A Dual Neural Network for Bi-Criteria Kinematic Control of Redundant Manipulators

Yunong Zhang, *Student Member, IEEE*, Jun Wang, *Senior Member, IEEE*, and Yangsheng Xu, *Senior Member, IEEE*

***Abstract*—A dual neural network is presented for the bi-criteria kinematic control of redundant manipulators. To diminish the discontinuity of minimum infinity-norm solutions, the kinematic-control problem is formulated in the bi-criteria of the infinity and Euclidean norms. Physical constraints such as joint limits and joint velocity limits are also incorporated simultaneously into the proposed kinematic control scheme. The single-layer dual neural network model with a simple structure is developed for bi-criteria redundant resolution of redundant manipulators subject to robot physical constraints. The dual neural network is shown to be globally convergent to optimal solutions in the bi-criteria sense, and is demonstrated to be effective in controlling the PA10 robot manipulator.**

***Index Terms*—Bi-criteria, dual neural network, joint limits, joint velocity limits, kinematically redundant manipulators.**

I. Introduction

KINEMATICALLY redundant manipulators are those having more degrees of freedom (DOF) than required for position and orientation [1]. The redundancy of such manipulators, including intrinsical redundancy and functional redundancy, can be utilized to avoid obstacles [2], singularities and physical limits [3], [4], and to optimize various performance criteria [5]–[7], in addition to the end-effector motion task. Redundant manipulator motion planning and control is, thus, an appealing area in robotics research. The inverse kinematics problem to find the joint motion for a given end-effector task is one of the vital and challenging issues in redundant manipulator control, because there are an infinite number of joint configurations which accomplish a specific end-effector task.

The end-effector position and orientation in Cartesian space are related to the joint space through a forward kinematic equation

$$r(t) = f(\theta(t)) \tag{1}$$

where $r(t) \in R^m$ is the vector of the end-effector position and orientation in the Cartesian space, $\theta(t) \in R^n$ is joint variable vector, and $f(\cdot)$ is a continuous nonlinear mapping function with a known structure and parameters for a given manipulator.

The inverse kinematics problem is to find the joint variable $\theta(t)$ for a given position and orientation of the end-effector $r(t)$ through the inverse mapping of (1)

$$\theta(t) = f^{-1}(r(t)). \tag{2}$$

Unfortunately, it is usually difficult, if not impossible, to find an analytic solution due to the nonlinearity and redundancy of $f(\cdot)$.

The inverse kinematics problem of manipulators is usually solved at velocity level where the end-effector velocities and joint velocities have a linear relationship. Differentiating (1) with respect to time yields the linear relation between the Cartesian velocity $\dot{r}$ and the joint velocity $\dot{\theta}$

$$J(\theta)\dot{\theta} = \dot{r} \tag{3}$$

where $J(\theta) \in R^{m\times n}$ is the Jacobian matrix defined as $J(\theta) = \partial f(\theta)/\partial\theta$. In a redundant manipulator, since $m < n$, (3) is underdetermined and, hence, admits an infinite number of solutions.

The conventional minimum two-norm solution of joint velocity vector including pseudoinverse-like solution has been widely investigated by the vast majority of researchers. Such minimization schemes minimize the sum of squared joint velocities, which does not necessarily minimize the magnitudes of individual joint velocity. It is used as the optimization criterion in many robotic applications, more because of its mathematical tractability than physical desirability [5]. The minimum infinity-norm solution of joint velocity vector, known as the minimum-effort solution or the minimum-amplitude solution, explicitly minimizes the largest component of the joint velocity vector in magnitude and is consistent with the physical limits. Moreover, the minimization of infinity-norm of joint velocity vector enables a better direct monitoring and control of the magnitude of individual joint velocities (e.g., robotic surgery) [7]–[9], [15]. It is, therefore, more desirable in the situation where low individual joint velocity is of primary concern.

The minimum infinity-norm solutions may, however, encounter a discontinuity problem. It is shown in [8] that the possibility of a discontinuity of the minimum infinity-norm solution exists purely because of the nonuniqueness possibility. In other words, if the manipulator trajectory orients the solution

Manuscript received November 13, 2001; revised May 7, 2002. This paper was recommended for publication by Associate Editor S. Chiaverini and Editor I. Walker upon evaluation of the reviewers' comments. This work was supported by the Hong Kong Research Grants Council under Grant CUHK4165/98E. This paper was presented in part at the 2002 International Joint Conference on Neural Networks, Honolulu, HI, May 2002.

The authors are with the Department of Automation and Computer-Aided Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong (e-mail: ynzhang@acae.cuhk.edu.hk; jwang@acae.cuhk.edu.hk; ysxu@acae.cuhk.edu.hk).

Digital Object Identifier 10.1109/TRA.2002.805651

space so that it is parallel to a hypercube face, the solution may jump from one edge to the other before continuing smoothly on its way. To remedy the discontinuity problem, a balancing scheme is presented in [8] that calculates the minimum infinity-norm and Euclidean-norm solutions separately, and then incorporates the two weighted solutions as the final solution. With $\dot\theta^*$, $\dot\theta^{(\infty)}$, and $\dot\theta^{(2)}$ denoting, respectively, the final solution, the minimum infinity-norm, and Euclidean-norm solutions, the balancing solution is

$$\dot\theta^* = \alpha\dot\theta^{(2)} + (1-\alpha)\dot\theta^{(\infty)}, \quad 0<\alpha<1. \tag{4}$$

The computation of the inverse kinematics solution is very time consuming, especially for high DOF robotic systems and the usual cases with many subtask criteria and/or physical constraints (e.g., joint limits and joint velocity limits). Compared to the common minimum-norm solution, the balancing scheme (4) may at least double the computational time, which is rather inefficient and may hinder online sensor-based robotic applications. Parallel computation methods, such as neural network approaches, are effective and efficient alternatives for real-time solutions to such a balanced inverse kinematics problem.

In recent years, many neural networks have been developed for robotic kinematic and dynamic control, e.g., [10]–[16]. In particular, the neural network approach that incorporates the pseudoinverse network [17] and the linear-programming neural network [18] is applied to the minimum infinity-norm kinematic control in [14]. As an improved neural network model of [14], a two-layered primal-dual neural network is then presented in [15] to online minimize the infinity-norm of joint velocity. To reduce network complexity and increase computational efficiency, a single-layered dual neural network is proposed for kinematic control of redundant manipulators by Xia and Wang [16]. In the aforementioned neural schemes, it is assumed implicitly that there exist no joint limits or joint velocity limits when solving the inverse kinematics problem. If a solution exceeds the mechanical joint limits or velocity limits and locks there, the desired motion path may fail to execute, not to mention the possibility of mechanical damage [4].

In this paper, to resolve the discontinuity deficiency of minimum infinity-norm solution, a dual neural network is developed for online kinematic control of physically constrained redundant manipulators, which is formulated as the inequality-constrained quadratic program with bi-criteria of the infinity and Euclidean norms.

The remainder of this paper is organized in five sections. Section II provides the background information and the problem formulation for the bi-criteria redundancy resolution of physically constrained manipulators. Section III presents the proposed dual neural network model for online bi-criteria kinematic control. The global convergence results are discussed in Section IV. Section V illustrates simulation results of the dual neural network and the 7-DOF PA10 manipulator to show their operating characteristics and performance. Last, Section VI concludes this paper with final remarks.

