJDF scheme (10) aided with DNN (29) for the manipulator completing the given task globally converge to zero.

Proof: Similarly, the proof can be divided into two parts, which is to prove the stability of ω and the stability of velocity error $\dot{\varepsilon} = \dot{\mathbf{r}} - \dot{\mathbf{r}}_{\mathrm{d}}$. Moreover, the stability of ω is proved in Section III-B. Therefore, we are supposed to prove the stability of velocity error $\dot{\varepsilon}$. The similar derivations are written as

$$\ddot{\mathbf{q}} = P_\nu(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}) + J^\dagger(\ddot{\mathbf{r}}_{\mathrm{a}}-\delta\mathbf{d}) - \lambda(I-J^\dagger J)\mathbf{c}). \tag{30}$$

From NALJDF scheme (10) and forward kinematics (3), we can derive the joint acceleration error $\ddot{\varepsilon} = \ddot{\mathbf{r}} - \ddot{\mathbf{r}}_{\mathrm{d}}$ as

$$\ddot{\varepsilon} = JP_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda(I-J^\dagger J)\mathbf{c}) - \ddot{\mathbf{r}}_{\mathrm{a}}. \tag{31}$$

Then, making use of $JJ^\dagger = I$ and $J(I - J^\dagger J) = 0$ obtains

$$\begin{aligned}\ddot{\varepsilon} =& J(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda(I-J^\dagger J)\mathbf{c})\\ &-J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}+\lambda(I-J^\dagger J)\mathbf{c}).\end{aligned}$$

Devise the Lyapunov function $V_n = \dot{\varepsilon}^{\mathrm{T}}\dot{\varepsilon}/2$ and its time derivative generates

$$\begin{aligned}\dot{V}_n =& \dot{\varepsilon}^{\mathrm{T}}\ddot{\varepsilon}\\ =& \dot{\varepsilon}^{\mathrm{T}}J(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda(I-J^\dagger J)\mathbf{c})\\ &-J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}+\lambda(I-J^\dagger J)\mathbf{c})\\ =& -\frac{1}{\varrho}((-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda(I-J^\dagger J)\mathbf{c})\\ &+J^\dagger\delta\mathbf{d}-J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}+\lambda(I-J^\dagger J)\mathbf{c})^{\mathrm{T}}\\ &\times(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}+J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}-J^\dagger\delta\mathbf{d}-\lambda(I-J^\dagger J)\mathbf{c})\\ &-J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}+\lambda(I-J^\dagger J)\mathbf{c}).\end{aligned} \tag{32}$$

In consideration of $P_\nu(a) = \arg\min_{b\in\nu}||b-a||$ as well as $||P_\nu(a)-a||^2 \leq ||a-b||^2$, $\forall b \in \nu$, devising $a = -\varrho J^{\mathrm{T}}\varepsilon + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c}$ and $b = J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c}$ generates

$$\begin{aligned}&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c} - P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon}\\ &+ J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})||^2 \leq ||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}}\\ &- J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c} - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}||^2,\end{aligned} \tag{33}$$

whose left side can be extended as

$$\begin{aligned}&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c} -\\ &P_\nu(-\varrho J^{\mathrm{T}}\varepsilon + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})||^2\\ =&||-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c}\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}||^2 +\\ &||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}||^2\\ &- 2((-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c})^{\mathrm{T}}\times\\ &(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}).\end{aligned} \tag{34}$$

In combination with (33) and (34), we have

$$\begin{aligned}&||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}||^2\\ \leq&2((-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c})^{\mathrm{T}}\times\\ &(P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d} - \lambda(I-J^\dagger J)\mathbf{c})\\ &- J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}).\end{aligned} \tag{35}$$

Then, combining (32) and (35) together obtains

$$\begin{aligned}&\dot{V}_n + \dot{\varepsilon}^{\mathrm{T}}JJ^\dagger\delta\mathbf{d} \leq -\frac{1}{2\varrho}||P_\nu(-\varrho J^{\mathrm{T}}\dot{\varepsilon} + J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} - J^\dagger\delta\mathbf{d}\\ &- \lambda(I-J^\dagger J)\mathbf{c}) - J^\dagger\ddot{\mathbf{r}}_{\mathrm{a}} + J^\dagger\delta\mathbf{d} + \lambda(I-J^\dagger J)\mathbf{c}||^2 \leq 0.\end{aligned}$$

According to (4) through (8), $\dot{\varepsilon}$ and $\mathbf{d}$ are designed as follows:

$$\mathbf{d} = (\beta_2+\gamma_2)\dot{\varepsilon} + \beta_2\gamma_2\varepsilon = -\ddot{\varepsilon}, \tag{36}$$

and

$$\dot{\varepsilon} = -\beta_2\varepsilon. \tag{37}$$

Then, it is evident to deduce

$$\mathbf{d} = -\beta_2(\beta_2+\gamma_2)\varepsilon + \beta_2\gamma_2\varepsilon, \tag{38}$$

and further obtain

$$\dot{\varepsilon}^{\mathrm{T}}JJ^\dagger\delta\mathbf{d} = \dot{\varepsilon}^{\mathrm{T}}\delta\mathbf{d} = \beta_2^2(\beta_2+\gamma_2)\varepsilon^{\mathrm{T}}\varepsilon - \beta_2^2\gamma_2\varepsilon^{\mathrm{T}}\varepsilon = \gamma_2^3\varepsilon^{\mathrm{T}}\varepsilon \geq 0.$$

In this sense, we have

$$\dot{V}_n \leq -\dot{\varepsilon}^{\mathrm{T}}JJ^\dagger\delta\mathbf{d} \leq 0. \tag{39}$$

