

M.H. Korayem · F. Davarpanah · H. Ghariblu

Load carrying capacity of flexible joint manipulators with feedback linearization

Received: 1 September 2004 / Accepted: 15 December 2004 / Published online: 24 March 2006
© Springer-Verlag London Limited 2006

Abstract A computational technique for obtaining the maximum load carrying capacity of robotic manipulators with joint elasticity, subject to accuracy and actuators constraints, is described herein. A feedback linearization technique is used to minimize end-effector deflection. An inversion algorithm is employed for the synthesis of a dynamic feedback control law that provides input-output decoupling and full state linearization. The linearizing input transformations and the corresponding state diffeomorphisms are presented. The proposed technique is then applied to a flexible joint robot. Linearizing control law is expressed in terms of different sets of model variables and their derivatives. As a result, different tracking errors and torques are introduced in the robot-given trajectory and different load carrying capacities are obtained.

Keywords Dynamic load · Elastic · Feedback linearization · Flexible joint · Robot

1 Introduction

The main source of vibration in industrial robot manipulators is the presence of joint elasticity between the driving actuators and the drive links. The origin of elasticity is transmission parts, such as harmonic drives, belts, or long shafts, during high-speed motion or hard contact with the environment [1, 2]. The main

intent of robot controllers is accurate and stable tracking along a desired trajectory, regardless of structural flexibility. As such, controllers should be designed using a more complete dynamic model of the robot [3]. The modelling of robots with elastic joints dates back to the early 1980s [4]. More recently, a detailed analysis of the model structure was given in [5], where it was used for improving asymptotic stability of a simple regulation controller. A reduced dynamic model was also introduced in [6].

For many industrial applications, current robotic manipulators with joint elasticity are relatively slow even when they are not fully loaded. Their speed, load carrying capacity and, hence, their productivity are limited by the deflection of the end-effector and the capability of their actuators. Increasing actuator size and power is largely self-defeating, because of increased cost and power consumption of the larger actuators as well as increased inertia of the actuators themselves. A more successful approach should maximize the load carrying capacity of the flexible manipulator, subject to the constraints imposed by actuator capacity and allowable end-effector deviation for a given dynamic trajectory. Thomas et al. [7] used the load capacity as a criterion for sizing the actuators at the design stage. In their work, piecewise rigid links and joints were assumed. If one removes the rigid body assumption, the Dynamic Load Carrying Capacity (DLCC) determined using the actuator constraint alone [8] would normally be too large. The DLCC for a two-link planar flexible arm is dealt with for only a single dynamic trajectory. In [9], a new method for determining the DLCC for flexible joint manipulators, subject to both actuator and end-effector deflection constraints, is introduced.

During the past decade, nonlinear systems and control theory have generated tremendous development. As one of the most active research areas, feedback linearization is a powerful tool for controlling nonlinear systems and has been applied to many engineering systems. The application to rigid and flexible joint robots is discussed in [10, 11]. Feedback linearization involves transforming a nonlinear system into a controllable linear system using state feedback and coordinate transformations. This problem has been studied with more general feedback transformations. Static state feedback linearization was solved in [12] for

M.H. Korayem
Robotic Research Laboratory, College of Mechanical Engineering,
Iran University of Science and Technology,
Tehran, Iran

F. Davarpanah
College of Mechanical Engineering, Sciences and Researches Branch,
Azad University,
Tehran, Iran

H. Ghariblu
Mechanical Engineering Department,
Zanjan University,
Zanjan, Iran
E-mail: Hkorayem@iust.ac.ir

single input systems, and in [13] for multi-input systems. For high order systems, the transformation to a state-space format, though straightforward, may obscure some relevant model structural properties and lead to complicated expressions [14].

In this paper, a new method of feedback linearization is proposed for flexible joint manipulators. The proposed method is used to determine the dynamic load carrying capacity of these manipulators. In the three case studies, linearizing control law are applied by considering different links or rotor angles as state feedback variables. In each case, different tracking errors and control signals (torque) result. Consequently, different load carrying capacities are obtained. It is shown that using the feedback linearization technique, the end-effector deflection is reduced and DLCC for a given trajectory is improved. Finally, simulation tests are carried out in order to verify the algorithm.

2 Feedback linearization controller

Each multi-input nonlinear system can be written as follows: [15]

$$\dot{x} = f(x) + \sum_{i=1}^m u_i g_i(x) = f(x) + g(x)u \quad (1)$$

where $f(x)$ and $g(x)$ are analytic functions on R^n , $f(0) = 0$, and $u \in R$. This system is feedback linearizable if there exists a region U in R^n containing the origin, a diffeomorphism $T : U \rightarrow R^n$. Nonlinear feedback is as follows:

$$u = \alpha(x) + \beta(x)v \quad (2)$$

with $\beta(x) \neq 0$ on U . Such that the transformed variables:

$$Y = T(x) \quad (3)$$

By satisfying the equation:

$$\dot{y}(x) = Ay(x) + bv \quad (4)$$

where:

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 0 \\ \dots \\ \dots \\ 1 \end{bmatrix}$$

In fact, the feedback linearization problem determines u in such a way that a system can be linearized. To calculate u , each diffeomorphism transformation is derived until u appears. The last derivation must then be set equal to v , and u is calculated. The value of v can be obtained as follows: [15]

$$v = y_d^{(r)} - k_{r-1}(y^{(r-1)} - y_d^{(r-1)}) - k_{r-2}(y^{(r-2)} - y_d^{(r-2)}) - \dots - k_0(y - y_d) \quad (5)$$

By applying this control law to the r^{th} order linear system, tracking error $e = y - y_d$ satisfies the r^{th} order linear equation. For tracking a desired path we must set $v = y_d$. In this way, the error equation will be obtained as below:

$$e^{(r)} + k_{r-1}e^{(r-1)} + k_{r-2}e^{(r-2)} + \dots + k_0e = 0 \quad (6)$$

Hence, the dynamic errors are determined by selecting ks . By choosing a suitable k value, the tracking error can be minimized.