II. Problem Formulation

In this section, we consider the following bi-criteria kinematic control problem to avoid discontinuities in minimum-effort solution

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\left[\alpha\|\dot\theta\|_2^2 + (1-\alpha)\|\dot\theta\|_\infty^2\right] && (5)\\ \text{subject to} \quad & J(\theta)\dot\theta = \dot r && (6)\\ & \dot\theta^- \leqslant \dot\theta \leqslant \dot\theta^+ && (7)\\ & \theta^- \leqslant \theta \leqslant \theta^+ && (8)\end{aligned}$$

where $\|\cdot\|_\infty$ and $\|\cdot\|_2$ denote the infinity and Euclidean norms, respectively, the coefficient $\alpha\in(0,1)$. $\theta^\pm$ and $\dot\theta^\pm$ denote, respectively, upper and lower limits for joints and joint velocities.

The limited joint range (8) can be formulated in terms of $\dot\theta$ by using variable bounds

$$\beta(\theta^- - \theta) \leqslant \dot\theta \leqslant \beta(\theta^+ - \theta)$$

where the positive coefficient β is used to scale the feasible region of $\dot\theta$. Joint limits (8) and joint velocity limits (7) can, thus, be combined into the following bound constraint:

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

where the ith elements of $\dot\eta^-$ and $\dot\eta^+$ are defined, respectively, as

$$\dot\eta_i^- = \max\left\{\dot\theta_i^-, \beta\left(\theta_i^- - \theta_i\right)\right\}$$
$$\dot\eta_i^+ = \min\left\{\dot\theta_i^+, \beta\left(\theta_i^+ - \theta_i\right)\right\}.$$

Next, let us convert the minimum infinity-norm part of (5) into a quadratic program. For $\dot\theta = \left[\dot\theta_1, \dot\theta_2, \ldots, \dot\theta_n\right]^T$, with the superscript ^{T} denoting the transpose operator, its infinity-norm $\|\dot\theta\|_\infty$ is defined as

$$\|\dot\theta\|_\infty = \max\left\{\left|\dot\theta_1\right|, \left|\dot\theta_2\right|, \ldots, \left|\dot\theta_n\right|\right\} = \max_{1\leqslant j\leqslant n}\left|e_j^T\dot\theta\right|$$

where $|\cdot|$ denotes the absolute value of the component, and $e_j\in R^n$ is the jth column of the identity matrix I. By defining $s(t) = \|\dot\theta(t)\|_\infty$, the minimization of $(1-\alpha)\|\dot\theta(t)\|_\infty^2/2$ is then equivalent to

$$\begin{aligned}\text{minimize} \quad & \frac{1-\alpha}{2}s^2(t)\\ \text{subject to} \quad & \left|e_j^T\dot\theta\right| \leqslant s(t)\end{aligned}$$

which can be rewritten equivalently as

$$\begin{aligned}\text{minimize} \quad & \frac{1-\alpha}{2}s^2(t)\\ \text{subject to} \quad & \begin{bmatrix} I & -1\\ -I & -1\end{bmatrix}\begin{bmatrix}\dot\theta(t)\\ s(t)\end{bmatrix} \leqslant \begin{bmatrix}0\\0\end{bmatrix}\end{aligned}$$

where $1 := [1,1,\ldots,1]^T$ and $0 := [0,0,\ldots,0]^T$ are vectors, respectively, of ones and zeros with appropriate dimensions hereafter.

Thus, by defining the variable vector $x = \left[\dot\theta^T, s\right]^T \in R^{n+1}$, the bi-criteria kinematic control problem (5)–(8) can be expressed as the following quadratic program:

$$\begin{aligned}\text{minimize} \quad & \frac{1}{2}x^TQx && (10)\\ \text{subject to} \quad & Ax \leqslant b && (11)\\ & Cx = d && (12)\\ & x^- \leqslant x \leqslant x^+ && (13)\end{aligned}$$

where the coefficient matrices and vectors

$$Q := \begin{bmatrix} \alpha I & \\ & (1-\alpha) \end{bmatrix} \in R^{(n+1)\times(n+1)}$$
$$A := \begin{bmatrix} I & -1 \\ -I & -1 \end{bmatrix} \in R^{2n\times(n+1)}$$
$$C := [\, J(\theta) \quad 0 \,] \in R^{m\times(n+1)}$$
$$b := 0 \in R^{2n},\ d := \dot{r}(t) \in R^{m},$$
$$x^- := \begin{bmatrix} \eta^- \\ 0 \end{bmatrix} \in R^{n+1}$$
$$x^+ := \begin{bmatrix} \eta^+ \\ \max_{1\leqslant j\leqslant n} |\dot{\theta}_j^{\pm}| \end{bmatrix} \in R^{n+1}.$$

Since the objective function (10) is strictly convex (due to $0 < \alpha < 1$ and $Q > 0$) and the feasible region of linear constraints (11)–(13) is a closed convex set, if not empty, it follows from [23] that the constrained minimizer to the bi-criteria quadratic program (5)–(8) is unique and satisfies the Karush–Kuhn–Tucker optimality conditions. In light of the uniqueness property, the continuity of the bi-criteria solution is guaranteed. As α approaches its lower bound 0, the bi-criteria solution is approximate to the infinity-norm solution. As $\alpha \to 1$, the bi-criteria solution becomes nearly the standard two-norm solution, which illustrates that the proposed bi-criteria control scheme is much more flexible than a single-criterion control scheme. In this study, the parameter α is usually selected between 0.3 and 0.6 to diminish the discontinuity, while keeping small the maximal magnitude of minimum-effort solutions. A systematic method for selecting α can be found in [8], while it seems not efficient enough for online computation of higher DOF manipulators by always checking the time-varying zero subspace angle condition. The design parameter β is selected such that the feasible region of $\dot{\theta}$ made by conversion of joint limits is not less than the one made by joint velocity limits; that is, β is selected not less than $\max_{1\leqslant i\leqslant n}\left\{\left(\dot{\theta}_i^+ - \dot{\theta}_i^-\right)/\left(\theta_i^+ - \theta_i^-\right)\right\}$. Note that large values of β may cause joint deceleration quickly when the manipulator approaches its joint limits.

III. Dual Neural Network Model

In this section, we first brief the motivation and development of recurrent neural networks. The dynamical system approach, as one of the important methods for solving optimization problems, was first proposed by Pyne [19] in the late 1950s. Due to the in-depth research in neural networks, numerous dynamic solvers based on neural networks have been developed and investigated [10]–[22] in the past two decades. Specifically, Tank and Hopfield [20] proposed their working neural network implemented on analog circuits, which contained finite penalty parameters and generated approximate solutions only. When solving inequality-constrained quadratic programs, the Lagrange neural network [10] may exhibit the premature defect, and the network dimensionality is larger than that of original problems. As a very flexible tool for exactly solving constrained quadratic programs, the primal-dual neural networks [15] were developed, with the feature that they handle the primal quadratic program and its dual problem simultaneously by minimizing the duality gap with gradient method. Unfortunately, the dynamic equations of the primal-dual neural network are usually complicated and contain high-order nonlinear terms, with the network size usually larger than the dimensionality of the primal and dual problems. As a special case of the primal-dual neural network, the dual neural network [16] is proposed by using the dual decision variables only. Different from the primal-dual neural network, the dual neural network is developed directly using Karush–Kuhn–Tucker conditions and the projection operator to reduce network complexity and increase computational efficiency.