Based on LaSalle's invariance principle [32], we design $\dot{V}_n = 0$ and it should also be satisfied that $\dot{\varepsilon}^{\mathrm{T}}JJ^\dagger\delta\mathbf{d} = 0$, which generates $\lim_{t\to+\infty}\dot{\varepsilon} = 0$ or $\lim_{t\to+\infty}\ddot{\varepsilon} = 0$. Further, recalling (36) and (37), one can come to the conclusion that $\lim_{t\to+\infty}\dot{\varepsilon} = 0$ and $\lim_{t\to+\infty}\ddot{\varepsilon} = 0$ are equivalent conditions and readily obtain

$$\lim_{t\to+\infty}\ddot{\varepsilon} = \lim_{t\to+\infty}\dot{\varepsilon} = \lim_{t\to+\infty}\varepsilon = 0. \tag{40}$$

Evidently, condition $\dot{V}_n = 0$ holds, if and only if $\dot{\varepsilon} = 0$. Hence, (40) is clearly a necessary and sufficient condition. Therefore, position error ε, velocity error $\dot{\varepsilon}$ and acceleration error $\ddot{\varepsilon}$ all converge to zero. The proof is thus completed.

D. Proposal of MOALJDF Scheme

It is worth pointing out that, generally speaking, numerous schemes [12]–[15] for redundancy resolution of redundant manipulator at the acceleration level are usually modeled, analyzed and controlled from a kinematic point of view as the GALJDF scheme (9) and NALJDF scheme (10) depict. However, in reality, the uncertainty in robot dynamics does affect the accuracy of robot control. Therefore, as an extended investigation, an MOALJDF scheme is designed and constructed to take both robot kinematics and dynamics into account as follows:

$$\begin{aligned}\text{min.}\quad & \alpha(\frac{1}{2}\ddot{\mathbf{q}}^{\mathrm{T}}\ddot{\mathbf{q}} + \lambda\mathbf{c}^{\mathrm{T}}(I-J^\dagger J)^{\mathrm{T}}\ddot{\mathbf{q}}) + \zeta\frac{1}{2}\tau^{\mathrm{T}}\tau && \text{(41a)}\\ \text{s.t.}\quad & \ddot{\mathbf{r}}_{\mathrm{a}} = J\ddot{\mathbf{q}} + \delta\mathbf{d} && \text{(41b)}\\ & \ddot{\mathbf{q}} \in \nu && \text{(41c)}\\ \text{with}\quad & \ddot{\mathbf{r}}_{\mathrm{a}} = \ddot{\mathbf{r}}_{\mathrm{d}} - \dot{J}\dot{\mathbf{q}}\\ & \tau = M\ddot{\mathbf{q}} + \mathbf{p}(\dot{\mathbf{q}},\mathbf{q}) + \mathbf{g}(\mathbf{q})\\ & \mathbf{c} = (\beta_1+\gamma_1)\dot{\mathbf{q}} + \beta_1\gamma_1(\mathbf{q}-\mathbf{q}_0)\\ & \mathbf{d} = (\beta_2+\gamma_2)(\dot{\mathbf{r}}-\dot{\mathbf{r}}_{\mathrm{d}}) + \beta_2\gamma_2(\mathbf{r}-\mathbf{r}_{\mathrm{d}}),\end{aligned}$$

where $\tau \in \mathbb{R}^m$ denotes the joint torque; $M \in \mathbb{R}^{m\times m}$ stands for the inertial matrix; $\mathbf{p}(\dot{\mathbf{q}},\mathbf{q}) \in \mathbb{R}^m$ symbolizes the centrifugal force vector; $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^m$ represents the gravity vector; $\alpha \in (0,\ 1)$ and $\zeta \in (0,\ 1)$ are the weight parameters of dynamics and kinematics, respectively, with $\alpha+\zeta = 1$. It can be readily observed that the MOALJDF scheme (41) incorporates the motion control indices of the minimum acceleration norm [10], repetitive motion planning [28] and minimum torque norm [34], which involves both kinematics and dynamics simultaneously. Correspondingly, from (19a), it can be naturally designed that

$$\ddot{\mathbf{q}} = \varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}} - \dot{\mathbf{r}}) + \alpha(J^{\mathrm{T}}\omega - \lambda\mathbf{c}) - \zeta M\tau, \tag{42}$$

which can be simplified as

$$\ddot{\mathbf{q}} = (I+\zeta MM)^{-1}(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}) + \alpha(J^{\mathrm{T}}\omega - \lambda\mathbf{c}) - \zeta M(\mathbf{p}+\mathbf{g})).$$

Further, imposing joint constraints to (42) and combining the recursive process (19b) lead to

$$\begin{aligned}\ddot{\mathbf{q}} =& P_\nu((I+\zeta MM)^{-1}(\varrho J^{\mathrm{T}}(\dot{\mathbf{r}}_{\mathrm{d}}-\dot{\mathbf{r}}) + \alpha(J^{\mathrm{T}}\omega - \lambda\mathbf{c})\\ &- \zeta M(\mathbf{p}+\mathbf{g}))), && \text{(43a)}\\ \dot{\omega} =& \eta(\ddot{\mathbf{r}}_{\mathrm{a}} - JJ^{\mathrm{T}}\omega - \delta\mathbf{d}), && \text{(43b)}\end{aligned}$$

which is the DNN corresponding to the MOALJDF scheme (41). As for the stability and error analysis of the system, similar to the proof to Theorem 1, one can simply imply that the DNN (43) is of great convergence globally and that the

