Proceedings of the 2007 IEEE
International Conference on Mechatronics and Automation
August 5 - 8, 2007, Harbin, China

Minimum-Energy Redundancy Resolution of Robot Manipulators Unified by Quadratic Programming and its Online Solution*

Yunong Zhang
Dept of Electronics and Communication Engineering
School of Information Science and Technology
Sun Yat-Sen University, Guangzhou, 510275 China
http://www.ee.sysu.edu.cn/teacher/detail.asp?sn=129
ynzhang@ieee.org

Shugen Ma[1,2]
[1]*Organization for Promotion of the COE Program,*
Ritsumeikan University 1-1-1 Nojihigashi,
Kusatsu-Shi, 525-8577 Japan
[2]*Robotics Laboratory, Shenyang Institute of Automation,*
CAS 114 Nanta Street, Shenyang, 110015 China

***Abstract*— This paper presents the latest result regarding the unification of minimum-energy redundancy resolution of robot manipulators via a quadratic program. The presented quadratic programming (QP) formulation is general in the sense that it incorporates equality, inequality and bound constraints, simultaneously. This QP formulation covers the online avoidance of joint physical limits and environmental obstacles, as well as the optimization of various performance indices. Every term is endowed with clear physical meaning and utility. Motivated by the real-time solution to such robotic problems, four QP online solvers are briefly reviewed. That is, standard QP optimization routines, compact QP method, dual neural network as a QP solver, and state-of-the-art LVI-based primal-dual neural network as a QP solver. The QP-based unification of robots' redundancy resolution is substantiated by a large number of computer simulation results based on PUMA560, PA10, and planar robot arms.**

Index Terms*— *Redundant manipulators, joint physical limits, obstacle avoidance, quadratic programming, PUMA560 robot.

I. Introduction

A redundant manipulator is defined when more degrees-of-freedom are available than the minimum number of DOF required to execute a given end-effector primary task [1][2]. Our human arm, elephant trunk and snake are also such redundant systems [3][4]. Compared to non-redundant manipulators, the redundant manipulator naturally has wider operational space and extra degrees to meet more functional constraints, such as, the online avoidance of joint physical limits [5] and environmental obstacles [6][7].

One of the most fundamental issues in operating the redundant manipulators is the redundancy-resolution problem. That is, given the Cartesian velocity/acceleration trajectories of the end-effector, we are required to generate the corresponding joint velocity, acceleration and/or torque trajectories in real time [1]. As the fundamental topic has given birth to a multitude of different redundancy-resolution schemes (see [1][2][5]-[12] and references therein), this paper shows that such different schemes could be unified into a general QP formulation simultaneously subject to equality, inequality and bound constraints. As those redundancy-resolution schemes either belong to or relate to the minimum energy or two-norm situation, the title of this paper is thus given as "minimum-energy $\cdots$". The unification of these redundancy-resolution schemes may bring more insights into the wealth of existing solutions as well as a better understanding of future researches.

*This research is funded by National Science Foundation of China (60643004) and by the Science and Technology Office of Sun Yat-Sen University. One of the authors, Yunong Zhang, was supported by various research fellowships/assistantship, when he was with National University of Ireland, University of Strathclyde, NUS, and CUHK.

II. Robotic Formulation

The robotic issue of our interest here is that, given the trajectory $r(t) \in R^m$ of the end-effector, we want to generate online the joint trajectory $\theta(t) \in R^n$ so as to command the manipulator motion. However, the redundancy (i.e., $n-m$ degrees of freedom) is generally resolved at the velocity level $\dot\theta(t) \in R^n$ or acceleration level $\ddot\theta(t) \in R^n$, due to the nonlinearity and redundancy of the forward-kinematic function from $\theta(t)$ to $r(t)$ [1][2].

The relationship between the end-effector velocity $\dot r(t)$ and the joint velocity $\dot\theta(t)$ via Jacobian matrix $J(\theta) \in R^{m\times n}$ can be represented as

$$\dot r = J(\theta)\dot\theta. \tag{1}$$

Differentiating (1) yields the relation between joint acceleration $\ddot\theta(t)$ and end-effector acceleration $\ddot r(t)$:

$$J(\theta)\ddot\theta = \ddot r_a \tag{2}$$

where $\ddot r_a := \ddot r - \dot J(\theta)\dot\theta \in R^m$, and $\dot J(\theta)$ is the time derivative of $J(\theta)$. Because the manipulator system is redundant, $m < n$. Equations (1) and (2) are thus under-determined, admitting an infinite number of solutions. The conventional solutions to equations (1) and (2) [i.e., to solve for $\dot\theta(t)$ and $\ddot\theta(t)$, respectively] were the pseudoinverse/nullspace-type solution

[1][2]. The research of recent ten years [1][8][12] shows that the redundancy-resolution problem could be solved in a more favorable manner via online optimization techniques.