3 Feedback linearization algorithm

State variables are set as link angles, rotor angles and their derivatives, and identified as x_1, x_2 , and \dots, x_m . The system is then in the form of Eq. 1 and is linearizable. With sufficient derivation from both parts of Eq. 3 we can obtain:

$$\begin{aligned} \dot{y}_i &= \dot{T}_i = \frac{\partial T_i}{\partial x_1} \dot{x}_1 + \dots + \frac{\partial T_i}{\partial x_n} \dot{x}_n = C_1(x_1, x_2, \dots, x_m) \\ \ddot{y}_i &= \ddot{T}_i = C_2(x_1, x_2, \dots, x_m) \\ y_i^{(3)} &= T_i^{(3)} = C_3(x_1, x_2, \dots, x_m) \\ &\vdots \\ y_i^{(n)} &= T_i^{(n)} = C_n(x_1, x_2, \dots, x_m) + D_n(u_1, \dots, u_i) = v_i \end{aligned} \quad (7)$$

where y_i s are the parameters of Y , and T_i s are the parameters of T . The control signal u_i can be obtained from the last derivation of Eq. 7 as follows:

$$u_i = \alpha_i(x_1, x_2, \dots, x_m) + \beta_i(x_1, x_2, \dots, x_m, v_1, \dots, v_i) \quad (8)$$

The feedback linearization parameters will be obtained and the system can then be linearized, as shown in Fig. 1.

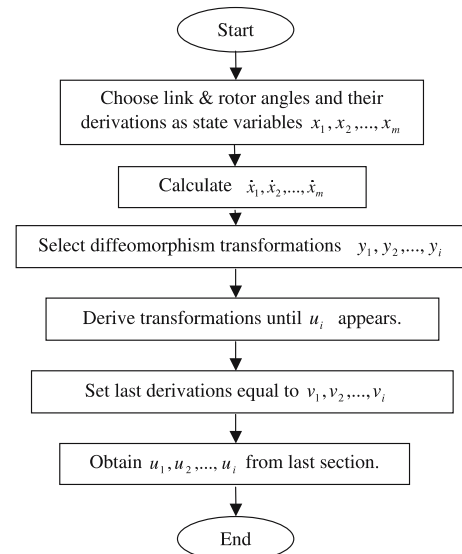


Fig. 1. Feedback linearization algorithm

4 DLCC for a desired trajectory

For a prescribed trajectory, the DLCC of a flexible joint manipulator is defined as the maximum load that the manipulator can carry in executing the trajectory with an acceptable tracking accuracy. The main constraints, which bound the DLCC of manipulators, are actuators and accuracy constraints.

4.1 Actuator constraint

The upper and lower bounds of the allowable torque are as follows:

$$\begin{aligned} u_a^+ &= k_1 - k_2 \dot{q} \\ u_a^- &= -k_1 - k_2 \dot{q} \end{aligned} \quad (9)$$

where $k_1 = \tau_s$, $k_2 = \tau_s/w_0$, τ_s is the stall torque, and w_0 is maximum speed of the motor without any load. A load coefficient, complying with the torque constraint, can be gained as follows:

$$(c_a)_j = \min \left\{ \frac{(\tau_a)_i}{\max\{\tau_e\} - \max\{\tau_n\}}, i = 1, \dots, n \right\} \quad (10)$$

where τ_n is the no-load torque, τ_e is the end-effector torque, and $(\tau_a)_i$ is the maximum allowable torque at joint i .

4.2 Accuracy constraint

The DLCC varies from place to place on a given trajectory. A load coefficient $(c_p)_j$ for point j , $j = 1, 2, \dots$, and m is as follows:

$$(c_p)_j = \frac{R_p - (\Delta_e)_j}{\max\{\Delta_e\} - \max\{\Delta_n\}} \quad (11)$$

where $(\Delta_n)_j$ is the no-load deflection, $(\Delta_e)_j$ is the deflection with added end-effector mass, and R_p indicates how much load can be carried without violating the deflection constraint through point j . A load coefficient c can be found as follows:

$$c = \min \{(c_p)_j, (c_a)_j\}, j = 1, \dots, m \quad (12)$$

The maximum mass m_{load} for this trajectory is then:

$$m_{\text{load}} = cm_e \quad (13)$$

where m_e is the mass of end-effector.

4.3 Algorithm of determination DLCC with feedback linearization

The algorithm shown in Fig. 2 has two parts. In the first part, the effects of the combined manipulator and load motions on actuator torques τ are computed. In the next part, assuming the motion of the robot is limited to the desired trajectory, the corresponding no-load torque τ_{nl} is computed. It is clear that with an increasing load mass, the torque requirements of motors will increase. By subtracting τ_{nl} from τ we can obtain the torque needed to carry the load τ_l as, $\tau_l = \tau - \tau_{nl}$.

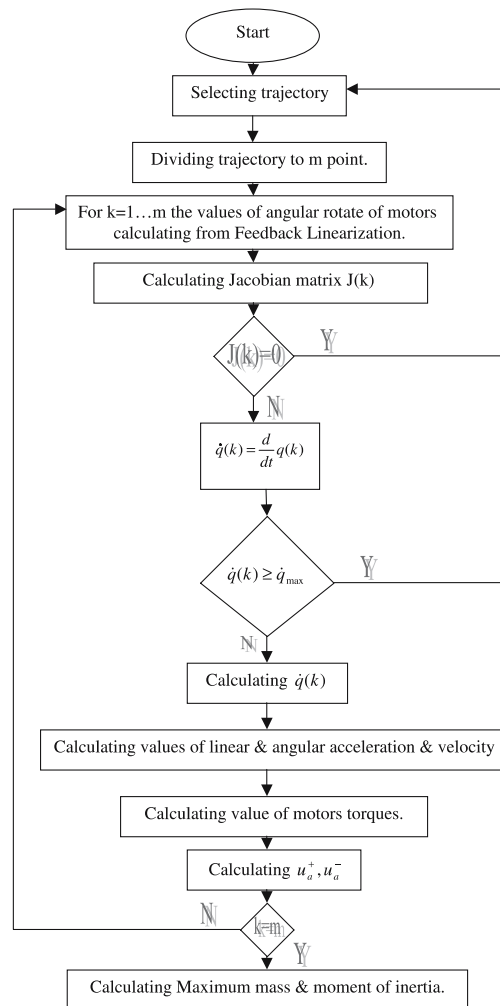


Fig. 2. Algorithm for determining maximum load carrying capacity

To determine the DLCC value, the desired path is first discretized to certain points. For these m points, robot kinematical variables are computed. To prevent a singularity condition, the determinant of the $J_a(k)$ matrix must be calculated. If the value of the determinant is zero then another path must be selected. The torque of each motor in both a loaded and unloaded robot will be calculated and the torque bounds can be then measured.

