Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 70 (2006) 513–524
NEUROCOMPUTING
www.elsevier.com/locate/neucom

A set of nonlinear equations and inequalities arising in robotics and its online solution via a primal neural network

Yunong Zhang*

Hamilton Institute, National University of Ireland at Maynooth, Maynooth, Co. Kildare, Ireland

Received 10 January 2005; received in revised form 25 November 2005; accepted 30 November 2005
Communicated by F.A. Mussa-Ivaldi
Available online 12 May 2006

Abstract

In this paper, for handling general minimum-effort inverse-kinematic problems, the nonuniqueness condition is investigated. A set of nonlinear equations and inequality is presented for online nonuniqueness-checking. The concept and utility of primal neural networks (NNs) are introduced in this context of dynamical inequalities and constraints. The proposed primal NN can handle well such a nonlinear online-checking problem in the form of a set of nonlinear equations and inequality. Numerical examples demonstrate the effectiveness and advantages of the primal NN approach.

Keywords: Minimum effort inverse kinematics; Nonuniqueness; Discontinuity; Nonlinear equations and inequalities; Primal neural network

1. Introduction

Redundant manipulators are robots having more degrees-of-freedom (DOF) than required to perform a given end-effector task [15]. Our human arm is also such a redundant system [8]. One fundamental issue in operating such systems is the redundancy-resolution problem [25,29,30]. By resolving the redundancy, the robots can avoid obstacles, joint physical limits, as well as to optimize various performance criteria. In addition to minimum-energy redundancy resolution [15,25,30], another interesting approach is the minimum-effort redundancy resolution [3,5,9,16,31]. The minimum-effort resolution may encounter discontinuities due to the nonuniqueness of the solution at some time instants [6]. The nonuniqueness of the minimum-effort solution was previously checked by a geometrical method [6]. As shown in this paper, the nonuniqueness can be checked more generally and efficiently by solving online a set of nonlinear equations and inequalities.

It is worth mentioning that in view of its fundamental role arising in numerous fields of science and engineering, the problem of solving groups of linear equations and inequalities has been investigated extensively for the past decades. For example, about the recent research based on recurrent neural networks (NNs) (specifically, the Hopfield-type NNs), see [2,10,11,18,19,21,26,27] and the references therein. The NN approach is now thought to be a powerful tool for online solutions, in light of its parallel distributed computing nature and hardware implementability [1,4,13,19].

In the aforementioned literature, most researches are devoted to solving linear equations and inequalities only (with the matrix-inverse problem as a special case). Among them, the mathematical description can be unified as $Ey\leqslant f$, $Cy=g$ where y is the unknown vector to be determined, given coefficient matrices E, C and coefficient vectors f, g. In the group of linear equations and inequalities, $Cy=g$ defines the problem to be solved while $Ey\leqslant f$ defines its constraints. To our best knowledge, there have been few research results about solving nonlinear or hybrid equations/inequalities based on NN approaches. On the other hand, the nonlinear situations are encountered in the minimum-effort redundancy resolution of robot

*Tel.: +353 17084534; fax: +353 17086269.
E-mail address: ynzhang@ieee.org.

0925-2312/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2005.11.006

manipulators. The set of such nonlinear equations and inequalities is in the form of $Ey \leqslant f$, $Cy = g$ and $y^{\mathrm{T}}y = h$ with given scalar $h > 0$. The real-time computation requirement becomes stringent, especially for sensor-based robotic systems of high DOF.

The design methods and techniques for recurrent NNs are briefly reviewed as follows. In terms of the types of decision variables involved, the NN models can be divided into two classes: the pure primal NNs and the nonprimal NNs. In detail, we have the following concepts.

Concept 1: A recurrent NN is called *primal* NN, if the network dynamic equation and implementation only use the original (or to say, *primal*) decision variables, y. For example, the neural models in [2,19,27] are primal NNs.

Concept 2: A recurrent NN is called *nonprimal* NN, if the network dynamic equation and implementation use any auxiliary decision variables, in addition to the original ones. For example, the primal–dual NNs in [29,30], the dual NNs in [20,25], and the Lagrange NN in [22].

The earlier primal NN models such as [19] use finite penalty parameters and generate approximate solutions only. By using auxiliary variables like dual decision variables, the nonprimal NNs usually have better convergence to exact/theoretical solutions as compared to primal NNs [20]. As the recent research shows [20,29], however, the primal NN is preferable in the context of dynamic constraints/inequalities, provided that the solution accuracy is acceptably good. In that case, the number of constraints/inequalities is dynamically changed, and thus the hardware implementation of nonprimal NNs could become less favorable, not to mention the network complexity due to using auxiliary neurons.

In this paper, motivated by the above observations, we generalize the NN design experience to handling the nonlinear/hybrid situations in minimum-effort robotics. The concept of primal NNs is thus formally proposed in solving online a set of nonlinear equations and inequalities. The remainder of this paper is organized in four sections. The problem formulation is given in Section 2. The primal NN, together with convergence results, is presented in Section 3. Illustrative numerical examples are discussed in Section 4. Lastly, Section 5 concludes this paper with final remarks. The main contributions of the paper are as follows:

(1) Based on linear programming and equation-solving, a more efficient criterion is proposed for checking nonuniqueness-points (as compared to geometric methods).
(2) The concept, utility and advantages of primal NNs are illustrated by developing a primal NN for such an online-checking criterion of nonuniqueness (as compared to nonprimal NNs).
(3) Previous robotic research results are confirmed by the conducted computer simulations and analysis. That is, nonuniqueness is nearly sufficient for the appearance of discontinuities in the manipulators with low DOF/redundancy.
(4) New robotic research results are summarized for the manipulators with high DOF/redundancy. That is, nonuniqueness is generally a necessary condition to the appearance of discontinuities in minimum-effort inverse-kinematic solutions.

2. Problem formulation

The minimum-effort redundancy resolution of robot manipulators is the following infinity-norm minimization problem [3,9,16]:

$$\text{minimize} \quad \|\dot{\theta}\|_\infty \tag{1}$$

$$\text{subject to} \quad J\dot{\theta} = \dot{r}, \tag{2}$$

$$\theta^- \leqslant \theta \leqslant \theta^+, \tag{3}$$

$$\dot{\theta}^- \leqslant \dot{\theta} \leqslant \dot{\theta}^+, \tag{4}$$

where $\theta \in R^n$ is the joint vector with physical limits $[\theta^-, \theta^+]$, $\dot{\theta} \in R^n$ is the joint-velocity vector with physical limits $[\dot{\theta}^-, \dot{\theta}^+]$, and $\dot{r} \in R^m$ is the commanded Cartesian velocity vector at the manipulator's end-effector that can be planned off-line or given in real time. The manipulator Jacobian matrix is an $m \times n$ rectangular matrix with $m \leqslant n$, and the infinity-norm of a vector is defined as $\|\dot{\theta}\|_\infty = \max_{1 \leqslant i \leqslant n} |\dot{\theta}_i|$.

Remark 1. It is worth mentioning here the significance of the minimum-effort redundancy resolution. As we know [3,31], the infinity-norm solution as in (1) minimizes the largest component of joint velocity vector, and is thus more consistent with joint physical limits. In addition to such a pursuit of low individual magnitude, the minimum-effort solution could be used for even distribution of workload (if joint torque is minimized). As we also know [25,28], the minimum-energy redundancy resolution could be performed by minimizing the two norm of joint velocity/acceleration/torque. Being another scenario, the minimum-effort solution could complement such a research on manipulators' redundancy resolution. This may further provide insights into human and animal motion control and diversity analysis. For example, why could we—human being—walk, run, and dance in different ways [7,8,23]? Like, sometimes with minimum-energy strategies and sometimes with minimum-effort strategies?

2.1. Geometrical nonuniqueness-checking

Bound constraints (3) and (4) are incorporated here for the avoidance of joint limits and joint velocity limits, respectively. In the geometrical method proposed in [6] for checking the nonuniqueness, bound constraints (3) and (4) were not considered. It was shown there that a discontinuity point of minimum-effort solution appears because of the nonuniqueness of the solution at some time instants. In other words, if the manipulator trajectory orients the solution space so that it is parallel to a hypercube face (which means the existence of multiple solutions), the solution may jump from one edge/point to another instead of continuing smoothly on its way [6].