For example, we can start with the following pure robotic problem formulation for manipulators' redundancy resolution at the joint-velocity level:

$$\min \quad \phi(\theta,\dot\theta) \tag{3}$$
$$\text{s.t.} \quad J\dot\theta = \dot r \tag{4}$$
$$A\dot\theta \leqslant b \tag{5}$$
$$\theta^- \leqslant \theta \leqslant \theta^+ \tag{6}$$
$$\dot\theta^- \leqslant \dot\theta \leqslant \dot\theta^+ \tag{7}$$

where

- equation (4) is exactly equation (1) used to perform the given end-effector primary tasks;
- joint physical limits $[\theta^-,\theta^+]$ and $[\dot\theta^-,\dot\theta^+]$ are to be avoided via (6) and (7) as examined in Section III [5][9];
- inequality constraint (5) is entailed for obstacle avoidance that will be examined in Section IV [6][11]; and,
- $\phi(\theta,\dot\theta)\in R$ is any suitable performance index that will be examined in Section V [1].

Similar to the problem formulation in (3)-(7), if the redundancy is to be resolved at the joint-acceleration level, we have the following robotic problem formulation:

$$\min \quad \phi(\theta,\dot\theta,\ddot\theta) \tag{8}$$
$$\text{s.t.} \quad H(\theta)\ddot\theta + c(\theta,\dot\theta) + g(\theta) = \tau \tag{9}$$
$$J\ddot\theta = \ddot r_a \tag{10}$$
$$A\ddot\theta \leqslant b \tag{11}$$
$$\theta^- \leqslant \theta \leqslant \theta^+ \tag{12}$$
$$\dot\theta^- \leqslant \dot\theta \leqslant \dot\theta^+ \tag{13}$$
$$\ddot\theta^- \leqslant \ddot\theta \leqslant \ddot\theta^+ \tag{14}$$

where

- equation (10) is exactly equation (2) used to perform the given end-effector primary tasks;
- joint physical limits $[\theta^-,\theta^+]$, $[\dot\theta^-,\dot\theta^+]$ and $[\ddot\theta^-,\ddot\theta^+]$ are to be avoided via (12)-(14) [5][9];
- inequality constraint (11) plays the same role as (5) for obstacle avoidance [6][11];
- $\phi(\theta,\dot\theta,\ddot\theta)$ is a performance index similar to (3) [1]; and
- (9) is the manipulator's dynamic equation [2][13] with $H(\theta)\in R^{n\times n}$ denoting the positive-definite inertia matrix, $c(\theta,\dot\theta)\in R^n$ denoting the Coriolis/centrifugal force vector, $g(\theta)\in R^n$ denoting the gravitational force vector, and $\tau\in R^n$ denoting the joint-torque vector.

III. Handling Joint Physical Limits

A. Velocity Level

For velocity-level redundancy resolution, the performance index and constraints in (3)-(7) all have to be converted to the expressions based on joint velocity $\dot\theta$ [5]. For example, the avoidance of joint limits $[\theta^-,\theta^+]$ in (6) could be reformulated as follows:

$$\kappa_p(\theta^- - \theta) \leqslant \dot\theta \leqslant \kappa_p(\theta^+ - \theta) \tag{15}$$

where $\kappa_p > 0$, termed intensity coefficient, is used to scale the feasible region of $\dot\theta$ and determine the magnitude of a deceleration when a joint approaches its limit. The value of κ_p is chosen such that the feasible region of $\dot\theta$ made by joint limits conversion (15) is normally not smaller than the original one made by joint velocity limits [i.e., (7)]. Note that larger values of κ_p cause joint deceleration more quickly. In general, κ_p is selected to be greater than $\max_{1\leqslant i\leqslant n}\{(\dot\theta_i^+ - \dot\theta_i^-)/(\theta_i^+ - \theta_i^-)\}$, while in a large number of computer simulations based on the PUMA560, PA10 and other manipulators, we choose $\kappa_p = 20$. By combining (15), the velocity-level avoidance of joint physical limits [(6) and (7)] becomes

