Tricriteria Optimization-Coordination Motion of Dual-Redundant-Robot Manipulators for Complex Path Planning

Zhijun Zhang, *Member, IEEE*, Yujun Lin, Shuai Li, *Member, IEEE*, Yuanqing Li, *Fellow, IEEE*, Zhuliang Yu, and Yamei Luo

***Abstract*— To remedy the discontinuity phenomenon existing in the infinity-norm velocity minimization (INVM) scheme, prevent the occurrence of high joint velocity, and decrease the joint-angle drift of redundant robot manipulators, a new type of tricriteria optimization-coordination-motion (TCOCM) scheme is proposed and investigated for dual-redundant-robot manipulators to track complex end-effector paths. Besides, the proposed scheme considers joint physical constraints (i.e., joint-angle limits and joint-velocity limits) and guarantees the joint velocity to approach zero at the end of tasks. Such a new-type TCOCM scheme combines the minimum velocity norm (MVN), repetitive motion planning (RMP), and INVM solutions via two weighting factor, and thus is termed MVN-RMP-INVM-TCOCM scheme. The proposed scheme consists of two subschemes, i.e., subscheme for the left robot manipulator and subscheme for the right robot manipulator. To control the dual arms simultaneously and collaboratively, two subschemes are reformulated as two general quadratic programming (QP) problems and further unified into one QP formulation. The unified QP problem is then solved by a simplified linear-variational-inequalities-based primal–dual neural network solver. Computer simulation results based on dual PUMA560 robot manipulators are illustrated to substantiate the advantage, efficacy, and applicability of the proposed MVN-RMP-INVM-TCOCM scheme to resolve the redundancy of dual-robot manipulators.**

***Index Terms*— Coordination control, dual arms, quadratic programming, redundant manipulators, tricriteria optimization.**

Manuscript received October 3, 2016; revised March 25, 2017; accepted May 10, 2017. Date of publication June 13, 2017; date of current version June 11, 2018. Manuscript received in final form May 20, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61603142, Grant 91420302, and Grant 61633010), in part by the Scientific Research Starting Foundation, South China University of Technology (973 Program) under Grant 2015CB351703, in part by the Guangdong Natural Science Foundation under Grant 2014A030312005, in part by the Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, in part by the 2016 Guangdong College Student Climbing Program Project under Grant pdjh2017b0041, and in part by the National Undergraduate Innovative and Entrepreneurial Training Programs under Grant 201610561192, Grant 201710561205, and Grant 201710561204. Recommended by Associate Editor S. Pirozzi. *(Corresponding author: Zhijun Zhang.)*

Z. Zhang, Y. Lin, Y. Li, and Z. Yu are with the School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China, and also with the Guangzhou Key Laboratory of Brain Computer Interface and Applications, Guangzhou 510640, China (e-mail: drzhangzhijun@gmail.com).

S. Li is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong.

Y. Luo is with the Patent Examination Cooperation Center of the Patent Office, State Intellectual Property Office, Guangdong 510530, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2017.2709276

I. Introduction

RECENTLY, robot manipulators have applied to many fields, such as industrial manufacturing/assembly line [1], [2] and medical instrument [3].

In contrast to nonredundant robot manipulators, a redundant robot manipulator has more degrees of freedom (DOFs) required to perform a specific end-effector task [4]–[6]. With redundant DOFs, the robot manipulator cannot only achieve the primary tasks of the end-effector but can also accomplish some additional and useful objectives, such as avoidance of joint limits [7], obstacles [8]–[10], and singularities [11]. Therefore, a redundant robot manipulator can achieve complex tasks more flexibly and have more potential applications. Dual-robot manipulators cannot only complete some normal tasks, but can also finish some complex and coordination tasks, which the single robot manipulator cannot finish. There is an increasing trend of dual-manipulator robots being used in vast applications, such as picking and moving heavy boxes [12], and cooking [13].

Redundancy-resolution problem, or termed inverse-kinematic problem, is a fundamental issue in the field of operating the redundant robot manipulators. Since there exist multiple solutions to the inverse-kinematic problem of a redundant robot manipulator, a variety of schemes have been proposed and exploited. The traditional solution to the redundancy-resolution problem is the pseudoinverse formulation that contains a homogeneous solution and a specific minimum-norm solution [14]. However, the traditional pseudoinverse method cannot consider the inequality constraint and need to compute the inverse/pseudoinverse of matrices, which would cost lots of time. To overcome these deficiencies, some quadratic programs-based optimization schemes have been proposed and investigated in recent years [8], [9], [15]–[27]. Among the existing schemes, some performance criteria are designed at velocity or levels [8], [9], [15]–[18], [28]. However, only single criterion is considered in these works.

In order to achieve multiple optimization design, a bicriteria optimization scheme was proposed in the past few years [22], [23], [25]. They were verified effective when executed a simple task based on a single redundant robot manipulator. As the growing demand of the complex tasks, dual-robot manipulators have been widely applied, and thus, it is very urgent to solve their coordinated control and planning

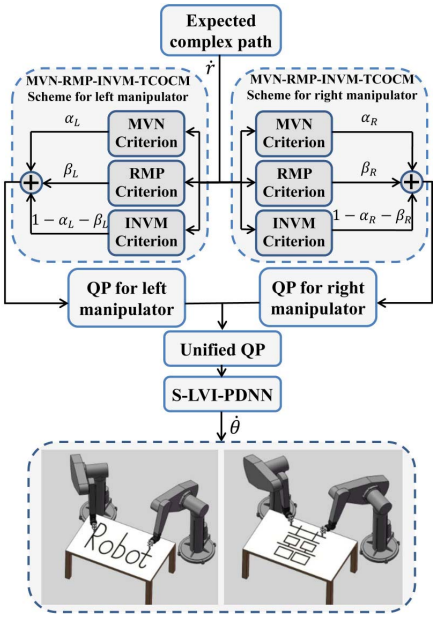


Fig. 1. System structure of TCOCM scheme for dual-redundant-robot-manipulators motion control.

problems [29]. Some novel control and planning strategies, such as force or impedance control, have been developed for extensive applications [30]. However, these methods relied on force sensors and were mainly for nonredundant manipulators. They could not be easily integrated in a unified optimization architecture or may require advanced integrating skills. To cooperatively control dual-redundant robot manipulators in a unified architecture, we have proposed a dual-arm cyclic-motion generation (DACMG) scheme, but it was single criterion optimization strategy [31]. According to our previous experience, a novel tricriteria optimization-coordination-control (TCOCM) scheme for dual-redundant-robot manipulators is proposed in this paper. The system structure of the proposed TCOCM scheme for dual-redundant-robot-manipulators motion control is shown in Fig. 1.

The remainder of this paper is organized into five sections. In Section II, related work is discussed. In Section III, two subschemes of the left and right redundant robot manipulators are constructed. In Section IV, the two subschemes are then reformulated into two QP problems, respectively, and then unified into a unified QP problem. The unified QP problem is solved by using the S-LVI-PDNN. Comparisons and simulation results of minimum velocity norm (MVN) scheme, INVM scheme, MVN-INVM bicriteria optimization-coordination-motion (BCOCM) scheme, and the proposed MVN-repetitive motion planning (RMP)-INVM-TCOCM scheme performed on dual-redundant-robot manipulators are illustrated in Section V. The conclusions are drawn in Section VI.

Before ending this section, the main contributions of this paper are listed as follows.

1) A novel TCOCM scheme is proposed for dual-redundant-robot manipulators to track complex end-effector tasks. This TCOCM scheme can effectively remedy the discontinuity phenomenon existing in pure INVM scheme and prevent the occurrence of high joint velocity. Besides, such a new type scheme cannot only remedy the joint-angle drift but also guarantee the joint velocity to approach zero at the end of the motion. In addition, the proposed scheme can consider both the joint-angle limits and the joint-velocity limits.
2) Compared with the single-criterion and bicriteria schemes, the proposed TCOCM scheme is more adjustable and flexible by adjusting the weights of MVN criteria, INVM criteria, and RMP criteria if needed.
3) Computer simulation results based on dual-robot manipulators with two complex path planning are conducted to demonstrate the efficacy, applicability, and superiority of the MVN-RMP-INVM-TCOCM scheme on the redundancy of dual-robot manipulators.

II. Related Work

Due to massive applications of collaborative operation in manufacturing and home service, motion planning and control of dual-redundant-robot manipulators attract more and more attention in recent years [30]. Most of the previous related research focused on cooperative robot manipulators holding an object with fixed grasp points [32] or an object by means of contact points or contact areas [33]. Force control strategies were mainly applied in such situations. For instance, Cao *et al.* [34] proposed a coordinated control method with hybrid force/position. Jiang *et al.* [35] presented a coordinated control method based on master–slave hybrid force. Actually, the hybrid-force/position-based control strategy is mainly based on the decomposition between the force and motion control loops. Impedance control is another method for dealing with control of dual-redundant manipulators. Ren *et al.* [36] proposed an object impedance control scheme with biomimetic adaption for dual-arm cooperation. However, most of the force/position control methods are applied for object manipulation and performed on nonredundant robot manipulators.

Considering coordinately tracking the end-effector paths and using dual-redundant-robot manipulators, optimization methods are preferred [31]. In the past few years, some optimization schemes have been proposed, such as MVN [8], minimum kinetic energy [15], RMP [16], [17], [37], infinity-norm velocity minimization (INVM) [18], minimum acceleration norm (MAN) [9], minimum torque norm (MTN) [19], inertia-inverse weighted torque [19], infinity-norm acceleration minimization [20], and infinity-norm torque minimization (INTM) [21].