This might be shown more clearly in Fig. 1, where the solution space is defined as $\mathscr{S} = \{\dot{\theta} | J\dot{\theta} = \dot{r}\}$. When $\mathscr{S}$ is parallel to a cube face (here $\dot{\theta} \in R^3$ just for the convenience of graphic interpretation), there exist multiple solutions as shown in Fig. 1(a) via a bold line. When $\mathscr{S}$ is not parallel to the cube faces, there exists only one solution being the intersection of solution space $\mathscr{S}$ and a cube face/edge, as shown in Fig. 1(b). As coefficients J and $\dot{r}$ change, the solution space $\mathscr{S}$ changes accordingly (based on a member of multiple solutions). A jump might thus be generated from the previous solution point to the current one. The magnitude of the jump depends on the distance between the two solution points. The jump (which becomes a discontinuity point if the jump magnitude is large) happens because of the nonuniqueness of the solution point; specifically, from the existence of multiple solutions to the existence of a unique solution or the existence of another set of multiple solutions.

A measure of such a nonuniqueness point of the minimum-effort solution is computed by the following zero subspace-angle method [6]. Define the nullspace of J by $\mathscr{N}$ represented with a matrix $N \in R^{n\times(n-m)}$ such that the orthogonal columns of N span the nullspace. The subspace angle v is defined as the angle between two hyperplanes embedded in a high-dimension space: $v(\mathscr{S}_1, \mathscr{S}_2) = \cos^{-1}(\min(\mathrm{diag}(\Sigma)))$ with $U\Sigma V^{\mathrm{T}} = S_1^{\mathrm{T}} S_2$ where subspaces $\mathscr{S}_1$ and $\mathscr{S}_2$ are represented with matrices S_1 and S_2, respectively. If the solution is nonunique then there exists some $\mathscr{S}_2$ spanned by elementary basis vectors such that $v = 0$. Assign $S_1 = N$ since $\mathscr{S}$ and $\mathscr{N}$ are parallel linear spaces. There are $n!/(m!(n-m)!)$ possible arrangements that m basis vectors span $\mathscr{S}_2$. For simplicity, a number, $d_1 = \min_{S_2}(|\det(N_1)|)$, was used to represent the subspace angle v rather than to find its real value, where N_1 is a matrix generated by the following rows swap of the concatenation matrix $[N|S_2]$:

$$[N|S_2] \rightarrow \begin{bmatrix} N_1 & 0 \\ N_2 & I \end{bmatrix} \quad \forall S_2.$$

Evidently, to check this zero subspace-angle condition for possible nonuniqueness points, several serial-processing techniques have to be used. Singular value decomposition, rows swap, and determinants minimization are entailed. In the ensuing subsection, in contrast to the geometrical formulation, a more straightforward optimization-based criterion is proposed for checking nonuniqueness points.

2.2. Optimization-based nonuniqueness-checking

We have the following lemma by transforming the minimum-effort redundancy resolution, (1)–(4), into a linear-programming (LP) problem to solve.

Lemma 1 (*Ge et al. [5], Zhang et al. [25,31]*). *The n-dimensional minimum-effort redundancy-resolution problem can be reformulated as an $(n+1)$-dimensional linear program*:

$$\begin{aligned} &\text{minimize} && p^{\mathrm{T}} x && (5)\\ &\text{subject to} && Ax \leqslant 0, && (6)\\ & && Cx = d, && (7)\\ & && x^- \leqslant x \leqslant x^+, && (8) \end{aligned}$$

where the augmented decision vector $x := [\dot{\theta}^{\mathrm{T}}, s]^{\mathrm{T}} \in R^{n+1}$ with s representing $\|\dot{\theta}\|_\infty$, and coefficients $p := [0, \ldots, 0, 1]^{\mathrm{T}} \in R^{(n+1)}$, $C := [J \;\; 0] \in R^{m\times(n+1)}$, $d := \dot{r}$, $x^\pm$ are the bounds of x, and

$$A := \begin{bmatrix} I & -1_{\mathrm{v}} \\ -I & -1_{\mathrm{v}} \end{bmatrix},$$

with $1_{\mathrm{v}} = [1, \ldots, 1]^{\mathrm{T}} \in R^n$ denoting a vector composed of ones.

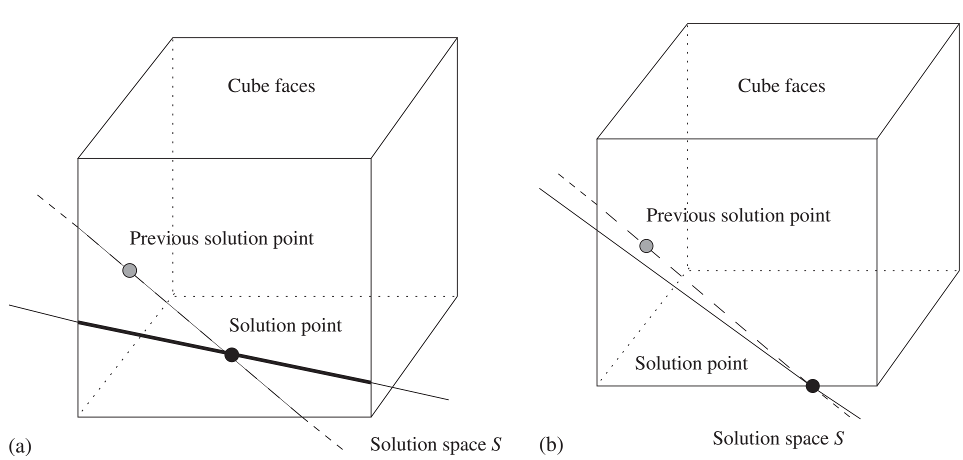

Fig. 1. A simple three-dimensional interpretation of minimum-effort solution, where $J \in R^{2\times3}$ and the solution space $\mathscr{S}$ is one-dimensional [6]: (a) multiple solutions and (b) unique solution.

Remark 2. The above lemma has reformulated the physically-constrained minimum-effort inverse-kinematics problem (1)–(4) into a time-varying linear-programming problem subject to hybrid constraints i.e., in (5)–(8). This reformulation separates major mathematic problems from originally very complex robotic contexts, making the inverse-kinematics task much clearer and easier to implement. In addition, we have used bound constraint (8) to accommodate joint physical limits (3) and (4) in the following form [5,31]:

$$x^- = \begin{bmatrix} \max(\kappa_p(\theta^- - \theta), \dot{\theta}^-) \\ 0 \end{bmatrix},$$

$$x^+ = \begin{bmatrix} \min(\kappa_p(\theta^+ - \theta), \dot{\theta}^+) \\ +\infty \end{bmatrix},$$

where κ_p is a design parameter determining the deceleration magnitude when robot joints approaching their physical limits. $+\infty$ here represents a very large positive number for numerical/simulative purposes [5,25,28,30,31]. For vectors u and v of the same dimension, the vector-valued operators $\max(u, v)$ and $\min(u, v)$ are defined as $\max(u, v) := [\max(u_1, v_1), \max(u_2, v_2), \dots]^T$ and $\min(u, v) := [\min(u_1, v_1), \min(u_2, v_2), \dots]^T$, respectively.

Remark 3. Conventionally, the joint physical limits could be imposed via a diagonal matrix (or a symmetric positive-definite matrix); e.g., as in the weighted pseudoinverse case [15]. The optimization-based redundancy resolution [5,25,31] could also incorporate this kind of weighting matrices. Like, minimizing a performance index weighted by a diagonal matrix (or a positive-definite matrix) to avoid joint physical limits. Some work has been done along this line [25,28]. However, by the author's numerical experience, the minimization of a performance index is just desirable but not compulsory. This means the joint limits imposed via a diagonal weighting matrix could still be violated. In contrast, the inequality/bound constraints imposing the joint physical limits [such as, in (8)] could be maintained more compulsorily. To an LP/QP solver, the priority of constraints is usually higher than that of a performance index [14].

After the transformation via Lemma 1 to an LP problem, we could introduce the following lemma to achieve the online checking of nonuniqueness, especially for high-DOF manipulators.

Proposition 1. *A solution x^* to linear program (5)–(8) at time instant t is nonunique if there exists a solution $y \in R^{n+1}$ to the following simultaneously*:

$$Ey \leqslant 0, \quad Cy = 0, \quad y^T y = 1, \tag{9}$$

where the coefficient matrix $E := [p, A_{\mathscr{K}}^T, I_{\mathscr{K}_+}^T, -I_{\mathscr{K}_-}^T]^T$ with the active-constraint index sets defined as $\mathscr{K} = \{i | A_i x^* = 0\}$, $\mathscr{K}_+ = \{i | x_i^* = x_i^+\}$ and $\mathscr{K}_- = \{i | x_i^* = x_i^-\}$, and A_i denotes the ith row of matrix A.

