Dynamic Neural Networks for Motion-Force Control of Redundant Manipulators: An Optimization Perspective

Zhihao Xu, *Member, IEEE*, Shuai Li, *Senior member, IEEE*, Xuefeng Zhou, *Member, IEEE*, Songbin Zhou, Taobo Cheng and Yisheng Guan, *Member, IEEE*

***Abstract*—Accurate position-force control is a core and challenging problem in robotics, especially for manipulators with redundant DOFs. For example, trajectory tracking based control usually fails for grinding robots due to intolerable impact forces imposed onto the end-effectors. The main difficulties lie in the coupling of motion and contact force, redundancy resolution and physical constraints, *etc.* In this paper, we propose a novel motion-force control strategy in the framework of projection recurrent neural networks. Tracking error and contact force are described in orthogonal spaces respectively, and by selecting minimizing joint torque as secondary task, the control problem is formulated as a quadratic-programming (QP) problem under multiple constraints. In order to obtain real-time optimization of joint toque which is non-convex relative to joint angles, the original QP is reconstructed in velocity level, where the original objective function is replaced by its time derivative. Then a dynamic neural network which is convergence provable is established to solve the modified QP problem online. This work generalizes projection recurrent neural network based position control of manipulators to that of position-force control, which opens a new avenue to shift position-force control of manipulators from pure control perspective to cross design with both convergence and optimality consideration. Numerical and experimental results show that the proposed scheme achieves accurate position-force control, and is capable of handling inequality constraints such as joint angular, velocity and torque limitations, simultaneously, consumption of joint torque can be decreased effectively.**

***Index Terms*—Dynamic neural network, redundancy resolution, joint torque optimization, motion-force control.**

I. Introduction

Redundant manipulators, which have more DOFs than those required to complete a given task, are more flexible than non-redundant ones. The redundant DOFs enable manipulators to realize fault tolerant control, improve operation performance and enhance reliability. Therefore, redundant manipulators have been widely used in industry, agriculture, military, space exploration, *etc*. Consequently, the research on redundant manipulator has been studied intensively [1]–[3].

Motion control and force control construct two major paradigms in redundant manipulator control. In motion control problems, a basic assumption is that there is no contact between the robot and environment, in other words, the robot can move freely in the work space [4]. This problem is manifested in applications such as painting, welding, palletizing, *etc*. Then the core issue is to design control commands, to drive the robot to move according to a predefined trajectory. The control command may be joint angular sequence [5], velocity sequence [6], acceleration sequence [7] or torque sequences [8]–[10]. The redundancy resolution is usually used to achieve a secondary task, such as avoiding obstacles [11] , avoiding singularities [12], *etc*. Different from motion control, force control involves the direct interaction between a robot and its environment. The control of contact force is capable of enhancing robot′s robustness and flexibility in weak structured environment, and thus strengthen the operating ability [13]. The corresponding typical applications can be found in tasks such as polishing, grinding, assembly [14], [15], *etc*. In [16], a theoretical framework of impedance control is proposed. The basic idea is to consider the environment as an admittance and the robot as an impedance. By maintaining a dynamic relationship between force and motion, the controller behaves as a spring-mass-damper system. In [17], a hybrid position-force controller is proposed by Raibert and Craig, which combines information of position and force together, to realize simultaneous control of position and force constrains. Based on these two control frameworks, a series of controllers are proposed and verified by simulations or experiments [18], [19].As to the system uncertainties, intelligent control methods based on neural networks(including RBF neural networks, fuzzy neural networks, recurrent neural networks) are also introduced, in which the networks are used to handle system uncertainties and disturbances. For example, in [20], a feedforward neural network (NN) based control strategy is realized in position-based impedance control, in which a two-layer NN is built to modify the reference trajectory. In

This work is supported by Foshan Key Technology Research Project (Grant NO. 1920001001148) Foshan Innovation and Entrepreneurship Team Project (Grant NO. 2018IT100173) Guangdong Province Key Areas R&D Program (Grant NO. 2019B090919002) GDAS' Project of Thousand doctors (postdoctors) Introduction (2019GDASYL-0103078) Guangzhou Science Research Plan C Major Project (Grant No. 201804020095) Guangdong Province Science and Technology Major Projects (Grant No. 2017B010110010) Guangdong Innovative Talent Project of Young College (Grant No. 2016TQ03X463). *(Corresponding authors: Shuai Li).* Copyright (c) 2009 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

Z. Xu, X. Zhou, S. Zhou and T. Cheng are with the Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou, 210094, China. (e-mails:zh.xu@giim.ac.cn; xf.zhou@giim.ac.cn; sb.zhou@giim.ac.cn; tb.cheng@giim.ac.cn); S. Li is with the School of Engineering, Swansea University, Swansea, United Kingdom. (e-mail: shuai.li@swansea.ac.uk); Y.Guan is with Biomimetic and Intelligent Robotics Lab (BIRL), School of Electro-Mechanical Engineering, Guangdong University of Technology, Guangzhou, 210094, China, (e-mail: ysguan@gdut.edu.cn); Z. Xu, X. Zhou and S. Li are also with Foshan Tri-Co Intelligent Robot Technology Co., Ltd.

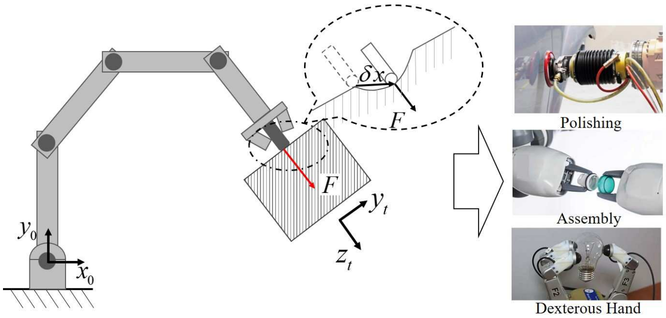


Fig. 1. A brief diagram of robotic force control and typical applications.

[21], a robust neural force control scheme is proposed. It is verified that the controller can make the robot to track a specified desired force, and at the same time, uncertainties in environment location and stiffness as well as robot dynamics can be compensated. In [22], a fuzzy recurrent wavelet neural network (RNN) based adaptive force/motion control scheme is proposed for a class of mobile manipulator, which combines the advantages of recurrent technique and wavelet networks. In [23], an adaptive impedance controller based on dynamic recurrent fuzzy neural network is developed for upper-limb rehabilitation robot, where the RNN is used to regulate the desired impedance control parameters between rehabilitation robotic end-effector and upper-limb. Considering the motion planning problem in nonprehensile manipulations, Kutsuzawa *et al.* propose a sequence-to-sequence based method [24], which is neural networks applicable for time series conversion. In [25], this work is extended to trajectory planning problem with contact model.

Although the above mentioned works have achieved great success in motion-force control of non-redundant manipulators, the control of redundant ones has received minor attention. Remarkably that the redundancy of a manipulator provides an opportunity to meet a secondary objective, but also sets difficulties in mathematically solving it. In [26], a position-force control strategy is proposed, where the robot′s motion is completely decoupled into two parts: end-effector′s motion is controlled to achieve position-force control to the environment, while the internal motion is designed to avoid obstacles via minimizing impact forces. In [27], a robust control strategy with the capability of regulating contact force and apparent impedances is designed, the controller also shows great robustness with respect to dynamic and kinematic uncertainties. In [28], by defining a function scaling the difference between joint angle and its boundaries, the limitation of joint angle is avoided as the second task. However, these literature requires continuous calculation of pseudo-inverse of Jacobian matrix, which would lead to a huge computational burden, and at the same time, it remains a difficult problem in dealing with multiple constraints [29].

In order to handle the redundancy resolution problem for redundant manipulators, one feasible method is to transform the control problem into an optimization one under constraints [30]. The objective function is built according to the secondary task, and the constraints are formulated based on the primary task and physical limitations. This optimization problem is usually described as a quadratic-programming (QP) problem [31]. With high efficiency of parallel computing, recurrent neural networks are often used to solve the QP based redundancy resolution online [32]. In recent studies, projection recurrent neural network (RNN) based controllers are introduced into motion control of redundant manipulators. In [33], the RNN is modified such that projection operations to nonconvex sets are allowed, and accumulation of tracking error against system noise is avoided either. In [34], a manipulability optimization scheme is proposed by maximizing its time derivative indirectly, numerical experiments show that this method achieves 40% improvement. Some extensions of projection based RNN can be found in multiple manipulator systems [35], flexible robots [36], model uncertainties [37], [38]. Despite the wonderful results on motion control of redundant manipulators based on projection RNN has achieved, to the best of our knowledge, few research on the application of projection RNN to motion-force control of manipulators has been reported.

In accordance with the foregoing description of the relevant literature review, the position-force control problem for redundant manipulators should consist of three aspects: position-force control, performance optimization, and ensuring physical limitations. Therefore, in this paper, we propose a projection RNN based position-force control scheme for robots with redundant DOFs, which is capable of handling the optimal position-force control problem while ensuring inequality constraints. This paper is an important extension of the recurrent neural network to the force control of robots. In Section II, the tracking error and contact force are modelled, and the control problem is written as a QP problem. In Section III, by rewriting objective function and constraints, the QP is reformulated in velocity level. In Section IV, a RNN is built to solve the redundancy resolution problem. Stability is also proved. In Section V, verifiable numerical experiments on a 4-DOFs planar manipulator are carried out. Finally, Section VI concludes the paper. Before ending this section, the main contributions are listed as below:

- This paper deals with motion-force control of redundant manipulators in the framework of projection RNNs, the control strategy realizes real-time optimization of joint torque, which could save energy in industrial applications.
- In the proposed control scheme, the motion-force control problem as well as redundancy resolution problem are reconstructed to facilitate practical implementations.
- The controller is capable of handling multiple inequality constraints, including but not limited to angle constraints, angular velocity constraints and torque constraints. This is of great significance in improving system security.
- By introducing model information into the neural network, the established RNN has a simpler structure, which is helpful for the application in practical engineering.

