Robotica (2017) volume 35, pp. 119–142. © Cambridge University Press 2015
doi:10.1017/S0263574715000028

Maximum load of flexible joint manipulators using nonlinear controllers

A. H. Korayem†, M. Irani Rahagi‡, H. Babaee† and M. H. Korayem†,*

†*Robotic Research Laboratory, Center of Excellence in Experimental Solid Mechanics and Dynamics, School of Mechanical Engineering, Iran University of Science and Technology, P.O. Box 18846, Tehran, Iran*
‡*Department of Solid Mechanics, Faculty of Mechanical Engineering, University of Kashan, P.O. Box 87317-51167, Kashan, Iran*

(Accepted January 06, 2015. First published online: February 27, 2015)

SUMMARY
The main innovation of this paper is determining the dynamic load carrying capacity (DLCC) of a flexible joint manipulator (FJM) using a closed form nonlinear optimal control approach. The proposed method is compared with two closed loop nonlinear methods that are usually applied to robotic systems. As a new idea, DLCC of the manipulator is considered as a criterion to compare how well controllers perform point to point mission for the FJMs. The proposed method is compared with feedback linearization (FL) and robust sliding mode control (SMC) methods to show better performance of proposed nonlinear optimal control approach. An optimal controller is designed by solving a nonlinear partial differential equation named the Hamilton–Jacobi–Bellman (HJB) equation. This equation is complicated to solve exactly for complex dynamics so it is solved numerically using an iterative approximation combined with the Galerkin method. In the FL method, angular position, velocity, acceleration and jerk of links are considered as new states to linearize the dynamic equations. In the case of SMC, the dynamic equations of manipulator are changed to the standard form then the Slotine method is used to design the sliding mode controller. Two simulations are performed for a planar and a spatial Puma manipulator and performances of controllers are compared. Finally an experimental test is done on 6R manipulator and the Stereo vision method is used to determine the position and orientation of the end-effector.

KEYWORDS: Flexible joint manipulator; DLCC; Nonlinear control; Optimal control; Point to point.

1. Introduction

In recent years, mass production and high speed and high precision industrial manufacturing has led to the use of mechanical manipulators in automatic production lines. Also, manipulators help us during some special activities such as transportation of heavy parts and doing missions in environments dangerous for human such as space missions. In order to achieve the best performance, flexibility in joints of the manipulator must be considered in both dynamic equations and controller design procedure. This flexibility usually arises from the behavior of the joint transmission elements such as gears, harmonic drives, belts and shafts and can be modeled as a spring. The joint flexibilities between actuators and links of manipulators can cause some vibrations in the trajectory of the end effector.

The main focus of this paper is to design a nonlinear optimal controller for FJMs. Various nonlinear controller design methods have been applied to FJMs. Ozgoli and Taghirad[1] have presented a survey and new classification of advances in the modeling and control of flexible joint robots. The singularly perturbed model, the composite control strategy and the concept of integral manifold have been

* Corresponding author. E-mail: hkorayem@iust.ac.ir

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

explained briefly. In ref. [2], some model-independent control techniques such as direct state feedback backstepping intelligent control, indirect state feedback backstepping intelligent control, and output feedback backstepping intelligent control have been investigated. It has been shown in this paper that these approaches are effective to control a system that is too complicated to be modeled accurately by physical laws. These control techniques have overcome limitations of traditional adaptive control and offline-learning of the intelligent systems. Moberg and Hanssen[3] have investigated the discrete implementation of the FL method for flexible joint robots concerning high speed tracking task. It has been assumed that the state measurement is done perfectly and the FL method has been compared with the feed-forward control. In refs. [4, 5], a robust output feedback control approach using the observer dynamic surface design technique has been proposed for flexible joint robots. The method has needed only position measurements, while for estimation of the link and actuator velocity information, an adaptive observer using self-recurrent wavelet neural network has been developed.

In the existence of uncertainties and disturbances, robust methods can be used to the FJM control system design. Kwan and Yeung[6] have presented a procedure to design a robust adaptive controller for the flexible joint robot using sliding mode technique. The parameter variations and bounded input torques have been considered in the design procedure. Their method has been the combination of both sliding and adaptive techniques; also the global stability was discussed in the sense of Lyapunov to guarantee that the errors of links position and velocity approach to zero. Chaoui *et al*.[7] have used a hybrid neural fuzzy control for the FJM with unknown dynamics. A feed-forward neural network has been applied to approximate the manipulator's inverse dynamics; also a fuzzy sliding mode feedback controller has been employed for the online adaptation of the neural network-based controller. Fateh[8] has devoted a voltage control strategy to the nonlinear tracking control of the electrically driven FJM. This strategy has been independent from the manipulator dynamics and has been a simple robust adaptive controller under both structured and unstructured uncertainty. The proposed control law has used a decentralized controller using feedbacks of the joint position, rotor position, and motor current.

The main novelty of this paper is using nonlinear optimal control methodology to design the state feedback control law. It should be noted that finding DLCC is an optimization problem and it is more appropriate to apply an optimal feedback controller to reduce the amount of torque in joints. If a manipulator carry specified load via less joint torques, the DLCC will be increased.[9] Some papers have focused on using optimal feedback for manipulators. Beeler *et al*.[10] have considered some computational methods to design feedback controller. Power series approximation, state dependent Riccati equation, successive Galerkin approximation, interpolation of two point boundary value problem (TPBVP), and interpolation of iterative solutions are methods which are described, afterwards some examples have been considered to compare these methods for the regulation problem. Cimen[11] has presented a survey on state-dependent Riccati equation (SDRE) method to solve the nonlinear optimal control problem. Also the details of this method, structure, stability, optimality, and state-dependent parameterization have been explained. Korayem *et al*.[12] have used SDRE method to design nonlinear optimal controller and determine the DLCC of rigid mobile manipulators. In ref. [13], the nonlinear system equations have been separated into linear and nonlinear parts using the Maclaurin Expansion and then the optimal output tracking control problem has been solved. The control law has consisted of a feed forward and a feedback terms that have been found by solving the Sylvester equation and approximation using a successive approach, respectively. Some other articles have used the nonlinear partial differential HJB equation.[14] Because there is no analytical solution to this equation several numerical computation methods have been proposed to obtain an approximate solution to this equation.[15]

Another contribution of this paper is determining the DLCC of the FJM by using the nonlinear optimal controller. The DLCC is the maximum allowable load that can be carried by the manipulator in a point to point or a predefined trajectory tasks. The productivity and economic usage of the manipulators will be increased if FJMs work in full load conditions. Wang *et al*.[16] have considered a point to point motion planning for mechanical manipulators to carry the maximum load. This optimization problem has been converted into a direct SQP parameter optimization in which the gradient has been determined analytically. Korayem and Nikoobin[17] have used an open loop optimal control approach to compute the DLCC of FJM in point to point task, also the optimal trajectory of the end effector and the related angular positions and velocities and motor torques have been determined. Based on Pontryagin's minimum principle, the optimal control problem has been changed to a TPBVP.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

In the existence of uncertainties, the open loop solution is generally not useful for practical applications and input torques must be determined using closed loop control laws. Also using closed loop controller, the system responses will be more applicable than using open loop controllers. In ref. [18], the DLCC of a flexible planar mobile manipulator has been calculated using FL method. Korayem *et al.*[19] have used a controller that is divided into two parts: partial FL and SMC. They have finally proposed an algorithm to improve the DLCC of manipulator. Korayem and Pilechian[20] have presented a modification on the DLCC determination algorithm using sliding mode controller. They have applied this algorithm for a planar manipulator and the SCARA robot.

The structure of paper is as follows; in Section 2, the dynamic model of a planar FJM is considered. The structure of nonlinear optimal controller is presented in Section 3. FL and SMC methods are explained in Sections 4 and 5, respectively. The definition of the DLCC of manipulators and corresponding constraints are in Section 6. In Section 7, the implementation of three methods and simulation results for point to point task are presented for two different manipulators: a planar FJM and a flexible joint Puma robot. The DLCC of these manipulators are determined and the end effector trajectories, the angular positions and velocities are compared. Finally in Section 8, an experimental test is provided for a 6R manipulator to verify the simulation results.

2. Dynamic Model of Flexible Joint Manipulator

Dynamic equations of motion for an n-link FJM can be obtained using Lagrange approach as:

$$\begin{aligned} D(q_1)\ddot{q}_1 + C(q_1, \dot{q}_1) + G(q_1) + K(q_1 - q_2) &= 0 \\ J\ddot{q}_2 + K(q_2 - q_1) &= u, \end{aligned} \tag{1}$$

where q_1 and q_2 are $n \times 1$ vectors of angular position of the links and motors, respectively. Also, $D(q)$ is the inertia matrix, J is moment of inertia matrix of the motors, $C(q, \dot{q})$ is the $n \times 1$ centrifugal and Coriolis forces, $G(q)$ is the $n \times 1$ gravity effects, K is $n \times n$ stiffness matrix duo to joint elasticity and u mentions the motor torques. For the FJM that has two links and is shown in Fig. 1, q_1 and q_2 are $q_1^T = [\theta_1, \theta_3]$ and $q_2^T = [\theta_2, \theta_4]^T$ where θ_1 and θ_3 are the links rotation angle and θ_2 and θ_4 are the rotor angles. Also other parameters in Eq. (1) are as follows:

$$D = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \quad K = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \quad J = \begin{bmatrix} J_{r1} & 0 \\ 0 & J_{r2} \end{bmatrix}, \tag{2}$$

$$C = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} -a_3 \sin(\theta_3)(2\dot{\theta}_1\dot{\theta}_3 + \dot{\theta}_3^2) \\ a_3 \sin(\theta_3)\dot{\theta}_1^2 \end{bmatrix}, \tag{3}$$

$$G = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} g(a_4 \cos(\theta_1) + a_5 \cos(\theta_1 + \theta_3)) \\ g a_5 \cos(\theta_1 + \theta_3) \end{bmatrix}, \tag{4}$$

$$u = [\tau_{m1}, \tau_{m2}]^T. \tag{5}$$

Elements of matrix D in Eq. (2) are expressed as:

$$\begin{aligned} d_{11} &= a_1 + a_2 + 2a_3 \cos(\theta_3) \\ d_{12} &= a_2 + a_3 \cos(\theta_3) \\ d_{22} &= a_2. \end{aligned} \tag{6}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

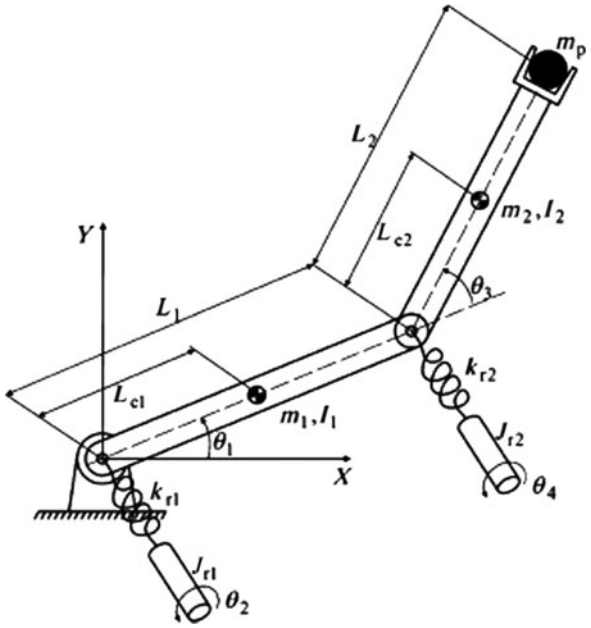


Fig. 1. Schematic view of the flexible joint manipulator.[9]

In these equations, the constant parameters are as below:

$$\begin{aligned} a_1 &= I_1 + m_1 L_{c1}^2 + (m_2 + m_p) L_1^2 \\ a_2 &= I_2 + I_p + m_2 L_{c2}^2 + m_p L_2^2 \\ a_3 &= L_1 (m_2 L_{c2} + m_p L_2) \\ a_4 &= m_1 L_{c1} + (m_2 + m_p) L_1 \\ a_5 &= m_2 L_{c2} + m_p L_2. \end{aligned} \tag{7}$$

For the state-space representation of an n-link FJM, the angular position and velocity of links and rotors are selected as the state variables:

$$x_1 = q_1; \quad x_2 = \dot{q}_1; \quad x_3 = q_2; \quad x_4 = \dot{q}_2. \tag{8}$$

Using these states, the dynamic equations of motion of manipulator can be transferred to the state space model:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -D^{-1} [h + K(x_1 - x_3)] \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = J^{-1} K (x_1 - x_3) + J^{-1} u \end{cases}. \tag{9}$$

This model mentions that the system has nonlinear form. Parameter h in Eq. (9) is defined for simplicity as below:

$$h = C + G. \tag{10}$$

3. Structure of Nonlinear Optimal Controller

Nonlinear state space equations of the n-link FJM (Eq. (9)) can be written in below form:

$$\dot{x} = f(x) + g(x) u(x), \tag{11}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

where $f(x)$ and $g(x)$ have following forms:

$$f(x) = \begin{bmatrix} x_2 \\ -D^{-1}[h + K(x_1 - x_3)] \\ x_4 \\ J^{-1}K(x_1 - x_3) \end{bmatrix}, \tag{12}$$

$$g(x) = \begin{bmatrix} 0_{6\times 2} \\ J^{-1} \end{bmatrix}. \tag{13}$$

Consider a scalar cost function:

$$J(x) = \int_0^\infty (x^T Q x + u^T(x) R u(x)) dt. \tag{14}$$

In Eqs. (13) and (14), $x \in R^{4n}$, $u \in R^n$ and Q and R are constant $4n \times 4n$ symmetric positive semi-definite and $n \times n$ symmetric positive definite weighted matrices. These matrices can be defined as the weighted coefficients to determine importance of each variable. The most common procedure to specify the matrices Q and R is searching by trial and error while diagonal matrices are common. The optimal control problem is subject to minimize the cost function $J(x)$ via finding a state feedback control $u(x)$. For a linear system with linear f and g, the optimal control must be found through solving the Riccati equation. According to the calculus of variations, the optimal control law is found through solving the HJB partial differential equation.[14] This equation has the following form:

$$J_x^{*^T}(f + gu^*) + x^T Q x + u^{*^T} R u^* = 0. \tag{15}$$

The optimal control law is determined from:

$$u^* = -\frac{1}{2} R^{-1} g^T J_x^*. \tag{16}$$

The HJB equation, a nonlinear partial differential equation, is very difficult to solve analytically. So, it must be solved numerically and a useful method is the iterative solution. In this method, Eqs. (15) and (16) are solved together with an iterative process.[10] This iterative process is started by an initial control law $u^{(0)}(x)$ with a stability region Ω and then the iterative algorithm that is shown in Fig. 2 is performed to find the solution of the HJB equation.

In Fig. 2, u_ε^* is the approximate value of u^* with ε accuracy. It should be noted that in this algorithm, the equation that is solved iteratively is named the generalized HJB equation (GHJB). This equation is of the form:

$$\frac{\partial J^{(i)^T}}{\partial x}\left(f + gu^{(i)}\right) + x^T Q x + u^{(i)^T} R u^{(i)} = 0, \tag{17}$$

where u is a known function of x and it is computed in the previous iteration. The GHJB equation is difficult to solve analytically thus the Galerkin method can be used for numerically solving.

In the Galerkin method, N suitable basis functions such as $\{\phi_j(x)\}_{j=1}^N$ are selected. Then a finite approximation of solution to the GHJB equation is considered to have the below form:

$$J_N(x) = \sum_{j=1}^{N} c_j \phi_j(x). \tag{18}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

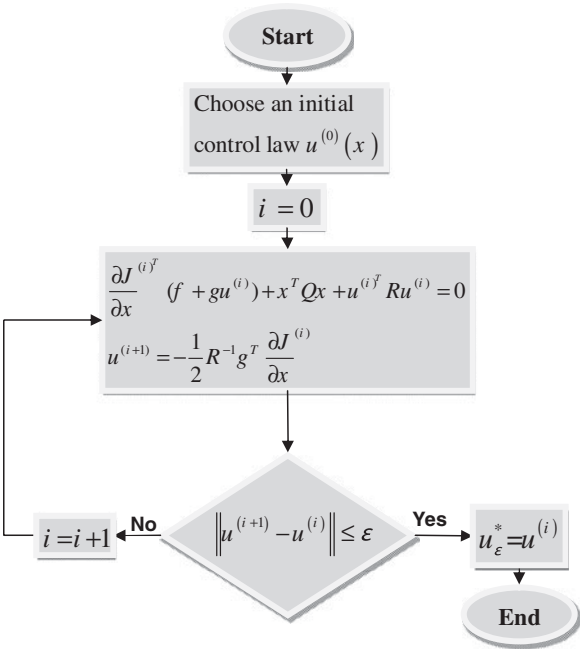


Fig. 2. Iterative solution algorithm.

The coefficients c_j are determined by solving set of N algebraic equations:[21]

$$\int_\Omega \left(\frac{\partial J_N(x)}{\partial x}(f + gu) + x^T Qx + u^T Ru \right). \quad \phi_j(x)\, dx = 0 \quad j = 1, ..., N. \tag{19}$$

This equation can be rewritten in the simple and suitable following form using notation $C_N = [c_1, ..., c_N]^T$ and $\Phi_N = [\phi_1, ..., \phi_N]^T$:[22]

$$\left(A_f + A_u\right) C_N = B_Q + B_u, \tag{20}$$

where:

$$A_f = \int_\Omega \Phi_N f^T \nabla \Phi_N^T dx, \tag{21}$$

$$A_u = \int_\Omega \Phi_N u^T g^T \nabla \Phi_N^T dx, \tag{22}$$

$$B_Q = \int_\Omega x^T Qx \Phi_N dx, \tag{23}$$

$$B_u = -\int_\Omega \Phi_N u^T R^{-1} u dx. \tag{24}$$

If in Eq. (19) u is a known function of x that is shown as $u_N(x)$:

$$u_N(x) = -\frac{1}{2} R^{-1} g^T \frac{\partial J_N(x)}{\partial x} = -\frac{1}{2} R^{-1} g^T \nabla \Phi^T c_N, \tag{25}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

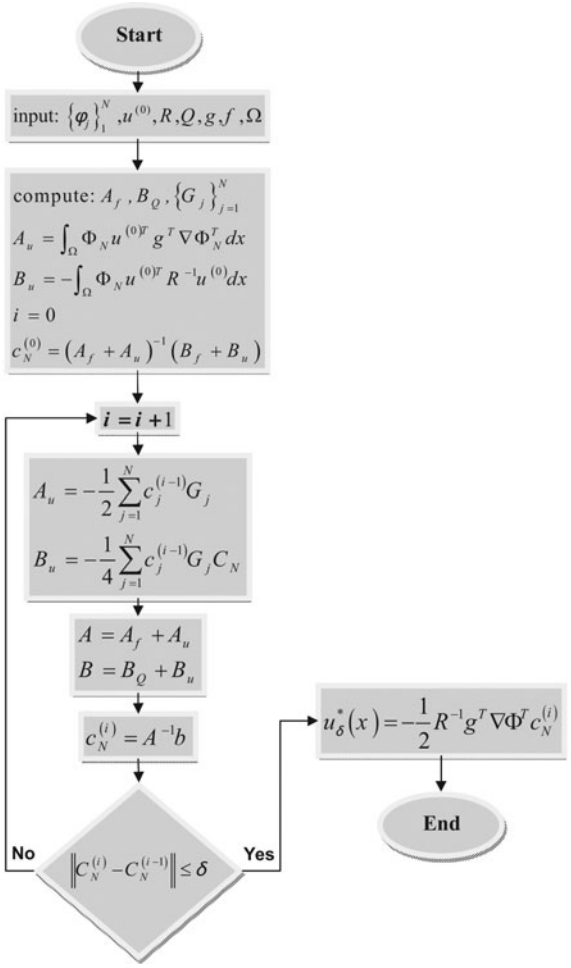


Fig. 3. Algorithm for numerical solution to the GHJB equation.

then A_u and B_u can be calculated as $A_u = -\frac{1}{2}\sum_{j=1}^N c_j G_j$ and $B_u = -\frac{1}{4}\sum_{j=1}^N c_j G_j C_N$ where G_j is:

$$G_j = \int_\Omega \Phi_N \frac{\partial \phi_j^T}{\partial x} g R^{-1} g^T \nabla \Phi_N^T dx. \tag{26}$$

Finally the vector of unknown coefficients C_N will be computed using below simple equation:

$$C_N = A^{-1} B, \tag{27}$$

where $A = A_f + A_u$ and $B = B_Q + B_u$. This procedure can be applied to the iterative solution algorithm (Fig. 2) in order to derive new algorithm for computing a sequence of coefficients vector C_N and finding $u_\delta^*(x)$ that is an approximate value of $u_\varepsilon^*(x)$. A new algorithm is shown in Fig. 3 and it continues until $C_j^{(i)}$ and $C_j^{(i+1)}$ be closed to each other with δ accuracy.

4. Feedback Linearization Method

In the FL method, one seeks a global diffeomorphism (a feedback linearizing transformation) in which the nonlinearities of the system equations can be exactly canceled by one or more of the inputs. In decoupled system equations, the system is linearized in such a way that the resulted linear system is composed of the subsystems, each of which is affected by only a single one of the outer loop control inputs.[23]

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

In many mechanical cases, the appropriate state variables are defined so that the system equation can be linearized by the nonlinear feedback of position, velocity, acceleration, and jerk of the links. The following feedback linearizing transformation is selected for the n-link FJM:[23]

$$\begin{aligned}
y_1 &= x_1 \\
y_2 &= \dot{y}_1 = x_2 \\
y_3 &= \dot{y}_2 = \dot{x}_2 = -D^{-1}\left[h + K(x_1 - x_3)\right] \\
y_4 &= \dot{y}_3 = -\frac{dD}{dt}\left[h + K(x_1 - x_3)\right] \\
&\quad -D^{-1}\left[\frac{\partial h}{\partial x_1}x_2 + \frac{\partial h}{\partial x_2}\left[-D^{-1}\left[h + K(x_1 - x_3)\right]\right] + K(x_2 - x_4)\right].
\end{aligned} \tag{28}$$

Right-hand side of y_4 in Eq. (28) can be written in the below form:

$$\begin{aligned}
y_4 &= -\frac{dD}{dt}\left[h + K(x_1 - x_3)\right] \\
&\quad - D^{-1}\left[\frac{\partial h}{\partial x_1}x_2 + \frac{\partial h}{\partial x_2}\left[-D^{-1}\left[h + K(x_1 - x_3)\right]\right] + Kx_2\right] + D^{-1}Kx_4,
\end{aligned} \tag{29}$$

$$y_4 = f(x_1, x_2, x_3) + D^{-1}Kx_4. \tag{30}$$

In Eq. (30), $f(x_1, x_2, x_3)$ is defined as the right-hand side of y_4 except the last term and it depends only on x_1, x_2 and x_3. This function is:

$$f(x_1, x_2, x_3) = -\frac{dD}{dt}\left[h + K(x_1 - x_3)\right] - D^{-1}\left[\frac{\partial h}{\partial x_1}x_2 + \frac{\partial h}{\partial x_2}\left[-D^{-1}\left[h + K(x_1 - x_3)\right]\right] + Kx_2\right]. \tag{31}$$

Computing derivatives of Eq. (28), the system equations can be changed into feedback linearizable form:

$$\begin{aligned}
\dot{y}_1 &= y_2 \\
\dot{y}_2 &= y_3 \\
\dot{y}_3 &= y_4 \\
\dot{y}_4 &= \dot{y}_3 = \frac{df}{dt} + \frac{d}{dt}\left[D^{-1}Kx_4\right].
\end{aligned} \tag{32}$$

Last row in Eq. (32) is of the form:

$$\begin{aligned}
\dot{y}_4 &= \frac{\partial f}{\partial x_1}x_2 - \frac{\partial f}{\partial x_2}D^{-1}\left[h + K(x_1 - x_3)\right] \\
&\quad + \frac{\partial f}{\partial x_3}x_4 + \frac{dD^{-1}}{dt}Kx_4 + D^{-1}K\left[J^{-1}K(x_1 - x_3) + J^{-1}u\right].
\end{aligned} \tag{33}$$

Defining two below expressions:

$$\begin{aligned}
a(x) &= \frac{\partial f}{\partial x_1}x_2 - \frac{\partial f}{\partial x_2}D^{-1}\left[h + K(x_1 - x_3)\right] + \frac{\partial f}{\partial x_3}x_4 \\
&\quad + \frac{dD^{-1}}{dt}Kx_4 + D^{-1}KJ^{-1}K(x_1 - x_3) \\
b(x) &= D^{-1}KJ^{-1}.
\end{aligned} \tag{34}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Equation (33) can be rewritten as:

$$\dot{y}_4 = a(x) + b(x)\,u. \tag{35}$$

Equation (32) has a linear form by using the below feedback linearizing law:

$$u = b^{-1}(x)\,[v - a(x)]\,. \tag{36}$$

With the nonlinear transformation 28 and nonlinear feedback 36, the transformed system has linear form:

$$\dot{y} = \begin{bmatrix} 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{bmatrix} y + \begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix} v. \tag{37}$$

Finally, the linear control laws such as PID or standard LQR are used to determine input vector v that is given by:

$$v = \dot{y}_4^d - k_1\left(y_1 - y_1^d\right) - k_2\left(y_2 - y_2^d\right) - k_3\left(y_3 - y_3^d\right) - k_4\left(y_4 - y_4^d\right). \tag{38}$$

It should be noted that the feedback control law (Eq. (38)) is stated in terms of new variable y.

5. Sliding Mode Control Method

Consider the dynamic equations of the n-link FJM (Eq. (1)). The objective of this section is to design a control law u such that the position and velocity of the end-effector track their desired values. On the other hand, the joint angular positions and velocities, q_1 and $\dot{q}_1$, is to track their predefined desired values, q_{1d} and $\dot{q}_{1d}$. This objective is to be achieved under any bounded uncertainty in the system's dynamics. The dynamic equation in term of q_1 and q_2 is in the following form:

$$D(q_1)\,\ddot{q}_1 + h(q_1, \dot{q}_1) + K(q_1 - q_2) = 0, \tag{39}$$

where $h(q_1, \dot{q}_1) = C(q_1, \dot{q}_1) + G(q_1)$. Computing derivatives of Eq. (39) leads to the following equations:

$$D(q_1)\,q_1^{(3)} + \dot{D}(q_1)\,\ddot{q}_1 + \dot{h}(q_1, \dot{q}_1) + K(\dot{q}_1 - \dot{q}_2) = 0, \tag{40}$$

$$D(q_1)\,q_1^{(4)} + 2\dot{D}(q_1)\,q_1^{(3)} + \ddot{D}(q_1)\,\ddot{q}_1 + \ddot{h}(q_1, \dot{q}_1) + K(\ddot{q}_1 - \ddot{q}_2) = 0. \tag{41}$$

$\ddot{q}_2$ can be computed from Eq. (1):

$$\ddot{q}_2 = J^{-1}(u - K(q_2 - q_1)). \tag{42}$$

Equation (41) can be refreshed using Eq. (42):

$$D(q_1)\,q_1^{(4)} + c - KJ^{-1}k(q_1 - q_2) = KJ^{-1}u, \tag{43}$$

where:

$$c = 2\dot{D}(q_1)\,q_1^{(3)} + \ddot{D}(q_1)\,\ddot{q}_1 + \ddot{h}(q_1, \dot{q}_1) + K\ddot{q}_1. \tag{44}$$

Equation (43) can be rewritten as:

$$q_1^{(4)} = -D^{-1}(q_1)\,g + D^{-1}(q_1)\,KJ^{-1}u. \tag{45}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

g in Eq. (45) is:

$$g = 2\dot{D}(q_1)q_1^{(3)} + (KJ^{-1}KD(q_1) + \ddot{D}(q_1) + K)\ddot{q}_1 + \ddot{h}(q_1, \dot{q}_1) + KJ^{-1}Kh(q_1, \dot{q}_1). \quad (46)$$

Equation (45) is in the well-known form that is suitable to use the SMC method:

$$x^{(n)} = f\left(x, \dot{x}, ..., x^{(n-1)}\right) + g\left(x, \dot{x}, ..., x^{(n-1)}\right)u. \quad (47)$$

The sliding surface is considered to be as:

$$S = \left(\frac{d}{dt} + \lambda\right)^{n-1} e \qquad e = x - x_d. \quad (48)$$

Finally the SMC law is as follow:

$$u = JK^{-1}D^{-1}(q_1)\left(-k.\tanh(S) + q_d^{(4)} + M^{-1}g\right). \quad (49)$$

It will be shown that utilizing this control law results some robustness against load uncertainties.

6. Dynamic Load Carrying Capacity

For a predefined trajectory, the DLCC of a manipulator is the maximum allowable load that the manipulator can carry in executing the trajectory with an acceptable tracking accuracy. The flexibility of joints may degrade the precision of motion on the desired trajectory. Two main constraints, which bound the DLCC of manipulators, are actuators and accuracy constraints.[9] The DLCC of the manipulator is computed with respect to the constraints on the motors and tracking error. Upper and lower limits of the motor torques can be computed as:

$$\tau_{\max} = \tau_s - \frac{\tau_s}{\omega_s}\omega, \quad (50)$$

$$\tau_{\min} = -\tau_s - \frac{\tau_s}{\omega_s}\omega. \quad (51)$$

In Eqs. (50) and (51), τ_s is the stall torque of motor, ω is the angular velocity and ω_s is the no load speed of motor. Also the tracking error is calculated as:

$$E = \sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2}, \quad (52)$$

where x, y, z are the components of the actual position of end-effector and x_d, y_d, z_d are the components of the desired position. The DLCC of the manipulator based on the optimal control method is determined using a new algorithm as shown in Fig. 4. This algorithm can be modified easily for other control methods.

7. Simulation Results

7.1. Flexible joint planar manipulator

In this section, simulations for the planar manipulator with flexible joints are performed using three described methods. Three nonlinear feedback controllers are designed for the manipulator using optimal, FL and SMC methods and then the DLCC of the manipulator is determined for point to point motion. Consider a manipulator as shown in Fig. 1 with parameters of Table I.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Table I. Parameters of flexible joint manipulator[9].

Parameter	Value	Unit
Length of links	$L_1 = L_2 = 1$	M
Mass	$m_1 = m_2 = 2$	Kg
Moment of inertia	$I_1 = I_2 = 0.166$; $J_{r1} = J_{r2} = 2$	$\frac{kg}{m^2}$
Max. no load speed	$\omega_{s1} = \omega_{s2} = 5.6$	$\frac{rad}{sec}$
Actuator stall torque	$\tau_{s1} = \tau_{s2} = 104$	N.m
Spring factor	$k_{r1} = k_{r2} = 1000$	$\frac{N.m}{rad}$

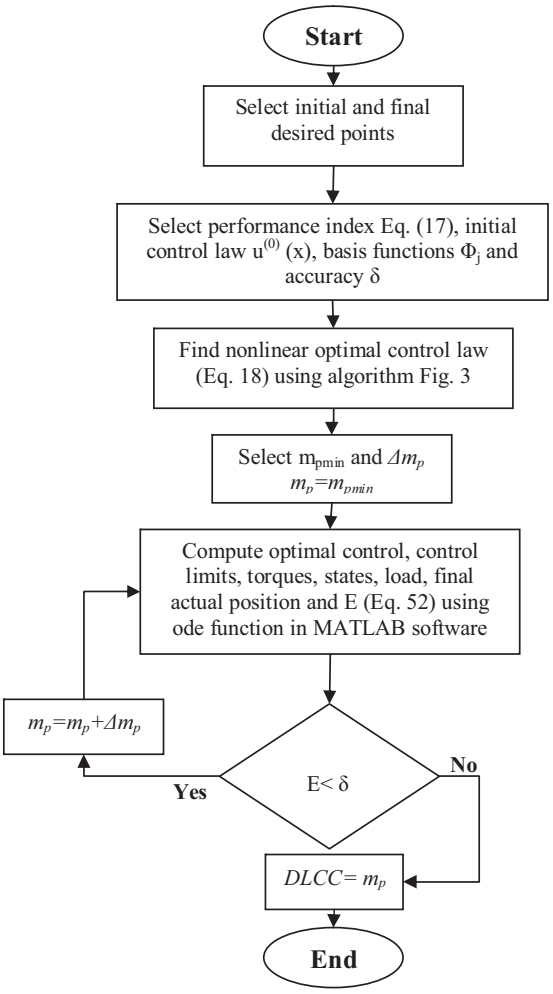


Fig. 4. Algorithm for calculating the DLCC.

The flexible manipulator must carry the maximum load from start point *A(0,1.73)* to end point *B(1,0)*. *R* and *Q* matrices that appeared in Eq. (14), are chosen as:

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix}; \quad R = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}. \tag{53}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Table II. DLCC of manipulator.

Controller type	DLCC (kg)
Optimal control	*16.4*
Feedback linearization	*6*
Sliding mode control	*10.5*

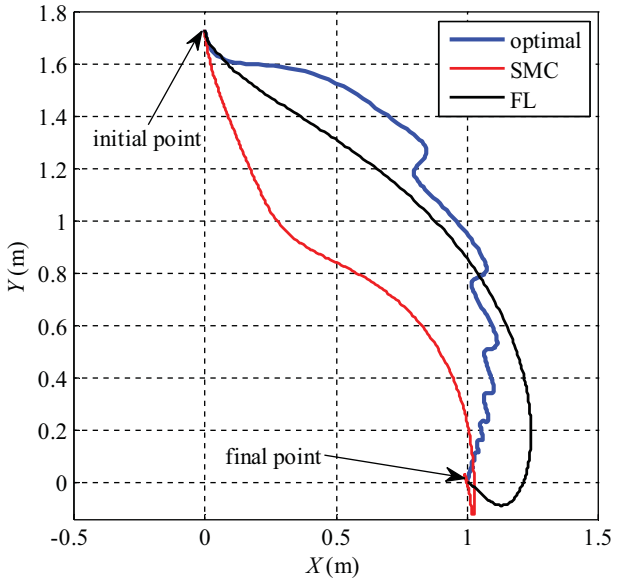


Fig. 5. End effector trajectory.

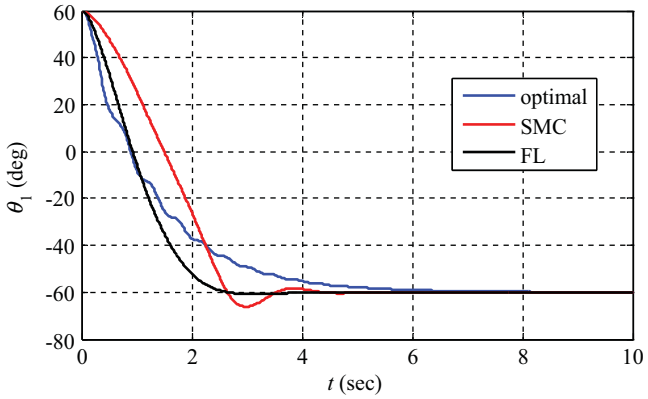


Fig. 6. Angular position of first link.

Three simulations are implemented using three control laws (Eqs. (25), (38) and (49)) for optimal, FL, and SMC methods, respectively. The trajectories of the end effector are shown in Fig. 5 and variations of the angular position and velocities for links are displayed in Figs. 6–9.

The DLCC is calculated for this point to point motion defining *0.001 m* as the maximum allowable error in the final position. All three values of the DLCC using the three nonlinear control methods are listed in Table II.

It is illustrated from Fig. 5, that in the case of optimal control, some vibrations are appeared in the end effector trajectory, also Figs. 6 and 7 show that these vibrations exist in the angular position of the links, and some oscillations are observed in the angular velocities (Figs. 8 and 9), that are originated from the vibratory nature of Eq. (1). It should be noted that the velocities can be kept smooth using optimal controller with less DLCC. Table II shows that the manipulator can carry more

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

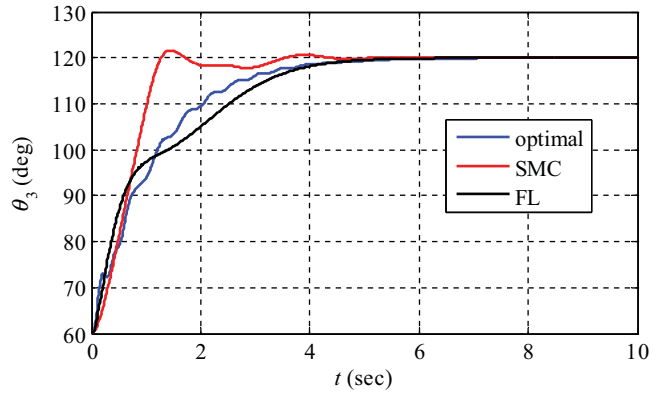


Fig. 7. Angular position of second link.

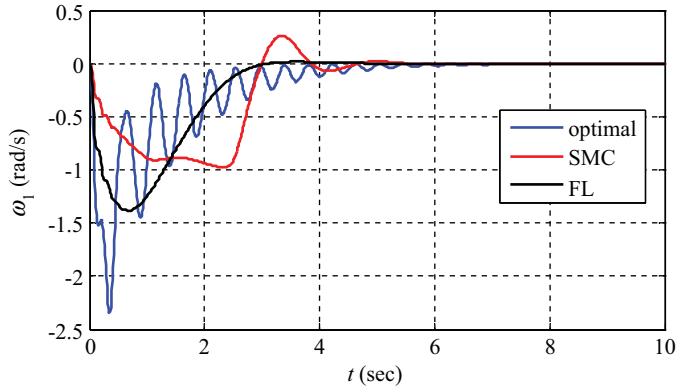


Fig. 8. Angular velocity of first link.

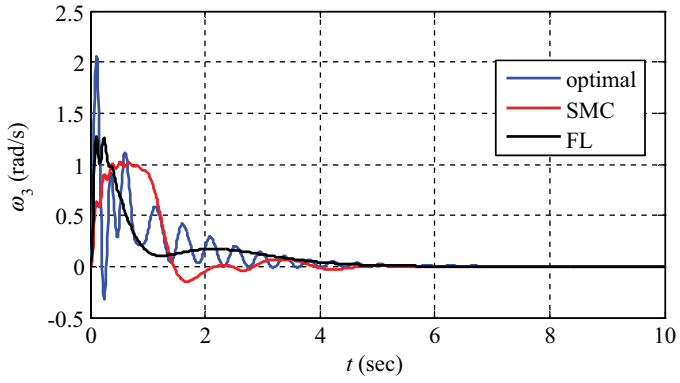


Fig. 9. Angular velocity of second link.

load than SMC or FL cases if optimal controller is used. Another result that can be mentioned is that the DLCC for FL case is less than other cases because of the sensitivity of this method to deviations in the dynamic parameters (load) and disturbances force on the system.

7.2. Flexible spatial manipulator (flexible Puma)

In this section, performances of three nonlinear controllers are compared for point to point motion of flexible joint Puma manipulator (Fig. 10) and the DLCC of this robot is determined. Three simulations are performed for a point to point motion between initial point *A(−0.7, 0, 1.5)* to end point *B(1.6, −0.5, 0.8)*. Characteristics of flexible Puma robot are focused in Table III.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Table III. Parameters of flexible Puma manipulator.

Parameter	Value	Unit
Length of links	$L_1, L_2, L_3 = 1$	m
Mass	$m_1, m_2, m_3 = 1$	kg
Moment of inertia	$I_1 = I_2 = I_3 = 0.166$; $J_{r1} = J_{r2} = J_{r3} = 2$	$\frac{kg}{m^2}$
Max. no load speed	$\omega_{s1} = \omega_{s2} = \omega_{s3} = 5.6$	$\frac{rad}{sec}$
Actuator stall torque	$\tau_{s1} = \tau_{s2} = \tau_{s3} = 104$	N.m
Spring factor	$k_{r1}, k_{r2}, k_{r3} = 1000$	$\frac{N.m}{rad}$

Table IV. DLCC of manipulator.

Controller type	DLCC (kg)
Optimal control	*8.1*
Feedback linearization	*4*
Sliding mode control	*4.6*

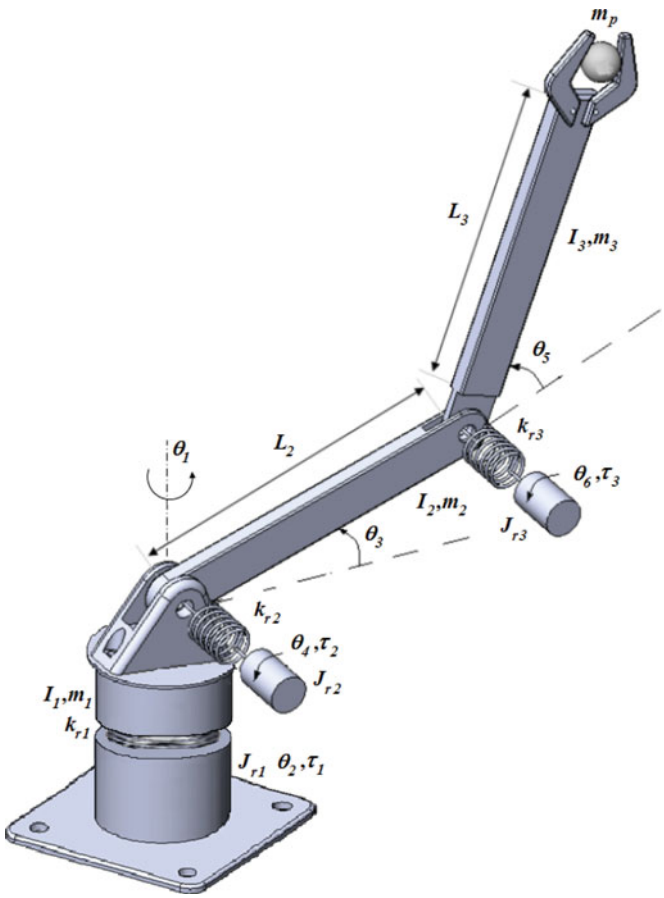


Fig. 10. Flexible joint Puma.

The end effector trajectories are shown in Fig. 11; also 2D views are displayed in Fig. 12 that show better performance of optimal control method than other methods to reach the final point. Angular position and angular velocity of the links are displayed in Figs. 13–18.

Values of the DLCC with *0.001 m* allowable error in the end position are listed in Table IV that show better performance of optimal controller and some oscillations appeared in angular position of the links.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

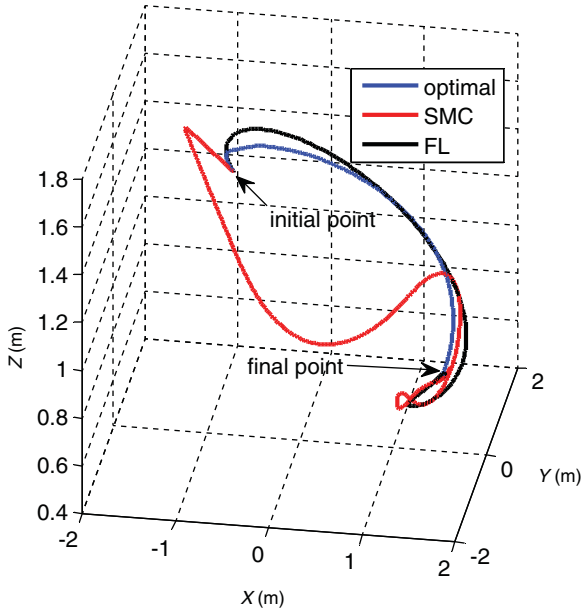


Fig. 11. End effector trajectory.

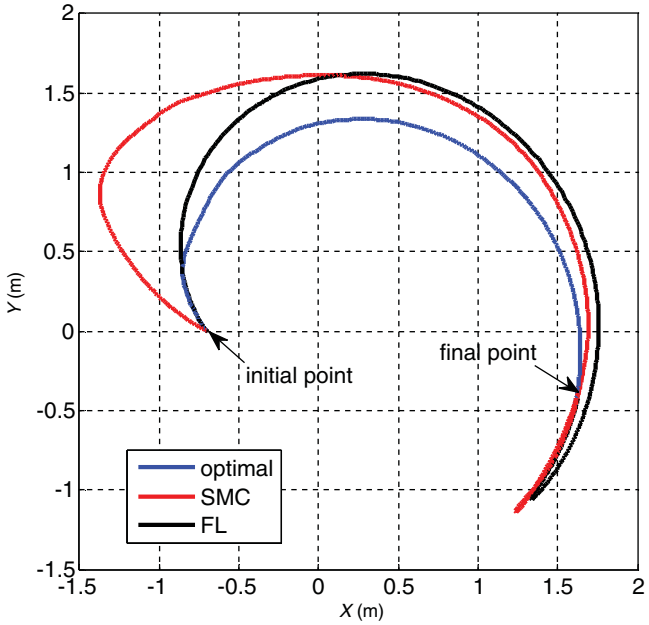


Fig. 12. End effector trajectory, 2D view.

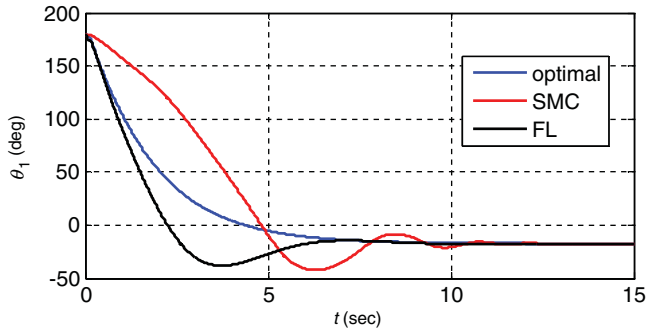


Fig. 13. Angular position of first link.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

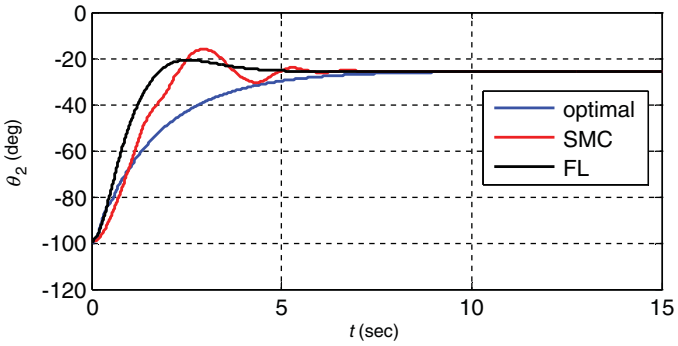


Fig. 14. Angular position of second link.

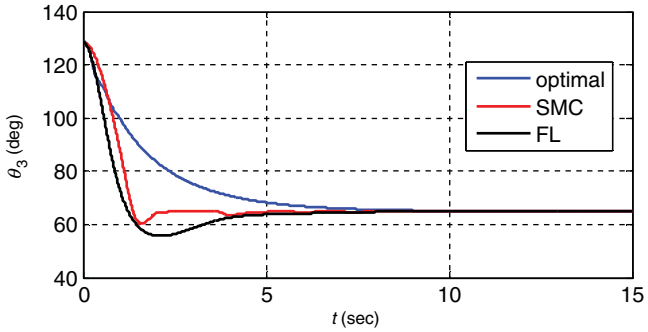


Fig. 15. Angular position of third link.

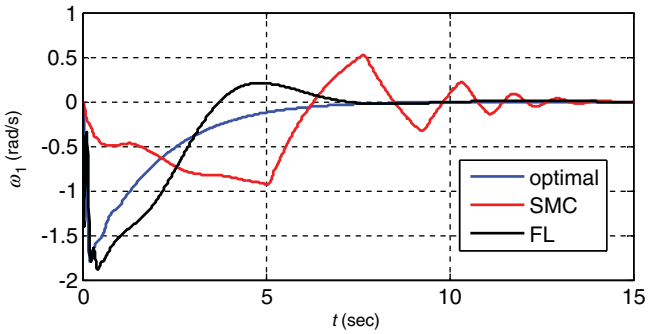


Fig. 16. Angular velocity of first link.

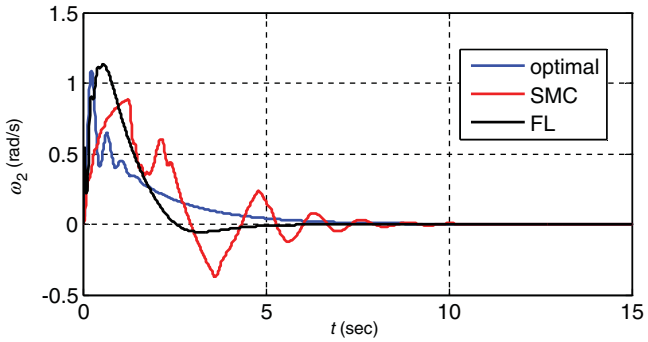


Fig. 17. Angular velocity of second link.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Table V. Denavit–Hartenberg parameters of 6R.

Joint	a_i (mm)	d_i (mm)	αi (deg)	θi (deg)	Motion of
1	36.5	438	−90	θ_1	Link 1
2	251.5	0	0	θ_2	Link 2
3	125	0	0	θ_3	Link 3
4	92	0	90	θ_4	Gripper YAW
5	0	0	−90	θ_5	Gripper PITCH
6	0	152.8	0	θ_6	Gripper ROLL

Table VI. Characteristics of motors.

Joint	U_s(N.m)	ω_s(rad/s)
1	114	1.32
2	98	4.19
3	382.2	0.73
4	19	450.29
5	40.4	9.01
6	40.4	9.01

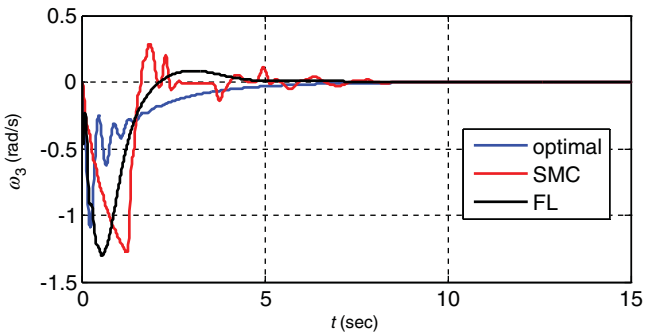


Fig. 18. Angular velocity of third link.

8. Experimental Test on the 6R Manipulator

To verify the simulation results, the proposed optimal control method is implemented on the 6R manipulator that is shown in Fig. 19. Denavit–Hartenberg parameters of the 6R manipulator and the characteristics of the motors are mentioned in Tables V and VI, respectively.

8.1. Vision system

Stereo vision measurement system, also two images, object, image (local), and reference frames are shown in Fig. 20.

The transformation between the object position in the reference frame and each of the local frames can be mentioned as:

$$M^T = RT P^T. \tag{54}$$

P and M are the local and global positions of object and have the following forms:

$$P = [x, y, z, 1], \quad M = [X, Y, Z, 1]. \tag{55}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press



Fig. 19. 6R manipulator.

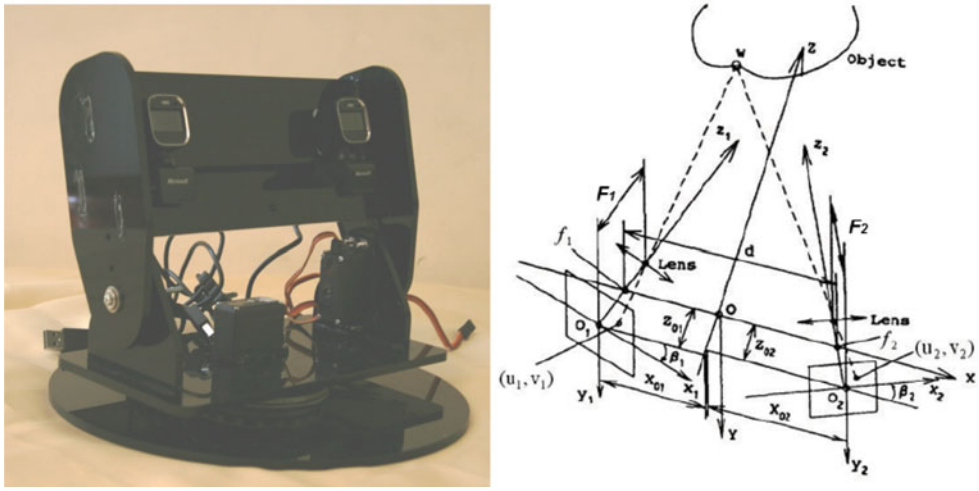


Fig. 20. Stereo vision measurement system and coordinates.

Also R and T in Eq. (54) are the rotation and translation matrices, respectively and have the following forms:

$$R = \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad T = \begin{bmatrix} 1 & 0 & 0 & X_0 \\ 0 & 1 & 0 & Y_0 \\ 0 & 0 & 1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{56}$$

In these equations β is the rotation of optical axis of the camera and $[X_0, Y_0, Z_0]$ are position coordinates of the center of local frame in the reference frame. Considering F as focal length, the perspective relations for any image can be written as:

$$\frac{x}{u} = -\frac{z-F}{F}; \quad \frac{y}{v} = -\frac{z-F}{F}. \tag{57}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

By substituting Eq. (57) in Eq. (54), 3D position of the object in the reference frame can be computed as:

$$\begin{cases} X = \dfrac{u_1 F_1 - m_1 X_{01} - n_1 (Z + Z_{01})}{m_1} \\ Y = v_1 - Y_{01} - \dfrac{v_1}{F_1} [(X - X_{01}) \sin \beta_1 + (Z + Z_{01}) \cos \beta_1] \\ Z = \dfrac{m_1 m_2 (X - X_{01}) + m_1 F_2 u_2 - m_2 F_1 u_1 + m_2 n_1 Z_{01} - m_1 n_2 Z_{02}}{m_1 n_2 - m_2 n_1} \end{cases} . \tag{58}$$

New parameters in Eq. (58) are as follows:

$$\begin{aligned} m_1 &= F_1 \cos \beta_1 + u_1 \sin \beta_1 \\ m_2 &= F_2 \cos \beta_2 - u_2 \sin \beta_2 \\ n_1 &= u_1 \cos \beta_1 - F_1 \sin \beta_1 \\ n_2 &= u_2 \cos \beta_2 + F_2 \sin \beta_2, \end{aligned} \tag{59}$$

where [*ui, vi*] are coordinates of the object projected onto *i*th image screen. If the two cameras are placed parallel to each other ($\beta = 0$) then the following simple relations can be considered:

$$\begin{aligned} Y_{01} &= Y_{02} = 0 \\ \beta_1 &= \beta_2 = 0 \\ F_1 &= F_2 = F \\ X_{01} &= X_{02} = \frac{B}{2}. \end{aligned} \tag{60}$$

Finally, Eq. (58) can be rewritten as follow:

$$\begin{aligned} X &= \frac{F - Z}{F} u_1 - \frac{B}{2} \\ Y &= \frac{F - Z}{F} v_1 \\ Z &= F - \frac{BF}{u_1 - u_2}. \end{aligned} \tag{61}$$

$u_1 - u_2$ in Eq. (61) is the difference between the x coordinates of two corresponding pixels that is named disparity.

According to above method and using stereo vision setup, one can calculate the position of three points of the robot end effector which are determined in Fig. 21 as P_1, P_2 and P_3.

The direction vectors $\vec{x}_b$, $\vec{y}_b$ and $\vec{z}_b$ can be represented as:

$$\begin{aligned} \vec{x}_b &= \frac{P_2 - P_1}{\|P_2 - P_1\|} \\ \vec{z}_b &= \frac{(P_2 + P_1)/2 - P_3}{\|(P_2 + P_1)/2 - P_3\|} \\ \vec{y}_b &= \vec{z}_b \times \vec{x}_b. \end{aligned} \tag{62}$$

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

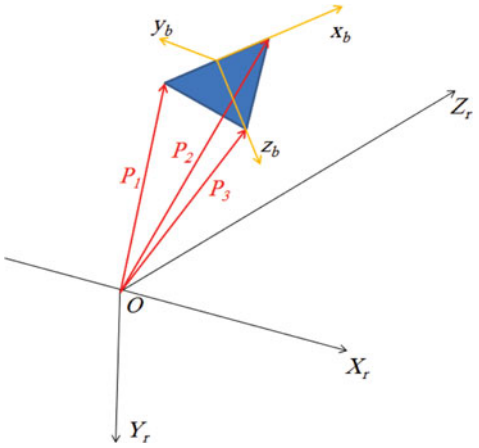


Fig. 21. Three point on end effector.

Considering the rotation matrices $R_{Z,\phi}$, $R_{Y,\theta}$ and $R_{X,\psi}$ for roll, pitch and yaw angles, the rotation matrix between the reference and the body frames can be mentioned as below equations:

$$\begin{bmatrix} \vec{x}_b & \vec{y}_b & \vec{z}_b \end{bmatrix} = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix}, \tag{63}$$

$$\begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix} = \begin{bmatrix} C_\phi C_\theta & C_\phi S_\theta S_\psi - S_\phi C_\psi & C_\phi S_\theta S_\psi + S_\phi S_\psi \\ S_\phi C_\theta & S_\phi S_\theta S_\psi - C_\phi C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi \\ -S_\theta & C_\theta S_\psi & C_\theta C_\psi \end{bmatrix}, \tag{64}$$

where $C_i = \cos(i)$ and $S_i = \sin(i)$.

Therefore the roll, pitch, and yaw angles of the robot end effector are determined as:

$$\begin{aligned} \theta_x &= A\tan 2\left(\frac{X_y}{X_x}\right) \\ \theta_y &= A\tan 2\left(\frac{-X_z}{\sqrt{1-X_z^2}}\right) \\ \theta_z &= A\tan 2\left(\frac{Y_z}{Z_z}\right). \end{aligned} \tag{65}$$

8.2. Experimental test

A simulation is performed for a point to point motion between initial point $A(0.2, -0.1, 0.4)$ and end point $B(-0.2, -0.2, 0.3)$. A stereo vision measurement setup as shown in Fig. 20 is used to measure 3D position and orientation of the end effector. The accuracy of this measurement method is less than 1 mm. Another way to measure 3D position is using rotary encoders to measure the angular position of the joints and then finding the end effector position via direct kinematic equations. But the errors in encoders' outputs and geometric errors can affect on calculations and produce errors in resulted end effector position. The resulted path from both experimental and simulation tests are compared in Fig. 22.

Simulated and experimental components of 3D position of the end effector of 6R manipulator are displayed in Figs. 23–25. Also three orientation angles with respect to the reference axis for both simulation and experimental tests are shown in Figs. 26–28. As these figures mention, the simulation

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

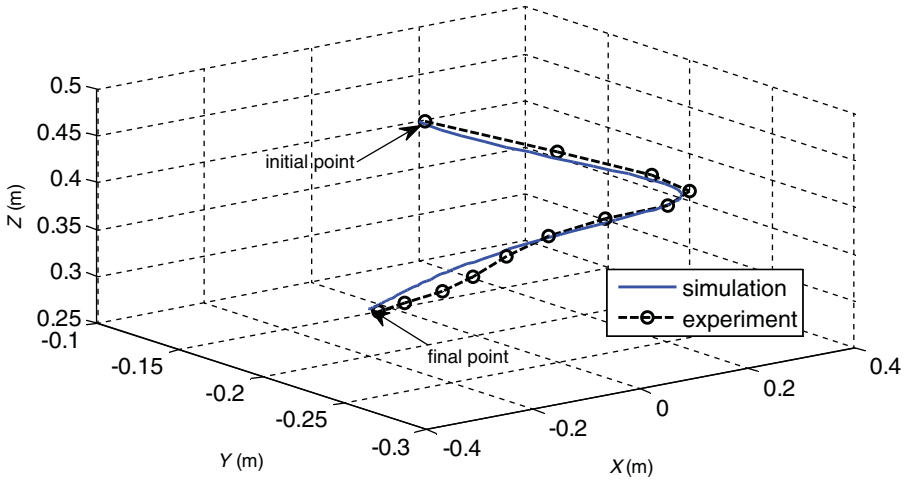


Fig. 22. End effector position in experimental test.

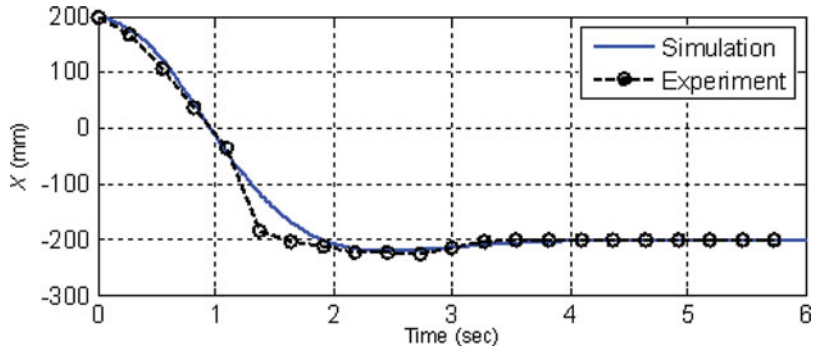


Fig. 23. End effector position in *X* direction.

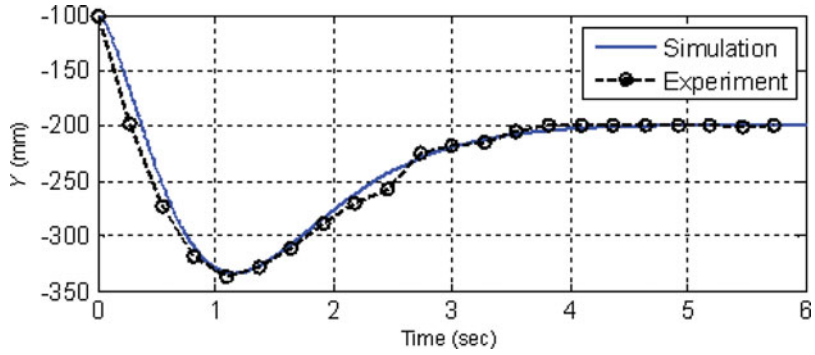


Fig. 24. End effector position in *Y* direction.

and experimental results are close to each other and thus the designed optimal controller is applicable for point to point tasks.

9. Conclusions

In this article, three controller design methods for FJMs with nonlinear dynamic equations are presented. Three methods are compared to evaluate the performance of each method in point to point task. First method is the nonlinear optimal control that is based on the solution of the nonlinear HJB equation which is a nonlinear partial differential equation. An iterative procedure combined to the Galerkin method is used to solve the HJB equation numerically.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

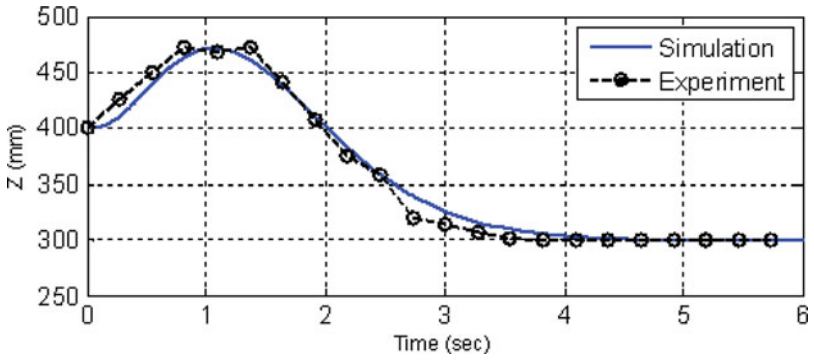


Fig. 25. End effector position in *Z* direction.

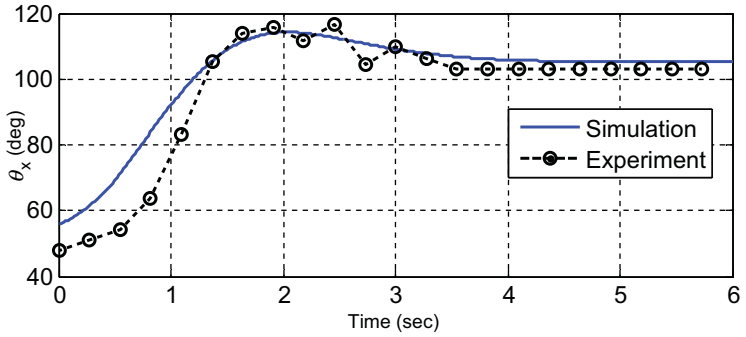


Fig. 26. End effector orientation with respect to *X* axis.

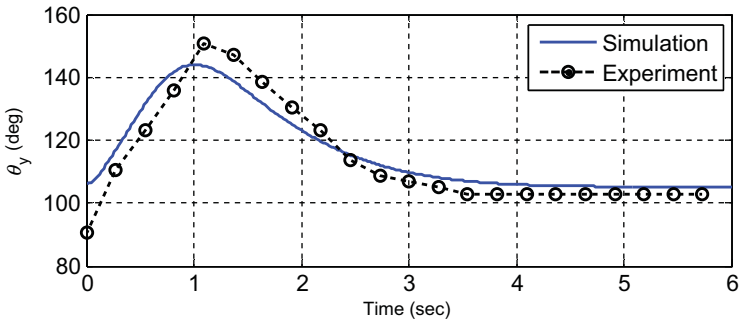


Fig. 27. End effector orientation with respect to *Y* axis.

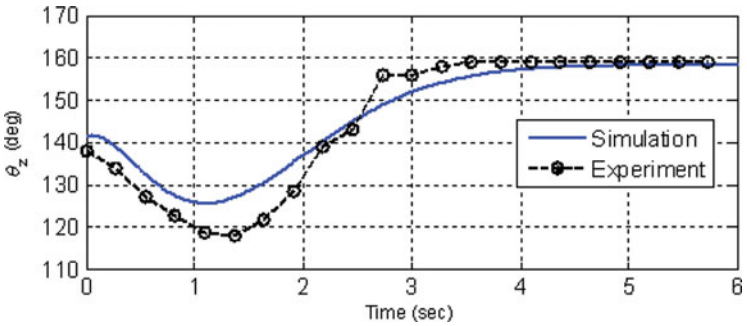


Fig. 28. End effector orientation with respect to *Z* axis.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

Another method is the FL method that linearizes nonlinear dynamic equations using a nonlinear state transformation and a nonlinear feedback law so that the dynamic equations in the new coordinate are in the standard linear form (canonical controllable form). Angular positions, velocities, accelerations, and jerks are considered as new coordinates of the manipulator system. The other approach is the SMC method that is a robust procedure. In order to use this method, the dynamic equations of the FJM are manipulated to change into the standard form and then the standard Slotine method is used to design the SMC law.

The DLCC of the manipulator is used as a performance index to compare the performances of any controller. The DLCC of a manipulator is an important characteristic and it is measured as the maximum allowable load on the end effector that can be carried while restrictions are applied on the joint torques and also the position error in the final point is limited.

Simulations are performed for the planar FJM and the spatial Puma robot and the end effector trajectory, angular position, and velocity of joints are compared. The results show that the value of the DLCC for the optimal control is more than the other controllers and value of the FL method is less than others because of the weak robustness of this method against disturbances and parameters variations. Also a few oscillations are appeared in the angular position in the case of the optimal controller. The simulation results are validated for the 6R manipulator via an experimental test. Position and orientation of the end effector are measured using stereo vision method.

As a final result, the optimal control method has more DLCC than the SMC and the FL methods, however as a further attempt, other solving methods of the optimal control must be proposed to reduce the complexity and the time consuming process of this approach.

References

1. S. Ozgoli and H. D. Taghirad, "A survey on the control of flexible joint robots," *Asian J. Control* **8**(4), 1–15 (2006).
2. W. Chatlatanagulchai and P. H. Meckl, "Model-independent control of a flexible-joint robot manipulator." *J. Dyn. Syst. Meas. Control* **131**, 19–29, (2009).
3. S. Moberg and S. Hanssen, "On Feedback Linearization for Robust Tracking Control of Flexible Joint Robots," *Technical Report from Automatic Control at Linköpings Universitet, 17th IFAC World Congress*, Seoul, Korea (2008).
4. S. J. Yoo, J. B. Park and Y. H. Choi, "Adaptive output feedback control of flexible-joint robots using neural networks: Dynamic surface design approach," *IEEE Trans. Neural Netw.* **19**(10), 1712–1726 (2008).
5. S. J. Yoo, J. B. Park and Y. H. Choi, "Output feedback dynamic surface control of flexible-joint robots," *Int. J. Control, Autom. Syst.* **6**(2), 223–233 (2008).
6. C. M. Kwan and K. S. Yeung, "Robust adaptive control of revolute flexible-joint manipulators using sliding technique," *Syst. Control Lett.* **20**, 279–288 (1993).
7. H. Chaoui, W. Gueaieb, M. Yagoub and P. Sicard, "Hybrid Neural Fuzzy Sliding Mode Control of Flexible-Joint Manipulators with Unknown Dynamics," *IEEE Industrial Electronics IECON*, Paris, 4082–4087 (2006).
8. M. M. Fateh, "Nonlinear control of electrical flexible-joint robots," *Nonlinear Dyn.* **67**(4), 2549–2559 DOI: 10.1007/s11071- 011-0167-3 (2011).
9. M. H. Korayem, M. Irani and S. Rafee Nekoo, "Load maximization of flexible joint mechanical manipulator using nonlinear optimal controller," *Acta Astronaut.* **69**, 458–469 (2011).
10. S. C. Beeler, H. T. Tran and H. T. Banks, "Feedback control methodologies for nonlinear systems," *J. Optim. Theory Appl.* **107**(1), 1–33 (2000).
11. T. Cimen, "State-Dependent Riccati Equation (SDRE) Control: A Survey," *Proceedings of the 17th World Congress the International Federation of Automatic Control*, COEX, South Korea (2008) pp. 3762–3775.
12. M. H. Korayem M. Irani and S. Rafee Nekoo, "Analysis of manipulators using SDRE: A closed loop nonlinear optimal control approach," *Sci. Iranica Trans. B* **17**(6), 456–467 (2010).
13. G. Tang, Y. Zhao and B. Zhang, "Optimal output tracking control for nonlinear systems via successive approximation approach," *Nonlinear Anal.* **66**, 1365–1377 (2007).
14. D. Kirk, *Optimal Control Theory* (Prentice-Hall Inc., Englewood Cliffs, NJ, 2007).
15. M. H. Korayem and M. Irani, "Maximum mynamic load determination of mobile manipulators via nonlinear optimal feedback," *Sci. Iranica Trans. B* **17**(2), 121–135 (2010).
16. C. Y. E. Wang, W. K. Timoszyk and J. E. Bobrow, "Payload maximization for open chained manipulators: Finding weightlifting motions for a puma 762 Robot," *IEEE Trans. Robot. Autom.* **17**(2), 218–224 (2001).
17. M. H. Korayem and A. Nikoobin, "Maximum payload for flexible joint manipulators in point-to-point task using optimal control approach," *Int. J. Adv. Manuf. Technol.* **38**, 1045–1060 (2008).
18. M. H. Korayem, S. Firouzy and A. Heidari, "Dynamic Load Carrying Capacity of Mobile-Base Flexible-Link Manipulators: Feedback Linearization Control Approach," *IEEE International Conference on Robotics and Biomimetics, Sanya* (2007) pp. 2172–2177.

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press

19. M. H. Korayem, R. Haghighi, A. Nikoobin, A. Alamdari and A. H. Korayem, "Determining maximum load carrying capacity of flexible link manipulators," *Sci. Iranica Trans. B: Mech. Eng.* **16**(5), 440–450 (2009).
20. M. H. Korayem and A. Pilechian, "Modification of Algorithms for Determination of Dynamic Load Carrying Capacity in Flexible Joint Robots," *International Conference on Robotics and Biomimetics, Kunming, China* (2006) pp. 924–929.
21. W. H. Chen, D. J. Ballance and P. J. Gawthrop, "Optimal control of nonlinear systems: A predictive control approach," *Automatica* **39**(4), 633–641 (2003).
22. R. W. Beard and T. W. Mclain, "Successive galerkin approximation algorithms for nonlinear optimal and robust control," *Int. J. Control* **71**(5), 717–743 (1998).
23. M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*, Wiley India Pvt, (1989).

https://doi.org/10.1017/S0263574715000028 Published online by Cambridge University Press