Proof. By Theorem 2 in [12], the nonuniqueness of solution x^* to (5)–(8) is the existence of y satisfying simultaneously

$$Cy = 0, \quad p^T y \leqslant 0, \quad A_{\mathscr{K}} y \leqslant 0,$$

$$I_{\mathscr{K}_+} y \leqslant 0, \quad -I_{\mathscr{K}_-} y \leqslant 0, \quad y \neq 0.$$

After defining E as in the proposition, the condition for x^* nonuniqueness is the existence of nonzero solution $y \neq 0 \in R^{n+1}$ to the following simultaneously:

$$Cy = 0, \quad Ey \leqslant 0. \tag{10}$$

It follows from solution characteristics [14] that if there exists a nonzero solution y to (10), the solution is evidently not unique but in an open convex set containing $y = 0$. This means that in that case, a solution y with $\|y\|_2 = 1$ exists. Thus, the existence of a nonzero y to (10) is the solution existence of the following simultaneous equations and inequality: $Cy = 0$, $Ey \leqslant 0$, $y^T y = 1$; i.e., (9). In other words, if a solution to (9) exists, then the solution x^* to (5)–(8) is nonunique. □

Proposition 1 converts a pure robotic issue into a mathematical problem of solving online a set of equations and inequalities during the task duration $[t_0, t_f]$; i.e., in Eq. (9). Such a new problem formulation is straightforward. The efficient parallel solution is worth exploring, especially in view of high DOF (e.g., in highly-redundant robots of serpentine or elephant-trunk type) and small sampling interval for high precision control.

3. Primal neural network

Locating the nonuniqueness points of minimum-effort inverse kinematics has now become the solution to a set of nonlinear/hybrid equations and inequality. The problem of interest can be generalized to the following form:

$$Ey \leqslant f, \quad Cy = g, \quad y^T y = h, \tag{11}$$

where, in our situation, $f = 0$, $g = 0$, $h = 1$. The aforementioned linear situation is a special case of (11). Here, we concentrate on the recurrent NN approach.

As shown in Concepts 1 and 2, there could exist two kinds of recurrent NNs: primal NNs and nonprimal NNs. Review Proposition 1 for the definition of coefficient matrix E in (11). E is of a dynamic size in the sense that the number of its rows varies. If we apply nonprimal NNs [by introducing dual decision variables or slack/surplus variables for (11)], the number of auxiliary neurons varies accordingly, making hardware implementation less favorable. Thus, in the context of dynamic constraints/inequalities, we propose the following primal NN to solve problem (11):

$$\dot{y} = -\gamma\{\alpha C^T(Cy - g) + E^T \max(Ey - f, 0) + (y^T y - h)y\}, \tag{12}$$

where design parameter $\gamma > 0$ determines the convergence rate of the network, and α is a large penalty parameter.

Related to developing the above neural dynamics, we have the following lemmas.

Lemma 2. *The derivative of* $\|\sqrt{\alpha}(Cy-g)\|_2^2$ *with respect to* y *is* $2\alpha C^{\mathrm{T}}(Cy-g)$; *in mathematics*,

$$\frac{\mathrm{d}\|\sqrt{\alpha}(Cy-g)\|_2^2}{\mathrm{d}y} = 2\alpha C^{\mathrm{T}}(Cy-g).$$

Proof. It follows from $\|\sqrt{\alpha}(Cy-g)\|_2^2 = \alpha(Cy-g)^{\mathrm{T}}(Cy-g)$ and simple matrix calculus [20,27]. □

Lemma 3. *The derivative of* $\|\max(Ey-f,0)\|_2^2$ *with respect to* y *is* $2E^{\mathrm{T}}\max(Ey-f,0)$; *in mathematics*,

$$\frac{\mathrm{d}\|\max(Ey-f,0)\|_2^2}{\mathrm{d}y} = 2E^{\mathrm{T}}\max(Ey-f,0).$$

Proof. See Appendix A. □

Lemma 4. *The derivative of* $(y^{\mathrm{T}}y-h)^2/2$ *with respect to* y *is* $2(y^{\mathrm{T}}y-h)y$; *in mathematics*,

$$\frac{\mathrm{d}(y^{\mathrm{T}}y-h)^2/2}{\mathrm{d}y} = 2(y^{\mathrm{T}}y-h)y.$$

Proof. See Appendix B. □

In our specific robotic context, in view of $f=0$, $g=0$, $h=1$, the dynamics of the primal NN (12) thus reduces to

$$\dot{y} = -\gamma\{\alpha C^{\mathrm{T}}Cy + E^{\mathrm{T}}\max(Ey,0) + (y^{\mathrm{T}}y-1)y\}, \qquad (13)$$

and we have the following network-convergence results.

Proposition 2. *Starting from any nonzero initial condition, the state of primal NN* (13) *converges to an equilibrium* y^*; *and if*

$$U(y^*) := \|\sqrt{\alpha}Cy^*\|_2^2 + \|\max(Ey^*,0)\|_2^2 + (y^{*\mathrm{T}}y^*-1)^2/2 = 0,$$

then a nonuniqueness point exists in the original inverse-kinematic solution.

Proof. Network (13) is designed based on the gradient-descent method, and its convergence are thus easy to prove. Construct the energy function

$$U(y) = \|\sqrt{\alpha}Cy\|_2^2 + \|\max(Ey,0)\|_2^2 + (y^{\mathrm{T}}y-1)^2/2, \qquad (14)$$

where $U(y)=0$ is equivalent to the solution existence of (9). In view of Lemmas 2–4 with $f=0$, $g=0$ and $h=1$, the time derivative of $U(y)$ along network trajectories (13) is

$$\begin{aligned}\frac{\mathrm{d}U}{\mathrm{d}t} &= \frac{\partial U(y)}{\partial y}^{\mathrm{T}}\frac{\mathrm{d}y}{\mathrm{d}t}\\ &= -2\gamma\|\alpha C^{\mathrm{T}}Cy + E^{\mathrm{T}}\max(Ey,0) + (y^{\mathrm{T}}y-1)y\|_2^2\\ &\leqslant 0.\end{aligned}$$

According to Lyapunov theory [17,31], network (13) is stable and globally converges to an equilibrium y^* in the set, $Y = \{y^* \in R^{n+1} | \alpha C^{\mathrm{T}}Cy^* + E^{\mathrm{T}}\max(Ey^*, 0) + (y^{*\mathrm{T}}y^*-1)y^* = 0\}$, in view of $\dot{U}(y^*)=0$ being $\dot{y}^*=0$. There are two possibilities about y^*: case (i) if there is no solution to (9), then the network may converge to any equilibrium $y^* \in Y$, like the trivial one $y^*=0$, of which $U(y^*)\neq 0$; case (ii) if a solution to (9) exists, say y^*, then the solution y^* must be in Y and $U(y^*)=0$ due to $Cy^*=0$, $Ey^*\leqslant 0$ and $y^{*\mathrm{T}}y^*-1=0$; i.e., if $U(y^*)=0$, then by Lemma 1 and Proposition 1 a nonuniqueness point exists. □

Proposition 3. *Define the solution set* $Y_{\mathrm{s}} = \{y^* | Cy^*=0, Ey^*\leqslant 0, y^{*\mathrm{T}}y^*-1=0\}$ *and the equilibrium set* $Y = \{y^* | \alpha C^{\mathrm{T}}Cy^* + E^{\mathrm{T}}\max(Ey^*,0) + (y^{*\mathrm{T}}y^*-1)y^*=0\}$ *for primal NN* (13). *If* $Y_{\mathrm{s}}\neq\emptyset$ *and penalty parameter* α *satisfies*

$$\alpha \geqslant \frac{1}{\min_{1\leqslant i\leqslant n+1,\lambda_i\neq 0}\lambda_i(C^{\mathrm{T}}C)}, \qquad (15)$$

with $\lambda_i(\cdot)$ *denoting the ith eigenvalue, then* $Y_{\mathrm{s}} \simeq Y$.