II. Problem Formulation

A. Problem Formulation

In this paper, we focus on position-force control problem for redundant manipulators. Fig.(1) also gives a brief introduction of a redundant robot and its operation on a workpiece. The robot is expected to offer a desired contact force in the vertical direction of the contact surface, at the same time, the end-effector is required to track a predefined trajectory along the surface. In the base coordinate frame $R_0(0_0, \boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{z}_0)$, forward kinematics of a serial manipulator can be written as

$$\mathbf{f}(\boldsymbol{\theta}(t)) = \mathbf{x}(t), \tag{1}$$

where $\boldsymbol{\theta} \in \mathbb{R}^n$ is the vector of joint angles, and $\mathbf{x} \in \mathbb{R}^m$ represents the end-effector′s coordinate vector in frame R_0, $\mathbf{f}(\bullet) : \mathbb{R}^n \to \mathbb{R}^m$ is used to describe the forward kinematics operator. For a redundant manipulator, we have $n > m$.

By differentiating $\mathbf{x}(t)$ with respect to time t, we can get the relationship between Cartesian velocity $\dot{\boldsymbol{x}}(t) \in \mathbb{R}^m$ and joint velocity (or joint control signal) $\dot{\boldsymbol{\theta}}(t) \in \mathbb{R}^n$ as follows:

$$\mathbf{J}(\boldsymbol{\theta}(t))\dot{\boldsymbol{\theta}}(t) = \dot{\boldsymbol{x}}(t), \tag{2}$$

where $\mathbf{J}(\boldsymbol{\theta}(t)) = \partial \mathbf{f}(\boldsymbol{\theta}(t))/\partial \boldsymbol{\theta}(t)$ is called Jacobian matrix.

In position-force control tasks, the end-effector′s motion is constrained by the contact surface. For simplicity, we define a tool coordinate system as $R_t(\boldsymbol{x}_t, \boldsymbol{y}_t, \boldsymbol{z}_t)$, in which the axis $\boldsymbol{z}_t$ is set in alignment with the vertical direction of the contact surface. Obviously, the motion of end-effector can be specified along $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$. In this paper, frictional force between the robot and contact surface is ignored, therefore, the contact force $\boldsymbol{F}$ is in alignment with $\boldsymbol{z}_t$.

In the tool coordinate system R_t, let $\delta \boldsymbol{X}_t$ be the displacement between effector and its position command, then the contact force $\boldsymbol{F}_t$ can be formulated as

$$\boldsymbol{F}_t = k_f \boldsymbol{\Sigma}_f \delta \boldsymbol{X}_t, \tag{3}$$

where $k_f > 0$ is the stiffness coefficient, $\Sigma_f = diag(0, 0, 1)$. Diagonal matrix Σ_f describes the relationship between the contact force and relative displacement along different axes: 1 means that displacement component along $\boldsymbol{z}_t$ affects the contact force, and 0 otherwise.

Similarly, in tool coordinate system R_t, the position tracking error $\boldsymbol{e}_t$ can be written as

$$\boldsymbol{e}_t = \bar{\Sigma}_f \delta \boldsymbol{X}_t, \tag{4}$$

where $\bar{\Sigma}_f = \boldsymbol{I} - \Sigma_f = diag(1, 1, 0)$, 1 means there is a DOF of movement along the corresponding direction, and 0 otherwise.

When the contact surface is prior known, R_t can be obtained from R_0 by a rotation matrix $\boldsymbol{S}_t$. Let $\boldsymbol{F}$, $\boldsymbol{e}_0$ and $\delta \boldsymbol{X}$ be the corresponding description of $\boldsymbol{F}_t$, $\boldsymbol{e}_t$ and $\delta \boldsymbol{X}_t$ in coordinate frame R_0, then we have $\boldsymbol{F} = \boldsymbol{S}_t^{\mathrm{T}} \boldsymbol{F}_t$, $\boldsymbol{e}_t = \boldsymbol{S}_t \boldsymbol{e}_0$ and $\delta \boldsymbol{X}_t = \boldsymbol{S}_t \delta \boldsymbol{X}$. Therefore, $\boldsymbol{F}$ and $\boldsymbol{e}_0$ can be rewritten as

$$\begin{cases} \boldsymbol{F} = k_f \boldsymbol{S}_t^{\mathrm{T}} \Sigma_f \boldsymbol{S}_t \delta \boldsymbol{X}, \\ \boldsymbol{e}_0 = \boldsymbol{S}_t^{\mathrm{T}} \bar{\Sigma}_f \boldsymbol{S}_t \delta X. \end{cases} \tag{5}$$

Notable that in frame R_0, the displacement $\delta \boldsymbol{X}$ can be described as $\delta \boldsymbol{X} = \boldsymbol{x} - \boldsymbol{x}_{\mathrm{d}}$, where $\boldsymbol{x}_{\mathrm{d}}$ is the desired position signals described in R_0. Using (1), (5) can be rewritten as

$$\begin{cases} \boldsymbol{F} = k_f \boldsymbol{S}_t^{\mathrm{T}} \Sigma_f \boldsymbol{S}_t (\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}), \\ \boldsymbol{e}_0 = \boldsymbol{S}_t^{\mathrm{T}} \bar{\Sigma}_f \boldsymbol{S}_t (\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}). \end{cases} \tag{6}$$

Remark 1: Equation (6) gives the unified description of the relationship between the contact force $\boldsymbol{F}$, position tracking error $\boldsymbol{e}_0$ and displacement $\delta \boldsymbol{X}$ in R_0. $\delta \boldsymbol{X}$ will lead to contact force $\boldsymbol{F}$ in the vertical direction, and position tracking error $\boldsymbol{e}_0$ along the contact surface. Σ_f and $\bar{\Sigma}_f$ are used to realize the decoupling of the contact force and tracking error of the end-effector. When the contact surface is known, the combination of Σ_f, $\bar{\Sigma}_f$ and $\boldsymbol{S}_t$ enables the normalized description of the control tasks.

In real implementations, given the desired contact force $\boldsymbol{F}_{\mathrm{s}}$ and trajectory command $\boldsymbol{x}_{\mathrm{d}}$, the manipulator′s end-effector is expected to offer contact force $\boldsymbol{F}_{\mathrm{d}}$ while tracking $\boldsymbol{x}_{\mathrm{d}}$, *i.e.*, $\boldsymbol{F} \to \boldsymbol{F}_{\mathrm{d}}$, $\boldsymbol{e}_0 \to 0$. For the convenience of writing in the following sections, let $\boldsymbol{A} = [k_f \boldsymbol{S}_t^{\mathrm{T}} \Sigma_f \boldsymbol{S}_t; \boldsymbol{S}_t^{\mathrm{T}} \bar{\Sigma}_f \boldsymbol{S}_t] \in \mathbb{R}^{2m \times m}$, $\boldsymbol{r} = [\boldsymbol{F}^{\mathrm{T}}, \boldsymbol{e}_0^{\mathrm{T}}]^{\mathrm{T}}$, and $\boldsymbol{r}_{\mathrm{d}} = [\boldsymbol{F}_{\mathrm{d}}; \boldsymbol{0}]$. Then (6) can be reformulated as

$$\boldsymbol{A}(\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}) = \boldsymbol{r}. \tag{7}$$

Therefore, the control objective of position-force control is to adjust the joint angles $\boldsymbol{\theta}$, to ensure $\boldsymbol{r} \to \boldsymbol{r}_{\mathrm{d}}$.

B. Joint Torque and Physical Constraints

When the end-effector offers a contact force $\boldsymbol{F}$, the corresponding torque is provided by motors at every joint. The relationship between contact force $\boldsymbol{F}$ and the joint torque $\boldsymbol{\tau}$ can be formulated as

$$\boldsymbol{\tau} = \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta})\boldsymbol{F}. \tag{8}$$

In the control of redundant manipulators, there would be infinite groups of solutions to a certain control task. In order to save energy during the control process, we select a objective function scaling energy consumption as $\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{\tau}/2$. The smaller $\boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{\tau}/2$, the less energy consumption.

In real implementations, the system is limited by physical constraints. For example, the joint angles $\boldsymbol{\theta}$ and velocities $\dot{\boldsymbol{\theta}}$ must not exceed their limits: $\boldsymbol{\theta}^{\min}$,$\boldsymbol{\theta}^{\max}$, $\dot{\boldsymbol{\theta}}^{\min}$, $\dot{\boldsymbol{\theta}}^{\max}$, since the possible collisions or overheating of motor would lead to irreversible damages. At the same time, considering the bounded torque output of the motors, the limitation of joint torque $\boldsymbol{\tau}$ is described as $\boldsymbol{\tau}^{\min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}^{\max}$.

C. Optimization Problem Formulation

According to the descriptions above, the position-force control problem for redundant manipulators considering torque optimization can be formulated as

$$\min \quad G_1 = \boldsymbol{\tau}^{\mathrm{T}}\boldsymbol{\tau}/2, \tag{9a}$$

$$\text{s.t.} \quad \boldsymbol{\tau} = \boldsymbol{J}^{\mathrm{T}}\boldsymbol{F}, \tag{9b}$$

$$r_{\mathrm{d}} = \boldsymbol{A}(\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}), \tag{9c}$$

$$\boldsymbol{\theta}^{\min} \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}^{\max}, \tag{9d}$$

$$\dot{\boldsymbol{\theta}}^{\min} \leq \dot{\boldsymbol{\theta}} \leq \dot{\boldsymbol{\theta}}^{\max}, \tag{9e}$$

$$\boldsymbol{\tau}^{\min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}^{\max}, \tag{9f}$$

with $\boldsymbol{\theta}$ being the decision variable. Equation (9a) is the cost function to be minimized, the equality constraints (9b) describes the relationship between the resulting joint torque $\boldsymbol{\tau}$ and contact force $\boldsymbol{F}$. The force and motion tasks of the robot are described in (9c), and inequality constraints (9e)(9d)(9f) show the physical limitations to be satisfied. By substituting (9b) into (9a), the optimization problem can be rewritten as

$$\begin{aligned} \min \quad & G_1 = \boldsymbol{F}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}/2, && (10a) \\ \text{s.t.} \quad & \boldsymbol{r}_{\mathrm{d}} = \boldsymbol{A}(\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}), && (10b) \\ & \boldsymbol{\theta}^{\min} \le \boldsymbol{\theta} \le \boldsymbol{\theta}^{\max}, && (10c) \\ & \dot{\boldsymbol{\theta}}^{\min} \le \dot{\boldsymbol{\theta}} \le \dot{\boldsymbol{\theta}}^{\max}, && (10d) \\ & \boldsymbol{\tau}^{\min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}^{\max}. && (10e) \end{aligned}$$

To solve (10), there are two main challenges. Firstly, as an objective function to be minimized, $\boldsymbol{F}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}/2$ is usually non-convex relative to $\boldsymbol{\theta}$, because it is a function of $\boldsymbol{J}(\boldsymbol{\theta})$. Secondly, the equation constrain (10b) is highly nonlinear, and at the same time, it remains difficult to handle the inequality constraints, especially (10d) and (10e).