Now, following Xia and Wang's approach [16], a dual neural network is generalized for optimal bi-criteria kinematic control of redundant manipulators. Let us reformulate the constrained quadratic program into a unified form. That is, to treat equality and inequality constraints as special cases of bound constraints, we define

$$\xi^- := \begin{bmatrix} b^- \\ d \\ x^- \end{bmatrix}, \quad \xi^+ := \begin{bmatrix} b \\ d \\ x^+ \end{bmatrix}$$
$$E := \begin{bmatrix} A \\ C \\ I \end{bmatrix} \in R^{(3n+m+1)\times(n+1)}.$$

where $b^- \in R^{2n}$, and $\forall\, j \in \{1,\ldots,2n\}$, $b_j^- \ll 0$ sufficiently negative to represent $-\infty$. Then, (10)–(13) are rewritten in the following form:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}x^TQx \\ &\text{subject to} && \xi^- \leqslant Ex \leqslant \xi^+. \end{aligned} \tag{14}$$

In the above formulation, the generalized feasibility region $[\xi^-, \xi^+]$ is constructed as a closed convex set to facilitate the design and analysis of dual neural network via the Karush–Kuhn–Tucker condition and the projection operator.

At any time instant, the constrained quadratic programming problem (10)–(13) may be viewed as a parametric optimization problem. It follows from the Karush–Kuhn–Tucker condition that x is a solution to (14) if and only if there exists $u \in R^{3n+m+1}$, such that $Qx - E^Tu = 0$ and

$$\begin{cases} [Ex]_i = \xi_i^-, & \text{if } u_i > 0 \\ [Ex]_i = \xi_i^+, & \text{if } u_i < 0 \\ \xi_i^- \leqslant [Ex]_i \leqslant \xi_i^+, & \text{if } u_i = 0. \end{cases} \tag{15}$$

The complementary condition (15) is equivalent to the system of piecewise linear equations $Ex = g(Ex - u)$ [24]–[26]. The vector-valued function $g(v) = [g_1(v_1),\ldots,g_{3n+m+1}(v_{3n+m+1})]^T$ is defined as $\forall\ i \in \{1,\ldots,3n+m+1\}$

$$g_i(v_i) = \begin{cases} \xi_i^-, & \text{if } v_i < \xi_i^- \\ v_i, & \text{if } \xi_i^- \leqslant v_i \leqslant \xi_i^+ \\ \xi_i^+, & \text{if } v_i > \xi_i^+ \end{cases} \tag{16}$$

which may include three situations as depicted in Fig. 1 by the definitions of ξ^+ and ξ^-. Therefore, x is a solution to (14) if and only if there exists a dual decision vector u such that $Qx - E^Tu = 0$ and $Ex = g(Ex - u)$; that is

$$\begin{cases} x = Q^{-1}E^Tu \\ g(EQ^{-1}E^Tu - u) = EQ^{-1}E^Tu. \end{cases} \tag{17}$$

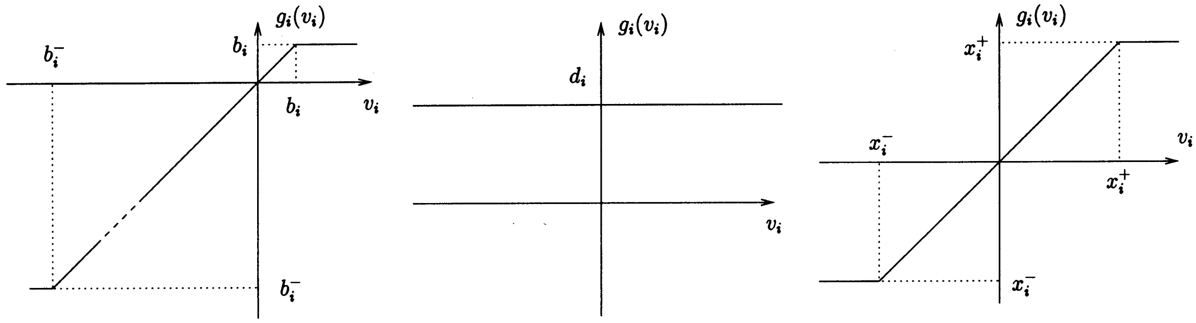


Fig. 1. Projection operator $g_i(v_i)$ on $[\xi_i^-, \xi_i^+]$.

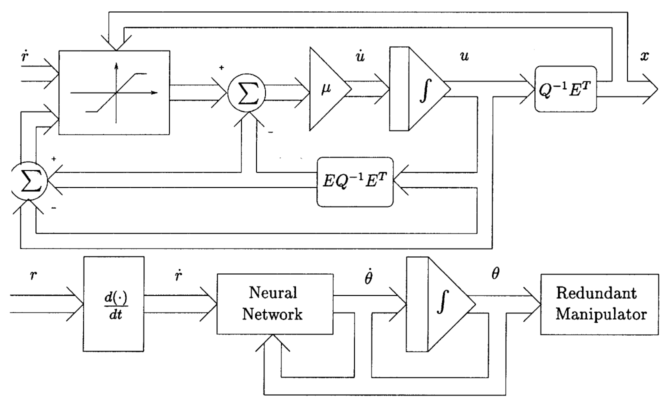


Fig. 2. Block diagrams of the neural network-based inverse kinematic control system with the bi-criteria scheme (5)–(8).

The above optimality condition yields a dual neural network model for solving (14) with the following dynamical equation and output equation:

$$\dot u = \mu \left\{ g(EQ^{-1}E^T u - u) - EQ^{-1}E^T u \right\} \quad (18)$$

$$x = Q^{-1}E^T u \quad (19)$$

where $\mu \in R$ is a positive design parameter to scale the convergence rate of the dual network. For superior online performance, the parameter μ, like an inductance parameter or the reciprocal of a capacitive parameter, is set as large as hardware permits [e.g., in analog circuits or very large scale integration (VLSI)] [21]. For experimental and/or simulative purposes, μ is usually selected between 10^6 and 10^8, similar to previous studies [10], [13]–[20].

The dynamic equation described in (18) shows that the dual neural network is composed of only one layer of no more than $3n+m+1$ neurons and without using any analog multiplier or penalty parameter, as opposed to the neural network approach in [13]. Compared to the prime-dual neural network [15], the dynamic equation of a dual neural network is piecewise linear without any high-order nonlinear term. Consequently, the architecture of the dual neural network to be implemented finally on VLSI is much simpler than those of the existing recurrent neural network approaches. The block diagram of the dual neural network system is depicted in Fig. 2. A circuit realizing the dual neural network consists of summers, integrators, and weighted connections, and the piecewise linear activation function $g_i(\cdot)$ may be implemented by using an operational amplifier known as a limiter [20]–[22]. In the bi-criteria kinematic control process which is delineated in Fig. 2, the desired velocity vector $\dot r(t)$ is input into the dual neural network, and spontaneously, the network outputs the signal $x(t)$, of which the first n elements are the estimated joint velocity vector $\dot\theta(t)$.