Proof. With E_i denoting the ith row of matrix E and defining the ith row of matrix $\bar{E}(y^*)$ as

$$\bar{E}_i(y^*) = \begin{cases} 0 & \text{if } E_iy^*\leqslant 0,\\ E_i & \text{if } E_iy^*>0.\end{cases}$$

The equilibrium equation of set Y becomes

$$\{\alpha C^{\mathrm{T}}C + E^{\mathrm{T}}\bar{E}(y^*) + (y^{*\mathrm{T}}y^*-1)I\}y^* = 0. \qquad (16)$$

It follows from their definitions that $E^{\mathrm{T}}\bar{E}(y^*) = \bar{E}^{\mathrm{T}}(y^*)\bar{E}(y^*)\geqslant 0$ and

$$C^{\mathrm{T}}C = \begin{bmatrix} J^{\mathrm{T}}J & 0\\ 0 & 0\end{bmatrix} \geqslant 0.$$

In light of Jacobian matrix J being nonzero, we have

$$\max_{1\leqslant i\leqslant n+1}\lambda_i(C^{\mathrm{T}}C) = \max_{1\leqslant i\leqslant n}\lambda_i(J^{\mathrm{T}}J)>0,$$
$$\min_{1\leqslant i\leqslant n+1}\lambda_i(C^{\mathrm{T}}C) = 0.$$

Thus, if selecting α as in (15), we know that $\alpha C^{\mathrm{T}}C + E^{\mathrm{T}}\bar{E}(y^*) + (y^{*\mathrm{T}}y^*-1)I$ of (16) cannot be zero, since some of its eigenvalues are much greater than 0 in terms of $E^{\mathrm{T}}\bar{E}(y^*)\geqslant 0$, $y^{*\mathrm{T}}y^*-1\geqslant -1$, and

$$\lambda_{i_1}(\alpha C^{\mathrm{T}}C) \geqslant \frac{\lambda_{i_1}(C^{\mathrm{T}}C)}{\min_{1\leqslant i\leqslant n+1,\lambda_i\neq 0}\lambda_i(C^{\mathrm{T}}C)} \geqslant 1,$$
$$\forall i_1 \in \{i_1 | \lambda_{i_1}(C^{\mathrm{T}}C)\neq 0\},$$
$$\lambda_{i_2}(\alpha C^{\mathrm{T}}C) = \frac{\lambda_{i_2}(C^{\mathrm{T}}C)}{\min_{1\leqslant i\leqslant n+1,\lambda_i\neq 0}\lambda_i(C^{\mathrm{T}}C)} = 0,$$
$$\forall i_2 \in \{i_2 | \lambda_{i_2}(C^{\mathrm{T}}C) = 0\}.$$

Moreover, assuming Y_{s} nonempty, we know that $y^*=0$ is a saddle point. This is because there exist eigenvalues of $\{\alpha C^{\mathrm{T}}C + E^{\mathrm{T}}\bar{E}(y^*) + (y^{*\mathrm{T}}y^*-1)I\}|_{y^*=0}$ equal to -1 corresponding to $\lambda_{i_2}(C^{\mathrm{T}}C)=0$, and subsequently $\dot{y}_i = 2\gamma y_i$. That is, if any element of $y(0)$ is nonzero, then the network state trajectory $y(t)$ will not converge to $y^*=0$. Thus, we could conclude that with α selected in (15), the set Y contains the subset Y_{s} mainly, and that there is almost no attractive equilibrium y^* outside Y_{s}; i.e., $Y_{\mathrm{s}} \simeq Y$. □

Propositions 2 and 3 solve a set of nonlinear inequalities in (9) by using a primal NN approach. No auxiliary neurons are entailed. The robotic task of online non-uniqueness-checking is thus performed by monitoring the NN output, $U(y)$, in real time. If $U(y)=0$ at time instant t, then a nonuniqueness point exists in such a minimum-effort resolution.

Remark 4. The nonuniqueness-checking criterion (i.e., Proposition 1) and online solution (i.e., Propositions 2 and 3) are designed for minimum-infinity-norm/minimum-effort redundancy resolution. This approach might be extendable to other research topics on the relationship between nonuniqueness and discontinuities. For example, as in the minimum-energy/minimum-two-norm redundancy resolution, there could also exist the problem of nonuniqueness and discontinuities. Such a problem might arise due to different reasons, such as, the Euler-angle based parameterization as mentioned by a reviewer, the situation of no feasible solutions at some points as experienced in special numerical experiments. For different situation, we may have to design different criterion and solution to check and compensate the condition of nonuniqueness and discontinuities. The above could be a future research direction by generalizing the LP-based online-checking approach from minimum-effort solutions to minimum-energy solutions.

4. Computer simulations

Theoretical results about nonuniqueness-checking (9) and primal NN (13) depicted in the previous section are substantiated by the following computer simulations. The first two examples are the static ones of solving (11) with constant coefficients E and C. The remaining three examples are the robotic application of primal NN (13) to a four-link planar robot [6], a 6DOF PUMA560 robot arm [25,28], and a 7DOF PA10 industrial manipulator [31].

Example 1. Consider the simplified model of LP (5)–(8) with $n=2, m=1, x^+=-x^-=[1,1,1]^{\mathrm{T}}, J:=[j_{11}\ j_{12}]$ and d. Different values of j_{11}, j_{12} and d correspond to different cases of the LP solutions. For example, case (i) if $j_{11}=j_{12}=d=1$, the solution, $x^*=[0.5,0.5,0.5]^{\mathrm{T}}$, is unique, and the resultant matrix E is

$$E=\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & -1\\ 0 & 1 & -1\end{bmatrix}.$$

Starting from any randomly generated initial condition for hundreds of times, the primal NN always converges to the equilibrium y^* with $U(y^*)>0$ [e.g., $U(y^*)=0.1777$]. Case (ii) if $j_{11}=0$ and $j_{12}=d=1$, there are multiple solutions to this LP problem, and one of the multiple solutions is $[0,1,1]^{\mathrm{T}}$ with the resultant matrix E being

$$E=\begin{bmatrix}0 & 0 & 1\\ 0 & 1 & -1\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}.$$

Starting from randomly-generated initial conditions for hundreds of times, corresponding to case (ii), the primal NN always converges to the equilibrium y^* with $U(y^*)=0$.

Example 2. Here, we consider solving more general nonlinear equations and inequality via the proposed primal NN (13). That is, coefficients E and C in (11) are randomly generated and no longer use the specific form as in Lemma 1 and Proposition 1. In the circumstances of no solution existing for (11), the left plot of Fig. 2 shows that starting from different initial conditions, $U(y)$ always converges to $0.068\neq 0$, while the right plot of Fig. 2 shows a typical transient of y convergence of network (13). Under the solution existence of (11), the left plot of Fig. 3 shows that starting from different initial conditions, $U(y)$ always converges to zero, and the right plot of Fig. 3 shows a

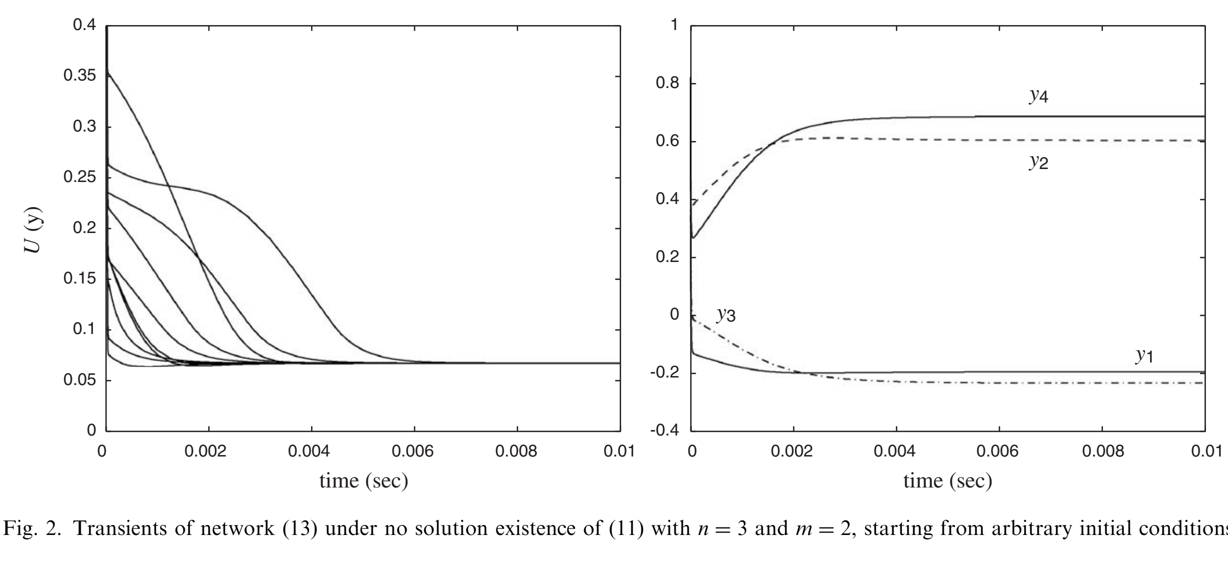

Fig. 2. Transients of network (13) under no solution existence of (11) with $n=3$ and $m=2$, starting from arbitrary initial conditions.

typical transient of y convergence of network (13). It is worth mentioning that all the tests are based on random initial conditions, and that for the same value of m, the solution existence of (11) is more frequently encountered as dimension n increases.