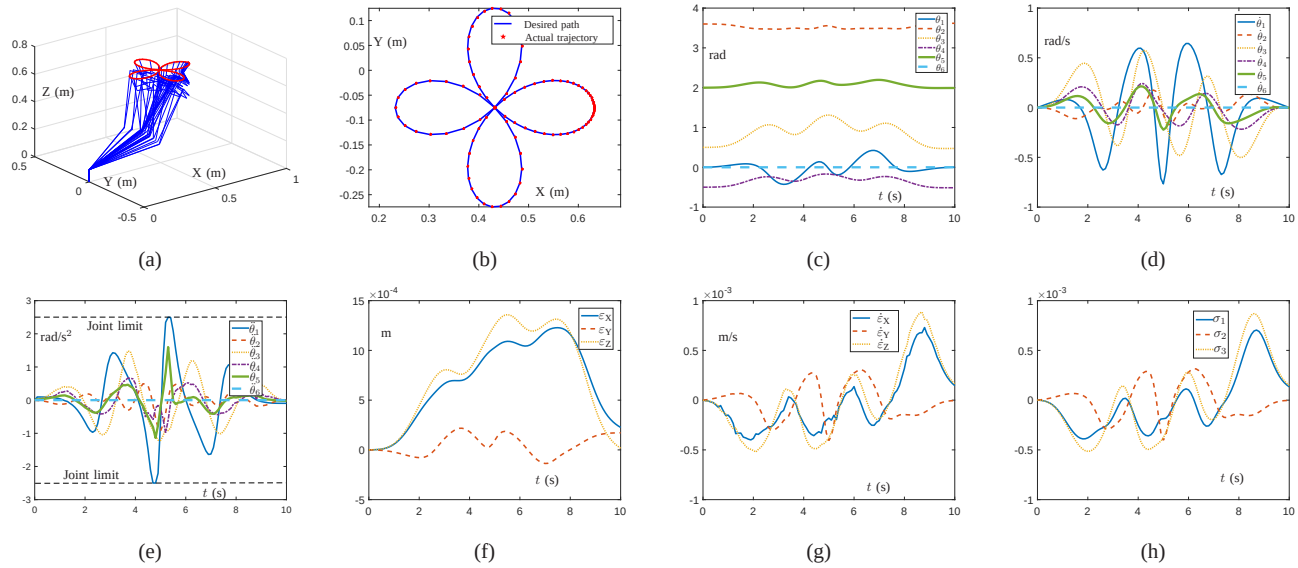


Fig. 2. Computer simulations on employing the GALJDF scheme (9) to control the UR5 manipulator for tracking a four-leaf clover path aided with DNN (19). (a) Motions of redundant manipulator. (b) Desired path and end-effector trajectory. (c) Time history of joint angle. (d) Time history of joint velocity. (e) Time history of joint acceleration with joint limit. (f) Time history of position error. (g) Time history of velocity error. (h) Time history of $\sigma = -(J^{\mathrm{T}})^{\dagger}(\lambda \mathbf{c} + J^{\dagger}\delta \mathbf{d})/\varrho$.

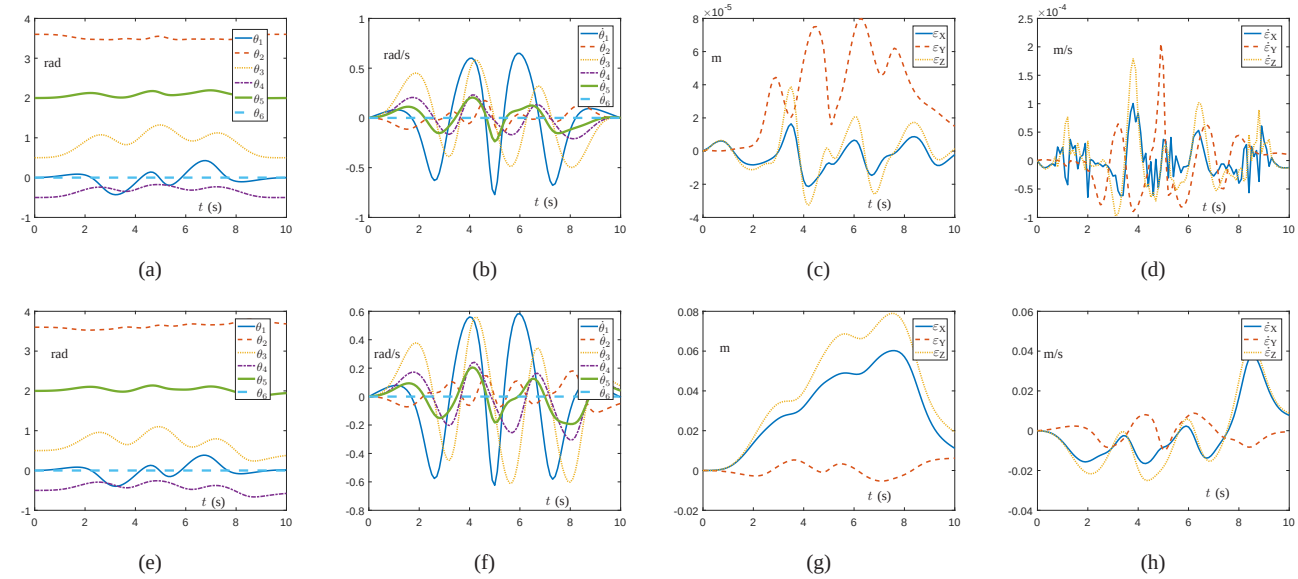


Fig. 3. Computer simulations on employing the GALJDF scheme (9) and the NALJDF scheme (10) to control the UR5 manipulator for tracking a four-leaf clover path aided with DNN (19) and DNN (29). (a) Time history of joint angle synthesized by DNN (29). (b) Time history of joint velocity synthesized by DNN (29). (c) Time history of position error synthesized by DNN (29). (d) Time history of velocity error synthesized by DNN (29). (e) Time history of joint angle synthesized by DNN (19). (f) Time history of joint velocity synthesized by DNN (19). (g) Time history of position error synthesized by DNN (19). (h) Time history of velocity error synthesized by DNN (19).

velocity error generated by the DNN (43) is positively related to the uncertainty of dynamics parameters. Due to the space limitation, the specific derivations are omitted, which can be referred to the proof to Theorem 1.