In addition, as for the computation load of the dual neural network approach, the proposed bi-criteria formulation (10)–(13) has been converted to an $\{n+1\}$-dimensional quadratic program, while the numerical balancing scheme (4) contains two n-dimensional quadratic programs to solve individually. Under the same computing capability, and for high-DOF robotic systems, the proposed dual neural network approach yields continuous solutions with almost one-half of the computational time as does the balancing scheme (4).

IV. Convergence Results

In this section, we prove the global convergence of the proposed dual neural network for kinematics control of redundant manipulators, and estimate the position tracking error.

Related definitions and a lemma are first presented [15], [16]. A neural network is said to be globally convergent, if starting from any initial point taken in the whole associated Euclidean space, every state trajectory of the neural network converges to an equilibrium point that depends on the initial state of the trajectory. Furthermore, the neural network is said to be globally exponentially convergent if every trajectory starting from any initial point $x(t_0)$ satisfies $\forall\, t \geqslant t_0 \geqslant 0$

$$\|x(t) - x^*\| \leqslant \eta \|x(t_0) - x^*\| \exp(-\lambda(t - t_0))$$

where η and λ are positive constants, x^* is an equilibrium point depending on initial states, and the symbol $\|\cdot\|$ hereafter denotes the Euclidean norm of a matrix or vector, unless specified otherwise. The exponential convergence implies that the system converges arbitrarily fast. The following projection property is often used in optimization literature [27]–[29].

Lemma 1 [25]: Assume that the set $\Omega \subset R^q$ is a closed convex set; then, the following inequality holds:

$$(P_\Omega(v) - \omega)^T (v - P_\Omega(v)) \geqslant 0, \quad \forall\, v \in R^q, \omega \in \Omega$$

where $P_\Omega : R^q \to \Omega$ is a projection operator defined as $P_\Omega(v) = \arg\min_{\omega\in\Omega} \|v - \omega\|$.

It is clear that the set $\Omega := \{v \in R^{3n+m+1} | \xi^- \leqslant v \leqslant \xi^+\}$ is a closed convex set, and $g(\cdot)$ defined in (16) possesses the above projection property. The convergence of the proposed dual neural network for constrained bi-criteria inverse kinematics is then obtained as follows.

Theorem 1: Starting from any initial state, the state vector $u(t)$ of dual neural network (18) is convergent to an equilibrium point u^*, and the output vector $x(t)$ converges to the optimal solution of the bi-criteria inverse kinematics problem (10)–(13). Moreover, the exponential convergence can be achieved, if there exists a constant $\rho > 0$, such that

$$\|g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu\|^2 \geqslant \rho\|u-u^*\|^2.$$

Proof: To show the convergence property, the following numbered inequalities such as (24) are derived. At u^*, we have the inequality property [16], [26], [29]

$$(\omega - EQ^{-1}E^Tu^*)^Tu^* \geqslant 0,\ \forall\,\omega\in\Omega \tag{20}$$

which can be obtained by discussing the following three cases.

Case 1) If for some $i \in \{1,\ldots,3n+m+1\}$, $u_i^* = 0$, $\xi_i^- \leqslant [Ex^*]_i \leqslant \xi_i^+$, then $(\omega_i - [Ex^*]_i)\,u_i^* = 0$.

Case 2) If for some $j \in \{1,\ldots,3n+m+1\}$, $u_j^* > 0$, $[Ex^*]_j = \xi_j^-$, and $\xi_j^- \leqslant \omega_j \leqslant \xi_j^+$, then $\omega_j - [Ex^*]_j \geqslant 0$, and thus, $(\omega_j - [Ex^*]_j)u_j^* \geqslant 0$.

Case 3) If for some $k \in \{1,\ldots,3n+m+1\}$, $u_k^* < 0$, $[Ex^*]_k = \xi_k^+$, and $\xi_k^- \leqslant \omega_k \leqslant \xi_k^+$, then $\omega_k - [Ex^*]_k \leqslant 0$, and thus, $(\omega_k - [Ex^*]_k)\,u_k^* \geqslant 0$.

Therefore, from (20), we have

$$\left(g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu^*\right)^T u^* \geqslant 0. \tag{21}$$

In addition, it follows from *Lemma 1* that, $\forall\, u\in R^{3n+m+1}$

$$\left(g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu^*\right)^T \times \left((EQ^{-1}E^Tu-u)-g(EQ^{-1}E^Tu-u)\right) \geqslant 0. \tag{22}$$

Then, adding (21) and (22) yields

$$\left(g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu^*\right)^T \times \left(u^*+(EQ^{-1}E^Tu-u)-g(EQ^{-1}E^Tu-u)\right) \geqslant 0. \tag{23}$$

Defining $\tilde g := g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu$, it follows from (23) that

$$\left(\tilde g + EQ^{-1}E^T(u-u^*)\right)^T\left((u-u^*)+\tilde g\right) \leqslant 0$$

and thus, we have

$$(u-u^*)^T\tilde g + (u-u^*)^TEQ^{-1}E^T\tilde g \leqslant -\tilde g^T\tilde g - (u-u^*)^TEQ^{-1}E^T(u-u^*) \leqslant 0. \tag{24}$$

Now, we choose a Lyapunov function candidate as

$$V(u(t)) = \frac{1}{2}\|M(u(t)-u^*)\|^2 \tag{25}$$

where the matrix M is symmetric positive definite and $M^2 = (1/\mu)(I+EQ^{-1}E^T) > 0$. Clearly, $V(u)$ is positive definite (i.e., $V>0$ if $u\neq u^*$, and $V=0$ if $u=u^*$) for any u taken in R^{3n+m+1}. In view of (24), we have

$$\begin{aligned}\frac{dV}{dt} &= (u-u^*)^TM^2\dot u\\ &= (u-u^*)^T(I+EQ^{-1}E^T)\tilde g\\ &\leqslant -\|g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu\|^2\\ &\quad -(u-u^*)^TEQ^{-1}E^T(u-u^*)\leqslant 0.\end{aligned} \tag{26}$$

Moreover, $\dot V = 0$ if and only if $u=u^*$, since by the Karush–Kuhn–Tucker optimality condition (17) $\|g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu\| = 0$ amounts to $u=u^*$ only. Thus, by the positive definiteness of $V(u)$ and the negative definiteness of $dV(u)/dt$, it follows from [29, pp. 1334–1336] that the dual neural network (18) is globally convergent to the equilibrium point u^*. Thus, the network output $x(t)$ converges to $Q^{-1}E^Tu^*$, which is the optimal solution to the bi-criteria inverse kinematics problem (10)–(13) according to the output equation.

