Robotica (2009) volume 27, pp. 873–881. © Cambridge University Press 2009
doi:10.1017/S0263574708005274

Adaptive control of redundant robot manipulators with sub-task objectives*

Enver Tatlicioglu†**, David Braganza‡, Timothy C. Burg§ and Darren M. Dawson§

†*Department of Electrical & Electronics Engineering, Izmir Institute of Technology, Gulbahce Koyu, Urla, Izmir, 35430, Turkey*
‡*Optical Fiber Solutions (OFS), 50 Hall Road, Stourbridge, MA 01566, USA*
§*Department of Electrical & Computer Engineering, Clemson University Clemson, SC 29634-0915, USA*

(Received in Final Form: December 9, 2008. First published online: January 15, 2009)

SUMMARY

In this paper, adaptive control of kinematically redundant robot manipulators is considered. An end-effector tracking controller is designed and the manipulator's kinematic redundancy is utilized to integrate a general sub-task controller for self-motion control. The control objectives are achieved by designing a feedback linearizing controller that includes a least-squares estimation algorithm to compensate for the parametric uncertainties. Numerical simulation results are presented to show the validity of the proposed controller.

KEYWORDS: Adaptive control; redundant robot; sub-task.

1. Introduction

When the number of joints of a robot manipulator is greater than the dimension of its task-space position vector then it is called a kinematically redundant robot manipulator. In many applications, robot manipulators with such additional degrees of freedom are preferred to execute complicated tasks. This kinematic redundancy can result in joint motion in the null space of the Jacobian matrix that does not affect the end-effector position, this phenomenon is commonly referred to as *self-motion*. There are generally an infinite number of solutions for the inverse kinematics of redundant robot manipulators[2–4]; this complicates the control of kinematically redundant robot manipulators since it is difficult to select a reasonable, desired joint trajectory for a given desired task-space trajectory.

In our previous work,[5] an adaptive full-state feedback-quaternion-based controller developed in ref. [6] was utilized and a general sub-task controller was designed. In ref. [5], the sub-task controller was systematically integrated into the stability analysis and specific sub-task objectives (such as singularity avoidance, joint limit avoidance, bounding the impact forces and bounding the potential energy) were introduced to make use of the kinematic redundancy. In ref. [7], configuration control of redundant robot manipulators was investigated. The proposed controller achieved task-space tracking and the redundancy was utilized to impose kinematic and dynamic constraints or posture control. Hsu *et al.*[8] proposed a dynamic feedback linearizing control law that guarantees asymptotic tracking of a desired task-space trajectory. However, the controller in ref. [8] required that the exact dynamic model of the robot manipulator should be known. Zergeroglu *et al.*[9] used the controller in ref. [8] as a basis and developed an adaptive controller to compensate for the parametric uncertainty in the dynamic model. In refs. [8] and [9] both, the researchers provided control of the redundant link velocities to accomplish desirable sub-task objectives. Peng and Adachi[10] proposed two compliant motion controllers for redundant manipulators where the redundancy was utilized to realize additional constraints that optimize a user-defined objective function. Nakamura[2] categorized the tasks to be accomplished by a kinematically redundant robot manipulator based on their respective priorities. Specifically, in chapter 4 of ref. [2], examples of how orientation of the end-effector of a redundant robot manipulator may have more significance than the position of the end-effector or vice versa were provided and then the tasks were classified based on their level of significance. For a more detailed overview of the research on redundant robot manipulators, the reader is referred to refs. [2, 6, 9, 11, 12] and the references therein.

In this paper, the feedback linearizing controller in ref. [8] is redesigned to compensate for parametric uncertainties present in the dynamic model. The control development is presented in two parts, namely, task-space control and null-space control. The task-space controller is designed to meet the task-space tracking objective, that is tracking a desired task-space trajectory. The design of the task-space controller allows the integration of a null-space controller to make use of the redundancy resolution. This auxiliary null-space controller is designed through the joint motion in the null-space of the Jacobian matrix (i.e. self-motion) and by controlling the joint velocities in the null-space, we can integrate sub-task control objectives (the reader is referred to ref. [5] for specific sub-task objectives) and achieve a stable system. The integration of a sub-task objective into the controller is completed by designing a framework that

* A preliminary version of this paper has appeared in [1].
**Corresponding author. E-mail: enver@envertatlicioglu.com.

places preferences on desirable configurations based on the sub-task objective. In the design of both task-space and null-space controllers, Lyapunov-based analysis techniques are utilized. This work demonstrates a major improvement to our previous work,[5] in the sense that in this paper a null-space velocity tracking objective is defined and proven to be met by driving the null-space velocity tracking error to zero. This null-space control objective allows the sub-task control objectives to be injected into the task-space controller to control the body motion of the robot manipulator. When compared to ref. [2], our control design utilizes Lyapunov-based analysis, which is a powerful tool in control of non-linear systems and in our design the position and orientation of the end-effector is always the primary control objective and the sub-task objectives are utilized to control only the body motion of the manipulator. Review of the adaptive redundant robot control literature (e.g. refs. [5, 9, 13]) suggests that researchers typically prefer gradient-type algorithms for parameter estimation. The design proposed here uses a least-squares algorithm in a seemingly novel departure from adaptive redundant robot control.

2. Dynamic and Kinematic Model

The dynamic model for an n-joint ($n \geq 6$), revolute, direct drive robot manipulator is described by the following expression:

$$M(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) = \tau, \tag{1}$$

where $\theta(t), \dot{\theta}(t), \ddot{\theta}(t) \in \mathbb{R}^n$ denote the position, velocity and acceleration in the joint-space, respectively. In Eq. (1), $M(\theta) \in \mathbb{R}^{n \times n}$ represents the inertia effects, $N(\theta, \dot{\theta}) \in \mathbb{R}^n$ represents other dynamic effects (centripetal-Coriolis effects, gravitational effects, dynamic frictional effects) and $\tau(t) \in \mathbb{R}^n$ represents the control input torque vector. The subsequent development is based on the following properties.[14]

Property 1 *The inertia matrix $M(\theta)$ is symmetric and positive-definite, and satisfies the following inequalities*

$$m_1 \|\xi\|^2 \leq \xi^T M(\theta) \xi \leq m_2 \|\xi\|^2 \qquad \forall \xi \in \mathbb{R}^n, \tag{2}$$

where $m_1, m_2 \in \mathbb{R}$ are positive constants and $\|\cdot\|$ denotes the standard Euclidean norm.

Property 2 *The left-hand side of (1) can be linearly parameterized as*

$$M(\theta)\ddot{\theta} + N(\theta, \dot{\theta}) = Y(\theta, \dot{\theta}, \ddot{\theta})\phi, \tag{3}$$

where $\phi \in \mathbb{R}^p$ contains the constant system parameters and the regression matrix $Y(\cdot) \in \mathbb{R}^{n \times p}$ contains known functions dependent on the signals $\theta(t)$, $\dot{\theta}(t)$ and $\ddot{\theta}(t)$.

The kinematic model for the robot manipulator is described by the following expression:

$$\dot{x} = J(\theta)\dot{\theta}, \tag{4}$$

where $x(t) \in \mathbb{R}^m$ is the task-space position and $J(\theta) \in \mathbb{R}^{m \times n}$ is the manipulator Jacobian matrix. The subsequent development is based on the assumption that $x(t)$, $\dot{x}(t)$, $\theta(t)$ and $\dot{\theta}(t)$ are measurable.

Remark 1 *The dynamic and kinematic terms for a general revolute robot manipulator, denoted by $M(\theta)$, $N(\theta, \dot{\theta})$, $J(\theta)$ and $J^+(\theta)$, are assumed to depend on $\theta(t)$ only as arguments of trigonometric functions, and hence remain bounded for all possible $\theta(t)$. During the control development, the assumption will be made that if $x(t)$ is bounded then $\theta(t)$ is a bounded signal.*

3. Pseudo-Inverse and Its Properties

The pseudo-inverse of the Jacobian, denoted by $J^+(\theta) \in \mathbb{R}^{n \times m}$, is defined as follows:

$$J^+ \triangleq J^T (J J^T)^{-1}. \tag{5}$$

From (5), it is clear that $J^+(\theta)$ satisfies the following*

$$J J^+ = I_m. \tag{6}$$

As shown in ref. [2], the pseudo-inverse defined by (5) satisfies the Moore–Penrose conditions as follows:

$$J J^+ J = J, \quad J^+ J J^+ = J^+, \tag{7}$$

$$(J^+ J)^T = J^+ J, \quad (J J^+)^T = J J^+. \tag{8}$$

In addition to the above properties, the matrix $(I_n - J^+ J)$ satisfies the following properties:

$$(I_n - J^+ J)(I_n - J^+ J) = I_n - J^+ J, \tag{9}$$

$$(I_n - J^+ J)^T = (I_n - J^+ J), \tag{10}$$

$$J(I_n - J^+ J) = 0_{n \times 1}, \tag{11}$$

$$(I_n - J^+ J) J^+ = 0_{n \times 1}. \tag{12}$$

The following expression can be obtained for the time derivative of $J^+ J$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\{J^+ J\} = J_\phi + J^+ \dot{J}(I_n - J^+ J), \tag{13}$$

where $J_\phi(t) \in \mathbb{R}^{n \times n}$ is an auxiliary function defined as follows:

$$J_\phi \triangleq \dot{J}^+ J + J^+ \dot{J} J^+ J. \tag{14}$$

It should be noted that $J_\phi(t)$ satisfies the following property:

$$\begin{aligned} J J_\phi &= (J \dot{J}^+ + J J^+ \dot{J} J^+) J, \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\{J J^+\} J, \\ &= 0_{n \times n}, \end{aligned} \tag{15}$$

* Throughout the paper, I_n and $0_{m \times r}$ will be used to represent an $n \times n$ standard identity matrix and an $m \times r$ zero matrix, respectively.

where Eq. (6) was used. In addition, the following property will also be utilized throughout the subsequent analysis:

$$(I_n - J^+J)J_\phi = J_\phi - J^+JJ_\phi, \\ = J_\phi, \qquad (16)$$

where Eq. (15) was used.

Remark 2 *During the subsequent control development, the assumption is made that the minimum singular value of the manipulator Jacobian matrix, denoted by σ_m, is greater than a known small positive constant $\delta > 0$, such that* $\max\{\|J^+(\theta)\|\}$ *is known a priori and all kinematic singularities are always avoided.*

4. Task-Space Controller Development

The primary control design objective is to formulate a control input that ensures that the end-effector of the manipulator tracks a desired trajectory. The task-space tracking error denoted by $e(t) \in \mathbb{R}^m$ is defined as follows:

$$e \triangleq x_d - x, \qquad (17)$$

where $x_d(t) \in \mathbb{R}^m$ is the task-space desired trajectory. In the subsequent development, it will be assumed that $x_d(t)$, $\dot{x}_d(t)$ and $\ddot{x}_d(t)$ are bounded signals.

Based on (4), the following expression can be obtained for the joint velocities:

$$\dot{\theta} = J^+\dot{x} + (I_n - J^+J)\dot{\theta}. \qquad (18)$$

To facilitate the task-space controller development, the time derivative of (18) is given as follows:

$$\ddot{\theta} = J^+\ddot{x} + \dot{J}^+\dot{x} - \frac{\mathrm{d}}{\mathrm{d}t}\{J^+J\}\dot{\theta} + (I_n - J^+J)\ddot{\theta}. \qquad (19)$$

After using Eq. (4), the following simplified expression can be obtained for $\ddot{\theta}(t)$:

$$\ddot{\theta} = J^+(\ddot{x} - \dot{J}\dot{\theta}) + \ddot{\theta}_N, \qquad (20)$$

where $\ddot{\theta}_N(t) \in \mathbb{R}^n$ is defined as follows:

$$\ddot{\theta}_N \triangleq (I_n - J^+J)\ddot{\theta}. \qquad (21)$$

The estimation form of (3) is defined as

$$\hat{M}(t)\ddot{\theta} + \hat{N}(t) = Y(\theta, \dot{\theta}, \ddot{\theta})\hat{\phi}, \qquad (22)$$

where $\hat{\phi}(t) \in \mathbb{R}^p$, $\hat{M}(t)$ and $\hat{N}(t)$ are the estimates of ϕ, $M(\theta)$ and $N(\theta, \dot{\theta})$, respectively. After subtracting (22) from the manipulator's dynamics in (1), the following is obtained:

$$Y\tilde{\phi} = \tau - (\hat{M}\ddot{\theta} + \hat{N}), \qquad (23)$$

where $\tilde{\phi}(t) \in \mathbb{R}^p$ is the parameter estimation error defined as

$$\tilde{\phi} \triangleq \phi - \hat{\phi}. \qquad (24)$$

After pre-multiplying (23) by $\hat{M}^{-1}(t)$, the following expression can be obtained:

$$\hat{M}^{-1}Y\tilde{\phi} = \hat{M}^{-1}\tau - \hat{M}^{-1}\hat{N} - \ddot{\theta}, \qquad (25)$$

for the open-loop error system. To facilitate the subsequent analysis the control input $\tau(t)$ is designed as follows:

$$\tau \triangleq \hat{M}[J^+u_1 + \phi_N] + \hat{N} + u_2, \qquad (26)$$

where $u_1(t) \in \mathbb{R}^m$, $u_2(t) \in \mathbb{R}^n$ are auxiliary control inputs and $\phi_N(t) \in \mathbb{R}^n$ is a vector in the null-space of $J(t)$. The auxiliary control input $u_1(t)$ is designed as

$$u_1 \triangleq \ddot{x}_d + k_v\dot{e} + k_pe - \dot{J}\dot{\theta} + u_{\text{aux}}, \qquad (27)$$

where k_v and k_p are positive constants, and $u_{\text{aux}}(t) \in \mathbb{R}^m$ is another auxiliary control input that will be designed subsequently. After substituting (26) and (27) into the open-loop error system in (25), the following expression is obtained:

$$\hat{M}^{-1}Y\tilde{\phi} = J^+(\ddot{e} + k_v\dot{e} + k_pe + u_{\text{aux}}) \\ + \hat{M}^{-1}u_2 + \phi_N - \ddot{\theta}_N. \qquad (28)$$