IV. Illustrative Simulations

This section provides multiple simulations based on different redundant manipulators synthesized by the GALJDF scheme (9), the NALJDF scheme (10) and the MOALJDF scheme (41) with the assistance of DNN (19), DNN (29) and DNN (43). In what follows, the used redundant manipulators include the UR5 manipulator, 6-DOFs planar robot and KUKA manipulator with structural parameters and dynamics parameters referred to [33]–[35]. As far as the fixed parameters are concerned, it is selected as $\beta_1 = \gamma_1 = \beta_2 = \gamma_2 = 1$, $\eta = 1000$ and $\delta = 250$ with the task execution time $T = 10$ s for Section IV and Section V.

A. GALJDF Scheme on Redundant Manipulator

In this part, the GALJDF scheme (9) is employed on the UR5 manipulator tracking a four-leaf clover path with the aid

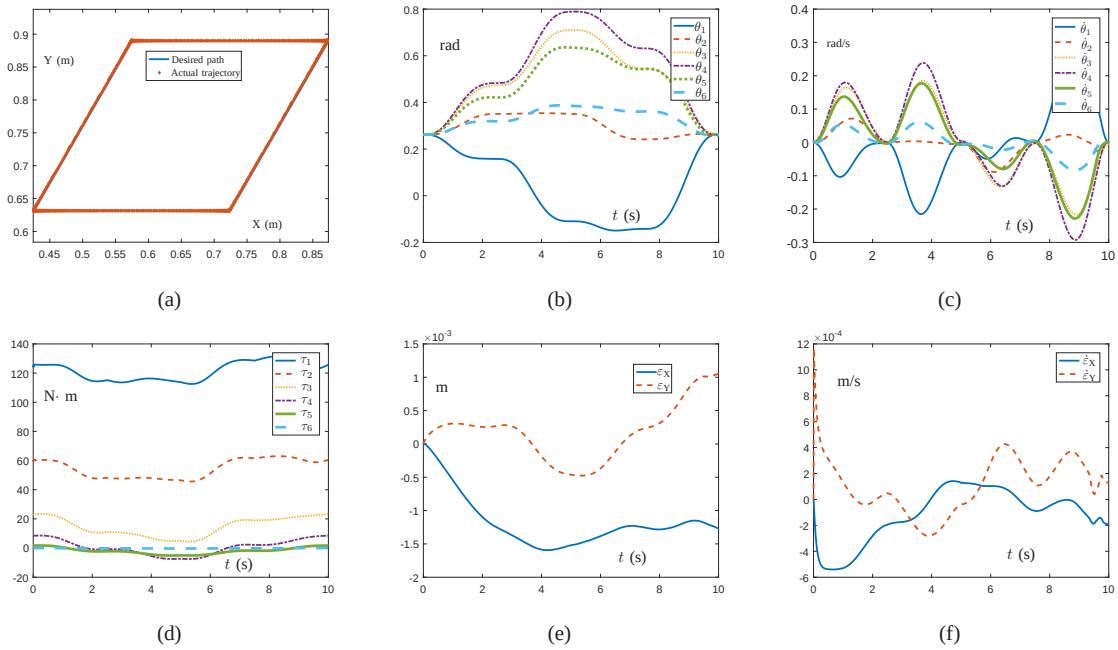


Fig. 4. Computer simulations on employing the MOALJDF scheme (41) aided with DNN (43) to control a 6-DOFs redundant manipulator for tracking a parallelogram path. (a) Desired path and actual trajectory. (b) Time history of joint angle. (c) Time history of joint velocity. (d) Time history of joint torque. (e) Time history of position error. (f) Time history of velocity error.

of DNN (19). Parameter settings are $\lambda = 3$ and $\varrho = 20000$. In addition, the initial angle state $\mathbf{q}_0 = [0,\ 3.6,\ 0.5,\ -0.5,\ 2,\ 0]^{\mathrm{T}}$ rad and the joint limit is $\nu^+ = -\nu^- = 2.5$ rad/s^2. In Fig. 2(a) and (b), it is concluded that the given task is completed successfully with the actual trajectory coinciding with the desired trajectory. The generated joint angle in Fig. 2(c) and the joint velocity in Fig. 2(d) all return to their original states. As Fig. 2(e) plots, the joint limit is activated with the acceleration fluctuation. Afterwards, the position error shown in Fig. 2(f) sustains increasing with the maximum error less than 1.5×10^{-3} m and the velocity error is of the order 10^{-3} m/s as shown in Fig. 2(g). It is worth noting that the curve of $\sigma = -(J^{\mathrm{T}})^{\dagger}(\lambda \mathbf{c} + J^{\dagger}\delta \mathbf{d})/\varrho$ in Fig. 2(h) and the velocity error in Fig. 2(g) have similar or even identical values and trends, which verifies the correctness of the theoretical derivation as deduced in (28). These results embody the stability and feasibility of the GALJDF scheme (9) aided with DNN (19).