As for the global exponential convergence, review $V(u)$ and $\dot V$. It follows from (25) that $c_1\|u-u^*\|^2 \leqslant V(u) \leqslant c_2\|u-u^*\|^2$, where $c_2 \geqslant c_1 > 0$ are, respectively, the maximal and minimal eigenvalues of $(1/\mu)(I+EQ^{-1}E^T)$. From (26) and the extra condition (i.e., if there exists a $\rho>0$, such that $\|g(EQ^{-1}E^Tu-u)-EQ^{-1}E^Tu\|^2 \geqslant \rho\|u-u^*\|^2$), we have

$$\begin{aligned}\frac{dV(u)}{dt} &\leqslant -\rho\|u-u^*\|^2-(u-u^*)^TEQ^{-1}E^T(u-u^*)\\ &= -(u-u^*)^T(\rho I+EQ^{-1}E^T)(u-u^*)\\ &\leqslant -\lambda V(u(t))\end{aligned}$$

where $\lambda = \rho/c_2 > 0$ is proportional to the design parameter μ. Thus, we have $V(u(t)) = O(e^{-\lambda(t-t_0)})$, $\forall\, t\geqslant t_0$, and hence, $\|u(t)-u^*\| = O(e^{-\lambda(t-t_0)/2})$, $\forall\, t\geqslant t_0$, which completes the exponential convergence property of the proposed dual neural network.

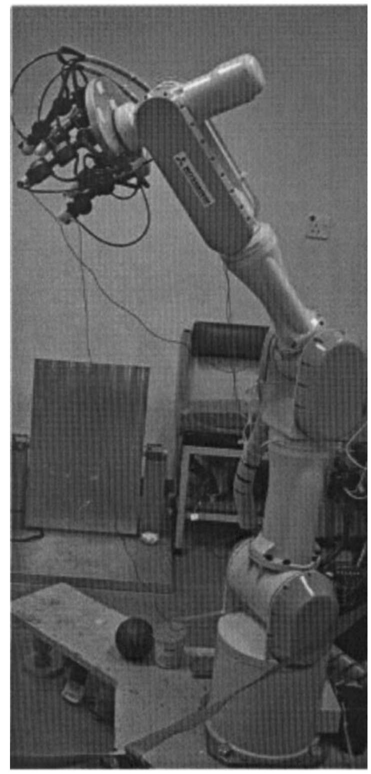


Fig. 3. Seven-DOF PA10 manipulator in the laboratory.

TABLE I
JOINT LIMITS AND JOINT VELOCITY LIMITS OF PA10 MANIPULATOR

#	joint	axis	$\theta^{\pm}$	$\dot\theta^{\pm}$
1	shoulder 1	rotating	$\pm\pi$rad	±1rad/s
2	shoulder 2	pivoting	±1.7637rad	±1rad/s
3	shoulder 3	rotating	$\pm\pi$rad	±2rad/s
4	elbow 1	pivoting	±2.6831rad	±2rad/s
5	elbow 2	rotating	$\pm3\pi/2$rad	$\pm2\pi$rad/s
6	wrist 1	pivoting	$\pm\pi$rad	$\pm2\pi$rad/s
7	wrist 2	rotating	$\pm2\pi$rad	$\pm2\pi$rad/s

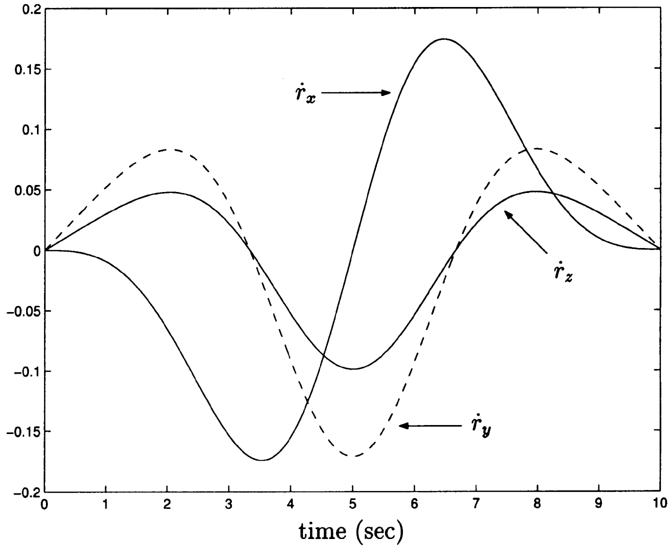


Fig. 4. Demanded end-effector Cartesian velocity $\dot{r}$ in meters per second.

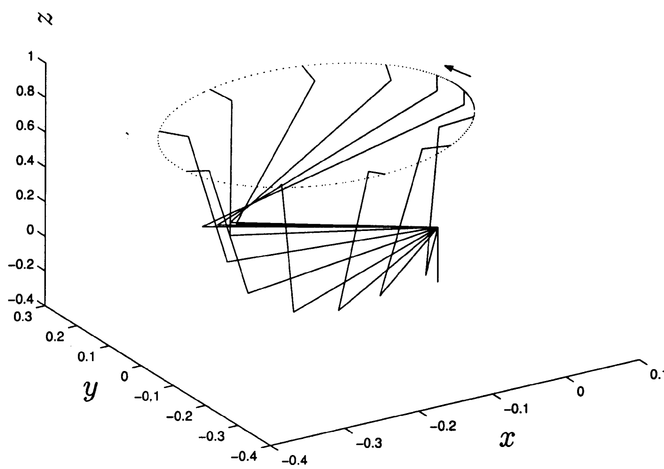


Fig. 5. Motion trajectory of PA10 manipulator under infinity-norm joint velocity minimization, where the radius of the circular path is 0.2 m, and the revolute angle made with x is $\pi/6$ rad.

As the exponential convergence rate is proportional to design parameter μ, the dual neural network can be expedited sufficiently fast as long as hardware permits. $J(\theta)$ and $r(t)$ are, thus, time varying in a time scale sufficiently slower than that of (18). Specifically, starting from any $t_1 \in [0, t_f]$, and within some small time interval Δt, the maximal variation of (J, r) is sufficiently small, while the proposed dual neural network has been asymptotically convergent to the corresponding theoretical solution u^* with a sufficiently small relative error. In the finite-time path-following task, the worst case of $\|u(t) - u^*\|$ and $\|\dot{\theta}(t) - \dot{\theta}^*(t)\|$, $t \in [0, t_f]$ can be estimated on average as ϵ and $\bar{\epsilon}$, respectively, where $\epsilon, \bar{\epsilon} > 0$ depend on $1/\mu$. In view of $\theta^*(0) = \theta(0)$ and $\dot{\theta}^*(0) = \dot{\theta}(0)$, we can estimate the inverse-kinematics joint configuration deviation as

$$\begin{aligned}\|\theta(t) - \theta^*(t)\| &= \| \int_0^t (\dot{\theta} - \dot{\theta}^*) dt \| \\ &\leqslant \int_0^t \|\dot{\theta} - \dot{\theta}^*\| dt \leqslant \int_0^t \bar{\epsilon} dt \leqslant \bar{\epsilon} t\end{aligned}$$