After pre-multiplying (28) by $J(t)$ and rearranging, the following expression can be obtained:

$$\ddot{e} + k_v\dot{e} + k_pe + u_{\text{aux}} = J\hat{M}^{-1}(Y\tilde{\phi} - u_2), \qquad (29)$$

where Eq. (6) and the following facts were used:

$$J\phi_N = 0_{m\times 1}, \quad J\ddot{\theta}_N = 0_{m\times 1}. \qquad (30)$$

It should be noted that since $\ddot{\theta}(t)$ is an unmeasurable signal, the regression matrix $Y(\theta, \dot{\theta}, \ddot{\theta})$ introduced in Eq. (3) is unmeasurable. To tackle this issue, a filtered regression matrix $Y_f(t) \in \mathbb{R}^{n\times p}$ is introduced[15]:

$$\dot{Y}_f \triangleq -\alpha Y_f + \alpha Y, \quad Y_f(t_0) \triangleq 0_{n\times p}, \qquad (31)$$

where $\alpha \in \mathbb{R}$ is a positive constant. Notice that Eq. (31) cannot be implemented since $Y(\theta, \dot{\theta}, \ddot{\theta})$ is unmeasurable. For an implementable form of the filtered regression matrix see Appendix B. A filtered control input is defined similarly[15]

$$\dot{\tau}_f \triangleq -\alpha\tau_f + \alpha\tau, \quad \tau_f(t_0) \triangleq 0_{n\times 1}. \qquad (32)$$

To facilitate the subsequent analysis a prediction error, denoted by $z(t) \in \mathbb{R}^n$, is defined as follows:

$$z \triangleq \hat{M}^{-1}(\tau_f - Y_f\hat{\phi}). \qquad (33)$$

After using the development given in Appendix A, the prediction error in Eq. (33) can be written as follows:

$$z = \hat{M}^{-1} Y_f \tilde{\phi}, \tag{34}$$

where (24) was also utilized. The auxiliary control input $u_2(t)$ is designed as

$$u_2 \overset{\triangle}{=} \frac{1}{\alpha} Y_f \dot{\hat{\phi}} + \frac{1}{\alpha} \dot{\hat{M}} z. \tag{35}$$

After substituting $u_2(t)$ into Eq. (29), the following expression can be obtained:

$$\ddot{e} + k_v \dot{e} + k_p e = J\left(\frac{1}{\alpha}\dot{z} + z\right) - u_{\text{aux}}, \tag{36}$$

where Eq. (34) and its time derivative were utilized. A filtered tracking error, denoted by $r(t) \in \mathbb{R}^m$, is defined to be of the following form:

$$r \overset{\triangle}{=} \dot{e} + \sigma_1 e, \tag{37}$$

where $\sigma_1 \in \mathbb{R}$ is a positive constant. After setting the constant control gains k_v and k_p, which were introduced in Eq. (27), as

$$k_v \overset{\triangle}{=} \sigma_1 + \sigma_2, \quad k_p \overset{\triangle}{=} \sigma_1 \sigma_2, \tag{38}$$

the left-hand side of Eq. (36) can be written as

$$\ddot{e} + k_v \dot{e} + k_p e = \dot{r} + \sigma_2 r, \tag{39}$$

where $\sigma_2 \in \mathbb{R}$ is a positive constant. After using Eq. (39), the expression in Eq. (36) can be rewritten as

$$\dot{r} + \sigma_2 r = J\left(\frac{1}{\alpha}\dot{z} + z\right) - u_{\text{aux}}. \tag{40}$$

To facilitate the subsequent stability analysis, an auxiliary error signal, denoted by $y(t) \in \mathbb{R}^m$ is defined as

$$y \overset{\triangle}{=} r - \frac{1}{\alpha} J z. \tag{41}$$

The time derivative of $y(t)$ is given as

$$\dot{y} = -\sigma_2 y + \left(1 - \frac{\sigma_2}{\alpha}\right) J z - \frac{1}{\alpha} \dot{J} z - u_{\text{aux}}, \tag{42}$$

where Eqs. (40) and (41) were utilized. The auxiliary control input $u_{\text{aux}}(t)$ is designed as

$$u_{\text{aux}} \overset{\triangle}{=} \left(1 - \frac{\sigma_2}{\alpha}\right) J z - \frac{1}{\alpha} \dot{J} z. \tag{43}$$

After substituting (43) into (42), the following simplified expression is obtained for the dynamics of $y(t)$:

$$\dot{y} = -\sigma_2 y. \tag{44}$$

From (44), standard analysis techniques can be utilized to show that

$$y(t) = y(t_0)\exp(-\sigma_2 t) \tag{45}$$

from which we can conclude that $\|y(t)\| \to 0$ exponentially fast. Motivated by the subsequent stability analysis, the parameter estimate vector $\hat{\phi}(t)$ is generated by the following update law:

$$\dot{\hat{\phi}} \overset{\triangle}{=} \Gamma Y_f^T \hat{M}^{-T} z, \tag{46}$$

where $\Gamma(t) \in \mathbb{R}^{p \times p}$ is a least-squares estimation gain matrix designed as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}(\Gamma^{-1}) \overset{\triangle}{=} Y_f^T \hat{M}^{-T} \hat{M}^{-1} Y_f. \tag{47}$$

Remark 3 *It should be noted that when $\Gamma^{-1}(t_0)$ is selected to be positive definite and symmetric, then it is clear that $\Gamma(t_0)$ is also positive definite and symmetric. Therefore, it follows that both $\Gamma^{-1}(t)$ and $\Gamma(t)$ will remain positive definite and symmetric $\forall t$. From (47), the following expression can be obtained*:

$$\dot{\Gamma} = -\Gamma Y_f^T \hat{M}^{-T} \hat{M}^{-1} Y_f \Gamma. \tag{48}$$

From (48), it is easy to see that $\dot{\Gamma}(t)$ is negative semi-definite; therefore, the estimation gain matrix $\Gamma(t)$ is always constant or decreasing, and hence $\Gamma(t)$ is bounded (for more details, the reader is referred to refs. [15] and [16]).