B. NALJDF Scheme on Redundant Manipulator

In this part, for comparison, the NALJDF scheme (10) and the GALJDF scheme (9) are respectively conducted on the UR5 manipulator with the aid of DNNs (29) and (19) for tracking a four-leaf clover path. To prove the paradox in Remark 3, we devise a large joint drift feedback coefficient $\lambda = 10$ and a small velocity error feedback coefficient $\varrho = 2000$. Simulation results are presented in Fig. 3. It is worth noting that Fig. 3(a) through (d) are synthesized by the NALJDF scheme (10) and Fig. 3(e) through (h) are synthesized by the GALJDF scheme (9). From Fig. 3(a) and (b), it is evident that the UR5 manipulator successfully completes the given task with the angle and the joint velocity back to their original states. Moreover, the velocity error is of the order 10^{-4} m/s with the maximum about 2.0×10^{-4} m/s and the position error is of the order 10^{-5} m with the maximum less than 8×10^{-5} m in an accurate state as shown in Fig. 3(c) and (d), respectively. As a comparison, for the GALJDF scheme (9), we discover that Fig. 3(e) through (h) are all in an imprecise state which is deemed to fulfill the given task unsuccessfully. As Fig. 3(g) and (h) illustrate, the velocity error and the position error are too large and are of the order 10^{-2}. Due to the large position error and the velocity error, the trajectory deviates from the desired path, which causes large joint drift as indicated in Fig. 3(e) and (f) with joint angles and joint velocities not returning to their original states. It is evident that under the design parameter settings, the GALJDF scheme (9) fails to complete the given task. The above comparison simulations indicate the correctness of the paradox in Remark 3 of the existing ALJDF scheme [12]–[15] as well as the superior and accurate performance of the NALJDF scheme (10).

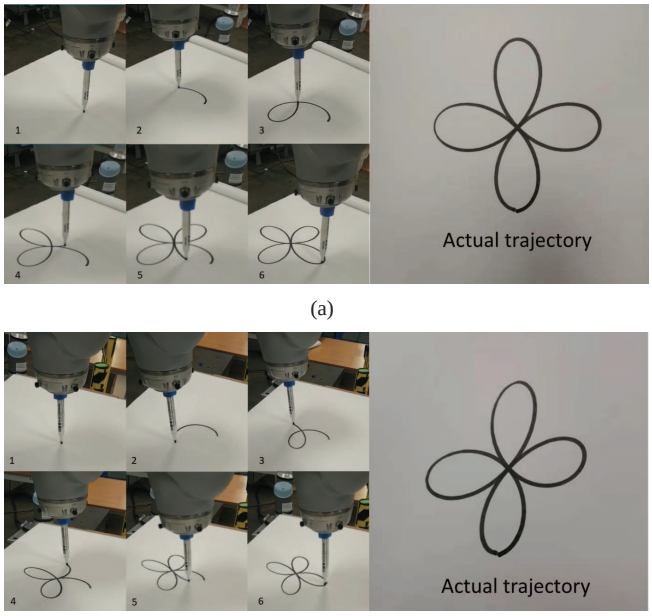
C. MOALJDF Scheme on Redundant Manipulator

It is worth pointing out that in some of the existing literatures on robot motion planning [12]–[15], [25]–[28], redundant robot manipulators are modeled, analyzed and controlled only from the perspective of kinematics. Nevertheless, it is beyond all questions that the uncertainty in robot dynamics indeed affects the accuracy of robot control. As a result, simultaneously considering robot dynamics and kinematics is relatively comprehensive, which is much closer to real-world applications. Therefore, the simulations driven by the MOALJDF scheme (41) aided with DNN (43) are conducted based on a 6-DOFs planar redundant manipulator, whose structural parameters and dynamics parameters can be referred to [34], for JDF task with results plotted in Fig. 4. In addition,

TABLE I
COMPARISONS AMONG GALJDF SCHEME AND NALJDF SCHEME FOR CONTROLLING UR5 WITH DIFFERENT COEFFICIENTS

(ϱ=10^5)	The GALJDF scheme (9) as a systematization of [12]–[15]			The NALJDF scheme (10)		
λ	MPE* (m)	MVE* (m/s)	JD$^\circ$ (rad)	MPE* (m)	MVE* (m/s)	JD$^\circ$ (rad)
0.1	5.14×10^{-5}	1.68×10^{-4}	5.37×10^{-2}	4.06×10^{-5}	1.06×10^{-4}	5.37×10^{-2}
0.5	2.77×10^{-4}	2.45×10^{-4}	9.10×10^{-3}	4.46×10^{-5}	1.70×10^{-4}	9.10×10^{-3}
1	5.59×10^{-4}	4.54×10^{-4}	2.80×10^{-3}	4.43×10^{-5}	1.09×10^{-4}	2.90×10^{-3}
2	1.10×10^{-3}	9.44×10^{-4}	1.60×10^{-3}	4.41×10^{-5}	1.62×10^{-4}	1.62×10^{-3}
3	1.70×10^{-3}	1.30×10^{-3}	6.84×10^{-4}	4.41×10^{-5}	1.01×10^{-4}	7.26×10^{-4}
5	2.80×10^{-3}	2.20×10^{-3}	3.74×10^{-4}	4.41×10^{-5}	1.05×10^{-4}	1.38×10^{-4}
10	5.50×10^{-3}	4.40×10^{-3}	6.37×10^{-4}	4.42×10^{-5}	1.10×10^{-4}	1.03×10^{-4}
50	2.46×10^{-2}	1.91×10^{-2}	2.80×10^{-3}	4.41×10^{-5}	1.31×10^{-4}	2.57×10^{-4}
100	4.42×10^{-2}	3.27×10^{-2}	5.20×10^{-3}	4.42×10^{-5}	1.27×10^{-4}	2.75×10^{-4}

* The 'MPE' stands for the maximum position error in time history.
* The 'MVE' stands for the maximum velocity error in time history.
$^\circ$ The 'JD' stands for the mean of the elements of vector $|\mathbf{q}_0 - \mathbf{q}(T)|$ with task cycle duration $T = 10$ s.