After showing the solution procedure of the above static equation-solving problems, we proceed to the application of criterion (9) and network (13) to the robotic research.

Example 3. Computer simulations are first performed based on the four-link planar robot [6]. As shown in Fig. 4, the manipulator end-point is desired to move along a circular path initiated from the joint state, $\theta(0) = [3\pi/4, -\pi/2, \pi/4, 0]^T$ rad. The task duration $T = 10$ s. By solving the minimum-effort inverse-kinematic problem, we have Fig. 5 showing the relationship between the discontinuity and nonuniqueness of the minimum-effort solution. They coincide quite well in this example ($m = 2$ and $n = 4$). As shown in Fig. 5(a), the second–fourth joint velocities have discontinuities during the time periods [3.0, 6.1] and [6.6, 9.75] in seconds. As shown in Fig. 5(b), there are extra nonuniqueness time-periods near the aforementioned ones like [2.7, 3.0], [6.1, 6.25] and [9.75, 9.97] in seconds where $U(y) = 0$. This confirms the results in [6] very well that the nonuniqueness of solution x^* to (5)–(8) is a necessary condition to the discontinuity phenomenon of minimum-effort solutions.

Example 4. Computer simulations of using primal NN (13) are then performed based on the 6DOF Unimation PUMA560 robot arm [25,28] where $n = 6$ and $m = 3$. As shown in Fig. 6, the initial joint state is $\theta(0) = [0, 0, 0, 0, 0, 0]^T$ rad, and the desired end-effector Cartesian path is a straight-line trajectory of length $0.2\sqrt{2}$ m with task duration $T = 5$ s. Fig. 7(a) shows that there is only one

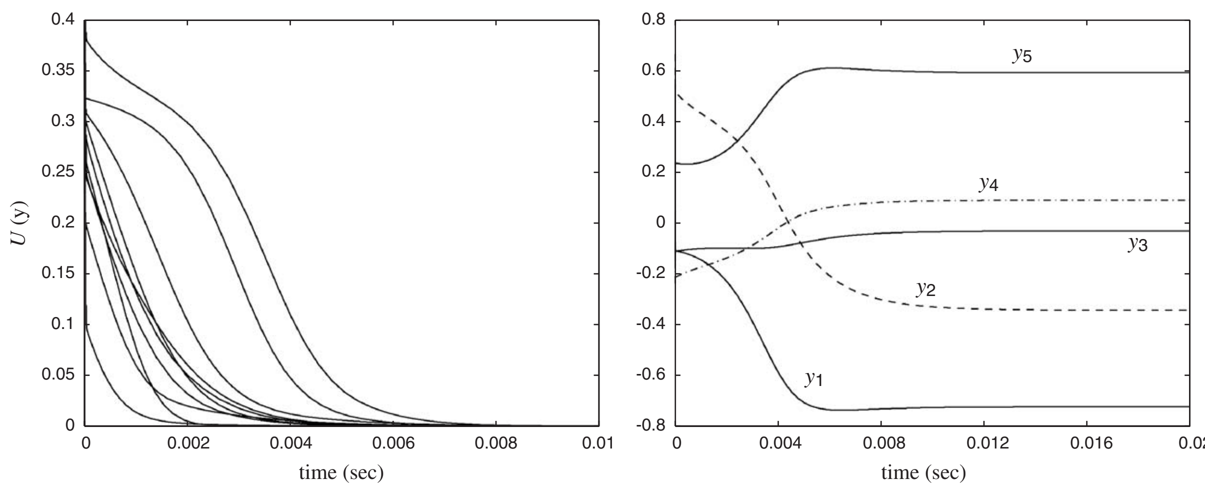

Fig. 3. Transients of network (13) under the solution existence of (11) with $n = 4$ and $m = 2$, starting from arbitrary initial conditions.

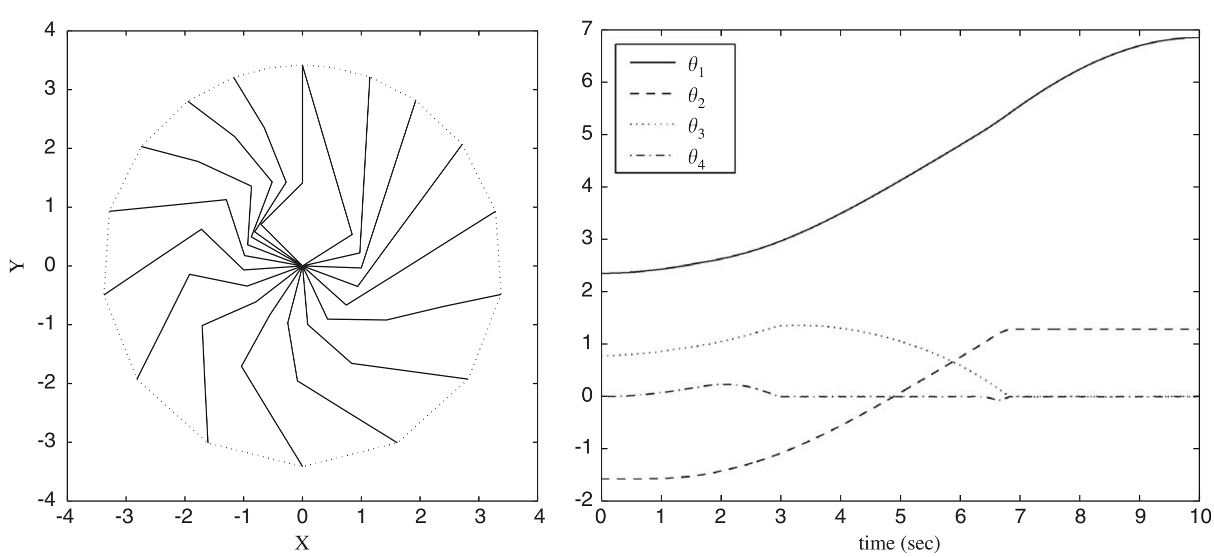

Fig. 4. Motion trajectories of a four-link planar robot synthesized by minimum-effort redundancy resolution.

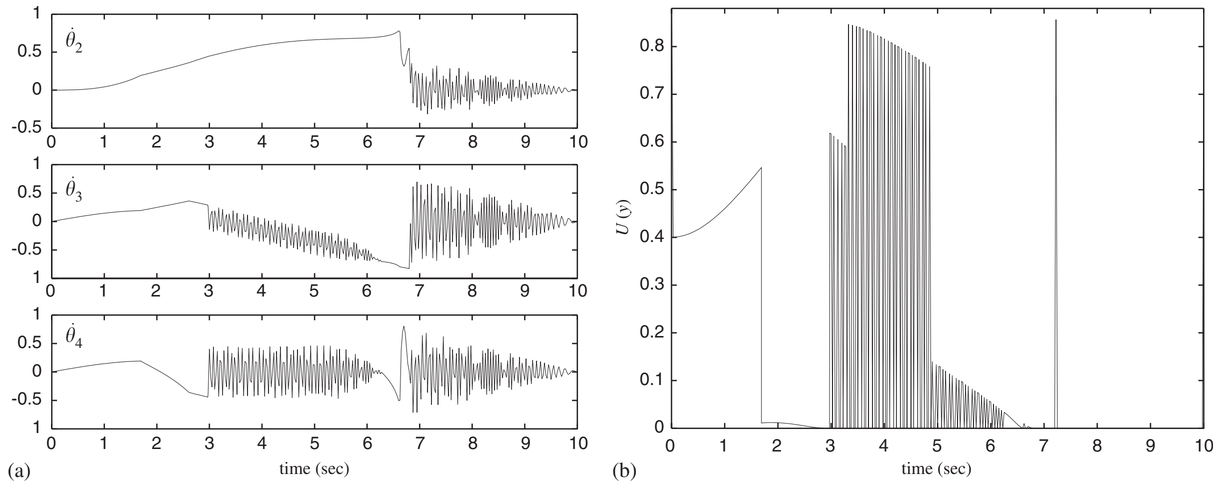

Fig. 5. Relationship between the discontinuity and nonuniqueness of a minimum-effort solution of a four-link planar robot: (a) discontinuities and (b) nonuniqueness.