$$\xi^- \leqslant \dot\theta \leqslant \xi^+ \tag{16}$$

where $\xi^- := \max(\kappa_p(\theta^- - \theta),\ \dot\theta^-)$, and

$$\xi^+ := \min(\kappa_p(\theta^+ - \theta),\ \dot\theta^+).$$

B. Acceleration Level

For acceleration-level redundancy resolution, the performance index and constraints in (8)-(14) have to be converted to the expressions based on joint acceleration $\ddot\theta$ [9]. In light of the inertia movement, the avoidance of joint limits $[\theta^-,\theta^+]$ in (12) is converted as

$$\kappa_p(\eta\theta^- - \theta) \leqslant \dot\theta \leqslant \kappa_p(\eta\theta^+ - \theta) \tag{17}$$

where κ_p is defined the same as (15), and $\eta\in(0,1)$, termed critical-area coefficient, is used to define two critical areas (i.e., $[\theta^-,\eta\theta^-]$ and $[\eta\theta^+,\theta^+]$) such that there will appear a deceleration when a joint enters such areas. The avoidance of joint velocity limits $[\dot\theta^-,\dot\theta^+]$ in (13) can be similarly converted as an $\ddot\theta$-based expression:

$$\kappa_v(\dot\theta^- - \dot\theta) \leqslant \ddot\theta \leqslant \kappa_v(\dot\theta^+ - \dot\theta). \tag{18}$$

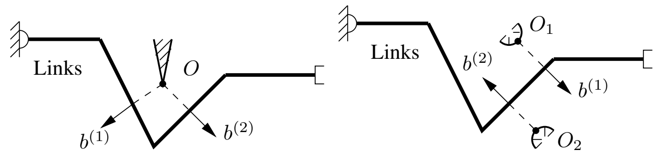
By using (17) and (18), the acceleration-level avoidance of joint physical limits (12)-(14) becomes $\xi^- \leqslant \ddot\theta \leqslant \xi^+$, where $\xi^- := \max(\kappa_p(\eta\theta^- - \theta),\ \kappa_v(\dot\theta^- - \dot\theta),\ \ddot\theta^-)$, and

$$\xi^+ := \min(\kappa_p(\eta\theta^+ - \theta),\ \kappa_v(\dot\theta^+ - \dot\theta),\ \ddot\theta^+).$$

In our computer simulations based on PUMA560, PA10 and other manipulators, $\kappa_p = \kappa_v = 20$ and $\eta = 0.9$.

IV. Avoiding Obstacles

Avoiding collision with obstacles is a basic and important requirement for operating robot manipulators working in a crowded environment [6][11]. Like joint physical limits, various obstacles do exist in almost any manipulator's operational space. For example, even if there is no external obstacles,



(a) over-determination (b) dilemma

Fig. 1. Contradicting situations in equality-based collision-free formulation.

collisions may still occur between manipulator links and the pedestal. If a robot encounters a collision and then fails there, the desired end-effector trajectory $r(t)$ becomes impossible to realize, not to mention the physical damage possibly caused.

As compared to distance-maximizing methods [16][17], some researches [18]-[20] treat the collision-free requirement as an equality constraint. That is, when a robot link $\mathcal{L}$ approaches an obstacle $\mathcal{O}$ within the influential range of radius ϵ, an escape velocity $b \in R^3$ is generated to push the link $\mathcal{L}$ to move away from the obstacle $\mathcal{O}$. This escape velocity was imposed in an equality-constraint form [18][20]; i.e., $A\dot{\theta} = b$ where $A \in R^{3\times m}$ is the critical-point Jacobian matrix at the link $\mathcal{L}$ corresponding to the obstacle point $\mathcal{O}$. The weakness of such an equality-based collision-free formulation is that it is always difficult (or sometimes impossible) to determine a suitable magnitude of escape velocity b. For example, see Fig. 1 for two contradicting situations [6][11].

In view of the above observation, an inequality-constraint based collision-free formulation is developed in [6][7][11] by only using the directions of escape velocities pointing from obstacles to links. Specifically-speaking, the design procedure is as follows.

i) After calculating the Cartesian coordinates of obstacle point O and the critical point C of a most vulnerable link, the escape-velocity direction could be denoted by $\overrightarrow{OC} \in R^3$.
ii) After calculating the Jacobian matrix $J_C \in R^{3\times n}$ at the critical point C of the link, the new matrix $A := -\mathrm{sgn}(\overrightarrow{OC}) \diamond J_C$ where $\diamond$ is a vector-matrix multiplication operator defined in [6][7][11].
iii) The inequality-based criterion is thus $A\dot{\theta} \leqslant 0$.
iv) To avoid suddenly imposing such a constraint, a smoothed version is used by modifying the critical-point speed gradually, in the form of $A\dot{\theta} \leqslant b$ [6][7][11].

The details and mathematical proof of such an inequality-constraint based formulation,

$$A\dot{\theta} \leqslant b, \tag{19}$$

are given in [6][7][11] respectively for point obstacles and window-shaped obstacle avoidance. Compared to distance-maximization and equality-based approaches, the inequality-based formulation (19) has the following advantages.

- Distance maximization as a performance index is only preferable, whereas constraint satisfaction such as (19) is more imperative.
- Equality constraints may unnecessarily reduce the solution space, whereas the inequality-based criterion (19) is proved to be able to generate a variable-magnitude escape velocity.
- Inequality-based criterion (19) is suitable for multiple obstacles avoidance with no gradient/derivative computation required, as compared to the distance-maximization method.

V. Various Performance Indices

After reformulating and interpreting the constraints in both velocity-level redundancy resolution (3)-(7) and acceleration-level redundancy resolution (8)-(14), we come to interpret and handle various typical performance indices $\phi(\theta, \dot{\theta}, \ddot{\theta})$ proposed so far in the literature.

A. Resolved at Velocity Level

<u>*A.1 The MVN scheme*</u>

The simplest and most effective performance index at the velocity-level redundancy resolution is the so-called minimum-velocity-norm (MVN) scheme; i.e.,

$$\phi(\dot{\theta}) := \|\dot{\theta}\|_2^2/2 = \dot{\theta}^T\dot{\theta}/2.$$

If we rewrite it in a quadratic-minimization form such as $\phi(\dot{\theta}) = \dot{\theta}^T W \dot{\theta}/2 + q^T\dot{\theta}$, then in the MVN redundancy-resolution scheme $W := I$ and $q := 0$.

<u>*A.2 The RMP scheme*</u>

A natural extension from the MVN scheme is the repetitive motion planning (RMP) by minimizing the joint displacements between current states and initial states [5][21]. That is, to minimize the following performance index $\phi(\theta, \dot{\theta})$ so that the robot could perform cyclic tasks:

$$\phi(\theta, \dot{\theta}) := (\dot{\theta} + z)^T(\dot{\theta} + z)/2 \text{ with } z = 4(\theta - \theta(0)).$$

If we rewrite it in a quadratic form such as $\phi(\dot{\theta}) = \dot{\theta}^T W \dot{\theta}/2 + q^T\dot{\theta}$, then in the RMP redundancy-resolution scheme $W := I$ and $q := z = 4(\theta - \theta(0))$.

<u>*A.3 The MKE scheme*</u>

Being a local counterpart of global kinetic-energy minimization [9][22], the following inertia-weighted performance index is defined and resolved at the velocity level:

$$\phi(\theta, \dot{\theta}) := \dot{\theta}^T H \dot{\theta}/2.$$

This performance index minimizes the instantaneous kinetic energy, and hence the resulting scheme is termed the minimum-kinetic-energy (MKE) redundancy resolution. If we rewrite it in a quadratic form such as $\phi(\dot{\theta}) = \dot{\theta}^T W \dot{\theta}/2 + q^T\dot{\theta}$, then in the MKE redundancy-resolution scheme $W := H(\theta)$ and $q := 0$.

B. Resolved at Acceleration Level

B.1 The MAN scheme

As discussed in Remark 2, acceleration-level redundancy resolution could accommodate more joint physical limits; and, with such physical limits considered, acceleration-level schemes could also be used for long trajectory movements. Based on a large number of computer simulations, the minimum-acceleration-norm (MAN) scheme is thought to be one of the most effective acceleration-level redundancy-resolution schemes for long trajectory movements [1][15]. The performance index is $\phi(\ddot{\theta}) := \ddot{\theta}^T\ddot{\theta}/2$ which defines $W = I$ and $q = 0$, if we rewrite it in the quadratic form $\phi(\ddot{\theta}) = \ddot{\theta}^T W\ddot{\theta}/2 + q^T\ddot{\theta}$.

B.2 The MTN scheme

A natural extension from the MAN scheme to the joint torque minimization is the minimum-torque-norm (MTN) redundancy-resolution scheme. That is, to minimize the following two norm of instantaneous joint torques for a better distribution of actuator power [1][8][13]:

$$\phi(\theta, \dot{\theta}, \ddot{\theta}) := \|\tau\|_2^2/2 = \tau^T\tau/2. \tag{20}$$

As the redundancy is to be resolved at the joint-acceleration level, the performance index in (20) is converted to an expression based on $\ddot{\theta}$ via manipulator dynamic equation (9). Simple matrix manipulations yield the following minimization [1], $\phi(\theta, \dot{\theta}, \ddot{\theta}) := \ddot{\theta}^T H^2\ddot{\theta}/2 + (c+g)^T H\ddot{\theta}$ which defines $W = H^2$ and $q = H(c+g)$, if we rewrite it in the quadratic form $\phi(\ddot{\theta}) = \ddot{\theta}^T W\ddot{\theta}/2 + q^T\ddot{\theta}$.

B.3 The IIWT scheme

Another acceleration-level redundancy-resolution scheme is the inertia-inverse weighted torque (IIWT or simply IWT) minimization by minimizing the joint torque weighted by inertia inverse [9][22]. As analyzed via calculus of variations, the IIWT scheme results in resolutions with global characteristics. The performance index to be minimized is

$$\phi(\theta, \dot{\theta}, \ddot{\theta}) := \tau^T H^{-1}\tau/2. \tag{21}$$

The redundancy is resolved at the joint-acceleration level (i.e., based on $\ddot{\theta}$). The above performance index is thus converted to the following one by using the manipulator dynamic equation (9) [9][13]:

$$\phi(\theta, \dot{\theta}, \ddot{\theta}) := \ddot{\theta}^T H\ddot{\theta}/2 + (c+g)^T\ddot{\theta}.$$

This defines $W = H$ and $q = (c+g)$ in the IIWT context, if we rewrite the performance index in the unified quadratic form $\phi(\ddot{\theta}) = \ddot{\theta}^T W\ddot{\theta}/2 + q^T\ddot{\theta}$.

VI. Unified QP Formulation

Following the above analysis procedures (especially from Sections III to V), we can see that both velocity-level redundancy resolution (3)-(7) and acceleration-level redundancy resolution (8)-(14) could be reformulated as a quadratic-programming problem.

Theorem. Consider the avoidance of joint physical limits and environmental obstacles. The velocity-level redundancy resolution (i.e., the MVN, RMP and MKE schemes) and acceleration-level redundancy resolution (i.e., the MAN, MTN and IIWT schemes) all could be rewritten in the following quadratic-programming form:

$$\begin{aligned} \text{minimize} \quad & x^T W x/2 + q^T x, & (22)\\ \text{subject to} \quad & Jx = d, & (23)\\ & Ax \leqslant b, & (24)\\ & \xi^- \leqslant x \leqslant \xi^+, & (25) \end{aligned}$$

where decision vector x is defined respectively as $\dot{\theta}$ in velocity-level schemes or $\ddot{\theta}$ in acceleration-level schemes. Coefficients W, q, A, b, d, and $\xi^{\pm}$ are defined accordingly for a specific redundancy-resolution scheme. □

VII. Online QP Solvers

In the previous sections, we have reformulated the physically-constrained redundancy-resolution problem into a time-varying quadratic program subject to hybrid kinds of constraints. Each term has interpretably physical meaning and utility. This reformulation extracts major mathematic problems from an originally very complex robotic context, making the redundancy-resolution task much clearer and easier to understand. To solve QP (22)-(25), the following four QP solvers are reviewed.

A. QP Routines

The first option without derivation could be the stanadrd QP optimization routines performed on digital computers. For example, among the MATLAB optimization routines [23], "QUADPROG" could be used with syntax being

$$x = \text{QUADPROG}(W, q, A, b, J, d, \xi^-, \xi^+).$$

B. Compact QP-Method

A compact-QP method was developed to improve the computational efficiency of solving quadratic programs [21][24]. The compact-QP method entails Gaussian elimination with partial pivoting, which is possibly of $O(n^3)$ operations. In general, it is a more efficient numerical QP method, compared to general-purpose optimization algorithms.

C. Dual Neural Network

As a parallel-processing dynamic-system based alternative to continuous-time optimization, a dual-neural-network QP solver was developed in [25]. It has piecewise linear dynamics, global (exponential) convergence to optimal solutions, and capability of handling hybrid constraints simultaneously. However, because the dual-neural-network solver requires the inverse of coefficient matrix W, it is only able to solve strictly-convex quadratic programming problems (specifically, W being positive-definite and preferably constant).

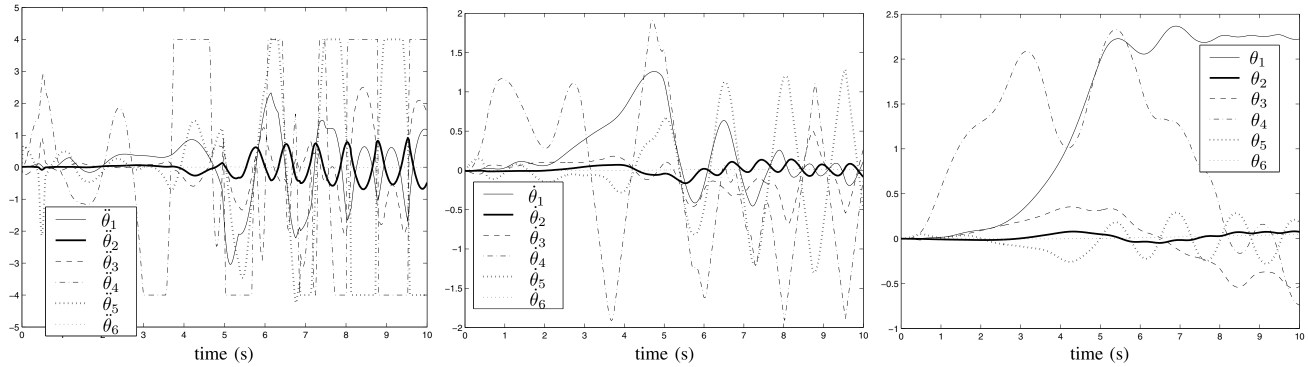


Fig. 2. PUMA560 transients synthesized by the QP-based MTN scheme.

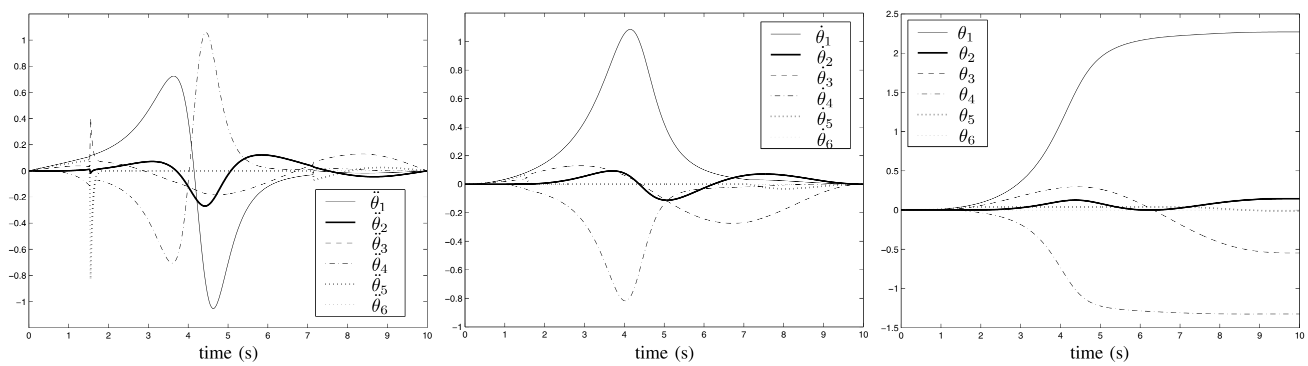


Fig. 3. PUMA560 transients synthesized by the QP-based MKE scheme.

D. LVI-based Primal-Dual Neural Network

Considering that W could be time-varying and positive semi-definite while $\dot{x}$ is required for velocity-level redundancy resolution when applied to joint torque control [1], we have recently developed a primal-dual neural network (PDNN) based on linear variational inequalities (LVI). The LVI-based primal-dual neural network is of simple piecewise linear dynamics, global (exponential) convergence to optimal solutions, and capability of handling general QP and LP problems in the same inverse-free manner [1][26].

We can first convert QP (22)-(25) to a set of linear variational inequalities, and then to the following piecewise-linear equation, $\mathcal{P}_\Omega(y-(Qy+p))-y=0$, where the primal-dual decision vector y, together with its lower/upper bounds, is defined as

$$y=\begin{bmatrix}x\\u\\v\end{bmatrix},\ y^-=\begin{bmatrix}\xi^-\\-1_{\mathrm{v}}\varpi\\0\end{bmatrix},\ y^+=\begin{bmatrix}\xi^+\\1_{\mathrm{v}}\varpi\\1_{\mathrm{v}}\varpi\end{bmatrix},$$

with $1_{\mathrm{v}}:=[1,\cdots,1]^T$ denoting an appropriately dimensioned vector composed of ones, and $\varpi\gg 0$ being sufficiently large to represent $+\infty$. The augmented coefficients are defined as

$$Q=\begin{bmatrix}W & -J^T & A^T\\ J & 0 & 0\\ -A & 0 & 0\end{bmatrix},\ p=\begin{bmatrix}q\\-d\\b\end{bmatrix}.$$

The set $\Omega:=\{y|y^-\leqslant y\leqslant y^+\}$, and the piecewise linear projection operator $\mathcal{P}_\Omega(\cdot)$ is defined as $\mathcal{P}_\Omega(y):=[\mathcal{P}_\Omega(y_1),\cdots,\mathcal{P}_\Omega(y_i),\cdots]^T$ with the ith element being [25]

$$\mathcal{P}_\Omega(y_i)=\begin{cases}y_i^- & \text{if } y_i<y_i^-\\ y_i & \text{if } y_i^-\leqslant y_i\leqslant y_i^+\\ y_i^+ & \text{if } y_i>y_i^+\end{cases},\ \forall i.$$

From our neural-network design experience ([1][25][26] and references therein), it follows that the LVI-based primal-dual neural network, being the QP solver for (22)-(25), can use the following dynamics (see [1] and references therein):

$$\dot{y}=\gamma(I+Q^T)\{\mathcal{P}_\Omega(y-(Qy+p))-y\} \tag{26}$$

where $\gamma>0$ is the design parameter used to scale the network convergence. Furthermore, we have global (exponential) convergence of neural network (26) [1][26].

VIII. Simulation Studies

The research of the past ten years has finally achieved a unification of various redundancy-resolution schemes through the quadratic-programming formulation. In the 1990s, computer simulations were mainly based on planar or theoretical robot arms; e.g., three- or four-link planar robots [17][20][21]. Recently, a large number of QP-based computer simulations have been performed more oriented towards PUMA560, PA10, or other industrial spatial robot arms [1][5][6][9]-[12]. Due to space limitation, only closely-related simulation results are summarized below with illustrative observations presented.

- This QP formulation could keep the joints within their physical limits [5][9]. One more example is in Fig. 2, which corresponds to Figs. 3 and 5 (the MTN sub-figure) in [1].
- This QP formulation could avoid multiple obstacles in a crowded dynamic environment which includes point obstacles and window-shaped obstacle. See [6][11].
- This QP formulation could handle various performance indices in the same unified framework, such as, the aforementioned MVN, RMP, MKE, MAN, MTN and IIWT schemes [1][5][9][10].
- This QP formulation and its dynamic solver (26) could generate accurately joint acceleration information for velocity-level redundancy-resolution schemes if required for joint torque control. See Fig. 3 which corresponds to Figs. 4 and 5 (MKE sub-figures) in [1].
- The velocity-level redundancy resolution in (3)-(7) is more desirable for long trajectory movements [1][8][14]. In contrast, the acceleration-level redundancy resolution in (8)-(14) can handle more joint physical limits simultaneously, which could naturally remedy the torque divergence problem as well [1][15], where the MAN scheme is preferred one.

IX. Conclusions

This paper has unified the minimum-energy redundancy resolution of robot manipulators into a general QP formulation; i.e., in (22)-(25). This QP formulation can cover the online avoidance of joint physical limits and environmental obstacles, as well as optimizing various performance indices. Every term has clear physical meaning and utility. Four QP online solvers have also been reviewed for solving QP (22)-(25). The LVI-based primal-dual neural network is one of the state-of-the-art QP real-time solvers, which have been applied to the online minimum-energy redundancy resolution of redundant manipulators. A large number of simulation results have been summarized based on PUMA560, PA10, and other robot arms. This research of recent ten years has basically answered the issue of manipulators' redundancy resolution via quadratic-programming approaches.

References

[1] Y. Zhang, S. S. Ge, and T. H. Lee, "A unified quadratic programming based dynamical system approach to joint torque optimization of physically constrained redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, pp. 2126-2132, 2004.

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

[3] M. L. Latash, *Control of Human Movement*, Chicago: Human Kinetics Pubisher, 1993.

[4] S. Ma, Y. Ohmameuda, and K. Inoue, "Dynamic analysis of 3-dimensional snake robots," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp. 767-772, 2004.

[5] Y. Zhang, J. Wang, and Y. Xia, "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 658-667, 2003.

[6] Y. Zhang and J. Wang, "Obstacle avoidance for kinematically redundant manipulators using a dual neural network," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, pp. 752-759, 2004.

[7] Y. Zhang, Z. Li, and H.-Z. Tan, "Inequality-based manipulator-obstacle avoidance using the LVI-based primal-dual neural network," *Proceedings of IEEE International Conference on Robotics and Biomimetics,* pp. 1459-1464, 2006.

[8] T.-H. Chen, F.-T. Cheng, Y.-Y. Sun, and M.-H. Hung, "Torque optimization schemes for kinematically redundant manipulators," *Journal of Robotic Systems,* vol. 11, no. 4, pp. 257-269, 1994.

[9] Y. Zhang and J. Wang, "A dual neural network for constrained joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B* vol. 32, no. 5, pp. 654-662, 2002.

[10] Y. Zhang, J. Wang, and Y. Xu, "A dual neural network for bi-criteria kinematic control of redundant manipulators," *IEEE Transactions on Robotics and Automation,* vol. 18, no. 6, pp. 923-931, 2002.

[11] Y. Zhang and J. Wang, "Obstacle avoidance of redundant manipulators using a dual neural network," *Proceedings of IEEE International Conference on Robotics and Automation,* 2003, vol. 2, pp. 2747-2752.

[12] J. Wang and Y. Zhang, "Recurrent neural networks for real-time computation of inverse kinematics of redundant manipulators," *Machine Intelligence: Quo Vadis?* Singapore: World Scientific, pp. 299-319, 2004.

[13] S. Ma, "A new formulation technique for local torque optimization of redundant manipulators," *IEEE Transactions on Industrial Electronics,* vol. 43, no. 4, pp. 462-468, 1996.

[14] K. A. O'Neil, "Divergence of linear acceleration-based redundancy resolution schemes," *IEEE Transactions on Robotics and Automation,* vol. 18, no. 4, pp. 625-631, 2002.

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

[16] J. Guo and T. C. Hsia, "Joint trajectory generation for redundant robots in an environment with obstacles," *Journal of Robotic Systems,* vol. 10, pp. 199-215, 1993.

[17] F.-T. Cheng, Y.-T. Lu, and Y.-Y. Sun, "Window-shaped obstacle avoidance for a redundant manipulator," *IEEE Transactions on Systems, Man, and Cybernetics, Part B,* vol. 28, no. 6, pp. 806-815, 1998.

[18] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *International Journal of Robotic Research,* vol. 4, pp. 109-117, 1985.

[19] K. Glass, R. Colbaugh, D. Lim, and H. Seraji, "Real-time collision avoidance for redundant manipulators," *IEEE Transactions on Robotics and Automation,* vol. 11, no. 3, pp. 448-457, 1995.

[20] W. Tang, L. Lam, and J. Wang, "Kinematic control and obstacle avoidance for redundant manipulators using a recurrent neural network," *Proceedings of International Conference on Artificial Neural Networks,* pp. 922-929, 2001.

[21] F.-T. Cheng, T.-H. Chen, and Y.-Y. Sun, "Resolving manipulator redundancy under inequality constraints," *IEEE Transactions on Robotics and Automation,* vol. 10, no. 1, pp. 65-71, 1994.

[22] A. Nedungadi and K. Kazerounian, "A local solution with global characteristics for joint torque optimization of a redundant manipulator," *Journal of Robotic Systems,* vol. 6, pp. 631-654, 1989.

[23] The MathWorks Inc. *MATLAB Optimization Toolbox*, 2003.

[24] F.-T. Cheng, R.-J. Sheu, and T.-H. Chen, "The improved compact QP method for resolving manipulator redundancy," *IEEE Transactions on Systems, Man, and Cybernetics, Part B,* vol. 25, no. 11, pp. 1521-1530, 1995.

[25] Y. Zhang and J. Wang, "A dual neural network for convex quadratic programming subject to linear equality and inequality constraints," *Physics Letters A,* vol. 298, no. 4, pp. 271-278, 2002.

[26] Y. Zhang, "On the LVI-based primal-dual neural network for solving online linear and quadratic programming problems," *Proceedings of American Control Conference,* pp. 1351-1356, 2005.