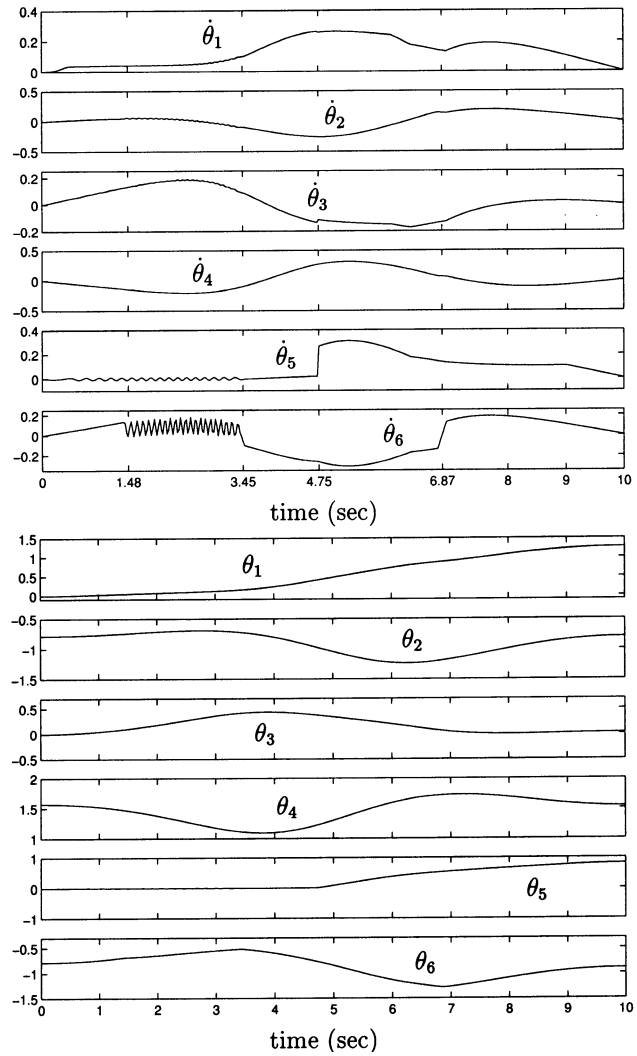


Fig. 6. Profiles of joint velocity variables and joint variables for the circular trajectory under infinity-norm minimization, where $\dot{\theta}_7$ and θ_7 constantly remain zero, and thus, are omitted.

where, theoretically, $\bar{\epsilon}$ can be made arbitrarily small by increasing μ; namely, increasing the convergence rate of (18). Based on the Taylor series expansion of (1), $\exists \hat{\epsilon} > 0$, the position tracking error $\|r(t) - r^*(t)\|$ is bounded by the function $\hat{\epsilon} t$ with $\|r(t_f) - r^*(t_f)\|$ less than $\hat{\epsilon} t_f$, where the coefficient $\hat{\epsilon}$ can be made arbitrarily small by increasing μ.

From the above position error estimation, we know that the error can be made small by increasing μ. Moreover, though a small t_f can lessen the position error too, it may cause high joint velocity or acceleration. The ensuing simulation results will verify the soundness of the proposed error estimation.

V. Simulation Results

Consider the Mitsubishi PA10 robot arm executing a circular trajectory in three-dimensional task space as in Fig. 3. The mechanical configuration and coordinate system of the 7-DOF PA10 manipulator are depicted in [10]. Joint limits $\theta^{\pm}$ and joint velocity limits $\dot{\theta}^{\pm}$ are shown in Table I. In the study, without loss of generality, only the positioning of the manipulator

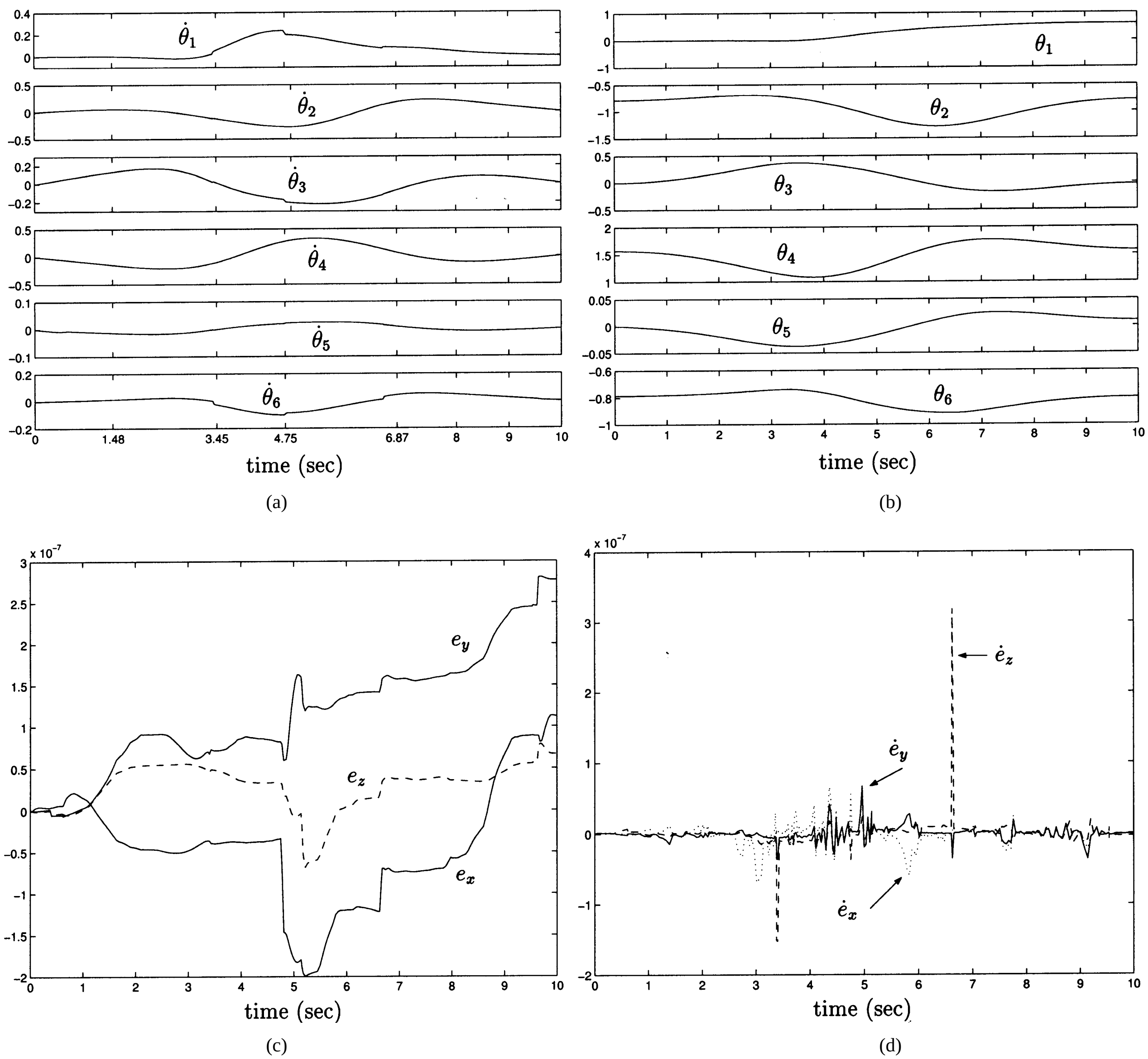


Fig. 7. Transients of the PA10 manipulator tracking the circular trajectory under bi-criteria minimization.

end-effector is considered, the Jacobian matrix is, thus, 3×7 in dimension and the degree of redundancy is four.