5 Controller design

Equations for a two-link flexible joint manipulator (shown in Fig. 3) are provided below:

a. Kinematic equations.

$$\begin{aligned} (-l_1 \sin(q_1) - l_2 \sin(q_1 + q_3))\ddot{q}_1 - l_2 \sin(q_1 + q_3)\ddot{q}_2 &= R_{r1} \\ (-l_1 \cos(q_1) + l_2 \cos(q_1 + q_3))\ddot{q}_1 + l_2 \cos(q_1 + q_3)\ddot{q}_2 &= R_{r2} \end{aligned} \quad (14)$$

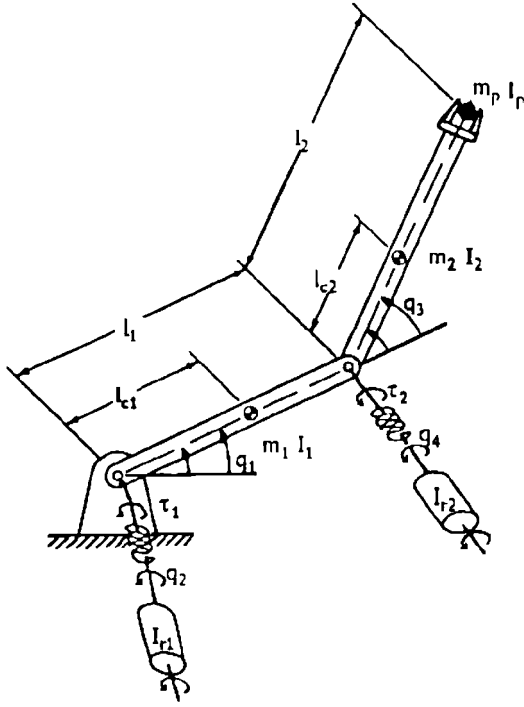


Fig. 3. Two-Link flexible joint manipulator model

b. Dynamic equations based on Lagrangian approach.

$$\begin{aligned}
 D_{11}\ddot{q}_1 + D_{12}\ddot{q}_3 - C\dot{q}_3^2 - 2C\dot{q}_1\dot{q}_3 + K(q_1 - q_2) &= 0 \\
 D_{22}\ddot{q}_3 + D_{12}\ddot{q}_1 + C\dot{q}_1^2 + K(q_3 - q_4) &= 0 \\
 I_{r1}\ddot{q}_2 + K(q_2 - q_1) &= \tau_1 \\
 I_{r2}\ddot{q}_4 + K(q_4 - q_3) &= \tau_2
 \end{aligned} \tag{15}$$

where

$$\begin{aligned}
 D_{11} &= m_1 l_{c1}^2 + I_1 + m_2 [l_1^2 + l_{c2}^2 + 2l_1 l_{c2} \cos(q_3)] + I_2 \\
 &\quad + m_p [l_1^2 + l_2^2 + 2l_1 l_2 \cos(q_3)] + I_p \\
 D_{22} &= m_2 l_{c2}^2 + I_2 + m_p l_2^2 + I_p \\
 D_{12} &= m_2 l_1 l_{c2} \cos(q_3) + m_2 l_{c2}^2 + I_2 + m_p l_1 l_2 \cos(q_3) + m_p l_2^2 + I_p \\
 C &= m_2 l_1 l_{c2} \sin(q_3) + m_p l_1 l_2 \sin(q_3)
 \end{aligned}$$

In state space we set

$$\begin{aligned}
 X_1 &= q_1 & X_2 &= \dot{q}_1 \\
 X_3 &= q_3 & X_4 &= \dot{q}_3 \\
 X_5 &= q_2 & X_6 &= \dot{q}_2 \\
 X_7 &= q_4 & X_8 &= \dot{q}_4
 \end{aligned} \tag{16}$$

where q_1 and q_3 are link angles, and q_2 and q_4 are rotor angles. At this time, the two-link flexible joint manipulator system

Eq. 15 can be re-written as shown below:

$$\begin{aligned}
 \dot{X}_1 &= X_2 \\
 \dot{X}_2 &= \frac{D_{22}(CX_4(2X_2 + X_4) + K(X_5 - X_1))}{D_{11}D_{22} - D_{12}^2} \\
 &\quad + \frac{D_{12}(CX_2^2 + K(X_3 - X_7))}{D_{11}D_{22} - D_{12}^2} \\
 \dot{X}_3 &= X_4 \\
 \dot{X}_4 &= \frac{D_{12}(CX_4(2X_2 + X_4) + K(X_5 - X_1))}{-D_{11}D_{22} + D_{12}^2} \\
 &\quad + \frac{D_{11}(CX_2^2 + K(X_3 - X_7))}{-D_{11}D_{22} + D_{12}^2} \\
 \dot{X}_5 &= X_6 \\
 \dot{X}_6 &= \frac{1}{I}(u_1 + K(X_1 - X_5)) \\
 \dot{X}_7 &= X_8 \\
 \dot{X}_8 &= \frac{1}{I}(u_2 + K(X_3 - X_7))
 \end{aligned} \tag{17}$$

The system Eq. 15 then leads to:

$$\dot{x} = f(x) + g_1(x)u_1 + g_2(x)u_2, \quad x \in \mathbb{R}^n$$

where

$$f(x) = \begin{bmatrix} X_2 \\ \frac{D_{22}(CX_4(2X_2+X_4)+K(X_5-X_1))+D_{12}(CX_2^2+K(X_3-X_7))}{D_{11}D_{22}-D_{12}^2} \\ X_4 \\ \frac{D_{12}(CX_4(2X_2+X_4)+K(X_5-X_1))+D_{11}(CX_2^2+K(X_3-X_7))}{-D_{11}D_{22}+D_{12}^2} \\ X_6 \\ \frac{K}{I}(X_1-X_5) \\ X_8 \\ \frac{K}{I}(X_3-X_7) \end{bmatrix},$$

$$g_1(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{I} \\ 0 \\ 0 \end{bmatrix}, \quad g_2(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{I} \end{bmatrix} \tag{18}$$

There are different choices for solving the problem of feedback linearization, because there is a range of functions that can be chosen for diffeomorphism transformations. Two methods are introduced: in the first one, the link angles are selected for diffeomorphism transformation; and in the second one, the rotor angles are selected.

Method 1. In the first method, the diffeomorphism transformations for a two-link robot arm are $T_1 = X_1 = q_1$, $T_2 = X_3 = q_3$. Feedback linearization parameters u_1 and u_2 will be obtained with four-time deriving of desired values (q_1, q_3). Thus, we

have:

$$\begin{aligned} u_1 &= tm_{30}(-v_1 + tm_{23}) + tm_{31}(-v_2 + tm_{25}) \\ u_2 &= tm_{32}(-v_1 + tm_{23}) + tm_{33}(-v_2 + tm_{25}) \end{aligned} \quad (19)$$

where the relative terms of these equations are given in the Appendix A.

Method 2. In the second method, $T_1 = X_5 = q_2$ and $T_2 = X_7 = q_4$ are the diffeomorphism transformations for a two-link robot arm. This means that the desired values are (q_2, q_4) and the feedback linearization parameters are u_1 and u_2 . The results from two-times deriving of these transformations are provided below:

$$\begin{aligned} u_1 &= I.v_1 - K(X_1 - X_5) \\ u_2 &= I.v_2 - K(X_3 - X_7) \end{aligned} \quad (20)$$

6 Simulation study

Determining the tracking error is possible once the feedback linearization parameters have been calculated. Parameters v_1, v_2 from Eq. 20 are obtained as follow:

$$\begin{aligned} v_1 &= \ddot{q}_{d2} - k_{11}(\dot{q}_2 - \dot{q}_{d2}) - k_{12}(q_2 - q_{d2}) \\ v_2 &= \ddot{q}_{d4} - k_{21}(\dot{q}_4 - \dot{q}_{d4}) - k_{22}(q_4 - q_{d4}) \end{aligned} \quad (21)$$

The error diminishes to zero when $v_1 = \dot{q}_2$ and $v_2 = \dot{q}_4$. Hence, the error equations leads to:

$$\begin{aligned} \ddot{e}_1 + k_{11}\dot{e}_1 + k_{12}e_1 &= 0 \\ \ddot{e}_2 + k_{21}\dot{e}_2 + k_{22}e_2 &= 0 \end{aligned} \quad (22)$$

By appropriate selection of k_{22}, k_{21}, k_{12} , and k_{11} , the tracking error can be minimized. A closed loop system controller was designed to have duplicated poles in -10, which means that $k_{22} = k_{21} = k_{12} = k_{11} = 20$. To simplify the simulation, we set $I_p = 0$.

Fig. 4. Controller 1 – tracking link angles (q_1, q_3)

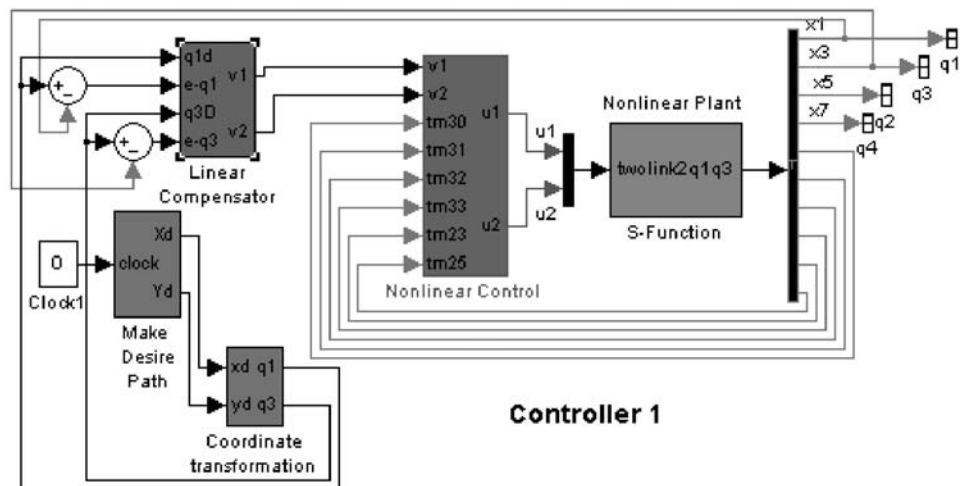


Table 1. Numerical values for simulation

Parameter	Value	Unit
Spring constant	$k_1 = k_2 = 5.5$	N/M
Area moment of inertia	$I_{r1} = I_{r2} = 9.9 \times 10^{-12}$	m^4
Link length	$L_1 = L_2 = 1, l_{c1} = l_{c2} = 0.5$	m
Link mass	$m_1 = m_2 = 1$	Kg
Actuator constant	$k_1 = 18.9, k_2 = 5.4$	N-m & N-m/rad respectively
Mass of end-effector	$m_e = 0.2$	Kg
Initial condition	$q_1 = q_2 = 0$	Degree
Link linear mass density	$\mu_1 = \mu_2 = 0.405$	Kg/m
Stall torque	$\tau_s = 0.63$	N-m
No load speed	$w_0 = 3.5$	Rad/s

For this simulation, the desired trajectory as a function of time is $X_d = 1.05 - 0.02t^2$ and $Y_d = 1.05 + 0.01t^2$. Other robot parameters used in these simulations are listed in the Table 1.

Figure 4 presents the first controller that tracks link angles. As shown, the errors are derived by subtracting coordinate transformations of the desired path from output of the system. From equation Eq. 21, v_1 and v_2 are calculated using a linear compensator. Nonlinear control can then obtain the u_1 and u_2 parameters and send these to the robot system (nonlinear plant). Figure 5 shows the desired and actual path and the accuracy bounds of controller 1. To determine the upper and lower bounds of the end-effector accuracy, the R_p value should be 0.2 mm. Figure 6 shows the torque of both links in full load and no-load status in the first controller.

For the second controller, a similar approach was used. In Fig. 7, the second controller tracks rotor angles such that coordinate transformations can be done on the desired path. The rotor angles are then calculated from Eq. 15. Figure 8 shows the desired and actual path of controller 2, as well as the accuracy bounds. Figure 9 demonstrates the torque of both links in full

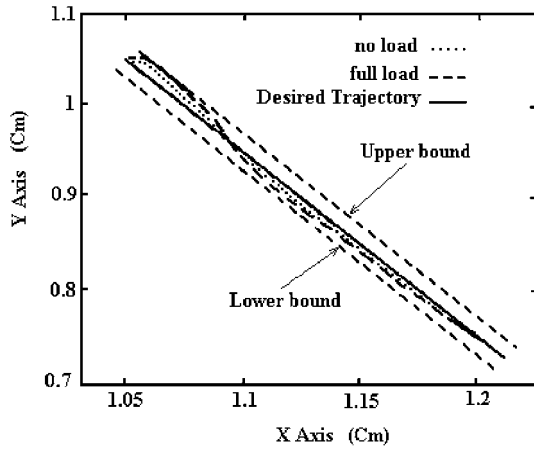


Fig. 5. Desired and actual trajectory of controller 1

load and no load status in the second controller. The maximum amount of torque used is also shown.

Employing controller 1, the DLCC of the manipulator is computed as $m_{load} = 1.73$ kg. It was found to be the maximum load that the given actuators can carry while executing the trajec-

tory in the first controller. Applying heavier loads may result in increased tracking error, beyond the acceptable condition.

Similarly for controller 2, $m_{load} = 1.7$ kg was obtained to determine the maximum load that can be ported by the actuators in the second controller.

In the last method of simulation, we use (q_1, q_3) separately for tracking the trajectory. In this method, the robot system was decoupled into two parts so that each link worked separately and corrected its error independently. This can reduce state space variables to four, and hence speed should increase. This type of controller is shown in Fig. 10. As shown, the errors are obtained by subtracting the coordinate transformation of the desired path from the output of both decoupled systems. Errors then go to the linear compensator to generate v . The nonlinear control unit creates u from that, and each part of the decoupled system sends u to the robot system. In Fig. 11, the bound accuracy of the end-effector is assessed using R_p , which should be 0.2 mm. In comparison with the last controllers, it's clear that the deflection is increased. Figure 12 shows the torque of both links in full load, which uses maximum allowable torque and no-load status. The maximum dynamic load carrying capacity of the robot is obtained from $m_{load} = 1.57$ kg in controller 3.

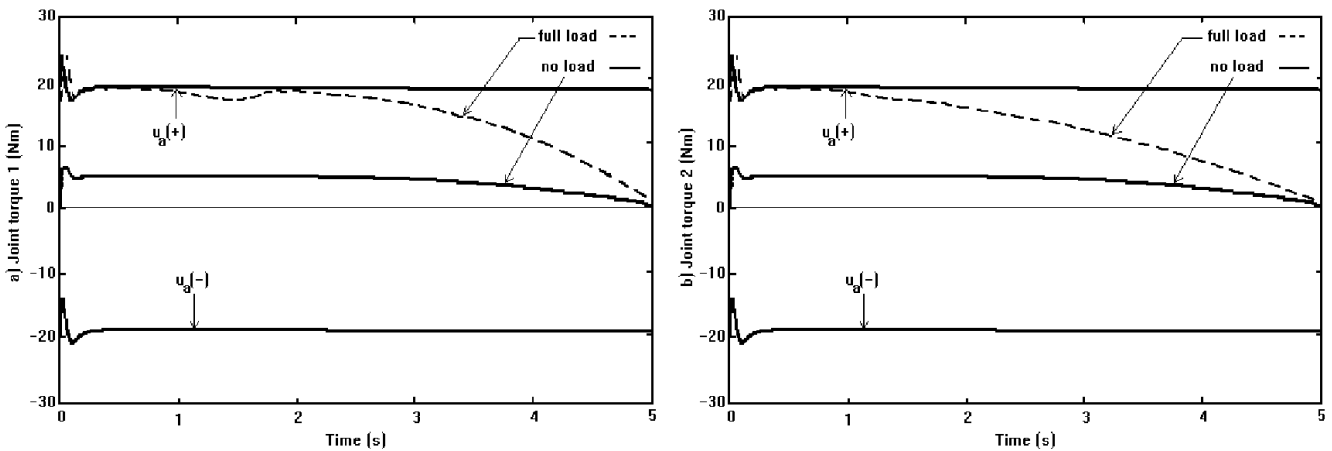
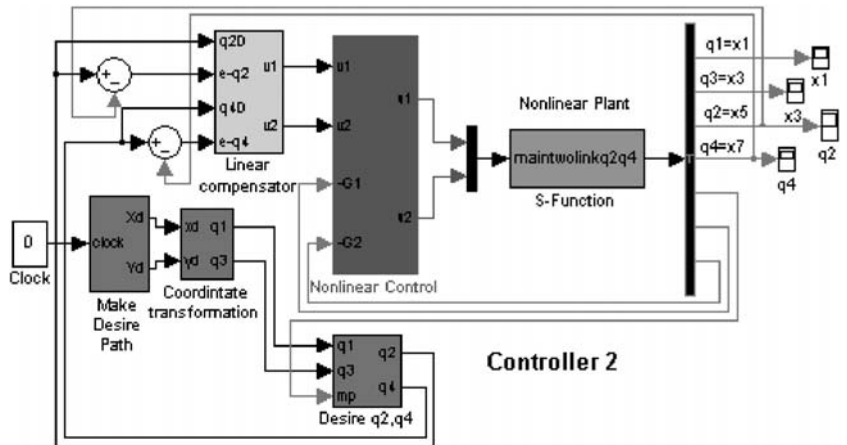


Fig. 6. Necessary torques and acceptable bound for trajectory tracking of controller 1

Fig. 7. Controller 2 – tracking rotor angles (q_2, q_4)



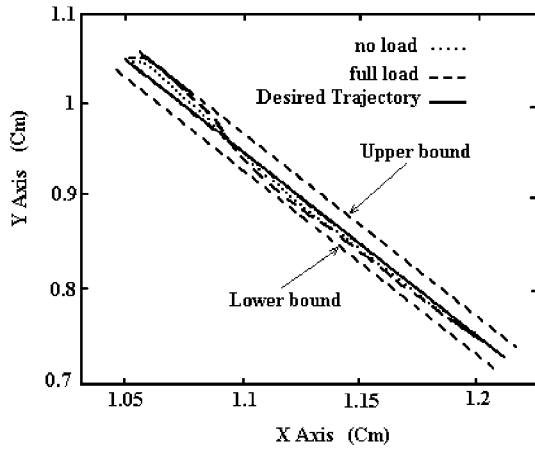


Fig. 8. Desired and actual trajectory of controller 2

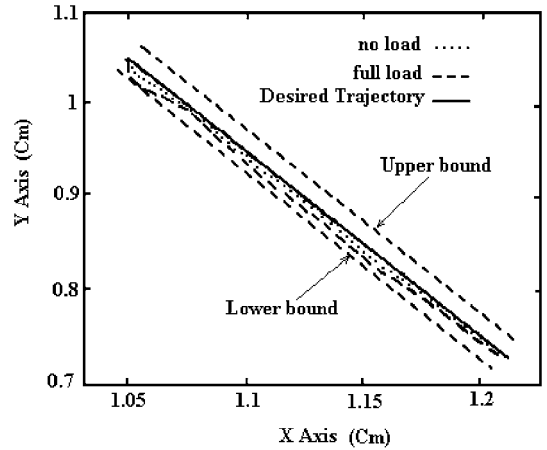


Fig. 11. Desired and actual trajectory of controller 3

6.1 Two-link flexible joint manipulator simulation without F.L.

The deflection of joints are greater in the open loop systems than in closed loop systems, which leads the manipulator into a smaller DLCC. In this method, the DLCC is found by setting

$m_{load} = 0.67$ kg. The path of the robot in this method is shown in Fig. 13.

It is clear that the DLCC for the robot was increased by using feedback linearization, due to the reduced deflection in the robot.

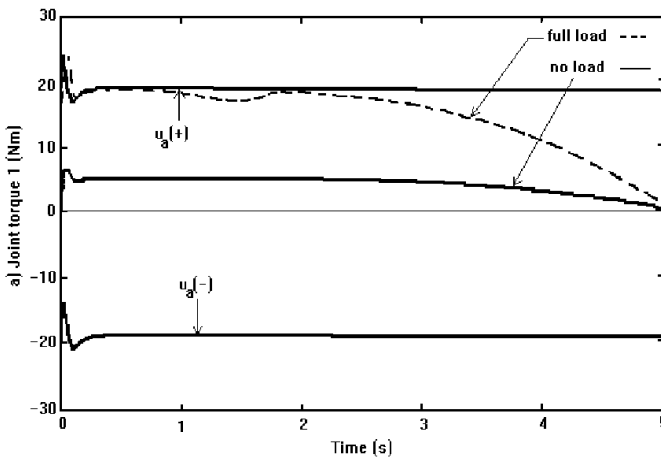


Fig. 9. Joint torques of controller 2

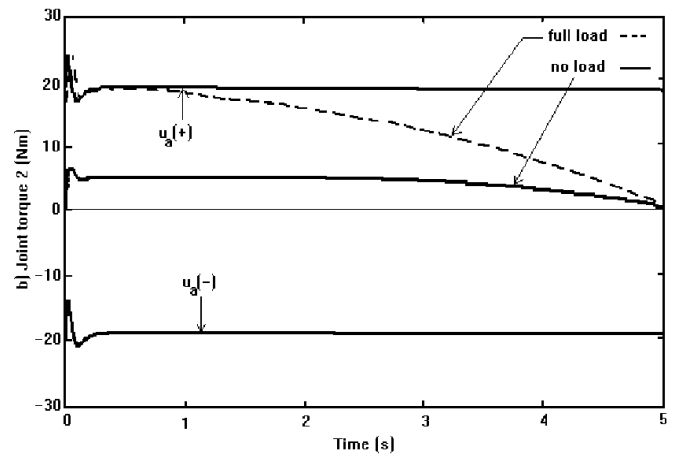
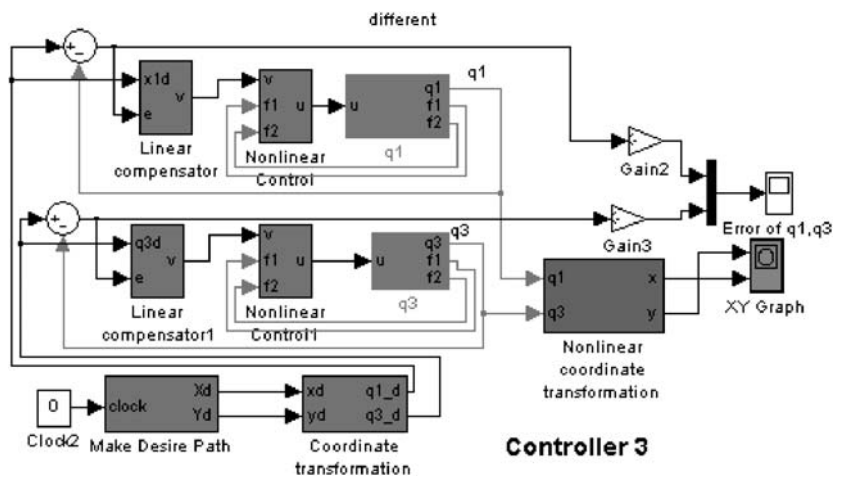


Fig. 10. Controller 3 – (tracking q1, q3 separately)



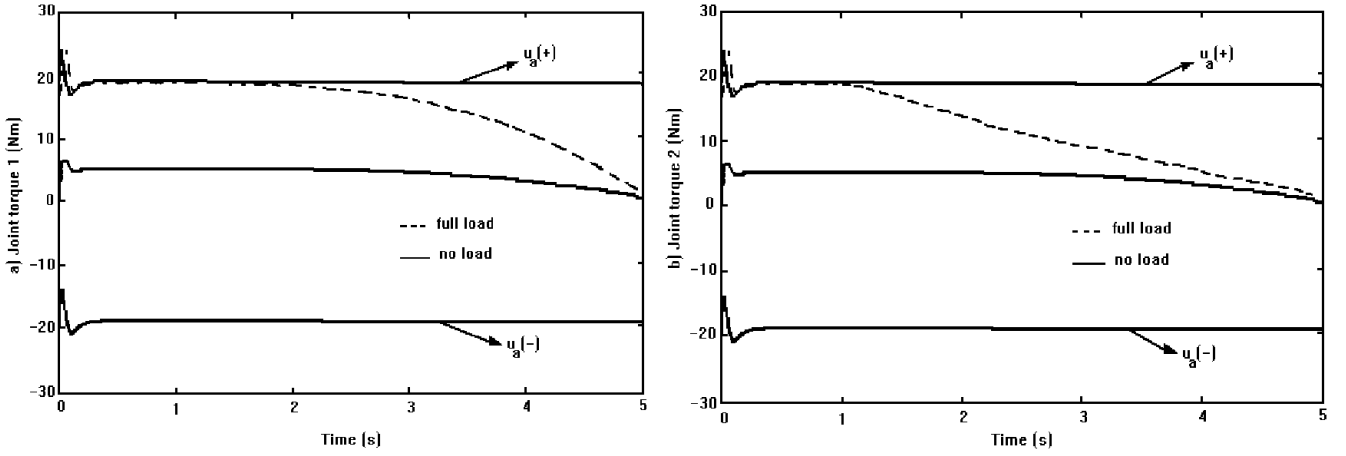


Fig. 12. Joint torques of controller 3

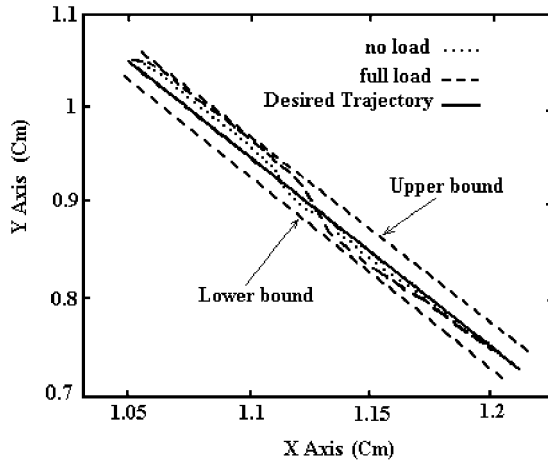


Fig. 13. The desired and actual trajectory without feedback linearization

Using the feedback linearization method, the tracking error is minimized at each point along the given trajectory compared to the open loop method. As a result, the dynamic load carrying capacity of manipulator is increased.

Controller 2 is faster than controller 1, because of a shorter feedback linearization formulation (u). Also, controller 3 is faster than both of the previous controllers because the system is decoupled and the state space variables are reduced to four. However, the feedforward method is still faster as it needs less calculations. Controller 3 has a smaller DLCC than the other two controllers because it has more deflection. The first two controllers have roughly the same torque and deflections and, thus, have the same DLCC. These two controllers have very similar designs, the only difference being their diffeomorphism transformation.

7 Conclusions

In this paper, a new method of feedback linearization for robots with elastic joints is introduced. In the simulation studies, three methods for two-link manipulators with feedback linearization

are used. The methods used different tracking parameters, which produced different tracking errors and control signals (torque of joints). Hence, the maximum load carrying capacity in each case was different. In each method, the DLCC of the manipulators was determined with feedback linearization. In comparison with feedforward method, the use of feedback linearization can increase maximum dynamic load carrying capacity of robots because of the reduction in deflection.

Appendix A: Terms of u in Eq. 19

$$tm_{33} = -\frac{tm_{26}}{tm_3tm_{22}}$$

$$tm_{32} = -\frac{tm_{24}}{tm_3tm_{22}}$$

$$tm_{31} = -tm_{29}tm_{26}$$

$$tm_{30} = tm_{28} - tm_{29}tm_{24}$$

$$tm_{29} = \frac{tm_1}{tm_3tm_{27}}$$

$$tm_{28} = -1.8^{-12} \frac{1}{0.25 + mp} tm_3$$

$$tm_{27} = (0.25 + mp)tm_{22}$$

$$tm_{26} = 5.5^{11} (0.25 + mp)$$

$$tm_{25} = \frac{ttm_6tm_9 - tm_{14}}{tm_3^2 + \frac{tm_{21}}{tm_3}}$$

$$tm_{24} = 5.5 \wedge 11 tm_1$$

$$tm_{23} = \frac{ttm_5tm_9 - tm_{13}}{tm_3^2 + \frac{tm_{20}}{tm_3}}$$

$$tm_{22} = \frac{3.9^{23} (-tm_1^2 + (0.25 + mp)tm_2)}{tm_3^2}$$

$$tm_{21} = -(0.5 + mp)tm_4tm_{11} + (1 + mp)tm_5tm_{11}$$

$$+ 2(0.5 + mp) \sin(X_3)X_4tm_8$$

$$- 2(1 + mp) \sin(X_3)X_4tm_{10} + tm_2tm_{15} - tm_1tm_{19}$$

$$\begin{aligned}
tm_{20} &= -(0.5 + m_p)tm_5tm_{11} + 2(0.5 + m_p) \sin(X_3)X_4tm_{10} \\
&\quad - tm_1tm_{15} + (0.25 + m_p)tm_{19}) \\
tm_{19} &= 5.5^{12}(X_1 - X_5) \\
&\quad + \frac{-5.5ttm_5 + 4(0.5 + m_p) \cos(X_3)X_4^2ttm_6}{tm_3} \\
&\quad + 2(0.5 + m_p)X_2X_4tm_{12} \\
&\quad + (0.5 + m_p)X_4^2tm_{12} + 4(0.5 + m_p) \\
&\quad \times \cos(X_3)X_4 \left(\frac{X_4ttm_5}{tm_3} + \frac{X_2ttm_6}{tm_3} \right) \\
&\quad + (0.5 + m_p) \sin(X_3)tm_{18} + 2(0.5 + m_p) \sin(X_3)tm_{17} \\
tm_{18} &= -\frac{2X_4tm_6ttm_6 + 2ttm_6^2}{tm_3^2} + \frac{tm_{16}}{tm_3} \\
tm_{17} &= \frac{2ttm_5ttm_6}{tm_3^2} + X_4 \left(-\frac{tm_6ttm_5}{tm_3^2} \right) \\
&\quad + X_4 \left(\frac{(0.5 + m_p) \sin(X_3)X_4tm_5 + (0.25 + m_p)tm_8 - tm_1tm_{10}}{tm_3} \right) \\
&\quad + X_2 \left(-\frac{tm_6ttm_6}{tm_3^2} + \frac{ttm_{11}}{tm_3} \right) \\
tm_{16} &= 2X_4ttm_{11} \\
tm_{15} &= 5.5^{12}(X_3 - X_7) - \frac{5.5ttm_6}{tm_3} \\
&\quad - (0.5 + m_p) \left(\frac{4 \cos(X_3)X_2X_4ttm_5}{tm_3} + X_2^2tm_{12} \right) \\
&\quad - (0.5 + m_p) \left(\sin(X_3) \left(\frac{2X_2tm_6ttm_5 + 2ttm_5^2}{tm_3^2} \right) \right) \\
&\quad + (0.5 + m_p) \left(\sin(X_3) \left(\frac{2X_2((0.5 + m_p) \sin(X_3)X_4tm_5)}{tm_3} \right) \right) \\
&\quad + (0.5 + m_p) \left(\sin(X_3) \left(\frac{((0.25 + m_p)tm_8 - tm_1tm_{10})}{tm_3} \right) \right) \\
tm_{14} &= 2tm_6ttm_{11} \\
tm_{13} &= 2tm_6((0.5 + m_p) \sin(X_3)X_4tm_5 \\
&\quad + (0.25 + m_p)tm_8 - tm_1tm_{10}) \\
tm_{12} &= -\sin(X_3)X_4^2 + \frac{\cos(X_3)ttm_6}{tm_3} \\
tm_{11} &= -\cos(X_3)X_4^2 - \frac{\sin(X_3)ttm_6}{tm_3} \\
ttm_{11} &= (0.5 + m_p) \sin(X_3)X_4tm_4 - (1 + m_p) \sin(X_3)X_4tm_5 \\
&\quad - tm_1tm_8 + tm_2tm_{10} \\
tm_{10} &= -5.5X_4 - (0.5 + m_p) \cos(X_3)X_2^2X_4 \\
&\quad - \frac{2(0.5 + m_p) \sin(X_3)X_2ttm_5}{tm_3} + 5.5X_8 \\
tm_9 &= \frac{2tm_6^2}{tm_3^3} - ((0.25 + m_p)(1 + m_p) \cos(X_3)X_4^2 \\
&\quad + 2(0.5 + m_p) \cos(X_3)tm_1X_4^2 \\
&\quad - 2(0.5 + m_p)^2 \sin^2(X_3)X_4^2
\end{aligned}$$

$$\begin{aligned}
&\quad - \frac{(0.25 + m_p)(1 + m_p) \sin(X_3)ttm_6}{tm_3} \\
&\quad + \frac{2(0.5 + m_p)tm_1 \sin(X_3)ttm_6}{tm_3} / tm_3^2 \\
tm_8 &= -5.5X_2 + (0.5 + m_p) \cos(X_3) \left(2X_2X_4^2 + X_4^3 \right) + 5.5X_6 \\
&\quad + \frac{tm_7 + 2(0.5 + m_p) \sin(X_3)(-tm_1tm_4 + tm_2tm_5)(X_2 + X_4)}{tm_3} \\
tm_7 &= 2(0.5 + m_p) \sin(X_3)X_4 \left((0.25 + m_p)tm_4 - tm_1tm_5 \right) \\
ttm_6 &= -tm_1tm_4 + tm_2tm_5 \\
tm_6 &= -(0.25 + m_p)(1 + m_p) \sin(X_3)X_4 \\
&\quad + 2(0.5 + m_p)tm_1 \sin(X_3)X_4 \\
ttm_5 &= (0.25 + m_p)tm_4 - tm_1tm_5 \\
tm_5 &= -(0.5 + m_p) \sin(X_3)X_2^2 - 5.5X_3 + 5.5X_7 \\
tm_4 &= (-5.5X_1 + (0.5 + m_p) \sin(X_3) \left(2X_2X_4 + X_4^2 \right) + 5.5X_5 \\
tm_3 &= tm_1^2 + (0.25 + m_p)tm_2 \\
tm_2 &= 1.5 + 1.25m_p + (1 + m_p) \cos(X_3) \\
tm_1 &= 0.25 + m_p + (0.5 + m_p) \cos(X_3)
\end{aligned}$$

References

- Sweet LM, Good MC (1985) Redefinition of the robot motion control problem. *IEEE Control Syst* 5(3):18-24
- Rivin E (1988) *Mechanical design of robots*. McGraw-Hill, New York
- De Luca A, Tomei P (1996) Elastic joints. In: Canudas de Wit C, Siciliano B, Bastin G (eds) *Theory of robot control*. Springer, Berlin Heidelberg New York, pp 179-217
- Cesareo G, Nicol'o F, Nicosia S (1984) DYMR: a code for generating dynamic model of robots. In: *IEEE international conference on robotics and automation*, Atlanta, GA
- Tomei P (1991) A simple PD controller for robots with elastic joints. *IEEE Trans Automat Control* 36(10):1208-1213
- Spong MW (1987) Modeling and control of elastic joint robots. *ASME J Dyn Syst Meas Control* 109:310-319
- Thomas M, Yuan-Chou HC, Tesar D (1985) Optimal actuator sizing for robotic manipulators based on local dynamic criteria. *ASME J Mech Trans Automat Des* 107:163-169
- Wang LT, Ravani B (1988) Dynamic load carrying capacity of mechanical manipulators - part I: problem formulation. *ASME J Dyn Syst Meas Control* 110:46-52
- Korayem MH, Basu A (1994) Dynamic load carrying capacity of robotic manipulators with joint elasticity imposing accuracy constraints. *Robot Autonomous Syst* 13:219-229
- Spong MW, Vidyasagar M (1989) *Robot dynamics and control*. Wiley, New York
- De Luca A (1996) Decoupling and feedback linearization of robots with mixed rigid/elastic points. In: *IEEE international conference on robotics and automation*, Minneapolis, MN
- Brockett RW (1978) Feedback invariants for nonlinear systems. In: *Proceedings of sixth IFAC world congress* pp. 1155-1120
- Jakubczyk B, Respondek W (1980) On linearization of control systems. *Bulletin de l'Academie Polonaise des Sciences, Serie des Sciences Matematyka* 28:517-522
- De Luca A (1998) A general algorithm for dynamic feedback linearization of robots with elastic joints. In: *Proceedings of 37th conference on decision and control*, Leuven, Belgium
- Slotine JJ, Li W (1991) *Applied nonlinear control*. Prentice-Hall, New York