(a)

(b)

Fig. 5. Snapshots of the physical experiments on the KUKA redundant manipulator for tacking a four-leaf clover path. (a) Physical experiments synthesized by NALJDF scheme (10) aided with DNN (29). (b) Physical experiments synthesized by GALJDF scheme (9) aided with DNN (19).

we select $\alpha = 0.995$, $\zeta = 1 - \alpha = 0.005$, original state $\theta_0 = [\pi/12]_{6\times1}$ rad, $\lambda = 3$, and $\varrho = 10^4$. One can readily obtain from Fig. 4(a) through (c) that the given JDF task is well accomplished with joint velocity and the joint angle back to their initial states. Furthermore, joint torques vary within a reasonable range in Fig. 4(d). It is worth noting that the initial values of joint torque are nonzero, which does affect the generated error and acceleration. As depicted in Fig. 4(e) and (f), the velocity error is large at 0 s and then gradually approaches zero with position error less than 2×10^{-3} m. From the above results, we can draw a general conclusion that the unknown parameter variables in kinematics can affect the execution accuracy of the task and that the designed DNN (43) with great convergence can reduce the affected error as much as possible, thus getting a relatively precise state.

V. PHYSICAL EXPERIMENTS AND COMPARISONS

For further comparison with the GALJDF scheme (9) and the NALJDF scheme (10), plenty of simulations for the two ALJDF schemes on UR5 manipulator tracking a four-leaf clover path are conducted with different joint drift feedback coefficients in Table I. For other parameter settings, we design $\varrho = 10^5$ and the initial angle state $\mathbf{q}_0 = [0,\ 3.6,\ 0.5,\ 0.5,\ 2,\ 0]^{\mathrm{T}}$ rad with different λ. It is worth highlighting that the GALJDF scheme (9) is the systematization of the existing ALJDF schemes [12]–[15]. Observing Table I, it is evident that as the coefficient λ increases, the velocity error and the position error increase synchronously with accuracy reduction which reduces the quality of the task for the GALJDF scheme (9). Meanwhile, the joint drift starts to decrease when λ gets a small value and then increases, which coincides with Remark 3. Nevertheless, the NALJDF scheme (10) demonstrates a better performance by comparison. It is easy to discover that the position error and the velocity error maintain accurate which are of the order 10^{-5} m and 10^{-4} m/s all the time with different joint drift feedback coefficients, which exhibits the decoupled relationship as depicted in Theorem 2. In addition, the joint drift decreases as the coefficient λ increases, which becomes more and more accurate. By comparison, performance analyses on the two ALJDF schemes depict that the GALJDF scheme (9) is only suitable for a small range of joint drift feedback, that the NALJDF scheme (10) is not limited by the coefficients to keep a precise state, and that for the same parameter setting, the NALJDF scheme (10) can achieve more accurate performance with tiny tracking error, which gives full expressions to the superiority of the proposed NALJDF scheme (10).

To illustrate the performance comparisons of the GALJDF scheme (9) and the NALJDF scheme (10) more vividly and convincingly, physical experiments on the KUKA redundant manipulator, which is equipped with 7-DOFs and a marking pen attached to its end-effector, for tracking a four-leaf clover path are performed driven by the DNN (29) and DNN (19). Specifically speaking, the desired trajectory signals ($\mathbf{r}_{\mathrm{d}}$, $\dot{\mathbf{r}}_{\mathrm{d}}$ and $\ddot{\mathbf{r}}_{\mathrm{d}}$) are set up in advance; the status signals of the KUKA manipulator ($\mathbf{q}$, $\dot{\mathbf{q}}$, $\mathbf{r}$ and $\dot{\mathbf{r}}$) are measured in real time from the KUKA manipulator; the structure signals (J and $\dot{J}$) are

TABLE II
COMPARISONS AMONG DIFFERENT NEURAL-NETWORK-BASED METHODS FOR ELIMINATING JOINT DRIFT OF REDUNDANT MANIPULATORS

	Scheme level	Repetitive motion	Error compensation	Joint limits	Joint drift	Error decoupling	Theoretical tracking error
Method (9)	Acceleration	Yes	Yes	Yes	Non-zero	No	Non-zero
Method (10)	Acceleration	Yes	Yes	Yes	Zero	Yes	Zero
Method in [12]	Acceleration	Yes	No	Yes	Non-zero	No	Non-zero
Method in [13]	Acceleration	Yes	No	Yes	Non-zero	No	Non-zero
Method in [14]	Acceleration	Yes	No	Yes	Non-zero	No	Non-zero
Method in [15]	Acceleration	Yes	No	Yes	Non-zero	No	Non-zero
Method in [25]	Velocity	Yes	No	Yes	Non-zero	No	Non-zero
Method in [26]	Velocity	Yes	No	No	Non-zero	No	Non-zero
Method in [27]	Velocity	Yes	No	No	Non-zero	No	Non-zero
Method in [28]	Velocity	Yes	Yes	Yes	Non-zero	No	Non-zero

generated by calculation. On this basis, the DNN (29) and DNN (19) continuously generate the acceleration commands to control the KUKA manipulator in real time with the tracking error reduced by the position error feedback. In addition, we design $\lambda = 30$ and $\varrho = 2000$, and the snapshots of the experiment results are portrayed in Fig. 5(a) and (b). It is deserved to point out that as demonstrated in Fig. 5(a), the KUKA redundant manipulator driven by NALJDF scheme (10) achieves excellent performance with tiny error and thus the practicability is verified. On the contrary, the GALJDF scheme (9) embodies poor performance as illustrated in Fig. 5(b). It is evident that the asymmetric trajectory deviates from the desired path in Fig. 2(b) with a large tracking error. Furthermore, these undesired results in Fig. 5(b) verify the deficiency of the existing ALJDF schemes [12]–[15] that the large feedback coefficient of joint drift can lead to the decline of end-effector task precision.