III. Reconstruction of Optimization Problem

In this section, in order to overcome the above difficulties, the redundancy resolution problem (10) will be reconstructed. The objective function is firstly redefined, and both equality and inequality constrains are rebuilt in velocity level.

A. Reconstruction of Objective Function

As to $\boldsymbol{F}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}/2$, we will replace $\boldsymbol{F}$ with the desired value $\boldsymbol{F}_{\mathrm{d}}$. Therefore, the optimization function can be formulated as $G_2 = \boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}}/2$.

Remark 2: There are two main reasons: firstly, according to the control objective, the contact force $\boldsymbol{F}$ is expected to track $\boldsymbol{F}_{\mathrm{d}}$, if the controller is proper designed, $\boldsymbol{F}$ will eventually converge to $\boldsymbol{F}_{\mathrm{d}}$, consequently, $\boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}}/2$ will be equivalent to $\boldsymbol{F}^{\mathrm{T}} \boldsymbol{J}(\boldsymbol{\theta}) \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}/2$. Secondly, $\boldsymbol{F}_{\mathrm{d}}$ is independent of $\boldsymbol{\theta}$, this replacement will reduce the computational complexity in the control process.

Differentiating G_2 with respect to time, we have

$$\dot{G}_2 = (\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}})^{\mathrm{T}} \frac{\mathrm{d}(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}})}{\mathrm{d}t}. \tag{11}$$

Obviously, $\dot{G}_2$ describes the change of G_2. By minimizing $\dot{G}_2$, the system will be compelled to develop in the direction of decreasing G_2 . Therefore, in this paper, we use $\dot{G}_2$ instead of G_2 as the new objective function. Notable that $\mathrm{d}(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}})/\mathrm{d}t$ can be formulated as

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}}) &= \sum_{i=1}^{n} \frac{\partial(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}})}{\partial \boldsymbol{\theta}_i} \dot{\boldsymbol{\theta}}_i + \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \dot{\mathbf{F}}_{\mathrm{d}} \\ &= [\boldsymbol{H}_i, \cdots, \boldsymbol{H}_n] \dot{\boldsymbol{\theta}} + \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \dot{\mathbf{F}}_{\mathrm{d}}, \end{aligned} \tag{12}$$

where $\boldsymbol{H}_i \in \mathbb{R}^n$ is

$$\boldsymbol{H}_i = \frac{\partial(\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{\theta}) \boldsymbol{F}_{\mathrm{d}})}{\partial \boldsymbol{\theta}_i} = \begin{bmatrix} \sum_{j=1}^{m} (\partial(\boldsymbol{J}(j,1) \boldsymbol{F}_{\mathrm{d}}(j))/\partial \boldsymbol{\theta}_i) \\ \sum_{j=1}^{m} (\partial(\boldsymbol{J}(j,2) \boldsymbol{F}_{\mathrm{d}}(j))/\partial \boldsymbol{\theta}_i) \\ \cdots \\ \sum_{j=1}^{m} (\partial(\boldsymbol{J}(j,n) \boldsymbol{F}_{\mathrm{d}}(j))/\partial \boldsymbol{\theta}_i) \end{bmatrix}.$$

Let $\boldsymbol{H} = [\boldsymbol{H}_1, \cdots, \boldsymbol{H}_n]$, then (11) can be converted as

$$\dot{G}_2 = \boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{H} \dot{\boldsymbol{\theta}} + \boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{J}^{\mathrm{T}} \dot{\mathbf{F}}_{\mathrm{d}}. \tag{13}$$

It is worth pointing that the second term of (13) is independent of $\dot{\theta}$, therefore, the objective function is equivalent to $\boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{H} \dot{\boldsymbol{\theta}}$.

B. Reconstruction of Constraints

In this subsection, we will transform the constrains into velocity level. First of all, we define a concatenated vector describing force and position errors as $\boldsymbol{e} = \boldsymbol{r} - \boldsymbol{r}_{\mathrm{d}} = [\boldsymbol{F} - \boldsymbol{F}_{\mathrm{d}}; \boldsymbol{e}_0]$, according to (7), $\boldsymbol{e}$ can be formulated as

$$\boldsymbol{e} = \boldsymbol{A}(\boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}) - \boldsymbol{r}_{\mathrm{d}}. \tag{14}$$

Differentiating $\boldsymbol{e}$ and combing (2) yields

$$\dot{\boldsymbol{e}} = \boldsymbol{A}(\boldsymbol{J} \dot{\boldsymbol{\theta}} - \dot{\boldsymbol{x}}_{\mathrm{d}}) - \dot{\boldsymbol{r}}_{\mathrm{d}}. \tag{15}$$

In order to ensure $\boldsymbol{e}$ converge to zero, a simple controller can be designed as $\dot{\boldsymbol{e}} = -k\boldsymbol{e}$, where $k > 0$ is a positive constant. According to (14)(15), the equality constrains can be converted in velocity level as

$$\boldsymbol{A} \boldsymbol{J} \dot{\boldsymbol{\theta}} = \dot{\boldsymbol{r}}_{\mathrm{d}} + \boldsymbol{A} \dot{\boldsymbol{x}}_{\mathrm{d}} - k(\boldsymbol{A} \boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}}). \tag{16}$$

As to the inequality constraints (10c)(10d), according to [30], let $\boldsymbol{\omega} = \dot{\boldsymbol{\theta}}$ and define $\alpha \ge 0$ as a constant parameter to scale the negative feedback to conform the joint constraints, these two constraints can be formulated in the speed level as

$$\boldsymbol{\omega}^{\min} \le \boldsymbol{\omega} \le \boldsymbol{\omega}^{\max}, \tag{17}$$

where $\boldsymbol{\omega}^{\min} = \max\{\alpha(\boldsymbol{\theta}^{\min} - \boldsymbol{\theta}), \dot{\boldsymbol{\theta}}^{\min}\}$, and $\boldsymbol{\omega}^{\max} = \min\{\alpha(\boldsymbol{\theta}^{\max} - \boldsymbol{\theta}), \dot{\boldsymbol{\theta}}^{\max}\}$.

Similarly, (10e) can be built indirectly by limiting its derivative: $\beta(\boldsymbol{\tau}^{\min} - \boldsymbol{\tau}) \le \dot{\boldsymbol{\tau}} \le \beta(\boldsymbol{\tau}^{\max} - \boldsymbol{\tau})$, where β is a positive constant. By combining (12), the boundedness of joint torque can be rewritten as an inequality constraint about a function $\boldsymbol{g}(\boldsymbol{\omega})$ as

$$\boldsymbol{g}(\boldsymbol{\omega}) \le \boldsymbol{0}, \tag{18}$$

where $\boldsymbol{g}(\boldsymbol{\omega}) = [\beta(\boldsymbol{\tau}^{\min} - \boldsymbol{\tau}) - \boldsymbol{J}^{\mathrm{T}} \dot{\mathbf{F}}_{\mathrm{d}} - \mathbf{H}\boldsymbol{\omega}, \mathbf{H}\boldsymbol{\omega} - \beta(\boldsymbol{\tau}^{\max} - \mathbf{tau}) + \mathbf{J}^{\mathrm{T}} \dot{\mathbf{F}}_{\mathrm{d}}]^{\mathrm{T}} \in \mathbb{R}^{\mathbf{2n}}$.

C. Reformulation and Convexification

According to the above description, in order to achieve position-force control of redundant manipulators, instead of solving (10) directly, one feasible solution is to solve the optimization problem in velocity level as

$$\begin{aligned} \min \quad & \boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{H} \boldsymbol{\omega}, && (19a) \\ \text{s.t.} \quad & \boldsymbol{r}_r = \boldsymbol{A} \boldsymbol{J} \boldsymbol{\omega}, && (19b) \\ & \boldsymbol{g}(\boldsymbol{\omega}) \le \boldsymbol{0}, && (19c) \\ & \boldsymbol{\omega} \in \boldsymbol{\Omega}, && (19d) \end{aligned}$$

where $\boldsymbol{r}_r = \dot{\boldsymbol{r}}_d + \boldsymbol{A} \dot{\boldsymbol{x}}_{\mathrm{d}} - k(\boldsymbol{A} \boldsymbol{f}(\boldsymbol{\theta}) - \boldsymbol{x}_{\mathrm{d}})$, $\boldsymbol{\Omega} = \{\boldsymbol{\omega} \in \mathbb{R}^n | \boldsymbol{\omega}_i^{\min} \le \boldsymbol{\omega}_i \le \boldsymbol{\omega}_i^{\max}\}$ is a convex set. It is remarkable that the objective function described in (19a) is non-convex relative to $\boldsymbol{\omega}$. Therefore, (19b) is introduced to convexify (19a). The final form of optimization problem is described as

$$\begin{aligned} \min \quad & \boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}} \boldsymbol{J} \boldsymbol{H} \boldsymbol{\omega} + (\boldsymbol{A} \boldsymbol{J} \boldsymbol{\omega} - \boldsymbol{r}_r)^{\mathrm{T}} (\boldsymbol{A} \boldsymbol{J} \boldsymbol{\omega} - \boldsymbol{r}_r), && (20a) \\ \text{s.t.} \quad & \boldsymbol{r}_r = \boldsymbol{A} \boldsymbol{J} \boldsymbol{\omega}, && (20b) \\ & \boldsymbol{g}(\boldsymbol{\omega}) \le \boldsymbol{0}, && (20c) \\ & \boldsymbol{\omega} \in \boldsymbol{\Omega}. && (20d) \end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIE.2020.2970635, IEEE Transactions on Industrial Electronics

So far, we have reconstructed the position-force control with joint torque optimization problem into a quadratic programming issue under constraints. However, the QP problem (20) cannot be solved directly.

IV. RNN Based Redundancy Resolution

In this section, in order to solve the optimization problem (20), an expanded recurrent neural network is built to obtain the optimal solution of (20). Stability will be also discussed.