Since single criterion optimization schemes cannot satisfy multiple requirements in practical applications, multicriteria minimization schemes are needed. For instance, to consider both the velocity minimization and infinity-norm velocity minimization, Zhang *et al.* [22] presented a bicriteria velocity minimization scheme (combining MVN with INVM criteria). Baratcart *et al.* [20] implemented a two-norm and infinity-norm continuous switching scheme on a redundant robot. Another example is the bicriteria torque

minimization scheme, which considers both MTN and INTM criteria) [23]. Besides, Cai and Zhang presented a bicriteria scheme combining MKN and INVM criteria [24]. In addition, Guo and Zhang *et al*. investigated some different level bicriteria minimization schemes [25]–[27]. For instance, MVN-MAN bicriteria minimization scheme considers both the MVN criterion (joint-velocity level) and MAN criteria (joint-acceleration level) [25]. MVN-INTM bicriteria minimization scheme considers both the MVN (joint-velocity level) and INTM criteria (joint-torque level) [26], and MVN-MTN bicriteria minimization scheme considers both the MVN (joint-velocity level) and MTN criteria (joint-torque level) [27]. However, the existing bicriteria schemes do not consider the repetitive motion condition. In practical applications, it needs to adjust the state of the robot manipulator to the initial positions. In addition, the above schemes are only verified with some simple paths planning, such as straight lines and circles, but few efforts of complex path planning are considered. Moreover, all these schemes are only exploited for single redundant robot manipulator, and very few even no tricriteria minimization scheme was considered before.

Coordinated motion planning and control of dual-robot manipulators have been widely studied in recent years [29], [31], [38]–[41]. Jin and Zhang [38] presented a G2-Type simultaneous RMP and control scheme for two three-links planar robot arms. This scheme is developed for a planar link manipulator and only works in a 2-D space. Lee *et al.* [39] presented a method of implementing impedance control (with inertia, damping, and stiffness terms) on a dual-arm system, which is achieved by using the relative Jacobian technique. This scheme does not consider the inequality constraints. To cooperatively control dual arms of humanoid robot, we proposed a DACMG scheme by a neural-dynamic method [31]. This paper can be applied to the brain machine interface field in the future [42], [43]. According to our previous experience, the proposed scheme can be formulated as two subschemes, and then, two subschemes are reformulated as two general QP and further unified into one QP formulation. Finally, the unified QP problem is solved by an S-LVI-PDNN solver. In short, the advantages of the methods proposed in this paper are to exploit a TCOCM scheme for dual-redundant-robot manipulators.

III. Preliminaries and Scheme Formulation

In this section, some fundamental kinematics equations are first given hereinafter. Based on these kinematics equations, the MVN-RMP-INVM-TCOCM scheme of the left and right robot manipulators is then proposed.

Forward kinematics mappings of the left and right robot manipulators of dual arms are

$$r_L = f_L(\theta_L) \tag{1}$$

$$r_R = f_R(\theta_R) \tag{2}$$

where $r_L \in R^m$ and $r_R \in R^m$ are the end-effector vectors of the left and right robot manipulators, respectively, and $\theta_L \in R^n$ and $\theta_R \in R^n$ are the joint vectors of the left and right robot manipulators, respectively. Besides, $f_L(\cdot)$ and $f_R(\cdot)$ are smooth nonlinear functions and we can obtain them once the structure and parameters of the dual-robot manipulators are known. In addition, n is the dimension of the joint space of one robot manipulator, and m is the dimension of end-effector Cartesian space. For PUMA560 robot used in this paper, $n = 6$. If we just consider the position vector of the robot manipulators, $m = 3$ and the redundancy of one robot manipulator is $n - m = 3$. Equations (1) and (2) are usually computed at the velocity level, and they are rewritten as

$$J_L(\theta_L)\dot{\theta} = \dot{r}_L \tag{3}$$

$$J_R(\theta_R)\dot{\theta} = \dot{r}_R \tag{4}$$

where $J_L(\theta_L) \in R^{m \times n}$ and $J_R(\theta_R) \in R^{m \times n}$ are the Jacobian matrixes of the left and right robot manipulators, and defined as $J_L(\theta_L) = \partial f_L(\theta_L)/\partial\theta_L$ and $J_R(\theta_R) = \partial f_R(\theta_R)/\partial\theta_R$. $\dot{r}_L \in R^m$ and $\dot{r}_R \in R^m$ denote end-effector velocity vectors of the left and right robot manipulators, respectively. Note that $J_R(\theta_L)$ and $J_R(\theta_R)$ can be abbreviated to J_L and J_R for simplicity in this paper.

A. MVN-RMP-INVM-TCOCM Schemes of the Left and Right Manipulators

For simplicity, the MVN-RMP-INVM-TCOCM scheme of one robot manipulator is first proposed.

In order to integrate the optimization criteria of the MVN [8], the RMP [16], and INVM [18], a tricriteria optimization index is designed as

$$\begin{aligned}\min\ & \alpha_{L/R}\|\dot{\theta}_{L/R}\|_2^2/2 + \beta_{L/R}\|\dot{\theta}_{L/R} + c_{L/R}\|_2^2/2 \\ & + (1 - \alpha_{L/R} - \beta_{L/R})\|\dot{\theta}_{L/R}\|_\infty^2/2\end{aligned} \tag{5}$$

where $\|\cdot\|_2$ and $\|\cdot\|_\infty$ denote a two-norm and an infinite norm of a vector, respectively. In the second term of (5), $c_{L/R} = \lambda_{L/R}(\theta_{L/R} - \theta_{L/R}(0))$, and $\lambda_{L/R}$ is a positive design parameter. $\alpha_{L/R}$, $\beta_{L/R}$ and $(1 - \alpha_{L/R} - \beta_{L/R})$ are weighting parameters, which are used to adjust the weights of MVN, RMP, and INVM, respectively.

For the left robot manipulator, considering the forward kinematics equation, joint-angle constraint, and joint-velocity constraint, the MVN-RMP-INVM-TCOCM scheme can be formulated as

$$\begin{aligned}\min\ & \alpha_L\|\dot{\theta}_L\|_2^2/2 + \beta_L\|\dot{\theta}_L + c_L\|_2^2/2 \\ & + (1 - \alpha_L - \beta_L)\|\dot{\theta}_L\|_\infty^2/2\end{aligned} \tag{6}$$

$$\text{s.t. } J_L\dot{\theta}_L = \dot{r}_L \tag{7}$$

$$\theta_L^- \leqslant \theta_L \leqslant \theta_L^+ \tag{8}$$

$$\dot{\theta}_L^- \leqslant \dot{\theta}_L \leqslant \dot{\theta}_L^+ \tag{9}$$

$$\text{with } \ c_L = \lambda_L(\theta_L - \theta_L(0)) \tag{10}$$

where λ_L, α_L, β_L, and J_L are defined the same as before. Equation (6) uses the tricriteria (5), and Equation (7) is the forward kinematics equation of the left robot manipulator of dual arms. In (8), θ_L^- and θ_L^+ denote the upper and lower limits of the joint-angle vector. In (9), $\dot{\theta}_L^-$ and $\dot{\theta}_L^+$ denote the upper and lower limits of the joint velocity vector.

Similarly, the MVN-RMP-INVM-TCOCM scheme of the right robot manipulator is formulated as

$$\min\ \alpha_R\|\dot{\theta}_R\|_2^2/2+\beta_R\|\dot{\theta}_R+c_R\|_2^2/2 +(1-\alpha_R-\beta_R)\|\dot{\theta}_R\|_\infty^2/2 \tag{11}$$

$$\text{s.t. } J_R\dot{\theta}_R=\dot{r}_R \tag{12}$$

$$\theta_R^-\leqslant\theta_R\leqslant\theta_R^+ \tag{13}$$

$$\dot{\theta}_R^-\leqslant\dot{\theta}_R\leqslant\dot{\theta}_R^+ \tag{14}$$

$$\text{with } c_R=\lambda_R(\theta_R-\theta_R(0)) \tag{15}$$

where the definitions of λ_R, α_R, β_R, J_R, $\dot{r}_R$, $\theta_R^{\pm}$, and $\dot{\theta}_R^{\pm}$ are similar to those in scheme (6)–(10).

IV. QP Reformulation, Unification, and Solution

In this section, (6)–(15) are further reformulated as two QP problems, and then, the two QP problems are further unified into one standard QP formulation. Finally, the unified QP problem is solved by a simplified linear-variational-inequalities-based primal–dual neural network (S-LVI-PDNN).

A. QP Reformulation

The conversion of (6)–(9) to a standard QP is as follows.

1) Conversion of MVN Criterion and RMP Criterion: Through mathematical manipulation, we easily know that the first and second terms of (6) can be rewritten equivalently as

$$\alpha_L\dot{\theta}_L^T I\dot{\theta}_L/2\beta_L\big(\dot{\theta}_L^T I\dot{\theta}_L+\Lambda_L\dot{\theta}_L\big)/2 \tag{16}$$

where the superscript ^{T} denotes the transpose of a matrix (or a vector), $I\in R^{n\times n}$ is an identity matrix, and Λ_L=$2\lambda_L$.

2) Conversion of INVM Criterion: By defining $p_L=\|\dot{\theta}_L\|_\infty$, the INVM criterion $(1-\alpha_L-\beta_L)\|\dot{\theta}_L\|_\infty^2/2$ in the third term of (6) of the left robot manipulator is rewritten equivalently as

$$\min\ (1-\alpha_L-\beta_L)p_L^2/2 \tag{17}$$

$$\text{s.t. } \begin{bmatrix} I & -1_v\\ -I & -1_v\end{bmatrix}\begin{bmatrix}\dot{\theta}_L\\ p_L\end{bmatrix}\leqslant\begin{bmatrix}0\\0\end{bmatrix} \tag{18}$$

where 1_v is a vector of ones (i.e., $1_v=[1,1,...1]^T$).

3) Conversion of Joint Physical Limits: Because the schemes (6)–(15) are solved at the velocity level, joint physical limit (8) has to be transformed into the formulations about $\dot{\theta}_L$. Therefore, the new formulation can be written as

$$\mu_L(\theta_L^--\theta_L)\leqslant\dot{\theta}_L\leqslant\mu_L\big(\theta_L^+-\theta_L\big) \tag{19}$$

where μ_L is a positive design parameter which adjusts the feasible regions determined by joint-velocity vector.

Combining (19) with (9), a new joint physical constraint at the velocity level is defined as

$$\xi_L^-\leqslant\dot{\theta}_L\leqslant\xi_L^+ \tag{20}$$

where the ith element of ξ_L^- and ξ_L^+ is defined, respectively, as $\xi_{Li}^-=\max\{\dot{\theta}_{Li}^-,\mu_L(\theta_{Li}^--\theta_{Li})\}$, and $\xi_{Li}^+=\min\{\dot{\theta}_{Li}^+,\mu_L(\theta_{Li}^+-\theta_{Li})\}$.