Beyond that, comparisons between the existing neural-network-based JDF methods and the proposed scheme (10) are presented with results supplied in Table II. Different methods to solve the joint drift problem have their own characteristics from different aspects. In comparison with the existing neural-network-based methods for eliminating joint drift of redundant manipulators [12]–[15], [25]–[28], the proposed NALJDF method (10) has a great advantage in error analyses and error eliminations. On this basis, the proposed NALJDF method (10) is the first to decouple joint space errors and Cartesian space errors at the acceleration level and further eliminates the joint drift and position error, theoretically.

VI. CONCLUSIONS

Base on the robot kinematics, this paper has generalized the existing ALJDF schemes by presenting a GALJDF scheme and pointed out the deficiencies of the existing ALJDF schemes as a paradox about the theoretical existence of the joint velocity error. As an improvement, an NALJDF scheme has been proposed with the theoretical tracking error eliminated. Then, as an exploratory investigation, the uncertainty of the robot dynamics has been researched with the MOALJDF scheme provided. Afterward, the acceleration compensation has been leveraged to assist the gradient method in completing the construction of the corresponding DNNs, whose feasibility and stability have been provided by theoretical analyses. Moreover, computer simulations and physical experiments on employing the proposed schemes to control different redundant manipulators to track the given task have been conducted, which demonstrates the feasibility of the proposed schemes and the correctness of the theoretical analyses. In addition, comparison results among the existing approaches and the proposed methods have been provided to verify the precise performance and superiority of the proposed NALJDF scheme.

REFERENCES

[1] C. Yang, C. Zeng, C. Fang, W. He, and Z. Li, "A DMPs-based framework for robot learning and generalization of humanlike variable impedance skills," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1193–1203, Jun. 2018.

[2] R. Li and H. Qiao, "A survey of methods and strategies for high-precision robotic grasping and assembly tasks-some new trends," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2718–2732, Dec. 2019.

[3] Z. Li, S. Zhao, J. Duan, C.-Y. Su, C. Yang, and X. Zhao, "Human cooperative wheelchair with brain machine interaction based on shared control strategy," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 1, pp. 185–195, Feb. 2017.

[4] L. Xiao and Y. Zhang "A new performance index for the repetitive motion of mobile manipulators," *IEEE Trans. Cybern.*, vol. 44, no. 2, pp. 280–292, Feb. 2014.

[5] J. Li, Y. Zhang, S. Li, and M. Mao, "New discretization-formula-based zeroing dynamics for real-time tracking control of serial and parallel manipulators," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3416–3425, Aug. 2018.

[6] F. Xu, H. Wang, K. W. S. Au, W. Chen, and Y. Miao, "Underwater dynamic modeling for a cable-driven soft robot arm," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2726–2737, Dec. 2018.

[7] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot-environment interaction," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2568–2579, Jul. 2019.

[8] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 777–787, Mar. 2019.

[9] D. Guo and Y. Zhang, "A new inequality-based obstacle-avoidance MVN scheme and its application to redundant robot manipulators," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 1326–1340, Nov. 2012.

[10] D. Guo and Y. Zhang, "Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 6903–6914, Dec. 2014.

[11] D. Guo, and Y. Zhang, "Simulation and experimental verification of weighted velocity and acceleration minimizationfor robotic redundancy resolution," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 4, Oct. 2014.

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

[13] L. Jin and Y. Zhang, "G2-type SRMPC scheme for synchronous manipulation of two redundant robot manipulators," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 153–164, Feb. 2015.

[14] Z. Zhang and Y. Zhang, "Acceleration-level cyclic-motion generation of constrained redundant robots tracking different paths," *IEEE Trans. Syst., Man, Cybern. B*, vol. 42, no. 4, pp. 1257–1269, Aug. 2012.

[15] L. Xiao and Y. Zhang, "Acceleration-level repetitive motion planning and its experimental verification on a six-link planar robot manipulator," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 3, pp. 906–914, May 2013.

[16] X. Huang, H. Qiao, B. Zhang, and X. Nie, "Supervised polarimetric SAR image classification using tensor local discriminant embedding," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2966–2979, Jun. 2018.

[17] J. Chen and H. Qiao, "Muscle-synergies-based neuromuscular control for motion learning and generalization of a musculoskeletal system," *IEEE Trans. Syst., Man, Cybern., Syst.*, In Press with DOI 10.1109/TSMC.2020.2966818.

[18] L. Jin, J. Yan, X. Du, X. Xiao, and D. Fu, "RNN for solving time-variant generalized Sylvester equation with applications to robots and acoustic source localization," *IEEE Trans. Ind. Informat.*, In Press with 10.1109/TII.2020.2964817.

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

[20] C. Huang, X. Wang, L. Li, and X. Chen, "Multi-structure radial basis function neural networks based extended model predictive control: Application to clutch control," *IEEE/ASME Trans. Mechatronics*, In Press with DOI 10.1109/TMECH.2019.2949001.