Remark 4 *The matrix inverse of the estimate of $M(\theta)$ (i.e. $\hat{M}(\theta)$) can be guaranteed to exist through the use of a projection algorithm as described in ref. [17].*

Theorem 1 *The control law described in (26), (27), (35) and (43) and the adaptation law defined in (46) guarantee that $z(t)$, $r(t)$ and $e(t)$ are driven to zero as $t \to \infty$.*

Proof See Appendix C.

Remark 5 *The proof of Theorem 1 requires the boundedness of $\dot{\theta}(t)$ and $\phi_N(t)$. In the subsequent sections, an auxiliary null-space control signal, denoted by $g(\theta)$, will be designed to meet these conditions.*

5. Sub-Task Error System

In addition to the end-effector tracking objective, there may be sub-task objectives that are required for a particular redundant robot application. To integrate the sub-task objective into the controller, an auxiliary control signal, denoted by $g(\theta)$, will be introduced. The integration of this sub-task objective into the controller is completed by designing a framework that places preferences on desirable configurations based on the sub-task objective. The auxiliary null-space controller $g(\theta)$ is designed through the joint motion in the null-space of the Jacobian matrix (i.e. *self-motion*).

The null-space velocity tracking error is defined as[8]

$$\dot{e}_N \overset{\triangle}{=} (I_n - J^+ J)(g - \dot{\theta}), \tag{49}$$

where $g(t) \in \mathbb{R}^n$ is the subsequently designed null-space controller. The following expression can be obtained for the dynamics of $\dot{e}_N(t)$

$$\begin{aligned}\ddot{e}_N = (I_n - J^+ J)\dot{g} + (I_n - J^+ J)\left(\frac{1}{\alpha}\dot{z} + z - \phi_N\right)\\ -J_\phi(g - \dot{\theta}) - J^+ \dot{J}\dot{e}_N,\end{aligned} \tag{50}$$

where Eqs. (12), (13) and (49) were utilized along with the following expression for $\ddot{\theta}(t)$:

$$\ddot{\theta} = -\left(\frac{1}{\alpha}\dot{z} + z\right) + J^+ u_1 + \phi_N, \tag{51}$$

where Eqs. (20), (25), (26), (34), the time derivative of (34) and (35) were utilized. To facilitate the null-space control development, an auxiliary error signal, denoted by $p(t) \in \mathbb{R}^n$, is defined as follows:

$$p \overset{\triangle}{=} \dot{e}_N - \frac{1}{\alpha}(I_n - J^+ J)z. \tag{52}$$

The dynamics of $p(t)$ can be written as

$$\begin{aligned}\dot{p} = (I_n - J^+ J)\dot{g} + (I_n - J^+ J)(z - \phi_N)\\ -J_\phi\left(g - \dot{\theta} - \frac{1}{\alpha}z\right) - J^+ \dot{J} p,\end{aligned} \tag{53}$$

where Eqs. (13), (50) and (52) were utilized. The auxiliary null-space vector $\phi_N(t)$, introduced in Eq. (26), is now designed as follows:

$$\begin{aligned}\phi_N \overset{\triangle}{=} (I_n - J^+ J)(\dot{g} + k_n p + z)\\ + J_\phi\left(g - \dot{\theta} - \frac{1}{\alpha}z\right),\end{aligned} \tag{54}$$

where $k_n \in \mathbb{R}$ is a positive constant. After substituting $\phi_N(t)$ into Eq. (53) the following simplified expression is obtained for the dynamics of $p(t)$:

$$\dot{p} = -k_n(I_n - J^+ J)p - J^+ \dot{J} p, \tag{55}$$

where Eqs. (9) and (16) were used.

Theorem 2 *The auxiliary null-space vector described by (54) guarantees that $\dot{e}_N(t)$ is driven to zero as $t \to \infty$.*

Proof See Appendix D.

6. Sub-Task Controller

In this section, a general sub-task controller is developed. As proven in the subsequent stability analysis, the sub-task objective will be met if the Jacobian-related null-space matrix maintains full rank. Specifically, when the subsequently defined Jacobian-related null-space matrix loses rank, the sub-task objective will not be met.

An auxiliary positive function $y_a(t) \in \mathbb{R}$ is defined as

$$y_a \overset{\triangle}{=} \exp(-k_y \beta(\theta)), \tag{56}$$

where $k_y \in \mathbb{R}$ is a positive constant, $\beta(\theta) \in \mathbb{R}$ is a non-negative function that is specific to each sub-task and $\exp(\cdot)$ is the natural logarithmic exponential function. After taking the time derivative of (56), the following simplified expression is obtained for the dynamics of $y_a(t)$:

$$\dot{y}_a = J_s \dot{\theta}, \tag{57}$$

where $J_s(t) \in \mathbb{R}^{1\times n}$ is a Jacobian-type vector defined as follows:

$$J_s = \frac{\partial y_a}{\partial \theta}. \tag{58}$$

After adding and subtracting the terms $J_s J^+ J \dot{\theta}$ and $J_s(I_n - J^+ J)(g - \dot{\theta})$ to the right-hand side of Eq. (57), we obtain the following for the time derivative of $y_a(t)$:

$$\dot{y}_a = J_s J^+ \dot{x} + J_s(I_n - J^+ J)g - J_s \dot{e}_N, \tag{59}$$

where (4) and (49) were used. Based on the subsequent stability analysis, the sub-task controller is designed as

$$g \overset{\triangle}{=} -k_s J_s^T y_a, \tag{60}$$

where $k_s \in \mathbb{R}$ is a positive constant. After substituting Eq. (60) into Eq. (59), we obtain the following expression:

$$\begin{aligned}\dot{y}_a = -k_s J_s(I_n - J^+ J)(I_n - J^+ J)J_s^T y_a\\ +J_s J^+ \dot{x} - J_s \dot{e}_N,\end{aligned} \tag{61}$$

where Eq. (9) was utilized.

Remark 6 *The auxiliary signal $y_a(t)$ in Eq. (56) was preferred because of the useful properties of the logarithmic exponential function given that many different positive functions could also be utilized for the design of $y_a(t)$. From Eq. (56), it is clear that as $\beta(\theta)$ increases, $y_a(t)$ decreases and $y_a(t)$ satisfies these inequalities,* $0 < y_a(t) \leq 1$.

The following theorem is stated to show the performance of the sub-task controller.

Theorem 3 *The control law described by Eq. (60) guarantees that $y_a(t)$ is practically regulated (i.e. ultimately bounded) in the following sense*:

$$|y_a(t)| \leq \sqrt{y_a^2(t_0)\exp(-2\gamma t) + \frac{\varepsilon}{\gamma}}, \tag{62}$$

provided the following sufficient conditions hold:

$$\|J_s(I_n - J^+ J)\|^2 > \bar{\delta}, \tag{63}$$

$$\|J_s(J^+\dot{x} - \dot{e}_N)\| \leq \delta_1, \tag{64}$$

$$k_s > \frac{1}{\bar{\delta}\delta_2}, \tag{65}$$

where ε, γ, $\bar{\delta}$, δ_1, $\delta_2 \in \mathbb{R}$ are positive constants.

Proof The reader is referred to Appendix A of ref. [18] for a similar proof.

Remark 7 *For specific sub-task objectives including singularity avoidance, joint-limit avoidance, bounding the impact forces and bounding the potential energy, the reader is referred to ref. [18].*

Remark 8 *The sub-task objective is met only if the sufficient conditions described by the inequalities in (63)–(65) are satisfied.* From the result of Theorem 1, the task-space tracking objective is guaranteed and the sub-task objective is always secondary to it. When the sub-task controller forces the end-effector of the robot manipulator to take a path not allowed by the task-space tracking controller, the condition in (63) will not be satisfied; hence, the result of Theorem 3 will not hold. To meet the task-space tracking and sub-task objectives simultaneously, careful consideration is required in the design of the desired task-space trajectory and the sub-task objective.*

7. Numerical Simulation Results

To illustrate the performance of the task-space and sub-task controllers, a numerical simulation was conducted using the model of a planar, three-joint, revolute robot manipulator. Since most real robot manipulators have mechanical limits on the joint rotations, joint limit avoidance is considered as an interesting sub-task control objective. The objective for this sub-task is to keep each joint away from its respective joint limits, while meeting the task-space tracking control objective. For this sub-task, the auxiliary signal $\beta(\theta)$ is defined as follows^5:

$$\beta \triangleq \prod_{i=1}^{3}\left[\left(1 - \frac{\theta_i}{\theta_i^{\max}}\right)\left(\frac{\theta_i}{\theta_i^{\min}} - 1\right)\right], \tag{66}$$

where $\theta_i^{\min}$ and $\theta_i^{\max} \in \mathbb{R}^+$ are the minimum and maximum joint limits for the ith joint, respectively. For this sub-task objective, the problem is set up to satisfy $\beta(\theta) > 0$ which can be achieved by keeping $y_a(t) < 1$. Based on the result of Theorem 3, from (62), it is clear that $y_a(t) < 1$ if the following inequality holds:

$$\sqrt{y_a^2(t_0) + \frac{\varepsilon}{\gamma}} < 1, \tag{67}$$

which can be achieved by proper selection of the initial conditions of the robot manipulator, control gains k_s, k_y and bounding constants. From (66), it is easy to see that $\beta(\theta) > 0$ will be satisfied as long as all joint angles are within their respective joint limits. From (56), (62), (66) and (67), it is clear that $\beta(\theta) > 0\ \forall t$, provided the sufficient conditions in Eqs. (63)–(65) are met; hence meeting the sub-task objective.

For the simulation, the dynamic model of the robot manipulator was considered to be of the following form:

$$M(\theta)\ddot{\theta} + V_m(\theta, \dot{\theta})\dot{\theta} + F_d\dot{\theta} = \tau. \tag{68}$$

In Eq. (68), $M(\theta) = (m_{i,j})_{3\times3}$, $V_m(\theta, \dot{\theta}) = (v_{i,j})_{3\times3}$ and $F_d \in \mathbb{R}^{3\times3}$ represent the symmetric inertia matrix, the centripetal-Coriolis terms and the frictional effects, respectively. The entries of the inertia matrix and the centripetal-Coriolis terms have the following form**

$$\begin{aligned}
m_{11} &= \phi_1 + 2\phi_4 c_2 + 2\phi_5 c_{23} + 2\phi_6 c_3, & m_{22} &= \phi_2 + 2\phi_6 c_3,\\
m_{12} &= \phi_2 + \phi_4 c_2 + \phi_5 c_{23} + 2\phi_6 c_3, & m_{23} &= \phi_2 + \phi_6 c_3,\\
m_{13} &= \phi_2 + \phi_5 c_{23} + \phi_6 c_3, & m_{33} &= \phi_3,
\end{aligned}$$

$$\begin{aligned}
v_{11} &= -\phi_4 s_2\dot{\theta}_2 - \phi_5 s_{23}\dot{\theta}_2 - \phi_5 s_{23}\dot{\theta}_3 - \phi_6 s_3\dot{\theta}_3,\\
v_{12} &= -(\phi_4 s_2 + \phi_5 s_{23})(\dot{\theta}_1 + \dot{\theta}_2) - \phi_5 s_{23}\dot{\theta}_3 - 2\phi_6 s_3\dot{\theta}_3,\\
v_{13} &= -(\phi_5 s_{23} + \phi_6 s_3)(\dot{\theta}_1 + \dot{\theta}_3) - \phi_5 s_{23}\dot{\theta}_2,\\
v_{21} &= (\phi_4 s_2 + \phi_5 s_{23})\dot{\theta}_1,\\
v_{22} &= -\phi_6 s_3\dot{\theta}_3,\\
v_{23} &= -\phi_6 s_3(2\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3),\\
v_{31} &= (\phi_5 s_{23} + \phi_6 s_3)\dot{\theta}_1,\\
v_{32} &= \phi_6 s_3(2\dot{\theta}_1 + \dot{\theta}_2),\\
v_{33} &= 0,
\end{aligned}$$

the dynamic frictional effects are of the form $F_d \triangleq \operatorname{diag}\{\phi_7, \phi_8, \phi_9\}$ and ϕ_i are the unknown parameters. The task-space is defined by $x(t) \triangleq [x_1(t)\quad x_2(t)]^T$ where $x_1(t), x_2(t) \in \mathbb{R}$ are scalar Euclidean coordinates. The planar three-joint robot manipulator has the following forward kinematics for the end-effector

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \ell_1 c_1 + \ell_2 c_{12} + \ell_3 c_{123} \\ \ell_1 s_1 + \ell_2 s_{12} + \ell_3 s_{123} \end{bmatrix},$$

and the manipulator Jacobian is obtained as follows:

$$J = \begin{bmatrix} -\ell_1 s_1 - \ell_2 s_{12} - \ell_3 s_{123} & \ell_1 c_1 + \ell_2 c_{12} + \ell_3 c_{123} \\ -\ell_2 s_{12} - \ell_3 s_{123} & \ell_2 c_{12} + \ell_3 c_{123} \\ -\ell_3 s_{123} & \ell_3 c_{123} \end{bmatrix}^T,$$

where the link lengths were selected as $\ell_1 = 0.40$ [m], $\ell_2 = 0.36$ [m] and $\ell_3 = 0.32$ [m]. The desired task-space

*It should be noted that, the control gain k_s could be chosen to satisfy Eq. (65). After using Remark 1 and the results of Theorem 1, it can be concluded that $\theta(t)$, $J(\theta)$ and $J^+(\theta)$ are bounded. Since $J_s(\cdot)$ is a function of $\theta(t)$, then from Eq. (58), it is clear that $J_s(\cdot)$ is bounded function. From the results of Theorems 1 and 2, it can be seen that $\dot{x}(t)$ and $\dot{e}_N(t)$ are bounded functions, thus, it can be concluded that the term $\|J_s(J^+\dot{x} - \dot{e}_N)\|$ can be lower bounded by a positive constant.

** The notations c_i, s_i, c_{ij}, s_{ij}, c_{ijk} and s_{ijk} represent $\cos(\theta_i)$, $\sin(\theta_i)$, $\cos(\theta_i + \theta_j)$, $\sin(\theta_i + \theta_j)$, $\cos(\theta_i + \theta_j + \theta_k)$ and $\sin(\theta_i + \theta_j + \theta_k)$ ∀ i, j, k, respectively.

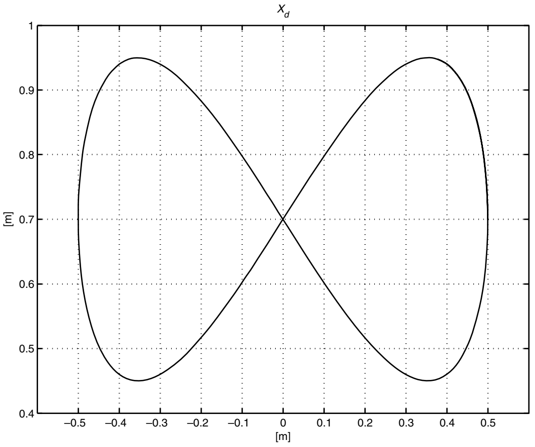

Fig. 1. Desired trajectory for three-link robot.

trajectory $x_d(t) \in \mathbb{R}^2$ is generated by the following bounded dynamic system:

$$\begin{bmatrix} \dot{x}_{d1} \\ \dot{x}_{d2} \end{bmatrix} = 0.05 \begin{bmatrix} -\sin(0.1t) \\ \cos(0.2t) \end{bmatrix}$$

and can be seen in Fig. 1. The control gains were selected as follows:

$$\begin{aligned} \alpha &= 1, \quad \sigma_1 = 4, \quad \sigma_2 = 8, \\ k_n &= 8, \quad k_y = 1, \quad k_s = 2. \end{aligned}$$

The initial conditions of the parameter estimates and estimation gain matrix were selected as follows:

$$\hat{\phi}(t_0) = 0_{9\times 1}, \; \Gamma(t_0) = I_9,$$

and ϕ=[1.2746 0.39464 0.0512 0.4752 0.128 0.1152 8 8 8]^T. The robot manipulator was set to be initially at rest at the following joint configuration:

$$\theta(t_0) = [0.5 \;\; 0.5 \;\; 3.3]^T$$

and the following values were chosen for the joint limits:

$$\begin{aligned} \theta_1^{\min} &= 0.5, \quad \theta_2^{\min} = 0.5, \quad \theta_3^{\min} = 0.1, \\ \theta_1^{\max} &= 2.0, \quad \theta_2^{\max} = 2.0, \quad \theta_3^{\max} = 6.0, \end{aligned}$$

all in radians. The initial configuration of the robot manipulator was intentionally selected to make $\beta(\theta(t_0)) = 0$ i.e. maximize $y_a(t_0)$) to demonstrate that Eq. (62) holds for the simulation. The auxiliary sub-task function $\beta(\theta)$ and the tracking error are presented in Figs. 2 and 3, respectively. From Figs. 2 and 3, it is clear that both the tracking objective and the joint limits sub-task objective were successfully satisfied. In Fig. 4, the parameter estimates are presented. To better demonstrate the effects of the sub-task controller, the simulation was run one more time without the sub-task controller. This was done by setting $k_s = 0$ and keeping everything else the same. The joint positions of the slave system and the auxiliary sub-task function $\beta(t)$ are presented in Figs. 5 and 6, respectively. It is clear that during the simulation run, the first joint position becomes less than its minimum joint limit while the second joint becomes greater than its maximum joint limits, and as a result $\beta(t)$ becomes less than zero.

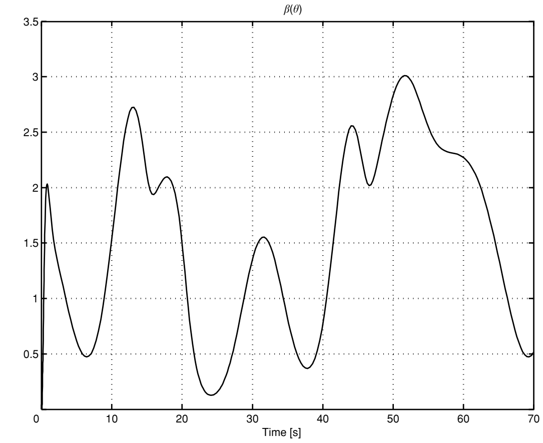

Fig. 2. $\beta(\theta)$ for joint limit avoidance sub-task.

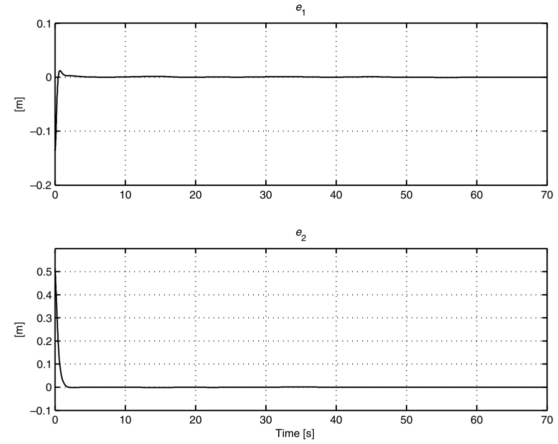

Fig. 3. Tracking error.

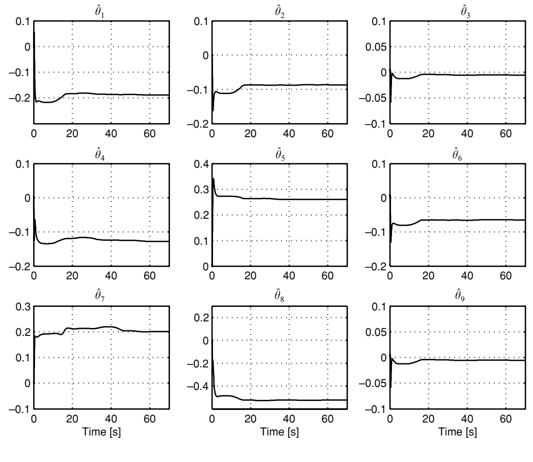

Fig. 4. The parameter estimates $\hat{\phi}(t)$.

8. Conclusions

Lyapunov-based stability analysis techniques were utilized to design a feedback linearizing adaptive controller for kinematically redundant robot manipulators. The controller

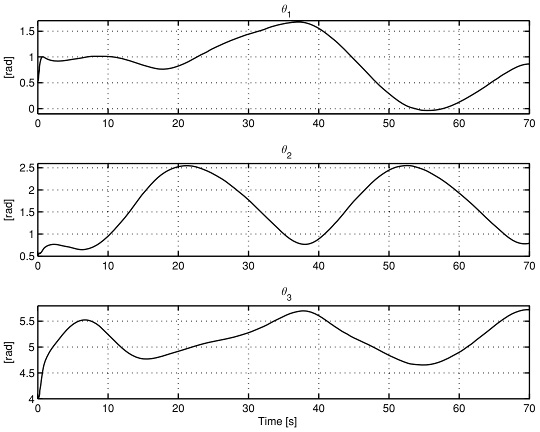

Fig. 5. The joint positions of the robot manipulator.

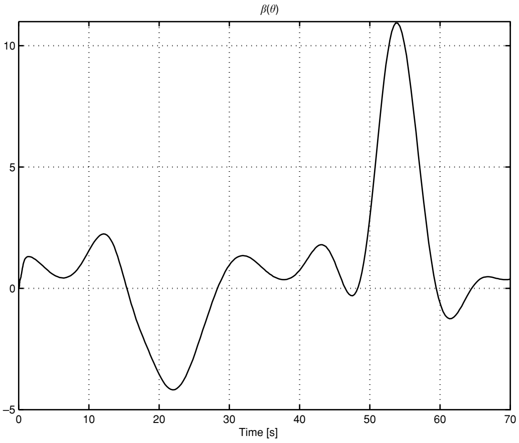

Fig. 6. $\beta(\theta)$ when there is no sub-task controller (*i.e.* $k_s = 0$).

compensates for the parametric uncertainties in the dynamic model using a least-squares estimation algorithm. To our best knowledge, this is novel when compared to the previous adaptive redundant robot control literature. Task-space tracking was achieved and the kinematic redundancy was utilized to integrate a general sub-task controller.

References

1. E. Tatlicioglu, D. Braganza, T. C. Burg and D. M. Dawson, "Adaptive Control of Redundant Robot Manipulators with Sub-task Extensions," *Proceedings of the American Control Conference*, Seattle, WA (2008) pp. 856–861.
2. Y. Nakamura, *Advanced Robotics Redundancy and Optimization* (Addison-Wesley, Reading, MA, 1991).
3. D. N. Nenchev, "Redundancy resolution through local optimization: A review," *J. Rob. Syst.* **6**(6), 769–798 (1989).
4. B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *J. Intell. Rob. Syst.* **3**(3), 201–212 (1990).
5. E. Tatlicioglu, M. L. McIntyre, D. M. Dawson and I. D. Walker, "Adaptive non-linear tracking control of kinematically redundant robot manipulators," *Int. J. Rob. Autom.* **23**(2), 98–105 (2008).
6. W. E. Dixon, A. Behal, D. M. Dawson and S. Nagarkatti, *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach* (Birkhauser, Boston, MA, 2003).
7. H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Rob. Autom.* **5**(4), 472–490 (1989).
8. P. Hsu, J. Hauser and S. Sastry, "Dynamic control of redundant manipulators," *J. Rob5. Syst.* **6**(3), 133–148 (1989).
9. E. Zergeroglu, D. M. Dawson, I. D. Walker and A. Behal, "Nonlinear tracking control of kinematically redundant robot manipulators," *Proceedings of the American Control Conference*, Chicago, IL (2000) pp. 2513–2517.
10. Z. Peng and N. Adachi, "Compliant motion control of kinematically redundant manipulators," *IEEE Trans. Rob. Autom.* **9**(6), 831–837 (1993).
11. R. Colbaugh and K. Glass, "Robust adaptive control of redundant manipulators," *J. Intell. Rob. Syst.* **14**(1), 68–88 (1995).
12. O. Khatib, "Dynamic control of manipulators in operational space," *IFTOMM Cong. Theory of Machines and Mechanisms*, New Delhi, India (1983) pp. 1–10.
13. S. Luo and S. Ahmad, "Adaptive control of kinematically redundant robots," *IMA J. Math. Control Inf.* **14**, 225–253 (1997).
14. F. L. Lewis, D. M. Dawson and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice* (Marcel Dekker Inc., New York, NY, 2004).
15. M. S. de Queiroz, D. M. Dawson, S. P. Nagarkatti and F. Zhang, *Lyapunov-Based Control of Mechanical Systems* (Birkhauser, Boston, MA, 1999).
16. M. Krstic, I. Kanellakopoulos and P. Kokotovic, *Nonlinear and Adaptive Control Design* (John Wiley and Sons, New York, NY, 1995).
17. P. Ioannou and J. Sun, *Robust Adaptive Control* (Prentice-Hall, Englewood Cliffs, NJ, 1996).
18. E. Tatlicioglu, *Control of Nonlinear Mechatronic Systems.* (VDM Verlag Dr. Mueller e.K., Germany, 2008).

Appendix A: Filter Development

From Eqs. (1), (3), (31) and (32) the following can be obtained:

$$\dot{\tau}_f + \alpha\tau_f = \dot{Y}_f\phi + \alpha Y_f\phi. \qquad \text{(A 1)}$$

The expression in Eq. (A1) can be rewritten as

$$(s+\alpha)\tau_f = (s+\alpha)Y_f\phi, \qquad \text{(A 2)}$$

where s is the Laplace variable. From Eq. (A2) the following can be obtained[15]:

$$\tau_f = Y_f\phi, \qquad \text{(A 3)}$$

where the initial condition information defined in Eqs. (31) and (32) were utilized.

Appendix B: Implementable Form of the Regression Matrix

In order to obtain an implementable form of Eq. (31) the entries of $Y(\theta, \dot{\theta}, \ddot{\theta})$ will be written in the following form:

$$Y_{ij}(\theta, \dot{\theta}, \ddot{\theta}) \overset{\triangle}{=} B_{ij}^T(\theta)\ddot{\theta} + A_{ij}(\theta, \dot{\theta}), \qquad \text{(B 1)}$$

where $B_{ij}^T(\theta) \in \mathbb{R}^{1\times n}$ and $A_{ij}(\theta, \dot{\theta}) \in \mathbb{R}$ for $\forall i = 1, \ldots, n$ and $\forall j = 1, \ldots, p$. An auxiliary filter signal, denoted by $P_{ij}(t) \in \mathbb{R}$, is designed as follows:

$$\dot{P}_{ij}(t) \overset{\triangle}{=} -\alpha Y_{f_{ij}} - \dot{B}_{ij}^T(\theta)\dot{\theta} + A_{ij}(\theta, \dot{\theta}), \qquad \text{(B 2)}$$

$$P_{ij}(t_0) \overset{\triangle}{=} -B_{ij}^T(\theta(t_0))\dot{\theta}(t_0), \qquad \text{(B 3)}$$

where $Y_{f_{ij}}(t)\ \forall i, j$ are defined as follows:

$$Y_{f_{ij}} \triangleq P_{ij} + B_{ij}^T(\theta)\dot{\theta}. \qquad \text{(B 4)}$$

From Eqs. (B2)–(B4), it is clear that Eq. (31) is satisfied and $Y_{f_{ij}}(t)$ defined in Eq. (B4) can be implemented without measuring $\ddot{\theta}(t)$.

Appendix C: Proof of Theorem 1

The following non-negative function is introduced to analyse the stability of the task-space controller:

$$V_1 \triangleq \frac{1}{2}\tilde{\phi}^T\Gamma^{-1}\tilde{\phi}. \qquad \text{(C 1)}$$

The time derivative of the Lyapunov function in Eq. (C1) is given as follows:

$$\begin{aligned}\dot{V}_1 &= \frac{1}{2}\tilde{\phi}^T\dot{\Gamma}^{-1}\tilde{\phi} - \tilde{\phi}^T\Gamma^{-1}\dot{\hat{\phi}}\\ &= \frac{1}{2}\tilde{\phi}^T Y_f^T\hat{M}^{-T}\hat{M}^{-1}Y_f\tilde{\phi} - \tilde{\phi}^T Y_f^T\hat{M}^{-T}z\\ &= -\frac{1}{2}z^T z, \end{aligned} \qquad \text{(C 2)}$$

where Eqs. (34), (46) and (47) were used. After integrating Eq. (C2) the following expression can be obtained:

$$V_1(t_0) - V_1(\infty) = \frac{1}{2}\int_{t_0}^{\infty} z^T(\tau)z(\tau)\,\mathrm{d}\tau. \qquad \text{(C 3)}$$

From (C1), it is clear that $V_1(t_0) > 0$, and from (C3), it is easy to see that $V_1(t) \le V_1(t_0)$, then we can conclude that $V(t)$ is bounded, hence $z(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\tilde{\phi}(t) \in \mathcal{L}_\infty$. From (24), $\hat{\phi}(t) \in \mathcal{L}_\infty$ hence $\hat{M}(t)$ and $\hat{N}(t)$ are also bounded. Based on Remark 4, the matrix inverse of the estimate of $M(\theta)$ (i.e. $\hat{M}^{-1}(t)$) exists and is bounded. Remark 1 can be utilized to show that $J(\theta)$ and $J^+(\theta)$ are bounded. These boundedness statements can be utilized along with Eq. (41) to prove that $r(t)$ is bounded; hence, from Eq. (37), we can conclude that $e(t), \dot{e}(t) \in \mathcal{L}_\infty$. Since the desired trajectory and its time derivative are assumed to be bounded then from Eq. (17) and its time derivative we can prove that $x(t), \dot{x}(t) \in \mathcal{L}_\infty$. Based on Remark 1, $\theta(t)$ is bounded. The rest of the development requires the joint velocities to be bounded. From the proof of Theorem 2 (see Appendix D), we know that $\dot{e}_N(t) \in \mathcal{L}_\infty$ and from the proof of Theorem 3, we know that $g(t) \in \mathcal{L}_\infty$. Based on these facts, we can show that $(I_n - J^+J)\dot{\theta} \in \mathcal{L}_\infty$. After utilizing this along with Eq. (4) and the fact that $\dot{x}(t) \in \mathcal{L}_\infty$, we can prove that $\dot{\theta}(t) \in \mathcal{L}_\infty$. After utilizing the facts that $\theta(t), \dot{\theta}(t) \in \mathcal{L}_\infty$, we can conclude that $M(\theta)$ and $N(\theta, \dot{\theta})$ are bounded. By utilizing the above boundedness statements it is easy to show that $\dot{J}(\theta) \in \mathcal{L}_\infty$. From the development in Appendix B, we can show that $Y_f(t), \dot{Y}_f(t) \in \mathcal{L}_\infty$. Then from Eq. (46), it is clear that $\dot{\hat{\phi}}(t)$ is also bounded. Thus $\dot{\hat{M}}(t)$ and $\frac{\mathrm{d}}{\mathrm{d}t}(\hat{M}^{-1}(t))$ can be shown to be bounded. We can utilize the time derivative of Eq. (34) to prove that $\dot{z}(t) \in \mathcal{L}_\infty$. The above boundedness statements can be utilized along with (35) and (43) to show that $u_{\text{aux}}(t), u_2(t) \in \mathcal{L}_\infty$, thus, from Eq. (27), $u_1(t)$ is also bounded. From the proof of Theorem 3, we know that $g(t), \dot{g}(t) \in \mathcal{L}_\infty$. After utilizing these facts and the previous boundedness statements along with Eq. (54), we can prove that $\phi_N(t) \in \mathcal{L}_\infty$. Then from (26) and (51), it is clear that $\tau(t), \ddot{\theta}(t) \in \mathcal{L}_\infty$. Since $z(t) \in \mathcal{L}_2 \cap \mathcal{L}_\infty$ and $\dot{z}(t) \in \mathcal{L}_\infty$, we can conclude that $\|z(t)\| \to 0$ as $t \to \infty$. Then from Eq. (41), it is clear that $\|r(t)\| \to 0$ as $t \to \infty$; thus from (37), $\|e(t)\|$, $\|\dot{e}(t)\| \to 0$ as $t \to \infty$.

Appendix D: Proof of Theorem 2

Let $V_2(t) \in \mathbb{R}$ denote the following non-negative function:

$$V_2 \triangleq \frac{1}{2}p^{\mathrm{T}}p. \qquad \text{(D 1)}$$

The time derivative of (D1) is given as follows:

$$\dot{V}_2 = -k_n p^T p + p^T J^+(k_n J - \dot{J})p, \qquad \text{(D 2)}$$

where the dynamics of $\dot{p}(t)$ in (55) was utilized. To facilitate the subsequent development the following property is introduced

$$\begin{aligned} p^T J^+ &= \left(g - \dot{\theta} - \frac{1}{\alpha}z\right)^T (I_n - J^+J)^T J^+\\ &= \left(g - \dot{\theta} - \frac{1}{\alpha}z\right)^T (I_n - J^+J)J^+\\ &= 0_{1\times m}, \end{aligned} \qquad \text{(D 3)}$$

where Eqs. (49) and (52) were utilized. In view of (D3), (D2) can be written in the following simple form:

$$\dot{V}_2 = -k_n p^{\mathrm{T}} p. \qquad \text{(D 4)}$$

From (D1) and (D4), standard linear analysis techniques can be utilized to show that

$$p(t) = p(t_0)\exp(-k_n t), \qquad \text{(D 5)}$$

from which we can conclude that $\|p(t)\| \to 0$ exponentially fast. Then from (52) and the proof of Theorem 1, it is easy to see that $\|\dot{e}_N(t)\| \to 0$ as $t \to \infty$.