A. RNN Design

Firstly, let $\boldsymbol{\lambda}_1 \in \mathbb{R}^{2m}$ and $\boldsymbol{\lambda}_2 \in \mathbb{R}^{2n}$ be dual variables to constraints (20b) and (20c), a Lagrange function is defined as

$$\begin{aligned} L =&\boldsymbol{F}_{\mathrm{d}}^{\mathrm{T}}\boldsymbol{J}\boldsymbol{H}\boldsymbol{\omega} + (\boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega} - \boldsymbol{r}_r)^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega} - \boldsymbol{r}_r) \\ &+ \boldsymbol{\lambda}_1^{\mathrm{T}}(\boldsymbol{r}_r - \boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega}) + \boldsymbol{\lambda}_2^{\mathrm{T}}\boldsymbol{g}(\boldsymbol{\omega}). \end{aligned} \tag{21}$$

According to Karush−Kuhn−Tucker condition, the optimal solution of the optimization problem (20) can be equivalently formulated as

$$\boldsymbol{\omega} = \boldsymbol{P}_{\Omega}(\boldsymbol{\omega} - \frac{\partial L}{\partial \boldsymbol{\omega}}), \tag{22a}$$

$$\boldsymbol{r}_r = \boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega}, \tag{22b}$$

$$\boldsymbol{\lambda}_2 = (\boldsymbol{\lambda}_2 + \boldsymbol{g}(\boldsymbol{\omega}))^{+}, \tag{22c}$$

where $\boldsymbol{P}_{\Omega}(\boldsymbol{x}) = \mathrm{argmin}_{\boldsymbol{y}\in\Omega}||\boldsymbol{y} - \boldsymbol{x}||$ is a projection operation to convex set Ω, and $(\boldsymbol{x})^{+} = (\boldsymbol{x}_1^{+}, \cdots, \boldsymbol{x}_{2n}^{+})^{\mathrm{T}}$, $\boldsymbol{x}_i^{+} = \max(\boldsymbol{x}_i, 0)$.

In order to solve (22), an expanded recurrent neural network is designed as

$$\begin{aligned} \epsilon\dot{\boldsymbol{\omega}} =& -\boldsymbol{\omega} + \boldsymbol{P}_{\Omega}(\boldsymbol{\omega} - \boldsymbol{H}^{\mathrm{T}}J^{\mathrm{T}}\boldsymbol{F}_{\mathrm{d}} - \boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega} - \boldsymbol{r}_r) \\ &+ \boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}_1 - \nabla\boldsymbol{g}\boldsymbol{\lambda}_2), \end{aligned} \tag{23a}$$

$$\epsilon\dot{\boldsymbol{\lambda}}_1 =\boldsymbol{r}_r - \boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega}, \tag{23b}$$

$$\epsilon\dot{\boldsymbol{\lambda}}_2 =(\boldsymbol{\lambda}_2 - (\boldsymbol{\lambda}_2 + \boldsymbol{g}(\boldsymbol{\omega}))^{+}), \tag{23c}$$

where $\nabla\boldsymbol{g} = (\frac{\partial \boldsymbol{g}_1}{\partial \boldsymbol{\omega}}, \cdots, \frac{\partial \boldsymbol{g}_{2m}}{\partial \boldsymbol{\omega}}) = [-\boldsymbol{H}^{\mathrm{T}}, \boldsymbol{H}^{\mathrm{T}}] \in \mathbb{R}^{n\times 2n}$, ϵ is a positive constant scaling the convergence of (23). The projection operator P_{Ω} plays an important role in guaranteeing the boundedness of the output of neural network,*i.e.*, the boundedness of ω can be ensured by introducing P_{Ω}. As described in Eq. (17), based on escape velocity method, both the boundedness of joint angles and velocities are guaranteed.

The schematic diagram of the proposed control structure is shown in Fig. 2. Based on the problem formulation, the position and force controller Eq.(14)-(16) are described in the form of equality constraint $\boldsymbol{r}_r = \boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega}$, and the physical constraints Eq.(10c)-(10e) are formulated as the inequality constraint $\boldsymbol{g}(\boldsymbol{\omega}) \leq \boldsymbol{0}$. With the introduction of joint torque optimization scheme Eq.(19a) and the QP-type problem formulation, a projection RNN is established to obtain the control command in realtime.

Remark 3: The architecture of the established RNN is also shown in Fig. 2. The network is organized in a one-layer architecture, which consists of $2m + 3n$ neurons, namely $\boldsymbol{\omega} \in \mathbb{R}^{n}$, $\boldsymbol{\lambda}_1 \in \mathbb{R}^{2m}$ and $\boldsymbol{\lambda}_2 \in \mathbb{R}^{2n}$. Despite the difference between the proposed neural network with traditional recurrent neural networks [20], from both the mathematical description Eq.(23) and the architecture, the states $\boldsymbol{\omega}$, $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ are updated recurrently, *i.e.*, one characteristic of the established neural network can be found that the neural network uses its historical information to calculate the output at current moment, which is also a typical feature of recurrent neural networks. According to the previous research [40]–[43], neural networks with similar structure proposed to solve robot kinematic control problem are also called RNN. We expand this kind of RNN to the field of robot force control, which is also a main contribution of this paper.

B. Stability Analysis

In this part, theoretical analysis of stability and convergence of closed-loop system using the proposed neural network (23).

First of all, several important definitions and Lemmas are presented, which is very useful in stability analysis.

Definition 1. A continuously differentiable function $\boldsymbol{F}(\bullet)$ is said to be monotone if $\nabla\boldsymbol{F} + \nabla\boldsymbol{F}^{\mathrm{T}}$ is positive semi-definite, where $\nabla\boldsymbol{F}$ is the gradient of $\boldsymbol{F}(\bullet)$.

Lemma 1. [39] A dynamic neural network is said to converge to the equilibrium point if it satisfies

$$\kappa\dot{\boldsymbol{x}} = -\boldsymbol{x} + \boldsymbol{P}_S(\boldsymbol{x} - \varrho\boldsymbol{F}(x)), \tag{24}$$

where $\kappa > 0$ and $\varrho > 0$ are constant parameters, and $\boldsymbol{P}_S = \mathrm{argmin}_{\boldsymbol{y}\in S}||\boldsymbol{y} - \boldsymbol{x}||$ is a projection operator to closed set S.

So far, a theorem about the convergence of the redundancy resolution problem can be described as follows

Theorem 1: Given the motion-force control problem for redundant manipulators with torque optimization under physical constraints as (20), the recurrent neural network (23) is stable and will globally converge to the optimal solution of (20).

Proof: Let $\boldsymbol{\xi} = [\boldsymbol{\omega}^{\mathrm{T}}, \boldsymbol{\lambda}_1^{\mathrm{T}}, \boldsymbol{\lambda}_2^{\mathrm{T}}]^{\mathrm{T}}$, the proposed RNN (23) can be written as

$$\epsilon\dot{\boldsymbol{\xi}} = -\boldsymbol{\xi} + \boldsymbol{P}_{\bar{\Omega}}[\boldsymbol{\xi} - \boldsymbol{F}(\boldsymbol{\xi})], \tag{25}$$

where $\boldsymbol{F}(\boldsymbol{\xi}) = [\boldsymbol{F}_1(\boldsymbol{\xi}), \boldsymbol{F}_2(\boldsymbol{\xi}), \boldsymbol{F}_3(\boldsymbol{\xi})]^{\mathrm{T}} \in \mathbb{R}^{2m+3n}$, in which

$$\begin{bmatrix}\boldsymbol{F}_1\\ \boldsymbol{F}_2\\ \boldsymbol{F}_3\end{bmatrix} = \begin{bmatrix}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{F}_{\mathrm{d}} + \boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}(\boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega} - \boldsymbol{r}_r) - \boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{\lambda}_1 + \nabla\boldsymbol{g}\boldsymbol{\lambda}_2\\ \boldsymbol{\lambda}_1 - \boldsymbol{r}_r + \boldsymbol{A}\boldsymbol{J}\boldsymbol{\omega}\\ -\boldsymbol{g}(\boldsymbol{\omega})\end{bmatrix}.$$

Let $\nabla\boldsymbol{F}(\boldsymbol{\xi}) = \partial\boldsymbol{F}/\partial\boldsymbol{\xi}$, we have

$$\nabla\boldsymbol{F}(\boldsymbol{\xi}) = \begin{bmatrix}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{J} & -\boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}} & \nabla\boldsymbol{g}\\ \boldsymbol{A}\boldsymbol{J} & \boldsymbol{I} & \boldsymbol{0}\\ -(\nabla\boldsymbol{g})^{\mathrm{T}} & \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}. \tag{26}$$

It is remarkable that

$$\nabla\boldsymbol{F}(\boldsymbol{\xi}) + (\nabla\boldsymbol{F}(\boldsymbol{\xi}))^{\mathrm{T}} = \begin{bmatrix}2\boldsymbol{J}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{J} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & 2\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}. \tag{27}$$

From Definition 1, $\boldsymbol{F}(\boldsymbol{\xi})$ is a monotone function of $\boldsymbol{\xi}$.

According to the description of (23) and (25), $\boldsymbol{P}_{\bar{\Omega}}$ can be formulated as $\boldsymbol{P}_{\bar{\Omega}} = [\boldsymbol{P}_{\Omega}; \boldsymbol{P}_R; \boldsymbol{P}_{\Lambda}]$, where $\boldsymbol{P}_R \in \mathbb{R}^{2m}$ is a projection operator of $\boldsymbol{\lambda}_1$ to set R, with the upper and lower bounds being ±∞. Furthermore, $(\bullet)^{+}$ is a special case of $\boldsymbol{P}_{\Lambda}$, in which $\Lambda = \mathbb{R}_{+}^{2n}$ is the nonnegative quadrant of $\mathbb{R}^{2n}$. Therefore, $\boldsymbol{P}_{\bar{\Omega}}$ is a projection operator to closed set $\bar{\Omega}$. Based

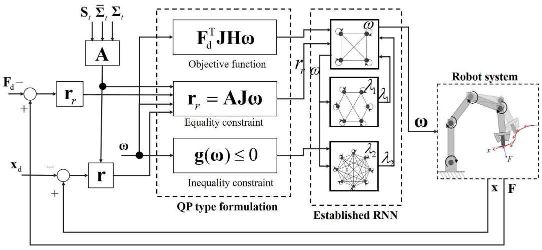


Fig. 2. The control structure of the proposed projection RNN based control strategy.

on Lemma 1, the proposed neural network (23) is stable and will globally converge to the optimal solution of (20). The proof is completed. □

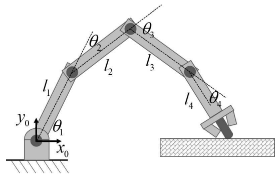
Remark 4: In this part, due to the limited space, the proof cannot be expanded in detail. When $\boldsymbol{F}(\boldsymbol{\xi})$ is a monotone function of $\boldsymbol{\xi}$, the rest of proof can be done in two steps: firstly, the boundedness of $\boldsymbol{\xi}$ should be proved, which plays an important role in ensuring the boundedness of $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$ and $\boldsymbol{\tau}$. Secondly, The convergence of the RNN should be proved. Similar deduction can be found in literature [34], [39].

V. Simulative Verifications and Comparisons

In this section, numerical results based on a planar 4-DOF manipulator is carried out to show validity of the established control scheme. First of all, we will check the control performance when the end-effector is expected to move to a specified point. Secondly, position-force control along a straight line is carried out, in which time-varying desired contact force is considered, and the optimization performance is verified by comparing with the scheme without optimization (by letting $\boldsymbol{H}^{\mathrm{T}}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{F}_{\mathrm{d}}$ in (23a) be 0). Finally, comparative simulations between the controller (23) with Jacobian-matrix-pseudo-inverse (JMPI) based methods are conducted to show the superiority of the proposed control scheme.

A. Simulation Setup

Fig. 3(a) gives a brief introduction of a 4-DOF planar redundant manipulator to be simulated in this paper, and D-H parameters of the robot is shown in Fig. 3(b). We have $m = 2$ and $n = 4$. A contact surface in the workspace can be described as $y = 0$, the end-effector can move freely along the horizontal axis, and the desired contact force $\boldsymbol{F}_{\mathrm{d}}$ is aligned with the vertical direction. It is notable that when the manipulator is planar, the coordination system $R_0(0_0, \boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{z}_0)$ degenerates to $R_0(0_0, \boldsymbol{x}_0, \boldsymbol{y}_0)$, then the matrices Σ_f and $\bar{\Sigma}_f$ are selected as $\Sigma_f = diag([0,1])$ and $\bar{\Sigma}_f = diag([1,0])$, respectively. The stiffness coefficient k_f is set to be 1000N/mm. Control positive control gains are set as $\alpha = 10$, $\beta = 10$, $k = 8$, $\epsilon = 0.005$, respectively. Physical constraints of joint angles, velocities and torque are defined as $\boldsymbol{\theta}^{\min} = [-2,-2,-2,-2]^{\mathrm{T}}$ rad, $\boldsymbol{\theta}^{\max} = [2,2,2,2]^{\mathrm{T}}$ rad, $\dot{\boldsymbol{\theta}}^{\min} = [-2,-2,-2,-2]^{\mathrm{T}}$ rad/s, $\dot{\boldsymbol{\theta}}^{\max} = [2,2,2,2]^{\mathrm{T}}$ rad/s, $\boldsymbol{\tau}^{\min} = [-10,-10,-10,-10]^{\mathrm{T}}$ Nm, $\boldsymbol{\tau}^{\max} = [10,10,10,10]^{\mathrm{T}}$ Nm, respectively.



(a)

Joint	a_i(m)	α_i(rad)	d_i(m)
1	0.3	0	0
2	0.3	0	0
3	0.1	0	0
4	0.2	0	0

(b)

Fig. 3. The 4-DOF redundant manipulator to be simulated in this paper. (a) Physical structure of the 4-link robot manipulator. (b) D-H parameters.

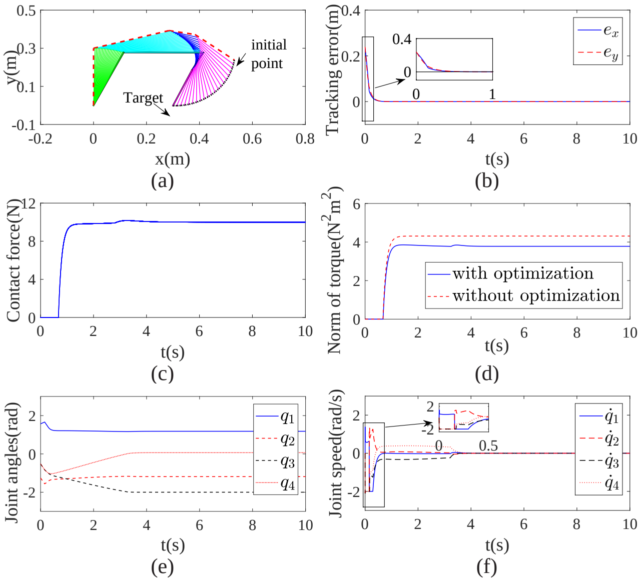


Fig. 4. Numerical results when the robot is controlled to offer constant force at a fixed point. (a) Profiles of the end-effector (black dashed line) and the corresponding joint configurations. (b) Profiles of position error. (c) Profiles of contact force. (d) Comparison of Euclidean norm of joint torque with and without optimization. (e) Profiles of joint angles. (f) Profiles of joint velocities.

B. Position-Force Control on A Fixed Point

In this section, joint torque optimization is introduced to make full use of redundancy resolution. The proposed position-force control scheme is firstly tested in a fixed point case, and then extended to dynamic cases.

In this simulation, the robot is wished to offer a constant contact force $\boldsymbol{F}_{\mathrm{d}} = [0,-10]^{\mathrm{T}}$ at a fixed point $\boldsymbol{x}_{\mathrm{d}} = [0.3, 0]^{\mathrm{T}}$. The initial values of joint angles are also $\boldsymbol{\theta}_0 = [1.57, -1.26, -0.52, -0.52]^{\mathrm{T}}$rad. Numerical results are shown in Fig. 4. At the beginning stage of simulation, the robot moves at its maximum speed (2rad/s), making the regulation error converge quickly. Then it slows down as the regulation error is small. At $t = 0.5s$, robot touches the surface, which leads to the emergence of contact force. Using the proposed controller, both control error of motion and force converge to 0 smoothly. Correspondingly, the Euclidean norm of joint torque also converges to a constant value ($3.7N^2m^2$). From Fig. 4(e) and (f), joint angles and velocities do net exceed their limits, showing that the proposed scheme could handle inequality constraints effectively. To further demonstrate the validity of the optimization scheme, comparative simulations without optimization is also carried out. The obtained Euclidean norm of joint torque without optimization is shown as the red dashed line ($4.3N^2m^2$ in stable state). After introducing joint torque optimization strategy, 16% off of torque consumption is achieved.

C. Position-Force Control Along A Straight Line

In this section, the robot is controlled to offer a constant contact force on the surface while tracking a given trajectory. The initial joint angles are selected as $\boldsymbol{\theta}_0 = [1.57, -1.26, -0.52, -0.52]^{\mathrm{T}}$ rad. The desired trajectory is defined as $\boldsymbol{x}_{\mathrm{d}} = [0.25 + 0.1cos(0.5t), 0]^{\mathrm{T}}$, and the contact force is defined as $\boldsymbol{F}_{\mathrm{d}} = [0, -1]^{\mathrm{T}}$N. In order to verify the torque optimization, we compare the performance of controller (23) with the scheme without optimization. Simulation results are shown in Fig. 5. When simulation begins ($t < 0.5s$), the position error is big, and there is no contact between the robot and surface. Correspondingly, both contact force and the result torque are zero. Under the projection RNN based controller (23), joint velocities reach the maximum value, the end-effector approaches to the surface rapidly from the initial position, and the tracking error converges to zero quickly, which is very similar to the previous simulation. After $t = 2s$, high precision trajectory tracking is realized by the control strategy, as well as the contact force. By comparing Fig. 5 (c),(d) and 5 (f), it can be found that the position and force tracking errors are quite similar,*i.e.*, the introduction of optimization scheme has little effect on the control errors. Fig. 5 (f) shows the comparison of Euclidean norms of joint torque with and without optimization. Correspondingly, $\boldsymbol{J}_{\boldsymbol{\tau}}$ decreases 16.2% from 142 to 119, showing the validity of the proposed scheme. It is notable that, after introducing optimization scheme, the 3^{rd} joint reaches its lower bound, which corresponds to the optimal joint configuration to minimize joint torque consumption. Dynamic change of joint configurations of both controllers are as shown in Fig. 5 (a) and (b).

D. Position-Force Control Along An Arc Surface

In this subsection, the end-effector is controlled to track a quarter-circular surface, which is centered at $[0.3, 0.3]^{\mathrm{T}}$m with radius 0.2m, and provide a constant force of 10N in the vertical direction. The initial values of joint angles are selected as $\boldsymbol{\theta}_0 = [1.5708, -0.9851, -1.1714, 0]^{\mathrm{T}}$rad. Numerical results are shown in Fig. (6). To further show the superiority of the proposed controller, comparison with JMPI based method is discussed. Using JMPI based method, the control command is obtained by calculating the pseudo inverse of Jacobian matrix, and torque optimization is done in the null space based on the idea (13). As shown in Fig. 6 (c), (d) and (e), both controller could realize the convergence of tracking errors and force error. The comparison of Euclidean norm of joint torque is shown in Fig. 6 (f), it can be observed that when $t < 10s$, controller (23) obtains better optimization performance than JMPI method, while $t > 10s$, JMPI method is better. By comparing Fig. 6 (g) and (h), when $t > 10s$, despite better better optimization performance JMPI method has achieved, the 3^{rd} joint based on JMPI method exceeds its lower bound, which means the controller fails in guaranteeing the safety of the robot system. Accordingly,controller (23) is capable of handling physical limitations while optimizing joint torque, which shows the superiority of the proposed control strategy. Dynamic change of joint configurations of both controllers are as shown in Fig. 6 (a) and (b).