[21] Y. Cai, Y. Wang, H. Xu, S. Sun, C. Wang, and L. Sun, "Research on rotor position model for switched reluctance motor using neural network," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2762–2773, Dec. 2018.

[22] P. S. Stanimirović, I. Živković, and Y. Wei, "Recurrent neural network for computing the drazin inverse," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 11, pp. 2830–2843, Nov. 2015.

[23] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2243–2254, Oct. 2017.

[24] S. Li, H. Wang, and M. U. Rafique, "A novel recurrent neural network for manipulator control with improved noise tolerance," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1908–1918, May 2018.

[25] Z. Zhang, Z. Yan, and T. Fu, "Varying-parameter RNN activated by finite-time functions for solving joint-drift problems of redundant robot manipulators," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5359–5367, Dec. 2018.

[26] Z. Zhang, T. Fu, Z. Yan, L. Jin, L. Xiao, Y. Sun, Z. Yu, and Y. Lin, "A varying-parameter convergent-differential neural network for solving joint-angular-drift problems of redundant robot manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 679–689, Apr. 2018.

[27] Z. Zhang, L. Zheng, J. Yu, Y. Li, and Z. Yu, "Three recurrent neural networks and three numerical methods for solving a repetitive motion planning scheme of redundant robot manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 3, pp. 1423–1433, Jun. 2017.

[28] Z. Xie, L. Jin, X. Du, X. Xiao, H. Li, and S. Li, "On generalized RMP scheme for redundant robot manipulators aided with dynamic neural networks and nonconvex bound constraints," *IEEE Trans. Ind. Informat.*, vol. 15, no. 9, pp. 5172–5181, Sep. 2019.

[29] Y. Zhang and S. S. Ge, "Design and analysis of a general recurrent neural network model for time-varying matrix inversion," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1477–1490, Nov. 2005.

[30] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2005.

[31] P. S. Stanimirović and M. D. Petković, "Gradient neural dynamics for solving matrix equations and their applications," *Neurocomputing*, vol. 306, pp. 200–212, Sep. 2018.

[32] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2001.

[33] P. M. Kebria, S. Al-wais, H. Abdi, and S. Nahavandi, "Kinematic and dynamic modelling of UR5 manipulator," *IEEE Int. Conf. Syst., Man, Cybern.(SMC)*, 2016, pp. 4229–4234.

[34] L. Jin, B. Liao, M. Liu, L. Xiao, D. Guo, and X. Yan, "Different-level simultaneous minimization scheme for fault tolerance of redundant manipulator aided with discrete-time recurrent neural network," *Front. Neurorobot.*, vol. 11, pp. 1–7, Sep. 2017.

[35] C. Li, C. Yang, and C. Giannetti, "Segmentation and generalisation for writing skills transfer from humans to robots," *Cogn. Comput. Syst.*, vol. 1, no. 1, pp. 20–25, Mar. 2019.

Long Jin (M'17) received the B.E. degree in automation and the Ph.D. degree in Information and Communication Engineering from Sun Yat-sen University, Guangzhou, China, in 2011 and 2016, respectively. He is currently a full professor with the School of Information Science and Engineering, Lanzhou University, Lanzhou, China. Before joining Lanzhou University in 2017, he was a Postdoctoral Fellow with the Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong. His main research interests include neural networks, robotics, and intelligent information processing. He is a member of the IEEE.

Zhengtai Xie received the B.S. degree from Lanzhou University, China, in 2019. He is currently working toward the M.E. degree in Communication and Information Systems from Lanzhou University, China. His main research interests include robotics, neural networks, intelligent information processing, artificial intelligence and optimization theory.

Mei Liu received the B.E. degree in communication engineering from the Yantai University, Yantai, China, in 2011, and the M.E. degree in pattern recognition and intelligent system from the Sun Yatsen University, Guangzhou, China, in 2014. She is currently a teacher with the School of Information Science and Engineering, Lanzhou University, Lanzhou, China. Before joining Lanzhou University in 2017, she was a lecturer with the College of Physics, Mechanical and Electrical Engineering, Jishou University, Jishou, China. Her main research interests include neural networks, computation and optimization.

Ke Chen is currently an associate professor in the South China University of Technology (SCUT), Guangzhou, China. He received his B.E. and M.E. degrees at the Sun Yat-sen University in 2007 and 2009 respectively, and Ph.D degree at the Queen Mary, University of London in 2013. Before he joined in SCUT, he worked at the Tampere University of Technology, Finland for five years. He has published more than 70 papers including top-tier CVPR, ICCV, IJCAI in the field. His research interests include computer vision, pattern recognition, neural dynamics and robotics.

Chunxu Li received the B.Engrg. degree in Engineering from Qingdao University of Science and Technology, Qingdao, China, in 2014, and the M.Sc. degree in Engineering from Swansea University, Swansea, UK, in 2016. He received his Ph.D. degree in Electrical and Electronic Engineering in 2019 with Swansea University, Swansea, UK. His research interests include robotics, automation, and computational intelligence. He is currently a lecturer at Centre for Robotics and Neural Systems, University of Plymouth.

Chenguang Yang (M'10-SM'16) is a Professor of Robotics. He received the B.Eng. degree in measurement and control from Northwestern Polytechnical University, Xian, China, in 2005, and the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010. He received postdoctoral training at Imperial College London, UK, from 2009 to 2010. He was awarded Marie Curie International Incoming Fellowship, EPSRC Innovation Fellowship, and the Best Paper Award of the IEEE Transactions on Robotics as well as over ten conference Best Paper Awards. His research interest lies in human robot interaction and intelligent system design.