Motion Control

53

Chien Chern Cheah and Reza Haghighi

Contents

Abstract

Robot manipulators have been widely used in industrial automation. In many modern robot control applications, sensory information such as visual feedback is used to improve positioning accuracy and robustness to uncertainty. This chapter introduces basic concepts and design methods that are employed for

C.C. Cheah (✉) • R. Haghighi
School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
e-mail: ecccheah@ntu.edu.sg; reza.h@ntu.edu.sg

A.Y.C. Nee (ed.), *Handbook of Manufacturing Engineering and Technology*,
DOI 10.1007/978-1-4471-4670-4_93

motion control of robot manipulators with uncertainty. The chapter covers both basic methods in joint-space control and advance topics in sensory task-space control.

Introduction

Robotic manipulation has shown to be a key technology in factory automation. In order for a robot to perform some specific tasks, the robot is required to move according to the commands from a motion controller. Most motion control applications of robot manipulators can be categorized into two main classes. The first is the point-to-point motion control or set-point regulation where the robot is required to move from an initial position to a final desired position in the workspace. Pick-and-place operations are typical examples of the point-to-point motion control applications. Other examples that require set-point control are spot welding and hole drilling. The second type is trajectory tracking applications, such as arc welding, machining, and painting, where the robot has to follow a desired trajectory.

Robot manipulator Robot manipulator consists of rigid links connected by joints. One end of the manipulator is fixed to a base and an end effector or tool is connected to the other end. The vector space in which the joint displacements are defined is often referred to as the joint space, and the coordinates in which the manipulator task of the end effector is specified is referred to as the task space, which can be a Cartesian space or an image space depending on the task requirements. The motion control problem of a robot can be formulated either in the joint space (Kelly et al. 2005) or in the task space (Spong et al. 2006). In the joint-space control methodology, the desired position of the end effector is converted to a corresponding desired joint configuration by solving an inverse kinematic problem, and a feedback control law is designed so that the robot joints follow the desired joint position. To eliminate the problem of solving the inverse kinematics, the robot motion control problem can be directly formulated and designed in task space. A transformation matrix Transformation matrix or Jacobian matrix Jacobian matrix is used to transform the task-space feedback error to joint control inputs.

The most commonly used controllers in industrial applications are PD and PID controllers (Ziegler and Nichols 1942). The main advantages of such controllers are the simplicity and ease of implementation. However, the kinematics and dynamics of a robot manipulator are highly nonlinear with coupling between joints, and hence, the linear control theory cannot be applied directly to design PD or PID controllers for a robot manipulator. By exploring physical properties of the robot dynamics, Takegaki and Arimoto (Takegaki and Arimoto 1981) first showed using Lyapunov method (Slotine and Li 1991) that a simple PD controller with gravity compensation is effective for set-point control of a robot manipulator Robot manipulator. The result was an important landmark in robot control theory. Inspired by the original work (Takegaki and Arimoto 1981), much progress has been made in understanding the robot motion control problem, and various control methods have been developed for a robot manipulator (Arimoto and Miyazaki 1984, 1985;

Arimoto et al. 1994; Ortega et al. 1995; Arimoto 1994; Wen and Bayard 1988; Niemeyer and Slotine 1991; Berghuis et al. 1993; Cheah et al. 1998, 2004, 2007, 2010; Cheah 2003; Dixon 2007; Liang et al. 2010; Braganza et al. 2005; Wang and Xie 2009; Garcia-Rodriguez and Parra-Vega 2012; Cheah and Liaw 2005; Kelly 1997; Wang et al. 2007).

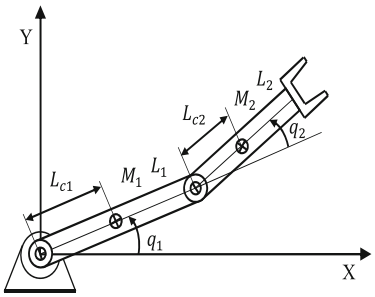
This chapter focuses on motion control methods of robot manipulators that were developed based on the Lyapunov method. In particular, several set-point and adaptive tracking controllers are presented in both the joint space and task space, for a robot manipulator with uncertainty. The robot kinematics and dynamics are nonlinear with coupling between joints, and a good understanding of the structure and properties of the models is essential for the design of simple and effective controllers. The dynamics and kinematics of robot manipulators and their basic properties are first introduced in section "Dynamics and Kinematics of Robot Manipulators" and several examples are given to illustrate the properties. The next two sections of this chapter present the standard motion controllers in robotics, which serve as foundation works for the design of most robot controllers based on the Lyapunov method. Section "Set-Point Control by PD Plus Gravity Controller" introduces the set-point controllers based on the PD plus gravity control strategy. Section "Adaptive Control of Robot Manipulators" presents adaptive control methodology for tracking control of robot manipulators. Recent advances in sensing technology has led to the research and development of sensory task-space feedback control laws for robot manipulators. The use of task-space sensory feedback information such as visual information improves the endpoint accuracy in the presence of uncertainty. Section "Approximate Jacobian Set-Point Control" presents the basic task-space sensory feedback control problem of a robot manipulator with kinematic and dynamic uncertainty for set-point control applications. Moreover, the results are extended to deal with uncertain gravitational force. Section "Adaptive Jacobian Tracking Control" presents the adaptive Jacobian controller for task-space sensory feedback tracking control applications. Simulation results are presented in section "Simulation Results." A brief review of the basic concepts and theories for stability analysis of nonlinear systems is also provided in the Appendix.

Dynamics and Kinematics of Robot Manipulators

In this section, the dynamic and kinematic equations of robot manipulators Robot manipulator are introduced and the properties which constitute the basis of controller design in this chapter are presented.

Dynamic Equation of Robots

For a robot manipulator of n DOF with the joint coordinate $q = [q_1, \ldots, q_n]^T$, the equation of motion using Lagrange approach takes the following form:



M_1, M_2: mass of the first and second link

L_1, L_2: length of the first and second link

L_{c1}, L_{c2}: distance between the center of the gravity and the joint

Fig. 1 A two-link robot manipulator

$$\sum_{j=1}^{n} M_{ij}\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n} \Gamma_{ijk}\dot{q}_j\dot{q}_k + D_{ii}(q)\dot{q}_i + g_i(q) = u_i, \quad i = 1, \ldots, n \tag{1}$$

such that Γ_{kji} is called the Christoffel symbols and it is defined as

$$\Gamma_{ijk} = \frac{1}{2}\left\{\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} - \frac{\partial M_{kj}(q)}{\partial q_i}\right\}. \tag{2}$$

Equation 1 can be expressed in the following compact form:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + g(q) = u \tag{3}$$

where $M(q) \in R^{n\times n}$ is the inertia matrix, $C(q,\dot{q})$ is the matrix of Coriolis and centripetal terms, $D(q) \in R^{n\times n}$ is the matrix of damping coefficients, $g(q) \in R^n$ denotes the gravitational force, and $u \in R^n$ is the vector of control input. The following example is presented for better understanding of the dynamic equation expressed in Eq. 3.

Example 1 Consider a two-link planar robot manipulator as depicted in Fig. 1.

The dynamic equation of the robot is expressed as follows:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} D_{11} & 0 \\ 0 & D_{22} \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = u \tag{4}$$

where parameters M_{ij}, C_{ij}, and g_i for $i, j = 1, 2$ are expressed in Appendix 2. Parameters D_{11} and D_{22} are positive constants that represent damping coefficients for each joint.

Some properties of dynamic equation of robot manipulators expressed by Eq. 3 are as follows:

Property 1 *The inertia matrix $M(q)$ is symmetric and positive definite for all $q \in R^n$.*

Property 2 *The matrix* $\dot{M}(q) - 2C(q,\dot{q})$ *is skew symmetric so that*

$$y^T\left\{\dot{M}(q) - 2C(q,\dot{q})\right\}y = 0 \quad \textit{for all}\, y \in R^n. \tag{5}$$

To show property 2, the elements of matrix $C(q,\dot{q})$ are obtained by using Eqs. 1 and 2:

$$C_{ij}(q,\dot{q}) = \sum_{k=1}^{n} \frac{1}{2}\left\{\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} - \frac{\partial M_{kj}(q)}{\partial q_i}\right\}\dot{q}_k \tag{6}$$

Therefore, the elements of matrix $\dot{M}(q) - 2C(q,\dot{q})$ can be expressed as

$$\begin{aligned}\left(\dot{M}(q) - 2C(q,\dot{q})\right)_{ij} &= \sum_{k=1}^{n}\frac{\partial M_{ij}(q)}{\partial q_k}\dot{q}_k - \left\{\frac{\partial M_{ij}(q)}{\partial q_k} + \frac{\partial M_{ik}(q)}{\partial q_j} - \frac{\partial M_{kj}(q)}{\partial q_i}\right\}\dot{q}_k \\ &= \sum_{k=1}^{n}\left\{-\frac{\partial M_{ik}(q)}{\partial q_j} + \frac{\partial M_{kj}(q)}{\partial q_i}\right\}\dot{q}_k.\end{aligned}$$

Switching i and j and considering the symmetricity of matrix $M(q)$ yield

$$\begin{aligned}\left(\dot{M}(q) - 2C(q,\dot{q})\right)_{ij} &= -\sum_{k=1}^{n}\left\{-\frac{\partial M_{jk}(q)}{\partial q_i} + \frac{\partial M_{ki}(q)}{\partial q_j}\right\}\dot{q}_k \\ &= -\left(\dot{M}(q) - 2C(q,\dot{q})\right)_{ji},\end{aligned}$$

which shows that the matrix $\left(\dot{M}(q) - 2C(q,\dot{q})\right)$ is skew symmetric.

Property 3 *The dynamic equation* Eq. 3 *can be linearly parameterized with respect to the constant parameters, as follows*:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + g(q) = Y(q,\dot{q},\dot{q},\ddot{q})\theta_d \tag{7}$$

where $Y(\cdot) \in R^{n\times p}$ *is the known regressor matrix and* θ_d *is the unknown parameter vector.*

The following example is presented to illustrate properties 1–3:

Example 2 Consider the two-link robot manipulator depicted in Fig. 1. The inertia matrix is expressed by Eq. 113 in the Appendix. To show the positive definiteness of the inertia matrix, it is sufficient to show that all leading principal minors of the matrix are positive. Hence, the following inequalities must hold:

$$\begin{cases} I: & \frac{4}{3}M_1L_{c1}^2 + \frac{4}{3}M_2L_{c2}^2 + M_2L_1^2 + 2M_2L_1L_{c2}\cos(q_2) > 0 \\ II: & \det(M) > 0 \end{cases}. \tag{8}$$

For inequality I, the worst case is $\cos(q_2) = -1$; then it gives

$$\frac{4}{3}M_1L_{c1}^2 + \frac{1}{3}M_2L_{c2}^2 + M_2(L_1 - L_{c2})^2 > 0. \tag{9}$$

For the second inequality, the determinant of $M(q)$ is computed as follows:

$$\frac{16}{9}M_1M_2L_{c1}^2L_{c2}^2 + M_2^2L_1^2L_{c2}^2\left(\frac{4}{3} - \cos^2(q_2)\right) > 0 \tag{10}$$

which yields the positive definiteness of the inertia matrix.

To show the skew-symmetric property of the matrix $\dot{M}(q) - 2C(q,\dot{q})$, let us form the matrix $\dot{M}(q)$:

$$\dot{M}(q) = \begin{bmatrix} -2M_2L_1L_{c2}\dot{q}_2\sin(q_2) & -M_2L_1L_{c2}\dot{q}_2\sin(q_2) \\ -M_2L_1L_{c2}\dot{q}_2\sin(q_2) & 0 \end{bmatrix} \tag{11}$$

Hence, the matrix $\dot{M}(q) - 2C(q,\dot{q})$ can be expressed as

$$\dot{M}(q) - 2C(q,\dot{q}) = \begin{bmatrix} 0 & \Psi \\ -\Psi & 0 \end{bmatrix} \tag{12}$$

where $\Psi = 2M_2L_1L_{c2}\dot{q}_1\sin(q_2) + M_2L_1L_{c2}\dot{q}_2\sin(q_2)$. It can be seen from Eq. 12 that the matrix $\dot{M}(q) - 2C(q,\dot{q})$ is skew symmetric.

To show the property Eq. 3, the elements of known regressor matrix $Y \in R^{(2\times 7)}$ are given as

$$\begin{aligned}
&Y_{11} = \ddot{q}_1, \quad Y_{12} = \ddot{q}_2 \\
&Y_{13} = (2\ddot{q}_1 + \ddot{q}_2)\cos(q_2) - (2\dot{q}_1\dot{q}_2 + \dot{q}_2^2)\sin(q_2) \\
&Y_{14} = \dot{q}_1, \quad Y_{15} = 0, \quad Y_{16} = \cos(q_1), \quad Y_{17} = \cos(q_1 + q_2) \\
&Y_{21} = 0, \quad Y_{22} = \ddot{q}_1 + \ddot{q}_2 \\
&Y_{23} = \ddot{q}_1\cos(q_2) + \dot{q}_1^2\sin(q_2) \\
&Y_{24} = 0, \quad Y_{25} = \dot{q}_2, \quad Y_{26} = 0, \quad Y_{27} = \cos(q_1 + q_2),
\end{aligned}$$

and the unknown parameter vector is expressed as follows:

$$\theta_d = \begin{bmatrix} \frac{4}{3}M_1L_{c1}^2 + \frac{4}{3}M_2L_{c2}^2 + M_2L_1^2 \\ \frac{4}{3}M_2L_{c2}^2 \\ M_2L_1L_{c2} \\ D_{11} \\ D_{22} \\ (M_1L_{c1} + M_2L_1)g \\ M_2L_{c2}g \end{bmatrix}.$$

Kinematic Equation Kinematic equation of Robots

The kinematic model represents the relationship between the joint angles and the end-effector position. Hence, the kinematic model connects the task space to joint space as follows:

$$x = h(q) \tag{13}$$

where $x \in R^m$ is the task-space vector and $h(\cdot) \in R^n \rightarrow R^m$ is a transformation describing the relation between the joint space and the task space. The relationship between task-space and joint-space velocities is given as

$$\dot{x} = J(q)\dot{q} \tag{14}$$

where $J(q) \in R^{m \times n}$ is the which provides a transformation from the joint space to the task space.

Remark 1 The task-space vector can either be defined in a Cartesian space or an image space. If cameras are used to measure the end-effector position, the task space is defined as an image space. Let x denote the vector of image feature parameters; the image velocity vector $\dot{x}$ is related to the joint velocity vector $\dot{q}$ as

$$\dot{x} = J_I(q)J_e(q)\dot{q} \tag{15}$$

where $J_I(q)$ is the image Jacobian matrix from the Cartesian space to the image space (Hutchinson et al. 1996) and $J_e(q)$ is the manipulator Jacobian matrix from the joint space to the Cartesian space. Therefore, the overall Jacobian from joint space to task space is $J(q) = J_I(q)J_e(q)$. If a position sensor is used to measure the end-effector position directly, the task-space vector x is defined as a Cartesian space and hence $J(q) = J_e(q)$.

A property of the kinematic equation Kinematic equation described by Eq. 14 is stated as follows:

Property 4 *The right-hand side of* Eq. 14 *is linear in a set of constant kinematic parameters* $\theta_k = (\theta_{k1}, \cdots, \theta_{kq})^T$, *such as link lengths and joint offsets. Hence,* Eq. 14 *can be expressed as*

$$\dot{x} = J(q)\dot{q} = Y_k(q, \dot{q})\theta_k, \tag{16}$$

where $Y_k(q, \dot{q}) \in R^{n \times q}$ *is called the kinematic regressor matrix.*

Example 3 Consider the two-link robot manipulator as depicted in Fig. 1. The kinematics of the robot can be expressed as follows:

$$x = \begin{bmatrix} L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) \\ L_1 \sin(q_1) + L_2 \sin(q_1 + q_2) \end{bmatrix}. \tag{17}$$

The task-space velocity can be obtained by differentiation Eq. 17 with respect to time, as

$$\dot{x} = \underbrace{\begin{bmatrix} -L_1 \sin(q_1) - L_2 \sin(q_1 + q_2) & -L_2 \sin(q_1 + q_2) \\ L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) & L_2 \cos(q_1 + q_2) \end{bmatrix}}_{J(q)} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}. \tag{18}$$

Equation 18 can be written as follows:

$$\dot{x} = \underbrace{\begin{bmatrix} -\sin(q_1)\dot{q}_1 & -\sin(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \\ \cos(q_1)\dot{q}_1 & \cos(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \end{bmatrix}}_{Y_k(q,\dot{q})} \underbrace{\begin{bmatrix} L_1 \\ L_2 \end{bmatrix}}_{\theta_k}. \tag{19}$$

Set-Point Control by PD Plus Gravity Controller

This section considers the set-point control problems or point-to-point control problems in which the robot is required to move from one point to another point without controlling the path taken by the robot between the two points. A simple and useful set-point controller for motion control is the PD plus gravity controller. This control strategy employs feedback of joint angles, velocity, and compensation of gravitational force as depicted in Fig. 2.

The feedback control law is expressed in the following form:

$$u = -K_v\dot{q} - K_p\tilde{q} + g(q) \tag{20}$$

where K_p is the proportional gain matrix, K_v is the derivative gain matrix, and $\tilde{q} = q - q_d$ such that q_d represents a constant desired position. Both K_p and K_v are positive diagonal matrices. The closed-loop system can be obtained by substituting Eq. 20 into the dynamic Eq. 3:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + K_v\dot{q} + K_p\tilde{q} = 0. \tag{21}$$

The stability of the closed-loop system is examined by using the Lyapunov-based stability analysis. Consider the following Lyapunov function candidate as

$$V = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\tilde{q}^T K_p \tilde{q}. \tag{22}$$

Differentiating the Lyapunov-like candidate with respect to time gives

$$\dot{V} = \dot{q}^T M(q)\ddot{q} + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \dot{q}^T K_p \tilde{q}. \tag{23}$$

Substituting $M(q)\ddot{q}$ from the closed-loop Eq. 21 yields

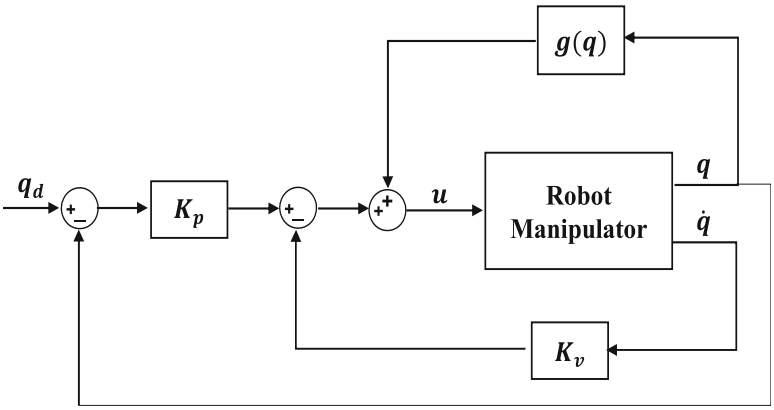


Fig. 2 A block diagram of the PD plus gravity controller

$$\dot{V} = \frac{1}{2}\dot{q}^T\left(\dot{M}(q) - 2C(q,\dot{q})\right)\dot{q} - \dot{q}^T D(q)\dot{q} \\ - \dot{q}^T K_v \dot{q} - \dot{q}^T K_p \tilde{q} + \dot{q}^T K_p \tilde{q}. \tag{24}$$

Considering the skew-symmetric property of the matrix $\dot{M}(q) - 2C(q,\dot{q})$, $\dot{V}$ is simplified as follows:

$$\dot{V} = -\dot{q}^T D(q)\dot{q} - \dot{q}^T K_v \dot{q} \leq 0. \tag{25}$$

Using LaSalle's invariance principle (see lemma A1 in the Appendix) yields

$$\dot{V} = 0 \Rightarrow \dot{q}^T\{D(q) + K_v\}\dot{q} = 0 \Leftrightarrow \dot{q} = 0. \tag{26}$$

From the closed-loop Eq. 21, one can conclude that

$$K_p\tilde{q} = 0 \Rightarrow \tilde{q} = 0 \tag{27}$$

yields $q \rightarrow q_d$ as $t \rightarrow \infty$.

In most robotic applications, the end effector of a robot manipulator is required to move to a desired position in the task space. This can be obtained by either solving the inverse kinematic problem to find the corresponding desired joint angles or developing task-space control methodology directly by using the task-space error. For task-space control, the Jacobian matrix is used to transform the task-space errors to joint control inputs. In the following, a task-space set-point control approach is presented. Consider the following feedback control law for the task-space regulation:

$$u = -J^T(q)\left(K_v\dot{x} - K_p\tilde{x}\right) + g(q) \tag{28}$$

where $\tilde{x} = x - x_d$ such that x_d is the desired position of the end effector. A block diagram of the task-space regulator described by Eq. 28 is illustrated in Fig. 3.

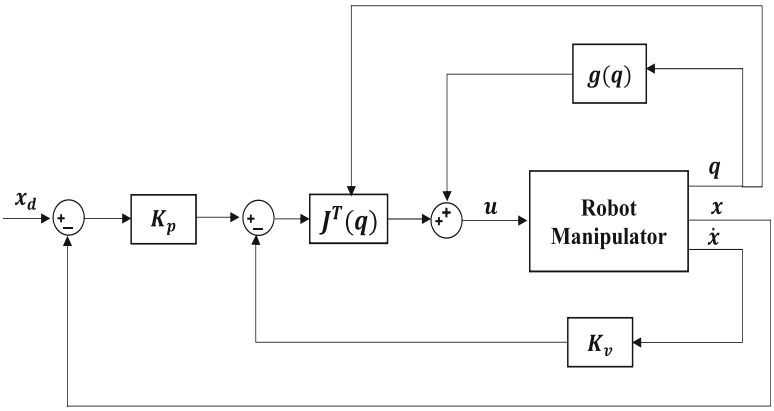


Fig. 3 A block diagram of the task-space PD plus gravity control system

Substituting the feedback law Eq. 28 into the dynamic Eq. 3 yields the following closed-loop system:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + J^T(q)\left(K_v\dot{x} - K_p\tilde{x}\right) = 0. \tag{29}$$

The following Lyapunov-like candidate can be considered for the stability analysis of the task-space regulation problem:

$$V = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}\tilde{x}^T K_p\tilde{x}. \tag{30}$$

Differentiating Eq. 30 with respect to time and substituting Eq. 29 into it yield

$$\dot{V} = -\dot{q}^T D(q)\dot{q} - \dot{q}^T J^T(q)K_v\dot{q} - \dot{q}^T J^T(q)K_p\tilde{x} + \dot{x}^T K_p\tilde{x} - \dot{q}^T D(q)\dot{q} - \dot{x}^T K_v\dot{x} \leq 0. \tag{31}$$

If the Jacobian matrix is of full rank, by using LaSalle's invariance principle (see lemma A1 in the Appendix), it is obtained

$$\dot{V} = 0 \Rightarrow \dot{q}^T D(q)\dot{q} + \dot{x}^T K_v\dot{x} \Leftrightarrow \begin{cases} \dot{q} = 0 \\ \dot{x} = 0 \end{cases}. \tag{32}$$

From the closed-loop Eq. 29, one can conclude that

$$J^T(q)K_p\tilde{x} = 0 \Rightarrow \tilde{x} = 0. \tag{33}$$

Hence, the position of the end effector converges to the desired position such as $x \to x_d$ as $t \to \infty$.

Remark 2 It was first shown in (Takegaki and Arimoto 1981) using the Lyapunov method that the simple PD controller with gravity compensation Eq. 20 is effective

for set-point control of a robot manipulator Robot manipulator. The gravity compensation term can also be computed off-line by using the desired position (Tomei 1991; Kelly 1993; Arimoto 1996):

$$u = -K_v\dot{q} - K_p\tilde{q} + g(q_d). \tag{34}$$

A main drawback of the PD plus gravity controller is that an exact knowledge of the gravity force of the robot manipulator is required. The existence of uncertainty in modeling the gravitational term results in steady-state position error. One way to alleviate the problem is to increase the P control gain which may lead to saturation of the control torques. A common practice to remove the steady-state error is to add an integral term (Arimoto and Miyazaki 1984; Arimoto 1996).

$$u = -K_v\dot{q} - K_p\tilde{q} - K_I\int \tilde{q}dt \tag{35}$$

where K_I is a symmetric positive definite matrix. The closed-loop system of the linear *PID* robot control system is asymptotically stable in a local sense. To achieve global asymptotic stability, a saturated or bounded nonlinear function of the position error can be used (Arimoto et al. 1994):

$$u = -K_v\dot{q} - K_p\tilde{q} - K_I\int s(\tilde{q})dt \tag{36}$$

where $s(.)$ is the saturation function. Other regulators for robot manipulators without using the gravitational term can be found in (Ortega et al. 1995; Kelly 1998). If the structure of the gravity force is known, adaptive PD controller with gravity regressor can also be used to eliminate the steady-state position error in the presence of uncertain parameters (Tomei 1991; Arimoto 1996):

$$u = -K_v\dot{q} - K_p\tilde{q} + Z(q)\hat{\varphi} \tag{37}$$

with the parameter update law

$$\dot{\hat{\varphi}} = -LZ^T(q)(\dot{q} + \alpha s(\tilde{q})) \tag{38}$$

where α is a positive constant. The gravity regressor matrix can be computed off-line by using the desired position (Kelly 1993).

Adaptive Control of Robot Manipulators

For tracking control applications, the manipulator is required to follow a desired time-varying trajectory. The simple PD controllers in the previous section are usually not effective for tracking control, especially for high-speed manipulation tasks. In this section, adaptive control method of robot manipulators is presented for tracking control applications.

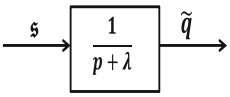


Fig. 4 Filter-like representation of Eq. 40 ($p = (d/dt)$ is the Laplace operator)

To design the adaptive control, a reference vector is defined as follows:

$$\dot{q}_r = \dot{q}_d - \lambda\tilde{q} \tag{39}$$

where λ is a positive constant. Using the reference vector Eq. 39, a sliding vector is defined as follows:

$$\mathfrak{s} = \dot{q} - \dot{q}_r = \dot{\tilde{q}} + \lambda\tilde{q}. \tag{40}$$

In fact, Eq. 40 represents a low-pass filter which is also depicted in Fig. 4. Hence, $\tilde{q}$ can be obtained as

$$\tilde{q}(t) = \int_0^t e^{-\lambda(t-\tau)}\mathfrak{s}(\tau)d\tau. \tag{41}$$

For a bounded sliding vector, i.e., $||\mathfrak{s}|| \leq \Phi$, then it yields

$$||\tilde{q}(t)|| \leq \Phi\int_0^t e^{-\lambda(t-\tau)}\mathfrak{s}(\tau)d\tau \leq \frac{\Phi}{\lambda}. \tag{42}$$

Therefore, inequality Eq. 42 shows that increasing λ leads to smaller tracking error.

By differentiating Eq. 40 with respect to time, $\dot{\mathfrak{s}}$ is obtained as follows:

$$\dot{\mathfrak{s}} = \ddot{q} - \ddot{q}_r = \ddot{\tilde{q}} + \lambda\dot{\tilde{q}}. \tag{43}$$

Substituting Eqs. 40 and 43 into Eq. 3 and using property 3, the following equation is obtained:

$$M(q)\dot{\mathfrak{s}} + C(q,\dot{q})\mathfrak{s} + D(q)\mathfrak{s} + \mathrm{Y}(q,\dot{q},\dot{q}_r,\ddot{q}_r)\theta_d = u \tag{44}$$

where $\mathrm{Y}(q,\dot{q},\dot{q}_r,\ddot{q}_r)$ is a known regressor matrix and θ_d is an unknown parameter vector, which are defined as follows:

$$\mathrm{Y}(q,\dot{q},\dot{q}_r,\ddot{q}_r)\theta_d = M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + D(q)\dot{q}_r + g(q). \tag{45}$$

In fact, adaptive control strategies for a nonlinear system are mainly based on the property of linear parameterization of the nonlinear dynamics with respect to unknown parameters.

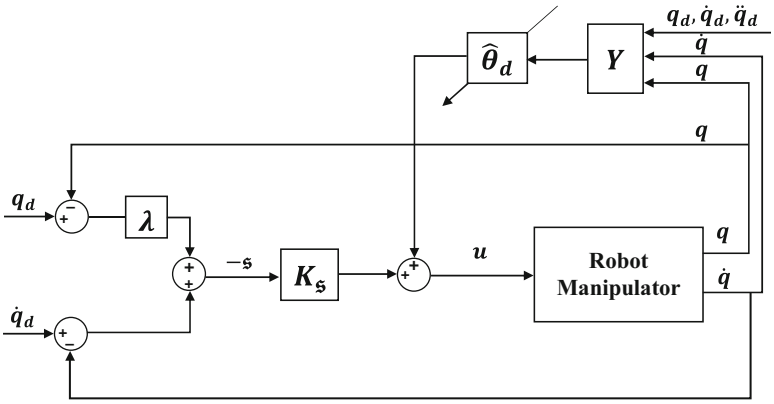


Fig. 5 A block diagram of the adaptive feedback law (46)

The adaptive law is expressed as follows:

$$u = -K_\mathfrak{s}\mathfrak{s} + \mathrm{Y}\big(q, \dot{q}, \dot{q}_r, \ddot{q}_r\big)\hat{\theta}_d \tag{46}$$

where $K_\mathfrak{s}$ is a positive definite matrix and $\hat{\theta}_d$ is the vector of estimated parameters, which is updated based on the following update law:

$$\dot{\hat{\theta}}_d = -L\mathrm{Y}^T(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\mathfrak{s} \tag{47}$$

such that L is a symmetric positive definite matrix. A block diagram of the adaptive law Eq. 46 is illustrated in Fig. 5.

The closed-loop equation for the robot manipulator is obtained by substituting the adaptive feedback law Eq. 46 into the dynamic Eq. 3, as follows:

$$M(q)\dot{\mathfrak{s}} + C(q, \dot{q})\mathfrak{s} + D(q)\mathfrak{s} + K_\mathfrak{s}\mathfrak{s} + \mathrm{Y}(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\Delta\theta_d = 0 \tag{48}$$

where $\Delta\theta_d = \theta_d - \hat{\theta}_d$. To examine the stability of the closed-loop system described by the Eq. 48, the following Lyapunov-like candidate is proposed:

$$V = \frac{1}{2}\mathfrak{s}^T M(q)\mathfrak{s} + \frac{1}{2}\Delta\theta_d^T L^{-1}\Delta\theta_d. \tag{49}$$

By differentiating the Lyapunov-like candidate with respect to time, substituting the closed-loop Eq. 48 into it yields

$$\begin{aligned}\dot{V} &= \mathfrak{s}^T M(q)\dot{\mathfrak{s}} + \frac{1}{2}\mathfrak{s}^T\dot{M}(q)\mathfrak{s} + \dot{\hat{\theta}}_d L^{-1}\Delta\theta_d \\ &= -\mathfrak{s}^T D(q)\mathfrak{s} - \mathfrak{s}^T K_\mathfrak{s}\mathfrak{s} - \mathfrak{s}^T\mathrm{Y}(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\Delta\theta_d \\ &\quad + \frac{1}{2}\mathfrak{s}^T\big\{\dot{M}(q) - 2C(q, \dot{q})\big\}\mathfrak{s} + \dot{\hat{\theta}}_d^T L^{-1}\Delta\theta_d.\end{aligned} \tag{50}$$

Considering the skew symmetricity of the matrix $\dot{M}(q) - 2C(q,\dot{q})$ and substituting Eq. 47 into Eq. 50, $\dot{V}$ is simplified as

$$\dot{V} = -\mathfrak{s}^T D(q)\mathfrak{s} - \mathfrak{s}^T K_s \mathfrak{s} \leq 0. \tag{51}$$

Since $M(q)$ and L are positive definite, V is positive definite in $\mathfrak{s}$ and $\Delta\theta_d$. Therefore, $\mathfrak{s}$ and $\Delta\theta_d$ are bounded. Hence, from closed-loop Eq. 48, one can conclude the boundedness of $\dot{\mathfrak{s}}$, and this leads to the boundedness of $\ddot{V}$, which means that $\dot{V}$ is uniformly continuous. By Barbalat's lemma (see lemma A2), it then follows $\dot{V}$ goes to zero as $t \to \infty$, so it can be concluded that s converges to zero. The convergence of $\mathfrak{s}$ yields the convergence of $\tilde{q}$ and $\dot{\tilde{q}}$ to zero.

Remark 3 The globally tracking convergent adaptive controller Eq. 46 was derived in (Slotine and Li 1987) for a robot with dynamic uncertainty. The dynamic regressor matrix can also be computed off-line by using the desired trajectory (Arimoto 1996; Sadegh and Horowitz 1990). An adaptive controller with off-line computed dynamic regressor is as follows (Arimoto 1996):

$$u = -k_p\tilde{q} - K_v\dot{\tilde{q}} + Y_d(q_d,\dot{q}_d,\dot{q}_d,\ddot{q}_d)\hat{\theta}_d \tag{52}$$

with the parameter update law as

$$\dot{\hat{\theta}}_d = -LY_d^T(q_d,\dot{q}_d,\dot{q}_d,\ddot{q}_d)\{\dot{\tilde{q}} + \alpha s(\tilde{q})\} \tag{53}$$

where α is a positive constant and $s(.)$ is the saturation function. Many other robot adaptive controllers have also been proposed for a robot with dynamic uncertainty (Wen and Bayard 1988; Niemeyer and Slotine 1991; Berghuis et al. 1993; Paden and Panja 1988; Ortega and Spong 1989; Craig et al. 1987; Middleton and Goodwin 1988; Koditschek 1987; Lee and Khalil 1997).

This section focuses only on adaptive control of robot manipulators. Other tracking control methods for robot manipulators with dynamic uncertainty include robust control (Slotine 1985; Spong 1992; Abdallah et al. 1991), learning control (Arimoto 1996; Arimoto et al. 1984), and neural network control (Lewis 1996).

Remark 4 Since the closed-loop system Eq. 48 is directly dependant on time due to the presence of the time-varying trajectory $q_d(t)$, LaSalle's invariance principle cannot be utilized to show the stability of the system.

Remark 5 A key point in adaptive control is that the tracking error will converge regardless of parameter convergence. That is, one does not need parameter convergence for tracking error convergence. However, parameter convergence can be obtained if the persistence of excitation condition is satisfied.

Remark 6 In many practical applications, the physical parameters are bounded by prescribed upper and lower limits. Using these bounds as a priori information, the parameter update law can be modified in a way that the estimates are constrained to a specified region. In this regard, a commonly used method is the standard projection algorithm (Ioannou and Sun 1996).

Approximate Jacobian Set-Point Control

In many modern control applications of robot manipulators, task-space sensory feedback information such as visual information is used to monitor the position of the end effector. By using sensory feedback, the control systems are robust to various modeling uncertainties. This gives the robot a high degree of flexibility in dealing with unforeseen changes and uncertainties. When a robot picks up several tools of different dimensions, unknown orientations, or gripping points, the kinematics changes and is difficult to derive exactly. In the presence of kinematic uncertainty, inverse kinematic algorithms cannot be used to derive the desired position in the joint space. This section presents a simple approximate Jacobian set-point controller for a robot manipulator with uncertain kinematics by using task-space sensory feedback of the position of the end effector. Though the position of the end effector can be measured by task-space sensors, the uncertainty of the Jacobian matrix poses a challenging robot control problem.

First the quasi-natural potential function $S_i(\phi)$ is defined which will be used in developing the feedback law.

Definition 1 *Consider the scalar function $S_i(\phi)$ and its derivative $s_i(\phi)$ as depicted in* Fig. 6 *, with the following properties*:

1. $S_i(\phi) > 0$ *for* $\phi \in R - \{0\}$ *and* $S_i(0) = 0$.
2. $S_i(\phi)$ *is twice continuously differentiable, and the derivative* $s_i(\phi) = \frac{dS_i(\phi)}{d\phi}$ *is strictly increasing in* ϕ *for* $|\phi| < \gamma_i$ *with some* γ_i *and saturated for* $|\phi| \geq \gamma_i$, *i.e.,* $s_i(\phi) = \pm s_i$ *for* $\phi \geq +\gamma_i$ *and* $\phi \leq -\gamma_i$, *respectively, where* s_i *is a positive constant*.
3. *There are constants* $\overline{c}_i > 0, d_i > 0, \overline{d}_i (> d_i) > 0$ *such that*

$$\overline{d}_i s_i^2(\phi) \geq \phi s_i(\phi) \geq d_i s_i^2(\phi) > 0, \quad S_i(\phi) \geq \overline{c}_i s_i^2(\phi), \tag{54}$$

for $\phi \neq 0$

The approximate Jacobian set-point controller is expressed as follows:

$$u = -\hat{J}^T(q)\left(K_p s(\tilde{x}) + K_v \dot{x}\right) + g(q) \tag{55}$$

where $\hat{J}(q) \in R^{m \times n}$ is the estimation of the Jacobian matrix; $J(q)$, K_p, K_v are positive definite diagonal feedback gains for the position and velocity, respectively; and

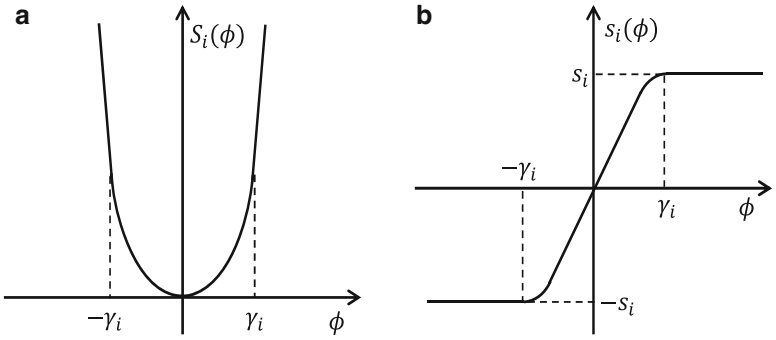


Fig. 6 (**a**) Quasi-natural potential $S_i(\phi)$, (**b**) derivative of $S_i(\phi)$

$s_i(\cdot), i = 1, \cdots, m$ are saturated functions defined in definition 1. Here, it is assumed that $\hat{J}^T(q)$ is chosen so that

$$||J^T(q) - \hat{J}^T(q)|| \leq \gamma, \tag{56}$$

where γ is a positive constant to be defined later. Substituting the feedback law Eq. 55 into the dynamic Eq. 3 gives the closed-loop equation as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q} + \hat{J}^T(q)\left(K_p s(\tilde{x}) + K_v \dot{x}\right) = 0. \tag{57}$$

To examine the stability of the closed-loop system, consider the following Lyapunov-like candidate:

$$\begin{aligned} V = &\frac{1}{2}\dot{q}^T M(q)\dot{q} + \alpha \dot{q}^T M(q)\hat{J}^{\dagger}(q)s(\tilde{x}) \\ &+ \sum_{i=1}^{m}\left(k_{pi} + \alpha k_{vi}\right)S_i(\tilde{x}_i), \end{aligned} \tag{58}$$

where α is a positive constant and $\hat{J}^{\dagger}(q)$ denotes the pseudo-inverse of $\hat{J}(q)$ such that $\hat{J}(q)\hat{J}^T(q)$ is non-singular and $\hat{J}(q)\hat{J}^{\dagger}(q) = I$. The parameters k_{pi}, k_{vi} denote the ith diagonal elements of K_p and K_v, respectively. Consider the following inequality:

$$\begin{aligned} &\frac{1}{4}\dot{q}^T M(q)\dot{q} + \alpha \dot{q}^T M(q)\hat{J}^{\dagger}(q)s(\tilde{x}) + \textstyle\sum_{i=1}^{m}\left(k_{pi} + \alpha k_{vi}\right)S_i(\tilde{x}_i) \\ &\quad = \frac{1}{4}\left(\dot{q} + 2\alpha \hat{J}^{\dagger}(q)s(\tilde{x})\right)^T M(q)\left(\dot{q} + 2\alpha \hat{J}^{\dagger}(q)s(\tilde{x})\right) - \\ &\quad\quad \alpha^2 s(\tilde{x})^T\left(\hat{J}^{\dagger}(q)\right)^T M(q)\hat{J}^{\dagger}(q)s(\tilde{x}) + \textstyle\sum_{i=1}^{m}\left(k_{pi} + \alpha k_{vi}\right)S_i(\tilde{x}_i) \\ &\quad\quad \geq \textstyle\sum_{i=1}^{m}\left\{k_{pi}\bar{c}_i + \alpha(k_{vi}\bar{c}_i - \alpha\lambda_m)\right\}s_i{}^2(\tilde{x}_i) \end{aligned} \tag{59}$$

where $\lambda_m \triangleq \lambda_{max}\left[\left(\hat{J}^{\dagger}(q)\right)^T M(q)\hat{J}^{\dagger}(q)\right]$. Substituting the inequality Eq. 59 into Eq. 58 yields

$$V \geq \frac{1}{4}\dot{q}^T M(q)\dot{q} + \sum_{i=1}^{m}\left\{k_{pi}\overline{c}_i + \alpha(k_{vi}\overline{c}_i - \alpha\lambda_m)\right\}s_i{}^2(\tilde{x}_i) \geq 0$$

where K_v and α can be chosen so that

$$k_{vi}\overline{c}_i - \alpha\lambda_m > 0. \tag{60}$$

Hence, V is positive definite and represents a Lyapunov function candidate for the set-point control of the robot with uncertain Jacobian matrix. Differentiating the Lyapunov function candidate V with respect to time, it can be shown that (refer to Appendix for details)

$$\begin{aligned}\dot{V} \leq &-\left\{\lambda_{\max}(K_v)\left(\delta_1 - \frac{\gamma}{2}(\delta_2 + 2b_J)\right) - \alpha c_0\right\}\|\dot{q}\|^2 \\ &- \left(\lambda_{\max}(K_p)\left(\alpha\delta_3 - \frac{\gamma}{2}\right) - \alpha c_1\right)\|s(\tilde{x})\|^2\end{aligned} \tag{61}$$

such that c_0, c_1 are positive constants, b_J is the upper bound of $\|J(q)\|$, and

$$\begin{aligned}\delta_1 &= \frac{\lambda_{\min}\left[J^T(q)K_v J(q) + D(q)\right]}{\lambda_{\max}(K_v)} \\ \delta_2 &= \frac{\lambda_{\max}(K_p)}{\lambda_{\max}(K_v)} \\ \delta_3 &= \frac{\lambda_{\min}(K_p)}{\lambda_{\max}(K_p)}.\end{aligned} \tag{62}$$

If the following inequality holds

$$\min\left\{\frac{2\delta_1}{\delta_2 + 2b_J}, 2\alpha\delta_3\right\} > \gamma, \tag{63}$$

then by proper selection of parameters α, K_p, and K_v, such that

$$\begin{aligned}&\left(\delta_1 - \frac{\gamma}{2}(\delta_2 + 2b_J)\right) - \frac{\alpha c_0}{\lambda_{\max}(K_v)} > 0 \\ &\left(\alpha\delta_3 - \frac{\gamma}{2}\right) - \frac{\alpha c_1}{\lambda_{\max}(K_p)} > 0,\end{aligned} \tag{64}$$

it can be guaranteed that $\dot{V} < 0$. This implies the convergence of the task-space error and joint-space velocity such that $\tilde{x} \to 0$, $\dot{q} \to 0$, as $t \to \infty$.

Remark 7 For redundant robots, the dimension of x is different with the dimension of q, but the LaSalle's invariance theorem can be used to deduce the convergence of $\tilde{x}$, $\dot{q}$ to zero.

An alternate condition for the bound γ can be derived by substituting $J^T(q) = \hat{J}^T(q) + \tilde{J}(q)$ into Eq. 120 so that

$$\begin{aligned}\dot{V} \leq & -\dot{q}^T\left\{\hat{J}^T(q)K_v\hat{J}(q) + D(q) - \alpha c_0 I\right\}\dot{q} - \alpha s(\tilde{x})^T\left(K_p - c_1 I\right)s(\tilde{x}) \\ & - \dot{q}^T\tilde{J}^T(q)K_v\hat{J}(q)\dot{q} + \dot{q}^T\tilde{J}^T(q)K_p s(\tilde{x}).\end{aligned} \tag{65}$$

Therefore, condition Eq. 63 is written as follows:

$$\min\left\{\frac{2\hat{\delta}_1}{\delta_2 + 2b_{\hat{J}}}, 2\alpha\delta_3\right\} > \gamma, \tag{66}$$

where $\hat{\delta}_1 = \frac{\lambda_{\min}\left[\hat{J}^T(q)K_v\hat{J}(q) + D(q)\right]}{\lambda_{\max}(K_v)}$ and $b_{\hat{J}}$ is the upper norm bound of $\hat{J}(q)$. This implies that the bound γ can be represented by the actual or estimated Jacobian matrix whichever is larger.

In order to further explore the condition Eq. 63, let $K_v = k_v I$, $K_p = ak_v I$. Hence, condition Eq. 63 is simplified as

$$\min\left\{\frac{2\lambda_{\min}\left[\hat{J}^T(q)\hat{J}(q) + {}^{D(q)}/_{k_v}\right]}{a + 2b_{\hat{J}}}, 2\alpha\right\} > \gamma. \tag{67}$$

For a sufficiently large $\alpha\left(\text{i.e., } \alpha > \frac{\lambda_{\min}\left[\hat{J}^T(q)\hat{J}(q) + {}^{D(q)}/_{k_v}\right]}{2b_{\hat{J}}}\right)$, condition Eq. 67 is written as

$$\frac{2\lambda_{\min}\left[\hat{J}^T(q)\hat{J}(q) + {}^{D(q)}/_{k_v}\right]}{a + 2b_{\hat{J}}} > \gamma. \tag{68}$$

The condition Eq. 63 is illustrated in Fig. 7. In order to guarantee the stability of the system with approximate Jacobian matrix, the allowable bound of the Jacobian uncertainty γ must be smaller than the curve as shown in the figure.

The following conditions can be obtained from Fig. 7:

(i) If γ is small, α can be chosen small and therefore from Eq. 60 it can be deduced that a smaller controller gain is required.
(ii) If γ is small, a wider range of the feedback-gains ratio a can be chosen. In fact, in the extreme case where $\gamma = 0$, a could be chosen as any value.
(iii) If γ is large, a larger controller gain is required and a narrower range of the feedback-gains ratio a is allowed.

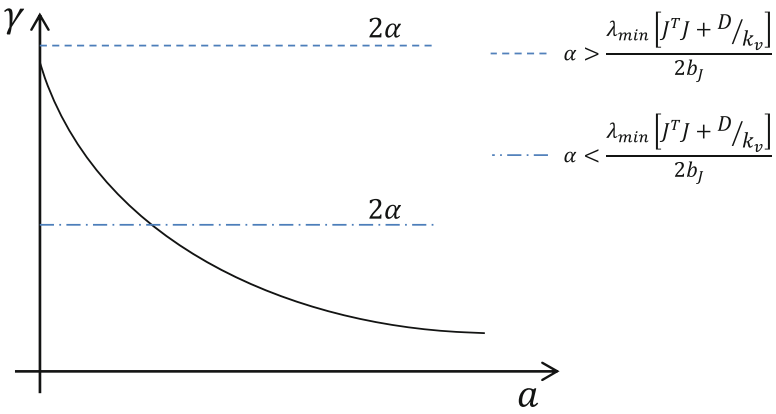


Fig. 7 An illustration of condition Eq. 63 where the vertical axis represents the bound of the Jacobian uncertainty and the horizontal axis represents the feedback-gains ratio

Moreover, if a is increased, then the allowable bound γ of the Jacobian uncertainty $\tilde{J}(q) = \hat{J}^T(q) - J^T(q)$ is reduced. Therefore, the feedback-gains ratio a should be kept smaller so that the allowable bound of the Jacobian uncertainty is larger. This can be easily done by either reducing K_p or increasing K_v. Though the condition is a sufficient condition, it is reasonable because increasing K_p amplifies the estimated Jacobian $\hat{J}(q)$ and hence more accuracy on the estimation or more damping is required. An important and piratical conclusion of the result is that when the system is unstable, redesign of $\hat{J}(q)$ may not be necessary as the instability may be due to the reason that the feedback-gains ratio a is not tuned properly. In practice, we should therefore try to stabilize the system or increase the margin of stability first by reducing the feedback-gains ratio a (Fig. 7).

In the presence of uncertainties in gravitational force and Jacobian matrix, the task-space control of robot manipulators can be obtained by using the concept of regressor for the robot dynamics (see Eq. 7) such that a gravity regressor is introduced to compensate for the gravity force.

The gravity term can be characterized by a set of parameters $\varphi = (\varphi_1, \cdots, \varphi_p)^T$ as

$$g(q) = Z(q)\varphi, \tag{69}$$

where $Z(q) \in R^{n \times p}$ is the gravity regressor.

Example 4 For the two-link robot manipulator depicted in Fig. 1, $Z(q)$ can be expressed as follows:

$$Z(q) = \begin{bmatrix} \cos(q_1) & \cos(q_1 + q_2) \\ 0 & \cos(q_1 + q_2) \end{bmatrix},$$

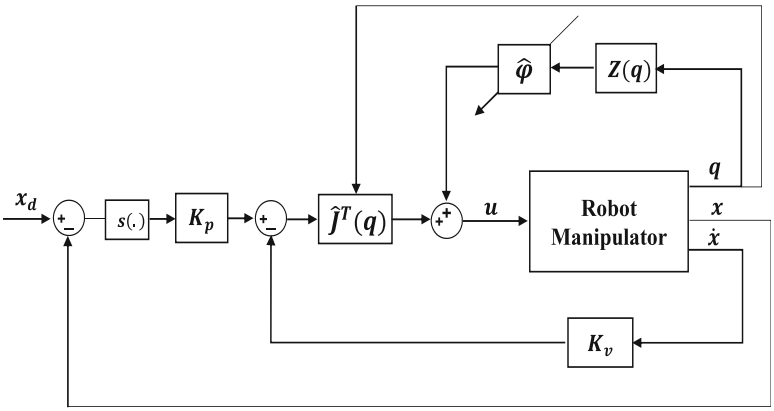


Fig. 8 A block diagram of the task-space approximate Jacobian regulator with adaptive gravity compensation

and the unknown parameter vector φ is expressed as

$$\varphi = \begin{bmatrix} (M_1 L_{c1} + M_2 L_1) g \\ M_2 L_{c2g} \end{bmatrix}.$$

The feedback control law is obtained as

$$u = -\hat{J}^T(q)\left(K_p s(\tilde{x}) + K_v \dot{x}\right) + Z(q)\hat{\varphi}, \tag{70}$$

such that the estimated parameter vector $\hat{\varphi}$ is updated as

$$\dot{\hat{\varphi}}(t) = -LZ^T(q)\left(\dot{q} + \alpha \hat{J}^{\dagger}(q) s(\tilde{x})\right), \tag{71}$$

where $L \in R^{p \times p}$ is a positive definite matrix. A block diagram of the task-space approximate Jacobian regulator Task-space approximate Jacobian regulator with adaptive gravity compensation described by Eqs. 70 and 71 is illustrated in Fig. 8.

Substituting Eqs. 69 and 70 into the dynamic Eq. 3 gives

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D(q)\dot{q}\hat{J}^T(q)\left(K_p s(\tilde{x}) + K_v \dot{x}\right) + Z(q)\Delta\varphi = 0, \tag{72}$$

where $\Delta\varphi = \varphi - \hat{\varphi}$. To examine the stability of the closed-loop system Eq. 72, the following Lyapunov function candidate is considered:

$$\begin{aligned} V = &\frac{1}{2}\dot{q}^T M(q)\dot{q} + \alpha \dot{q}^T M(q)\hat{J}^{\dagger}(q) s(\tilde{x}) \\ &+ \frac{1}{2}\Delta\varphi^T L^{-1}\Delta\varphi + \sum_{i=1}^{m}\left(k_{pi} + \alpha k_{vi}\right) S_i(\tilde{x}_i). \end{aligned} \tag{73}$$

The convergence of the task-space error and joint-space velocity can be shown by differentiating the Lyapunov function candidate with respect to time (similar to previous section), and using LaSalle's invariance principle.

Remark 8 The approximate Jacobian set-point controller Eq. 55 with task-space damping was proposed in (Cheah et al. 1998, 1999). The controller requires the measurement of a task-space position by using a sensor such as vision systems. The task-space velocity is obtained by numerical differentiation of the position. In (Cheah et al. 2003), an approximate Jacobian feedback controller using only joint-space damping was developed as

$$u = -\hat{J}^T(q)K_p s(\tilde{x}) - K_v\dot{q} + g(q). \tag{74}$$

The joint-space velocity is obtained by numerical differentiation of the joint position, which is usually less noisy than in the task space. In the presence of uncertainty in the gravitational force, the update law Eq. 71 or the integration control term (Cheah et al. 1999) also requires the use of the inverse Jacobian matrix. An approximate Jacobian controller with adaptive gravity compensation was developed as (Cheah et al. 2003)

$$u = -\hat{J}^T(q)\left(K_p s(\tilde{x}) + K_v\dot{x}\right) + Z(q)\hat{\varphi}, \tag{75}$$

such that the estimated parameter vector $\hat{\varphi}$ is updated as

$$\dot{\hat{\varphi}}(t) = -LZ^T(q)\left(\dot{q} + \alpha\hat{J}^T(q)s(\tilde{x})\right), \tag{76}$$

where α is positive constant and $s(.)$ is the saturation function. The main advantages of the controller are that the task-space velocity and the inverse of the approximate Jacobian matrix are not required. An update law can also be used to update the kinematic parameters of the estimated Jacobian matrix online (Cheah 2003; Dixon 2007), and the simpler stability conditions can be obtained as compared to using a fixed approximate Jacobian matrix.

Adaptive Jacobian Tracking Control

The approximate Jacobian controller in the previous section is only effective for point-to-point control problem. This section presents an adaptive Jacobian controller for tracking control of a robot manipulator with uncertain kinematics and dynamics. The concurrent adaptation to both kinematic and dynamic uncertainties is something "humanlike" as in tool manipulation. The key idea is to introduce a task-space adaptive sliding vector based on estimated task-space velocity and a dynamic regressor matrix based on the estimated kinematic parameters.

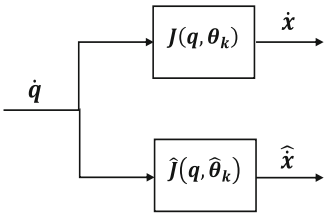


Fig. 9 The estimated task-space velocity and the actual task-space velocity

In the presence of kinematic uncertainty, the parameters of the Jacobian matrix are uncertain and an estimated velocity is defined using Eq. 16 as

$$\hat{\dot{x}} = \hat{J}\left(q,\hat{\theta}_k\right)\dot{q} = Y_k(q,\dot{q})\hat{\theta}_k \tag{77}$$

where $\hat{\dot{x}} \in R^n$ denotes the estimated task-space velocity, $\hat{J}\left(q,\hat{\theta}_k\right) \in R^{n\times n}$ is an estimated or adaptive Jacobian matrix, and $\hat{\theta}_k \in R^q$ denotes a set of estimated kinematic parameters. Figure 9 illustrates the difference between the estimated task-space velocity and the actual task-space velocity.

A reference vector $\dot{x}_r \in R^n$ is defined as follows:

$$\dot{x}_r = \dot{x}_d - \alpha(x - x_d) \tag{78}$$

where $x_d(t) \in R^n$ is the desired trajectory in the task space. Differentiating Eq. 78 with respect to time yields

$$\ddot{x}_r = \ddot{x}_d - \alpha(\dot{x} - \dot{x}_d) \tag{79}$$

where $\ddot{x}_d = \frac{d\dot{x}_d}{dt} \in R^n$ is the desired acceleration in the task space. Using the reference vector, an adaptive task-space sliding vector is defined as

$$\hat{\mathfrak{s}}_x = \hat{\dot{x}} - \dot{x}_r = \hat{J}\left(q,\hat{\theta}_k\right)\dot{q} - \dot{x}_r \tag{80}$$

where $\hat{J}\left(q,\hat{\theta}_k\right)\dot{q} = Y_k(q,\dot{q})\hat{\theta}_k$ as indicated in Eq. 77. The above vector is adaptive in the sense that the parameters of the estimated Jacobian matrix are updated by a parameter update law which will be introduced in the later development. The adaptive task-space sliding vector can also be written as follows:

$$\begin{aligned}\hat{\mathfrak{s}}_x &= \hat{J}\left(q,\hat{\theta}_k\right)\dot{q} - \dot{x}_d + \alpha\tilde{x} + \dot{x} - \dot{x} \\ &= \dot{\tilde{x}} + \alpha\tilde{x} - \left(\underbrace{J(q,\theta_k)\dot{q}}_{Y_k(q,\dot{q})\theta_k} - \underbrace{\hat{J}\left(q,\hat{\theta}_k\right)\dot{q}}_{Y_k(q,\dot{q})\hat{\theta}_k}\right) \\ &= \dot{\tilde{x}} + \alpha\tilde{x} - Y_k(q,\dot{q})\Delta\theta_k.\end{aligned} \tag{81}$$

Equation 81 shows that $\hat{\mathfrak{s}}_x$ converges to the sliding vector $\left(\dot{\tilde{x}} + \alpha\tilde{x}\right)$, as $\Delta\theta_k$ goes to zero.

Differentiating Eq. 80 with respect to time yields

$$\dot{\hat{\mathfrak{s}}}_x = \dot{\hat{x}} - \ddot{x}_r = \hat{J}\left(q, \hat{\theta}_k\right)\ddot{q} + \dot{\hat{J}}\left(q, \hat{\theta}_k\right)\dot{q} - \ddot{x}_r \tag{82}$$

where $\dot{\hat{x}}$ denotes the derivative of $\hat{x}$.

Now, a reference vector is defined in the joint space as

$$\dot{q}_r = \hat{J}^{-1}\left(q, \hat{\theta}_k\right)\dot{x}_r \tag{83}$$

where $\hat{J}^{-1}\left(q, \hat{\theta}_k\right)$ is the inverse of the approximate Jacobian matrix $\hat{J}\left(q, \hat{\theta}_k\right)$. Here, it is assumed that the robot is operating in a finite task space such that the approximate Jacobian matrix is of full rank. From Eq. 83, it is obtained:

$$\ddot{q}_r = \hat{J}^{-1}\left(q, \hat{\theta}_k\right)\ddot{x}_r + \dot{\hat{J}}^{-1}\left(q, \hat{\theta}_k\right)\dot{x}_r \tag{84}$$

Hence, an adaptive sliding vector in joint space can be defined as

$$\mathfrak{s} \;=\; \dot{q} - \dot{q}_r = \dot{q} - \hat{J}^{-1}\left(q, \hat{\theta}_k\right)\dot{x}_r \tag{85}$$

and

$$\dot{\mathfrak{s}} = \ddot{q} - \ddot{q}_r. \tag{86}$$

To obtain the relationship between the adaptive sliding vectors in the joint space and task space, multiply both sides of Eq. 85 by $\hat{J}\left(q, \hat{\theta}_k\right)$ and using Eq. 80 as follows:

$$\hat{J}\left(q, \hat{\theta}_k\right)\mathfrak{s} \;= \hat{J}\left(q, \hat{\theta}_k\right)\dot{q} - \dot{x}_r = \hat{\mathfrak{s}}_x. \tag{87}$$

Substituting Eqs. 85 and 86 into Eq. 3 and using property 3, the equations of motion can be expressed as

$$M(q)\dot{\mathfrak{s}} + C(q, \dot{q})\mathfrak{s} + D(q)\mathfrak{s} + \mathrm{Y}(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\theta_d = u \tag{88}$$

such that

$$\mathrm{Y}(q, \dot{q}, \dot{q}_r, \ddot{q}_r)\theta_d = M(q)\ddot{q}_r + C(q, \dot{q})\dot{q}_r + D(q)\dot{q}_r + g(q). \tag{89}$$

The adaptive Jacobian tracking controller is presented as follows:

$$u = -\hat{J}^T\left(q, \hat{\theta}_k\right)\left(K_v\dot{\tilde{x}} + K_p\tilde{x}\right) + Y\left(q, \dot{q}, \dot{q}_r, \ddot{q}_r\right)\hat{\theta}_d \tag{90}$$

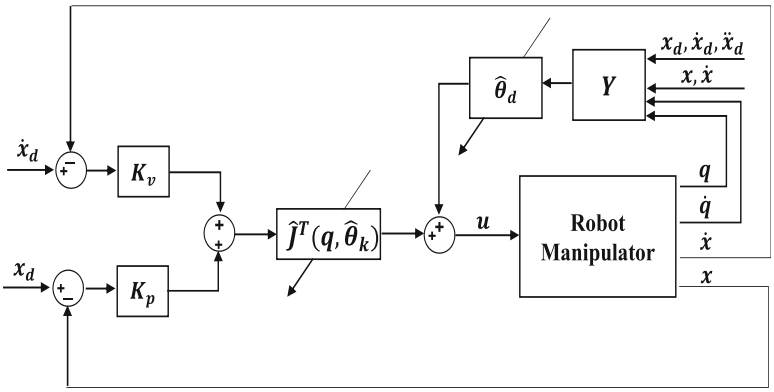


Fig. 10 A block diagram of the adaptive Jacobian tracking controller Eq. 90

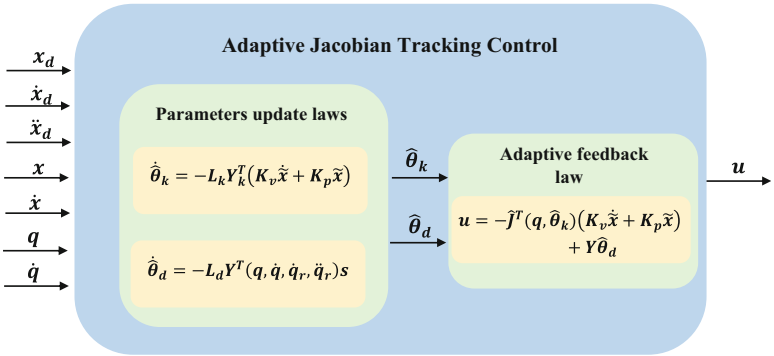


Fig. 11 A schematic diagram of the adaptive Jacobian tracking controller (Eq. 90) together with parameter update laws (91) and (92)

where $\tilde{x} = x - x_d$, $\dot{\tilde{x}} = \dot{x} - \dot{x}_d$, and $K_v \in R^{n\times n}$ and $K_p \in R^{n\times n}$ are symmetric positive definite gain matrices. The kinematic and dynamic adaptation laws are introduced as follows:

$$\dot{\hat{\theta}}_k = L_k Y_k^T(q,\dot{q})\left(k_v\dot{\tilde{x}} + K_p\tilde{x}\right) \tag{91}$$

$$\dot{\hat{\theta}}_d = -L_d Y^T(q,\dot{q},\dot{q}_r,\ddot{q}_r)\mathfrak{s}. \tag{92}$$

A block diagram and an illustration of the adaptive Jacobian tracking controller Eq. 90 are depicted in Figs. 10 and 11, respectively.

Substituting Eq. 90 into Eq. 88, the closed-loop equation can be expressed as

$$M(q)\dot{\mathfrak{s}} + C(q,\dot{q})\mathfrak{s} + D(q)\mathfrak{s} + \hat{J}^T\left(q,\hat{\theta}_k\right)\left(K_v\dot{\tilde{x}} + K_p\tilde{x}\right) + \mathrm{Y}(q,\dot{q},\dot{q}_r,\ddot{q}_r)\Delta\theta_d = 0 \tag{93}$$

where $\Delta\theta_d = \theta_d - \hat{\theta}_d$. For stability analysis, a Lyapunov-like function candidate is defined as

$$V = \frac{1}{2}\mathfrak{s}^T M(q)\mathfrak{s} + \frac{1}{2}\Delta\theta_d^T L_d^{-1}\Delta\theta_d + \frac{1}{2}\Delta\theta_k^T L_k^{-1}\Delta\theta_k + \frac{1}{2}\tilde{x}^T\left(K_p + \alpha K_v\right)\tilde{x} \tag{94}$$

where $\Delta\theta_k = \theta_k - \hat{\theta}_k$. Differentiating V with respect to time; substituting the closed-loop Eq. 93, $\dot{\hat{\theta}}_k$ from Eq. 91, and $\dot{\hat{\theta}}_d$ from Eq. 92 into it; and using property 2 yield

$$\begin{aligned}\dot{V} = &-\mathfrak{s}^T D(q)\mathfrak{s} - \hat{\mathfrak{s}}_x^T\left(K_v\dot{\tilde{x}} + K_p\tilde{x}\right) + \tilde{x}^T\left(K_p + \alpha K_v\right)\dot{\tilde{x}} \\ &- \Delta\theta_k^T Y_k^T(q,\dot{q})\left(K_v\dot{\tilde{x}} + K_p\tilde{x}\right).\end{aligned} \tag{95}$$

Substituting Eq. 81 into Eq. 95 yields

$$\dot{V} = -\mathfrak{s}^T D(q)\mathfrak{s} - \dot{\tilde{x}}^T K_v\dot{\tilde{x}} - \alpha\tilde{x}^T K_p\tilde{x} \leq 0. \tag{96}$$

Since $M(q)$ is positive definite, V in Eq. 94 is positive definite in $\mathfrak{s}$, Δx, $\Delta\theta_k$, and $\Delta\theta_d$. Therefore, Eq. 96 leads to boundedness of V, and consequently $\mathfrak{s}$, Δx, $\Delta\theta_k$, and $\Delta\theta_d$ are bounded, which implies that $\hat{\theta}_k$ and $\hat{\theta}_d$ are bounded and $\hat{\mathfrak{s}}_x = \hat{J}\left(q,\hat{\theta}_k\right)\mathfrak{s}$ is bounded as seen from Eq. 87. If x_d and its derivatives are bounded, then x and $\dot{x}_r$ (see Eq. 78) are bounded. Therefore, $\dot{q}_r$ in Eq. 83 is also bounded if the approximate Jacobian matrix is non-singular. From Eq. 85, $\dot{q}$ is bounded and the boundedness of $\dot{q}$ means that $\dot{x}$ is bounded since the Jacobian matrix is bounded. Hence, $\Delta\dot{x}$ is bounded and $\ddot{x}_r$ in Eq. 79 is also bounded if $\ddot{x}_d$ is bounded. From Eq. 91, $\dot{\hat{\theta}}_k$ is therefore bounded since Δx, $\Delta\dot{x}$, $\dot{q}$ are bounded and $Y_k(\cdot)$ is a trigonometric function of q. Therefore, $\ddot{q}_r$ in Eq. 84 is bounded. From the closed-loop Eq. 93, one can conclude that $\dot{\mathfrak{s}}$ is bounded. The boundedness of $\dot{\mathfrak{s}}$ implies the boundedness of $\ddot{q}$ as seen from Eq. 86. From Eq. 82, $\dot{\hat{\mathfrak{s}}}_x$ is therefore bounded. Finally, differentiating Eq. 81 with respect to time and rearranging yield

$$\ddot{\tilde{x}} + \alpha\dot{\tilde{x}} = \dot{\hat{\mathfrak{s}}}_x + \dot{Y}_k(q,\dot{q},\ddot{q})\Delta\theta_k - Y_k(q,\dot{q})\dot{\hat{\theta}}_k$$

which means that $\ddot{\tilde{x}} = \ddot{x} - \ddot{x}_d$ is also bounded. Therefore, $\ddot{V}$, which is shown as follows:

$$\ddot{V} = -2\mathfrak{s}^T B\dot{\mathfrak{s}} - 2\Delta\dot{x}^T K_v\ddot{\tilde{x}} - 2\alpha\tilde{x}^T K_p\dot{\tilde{x}},$$

is bounded. Hence, $\dot{V}$ is uniformly continuous. Using Barbalat's lemma, it gives $\tilde{x} = x - x_d \to 0$, $\dot{\tilde{x}} = \dot{x} - \dot{x}_d \to 0$, and $\mathfrak{s} \to 0$ as $t \to \infty$.

Remark 9 The adaptive Jacobian tracking controller Eq. 90 for robots with uncertainties in kinematics and dynamics was developed in (Cheah et al. 2004). The proposed controller requires the differentiation of the task-space position which is

noisy. An adaptive Jacobian tracking control law based on filtered differentiation of the measured task-space position was developed in (Cheah et al. 2006a). Observer design techniques can also be used to estimate the velocities (Liang et al. 2010). To avoid singularities associated with Euler angle representation, adaptive Jacobian tracking controller based on unit quaternion was developed (Braganza et al. 2005). An adaptive Jacobian controller based on prediction error was proposed in (Wang and Xie 2009). Sliding PID tracking control schemes with uncertain Jacobian were proposed in (Garcia-Rodriguez and Parra-Vega 2012).

When cameras are used to measure the position of the end effector, the vector of image feature parameter rates of change x is related to joint variables by the following equation (Cheah et al. 2007):

$$\dot{x} = \overbrace{\underbrace{Z^{-1}(q)L(x)}_{J_I} J_m(q)}^{J(q)} \dot{q} \tag{97}$$

where $\boldsymbol{\mathcal{Z}}$ (q) is a diagonal matrix, which contains the depth information of the feature points with respect to the camera image frame, and $L(x)$ is a Jacobian matrix. Since the overall Jacobian matrix $J(q)$ is inversely proportional to the depths, it is not linearly parameterizable. Therefore, the kinematic parameters in the image Jacobian cannot be extracted to form a lumped kinematic parameter vector that includes the unknown depth parameters, i.e.,

$$J(q)\dot{q} \neq Y(q,\dot{q})\boldsymbol{\theta}. \tag{98}$$

Thus, these abovementioned adaptive Jacobian controllers are effective only in cases where the depth information is constant or slowly time varying. Vision-based tracking controllers with uncertain depth were proposed in (Wang et al. 2007), but the uncertainty of robot kinematics and dynamics was not considered and the depth information was not updated online. However, the following linear parameterizations hold

$$\boldsymbol{\mathcal{Z}}(q)\dot{x} = Y_z(q,\dot{x})\boldsymbol{\theta}_z \tag{99}$$

$$J_e(q)\dot{q} = Y_k(q,\dot{q})\boldsymbol{\theta}_k \tag{100}$$

where $Y_z(q,\dot{x})$ is called the depth regressor matrix and $Y_k(q,\dot{q})$ is called the kinematic regressor matrix. Therefore, an adaptive Jacobian tracking control with uncertain depth information can be proposed as (Cheah et al. 2007)

$$u = -\hat{J}^T\left(q,\hat{\theta}_k\right)\hat{\boldsymbol{\mathcal{Z}}}^{-1}\left(q,\hat{\theta}_z\right)\left(K_p\tilde{x} + K_v\dot{\tilde{x}}\right) + Y_d\left(q,\dot{q},\dot{q}_r,\ddot{q}_r\right)\hat{\theta}_d \tag{101}$$

where $\hat{J}^T\left(q,\hat{\theta}_k\right)\hat{\boldsymbol{\mathcal{Z}}}^{-1}\left(q,\hat{\theta}_z\right)$ is the adaptive Jacobian matrix and K_p and K_v are symmetric positive definite matrices. The uncertain dynamic, kinematic, and depth parameters are updated by the following update laws:

$$\dot{\hat{\theta}}_d = -L_d Y_d^T(q, \dot{q}, \dot{q}_r, \ddot{q}_r)s \tag{102}$$

$$\dot{\hat{\theta}}_k = L_k Y_k^T(q, \dot{q})\hat{Z}^{-1}\left(q, \hat{\theta}_z\right)\left(K_p\tilde{x} + K_v\dot{\tilde{x}}\right) \tag{103}$$

$$\dot{\hat{\theta}}_z = -L_z Y_z^T(q, \dot{x})\hat{Z}^{-1}\left(q, \hat{\theta}_z\right)\left(K_p\tilde{x} + K_v\dot{\tilde{x}}\right). \tag{104}$$

Remark 10 In the redundant case (Cheah et al. 2006a, b), the null space of the approximate Jacobian matrix can be used to minimize a performance index. Hence, the reference vector in the joint space can be defined as

$$\dot{q}_r = \hat{J}^{\dagger}\left(q, \hat{\theta}_k\right)\dot{x}_r + \left(I_n - \hat{J}^{\dagger}\left(q, \hat{\theta}_k\right)\hat{J}\left(q, \hat{\theta}_k\right)\right)\psi \tag{105}$$

where $\hat{J}^{\dagger}\left(q, \hat{\theta}_k\right)$ is the pseudo-inverse of the approximate Jacobian matrix and $\psi \in R^n$ is minus the gradient of the convex function to be optimized.

Simulation Results

In this section, various feedback laws presented in the previous sections are applied on the two-link robot manipulator (depicted in Fig. 1). The parameters of the robot are shown in Table 1. The simulation was carried out in MATLAB/SIMULINK.

PD Plus Gravity Control Law

For the set-point regulation, PD plus gravity control law Eq. 20 is applied on two-link robot manipulator. The control gains were chosen as $K_p = diag\{10, 10\}$ and $K_v = diag\{5, 5\}$. The initial values of joint angles were chosen as $q_1(0) = \pi/4$ and $q_2(0) = \pi/6$, and the initial values of joint velocities were chosen as $\dot{q}_1(0), \dot{q}_2(0) = 0$. The desired joint angles were chosen as $q_{d1} = \pi/2$ and $q_{d2} = \pi/3$. The simulation results are depicted in Figs. 12 and 13 which show error of joints angles and joint velocities. The actuator torques are shown in Fig. 14.

Now assume there is 10 % mismatch in measurement of physical parameters, i.e.,

$$\overline{M}_1 = M_1 + 0.1M_1\ \overline{M}_2 = M_2 + 0.1M_2$$
$$\overline{L}_1 = L_1 + 0.1L_1\ \ \overline{L}_2 = L_2 + 0.1L_2.$$

The error of joints angles is depicted in Fig. 15. Figure 15 shows that there is an offset between the final values of joints angles and their desired values. To alleviate this steady-state error, let the controller gains increase such that $K_p = diag\{60, 60\}$

Table 1 Parameters of the two link robot manipulator

i	M_i(kg)	L_i(m)	L_{ci}(m)	D_i(Nms/rad)
1	5	0.2	0.1	1
2	5	0.2	0.1	1

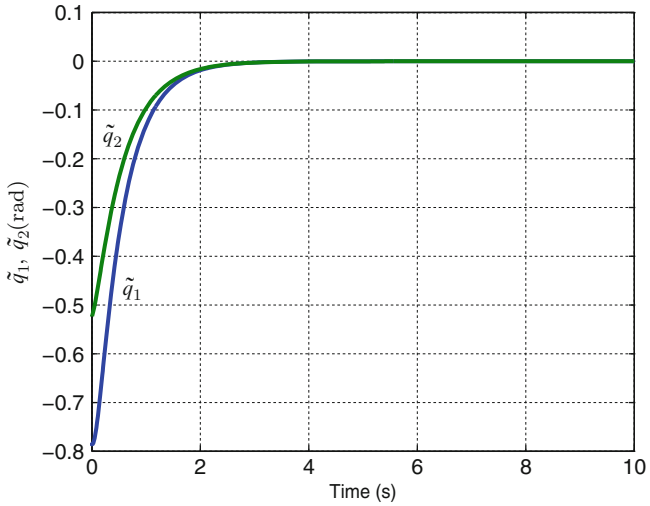


Fig. 12 Error of joints angles $\tilde{q}_1, \tilde{q}_2$

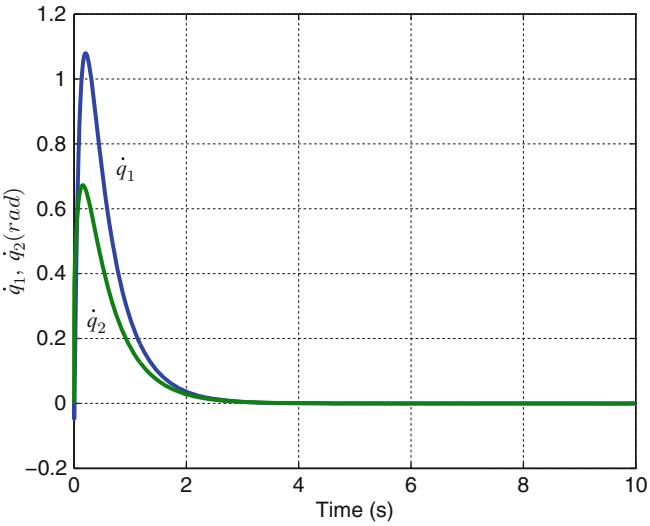


Fig. 13 Joints velocities $\dot{q}_1, \dot{q}_2$

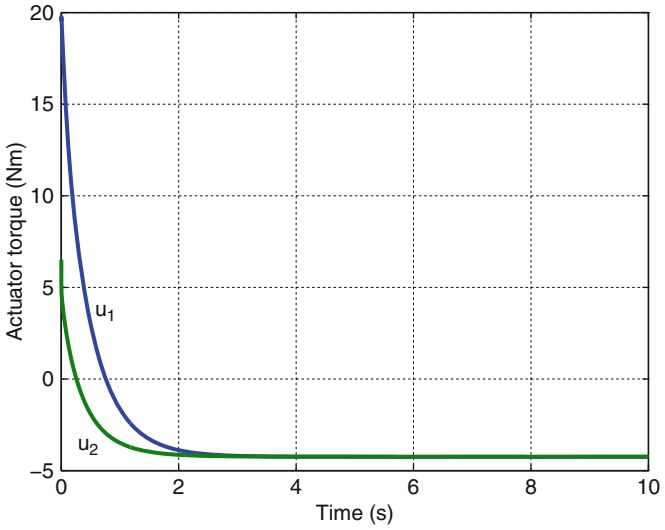


Fig. 14 Actuator torques u_1, u_2

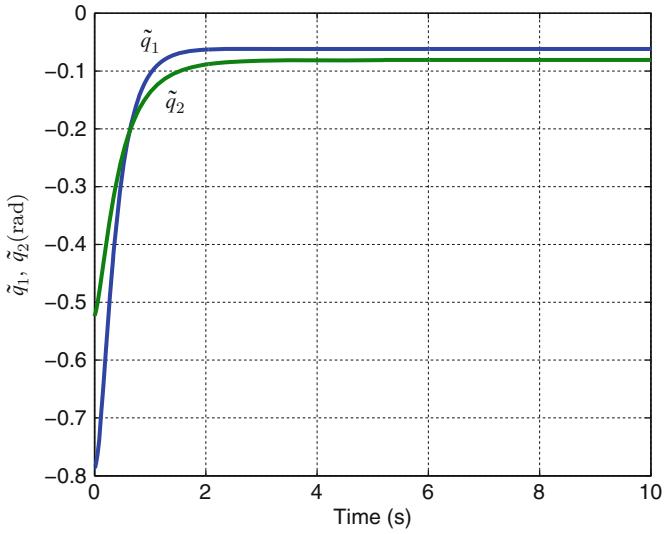


Fig. 15 $\tilde{q}_1, \tilde{q}_2$ in the presence of 10 % mismatch in measurement of physical parameters

and $K_v = diag\{25,\ 25\}$. Figure 16 shows that the steady-state errors improved considerably. However, if one plots the actuator toques as depicted in Fig. 17, it shows that the actuator torques are also increased which might lead to saturation. Hence, there is a trade-off between the accuracy and the control torque.

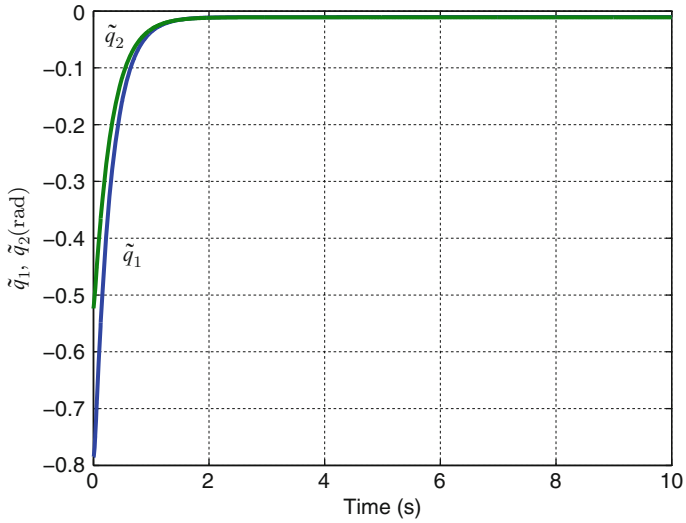


Fig. 16 $\tilde{q}_1, \tilde{q}_2$ for high control gains

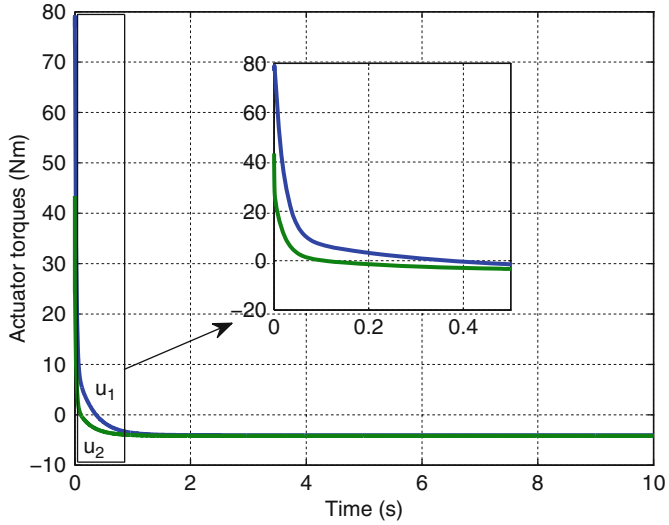


Fig. 17 Actuator torques u_1, u_2

Another method to improve the steady-state error in the presence of inaccurate measurements is to utilize an integral term to the PD plus gravity control law. Therefore, the feedback law for PID plus gravity law can be expressed as follows:

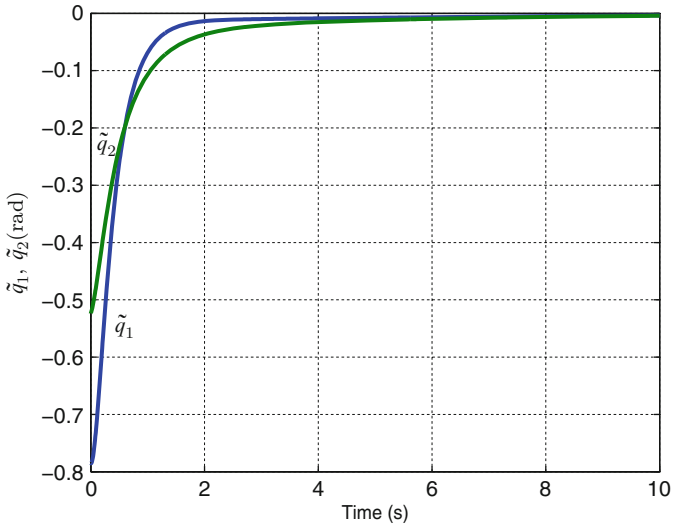


Fig. 18 $\tilde{q}_1, \tilde{q}_2$ for PID plus gravity feedback control

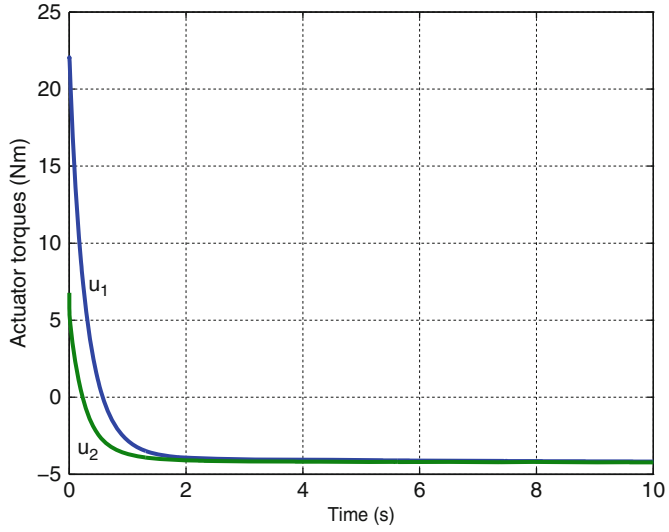


Fig. 19 Actuator torques u_1, u_2 for PID plus gravity feedback control

$$u = -K_v\dot{q} - K_p\tilde{q} - K_i\int \tilde{q}dt + g(q). \tag{106}$$

The simulation results for PID plus estimated gravity in the presence of 10% mismatch in measurement of physical parameters are depicted in Figs. 18 and 19. The control gains were chosen as $K_p = diag\{10,\ 10\}$, $K_i = diag\{2,\ 2\}$,

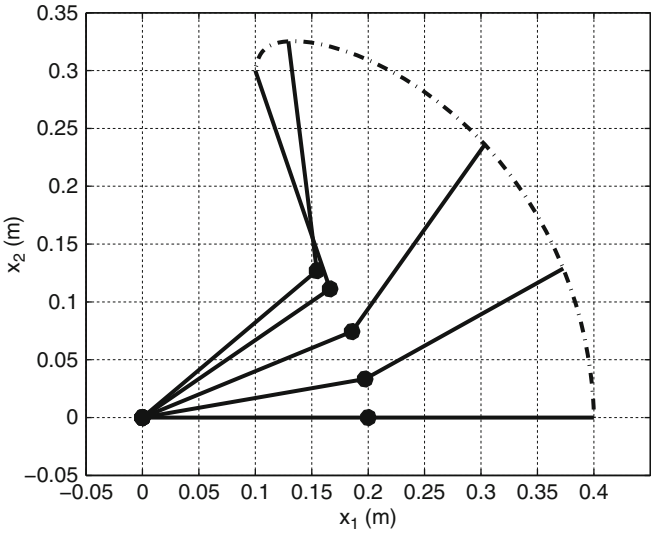


Fig. 20 The end-effector position for the task-space PD plus gravity regulator

and $K_v = diag\{5,\ 5\}$. It is shown that both accuracy and control torques are improved significantly.

Task-Space PD Plus Gravity

The task-space PD plus gravity control law Eq. 28 is applied for two-link robot manipulator to control its end-effector position to reach the desired position. The controller gains were chosen as $K_p = diag\{200, 200\}$ and $K_v = diag\{50, 50\}$. The initial position of robot was $x = [0.4, 0]$, and the desired end-effector position was considered as $x = [0.1, 0.3]$. The end-effector position is depicted in Fig. 20. The robot arms are depicted for different snapshots. The actuator toques are depicted in Fig. 21.

Adaptive Control

The adaptive feedback law Eq. 46 together with the parameter update law Eq. 47 is applied to the two-link robot manipulator for trajectory tracking. The controller gains were chosen as $K_s = diag\{15,\ 15\}$ and $\lambda = 10$. The gain for update law was chosen as $L = 50$. The initial position and velocity of the robot's joints were $q = [0,\ 0]$ and $\dot{q} = [0, 0]$, respectively. The desired trajectory was set as

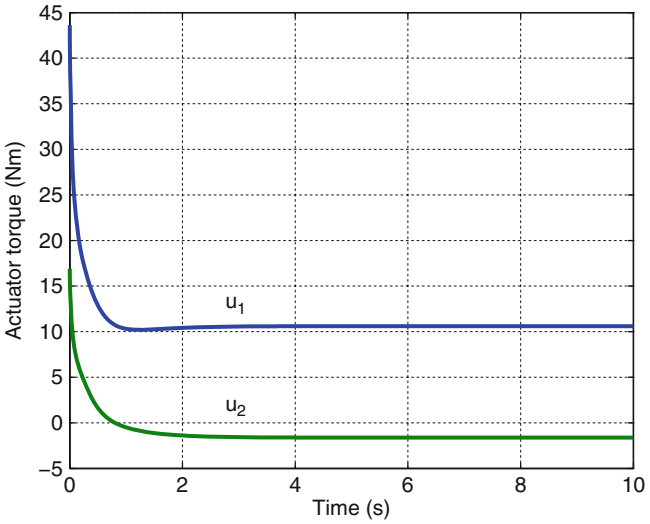


Fig. 21 Actuator torques u_1, u_2 for the task-space PD plus gravity regulator

$$q_d(t) = \begin{bmatrix} 0.1 + 0.1t + 0.2\sin(t) \\ 0.1 + 0.1t + 0.2\cos(t) \end{bmatrix}. \tag{107}$$

The initial values of the unknown parameters were chosen as $\hat{\theta}_d(0) = [0, 0, 0, 0, 0, 0, 0]^T$. The position and the desired reference of joint one and two are depicted in Figs. 22 and 23. The velocity and the desired velocity of joint one and two are depicted in Figs. 24 and 25. The actuator toques are depicted in Fig. 26.

Approximate Jacobian Set-Point Control with Uncertain Gravitational Force

For this part, the approximate Jacobian set-point control in the presence of uncertainties in gravitational force was considered for the simulation. The approximate Jacobian regulator together with adaptive gravity compensation (Eqs. 75 and 76) was applied for the two-link robot manipulator to control its end-effector position to reach the desired position. The controller gains were chosen as $K_p = diag\{1000, 1000\}$ and $K_v = diag\{100, 100\}$. The adaptive gains were chosen as $L = 500$ and $\alpha = 50$. The initial values of the unknown parameters were chosen as $\hat{\varphi}(0) = [0, 0]^T$. The initial position of the robot's end effector was $x = [0.4, 0]$, and the desired end-effector position was set as $x = [0.1,.3]$. The end-effector position is depicted in Fig. 27. The robot arms are depicted for different snapshots. The actuator toques are depicted in Fig. 28.

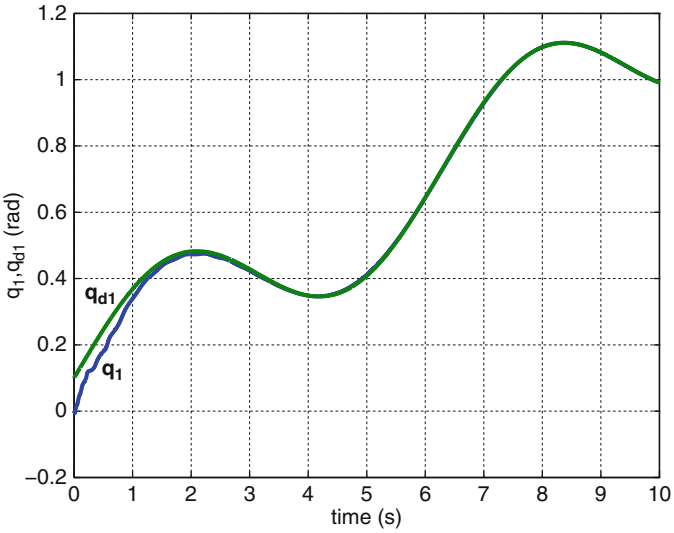


Fig. 22 The position and the desired reference of joint one for the adaptive trajectory tracking

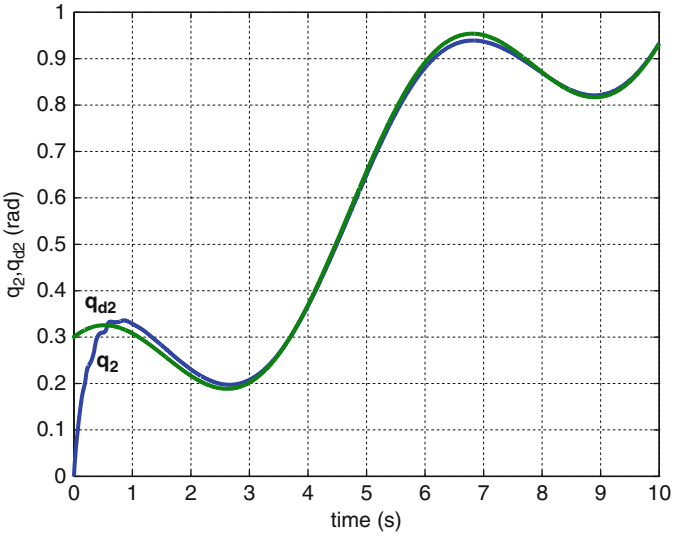


Fig. 23 The position and the desired reference of joint two for the adaptive trajectory tracking

Adaptive Jacobian Tracking Control

In this case, the adaptive Jacobian tracking control Eq. 90 together with the kinematic parameter update law Eq. 91 and dynamic parameter update law Eq. 92 is applied to the two-link robot manipulator for trajectory tracking in the

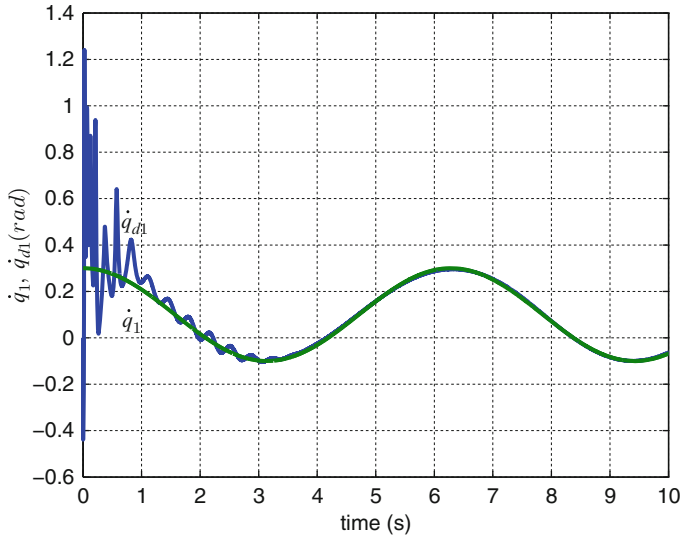


Fig. 24 The velocity and the desired velocity of joint one for the adaptive trajectory tracking

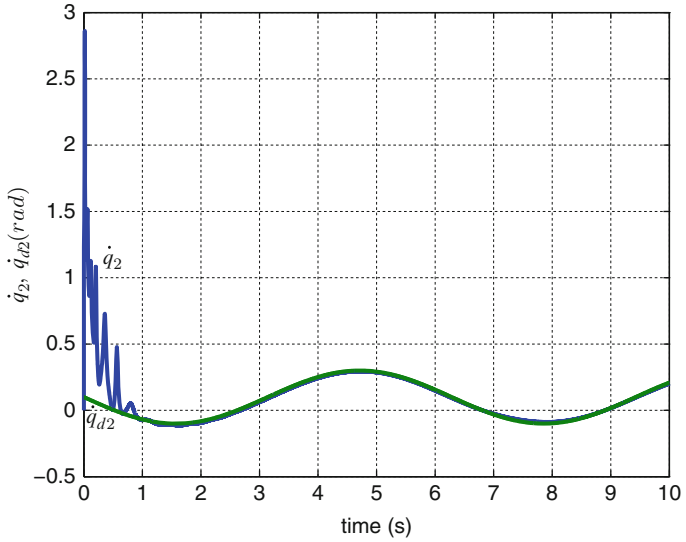


Fig. 25 The velocity and the desired velocity of joint two for the adaptive trajectory tracking

presence of uncertain kinematics and dynamics. The controller gains were chosen as $K_p = diag\{800,\ 800\}$ and $K_v = diag\{80,\ 80\}$. The adaptive gains were chosen as $L_k = 5$ and $L_d = 5$. The initial values of the unknown parameters were chosen as $\hat{\varphi}(0) = [0, 0]^T$. The initial position and velocity of robot's end effector were $x = [0,\ 0.15]$ and $x_d = [0,\ 0]$, respectively. The desired trajectory was set as

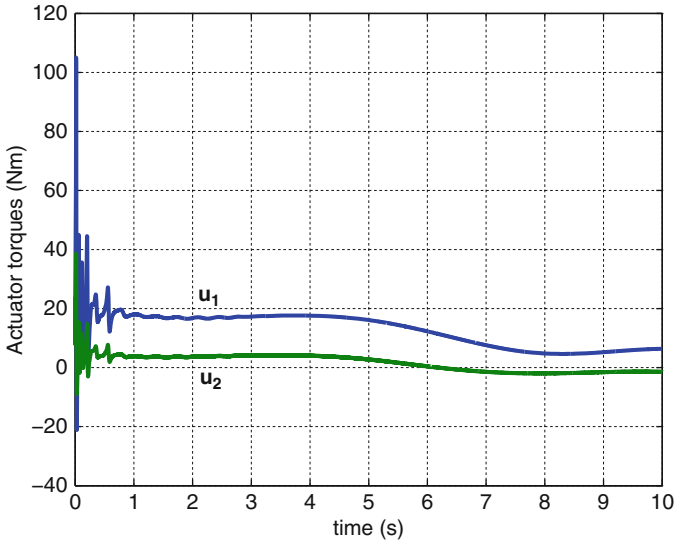


Fig. 26 Actuator torques u_1, u_2 for the adaptive trajectory tracking

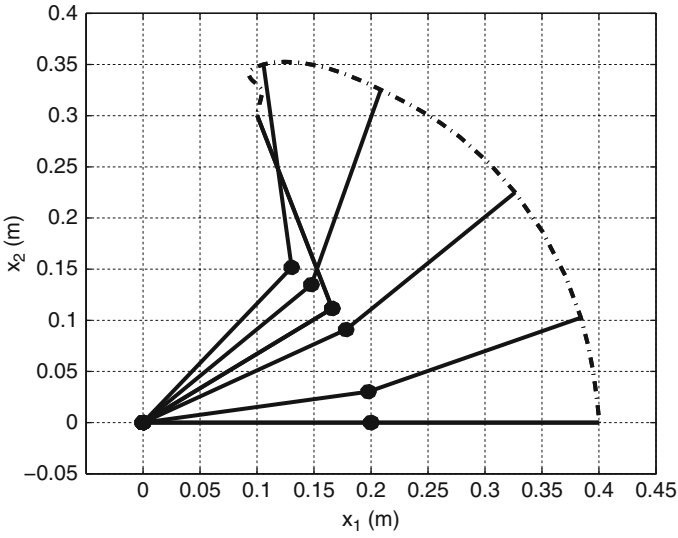


Fig. 27 The end-effector position for the task-space approximate Jacobian regulator with adaptive gravity compensation

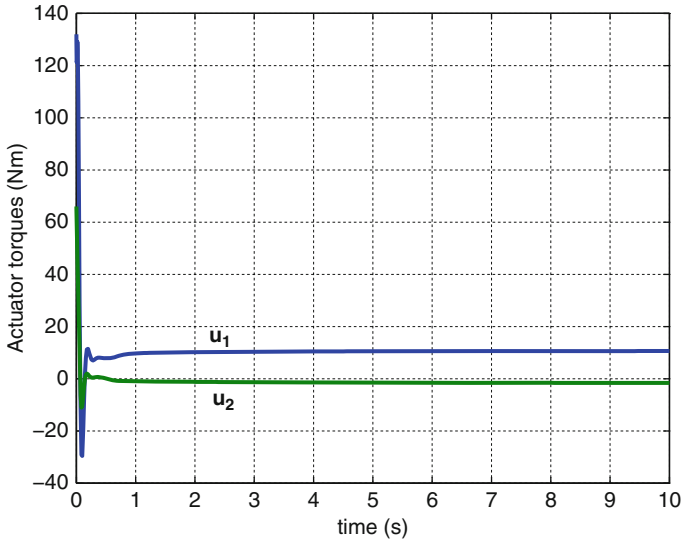


Fig. 28 Actuator torques u_1, u_2 for the task-space approximate Jacobian regulator with adaptive gravity compensation

$$x_d(t) = \begin{bmatrix} 0.15 + 0.05 \sin(t/2) \\ 0.15 + 0.05 \cos(t/2) \end{bmatrix}. \tag{108}$$

The initial values of the unknown kinematic and dynamic parameters $\left(\hat{\theta}_k(0) \text{ and } \hat{\theta}_d(0)\right)$ were chosen randomly from the interval [0, 10]. The end-effector position is depicted in Fig. 29. The robot arms are depicted for different snapshots.

Summary

This chapter introduces robot control design methods using the Lyapunov based method. By utilizing the physical properties of the robot kinematics and dynamics, several set-point and adaptive tracking controllers have been presented in both the joint space and task space. A simple motion controller that is effective for set-point regulation is the PD controller with gravity compensation. For tracking control tasks, the adaptive controller has been presented for a robot manipulator with uncertain dynamics. Using sensory feedback of the robot end-effector position in the task space, approximate Jacobian set-point controllers and adaptive Jacobian tracking controllers have been presented for robot manipulators with uncertainties in both kinematics and dynamics.

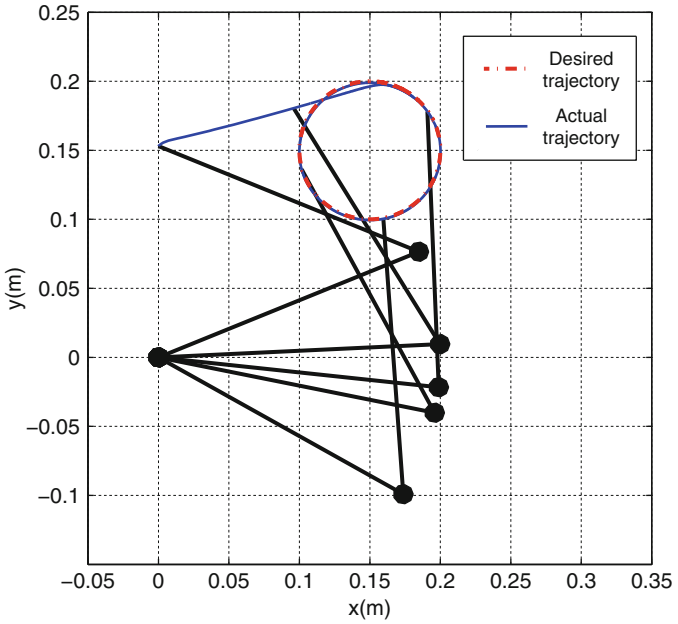


Fig. 29 The end-effector position for the task-space adaptive Jacobian tracking control

Appendix 1

Preliminaries on control theories:

Consider the following nonlinear system:

$$\dot{x} = f(x,t) \tag{109}$$

where $x \in R^n$ is a vector of state of the system and $f(.)$ is a nonlinear function.

Definition A1 *The equilibrium points of the system* Eq. 109 *are defined as the state vectors* x_e *of* x *for which if at specific time* t_0, $x = x_e$, *then* x *will remain unchanged for all* $t > t_0$. *In other words, at equilibrium points the state of the system satisfies* $f(x_e) = 0$.

It is often important to know whether the equilibrium point is stable or not. In the following, a definition of stable equilibrium point is put forward:

Remark A1 It can be always assumed that the equilibrium point is zero by using change of variable $y = x - x_e$.

Definition A2 *The equilibrium point* $x_e = 0$ *of the system* Eq. 109 *is said to be stable if for any* $\varepsilon > 0$, *there exists* $\delta > 0$ *such that if* $\|x(0) - 0\| < \delta$, *then* $\|x(t) - 0\| < \varepsilon$ *for all* $t > 0$. *It can be mathematically represents as follows:*

$$\forall \varepsilon > 0 \ \exists \delta > 0, \ \textit{if} \ \|x(0) - 0\| < \delta \Rightarrow \|x(t) - 0\| < \varepsilon \textit{ for } t > t_0. \tag{110}$$

To examine the stability of the equilibrium point, the Lyapunov theory can be utilized. The main advantage of the Lyapunov theory is that the stability of the system can be determined without solving the differential equations of the system. Moreover, the Lyapunov theory can be used to design controllers that stabilize nonlinear systems.

Before presenting the Lyapunov theory, a certain class of functions is introduced as follows:

Definition A3 *A continuous function $V : R^n \times R_+ \rightarrow R$ is a locally positive definite function (lpdf) if for some $\varepsilon > 0$ and some continuous, the following conditions hold:*

$$\begin{cases} I{:}V(0,t) = 0 \\ II{:}V(x,t) > 0 \quad \forall x \in B_\varepsilon, \forall t \geq 0 \end{cases} \tag{111}$$

where B_ε is a ball of size ε around the origin which is mathematically expressed as $B_\varepsilon = \{x \in R^n : \|x\| < \varepsilon\}$.

In addition, V is a positive definite function (pdf) if the condition (II) is true for all $x \in R$.

If in condition (II) $V(x, t) \geq 0$, then $V(x, t)$ is a (locally) positive semi-definite function.

Remark A2 If $V(x) = x^T M x$, where M is a real symmetric matrix, then V is a pdf if and only if M is a positive definite matrix.

Theorem A1 ***Lyapunov stability theorem:*** *Suppose $x_e = 0$ is an equilibrium point of the system* Eq. 109. *Let $V(x, t)$ be a nonnegative function with derivative $\dot{V}$ along trajectories of the system dynamics:*

$$\frac{dV}{dt} = \frac{\partial V}{\partial t}\dot{x} = \frac{\partial V}{\partial t} f(x,t). \tag{112}$$

(i) *If $V(x, t)$ is a locally positive definite function and $\dot{V}$ is a locally positive semi-definite function in x and for all t, then the origin of the system is locally stable.*

(ii) *If $V(x, t)$ is a locally positive definite function and $\dot{V}$ is a locally positive definite function in x and for all t, then the origin of the system is locally asymptotically stable.*

(iii) *If $V(x, t)$ is a positive definite function and $\dot{V}$ is a positive definite function in x and for all t, then the origin of the system is globally asymptotically stable.*

If the function $V(x, t)$ exists in the above theorem, then it is called a Lyapunov function.

Remark A3 The Lyapunov stability theorem only provides sufficient conditions for the stability of nonlinear systems; hence, the failure of finding a Lyapunov function does not prove the instability of the nonlinear system

In the case that $-\dot{V}(x,t)$ is a positive semi-definite function, the Lyapunov stability theorem cannot provide any information on the asymptotic stability of the system. To deal with the stability of nonlinear autonomous systems when $-\dot{V}(x)$ is a positive semi-definite function, LaSalle's invariance principle has been presented.

Lemma A1 ***LaSalle's invariance principle****: Let $V : R^n \rightarrow R$ be a positive definite function such that $\dot{V}(x) \leq 0$ in compact set Ω. Let D be the set of all points in Ω where $\dot{V}(x) = 0$. Therefore, every solution of the system $\dot{x} = f(x)$ starting in Ω approaches to the largest invariant set inside D. In particular, if D contains no trajectories other than $x = 0$, then 0 is locally asymptotically stable.*

LaSalle's invariance principle enables one to conclude asymptotic stability only for autonomous systems. For non-autonomous systems, Barbalat's lemma can be used.

Lemma A2 ***Barbalat's lemma****: If a function V (t, x) satisfies the following conditions:*

(i) *V(x, t) is lower bounded.*
(ii) *$\dot{V}(x,t)$ is negative semi-definite.*
(iii) *$\dot{V}(x,t)$is uniformly continuous in time or equivalently $\ddot{V}(t,x)$ is bounded. Then, $\dot{V}(x,t)$ goes to zero as $t \rightarrow \infty$.*

Appendix 2

Parameters of dynamic Eq. 4 of the two-link robot manipulator which is depicted in Fig. 1:

Elements of inertia matrix $M(q)$ are

$$\begin{aligned} M_{11} &= \frac{4}{3}M_1L_{c1}^2 + \frac{4}{3}M_2L_{c2}^2 + M_2L_1^2 + 2M_2L_1L_{c2}\cos(q_2) \\ M_{12} &= \frac{4}{3}M_2L_{c2}^2 + M_2L_1L_{c2}\cos(q_2) \\ M_{21} &= \frac{4}{3}M_2L_{c2}^2 + M_2L_1L_{c2}\cos(q_2) \\ M_{22} &= \frac{4}{3}M_2L_{c2}^2. \end{aligned} \tag{113}$$

Elements of matrix $C(q,\dot{q})$ are given as

$$\begin{aligned} C_{11} &= -M_2 L_1 L_{c2} \dot{q}_2 \sin(q_2) \\ C_{12} &= -M_2 L_1 L_{c2} [\dot{q}_1 + \dot{q}_2] \sin(q_2) \\ C_{21} &= M_2 L_1 L_{c2} \dot{q}_1 \sin(q_2) \\ C_{22} &= 0. \end{aligned} \tag{114}$$

Elements of gravitational force matrix are

$$\begin{aligned} g_1 &= (M_1 L_{c1} + M_2 L_1) \mathfrak{g} \cos(q_1) + M_2 L_{c2} \mathfrak{g} \cos(q_1 + q_2) \\ g_2 &= M_2 L_{c2} \mathfrak{g} \cos(q_1 + q_2) \end{aligned} \tag{115}$$

where $\mathfrak{g}$ is the gravity due to acceleration.

Appendix 3

Part of the proof of the task-space control law Eq. 55:

Differentiating the Lyapunov candidate V (expressed in Eq. 58) with respect to time gives

$$\begin{aligned} \dot{V} = {} & \dot{q}^T M(q) \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M}(q) \dot{q} + \alpha \ddot{q}^T M(q) \hat{J}^{\dagger}(q) s(\tilde{x}) \\ & + \alpha \dot{q}^T \dot{M}(q) \hat{J}^{\dagger}(q) s(\tilde{x}) + \alpha \dot{q}^T M(q) \dot{\hat{J}}^{\dagger}(q) s(\tilde{x}) \\ & + \alpha \dot{q}^T M(q) \hat{J}^{\dagger}(q) \dot{s}(\tilde{x}) + \dot{x}^T K_p s(\tilde{x}) + \alpha \dot{x}^T K_v s(\tilde{x}). \end{aligned} \tag{116}$$

Substituting the closed-loop Eq. 57 into Eq. 116, using property 2 and simplifying, yields

$$\begin{aligned} \dot{V} = {} & - \dot{q}^T D(q) \dot{q} - \dot{q}^T \hat{J}^T(q) \left(K_p s(\tilde{x}) + K_v \dot{x} \right) \\ & - \alpha \dot{q}^T \left\{ C(q,\dot{q}) + D(q) - \dot{M}(q) \right\} \hat{J}^{\dagger}(q) s(\tilde{x}) \\ & - \alpha s(\tilde{x})^T K_p s(\tilde{x}) + \alpha \dot{q}^T M(q) \dot{\hat{J}}^{\dagger}(q) s(\tilde{x}) \\ & + \alpha \dot{q}^T M(q) \hat{J}^{\dagger}(q) \dot{s}(\tilde{x}) + \dot{x}^T K_p s(\tilde{x}). \end{aligned} \tag{117}$$

Using Eq. 14, $\dot{V}$ can be written as follows:

$$\begin{aligned} \dot{V} = {} & -\dot{q}^T \left\{ \hat{J}^T(q) K_v J(q) + D(q) \right\} \dot{q} - \dot{q}^T \left\{ \hat{J}^T(q) - J^T(q) \right\} K_p s(\tilde{x}) \\ & - \alpha \dot{q}^T \left\{ C(q,\dot{q}) + D(q) - \dot{M}(q) \right\} \hat{J}^{\dagger}(q) s(\tilde{x}) - \alpha s(\tilde{x})^T K_p s(\tilde{x}) \\ & + \alpha \dot{q}^T M(q) \dot{\hat{J}}^{\dagger}(q) s(\tilde{x}) + \alpha \dot{q}^T M(q) \hat{J}^{\dagger}(q) \dot{s}(\tilde{x}). \end{aligned} \tag{118}$$

Since $s(\tilde{x})$ is bounded, there exist constants $c_0 > 0$ and $c_1 > 0$ so that

$$\begin{aligned} -\alpha\dot{q}^T\{C(q,\dot{q})+D(q)-\dot{M}(q)\}\hat{J}^{\dagger}(q)s(\tilde{x})+\alpha\dot{q}^T M(q)\dot{\hat{J}}^{\dagger}(q)s(\tilde{x}) \\ +\alpha\dot{q}^T M(q)\hat{J}^{\dagger}(q)\dot{s}(\tilde{x}) \le \alpha c_0\|\dot{q}\|^2+\alpha c_1\|s(\tilde{x})\|^2. \end{aligned} \tag{119}$$

Substituting inequality Eq. 119 into Eq. 118 yields

$$\begin{aligned} \dot{V} \le & -\dot{q}^T\left\{\hat{J}^T(q)K_v J(q)+D(q)-\alpha c_0 I\right\}\dot{q}-\alpha s(\tilde{x})^T(K_p-c_1 I)s(\tilde{x}) \\ & -\dot{q}^T\left\{\hat{J}^T(q)-J^T(q)\right\}K_p s(\tilde{x}). \end{aligned} \tag{120}$$

Let $\tilde{J}(q)$ be the Jacobian estimation error which is defined as $\tilde{J}(q)=J(q)-\hat{J}(q)$. Hence, Eq. 120 can be written with respect to $J(q)$ and $\tilde{J}(q)$ as follows:

$$\begin{aligned} \dot{V} \le & -\dot{q}^T\{J^T(q)K_v J(q)+D(q)-\alpha c_0 I\}\dot{q}-\alpha s(\tilde{x})^T(K_p-c_1 I)s(\tilde{x}) \\ & +\dot{q}^T\tilde{J}^T(q)K_v J(q)\dot{q}+\dot{q}^T\tilde{J}^T(q)K_p s(\tilde{x}). \end{aligned} \tag{121}$$

Since the Jacobian matrix contains trigonometric functions of q, $\|J(q)\| \le b_J$ and Eq. 121 can be rewritten as

$$\begin{aligned} \dot{V} \le & -\{\lambda_{\min}(J^T(q)K_v J(q)+D(q))-\gamma b_J\lambda_{\max}(K_v)-\alpha c_0\}\|\dot{q}\|^2 \\ & +\gamma\lambda_{\max}(K_p)\|s(\tilde{x})\|\|\dot{q}\|-\alpha(\lambda_{\min}(K_p)-c_1)\|s(\tilde{x})\|^2 \end{aligned} \tag{122}$$

where γ is defined in Eq. 56. Since

$$2\|s(\tilde{x})\|\|\dot{q}\| \le \|s(\tilde{x})\|^2+\|\dot{q}\|^2, \tag{123}$$

therefore, Eq. 122 is simplified as follows:

$$\begin{aligned} \dot{V} \le & -\left\{\lambda_{\min}(J^T(q)K_v J(q)+D(q))-\gamma b_J\lambda_{\max}(K_v)-\frac{1}{2}\gamma\lambda_{\max}(K_p)-\alpha c_0\right\}\|\dot{q}\|^2 \\ & -\left(\alpha\lambda_{\min}(K_p)-\frac{1}{2}\gamma\lambda_{\max}(K_p)-\alpha c_1\right)\|s(\tilde{x})\|^2. \end{aligned} \tag{124}$$

Equation 124 can be written as follows:

$$\begin{aligned} \dot{V} \le & -\left\{\lambda_{\max}(K_v)\left(\delta_1-\frac{\gamma}{2}(\delta_2+2b_J)\right)-\alpha c_0\right\}\|\dot{q}\|^2 \\ & -\left(\lambda_{\max}(K_p)\left(\alpha\delta_3-\frac{\gamma}{2}\right)-\alpha c_1\right)\|s(\tilde{x})\|^2 \end{aligned} \tag{125}$$

such that

$$\begin{aligned} \delta_1 &= \frac{\lambda_{\min}[J^T(q)K_v J(q)+D(q)]}{\lambda_{\max}(K_v)} \\ \delta_2 &= \frac{\lambda_{\max}(K_p)}{\lambda_{\max}(K_v)} \\ \delta_3 &= \frac{\lambda_{\min}(K_p)}{\lambda_{\max}(K_p)}. \end{aligned} \tag{126}$$

References

Abdallah CT, Dawson D, Dorato P, Jamshidi M (1991) Survey of robust control for rigid robots. IEEE Trans Control Syst Mag 11(2):24–30

Arimoto S (1994) A class of quasi-natural potentials and hyper-stable PID servo-loops for nonlinear robotic systems. Trans Soc Instrum Control Eng 30(9):1005–1012

Arimoto S (1996) Control theory of non-linear mechanical systems: a passivity-based and circuit-theoretic approach. Oxford University Press, New York

Arimoto S, Miyazaki F (1985)Asymptotic stability of feedback control for robot manipulators. In: Proceedings of IFAC symposium on robot control, Barcelona, pp 447–452

Arimoto S, Miyazaki F (1984)Stability and robustness of PID feedback control for robot manipulators of sensory capability. In: Proceedings of the 1st international symposium on robotics research, pp 783–799

Arimoto S, Kawamura S, Miyazaki F (1984) Bettering operation of robots by learning. J Robot Syst 1(2):123–140

Arimoto S, Naniwa T, Parra-Vega V, Whitcomb L (1994) A quasi-natural potential and its role in design of hyper-stable PID servo-loop for robotic systems. In: Proceedings of the CAI Pacific symposium control and industrial automation application, pp 110–117

Berghuis H, Ortega R, Nijmeijer H (1993) A robust adaptive robot controller. IEEE Trans Robot Autom 9(6):825–830

Braganza D, Dixon WE, Dawson DM, Xian B (2005)Tracking control for robot manipulators with kinematic and dynamic uncertainty. In: Proceedings of IEEE conference on decision and control, Seville, pp 5293–5297

Cheah CC (2003)Approximate Jacobian robot control with adaptive Jacobian matrix. In: Proceedings of IEEE international conference on decision and control, Hawaii, pp 5859–5864

Cheah CC, Liaw H (2005) Inverse Jacobian regulator with gravity compensation: stability and experiment. IEEE Trans Robot Autom 21(4):741–747

Cheah CC, Kawamura S, Arimoto S (1998) Feedback control for robotic manipulators with uncertain kinematics and dynamics. In: Proceedings of IEEE international conference on robotics and automation, Leuven, pp 3607–3612

Cheah CC, Kawamura S, Arimoto S (1999a) Feedback control for robotic manipulators with an uncertain Jacobian matrix. J Robot Syst 12(2):119–134

Cheah CC, Hirano M, Kawamura S, Arimoto S (2003) Approximate Jacobian control for robots with uncertain kinematics and dynamics. IEEE Trans Robot Autom 19(4):692–702

Cheah CC, Liu C, Slotine JJE (2004) Approximate Jacobian adaptive control for robot manipulators. In: Proceeding of IEEE international conference on robotics and automation, New Orleans, pp 3075–3080

Cheah CC, Liu C, Slotine JJE (2006a) Adaptive tracking control for robots with unknown kinematic and dynamic properties. Int J Robot Res 25(3):283–296

Chien-Chern Cheah, Chao Liu, Slotine J-JE (2006b) Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models. IEEE Trans Automat Contr 51(6):1024–1029

Cheah CC, Liu C, Slotine JJE (2007) Adaptive vision based tracking control of robots with uncertainty in depth information. In: Proceedings of IEEE conference on roboties and automation, Roma, pp 2817–2822

Cheah CC, Liu C, Slotine JJE (2010) Adaptive Jacobian vision based control for robots with uncertain depth information. Automatica 46:1228–1233

Cheah CC, Kawamura S, Arimoto S, Lee K (1999) PID control for robotic manipulator with uncertain jacobian matrix. In: Proceedings of IEEE international conference on robotics and automation, Detroit, pp 494–499

Craig JJ, Hsu P, Sastry SS (1987) Adaptive control of mechanical manipulators. Int J Robot Res 6(2):10–20

Dixon WE (2007) Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics. IEEE Trans Automat Control 52(3):488–493
Garcia-Rodriguez R, Parra-Vega V (2012) Cartesian sliding PID control schemes for tracking robots with uncertain Jacobian. Trans Inst Meas Control 34(4):448–462
Hutchinson S, Hager GD, Corke P (1996) A tutorial on visual servo control. IEEE Trans Autom Control 12(5):651–670
Ioannou P, Sun J (1996) Robust adaptive control. Prentice-Hall, Englewood Cliffs
Kelly R (1993) Comments on adaptive PD controller for robot manipulators. IEEE Trans Robot Autom 9:117–119
Kelly R (1997) PD control with desired gravity compensation of robotic manipulators: a review. Int J Robot Res 16(5):660–672
Kelly R (1998) Global positioning of robot manipulators via PD control plus a class of nonlinear integral actions. IEEE Trans Autom Control 43(7):934–938
Kelly R, Santibanez V, Loria A (2005) Control of robot manipulators in joint space. Springer–Verlag, London
Koditschek DE (1987)Adaptive techniques for mechanical systems. In: 5th Yale workshop on applications of adaptive systems theory, New Haven, pp 259–265
Lee KW, Khalil H (1997) Adaptive output feedback control of robot manipulators using high gain observer. Int J Control 67(6):869–886
Lewis FL (1996) Neural network control of robot manipulators. Intell Syst Appl 11(3):64–75
Liang X, Huang X, Wang M, Zeng X (2010) Adaptive task-space tracking control of robots without task-space- and joint-space-velocity measurements. IEEE Trans Robot 26(4):733–742
Middleton RH, Goodwin GC (1988) Adaptive computed torque control for rigid link manipulators. Syst Control Lett 10:9–16
Niemeyer G, Slotine JJE (1991) Performance in adaptive manipulator control. Int J Robot Res 10(2):149–161
Ortega R, Spong MW (1989) Adaptive motion control of rigid robots: a tutorial. Automatica 25(6):877–888
Ortega R, Loria A, Kelly R (1995) A semi-globally stable output feedback PI2D regulator for robot manipulators. IEEE Trans Autom Control 40(8):1432–1436
Paden B, Panja R (1988) A globally asymptotically stable PD+ controller for robot manipulator. Int J Control 47(6):1697–1712
Sadegh N, Horowitz R (1990) Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. Int J Robot Res 9(3):74–92
Slotine JJE (1985) The robust control of robot manipulators. Int J Robot Res 4(2):49–61
Slotine JJE, Li W (1987) On the adaptive control of robot manipulators. Int J Robot Res 6(3):49–59
Slotine JJE, Li W (1991) Applied nonlinear control. Prentice Hall, Englewood Cliffs
Spong MW (1992) On the robust control of robot manipulators. IEEE Trans Autom Control 37(11):1782–1786
Spong MW, Hutchinson S, Vidyasagar M (2006) Robot modeling and control. Wiley, New York
Takegaki M, Arimoto S (1981) A new feedback method for dynamic control of manipulators. ASME J Dyn Syst Meas Control 103:119–125
Tomei P (1991) Adaptive PD controller for robot manipulators. IEEE Trans Robot Autom 7:565–570
Wang H, Xie Y (2009) Prediction error based adaptive Jacobian tracking of robots with uncertain kinematics and dynamics. IEEE Trans Automat Control 54(12):2889–2894, art. no. 5332275
Wang H, Liu YH, Zhou D (2007) Dynamic visual tracking for manipulators using an uncalibrated fixed camera. IEEE Trans Robot 23(3):610–617
Wen JT, Bayard D (1988) New class of control laws for robotic manipulators Part 2. Adaptive case. Int J Control 47(5):1387–1406
Ziegler JG, Nichols NB (1942) Optimum settings for automatic controllers. ASME Trans 64:759–768