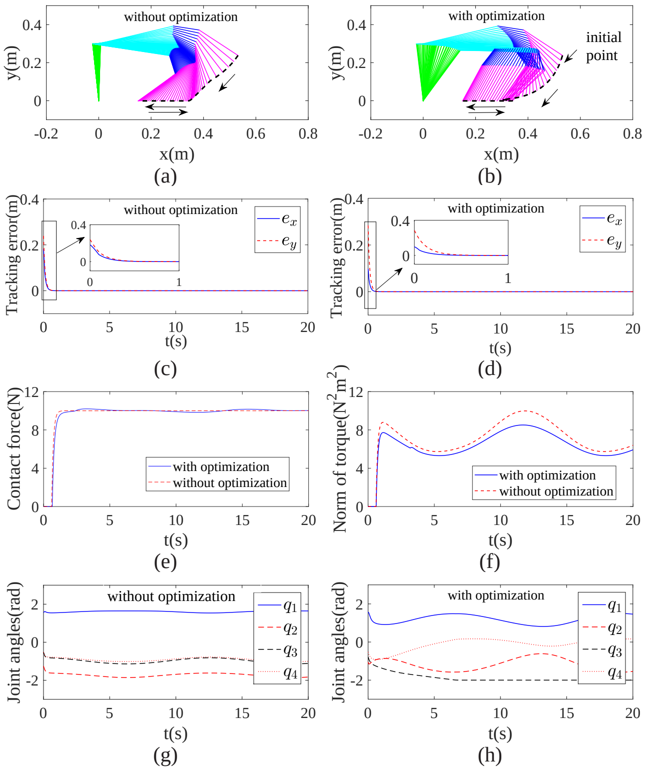


Fig. 5. Numerical results when tracking a straight line with and without torque optimization. (a) Profiles of the end-effector (black dashed line) and the corresponding joint configurations without optimization. (b) Optimized profiles of the end-effector (black dashed line) and the corresponding joint configurations without optimization. (c) Profiles of position error without optimization. (d) Optimized profiles of position error. (e) Comparison of contract force. (f) Comparison of Euclidean norm of joint torque with and without optimization. (g) Profiles of joint angles without optimization. (h) Optimized profiles of joint angles.

VI. Experimental Verification

In order to further show the efficiency of the developed control strategy, a group of experiments are conducted on a 7 DOF redundant manipulator KUKA iiwa. Parameters of iiwa can be found in [45].The control command is calculated on a host computer using ROS system, and then sent to robot controller to drive the robot. The end-effector is required to track certain trajectories on a horizontal surface, and provide a constant contact force along the vertical direction. At the same time, the orientation of the end-effector is expected to remain constant. Matrices Σ_f and $\bar{\Sigma}_f$ are selected as $\Sigma_f = diag([0, 0, 1])$ and $\bar{\Sigma}_f = diag([1, 0, 0])$, respectively. The robot's real physical constrains are $\boldsymbol{\theta}_i^{\min} = [-2.1]rad$, $\boldsymbol{\theta}_i^{\max} = [2.1]rad$, $\dot{\boldsymbol{\theta}}_i^{\min} = [-2]$, $\dot{\boldsymbol{\theta}}_i^{\max} = [2]$, in the interest of safety, in our experiments, we select the physical constraints as $\boldsymbol{\theta}^{\min}(i) = -2$ rad, $\boldsymbol{\theta}^{\max}(i) = 2$ rad, $\dot{\boldsymbol{\theta}}^{\min}(i) = -2$ rad/s, $\dot{\boldsymbol{\theta}}^{\max}(i) = 2$ rad/s, $\boldsymbol{\tau}^{\min}(i) = -10$ Nm, $\boldsymbol{\tau}^{\max}(i) = 10$ Nm for $i = 1, \cdots, 7$, respectively. The main purpose of selecting physical constraints is to verify the performance of ensuring inequality constraints while avoiding damage of physical

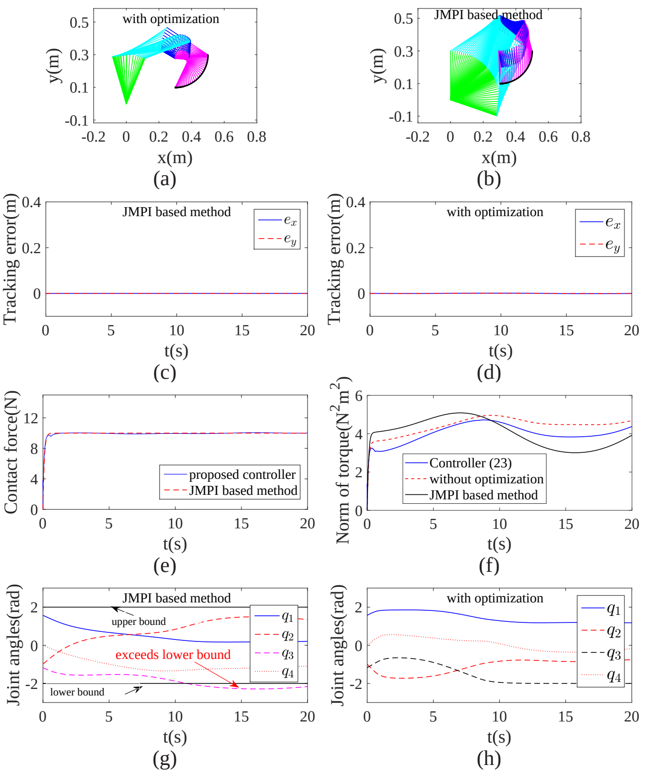


Fig. 6. Numerical results when tracking a time varying force command along an arc surface with and without torque optimization. (a) Profiles of the end-effector and the corresponding joint configurations without optimization. (b) Optimized profiles of the end-effector (black dashed line) and the corresponding joint configurations without optimization. (c) Profiles of position error without optimization. (d) Optimized profiles of position error. (e) Comparison of contract force. (f) Comparison of Euclidean norm of joint torque with and without optimization. (g) Profiles of joint angles without optimization. (h) Optimized profiles of joint angles.

mechanism of the robot. The initial joint is set as $\boldsymbol{\theta}_0 = [-0.118; 0.736; -0.004; -1.574; 0.004; 0.831; -0.909]$rad, and the corresponding position of the end-effector is $\boldsymbol{x}_0 = [0.56; -0.068; 0.046]$m. The control frequency is set as 100 Hz. It is remarkable that the value of ϵ would significantly affect the convergence of the RNN, $\epsilon = 0.02$ is selected in the same order as the control period.

Remark 5: From Eq. (23), it can be observed that the force control is realized by adjusting the joint velocities base on the RNN, which is consistent with the idea of admittance control. In our experiment, we assume that it provide an ideal response to the joint velocity command. It is remarkable that the uncertainties in the dynamic level such as friction and disturbances do affect the performance of position-force control in the outer loop, but these uncertainties can be suppressed by the closed-loop control mechanism of the controller itself.

A. Set-point Control

In this subsection, a set-point control is firstly conducted. The robot is controlled to provide a contact force $F_\mathrm{d} = [0; 0; 5]$N at the initial point. Experimental results are shown in Fig. 7. When experiment begins, the position and force error converge to 0 rapidly (7(a,b)), which is mainly due to the fact that the initial point is the same as the expected point. The joint angular velocities increase from zero, and reach the maximum value at $t = 2.5$s. After that, $\dot{\boldsymbol{\theta}}$ decrease to 0. Correspondingly, the robot adjusts its joint configuration to another fixed value (7(e)). As a result, the average norm of joint torque decreases from 3.2 to 2.8, which is a 12.5% reduction. It is also remarkable that during the robot moves, the position error, orientation error and force error are very small. The snapshots of the robot is given in Fig. 8, it can be observed that the robot rotates its shoulder-elbow-wrist joints significantly, and the end-effector is kept to be vertical. It is notable that beginning stage can be regarded as non-optimized case and the optimal configuration can be regarded as optimized results. The control errors in Fig. 9(a-c) also show the effectiveness of the optimization scheme.

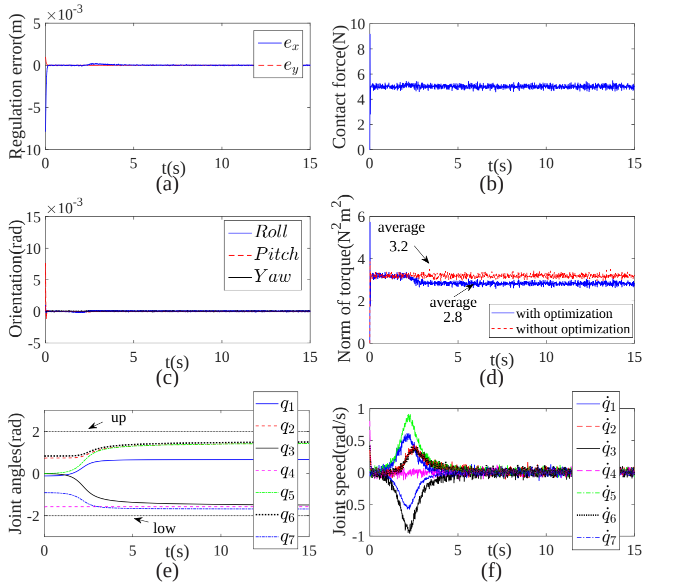


Fig. 7. Experimental results of the proposed controller in set-point control.(a) Profiles of the position errors. (b) Profiles of contact force. (c) Profiles of orientation error. (d) Profiles of norm of joint torque. (e)Profiles of joint angles. (f) Profiles of joint velocities.

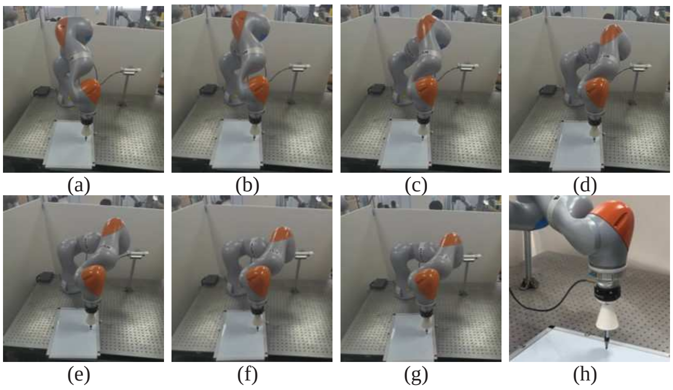


Fig. 8. Snapshots of set-point force control. (a) $t = 0$s. (b) $t = 1$s. (c) $t = 1.5$s. (d) $t = 3$s. (e) $t = 4$s. (f) $t = 5$s. (g) $t = 6$s. (h) The orientation of end-effector.

B. Force Control for Repeatable Trajectory and Comparison with JMPI based Method

In this subsection, position-force control on a repeatable trajectory is conducted. The repeatable path is defined as $\boldsymbol{x}_\mathrm{d} = [0.56 - 0.1sin(t/2); -0.168 + 0.1cos(t/2); 0.0458]$m,

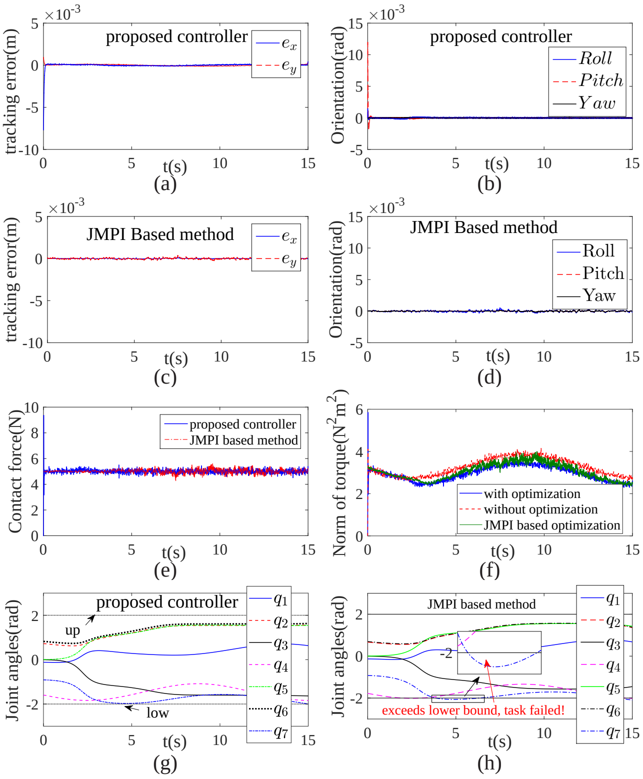


Fig. 9. Experimental results of the proposed controller along a circular trajectory and comparison with JMPI based Method. (a) Profiles of the position errors of the proposed controller. (b) Profiles of orientation error of the proposed controller. (c) Profiles of the position errors of JMPI based method. (d) Profiles of orientation error of JMPI based method. (e) Comparison of contact force. (f) Comparison of Euclidean norm of joint torque. (g) Profiles of joint angles of the proposed controller. (h) Profiles of joint angles of JMPI based method.

and contact force command is also $\boldsymbol{F}_{\rm d} = [0;0;5]$N. Experimental results are shown in Fig. 9. The control errors are given in Fig. 9(a)-(b), the maximum values during the control process are about 1×10^{-3}m, 1F and 2×10^{-3}rad, respectively. At the beginning stage, the motion of the robot is similar to that of the previous fixed-point experiment. It is notable that at $t=4$s, the 7th joint reaches its lower limit. After introducing joint torque optimization scheme, $\boldsymbol{J_\tau}$ reduces from 40.3 to 36.2 with a 10.8% off. The snapshots are shown in Fig. 10. It can be observed from both Fig. 10 and Fig. 9(g) that the robot adjusts its joint configuration from its initial state to an optimal one(the centers of shoulder-elbow-wrist joints are horizontal.) It is remarkable that the optimal configuration is the same as the set-point case, which corresponds to the fact that if the contact surface is certain, the optimal configuration is unique.

To further verify the efficacy of the established controller (23), a comparative experiment based on JMPI method is carried out. Experimental results are as shown in Fig. 9(c),(d) and (h). Similar to the simulation results in Fig. 6(f) and (g), the 5^{th} joint exceeds its lower bound -2rad during $4<t<7$s, which means the tasks fails when JMPI based controller is used to realize optimal control problem in presence of physical constraints in real-time. Therefore, it is point worthy that comparing with traditional JMPI based methods, our controller shows great potential in handling inequality constraints while realizing optimal force-position control problem in realtime.

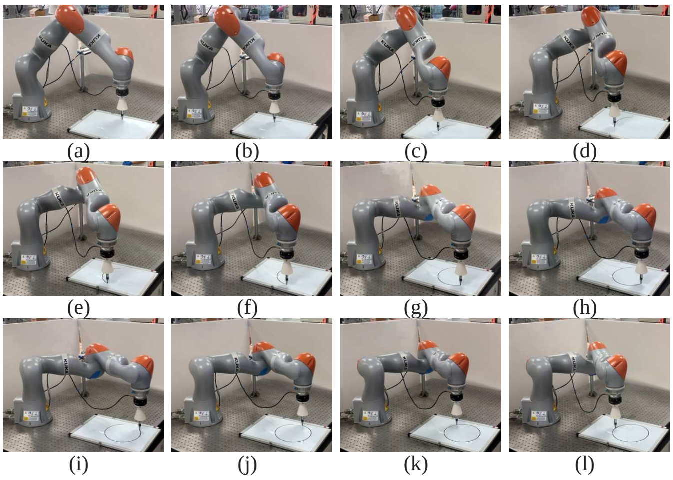


Fig. 10. Snapshots of force control along a circular trajectory. (a) $t=0$s. (b) $t=1$s. (c) $t=2$s. (d) $t=3$s. (e) $t=4.5$s. (f) $t=5.5$s. (g) $t=7$s. (h) $t=7.5$s. (i) $t=8.5$s. (j) $t=11.5$s. (k) $t=13$s. (l) $t=15$s.

C. Comparisons

In this subsection, comparisons between the proposed method with existing ones are shown to show the validity of this paper. As shown in Table I, using dynamic controllers such as [20], [21], [26], [28], [44], the force control is realized in internal loop based on system dynamics, the uncertain dynamics such as friction and disturbances are compensated by robust control or sliding mode control, *etc.* In [25], a sequence-to-sequence based method is introduced to robotic systems with contact model, which also considers physical limitations and performance optimization. In this paper, the motion-force controller is built in joint velocity level, which is based on the idea of admittance control. The main contribution is that the force control problem is transformed into an optimization problem for systems with redundant DOFs under multiple inequality constraints in realtime. The influence of uncertain dynamics is handled by the proposed controller in planning layer and the robot controller in execution layer. It is remarkable that this paper is an important expansion of projection RNN from traditional kinematic control [33], [34] to the field of motion-force control problems.

VII. Conclusions

This paper focuses on motion-force control problem for redundant manipulators, while physical constraints and torque optimization are taken into consideration. Firstly, tracking error and contact force are modelled in orthogonal spaces respectively, and then the control problem is turned into a QP problem, which is further rewritten in velocity level by rewriting objective function and constraints. To handle multiple physical constraints, a projection RNN based scheme is designed to solve the redundancy resolution online. Numerical and experimental results show the validity of the proposed control scheme. Before ending this paper, it is noteworthy that this is the first paper to deal with motion-force control of redundant manipulators in the framework of projection RNNs and redundant manipulators with force sensitivity.

TABLE I
COMPARISONS AMONG THE PROPOSED MOTION-FORCE CONTROL SCHEME AND EXISTING ONES

	Optimization Considered	RNN Based	Motion-force Control	Control Command	Pseudo-inverse Required	Handling Multiple Inequality Constraints	Chattering
This paper	Yes	Yes	Yes	Joint Velocity	No	Yes	No
Methods [33], [34]	Yes	Yes	No	Joint Velocity	No	Yes	No
Method [26]	No	No	Yes	Joint Torque	Yes	Restrictive‡	No
Methods [20], [21]	No	Yes	Yes	Joint Torque	Yes	No	No
Method [44]	No	Yes	No	Joint Torque	Yes	No	Yes
Method [25]	Yes	Yes	No	Joint Velocity	Yes	Yes	No

‡ In [26], the secondary task is defined to avoid physical limitations, therefore, only some certain kinds of inequalities can be handled by [26].

REFERENCES

[1] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng and C. Su, “Finite-time Convergence Adaptive Fuzzy Control for Dual-Arm Robot with Unknown Kinematics and Dynamics,” *IEEE Trans. Fuzzy Syst.*, vol. 27, no.3, pp. 574-588, Mar. 2019.

[2] J. Na, Q. Chen, X. Ren, Y. Guo, “Adaptive Prescribed Performance Motion Control of Servo Mechanisms with Friction Compensation,” *IEEE Trans. Ind. Electron.*, vol. 61, no.1, pp. 486-494, Jan. 2013.

[3] H. Wang, P. Shi, H. Li and Q. Zhou, “Adaptive Neural Tracking Control for a Class of Nonlinear Systems With Dynamic Uncertainties,” *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3075-3087, Sep. 2017.

[4] L. Cheng, Z. Hou and M. Tan, “Adaptive Neural Network Tracking Control for Manipulators with Uncertain Kinematics, Dynamics and Actuator Model,” *Automatica*, vol. 45, No. 10, pp. 2312-2318, Oct. 2009.

[5] L. Cheng, W. Liu, Z. Hou, Z. Li and M. Tan, “Neural Network Based Nonlinear Model Predictive Control for Piezoelectric Actuators,” *IEEE Trans. Ind. Electron.*, vol. 62, No. 12, pp. 7717-7727, Jun. 2015.

[6] H. Wang, W. Liu, J. Qiu and P. Liu, “Adaptive Fuzzy Decentralized Control for A Class of Strong Interconnected Nonlinear Systems with Unmodeled Dynamics,” *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 836-846, Oct. 2018.

[7] 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.

[8] J. Na, X. Ren and D. Zheng, “Adaptive Control for Nonlinear Pure-feedback Systems with High-order Sliding Mode Observer,” *IEEE Trans. Neur. Net. Lear.*, vol. 24, no. 3, pp. 370-382, Mar. 2013.

[9] C. Yang, Y. Jiang, W. He, J. Na, Z. Li and B. Xu, “Adaptive Parameter Estimation and Control Design for Robot Manipulators with Finite-Time Convergence,” *IEEE Trans. Ind. Electron.*, vol. 65, no.10, pp. 8112-8123, Nov. 2018.

[10] Y. Liu and S. Tong, “Barrier Lyapunov Functions-based Adaptive Control for A Class of Nonlinear Pure-feedback Systems with Full State Constraints,” *Automatica*, vol. 64, no. 1, pp. 70-75, Jan. 2016.

[11] 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.

[12] W. Xu, J. Zhang, B. Liang and B. Li, “Singularity Analysis and Avoidance for Robot Manipulators with Non-Spherical Wrists,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 277-290, Jan. 2016.

[13] N. Hogan, “On the Stability of Manipulators Performing Contact Tasks,” *IEEE J. Robot. Automat.*, vol. 4, no. 6, pp. 677-686, Dec. 1988.

[14] O. Khatib, “A Unified Approach for Motion and Force Control of Robot Manipulators: The Oerational Space Formulation,” *IEEE J. Robot. Automat.*, vol. 3, no. 1, pp. 43-53, Feb. 2003.

[15] K. Kiguchi and T. Fukuda, “Position/force Control of Robot Manipulators for Geometrically Unknown Objects Using Fuzzy Neural Networks,” *IEEE Trans. Ind. Electron.*, vol. 47, no. 3, pp. 641-649, Jun. 2000.

[16] N. Hogan, “Impedance Control: An Approach to Manipulation, Parts I, II, III,” in *J. Dyn. Syst., Meas. Control.*, vol. 107, no. 1, pp. 1-23, Jun. 1985.

[17] J. Craig, P. Hsu and S. Sastry “Adaptive Control of Mechanical Manipulators,” in *Proc. IEEE Int. Conf. Robotics and Automation.*, San Francisco, Ca, USA., Apr. 1986, pp. 190-195.

[18] Z. Li, Z. Huang, W. He and C. Su, “Adaptive Impedance Control for an Upper Limb Robotic Exoskeleton Using Biological Signals,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 1664-1674, Feb. 2017.

[19] C. Yang, K. Huang, H. Cheng, Y. Li and C. Su, “Haptic Identification by ELM-Controlled Uncertain Manipulator,” *IEEE Trans. Cybern.*, vol. 47, no. 8, pp. 2398-2409, Feb. 2017.

[20] S. Jung and T. Hsia, “Neural Network Impedance Force Control of Robot Manipulator,” *IEEE Trans. Ind. Electron.*, vol. 45, no. 3, pp. 451-461, Jun. 1998.

[21] S. Jung and T. Hsia, “Robust Neural Force Control Scheme Under Uncertainties in Robot Dynamics and Unknown Environment,” *IEEE Trans. Ind. Electron.*, vol. 47, no. 2, pp. 403-412, Apr. 2000.

[22] T. Mai and Y. Wang, “Adaptive Force/Motion Control System Based on Recurrent Fuzzy Wavelet CMAC Neural Networks for Condenser Cleaning Crawler-Type Mobile Manipulator Robot,” *IEEE Trans. Contr. Syst. Technol.*, vol. 22, no. 5, pp. 1973-1982, Sep. 2014.

[23] Y. Xia, J. Wang and L. Fok, “Grasping-Force Optimization for Multi-fingered Robotic Hands Using a Recurrent Neural Network,” *IEEE J. Robot. Automat.*, vol. 20, no. 3, pp. 549-554, Jun. 2004.

[24] K. Kutsuzawa, S. Sakaino, and T. Tsuji, “Sequence-to-Sequence Models for Trajectory Deformation of Dynamic Manipulation,” in *Proc. Annu. Conf. IEEE Ind. Electron. Soc.*, pp. 5227-5232, Nov, Jun. 2017.

[25] K. Kutsuzawa, S. Sakaino and T. Tsuji, “Sequence-to-Sequence Model for Trajectory Planning of Nonprehensile Manipulation Including Contact Model,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3606-3613, Oct. 2018.

[26] B. Nemec and L. Zlajpah, “Force Control of Redundant Robots in Unstructured Environment,” *IEEE Trans. Ind. Electron.*, vol. 49, no. 1, pp. 233-240, Feb. 2002.

[27] R. Patel, H. Talebi, J. Jayender and F. Shadpey, “A Robust Position and Force Control Strategy for 7-DOF Redundant Manipulators,” *IEEE/ASME Trans. Mechatron.*, vol. 14, no. 5, pp. 575-589, Oct. 2009.

[28] R. Patel and F. Shadpey, “Control of Redundant Robot Manipulators,” *Lecture Notes in Control, Information Sciences.*, Springer, 2005.

[29] W. He, Z. Yan and C. Sun, “Adaptive Neural Network Control of A Marine Vessel with Constraints Using The Symmetric Barrier Lyapunov Function,” *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1641-1651, Jul. 2017.

[30] Y. Zhang, J. Wang and Y. Xia, “A Dual Neural Network For Redundancy Resolution of Kinematically Redundant Manipulators Subject to Joint Limits and Joint Velocity Limits,” *IEEE Trans. Neural Networks*, vol. 14, no. 4, pp. 658-667, May. 2003.

[31] L. Jin and Y. Zhang, “G2-type SRMPC Scheme For Synchronous Manipulation of Two Redundant Robot Arms,” *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 153-164, Feb. 2015.

[32] B. Cai and Y. Zhang, “Different-Level Redundancy-Resolution and Its Equivalent Relationship Analysis for Robot Manipulators Using Gradient-Descent and Zhang ’s Neural-Dynamic Methods,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3146-3155, Aug. 2012.

[33] S. Li, Y. Zhang and L. Jin, “Kinematic Control of Redundant Manipulators Using Neural Networks,” *IEEE Trans. Neur. Net. Lear.*, vol. 28, no. 10, pp. 2243-2254, Oct. 2017.

[34] L. Jin, S. Li, H. La and X. Luo, “Manipulability Optimization of Redundant Manipulators Using Dynamic Neural Networks,” *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 4710-4720, Aug. 2017.

[35] S. Li, S. Chen, B. Liu, Y. Li and Y. Liang, “Decentralized Kinematic Control of A Class of Collaborative Redundant Manipulators via Recurrent Neural Networks,” *Neurocomputing.*, vol. 91, no. 1, pp. 1-10, Aug. 2012.

[36] W. He, Z. Yan, Y. Sun Y. Ou and C. Sun, “Neural-Learning-Based Control for a Constrained Robotic Manipulator With Flexible Joints,” *IEEE Trans. Neur. Net. Lear.*, vol. 29, no. 12, pp. 5993-6003, Dec. 2018.

[37] Y. Zhang, S. Chen, S. Li and Z. Zhang, “Adaptive Projection Neural Network for Kinematic Control of Redundant Manipulators With Unknown Physical Parameters,” *IEEE Trans. Ind. Electron.*, , vol. 65, no. 6, pp. 4909-4920, Jun. 2018.

[38] Z. Xu, S. Li, X. Zhou, Y. Wu, T. Cheng and D. Huang, “Dynamic Neural Networks Based Kinematic Control for Redundant Manipulators

with Model Uncertainties," *Neurocomputing.*, vol. 329, no. 1, pp. 255-266, Jan. 2019.

[39] X. Gao, "Exponential Stability of Globally Projected Dynamic Systems," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 426-431, Mar. 2003.

[40] Y. Xia, G. Feng and J. Wang, "A Novel Recurrent Neural Network for Solving Nonlinear Optimization Problems With Inequality Constraints," *IEEE Trans. Neural Netw.*, vol. 19, no. 8, pp. 1340-1353, Aug. 2008.

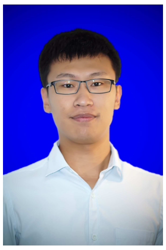
[41] X. Hu and J. Wang, "A Recurrent Neural Network for Solving a Class of General Variational Inequalities," *IEEE Trans. Syst., Man, Cybern. B*, vol. 37, no. 3, pp. 528-539, Jun. 2007.

[42] Y. Xia and J. Wang, "Global Exponential Stability of Recurrent Neural Networks for Solving Optimization and Related Problems," *IEEE Trans. Neural Netw.*, vol. 11, no. 4, pp. 1017-1022, Jul. 2000.

[43] Y. Xia and J. Wang, "A Recurrent Neural Network for Solving Nonlinear Convex Programs Subject to Linear Constraints," *IEEE Trans. Neural Netw.*, vol. 16, no. 2, pp. 379-386, Mar. 2005.

[44] Q. Xu, "Adaptive Discrete-Time Sliding Mode Impedance Control of a Piezoelectric Microgripper," *IEEE Trans. Robot.*, vol. 29, no. 3, pp. 663-673, Jun. 2013.

[45] Z. Xu, S. Li, X. Zhou and T. Cheng, "Dynamic Neural Networks Based Adaptive Admittance Control for Redundant Manipulators with Model Uncertainties," *Neurocomputing.*, vol. 357, no. 1, pp. 271-281, Feb. 2019.



Zhihao Xu (M'17) received the B.E. degree and the Ph.D. degree both in School of Automation, Nanjing University of Science and Technology, Nanjing, China, in 2010 and 2016, respectively. He is currently a Postdoctoral Fellow at the Robotics team, Guangdong Institute of Intelligent Manufacturing, Guangzhou, China. His main research interests include neural networks, force control, and intelligent information processing.

Shuai Li (STM'08-M'14-SM'17) received B.E. degree in precision mechanical engineering from Hefei University of Technology, Hefei, China, in 2005, M.E. degree in automatic control engineering from University of Science and Technology of China, Hefei, in 2008, and Ph.D. degree in electrical and computer engineering from Stevens Institute of Technology, Hoboken, NJ, USA, in 2014. He is currently an Associate Professor (Reader) at Swansea University, Wales, UK, leading the Robotic Lab, conducting research on robot manipulation and impedance control, multi-robot coordination, distributed control, intelligent optimization and control, and legged robots.

Xuefeng Zhou (M'14) received the Master degree and the Ph.D. degree both in Mechanical Engineering, South China University of Technology, Guangzhou, China, in 2006 and 2011, respectively. He is currently the director of Robotics team in Guangdong Institute of Intelligent Manufacturing, Guangzhou, China. He won the Best Student Paper Award at the 2010 IEEE International Conference on Robotics and Biomimetics. HIs research interests include neural networks, force control, and intelligent information processing.

Songbin Zhou received the M.S. degree in software engineering from South China University of Technology, Guangzhou, China, in 2004 and the Ph.D. degree in electronic & mechanical engineering from South China University of Technology, Guangzhou, China, in 2008. Since 2012, he has been a Professor with the Institute of Intelligent Manufacturing, Guangzhou, China. He is the leader of the intelligent sensing group. His research interest includes acoustic sensing, hyperspectral sensing and machine learning.

Taobo Cheng received the Ph.D. degree in Welding Engineering, South China University of Technology, Guangzhou, China, in 1998. He is currently the director of Guangdong Institute of Intelligent Manufacturing. His current research interests include intelligent manufacturing technology, automation and information technology.

Yisheng Guan (M'09) received the M.S. degree from Harbin Institute of Technology, Harbin, China, in 1990, and the Ph.D. degree from Beijing University of Aeronautics and Astronautics, Beijing, China, in 1998, both in mechanical engineering. He conducted research as a Postdoctoral Fellow in the Department of Computing Science, University of Alberta, Edmonton, AB, Canada, from 1998 to 2000, and in the Intelligent Systems Institute, AIST, Tsukuba, Japan, with a fellowship from the Japan Society for the Promotion of Science, from 2003 to 2005. He is currently a Professor and the Lead Biomimetics and Intelligent Robotics Laboratory in the School of Mechanical and Electronic Engineering, Guangdong University of Technology, Guangzhou, China. His research interests include biomimetic robotics, modular robotics, humanoid robotics, and soft robotics, human-robot collaboration, and robot learning.