4) Scheme Reformulation: By defining decision variable $x_L=[\dot{\theta}_L^T,p_L]^T\in R^{n+1}$, the MVN-RMP-INVM-TCOCM scheme of the left robot manipulator is reformulated into a standard QP, that is

$$\min\ x_L^TQ_Lx_L/2+g_L^Tx_L \tag{21}$$

$$\text{s.t. } A_Lx_L=b_L \tag{22}$$

$$C_Lx_L\leqslant d_L \tag{23}$$

$$x_L^-\leqslant x_L\leqslant x_L^+ \tag{24}$$

where coefficient matrices (or vectors) are defined as

$$Q_L=\begin{bmatrix}(\alpha_L+\beta_L)I & 0\\ 0 & (1-\alpha_L-\beta_L)\end{bmatrix}\in R^{(n+1)\times(n+1)}$$

$$g_L=\begin{bmatrix}\beta_L\Lambda_L(\theta_L-\theta_L(0))\\ 0\end{bmatrix}\in R^{n+1}$$

$$A_L=[J_L\ \ 0]\in R^{m\times(n+1)},\quad b_L=\dot{r}_L\in R^m,$$

$$C_L=\begin{bmatrix}I & -1_v\\ -I & -1_v\end{bmatrix}\in R^{2n\times(n+1)},\quad d_L=0\in R^{2n}$$

$$x_L^-=\begin{bmatrix}\xi_L^-\\ 0\end{bmatrix}\in R^{n+1},\quad x_L^+=\begin{bmatrix}\xi_L^+\\ \varpi\end{bmatrix}$$

ϖ is a large constant, which is used to replace $+\infty$ numerically.

Similarly, the QP reformulation of the right robot manipulator is the same as that of the left robot manipulator, which can be finally presented as

$$\min\ x_R^TQ_Rx_R/2+g_R^Tx_R \tag{25}$$

$$\text{s.t. } A_Rx_R=b_R \tag{26}$$

$$C_Rx_R\leqslant d_R \tag{27}$$

$$x_R^-\leqslant x_R\leqslant x_R^+ \tag{28}$$

where the definitions of Q_R, g_R, A_R, b_R, C_R, d_R, x_R^-, and x_R^+ are similar to those of the left robot manipulator.

B. QP Unification

In order to use the QP solver later and improve the computational efficiency, the two QPs (21)–(24) and (25)–(28) are unified into a standard QP, that is

$$\min\ z^TKz/2+w^Tz \tag{29}$$

$$\text{s.t. } Gz=h \tag{30}$$

$$Dz\leqslant e \tag{31}$$

$$z^-\leqslant z\leqslant z^+ \tag{32}$$

where the decision variables and the coefficient matrices (or vectors) are defined as

$$K=\begin{bmatrix}Q_L & 0\\ 0 & Q_R\end{bmatrix}\in R^{2(n+1)\times2(n+1)}$$

$$z=\begin{bmatrix}x_L\\ x_R\end{bmatrix}\in R^{2(n+1)},\quad w=\begin{bmatrix}g_L\\ g_R\end{bmatrix}\in R^{2(n+1)}$$

$$G=\begin{bmatrix}A_L & 0\\ 0 & A_R\end{bmatrix}\in R^{2m\times2(n+1)},\quad h=\begin{bmatrix}b_L\\ b_R\end{bmatrix}\in R^{2m}$$

$$D=\begin{bmatrix}C_L & 0\\ 0 & C_R\end{bmatrix}\in R^{4n\times2(n+1)},\quad e=0\in R^{4n}$$

$$z^-=\begin{bmatrix}x_L^-\\ x_R^-\end{bmatrix}\in R^{2(n+1)},\quad z^+=\begin{bmatrix}x_L^+\\ x_R^+\end{bmatrix}\in R^{2(n+1)}.$$

As for the unified QP (29)–(32) presented earlier, K is a positive-definite matrix, and thus, the performance index (29) is strictly convex. In view of the constraints (31) and (32), the feasible region is a closed convex set, and thus, the optimal solution to QP (29)–(32) is unique.

C. QP Solver

In this section, to solve the above unified QP problem (29)–(32) rapidly and efficiently, an S-LVI-PDNN is presented and applied.

Firstly, QP problem (29)–(32) is converted into an LVI problem (33). That is to seek a primal–dual equilibrium vector $y^* \in \Omega$, such that

$$(y - y^*)^T (My^* + q) \geqslant 0 \quad \forall y \in \Omega \tag{33}$$

where the convex set Ω is defined as $\Omega = \{y | y^- \leqslant y \leqslant y^+\} \subset R^{2(3n+m+1)}$. Besides, the primal–dual decision variable vector $y \in R^{2(3n+m+1)}$ and its bounds are defined as follows, that is:

$$y = \begin{bmatrix} z \\ u \\ \upsilon \end{bmatrix}, \quad y^- = \begin{bmatrix} z^- \\ -1_\upsilon \varpi \\ 0 \end{bmatrix}, \quad y^+ = \begin{bmatrix} z^+ \\ 1_\upsilon \varpi \\ 1_\upsilon \varpi \end{bmatrix}$$

where dual decision vector $u \in R^{2m}$ is defined for equality constraint (30) and $\upsilon \in R^{4n}$ is defined for inequality constraint (31). In addition, the augmented matrix $M \in R^{2(3n+m+1)\times 2(3n+m+1)}$ and vector $q \in R^{2(3n+m+1)}$ are defined as

$$M = \begin{bmatrix} K & -G^T & D^T \\ G & 0 & 0 \\ -D & 0 & 0 \end{bmatrix}, \quad q = \begin{bmatrix} w \\ -h \\ e \end{bmatrix}.$$

Secondly, LVI problem (33) is equivalent to system piecewise linear equations

$$P_\Omega(y - (My + q)) - y = 0 \tag{34}$$

where $P_\Omega(\cdot)$ is a projection operator $R^{2(3n+m+1)} \to \Omega$ and the ith element of $P_\Omega(y)$ is defined as

$$\begin{cases} y_i^-, & \text{if } y_i < y_i^- \\ y_i, & \text{if } y_i^- \leqslant y_i \leqslant y_i^+ \quad \forall i \in \{1, 2, \ldots, 2(3n + m + 1)\} \\ y_i^+, & \text{if } y_i > y_i^+. \end{cases}$$

Third, according to our experience [5], [31], [44], (34) is further computed by the following LVI-PDNN:

$$\dot{y} = \gamma (I + M^T)\{P_\Omega(y - (My + q)) - y\}. \tag{35}$$

Fourth, to further simplify the calculation and realization of real-time solver (35), the adjusting term $(I+M^T)$ is eliminated from (35), and then, we can obtain an S-LVI-PDNN as

$$\dot{y} = \gamma \{P_\Omega(y - (My + q)) - y\} \tag{36}$$

where positive parameter γ is designed to adjust the convergence rate of neural network (36).

Furthermore, the S-LVI-PDNN convergence lemma proposed in [31] guarantees that the output of the presented S-LVI-PDNN solver is globally exponentially convergent to optimal solution to QP (29)–(32) in real time.

TABLE I
JOINT PHYSICAL LIMITS OF DUAL PUMA560 ROBOT MANIPULATORS

Joint	$\theta^+_{\mathrm{L/R}}(rad)$	$\theta^-_{\mathrm{L/R}}(rad)$	$\dot{\theta}^+_{\mathrm{L/R}}(rad/s)$	$\dot{\theta}^-_{\mathrm{L/R}}(rad/s)$
1	2.7751	−2.7751	1.5000	−1.5000
2	0.7505	−3.1416	1.5000	−1.5000
3	3.1416	−0.9058	1.5000	−1.5000
4	2.9671	−1.9199	1.5000	−1.5000
5	0.0349	−1.7453	1.5000	−1.5000
6	3.1416	−3.1416	1.5000	−1.5000

V. COMPUTER SIMULATIONS AND COMPARISONS

In this section, two PUMA560 robot manipulators are set up and applied to conduct the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32). The distance between the dual-redundant-robot manipulators is 1.5 m.

In order to illustrate the advantages of the proposed scheme for tracking end-effector tasks, the simulation results of two single-criterion motion schemes (i.e., the MVN scheme and INVM scheme) and a BCOCM scheme (i.e., the MVN-INVM-BCOCM scheme) are also conducted and discussed. Specifically, in the simulations, two complex and decoupled path-tracking examples, i.e., an English word "Robot" and a Chinese character named double happiness, are shown in detail. Besides, another coupled path-tracking example, i.e., a four-pointed star pattern, is also illustrated. Note that, for the dual-robot manipulators, the joint physical limits of PUMA560 robot manipulator set in simulations are shown in Table I. In addition, the designed parameters $\alpha = \alpha_L = \alpha_R$, $\beta = \beta_L = \beta_R$, $\Lambda_L = \Lambda_R = 20$, $\mu_L = \mu_R = 20$, and $\gamma = 10^5$. The end-effector path functions (i.e., an English word "Robot") in the X-, Y-, and Z-axes are indicated in the Appendix.

A. English-Word Path-Tracking Example

In this section, end-effectors of dual PUMA560 robot manipulators are expected to simultaneously and collaboratively track an English word "Robot" path. The motion-task duration $T_{\mathrm{all}} = 17 \times T = 17s$ and the initial state $\theta_L(0) = [\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad and $\theta_R(0) = [-\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad. The corresponding simulation results are shown in Figs. 2–5 and Tables II–VII.

First of all, Figs. 2–5 show the simulation results synthesized by the MVN scheme [i.e., (29)–(32) with $\alpha = 1$ and $\beta = 0$], INVM scheme [i.e., (29)–(32) with $\alpha = 0$ and $\beta = 0$], MVN-INVM-BCOCM scheme [i.e., (29)–(32) with $\alpha = 0.5$ and $\beta = 0$], and MVN-RMP-INVM-TCOCM scheme (29)–(32) with $\alpha = 0.1$ and $\beta = 0.5$, respectively. Besides, Tables II–VII show the detailed joint-angle drift of MVN scheme, INVM scheme, MVN-INVM-BCOCM scheme, and MVN-RMP-INVM-TCOCM scheme.

Second, Fig. 2(a) shows the motion trajectories of dual PUMA560 robot manipulators synthesized by the MVN scheme [i.e., (29)–(32) with $\alpha = 1$ and $\beta = 0$]. As seen from Fig. 2(a), the path-tracking task is performed well, which is verified by the small end-effector positioning errors shown in Fig. 2(d) and (e). Fig. 2(b) and (c) shows that all the joint velocities are within their joint limits compared with Table I. It seems that the MVN scheme works in the path-tracking tasks. However, from Fig. 2(a), we see that the final joint state is far

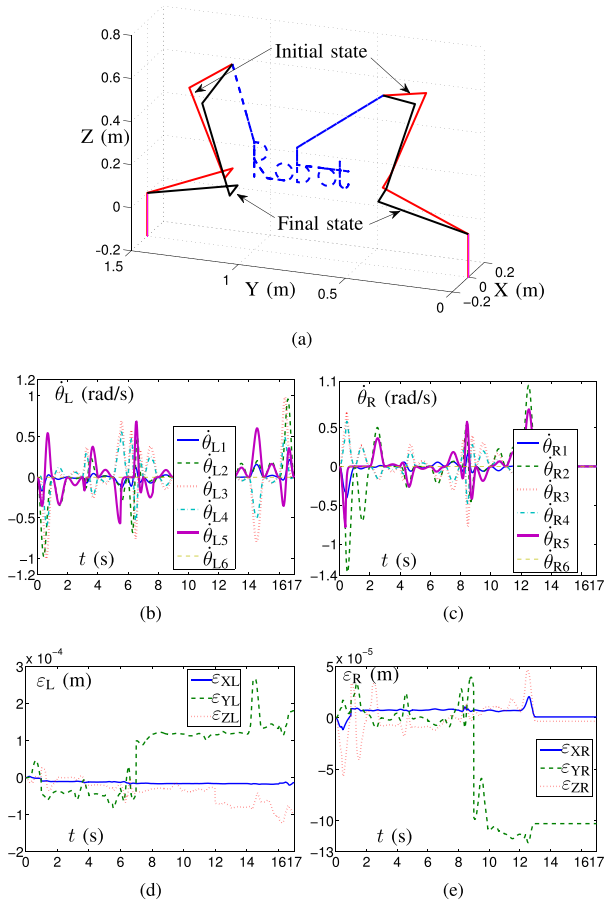


Fig. 2. Dual PUMA560 end-effectors tracking an English word "Robot" path synthesized by the MVN scheme [i.e., (29)–(32] with $\alpha = 1$ and $\beta = 0$). (a) Final states of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning errors of the left and right robot manipulators.

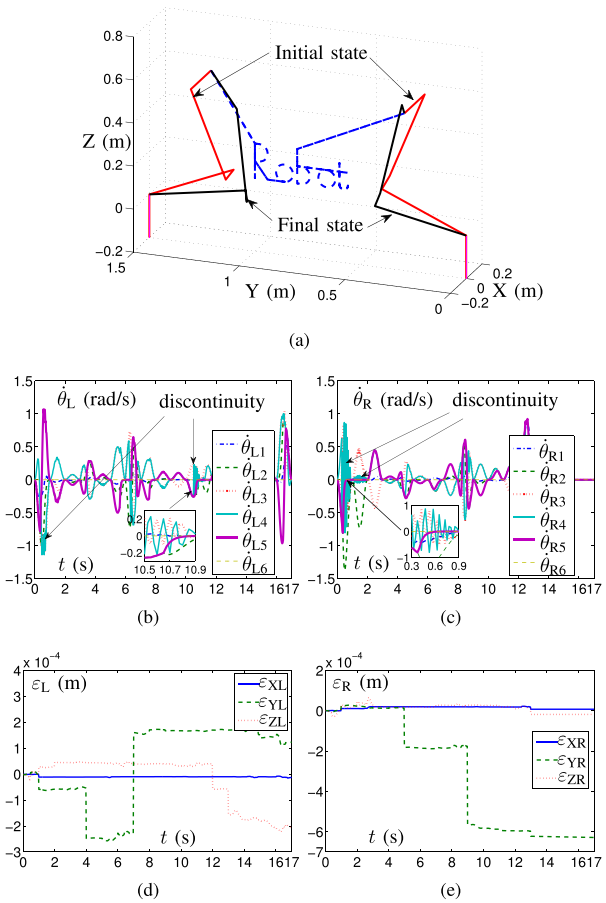


Fig. 3. Dual PUMA560 end-effectors tracking an English word "Robot" path synthesized by the INVM scheme [i.e., (29)–(32) with $\alpha = 0$ and $\beta = 0$]. (a) Final states of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning errors of the left and right robot manipulators.

TABLE II

JOINT-ANGLE DRIFTS (rad) SYNTHESIZED BY THE MVN SCHEME [i.e., (29)–(32) WITH $\alpha = 1$ AND $\beta = 0$) WHEN THE DUAL-ROBOT MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

Joint	$\theta_L(17) - \theta_L(0)$(rad)	Joint	$\theta_R(17) - \theta_R(0)$(rad)
θ_{L1}	-0.0043	θ_{R1}	-0.0015
θ_{L2}	-0.1665	θ_{R2}	-0.1844
θ_{L3}	$+0.0896$	θ_{R3}	$+0.0940$
θ_{L4}	$+0.1232$	θ_{R4}	$+0.1445$
θ_{L5}	$+0.3264$	θ_{R5}	$+0.3638$
θ_{L6}	0	θ_{R6}	0

TABLE III

JOINT-ANGLE DRIFTS (rad) SYNTHESIZED BY THE INVM SCHEME [i.e., (29)–(32) WITH $\alpha = 0$ AND $\beta = 0$) WHEN THE DUAL-ROBOT MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

Joint	$\theta_L(17) - \theta_L(0)$(rad)	Joint	$\theta_R(17) - \theta_R(0)$(rad)
θ_{L1}	$+0.0742$	θ_{R1}	$+0.2513$
θ_{L2}	-0.1726	θ_{R2}	-0.3123
θ_{L3}	$+0.0151$	θ_{R3}	$+0.0222$
θ_{L4}	$+0.2728$	θ_{R4}	$+0.5777$
θ_{L5}	$+0.2788$	θ_{R5}	$+0.4828$
θ_{L6}	-2.4931×10^{-14}	θ_{R6}	-2.6356×10^{-8}

TABLE IV

JOINT-ANGLE DRIFTS (rad) SYNTHESIZED BY THE MVN-INVM-BCOCM SCHEME [i.e., (29)–(32) WITH $\alpha = 0.5$ AND $\beta = 0$] WHEN THE DUAL-ROBOT MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

Joint	$\theta_L(17) - \theta_L(0)$(rad)	Joint	$\theta_R(17) - \theta_R(0)$(rad)
θ_{L1}	$+0.0088$	θ_{R1}	$+0.0263$
θ_{L2}	-0.2173	θ_{R2}	-0.2619
θ_{L3}	$+0.1061$	θ_{R3}	$+0.1135$
θ_{L4}	$+0.1735$	θ_{R4}	$+0.2374$
θ_{L5}	$+0.4316$	θ_{R5}	$+0.5296$
θ_{L6}	-2.1415×10^{-24}	θ_{R6}	-1.0698×10^{-14}

TABLE V

JOINT-ANGLE DRIFTS (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME [i.e., (29)–(32) WITH $\alpha = 0.1$ AND $\beta = 0.5$] WHEN THE DUAL-ROBOT MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

Joint	$\theta_L(17) - \theta_L(0)$(rad)	Joint	$\theta_R(17) - \theta_R(0)$(rad)
θ_{L1}	$+3.2551 \times 10^{-5}$	θ_{R1}	$+1.0719 \times 10^{-4}$
θ_{L2}	-3.6332×10^{-4}	θ_{R2}	$+1.5810 \times 10^{-4}$
θ_{L3}	$+8.6972 \times 10^{-4}$	θ_{R3}	-4.8684×10^{-4}
θ_{L4}	$+3.1309 \times 10^{-5}$	θ_{R4}	-3.3061×10^{-4}
θ_{L5}	$+1.5305 \times 10^{-3}$	θ_{R5}	$+3.8282 \times 10^{-4}$
θ_{L6}	$+4.5867 \times 10^{-150}$	θ_{R6}	-6.7211×10^{-22}

away from the initial joint state after finishing tasks, which means that joint-angle drift phenomenon happens. The detailed drift amplitudes are shown in Table II. In cyclic-motion tasks, the joint-angle drift leads to accumulative errors, even damage of the robot. To avoid the phenomenon happening, a state-shift scheme is required. Evidently, the MVN scheme is inefficient.

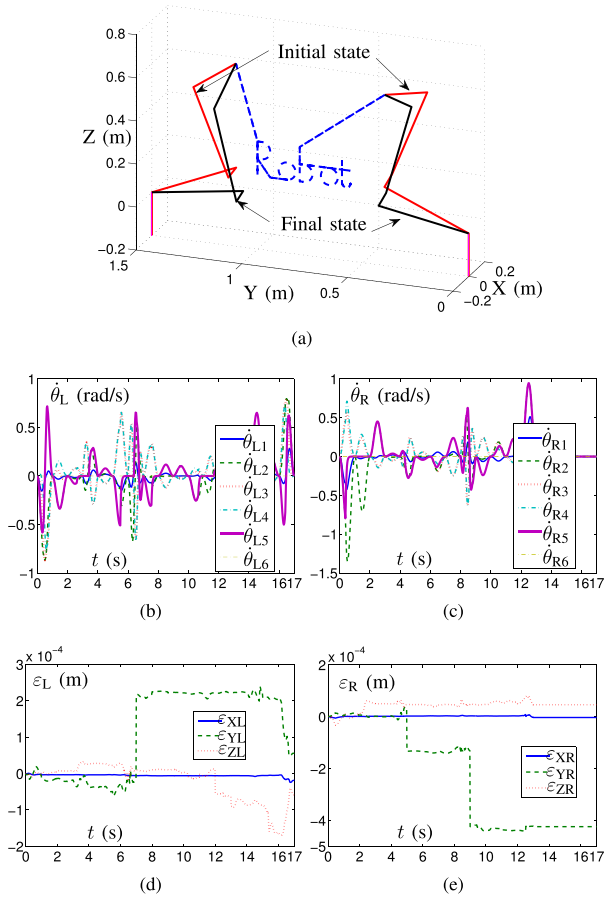


Fig. 4. Dual PUMA560 end-effectors tracking an English word "Robot" path synthesized by the MVN-INVM-BCOCM scheme [i.e., (29)–(32) with $\alpha = 0.5$ and $\beta = 0$). (a) Final states of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning errors of the left and right robot manipulators.

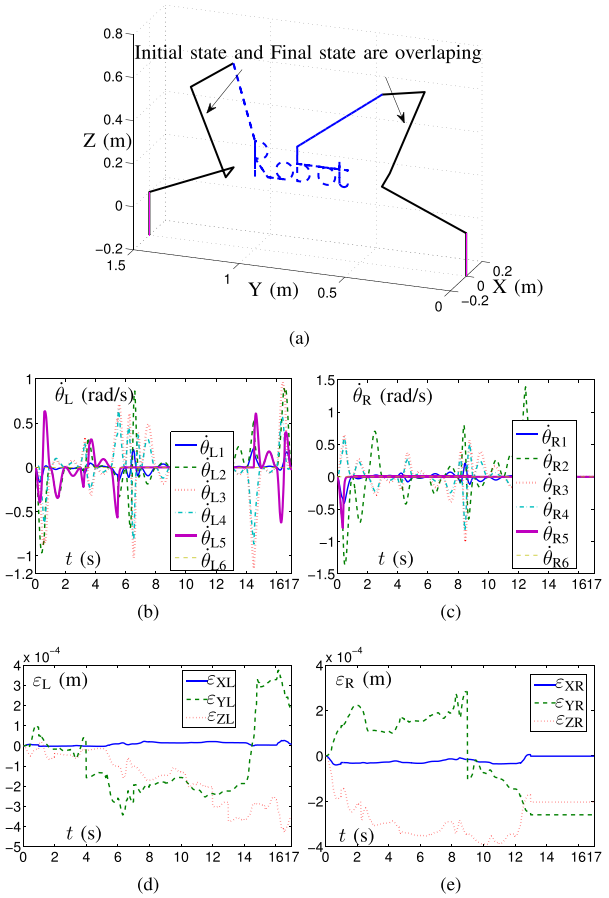


Fig. 5. Dual PUMA560 end-effectors tracking an English word "Robot" path synthesized by the MVN-RMP-INVM-TCOCM scheme (29)–(32) with $\alpha = 0.1$ and $\beta = 0.5$. (a) Final states of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning errors of the left and right robot manipulators.

TABLE VI

JOINT-ANGLE DRIFT-MEAN (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME [i.e., (29)–(32) WITH $\alpha = 0.1$ AND DIFFERENT VALUES OF β] WHEN THE DUAL-ROBOT MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

#	#	mean
$\beta = 0.3$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	8.877
$\beta = 0.5$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	3.554
$\beta = 0.7$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.342
$\beta = 0.9$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.761

TABLE VII

JOINT-ANGLE DRIFT-MEAN (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME [i.e., (29)–(32) WITH $\alpha = 0.3$ AND DIFFERENT VALUES OF β] WHEN THE DUAL-ROBOT-MANIPULATORS TRACKING AN ENGLISH WORD "ROBOT" PATH

#	#	mean
$\beta = 0.4$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	4.522
$\beta = 0.5$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.967
$\beta = 0.6$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	0.739
$\beta = 0.7$	$\lvert\theta(17) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	0.983

Third, Fig. 3(a) shows the motion trajectories of dual PUMA560 robot manipulators synthesized by the INVM scheme [i.e., (29)–(32) with $\alpha = 0$ and $\beta = 0$]. As shown from Fig. 3(a), the end-effector task is finished by the INVM scheme, and the corresponding joint-velocity profiles and end-effector positioning-errors are shown in Fig. 3(b)–(e). However, it can be seen from Fig. 3(a) that the joint-angle drift phenomenon occurs when path tracking. The detailed drift amplitudes are shown in Table III, and this result is unexpected. In addition, Fig. 3(b) and (c) shows that there are instable and discontinuous points in the joint-velocity profiles of dual-robot manipulators. This is because the INVM scheme, which minimizes the largest element of the joint-velocity vector in magnitude. It means that the velocities of joints instantaneously change during some time intervals. This is against actual requirements and damages the robot. Evidently, the INVM scheme is unapplicable in practical applications.

Fourth, Fig. 4(a) shows the motion trajectories of dual PUMA560 robot manipulators synthesized by the MVN-INVM-BCOCM scheme [i.e., (29)–(32) with $\alpha = 0.5$ and $\beta = 0$]. This scheme combines the MVN and INVM, which

can remedy the discontinuous problem occurred in the INVM scheme. In addition, the end-effector path-tracking task is finished well [see Fig. 4(a), (d), and (e)]. However, joint-drift problem still exists, which can be seen from Table IV. This is unapplicable in practical applications.

Last but not least, Fig. 5(a) shows the motion trajectories synthesized by the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32) with $\alpha = 0.1$ and $\beta = 0.5$. From Fig. 5(a), we see that the word "Robot" is tracked very well. In addition, the final joint state totally coincides with the initial one. The detailed joint-angle drifts can be seen from Table V, which are almost less than 9×10^{-4} rad except $\dot{\theta}_{L5}$. This means that the joint-angle-drift problem is remedied well by the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32).

Comparing Fig. 5(a) with Figs. 2(a), 3(a), and 4(a), we find that the joint-angle drifts of the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32) are negligible and the others are obvious, which can be further validated by comparing Table V with Tables II–IV. This is because the RMP criterion is considered in the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32). In addition, the remedy effect relies on the weighting factor β of RMP criterion. Usually, the closer the factor β to 1, the better the effect of remedying joint-angle drifts. The detailed joint-angle drifts when $\alpha = 0.1$ or $\alpha = 0.3$ with different values of β are shown in Tables VI and VII. With these comparisons, we draw a conclusion that the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32) have efficacy and accuracy in end-effector path-tracking tasks.

In summary, the comparison results verify the effectiveness and accuracy of the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32) used for dual-robot manipulators simultaneous and cooperatively tracking complex end-effector tasks.

B. Chinese Character (Named Double-Happiness) Path-Tracking Example

In order to further illustrate the effectiveness of the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32), end-effectors of dual PUMA560 manipulators are expected to collaboratively track a Chinese character, named double-happiness. The motion-task duration $T_{\text{all}} = 30 \times T = 30s$, the initial state $\theta_L(0) = [\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad, and $\theta_R(0) = [-\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad. The corresponding simulation results are shown in Fig. 6 and Tables VIII–X.

The path-tracking result, the joint velocities, and the position errors synthesized by the MVN-RMP-INVM-TCOCM scheme with $\alpha = 0.1$ and $\beta = 0.5$ are shown in Fig. 6. The detailed joint-angle drifts are listed in Table VIII. From Fig. 6(a)–(c), we see that the Chinese character is tracked well, and the final state coincides with the initial state, which means that the end-effector is completed very well and the repetitive motions is achieved. Fig. 6(d) and (e) shows the tiny error during the task execution. In addition, the detailed joint-angle drift means when $\alpha = 0.1$ or $\alpha = 0.3$ with different values of β are shown in Tables IX and X. This example further verifies the effectiveness and accuracy of the proposed MVN-RMP-INVM-TCOCM scheme for coordination control and complex path planning of dual-redundant-robot manipulators.

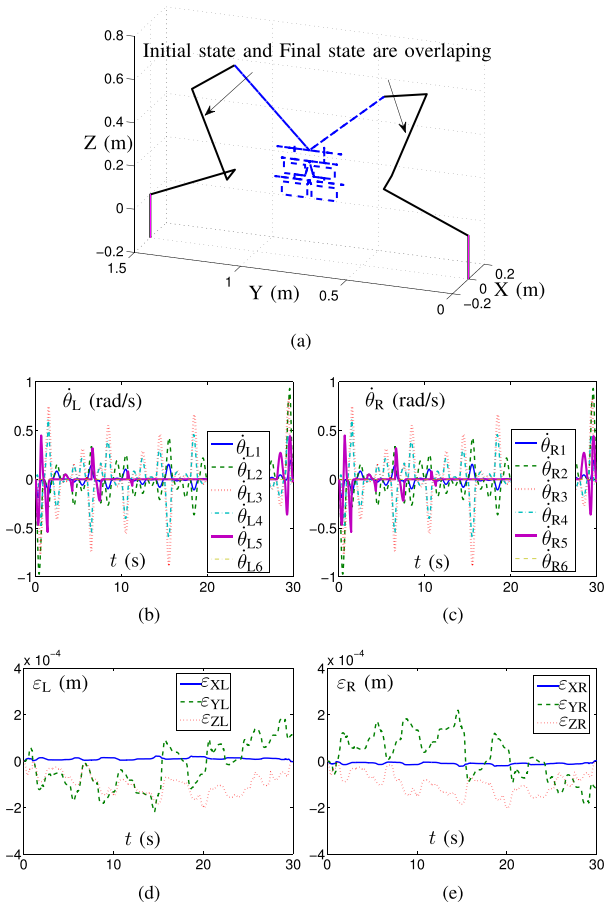


Fig. 6. Dual PUMA560 end-effectors tracking a Chinese-character (named double-happiness) path synthesized by the MVN-RMP-INVM-TCOCM scheme (with $\alpha = 0.1$ and $\beta = 0.5$). (a) Final states of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning-errors of the left and right robot manipulators.

TABLE VIII
JOINT-ANGLE DRIFTS (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME (29)–(32) ($\alpha = 0.1$ AND $\beta = 0.5$) WHEN THE DUAL-ROBOT MANIPULATORS TRACKING THE CHINESE-CHARACTER (NAMED DOUBLE-HAPPINESS) PATH

Joint	$\theta_L(30) - \theta_L(0)$(rad)	Joint	$\theta_R(30) - \theta_R(0)$(rad)
θ_{L1}	-9.6576×10^{-6}	θ_{R1}	-8.8593×10^{-6}
θ_{L2}	-6.6429×10^{-4}	θ_{R2}	-6.6032×10^{-4}
θ_{L3}	$+8.4884 \times 10^{-4}$	θ_{R3}	$+8.4463 \times 10^{-4}$
θ_{L4}	$+6.9120 \times 10^{-5}$	θ_{R4}	$+6.5982 \times 10^{-5}$
θ_{L5}	$+1.3051 \times 10^{-3}$	θ_{R5}	$+1.3099 \times 10^{-3}$
θ_{L6}	$+5.9784 \times 10^{-248}$	θ_{R6}	$+2.2994 \times 10^{-247}$

C. Coupled Task Tacking Example

In order to illustrate the effectiveness of the proposed MVN-RMP-INVM-TCOCM scheme (29)–(32) for solving a coupled task problem, end-effectors of dual PUMA560 manipulators are expected to collaboratively track a four-pointed star path. The relation of the left and right end-effector tasks can be formulated as

$$\begin{cases} \dot{r}_{\text{LX}}(t) = \dot{r}_{\text{RX}}(t) \\ \dot{r}_{\text{LY}}(t) = -0.8 \times \dot{r}_{\text{RY}}(t) \\ \dot{r}_{\text{LZ}}(t) = -0.8 \times \dot{r}_{\text{RZ}}(t) \end{cases} \quad \forall\, t \in [2T, 9T].$$

TABLE IX

JOINT-ANGLE DRIFT-MEAN (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME (29)–(32) ($\alpha = 0.1$ AND DIFFERENT VALUES OF β) WHEN THE DUAL-ROBOT-MANIPULATORS TRACKING THE CHINESE-CHARACTER (NAMED DOUBLE-HAPPINESS) PATH

#	#	mean
$\beta = 0.3$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	17.548
$\beta = 0.5$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	4.821
$\beta = 0.7$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.599
$\beta = 0.9$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.079

TABLE X

JOINT-ANGLE DRIFT-MEAN (rad) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME (29)–(32) ($\alpha = 0.3$ AND DIFFERENT VALUES OF β) WHEN THE DUAL-ROBOT-MANIPULATORS TRACKING THE CHINESE-CHARACTER (NAMED DOUBLE-HAPPINESS) PATH

#	#	mean
$\beta = 0.4$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	7.008
$\beta = 0.5$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	3.406
$\beta = 0.6$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.725
$\beta = 0.7$	$\lvert\theta(30) - \theta(0)\rvert(\times 10^{-4}\text{rad})$	1.448

TABLE XI

JOINT-ANGLE DRIFTS (RAD) SYNTHESIZED BY THE MVN-RMP-INVM-TCOCM SCHEME (29)–(32) ($\alpha = 0.1$ AND $\beta = 0.5$) WHEN THE DUAL-ROBOT-MANIPULATORS TRACKING A FOUR-POINTED STAR PATH

Joint	$\theta_{\text{L}}(10) - \theta_{\text{L}}(0)$(rad)	Joint	$\theta_{\text{R}}(10) - \theta_{\text{R}}(0)$(rad)
θ_{L1}	-4.2786×10^{-5}	θ_{R1}	-3.5255×10^{-4}
θ_{L2}	-1.1614×10^{-3}	θ_{R2}	$+7.6469 \times 10^{-4}$
θ_{L3}	$+1.1165 \times 10^{-3}$	θ_{R3}	-1.2577×10^{-3}
θ_{L4}	$+1.1410 \times 10^{-4}$	θ_{R4}	$+9.8897 \times 10^{-4}$
θ_{L5}	$+1.6659 \times 10^{-3}$	θ_{R5}	-6.6781×10^{-4}
θ_{L6}	$+1.1682 \times 10^{-103}$	θ_{R6}	$+5.3336 \times 10^{-104}$

The task duration $T_{\text{all}} = 10 \times T = 10s$ and the initial state $\theta_L(0) = [\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad and $\theta_R(0) = [-\pi/2, \pi/8, 0, \pi/3, -\pi/2, 0]^T$ rad. Generated by the MVN-RMP-INVM-TCOCM scheme with $\alpha = 0.1$ and $\beta = 0.5$, the simulation results are shown in Fig. 7 and Table XI. From Fig. 7, we can see that the coupled task is completed very well, the final state coincides well with the initial state, and the position errors are small (i.e., $< 10^{-3}$ m). In addition, the joint drifts are small ($< 10^{-3}$ rad shown in Table XI), which means that the scheme can remedy the nonrepetitive problem during the coupled task tracking duration. This example illustrates that a coupled task can be performed by dual-robot manipulators synthesized by the MVN-RMP-INVM-TCOCM scheme.

In summary, the above three simulations verify the effectiveness, accuracy, safety, and applicability of the proposed MVN-RMP-INVM-TCOCM scheme for coordination control and coupled path planning for dual-redundant-robot manipulators.

D. Compared With Hierarchical Quadratic Programming

For further testing the performance of the proposed MVN-RMP-INVM-TCOCM scheme, the presented method is conducted to compare with the hierarchical quadratic programming (HQP) method for solving the same problem. The

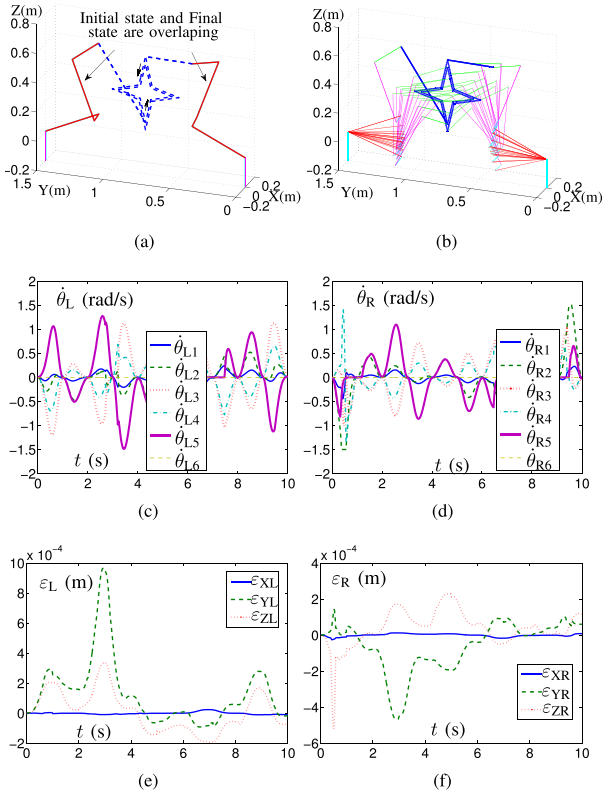


Fig. 7. Dual PUMA560 end-effectors tracking a four-pointed star path synthesized by the MVN-RMP-INVM-TCOCM scheme (with $\alpha = 0.1$ and $\beta = 0.5$). (a) Final states of dual PUMA560 robot manipulators. (b) Motion trajectory of dual PUMA560 robot manipulators. (b) and (c) Joint-velocity profiles of the left and right robot manipulators. (d) and (e) End-effector positioning errors of the left and right robot manipulators.

stratified system is the unified QP formulation (29)–(32). According to the HQP method [45]–[48], the first level is to consider a system with equality and bound constraints, that is

$$\begin{aligned} &\min\ z^T K z/2 + w^T z \\ &\text{s.t. } Gz = h \\ &\qquad z^- \leqslant z \leqslant z^+. \end{aligned}$$

After the solution z^* to the first system is obtained, the second system is to consider a system with an inequality constraint, that is

$$\begin{aligned} &\min\ z^T K z/2 + w^T z \\ &\text{s.t. } Gz = Gz^* \\ &\qquad \{d^j z \leqslant e^j\}_{j\in L} \\ &\qquad \{d^j z = d^j z^*\}_{j\in P} \end{aligned}$$

where the subset L is the rows satisfying $d^j z \leqslant e^j$ in inequality $Dz \leqslant e$ to enforce solution at point z^* and the complement subset P of inequality that were not satisfied.

Taking the Chinese character path tracking (when $\alpha = 0.3$ and $\beta = 0.6$) for example, the joint velocities generated by the MVN-RMP-INVM-TCOCM and HQP are shown in Fig. 8. From the comparison results, we see that both of the two methods can solve such a task-tracking problem, and the corresponding computation time with different parameters is listed in Table XII. Note that when joint velocities are generated by HQP, the first level is solved by using MATLAB

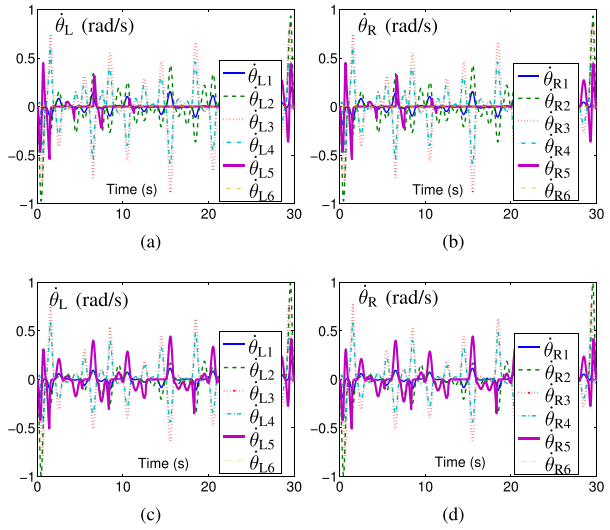


Fig. 8. Dual PUMA560 end-effectors tracking a Chinese-character (named double-happiness) path synthesized by the MVN-RMP-INVM-TCOCM scheme and HQP method (with $\alpha = 0.3$ and $\beta = 0.6$). (a) and (b) Joint-velocity profiles generated by MVN-RMP-INVM-TCOCM scheme. (c) and (d) Joint-velocity profiles generated by HQP method.

quadprog function. From Table XII, we can see that the proposed method is faster than that of the HQP method. This further illustrates the effectiveness of the proposed MVN-RMP-INVM-TCOCM method.

VI. Conclusion

In this paper, a novel TCOCM scheme, i.e., integrating MVN, repetitive motion, INVM (MVN-RMP-INVM-TCOCM scheme), has been proposed and investigated for complex motion planning and coordination control of dual-redundant-robot manipulators. Such an MVN-RMP-INVM-TCOCM scheme cannot only remove the discontinuity phenomenon brought from the INVM criterion but also prevent the occurrence of high joint-velocity. Besides, the proposed scheme can avoid the joint-physical limits and is able to remedy the joint-angle drifts of dual-redundant-robot manipulators well. In addition, the proposed scheme guarantees the joint-velocity close to zero at the end of motion. To do so, two subschemes (6)–(10) and (11)–(15) have been set up for the dual-redundant-robot manipulators. Two subschemes are then reformulated as two general quadratic programming (QP) problems (21)–(24) and (25)–(28). Then, the two general QP problems have been further unified into one standard QP formulation (29)–(32). Finally, the unified QP problem is solved by an S-LVI-PDNN solver. Comparison results based on dual PUMA560 robot manipulators have substantiated the efficacy, advantage, and applicability of the proposed MVN-RMP-INVM-TCOCM scheme on resolving the redundancy of dual-robot manipulators. Future work will consider the self-collision avoidance and multi-index optimization at different levels.

TABLE XII

Computation Time Comparisons Between the MVN-RMP-INVM-TCOCM and the HQP When Solving the Double-Happiness Path

#	t_{QP} (min)	t_{HQP} (min)
$\alpha = 0.3, \beta = 0.6$	0.45	602.3
$\alpha = 0.3, \beta = 0.7$	3.38	8.73
$\alpha = 0.2, \beta = 0.8$	4.03	5.99

Appendix

The end-effector path functions in the X-, Y-, and Z-axes are shown below.

$$\dot{r}_{LX}(t) = 0 \quad \forall\, t \in [0, 17T]$$

$$\dot{r}_{LY}(t) = \begin{cases} (\mathrm{Ini}_Y - \mathrm{Set}_Y)(1-\cos(2\pi t/T))/T, & \forall\, t \in [0, T] \\ 0, & \forall\, t \in [T, 2T] \\ -0.02(1-\cos(2\pi(t-2T)/T))/T, & \forall\, t \in [2T, 3T] \\ -0.02\pi^2 \sin\left(\frac{\pi}{2}\cos(\pi(t-3T)/T)\right)\sin(\pi(t-3T)/T)/T, & \forall\, t \in [3T, 4T] \\ 0.02(1-\cos(2\pi(t-4T)/T))/T, & \forall\, t \in [4T, 5T] \\ -0.14(1-\cos(2\pi(t-5T)/T))/T, & \forall\, t \in [5T, 6T] \\ 0.04\pi^2\cos(\pi\cos(\pi(t-6T)/T))\sin(\pi(t-6T)/T)/T, & \forall\, t \in [6T, 7T] \\ -0.1(1-\cos(2\pi(t-7T)/T))/T, & \forall\, t \in [7T, 8T] \\ 0.04(1-\cos(2\pi(t-8T)/T))/T, & \forall\, t \in [8T, 9T] \\ 0, & \forall\, t \in [9T, 10T] \\ 0, & \forall\, t \in [10T, 11T] \\ -0.01\pi^2\cos\left(\frac{\pi}{2}\cos(\pi(t-11T)/T)\right)\sin(\pi(t-11T)/T)/T, & \forall t \in [11T, 12T] \\ 0.01\pi^2\cos\left(\frac{\pi}{2}\cos(\pi(t-12T)/T)\right)\sin(\pi(t-12T)/T)/T, & \forall t \in [12T, 13T] \\ 0, & \forall\, t \in [13T, 14T] \\ 0.2(1-\cos(2\pi(t-14T)/T))/T, & \forall\, t \in [14T, 15T] \\ 0, & \forall\, t \in [15T, 16T] \\ (\mathrm{Set}_Y - \mathrm{Ini}_Y)(1-\cos(2\pi(t-16T)/T))/T, & \forall\, t \in [16T, 17T] \end{cases}$$

$$\dot{r}_{\mathrm{LZ}}(t) = \begin{cases} (\mathrm{Ini}_Z - \mathrm{Set}_Z)(1-\cos(2\pi t/T))/T, & \forall\ t \in [0, T] \\ -0.16(1-\cos(2\pi(t-T)/T))/T, & \forall\ t \in [T, 2T] \\ 0, & \forall\ t \in [2T, 3T] \\ 0.02\pi^2 \cos\left(\dfrac{\pi}{2}\cos(\pi(t-3T)/T)\right)\sin(\pi(t-3T)/T)/T, & \forall\ t \in [3T, 4T] \\ 0, & \forall\ t \in [4T, 5T] \\ 0, & \forall\ t \in [5T, 6T] \\ 0.04\pi^2 \sin(\pi\cos(\pi(t-6T)/T))\sin(\pi(t-6T)/T)/T, & \forall\ t \in [6T, 7T] \\ 0, & \forall\ t \in [7T, 8T] \\ 0, & \forall\ t \in [8T, 9T] \\ 0.04(1-\cos(2\pi(t-9T)/T))/T, & \forall\ t \in [9T, 10T] \\ -0.01(1-\cos(2\pi(t-10T)/T))/T, & \forall\ t \in [10T, 11T] \\ -0.01\pi^2 \sin\left(\dfrac{\pi}{2}\cos(\pi(t-11T)/T)\right)\sin(\pi(t-11T)/T)/T, & \forall t \in [11T, 12T] \\ 0.01\pi^2 \sin\left(\dfrac{\pi}{2}\cos(\pi(t-12T)/T)\right)\sin(\pi(t-12T)/T)/T, & \forall t \in [12T, 13T] \\ 0.06(1-\cos(2\pi(t-13T)/T))/T, & \forall\ t \in [13T, 14T] \\ 0, & \forall\ t \in [14T, 15T] \\ 0.08(1-\cos(2\pi(t-15T)/T))/T, & \forall\ t \in [15T, 16T] \\ (\mathrm{Set}_Z - \mathrm{Ini}_Z)(1-\cos(2\pi(t-16T)/T))/T, & \forall\ t \in [16T, 17T] \end{cases}$$

$$\dot{r}_{\mathrm{RX}}(t) = 0 \quad \forall\ t \in [0, 17T]$$

$$\dot{r}_{\mathrm{RY}}(t) = \begin{cases} (\mathrm{Ini}_Y - \mathrm{Set}_Y)(1-\cos(2\pi t/T))/T, & \forall\ t \in [0, T] \\ 0, & \forall\ t \in [T, 2T] \\ 0, & \forall\ t \in [2T, 3T] \\ -0.02/(1-\cos(2\pi(t-3T)/T))/T, & \forall\ t \in [3T, 4T] \\ -0.04\pi^2 \sin\left(\dfrac{\pi}{2}\cos(\pi(t-4T)/T)\right)\sin(\pi(t-4T)/T)/T, & \forall t \in [4T, 5T] \\ 0.02(1-\cos(2\pi(t-5T)/T))/T, & \forall\ t \in [5T, 6T] \\ -0.06(1-\cos(2\pi(t-6T)/T))/T, & \forall\ t \in [6T, 7T] \\ -0.08(1-\cos(2\pi(t-7T)/T))/T, & \forall\ t \in [7T, 8T] \\ -0.04\pi^2 \cos(\pi\cos(\pi(t-8T)/T))\sin(\pi(t-8T)/T)/T, & \forall t \in [8T, 9T] \\ 0.08(1-\cos(2\pi(t-9T)/T))/T, & \forall\ t \in [9T, 10T] \\ 0.06(1-\cos(2\pi(t-10T)/T))/T, & \forall\ t \in [10T, 11T] \\ 0, & \forall\ t \in [11T, 12T] \\ (\mathrm{Set}_Y - \mathrm{Ini}_Y)(1-\cos(2\pi(t-13T)/T))/T, & \forall\ t \in [12T, 13T] \\ 0, & \forall\ t \in [13T, 17T] \end{cases}$$

$$\dot{r}_{\mathrm{RZ}}(t) = \begin{cases} (\mathrm{Ini}_Z - \mathrm{Set}_Z)(1-\cos(2\pi t/T))/T, & \forall\ t \in [0, T] \\ -0.16(1-\cos(2\pi(t-T)/T))/T, & \forall\ t \in [T, 2T] \\ 0.16(1-\cos(2\pi(t-2T)/T))/T, & \forall\ t \in [2T, 3T] \\ 0, & \forall\ t \in [3T, 4T] \\ -0.04\pi^2 \cos\left(\dfrac{\pi}{2}\cos(\pi(t-4T)/T)\right)\sin(\pi(t-4T)/T)/T, & \forall t \in [4T, 5T] \\ 0, & \forall\ t \in [5T, 6T] \\ -0.08(1-\cos(2\pi(t-6T)/T))/T, & \forall\ t \in [6T, 7T] \\ 0, & \forall\ t \in [7T, 8T] \\ -0.04\pi^2 \sin(\pi\cos(\pi(t-8T)/T))\sin(\pi(t-8T)/T), & \forall t \in [8T, 9T] \\ 0, & \forall\ t \in [9T, 10T] \\ 0.08(1-\cos(2\pi(t-10T)/T))/T, & \forall\ t \in [10T, 11T] \\ 0.08(1-\cos(2\pi(t-11T)/T))/T, & \forall\ t \in [11T, 12T] \\ (\mathrm{Set}_Z - \mathrm{Ini}_Z)(1-\cos(2\pi(t-13T)/T))/T, & \forall\ t \in [12T, 13T] \\ 0, & \forall\ t \in [13T, 17T]. \end{cases}$$

REFERENCES

[1] P. Brizzi *et al.*, "Bringing the Internet of Things along the manufacturing line: A case study in controlling industrial robot and monitoring energy consumption remotely," in *Proc. IEEE 18th Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2013, pp. 1–8.

[2] K. Kaltsoukalas, S. Makris, and G. Chryssolouris, "On generating the motion of industrial robot manipulators," *Robot. Comput.-Integr. Manuf.*, vol. 32, pp. 65–71, Apr. 2015.

[3] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1261–1280, Dec. 2015.

[4] 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.

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

[6] A. Mohammadi, E. Rezapour, M. Maggiore, and K. Y. Pettersen, "Maneuvering control of planar snake robots using virtual holonomic constraints," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 884–899, May 2016.

[7] A. Atawnih, D. Papageorgiou, and Z. Doulgeri, "Kinematic control of redundant robots with guaranteed joint limit avoidance," *Robot. Auto. Syst.*, vol. 79, pp. 122–131, May 2016.

[8] 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.

[9] 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.

[10] Z. Zhu, J. Xiao, J.-Q. Li, F. Wang, and Q. Zhang, "Global path planning of wheeled robots using a multi-objective memetic algorithms," *Integr. Comput.-Aided Eng.*, vol. 22, no. 4, pp. 387–404, 2015.

[11] S. Yahya, M. Moghavvemi, and H. A. F. Mohamed, "Singularity avoidance of a six degree of freedom three dimensional redundant planar manipulator," *Comput. Math. Appl.*, vol. 64, no. 5, pp. 856–868, 2012.

[12] N. Kimura *et al.*, "Mobile dual-arm robot for automated order picking system in warehouse containing various kinds of products," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Dec. 2015, pp. 332–338.

[13] J. Zhai, W. Yan, Z. Fu, and Y. Zhao, "Kinematic analysis of a dual-arm humanoid cooking robot," in *Proc. Ind. Mechatron. Autom.*, Aug. 2015, pp. 249–254.

[14] C. A. Klein and K.-B. Kee, "The nature of drift in pseudoinverse control of kinematically redundant manipulators," *IEEE Trans. Robot. Autom.*, vol. 5, no. 2, pp. 231–234, Apr. 1989.

[15] D. Guo, K. Zhai, Z. Xiao, H. Tan, and Y. Zhang, "Acceleration-level minimum kinetic energy (MKE) scheme derived via Ma equivalence for motion planning of redundant robot manipulators," in *Proc. 7th Int. Symp. Comput. Intell. Design (ISCID)*, vol. 1. Dec. 2014, pp. 26–30.

[16] Y. Zhang and Z. Zhang, *Repetitive Motion Planning and Control of Redundant Robot Manipulators*. Berlin, Germany: Springer, 2014.

[17] Z.-J. Zhang and Y.-N. Zhang, "Equivalence of different-level schemes for repetitive motion planning of redundant robots," *Acta Autom. Sin.*, vol. 39, no. 1, pp. 88–91, Jan. 2013.

[18] Y. Zhang, "A set of nonlinear equations and inequalities arising in robotics and its online solution via a primal neural network," *Neurocomputing*, vol. 70, nos. 1–3, pp. 513–524, 2006.

[19] Y. Zhang, S. S. Ge, and T. H. Lee, "A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2126–2132, Oct. 2004.

[20] T. Baratcart, V. Salvucci, and T. Koseki, "Experimental verification of two-norm, infinity-norm continuous switching implemented in resolution of biarticular actuation redundancy," *Adv. Robot.*, vol. 29, no. 19, pp. 1243–1252, 2015.

[21] Y. Zhang, "Inverse-free computation for infinity-norm torque minimization of robot manipulators," *Mechatronics*, vol. 16, nos. 3–4, pp. 177–184, 2006.

[22] Y. Zhang, B. Cai, L. Zhang, and K. Li, "Bi-criteria velocity minimization of robot manipulators using a linear variational inequalities-based primal-dual neural network and PUMA560 example," *Adv. Robot.*, vol. 22, nos. 13–14, pp. 1479–1496, 2008.

[23] Y. Zhang and D. Guo, "Bi-criteria torque minimization of redundant robots performing different end-effector trajectories by using simplified LVI-PDNN," in *Proc. Int. Conf. Modeling, Identificat. Control (ICMIC)*, Jun. 2012, pp. 791–796.

[24] B. Cai and Y. Zhang, "Bi-criteria optimal control of redundant robot manipulators using LVI-based primal-dual neural network," *Optim. Control Appl. Methods*, vol. 31, no. 3, pp. 213–229, 2010.

[25] Z. Yunong, L. Kene, G. Dongsheng, and C. Binghuang, "Different-level simultaneous resolution of robot redundancy with end-effector path tracked and with joint velocity and acceleration both minimized," in *Proc. 31st Chin. Control Conf. (CCC)*, Jul. 2012, pp. 4856–4861.

[26] D. Guo and Y. Zhang, "Different-level two-norm and infinity-norm minimization to remedy joint-torque instability/divergence for redundant robot manipulators," *Robot. Auto. Syst.*, vol. 60, no. 6, pp. 874–888, 2012.

[27] Y. Zhang, D. Guo, and S. Ma, "Different-level simultaneous minimization of joint-velocity and joint-torque for redundant robot manipulators," *J. Intell. Robot. Syst.*, vol. 72, no. 3, pp. 301–323, 2013.

[28] P. Chen, C. Shan, J. Xiang, and W. Wei, "Moving obstacle avoidance for redundant manipulator via weighted least norm method," in *Proc. Control Decision Conf.*, May 2015, pp. 6181–6186.

[29] C. Smith *et al.*, "Dual arm manipulation—A survey," *Robot. Auto. Syst.*, vol. 60, no. 10, pp. 1340–1353, 2012.

[30] D. Nicolis, A. M. Zanchettin, and P. Rocco, "Constraint-based and sensorless force control with an application to a lightweight dual-arm robot," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 340–347, Jan. 2016.

[31] Z. Zhang, Z. Li, Y. Zhang, Y. Luo, and Y. Li, "Neural-dynamic-method-based dual-arm CMG scheme with time-varying constraints applied to humanoid robots," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3251–3262, Dec. 2015.

[32] F. Caccavale, P. Chiacchio, A. Marino, and L. Villani, "Six-DOF impedance control of dual-arm cooperative manipulators," *IEEE/ASME Trans. Mechatronics*, vol. 13, no. 5, pp. 576–586, Oct. 2008.

[33] T. Yoshikawa, "Multifingered robot hands: Control for grasping and manipulation," *Annu. Rev. Control*, vol. 34, no. 2, pp. 199–208, 2010.

[34] L. Cao, Q. Jia, G. Chen, L. Zhang, and H. Sun, "Coordinated hybrid force/position control for robot with three branches," in *Proc. IEEE 10th Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2015, pp. 939–944.

[35] M. Jiang, M.-Q. Fan, A.-M. Li, X.-W. Rong, H. Kong, and R. Song, "Coordination control of dual-arm robot based on modeled predictive control," in *Proc. IEEE Int. Conf. Real-Time Comput. Robot. (RCAR)*, Jun. 2016, pp. 495–499.

[36] Y. Ren, Y. Liu, M. Jin, and H. Liu, "Biomimetic object impedance control for dual-arm cooperative 7-DOF manipulators," *Robot. Auto. Syst.*, vol. 75, pp. 273–287, Jan. 2016.

[37] 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.

[38] 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.

[39] J. Lee, P. H. Chang, and R. S. Jamisola, "Relative impedance control for dual-arm robots performing asymmetric bimanual tasks," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3786–3796, Jul. 2014.

[40] A. Takhmar, I. G. Polushin, A. Talasaz, and R. V. Patel, "Cooperative teleoperation with projection-based force reflection for MIS," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 4, pp. 1411–1426, Jul. 2015.

[41] Y. Li, L. Chen, K. P. Tee, and Q. Li, "Reinforcement learning control for coordinated manipulation of multi-robots," *Neurocomputing*, vol. 170, pp. 168–175, Dec. 2015.

[42] E. Yin, Z. Zhou, J. Jiang, F. Chen, Y. Liu, and D. Hu, "A speedy hybrid BCI spelling approach combining P300 and SSVEP," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 2, pp. 473–483, Feb. 2014.

[43] T. Ma *et al.*, "The hybrid BCI system for movement control by combining motor imagery and moving onset visual evoked potential," *J. Neural Eng.*, vol. 14, no. 2, p. 026015, 2017.

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

[45] O. Kanoun, F. Lamiraux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Trans. Robot.*, vol. 27, no. 4, pp. 785–792, Aug. 2011.

[46] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 1006–1028, 2014.
[47] E. Lutscher and G. Cheng, "Hierarchical inequality task specification for indirect force controlled robots using quadratic programming," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2014, pp. 4722–4727.
[48] M. Johansson, "Online whole-body control using hierarchical quadratic programming: Implementation and evaluation of the HiQP control framework," M.S. thesis, Linköping Univ., Linköping, Sweden, 2016.

Zhijun Zhang (M'12) received the Ph.D. degree in communication and information system from Sun Yat-sen University, Guangzhou, China, in 2012.

He was a Post-Doctoral Research Fellow with the Institute for Media Innovation, Nanyang Technological University, Singapore, from 2013 to 2015. Since 2015, he has been an Associate Professor with the School of Automation Science and Engineering, South China University of Technology, Guangzhou. His current research interests include robotics, neural networks, and human–robot interaction.

Yujun Lin is currently pursuing the bachelor's degree in automation from the School of Automation Science and Engineering, South China University of Technology, Guangzhou, China.

His current research interests include robotics, neural networks, and machine learning.

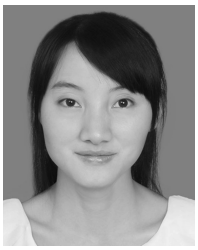
Shuai Li (M'14) received the B.E. degree in precision mechanical engineering from the Hefei University of Technology, Hefei, China, in 2005, the M.E. degree in automatic control engineering from the University of Science and Technology of China, Hefei, in 2008, and the Ph.D. degree in electrical and computer engineering from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2014.

He is currently with The Hong Kong Polytechnic University, Hong Kong. His current research interests include dynamic neural networks, wireless sensor networks, robotic networks, machine learning, and other dynamic problems defined on a graph.

Yuanqing Li (F'16) received the Ph.D. degree in control theory and applications from the South China University of Technology, Guangzhou, China, in 1997.

Since 1997, he has been with the South China University of Technology, where he became a Full Professor in 2004. From 2002 to 2004, he was with the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Saitama, Japan, as a Researcher. From 2004 to 2008, he was with the Laboratory for Neural Signal Processing, Institute for Infocomm Research, Singapore, as a Research Scientist. His current research interests include blind signal processing, machine learning, brain–computer interface, EEG, and fMRI data analysis.

Zhuliang Yu received the Ph.D. degree in electronic engineering from Nanyang Technological University, Singapore, in 2006. He joined the Center for Signal Processing, Nanyang Technological University in 2000, as a Research Engineer, and a Group Leader in 2001. In 2008, he joined the College of Automation Science and Engineering, South China University of Technology and was promoted to Full Professor in 2010. His current research interests include signal processing, pattern recognition, machine learning and their applications in communications, biomedical engineering, and robotics.



Yamei Luo received the M.S. degree in electronics and communication engineering from the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China, in 2015.

From 2009 to 2012, she was with the Guangzhou Institute of Measurement and Testing Technology, Guangzhou. Since 2012, she was with the Patent Examination Cooperation Center of the Patent Office, SIPO, Guangdong. Her current research interests include robotics and intelligent information processing.