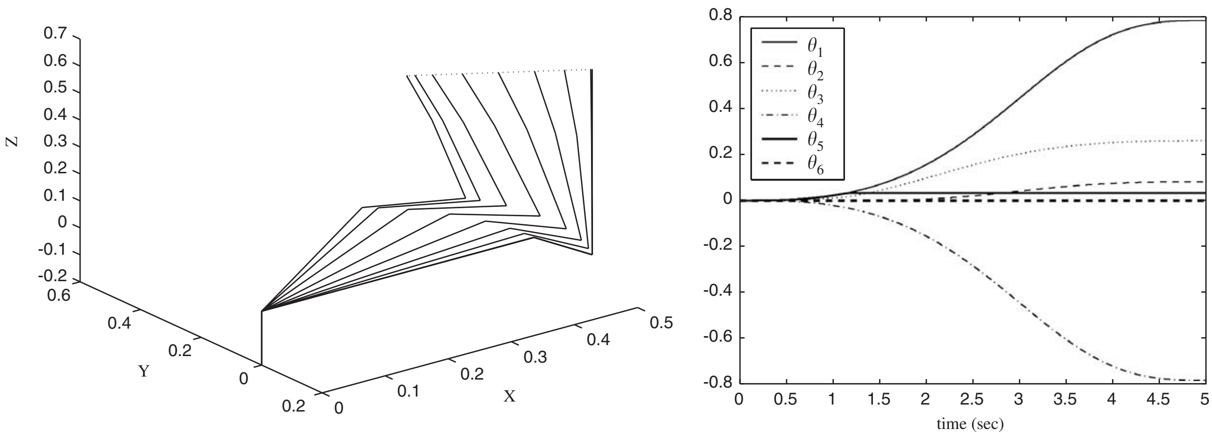

Fig. 6. Motion trajectories of the PUMA560 robot arm synthesized by minimum-effort redundancy resolution.

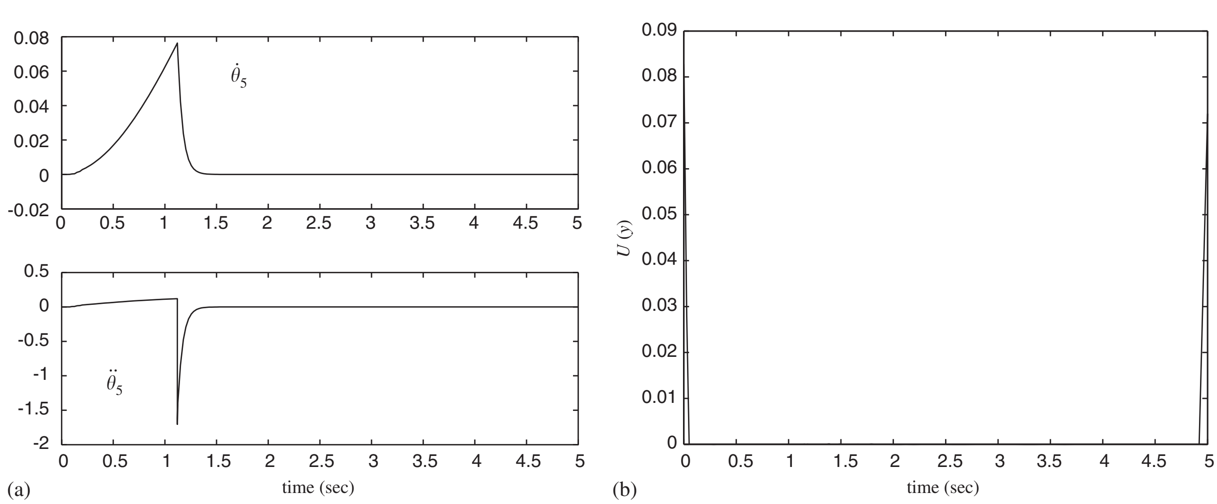

Fig. 7. Relationship between the discontinuity and nonuniqueness of a minimum-effort solution of the PUMA560 robot arm: (a) discontinuity and (b) nonuniqueness.

discontinuity point that exists in the fifth joint at time instant $t = 1.1187\,\mathrm{s}$. In contrast, as shown in Fig. 7(b), during almost all the task duration, the minimum-effort solution is not unique in view of $U(y)$ being 0. This implies that the nonuniqueness is only a necessary condition to the discontinuity phenomenon of minimum-effort redundancy resolution.

Example 5. The 7DOF Mitsubishi PA10 robot arm is finally simulated [31]. The initial state of the robot arm is $\theta(0) = [0, -\pi/4, 0, \pi/2, 0, -\pi/4, 0]^{\mathrm{T}}$ rad and the desired end-effector Cartesian path is a circular trajectory. The simulation is shown in Fig. 8. The comparison between time periods of nonuniqeness and discontinuity is given in Fig. 9. In this PA10 manipulator ($n = 7$ and $m = 3$), corresponding to the discontinuity time periods, the additional nonuniqueness time periods are [3.5, 4.7], [4.8, 6.75] and [6.95, 9.5] s, covering almost 90% of the task duration. This example, together with Example 4, substantiates that the nonuniqueness points of minimum-effort solutions usually exist in high-DOF manipulators, which may or may not cause the solution discontinuities. The following analysis might explain why nonuniqueness time periods appear more frequently over the increase of n. Since C is usually of rank m and y is of dimension $n+1$, the only zero solution of $Ey \leqslant 0$ and $Cy = 0$ lies in

$$E \to \begin{bmatrix} F \\ -F \\ \tilde{E} \end{bmatrix}, \quad F \in R^{(n+1-m)\times(n+1)}, \quad \operatorname{rank}\left(\begin{bmatrix} C \\ F \end{bmatrix}\right) = n+1.$$

That is, to find matrix F of rank $n+1-m$ from E via rows swap, of which the row vectors together with C rows span the R^{n+1} space. As E contains $4n+3$ rows at most, simply saying, it is empirically more probable to find $n+1-m$ basis rows from $4n+3$ rows as the manipulator DOF, n, increases (also refer to Example 2).

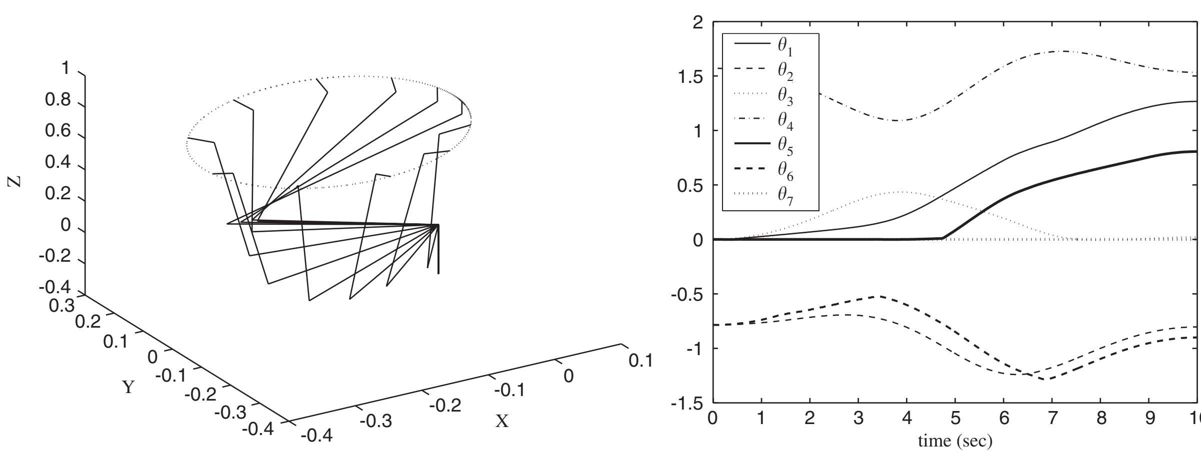

Fig. 8. Motion trajectories of the PA10 robot arm synthesized by minimum-effort redundancy resolution.

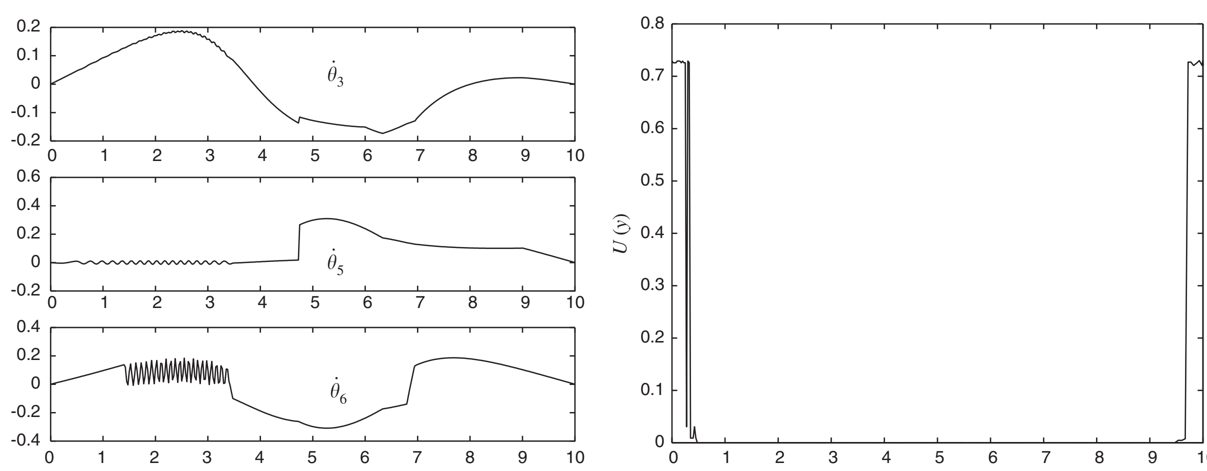

Fig. 9. Relationship between the discontinuity and nonuniqueness of a minimum-effort solution of the PA10 industrial manipulator: (a) discontinuities and (b) nonuniqueness.

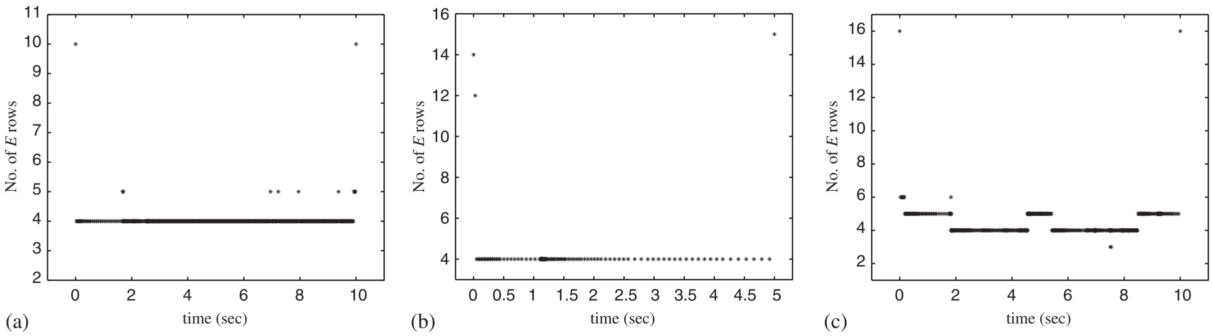

Fig. 10. The numbers of E rows are time-varying in the robotic context: (a) planar robot, (b) PUMA560 and (c) PA10.

Remark 5. The above observations generalize the robotic research results [6] to the high-DOF situation. It follows from our numerical experiments and analysis that, nonuniqueness is a necessary (not sufficient) condition to the discontinuities of minimum-effort inverse-kinematic solutions, and that when the minimum-effort redundancy resolution jumps from one solution set to another, the discontinuity may happen. This could also refer to Fig. 1 and the explaining paragraph in Section 2.1. As shown in Example 3, the nonuniqueness is nearly sufficient for manipulators with low degree-of-redundancy, thus able to remedy discontinuities effectively in [6,31]. For manipulators with high degree-of-redundancy, other weighting schemes like [5] might be used.

Remark 6. Moreover, as shown in Fig. 10, the numbers of rows of matrix E are time-varying: sometimes $4-6$ and sometimes more than 10. If nonprimal NNs are used in this context, $4n+3$ auxiliary neurons are required for handling E. In comparison, only $n+1$ neurons are used in the proposed primal NN approach. In addition to the computational efficiency, this shows another advantage of using primal NNs, especially in the situation of dynamic constraints and inequalities.

5. Concluding remarks

Minimum-effort solutions could complement the research on redundancy resolution in terms of low individual magnitude, even distribution of workload, and analyzing motion diversity. In this paper, for investigating the problem of discontinuities/nonuniqueness appearing in minimum-effort solutions, a set of nonlinear equations and inequalities has been formulated as a nonuniqueness criterion. Due to the real-time computation requirement, a primal NN has been developed to solve online this minimum-effort nonuniqueness problem. Numerical examples have substantiated the effectiveness and advantages of the proposed primal-NN computing scheme.

The observations drawn from Examples 3–5 have provided new insights and answers to the related solution-discontinuity problems [3,6,9,16,31]. For example, in minimum-effort redundancy resolution, the relationship between nonuniqueness and discontinuities could be summarized as follows. In the manipulators with low DOF/redundancy, the nonuniqueness could be a necessary and nearly sufficient condition to the appearance of discontinuities. In the manipulators with high DOF/redundancy, nonuniqueness could only be a necessary condition to the discontinuities.

Acknowledgements

The author would like to thank the editors and the anonymous reviewers for their time and effort in providing many constructive comments. Here, the gratitude is expressed to them, especially considering that I was making revisions around Thanksgiving day.

Appendix A

Proof of Lemma 3. With E_i denoting the ith row of matrix E, we have

$$
\begin{aligned}
&\|\max(Ey-f,0)\|_2^2\\
&\quad=\max(Ey-f,0)^{\mathrm{T}}\max(Ey-f,0)\\
&\quad=\begin{bmatrix}\max(E_1y-f_1,0)\\ \max(E_2y-f_2,0)\\ \cdots\\ \max(E_iy-f_i,0)\\ \cdots\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\max(E_1y-f_1,0)\\ \max(E_2y-f_2,0)\\ \cdots\\ \max(E_iy-f_i,0)\\ \cdots\end{bmatrix}\\
&\quad=\max(E_1y-f_1,0)^2+\max(E_2y-f_2,0)^2\\
&\qquad+\cdots+\max(E_iy-f_i,0)^2+\cdots\\
&\quad=\sum_{i=1}^{\dim(E)}\max(E_iy-f_i,0)^2. \qquad (17)
\end{aligned}
$$

For y_j (i.e., the jth element of y), if $E_i y - f_i > 0$, we have

$$\begin{aligned}\frac{\partial \max(E_i y - f_i, 0)^2}{\partial y_j} &= \frac{\partial (E_i y - f_i)^2}{\partial y_j} \\ &= 2(E_i y - f_i)\frac{\partial (E_i y - f_i)}{\partial y_j} \\ &= 2(E_i y - f_i)E_{ij} \\ &= 2\max(E_i y - f_i, 0)E_{ij}. \end{aligned} \tag{18}$$

The above result is also achievable in the case of $E_i y - f_i \leqslant 0$:

$$\frac{\partial \max(E_i y - f_i, 0)^2}{\partial y_j} = \frac{\partial 0}{\partial y_j} = 0 = 2\max(E_i y - f_i, 0)E_{ij}, \tag{19}$$

which is in view of $\max(E_i y - f_i, 0) = 0$ as $E_i y - f_i \leqslant 0$. Thus, no matter whether $E_i y - f_i > 0$ or not, it follows from (17), (18) and (19) that

$$\begin{aligned}\frac{\partial \|\max(Ey - f, 0)\|_2^2}{\partial y_j} &= \frac{\partial \sum \max(E_i y - f_i, 0)^2}{\partial y_j} \\ &= 2\sum_{i=1}^{\dim(E)} \max(E_i y - f_i, 0)E_{ij}.\end{aligned}$$

Now, we have the derivative of $\|\max(Ey - f, 0)\|_2^2$ with respect to y as

$$\begin{aligned}&\frac{\mathrm{d}\|\max(Ey - f, 0)\|_2^2}{\mathrm{d}y} \\ &= \begin{bmatrix} \partial\|\max(Ey - f, 0)\|_2^2/\partial y_1 \\ \partial\|\max(Ey - f, 0)\|_2^2/\partial y_2 \\ \cdots \\ \partial\|\max(Ey - f, 0)\|_2^2/\partial y_j \\ \cdots \end{bmatrix} \\ &= \begin{bmatrix} 2\sum_{i=1}^{\dim(E)} \max(E_i y - f_i, 0)E_{i1} \\ 2\sum_{i=1}^{\dim(E)} \max(E_i y - f_i, 0)E_{i2} \\ \cdots \\ 2\sum_{i=1}^{\dim(E)} \max(E_i y - f_i, 0)E_{ij} \\ \cdots \end{bmatrix},\end{aligned}$$

which equals the following matrix form

$$\begin{aligned}&2E^{\mathrm{T}}\max(Ey - f, 0) \\ &= 2\begin{bmatrix} E_{11} & E_{21} & \cdots & E_{i1} & \cdots \\ E_{12} & E_{22} & \cdots & E_{i2} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ E_{1j} & E_{2j} & \cdots & E_{ij} & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} \max(E_1 y - f_1, 0) \\ \max(E_2 y - f_2, 0) \\ \cdots \\ \max(E_i y - f_i, 0) \\ \cdots \end{bmatrix}.\end{aligned}$$

The proof is thus completed. □

Appendix B

Proof of Lemma 4. For y_j (the jth element of y), we could derive that

$$\frac{\partial (y^{\mathrm{T}}y - h)^2/2}{\partial y_j} = (y^{\mathrm{T}}y - h)\frac{\partial (y^{\mathrm{T}}y - h)}{\partial y_j} = (y^{\mathrm{T}}y - h)2y_j.$$

According to the definition of the derivative of a scalar function with respect to a vector, we thus have

$$\begin{aligned}\frac{\mathrm{d}(y^{\mathrm{T}}y - h)^2/2}{\mathrm{d}y} &= \begin{bmatrix} \frac{\partial (y^{\mathrm{T}}y - h)^2/2}{\partial y_1} \\ \frac{\partial (y^{\mathrm{T}}y - h)^2/2}{\partial y_2} \\ \cdots \\ \frac{\partial (y^{\mathrm{T}}y - h)^2/2}{\partial y_j} \\ \cdots \end{bmatrix} = \begin{bmatrix} (y^{\mathrm{T}}y - h)2y_1 \\ (y^{\mathrm{T}}y - h)2y_2 \\ \cdots \\ (y^{\mathrm{T}}y - h)2y_j \\ \cdots \end{bmatrix} \\ &= 2(y^{\mathrm{T}}y - h)y,\end{aligned}$$

which completes the proof. □

References

[1] N.M. Botros, M. Abdul-Aziz, Hardware implementation of an artificial neural network using field programmable gate arrays (FPGAs), IEEE Trans. Ind. Electron. 41 (6) (1994) 665–667.
[2] A. Cichocki, R. Unbehauen, Neural networks for solving systems of linear equation and related problems, IEEE Trans. Circuits Syst. I 39 (2) (1992) 124–138.
[3] A.S. Deo, I.D. Walker, Minimum effort inverse kinematics for redundant manipulators, IEEE Trans. Robotics Autom. 13 (5) (1997) 767–775.
[4] C. Diorio, R.P.N. Rao, Neural circuits in silicon, Nature 405 (6789) (2000) 891–892.
[5] S.S. Ge, Y. Zhang, T.H. Lee, An acceleration-based weighting scheme for minimum-effort inverse kinematics of redundant manipulators, Proceedings of the IEEE International Symposium on Intelligent Control, 2004, pp. 275–280.
[6] I.A. Gravagne, I.D. Walker, On the structure of minimum effort solutions with application to kinematic redundancy resolution, IEEE Trans. Robotics Autom. 16 (6) (2000) 855–863.
[7] K. Iqbal, Y.C. Pai, Predicted region of stability for balance recovery: motion at the knee joint can improve termination of forward movement, J. Biomechanics 33 (12) (2000) 1619–1627.
[8] M.L. Latash, Control of Human Movement, Human Kinetics Publisher, Chicago, 1993.
[9] J. Lee, A structured algorithm for minimum l_∞-norm solutions and its application to a robot velocity workspace analysis, Robotica 19 (3) (2001) 343–352.
[10] X. Liang, S.K. Tso, An improved upper bound on step-size parameters of discrete-time recurrent neural networks for linear inequality and equation system, IEEE Trans. Circuits Syst. 49 (5) (2002) 695–698.
[11] F.L. Luo, B. Zheng, Neural network approach to computing matrix inversion, Appl. Math. Comput. 47 (2–3) (1992) 109–120.
[12] O.L. Mangasarian, Uniqueness of solution in linear programming, Linear Algebra Appl. 25 (1979) 151–162.
[13] C. Mead, Analog VLSI and Neural Systems, Addison-Wesley, Reading, MA, 1989.
[14] R. Saigal, Linear Programming: a Modern Integrated Analysis, Kluwer Academic Publishers, Norwell, MA, 1995.

[15] L. Sciavicco, B. Siciliano, Modelling and Control of Robot Manipulators, Springer-Verlag, London, 2000.

[16] I.-C. Shim, Y.-S. Yoon, Stabilized minimum infinity-norm torque for redundant manipulators, Robotica 16 (1998) 193–205.

[17] J.-J.E. Slotine, W. Li, Applied Nonlinear Control, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[18] J. Song, Y. Yam, Complex recurrent neural network for computing the inverse and pseudo-inverse of the complex matrix, Appl. Math. Comput. 93 (2–3) (1998) 195–205.

[19] D.W. Tank, J.J. Hopfield, Simple neural optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit, IEEE Trans. Circuits Syst. 33 (1986) 533–541.

[20] J. Wang, Y. Zhang, Recurrent neural networks for real-time computation of inverse kinematics of redundant manipulators, in: Machine Intelligence: Quo Vadis?, World Scientific, Singapore, 2004.

[21] L.-X. Wang, J.M. Mendel, Three-dimensional structured network for matrix equation solving, IEEE Trans. Comput. 40 (12) (1991) 1337–1345.

[22] S. Zhang, A.G. Constantinides, Lagrange programming neural networks, IEEE Trans. Circuits Syst. 39 (7) (1992) 441–452.

[23] X. Zhang, D.B. Chaffin, An inter-segment allocation strategy for postural control in human reach motions revealed by differential inverse kinematics and optimization, Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, 1997, pp. 469–474.

[25] Y. Zhang, S.S. Ge, 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. Part B 34 (5) (2004) 2126–2132.

[26] Y. Zhang, D. Jiang, J. Wang, A recurrent neural network for solving Sylvester equation with time-varying coefficients, IEEE Trans. Neural Networks 13 (5) (2002) 1053–1063.

[27] Y. Zhang, J. Wang, Global exponential stability of recurrent neural networks for synthesizing linear feedback control systems via pole assignment, IEEE Trans. Neural Networks 13 (3) (2002) 633–644.

[28] Y. Zhang, J. Wang, A dual neural network for constrained joint torque optimization of kinematically redundant manipulators, IEEE Trans. Syst. Man Cybern. 32 (5) (2002) 654–662.

[29] Y. Zhang, J. Wang, Obstacle avoidance for kinematically redundant manipulators using a dual neural network, IEEE Trans. Syst. Man Cybern. Part B 34 (1) (2004) 752–759.

[30] Y. Zhang, J. Wang, Y. Xia, A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits, IEEE Trans. Neural Networks 14 (3) (2003) 658–667.

[31] Y. Zhang, J. Wang, Y. Xu, A dual neural network for bi-criteria kinematic control of redundant manipulators, IEEE Trans. Robotics Autom. 18 (6) (2002) 923–931.

Yunong Zhang was born in Xinyang, Henan, PR China in October 1973. He received the B.E. degree from the Huazhong University of Science and Technology (HUST) in 1996, the M.E. degree from the South China University of Technology (SCUT) in 1999. He completed his Ph.D. study in the Chinese University of Hong Kong (CUHK) in November 2002 with the degree received in 2003. Then, as a research fellow he had been with the National University of Singapore (NUS), and the University of Strathclyde, United Kingdom, respectively, in 2003 and 2004. After that, he has been with the National University of Ireland, Maynooth (NUIM) as a research scientist. His current research interests are redundant robot manipulators and the related biomechanics research, recurrent neural networks and their hardware/circuits implementation, and scientific computing and optimization such as Gaussian process regression.