Though simulation results [7], [13], [15] based on infinity-norm minimization show that the largest joint velocity in magnitude usually has a reduction of 10%–30% on average, with reference to that by the Euclidean-norm minimization, the discontinuity deficiency of such an infinity-norm solution may occur. For example, in the circular path following task in Figs. 5 and 6, the discontinuity deficiency occurs, where the initial state of the manipulator is $\theta(0) = [0, -\pi/4, 0, \pi/2, 0, -\pi/4, 0]^T$ radian and the desired end-effecter Cartesian velocity $\dot{r}(t)$ are depicted in Fig. 4. During the time period $[1.48, 3.45]$ s, the sixth joint velocity $\dot{\theta}_6$ oscillates around 0.08 rad/s with an amplitude about 0.07 rad/s, which is similar to the ones presented in [8]. Other discontinuous situations are at $t = 4.75$ s for $\dot{\theta}_5$ and at $t = 6.87$ s for $\dot{\theta}_6$. The aforementioned discontinuity cannot be accepted in practice, since the related joint acceleration variables are required to be positive or negative infinity at these time instants.

Now, the proposed dual neural network for bi-criteria kinematic control (5)–(8) is applied to the PA10. The design parameters are selected as $\mu = 10^8$, $\alpha = 0.5$, and $\beta = 2$. The neurally-computed joint velocities and joint variables are depicted in Fig. 7(a) and (b). Clearly, the oscillation and discontinuity problem of minimum infinity-norm kinematic control as depicted in Fig. 6 are solved and no discontinuous point appears in Fig. 7(a). This confirms the theoretical results about the uniqueness property of bi-criteria inverse kinematics solution. Besides, as shown in the figure, all the joint velocities and joint variables have been kept within their mechanical limits. In Fig. 7(c) and (d), the symbols e_x, e_y, and e_z denote the components of tracking position error e, respectively, along the x, y, and z axes of the base frame, and $\dot{e}_x$, $\dot{e}_y$, and $\dot{e}_z$ denote, respectively, the x, y, and z axis components of tracking velocity error $\dot{e}$ at the end-effector of the robot arm. As shown in Fig. 7(c) and (d), the Cartesian position and velocity errors obtained by the proposed dual neural network are, respectively, less than 3×10^{-4} mm and 4×10^{-4} mm/s.

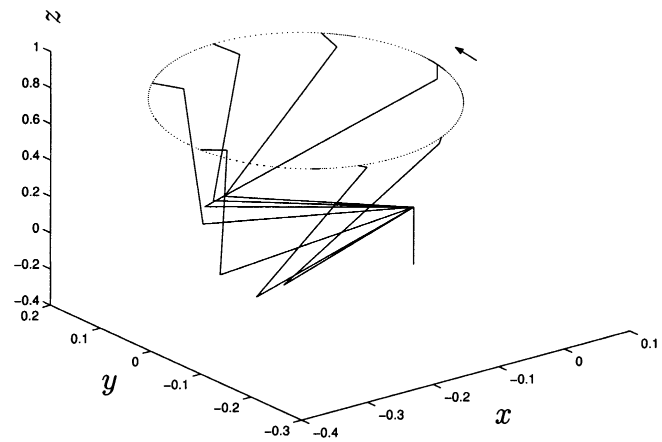


Fig. 8. Motion trajectory of PA10 manipulator under bi-criteria minimization.

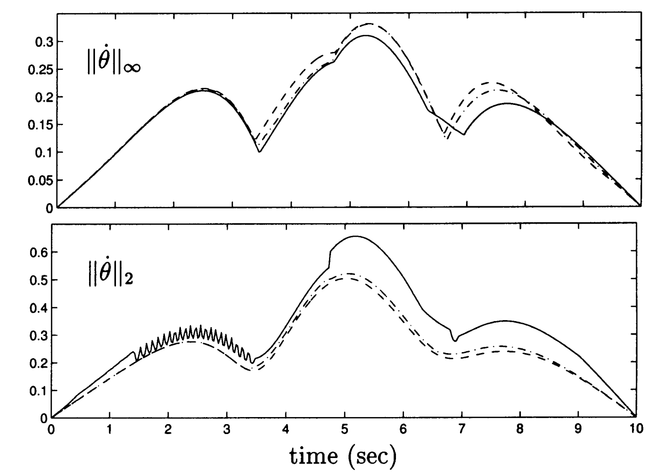


Fig. 9. Infinity-norm and Euclidean-norm comparison of PA10 joint velocity vectors under different optimization schemes, where the solid, dash-dotted, and dashed curves correspond to the minimum-effort, bi-criteria, and minimum-power schemes, respectively.

This demonstrates the capability of the proposed dual neural network for online bi-criteria kinematic control of physically constrained redundant manipulators.

Corresponding to Fig. 7, the PA10 motion trajectory under bi-criteria minimization is illustrated in Fig. 8. Comparative results between the minimum effort, bi-criteria, and minimum-power kinematic control schemes are shown in Fig. 9; that is, to compare the infinity-norm and the Euclidean-norm of PA10 joint velocity vectors neurally computed under the aforementioned three performance measures. In detail, in Fig. 9, the solid curves of the upper and lower subplots correspond, respectively, to the infinity-norm and the Euclidean-norm of minimum-effort joint velocity; the dash-dotted curves correspond to those of the joint velocity under bi-criteria minimization; the dashed curves correspond to those under minimum-power scheme. The comparison shows that the maximal amplitude and power consumption of bi-criteria solutions are usually between those of the minimum-effort and the minimum-power solutions. As opposed to the minimum Euclidean-norm solution, the bi-criteria solution always has a smaller maximal magnitude of $\dot{\theta}$, which means less impact and less pain to the body tissue in a robotic surgery. Moreover, removal of discontinuity points of the pure minimum infinity-norm solution also implies no sudden impact again. Hence, compared to single-criterion kinematic control schemes, the bi-criteria scheme and the proposed neural network-based solver are much more flexible in the sense that they can yield any combination of the minimum-effort and minimum-power solutions as needed, in addition to remedying the discontinuity deficiency of pure minimum-effort solutions.

It is worth pointing out the generalizability of the above simulation, where only the positioning of the manipulator end-effector is concerned. The formulation, design, and analysis of the bi-criteria dual neural network in Sections II–IV show that the approach is capable of handling inverse kinematic control of both the position and the orientation of the end-effector. The reason for considering only the end-effector position is that as the degree-of-redundancy increases, the discontinuity deficiency of minimum infinity-norm solution is empirically more possible to occur.

VI. Concluding Remarks

The proposed one-layer dual neural network model provides a new parallel distributed computational approach to online bi-criteria kinematic control of physically constrained redundant manipulators. The involved bi-criteria optimization scheme is a suitable remedy for the discontinuity problem usually occurring in minimum infinity-norm solutions. Compared with the supervised learning neural networks for robot kinematic control, the present approach eliminates the need of off-line training, and guarantees fast convergence, due to the exponential convergence. Compared with other recurrent neural network approaches, the proposed dual neural network is designed by optimizing the weighted infinity-norm and Euclidean-norm of joint velocity subject to bound constraints, and hence, able to resolve manipulator redundancy under physical constraints such as joint limits and joint velocity limits. Moreover, the dynamic equation of the dual neural network is piecewise linear and does not contain any high-order nonlinear term, and thus, the architecture is very simple. The simulation results based on the PA10 robot manipulator demonstrate that the dual neural network is capable for online kinematic control of joint-constrained redundant manipulators.

References

[1] L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators*. London, U.K.: Springer-Verlag, 2000.

[2] A. A. Maciejemski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. Robot. Res.*, vol. 4, no. 3, pp. 109–117, 1985.

[3] T. Yoshikawa, "Manipulability of robot mechanisms," *Int. J. Robot. Res.*, vol. 4, no. 2, pp. 3–9, 1985.

[4] B. Allotta, V. Colla, and G. Bioli, "Kinematic control of robot manipulators with joint constraints," *ASME J. Dynam. Syst., Meas., Contr.*, vol. 121, no. 3, pp. 433–442, 1999.

[5] C. A. Klein and C. H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC–13, pp. 245–250, 1983.

[6] F.-T. Cheng, R.-J. Sheu, and T.-H. Chen, "The improved compact QP method for resolving manipulator redundancy," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 1521–1530, Nov. 1995.

[7] A. S. Deo and I. D. Walker, "Minimum effort inverse kinematics for redundant manipulators," *IEEE Trans. Robot. Automat.*, vol. 13, pp. 767–775, Oct. 1997.

[8] I. A. Gravagne and I. D. Walker, "On the structure of minimum effort solutions with application to kinematic redundancy resolution," *IEEE Trans. Robot. Automat.*, vol. 16, pp. 855–862, Dec. 2000.

[9] J. Lee, "A structured algorithm for minimum l_∞-norm solutions and its application to a robot velocity workspace analysis," *Robotica*, vol. 19, pp. 343–352, 2001.

[10] J. Wang, Q. Hu, and D. Jiang, "A Lagrangian network for kinematic control of redundant manipulators," *IEEE Trans. Neural Networks*, vol. 10, pp. 1123–1132, May 1999.

[11] Z. Mao and T. C. Hsia, "Obstacle avoidance inverse kinematics solution of redundant robots by neural networks," *Robotica*, vol. 15, pp. 3–10, 1997.

[12] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*, Singapore: World Scientific, 1998.

[13] H. Ding and S. K. Tso, "A fully neural network-based planning scheme for torque minimization of redundant manipulators," *IEEE Trans. Ind. Electron.*, vol. 46, pp. 199–206, Jan. 1999.

[14] H. Ding and J. Wang, "Recurrent neural networks for minimum infinity-norm kinematic control of redundant manipulators," *IEEE Trans. Syst., Man, Cybern. A*, vol. 29, pp. 269–276, Mar. 1999.

[15] W. S. Tang and J. Wang, "A recurrent neural network for minimum infinity-norm kinematic control of redundant manipulators with an improved problem formulation and reduced architecture complexity," *IEEE Trans. Syst., Man, Cybern.*, vol. 31, pp. 98–105, Jan. 2001.

[16] Y. Xia and J. Wang, "A dual neural network for kinematic control of redundant robot manipulators," *IEEE Trans. Syst., Man, Cybern. B*, vol. 31, pp. 147–154, Jan. 2001.

[17] J. Wang, "Recurrent neural networks for computing pseudoinverses of rank-deficient matrices," *SIAM J. Sci. Comput.*, vol. 18, no. 5, pp. 1479–1493, 1997.

[18] Y. Xia, "A new neural network for solving linear programming problems and its application," *IEEE Trans. Neural Networks*, vol. 7, pp. 525–529, Mar. 1996.

[19] I. B. Pyne, "Linear programming on an electronic analogue computer," *Trans. Amer. Inst. Elect. Eng.*, vol. 75, pp. 139–143, 1956.

[20] D. W. Tank and J. J. Hopfield, "Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit," *IEEE Trans. Circuits Syst.*, vol. CAS–33, pp. 533–541, 1986.

[21] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.

[22] C. Diorio and R. P. N. Rao, "Neural circuits in silicon," *Nature*, vol. 405, pp. 891–892, 2000.

[23] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming—Theory and Algorithms*. New York: Wiley, 1993.

[24] O. L. Mangasarian, "Solution of symmetric linear complementary problems by iterative methods," *J. Optim. Theory Applicat.*, vol. 22, no. 2, pp. 465–485, 1979.

[25] D. P. Bertsekas, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[26] W. Li and J. Swetits, "A new algorithm for solving strictly convex quadratic programs," *SIAM J. Optim.*, vol. 7, no. 3, pp. 595–619, 1997.

[27] J.-S. Pang, "A posterior error bounds for the linearly-constrained variational inequality problem," *Math. Oper. Res.*, vol. 12, pp. 474–484, 1987.

[28] J.-S. Pang and J.-C. Yao, "On a generalization of a normal map and equation," *SIAM J. Contr. Optim.*, vol. 33, pp. 168–184, 1995.

[29] Y. Xia and J. Wang, "A general methodology for designing globally convergent optimization neural networks," *IEEE Trans. Neural Networks*, vol. 9, pp. 1331–1343, Nov. 1998.

[30] ——, "Global exponential stability of recurrent neural networks for solving optimization and related problems," *IEEE Trans. Neural Networks*, vol. 11, pp. 1017–1022, July 2000.

[31] ——, "Global asymptotic and exponential stability of a dynamic neural system with asymmetric connection weights," *IEEE Trans. Automat. Contr.*, vol. 46, pp. 635–638, Apr. 2001.

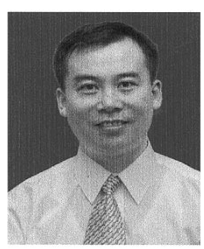
Yunong Zhang (S'02) received the B.S. and M.S. degrees in automatic control engineering from the Huazhong University of Science and Technology and the South China University of Technology, Guangzhou, China, in 1996 and 1999, respectively. He is working toward the Ph.D. degree in the Department of Automation and Computer-Aided Engineering, the Chinese University of Hong Kong.

His research interests include nonlinear systems, robotics, neural networks, and signal processing.

Jun Wang (S'89–M'90–SM'93) received the B.S. degree in electrical engineering and the M.S. degree in systems engineering from Dalian University of Technology, DaLian, China. He received the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH.

He is currently a Professor of Automation and Computer-Aided Engineering at the Chinese University of Hong Kong. Prior to coming to Hong Kong in 1995, he was an Associate Professor at the University of North Dakota, Grand Forks. His current research interests include neural networks and their engineering applications.

Dr. Wang is an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS and the IEEE TRANSACTIONS ON SYSTEMS, MAN, and CYBERNETICS.



Yangsheng Xu (S'89–M'90–SM'95) received the Ph.D. degree from the University of Pennsylvania, Philadelphia, in 1989.

He joined Carnegie Mellon University, Pittsburgh, PA in 1989. Since January 1997, he has been with The Chinese University of Hong Kong, where he is heading the Department of Automation and Computer-Aided Engineering. His current research interests are in the areas of aerospace robotics, human-interface that learns, and sensors.

Dr. Xu served as an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION.