Robotica (1998) volume 16, pp. 97–108. Printed in the United Kingdom © 1998 Cambridge University Press

Quadratic optimal control of a two-flexible-link robot manipulator

A.S. Morris and A. Madani

Department of Automatic Control and Systems Engineering, University of Sheffield Sheffield S1 3JD (UK)

(Received: November 11, 1996)

SUMMARY

Manipulators with some flexible links are attractive because they avoid the severe control problems associated with the large inertia forces generated when the large-mass, rigid links in conventional robot manipulators move at high speed. In fact, only two of the links within a typical six degrees of freedom revolute-geometry industrial robot cause significant inertia forces, and so only these two links need to be flexible. The development of a two-flexible-link system controller is therefore very relevant to larger manipulators, because it can be readily expanded by adding simple controllers for the other rigid links. Two alternative controllers are developed in this paper, a computed-torque controller and a quadratic optimal controller. Simulations confirm the superior performance of the latter.

1. INTRODUCTION

With the increasing demand for faster robot movements in manufacturing operations, there is now widespread interest in developing low-mass, flexible-link robot manipulators which avoid the severe control problems caused by the large inertia forces generated when the large-mass, rigid links associated with conventional robot manipulators move at high speed. The necessary pre-requisite for a flexible manipulator controller is the existence of a suitable-accurate model of the manipulator system. In practice, a typical revolute-geometry industrial robot manipulator has six degrees of freedom but the problematical inertia forces are due to just two of the links within it and so only these two links have to be lightweight, flexible ones. Hence, the necessary manipulator model can be divided into two connected sub-systems: a two-flexible-link model and a model of the other rigid links. The detailed development of a two-flexible-link system model has been described previously,[1] and so, only an overview is presented in this paper. The method of approach is to develop an accurate single-flexible-link model and then to expand this into a two-flexible-link model, taking proper account of the coupling and interactions between the two links. As errors are cumulative due to the dynamic coupling, it is essential that the basic single-link model on which the two-link model is built is of very high accuracy.

The controller of a flexible manipulator system must fulfil two functions. Firstly, it must compensate for the static deflection of the flexible links under gravity forces and, secondly, it must act to reduce both the magnitude and time duration of link oscillations which arise naturally out of its flexibility. A computed-torque controller is able to fulfil both of these functions to a limited extent, but simulations comparing its performance with that of a quadratic optimal controller confirm the superior performance of the latter.

2. MODEL FOR A SINGLE FLEXIBLE LINK

The assumed mode method (AMM) is a computationally efficient scheme which serves as a useful starting point in formulating a flexible link model. Assuming the magnitude of flexure to be low, the slope and static deflection of a flexible beam bending under gravity are described by:

$$\frac{du_m}{dx} = -\frac{mg}{2EIl}\left(l^2x - lx^2 + \frac{x^3}{3}\right); \qquad u_m = -\frac{mg}{2EIl}\left(\frac{l^2x^2}{2} - \frac{lx^3}{3} + \frac{x^4}{12}\right) \tag{1}$$

where m is the mass of the beam, l is the length of the beam, EI is the flexural stiffness of the beam, g is the gravity vector, x is the position on the beam of the point where the slope and deflection is measured and the subscript m denotes the slope or deflection resulting from the mass of the beam.

For a flexible link with an end-tip load m_t (Figure 1(a)), the mass m_t produces a negative slope and deflection given by:

$$\frac{du_{mt}}{dx} = -\frac{m_t g}{EI}\left(lx - \frac{x^2}{2}\right); \quad u_{mt} = -\frac{m_t g}{EI}\left(\frac{lx^2}{2} - \frac{x^3}{6}\right) \tag{2}$$

By the principle of superposition the total static slope and deflection for a flexible link are given by:

$$\frac{du}{dx} = \frac{du_m}{dx} + \frac{du_{mt}}{dx} = -\frac{g}{2EIl}\left((m + 2m_t)\frac{l^2x}{2} - (m - m_t)\frac{lx^2}{3} + m\frac{x^3}{6}\right) \tag{3}$$

$$u = u_m + u_{mt} = -\frac{g}{2EIl}\left((m + 2m_t)\frac{l^2x^2}{2} - (m + m_t)\frac{lx^3}{3} + m\frac{x^4}{12}\right) \tag{4}$$

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

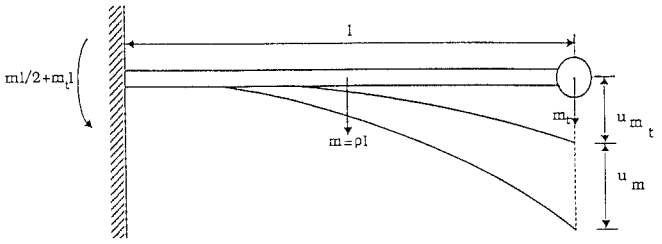


Fig. 1(a). Deflection of a flexible link due to its mass and the payload.

The maximum static slope and deflection of the flexible link occur at the free end, where $x = l$, i.e.

$$\frac{du_{\max}}{dx} = -\frac{l^2 g}{2EI}(m/3 + m_t);$$
$$u_{\max} = -\frac{l^3 g}{2EI}(m/4 + 2m_t/3) \tag{5}$$

The deflection of the link end-tip is calculated in the above equations on the assumption that the end-tip moves vertically downwards instead of in a circular arc. This is clearly only valid if the magnitude of flexure is low. This condition is unlikely to be satisfied in typical industrial flexible manipulator links, and modification of the equations is therefore necessary. Previous work[1] has shown that the case of large magnitude flexure can be handled by adding a correction factor to the basic equations. This is calculated by considering the link as a body composed of n equal sections and applying finite element analysis. The corrected coordinates of the end-tip are then given by:[1]

$$x_e = l - s; \quad y_e = u(l - s) \tag{6}$$

where:

$$u(l - s - w_n) = \frac{w_n}{v_n} u(l - s) \tag{7}$$

$$s = \sum_{i=1}^{n-1} w_i; \quad v_n = L - l/n; \quad w_n = \frac{v_n l}{nL};$$
$$L = \sqrt{[u(l - s) - u((n-1)l/n - s)]^2 + [l/n]^2} \tag{8}$$

2.1. Dynamic modelling

The equation of motion of an undamped flexible link without payload is described by:[2]

$$\rho \frac{\partial^2 u(x,t)}{\partial t^2} = -\frac{\partial^2}{\partial x^2}\left(EI \frac{\partial^2 u(x,t)}{\partial x^2}\right); \quad u(x,t) = \phi(x)q(t) \tag{9}$$

where ρ is the mass per unit length of the link, $u(x, t)$ is the deflection of the link, $\varphi(x)$ is the assumed mode shape function and $q(t)$ is the modal function. Assuming that EI is a constant, equation (9) can be written as:

$$\frac{1}{q(t)}\frac{d^2 q(t)}{dt^2} = -\frac{EI}{\rho}\frac{1}{\phi(x)}\frac{d^4 \phi(t)}{dx^4} \tag{10}$$

which leads to the two following differential equations:

$$\frac{d^4\phi(x)}{dx^4} - \beta^4 \phi(x) = 0; \quad \frac{d^2 q(t)}{dt^2} + \omega^2 q(t) = 0 \tag{11}$$

where ω is a constant and $\beta^4 = \rho\omega^2/EI$. The solution, as given in[2], is:

$$\phi_i(x) = C_i(\cos \beta_i x - \cosh \beta_i x) + (\sin \beta_i x - \sinh \beta_i x);$$
$$q_i(t) = A_i \cos \omega_i t + B_i \sin \omega_i t \tag{12}$$

where A_i, B_i, C_i, and ω_i are constants, i denotes the number of modes of vibration. The deflection is then given by:

$$u(x, t) = \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \tag{13}$$

From the boundary conditions $(u(0, t) = u(l, t) = \partial u(0, t)/\partial x = \partial^2 u(l, t)/\partial x^2 = 0)$ we obtain

$$C_i = \frac{\cos \beta_i l + \cosh \beta_i l}{\sin \beta_i l - \sinh \beta_i l} \tag{14}$$

and β_i as a solution to:

$$\cos \beta_i l \cdot \cosh \beta_i l = -1 \tag{15}$$

Solving equation (15) for the first four modes gives $\beta_1 l = 1.875$, $\beta_2 l = 4.694$, $\beta_3 l = 7.854$ and $\beta_4 l = 10.995$. From here, using the definition that $\beta^4 = \rho\omega^2/EI$, we can deduce the values of the natural frequencies ω_i of the flexible link for the first four modes. This means that, given an initial excitation F, the link is going to oscillate according to a combination of these four natural frequencies. The equation of motion can be generalised as an eigenvalue problem linking the two parts of the system (the assumed mode shape functions $\varphi_i(x)$ and the modal functions $q_i(t)$. Subsequent analysis[1] taking into account the first three modes ($i = 1, 2$ and 3) leads to the following equations for the vertical displacement $u(x, t)$ of any point x on the link at any time t, the slope $u'(x, t)$ of the link at any point x and any time t and the velocity $\dot{u}(x, t)$ of any point x on the link at any time t:

$$u(x, t) = \varphi_1(x)q_1(0)\cos(\omega_1 t) + \varphi_2(x)q_2(0)\cos(\omega_2 t) + \varphi_3(x)q_3(0)\cos(\omega_3 t) \tag{16}$$

$$u'(x, t) = \frac{\partial u(x, t)}{\partial x} = \phi_1'(x)q_1(0)\cos(\omega_1 t) + \varphi_2'(x)q_2(0)\cos(\omega_2 t) + \varphi_3'(x)q_3(0)\cos(\omega_3 t) \tag{17}$$

$$\dot{u}(x, t) = \frac{\partial u(x, t)}{\partial t} = \varphi_1(x)q_1(0)\omega_1 \sin(\omega_1 t) + \varphi_2(x)q_2(0)\omega_2 \sin(\omega_2 t) + \varphi_3(x)q_3(0)\omega_3 \sin(\omega_3 t) \tag{18}$$

subject to the initial conditions:

$$u(x, 0) = \sum_{i=1}^{\infty} \varphi_i(x)q_i(0) = f(x);$$
$$\dot{u}(x, 0) = \sum_{i=1}^{\infty} \varphi_i(x)\dot{q}_i(0) = g(x) \tag{19}$$

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

Using the orthogonal relation, the corresponding initial conditions in the *normal coordinates* (the normalisation or weighting is operated on all modes) are:

$$q_i(0) = \frac{\rho l}{m_{ii}} \int_0^l f(x)\varphi_i(x)\,dx;$$
$$\dot{q}_i(0) = \frac{\rho l}{m_{ii}} \int_0^l g(x)\varphi_i(x)\,dx \tag{20}$$

Similarly, $q_i(0)$ and $\dot{q}_i(0)$ can be obtained from the normalised flexural stiffness as

$$q_i(0) = \frac{EI}{k_{ii}} \int_0^l f''(x)\varphi_i''(x)\,dx;$$
$$\dot{q}_i(0) = \frac{EI}{k_{ii}} \int_0^l g''(x)\varphi_i''(x)\,dx \tag{21}$$

When an end-tip load is added to the link, an extra eigenvalue will appear in the boundary conditions and it can be shown[1] that the effect is to cause the link to vibrate at a slower frequency and for vibrations to persist for a longer period of time.

2.2 Inclusion of shear deformation effect

The assumed mode method calculates link deformation on the assumption that this is due only to the bending moment created by the mass and end-tip load of the link. This assumption appears to have been made in all flexible link models previously reported. However, a shear force also exists which acts in the opposite direction to the bending moment and, in the interests of accurate modelling, its effect must be included in both static and dynamic models of a flexible link. It is known that the shear force of flexible arms depends on the shape of the cross-section of the arm. Therefore, a physical quantity called the *numerical factor,* representing the geometric characteristics of the link cross-section, is required in the dynamic formulation of the manipulator. The numerical factor[3] of a flexible beam is defined as:

$$K = \frac{AQ}{I_a d} \tag{22}$$

where, I_a is the moment of inertia of the cross-sectional shape of the link computed with respect to its neutral axis, Q denotes the first moment about the neutral axis of the area contained between an edge of the cross-section of the beam parallel to the main axis and the surface at which the shear stress is to be computed, A is the cross-sectional area and d is the width of the cross-sectional area at which the shear deformation is required. For a uniform link of square cross section, the factor K is given by:

$$K = \frac{AQ}{I_a d} = \frac{d^2(\rho l)l^2/2}{(\rho l)l^2\, d/3} = \frac{3d}{2} \tag{23}$$

An element dx of the flexible link is deformed by the shear force V and the bending moment M shown in Figure 1(b). When the shear force is zero, the centre line

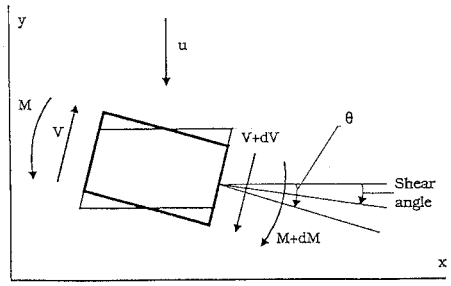


Fig. 1(b). Deformation of link due to shear forces.

of dx is normal to the face of the cross-section. If $\partial u(x, t)/\partial x$ is the slope due to the bending moment M, neglecting the interaction between the shear and the moment, the shear force will cause a rectangular element to become a parallelogram without a rotation of the faces. Thus, the slope of the deflection curve is decreased by the shear angle as formulated in the following equation:

$$\frac{\partial u_c(x, t)}{\partial x} = \frac{\partial u(x, t)}{\partial x} - \frac{V}{KAG} \tag{24}$$

where V is the value of the shear force and is equal to $EI\rho$, G the shear modulus of the material the link is made of and $\partial u_c/\partial x$ the total slope cause by both shear and moment. As a result of the above formulation, the equation of motion of an undamped flexible link after addition of the shear deformation becomes

$$EI\frac{\partial^4 u_c}{\partial x^4} + \rho\frac{\partial^2 u_c}{\partial t^2} - \frac{EI\rho}{KAG}\frac{\partial^4 u_c}{\partial x^2\,\partial t^2} = 0 \tag{25}$$

This equation is very difficult to solve because of the last term in the left hand-side. However, supposing that the shear force affects only the modal functions $q_i(t)$, a solution is found as being[4]

$$\varphi_i(x) = A_i(\cos(\beta_i x) - \cosh(\beta_i x)) + \sin(\beta_i x) - \sinh(\beta_i x);$$
$$q_{ci}(t) = q_{ci}(0)\cos(\omega_{ci}t + \Psi) \tag{26}$$

where β_i are obtained in the same manner as for the system without shear deformation, (i.e., as in section 2.1) and ω_{ci} are the transformed natural frequencies including the shear deformation and are given by

$$\omega_{ci} \equiv \omega_i\left[1 - \frac{EI}{2KAG}\beta_i^2\right] \tag{27}$$

ω_i being the natural frequencies of the system without shear deformation.

The angle Ψ is equal to:

$$\Psi = \frac{V}{KAG} = \frac{EI\rho}{KAG}$$

From this set of equations, we can see how the shear deformation decreases the natural frequencies of vibration of the link. The effect is more pronounced for the higher modes because of the existence of the term β_i^2 in the equation for ω_{ci}. The total slope is also decreased

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

by the amount Ψ. Thus it is apparent that the shear deformation acts similarly to a load on the frequency of the system, but operates in the opposite manner for the slope.

3. EXTENSION OF MODEL TO TWO-FLEXIBLE-LINK SYSTEM

The main difficulty in modeling multi-link flexible manipulators is that the rigid motion and the elastic motion are coupled together, and the elastic motion has direct effects on the transformation matrix between the link coordinates and the global coordinates. Due to the complexity of the problem, the modeling of flexible manipulators is initially simplified by neglecting the effect of the elastic motion on the transformation matrix and neglecting the effect of the elastic motion on the rigid motion. If the rigid motion is not affected by the elastic motion, the rigid system dynamic equations can be derived using the Lagrange–Euler principle. These equations can then be used to predict the dynamic stress and elastic deformation of the system, by applying the respective torques obtained for the rigid motion to the dynamic equations describing the elastic motion. For the two-link flexible case, this is difficult because of the cross-interaction between the two links. The task of modeling a two-link flexible manipulator is made even harder by the fact that this cross-interaction between the two links is permanently present, i.e., a small disturbance at the end-tip of the first link will cause this link to start a vibrational motion, causing the second link to engage in a motion that will affect the vibrational motion of the first link, and so on....

This problem can be avoided without compromising on accuracy if the first type of cross-interaction is ignored, i.e. by assuming that any energy produced in the second link is absorbed through the actuator of this link and therefore not propagated to the first link. A further simplification can be achieved by considering only the first three flexible modes, since higher modes have negligible influence on the behaviour of the system.[1] The full derivation of the two-flexible-link system model is given elsewhere,[1] and so only a summary of this is provided here: The rigid motion of a two-link manipulator can be described in terms of the Lagrange–Euler formulation:

$$\frac{d}{dt}\left[\frac{\partial L}{\partial \dot{\theta}_i}\right] - \frac{\partial L}{\partial \theta_i} = \tau_i; \quad i = 1, 2. \tag{29}$$

where L is the Lagrangian function and is equal to $K - P$, K is the total kinetic energy of the robot arm, P is the total potential energy of the robot arm, θ_i are the angular joint positions and τ_i are the generalised torques applied to the system at joint i to drive link i. These torques can be expressed in matrix form as:

$$\tau(t) = D(\theta(t))\ddot{\theta}(t) + h(\theta(t), \dot{\theta}(t)) + c(\theta(t)) \tag{30}$$

where:

$\tau(t)$ is a 2×1 generalised torque vector applied at joints $i = 1, 2$.

$\theta(t)$ is a 2×1 vector of the joint positions,

$\dot{\theta}(t)$ is a 2×1 vector of the joint velocities,

$\ddot{\theta}(t)$ is a 2×1 vector of the joint accelerations,

$D(\theta(t))$ is a 2×2 inertial acceleration-related symmetric matrix.

3.1 State-space representation of the flexible system

The equation of elastic motion of a flexible link which is part of a multi-link system can be written as:[1,2]

$$u_j(x, t) = \sum_{i=1}^{n} \varphi_{ij}(x) q_{ij}(t) \tag{31}$$

where the subscript j denotes the link number ($j = 1$ for the first link), the subscript i denotes the mode number, $u_j(x, t)$ is the vertical deflection of the link j at the distance x and time t, $\varphi_{ij}(x)$ is a shape function,[2] and $q_{ij}(t)$ is a modal function solution of the following second order differential equation:

$$\frac{d^2 q_{ij}(t)}{dt^2} + \frac{c_j}{m_{ij}} \frac{dq_{ij}(t)}{dt} + \omega_{ij}^2 q_{ij}(t) = \tau_j(t) \tag{32}$$

where c_j is the damping coefficient of the link, m_{ij} is the normalised mass for each mode i of the link j and ω_{ij} is the corresponding frequency equal to $\sqrt{k_{ij}/m_{ij}}$, k_{ij} being the normalised stiffness of the ith mode of the jth link. Typically, the contributions of the flexible modes attenuate rapidly with frequency such that it is always possible to characterise the system dynamics to any required degree of accuracy with only a finite number of the lower modes. Considering only the first three flexible modes of each link, the flexible system can be described in the following state variable form:

$$\begin{Bmatrix} \dot{q}_{1j} \\ \ddot{q}_{1j} \\ \dot{q}_{2j} \\ \ddot{q}_{2j} \\ \dot{q}_{3j} \\ \ddot{q}_{3j} \end{Bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\omega_{1j}^2 & \frac{-c_j}{m_{1j}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\omega_{2j}^2 & \frac{-c_j}{m_{2j}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -\omega_{3j}^2 & \frac{-c_j}{m_{3j}} \end{bmatrix} \times \begin{Bmatrix} q_{1j} \\ \dot{q}_{1j} \\ q_{2j} \\ \dot{q}_{2j} \\ q_{3j} \\ \dot{q}_{3j} \end{Bmatrix} + \begin{bmatrix} 0 \\ b_{1j} \\ 0 \\ b_{2j} \\ 0 \\ b_{3j} \end{bmatrix} \tau_j(t) \tag{33}$$

with the position vector given by:

$$u_j(x, t) = [\varphi_{1j}(x)\, 0\, \varphi_{2j}(x)\, 0\, \varphi_{3j}(x)\, 0] \times [q_{1j}\, \dot{q}_{1j}\, q_{2j}\, \dot{q}_{2j}\, q_{3j}\, \dot{q}_{3j}]^T \tag{34}$$

This description can also be expressed in the following simplified form:

$$\dot{q}_j = A_j q_j + B_j \tau_j; \quad u_j(x) = C_j(x) q_j \tag{35}$$

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

The constants b_{ij} are obtained after normalisation of the torques τ_j for each mode of the flexible link j. The state-space representation given by equation (33) is incomplete without the initial conditions relating to each link such as static deflection, relative position of each link in the reference frame and effect of the first link on the second link. These points will be discussed in the next sections by studying each link separately.

3.1.1 Link 1. A state-space representation of the elastic motion of the first flexible link is obtained by setting $j = 1$ in equations (33, 34). The mass term m_{i1} is then the mass of the payload for the first link, which consists of the mass m_a of the actuator of the second link, the mass ρl of the second link and the mass m_t of the payload at the end-tip of the second link, i.e.:

$$m_{i1} = \int_0^l [\rho + m_t \delta(x - l)]\varphi_{i1}^2 \, dx; \quad i = 1, 2, 3. \tag{36}$$

The normalised flexural stiffness k_{i1} relative to each mode is given by:

$$k_{i1} = \int_0^l EI(\varphi_{i1}'')^2 \, dx; \quad i = 1, 2, 3. \tag{37}$$

The constants b_{i1} are deduced from the normalisation of the principal torque $\tau_1(t)$ applied to the link, and are equal to:

$$b_{i1} = -\frac{\varphi_{i1}(l)}{k_{1i}} \tag{38}$$

The position vector giving the deflection of the link at any point x is

$$u_1(x, t) = [\varphi_{11}(x)\,0\,\varphi_{21}(x)\,0\,\varphi_{21}(x)\,0] \times [q_{11}\,\dot{q}_{11}\,q_{21}\,\dot{q}_{21}\,q_{31}\,\dot{q}_{31}]^T \tag{39}$$

This vector is increased by a value $u_{01}(x)$ corresponding to the static deflection caused by the mass of the link itself and the load attached to its end-tip. So, the deflection vector becomes:

$$u_1(x, t) = [\varphi_{11}(x)\,0\,\varphi_{21}(x)\,0\,\varphi_{21}(x)\,0] \times [q_{11}\,\dot{q}_{11}\,q_{21}\,\dot{q}_{21}\,q_{31}\,\dot{q}_{31}]^T + u_{01}(x) \tag{40}$$

with:

$$u_{01}(x) = -\frac{g}{2EIl}\left((\rho l + 2m_t)\frac{l^2x^2}{2} - (\rho l + m_t)\frac{lx^3}{3} + \frac{\rho l x^4}{12}\right) \tag{41}$$

The motion of the first link can be obtained from the inverse Laplace transform of the following equation:

$$u_1(x, s) = C_1(x)(sI - A_1)^{-1}B_1\tau_1(s) + u_{01}(x) \tag{42}$$

where s is the Laplace operator and I the identity matrix. To determine the deflection u_1 at the end-tip of the first link, the variable x is replaced in equation (41) by l. The limitation of this state-space representation is that the response $u_1(x, t)$ will be relatively correct only for very small deflections. Since a good accuracy in determining the position of the end-tip of the first link is needed to locate the origin of the second link, the necessity in using the correction factor (equations 6–8) arises once again. The difficulty in using the correction factor is that, unlike the normal procedure required by the state-space representation where the variable x is replaced by the value at which the deflection of the link is wanted (in this case $x = l$), calculation of a correction factor requires the value of the deflection all along the link. In order to obtain a good accuracy, the incrementation of the variable x has to be very small (independently from the number of finite elements chosen to approximate the flexible link). All this generates some extensive calculation and increases the computing time needed to obtain a response for the corrected deflection.

3.1.2 Link 2. A state-space representation of the elastic motion of the second flexible link is obtained by setting $j = 2$ in equations (33, 34). The position vector giving the deflection at any point x on the link is:

$$u_2(x, t) = [\varphi_{12}(x)\,0\,\varphi_{22}(x)\,0\,\varphi_{22}(x)\,0] \times [q_{12}\,\dot{q}_{12}\,q_{22}\,\dot{q}_{22}\,q_{32}\,\dot{q}_{32}]^T + u_{02}(x) \tag{43}$$

with $u_{02}(x)$ being the value of the static deflection of the link at a distance x from its origin. This static deflection is caused by the mass ρl of the link plus its payload m_t and is equal to

$$u_{02}(x) = -\frac{g}{2EIl}\left((\rho l + 2m_t)\frac{l^2x^2}{2} - (\rho l + m_t)\frac{lx^3}{3} + \frac{\rho l x^4}{12}\right) \tag{44}$$

In a similar manner to that of the first link, the response of the second link can be obtained from the inverse Laplace transform of the following equation:

$$u_2(x, s) = C_2(x)(sI - A_2)^{-1}B_2(\tau_1(s) + \tau_2(s)) + u_{02}(x) \tag{45}$$

The next step in the modeling of a two-link flexible manipulator consists of repositioning the origin of the second link. This is done by obtaining the corrected horizontal and vertical positions of the end-tip of the first link and assigning these values to the coordinates of the origin of the second link. Then, the vibration and (or) rotation of the first link is used an additional input to the second link. This is considered as the effect of the first link on the second link. This operation is carried out by obtaining the time-varying slope $du_1(l, t)/dx$ of the end-tip of the first link through the correction factor method and adding it to the angular position $\theta_2(t)$ of the second link.

3.2 Final model

The block diagram in Figure 2 summarises the modelling algorithm for the two-link flexible manipulator. The model is divided into two sub-systems, one for each link. Desired coordinates $(x_{1d}, y_{1d}, x_{2d}, y_2d)$ are fed into the system through the block representing the inverse kinematics of the system to generate the desired angles

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

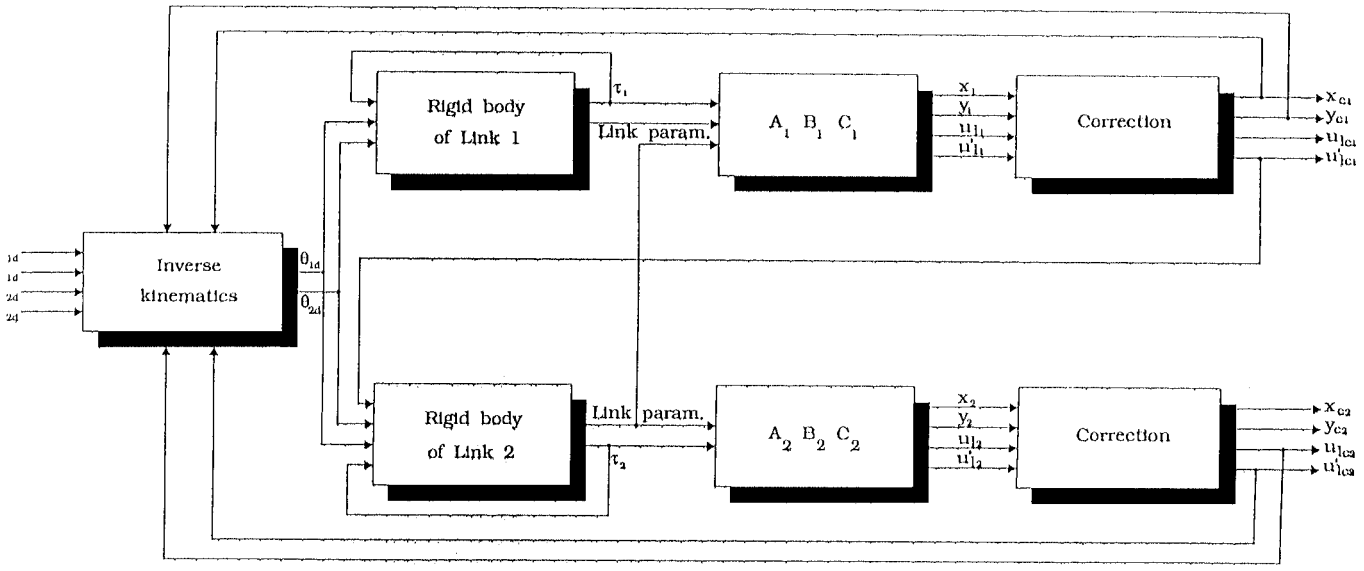


Fig. 2. Block diagram of algorithm for modelling a two-flexible-link manipulator.

of rotation $\theta_{1d}(t)$ and $\theta_{2d}(t)$. This operation can be omitted if these two desired angles are known initially. Both angles are then used as inputs for the rigid system models representing the two links. Outputs such as link masses, actuator mass, payload, stiffness, length and width of each link, etc., are passed to the second part of each sub-system which computes the elastic motion of each link. Other important outputs produced by each of the two rigid model sub-systems are the applied torques $\tau_1(t)$ and $\tau_2(t)$. Each torque induces the corresponding rigid body to rotate according to the corresponding angle of rotation $\theta_d(t)$ and, simultaneously, provokes a forced vibration into the corresponding flexible sub-system. The superposition of both motions (rigid and flexible) produces a set of variables for each link; among these are the end-tip vertical positions, the end-tip horizontal positions, the end-tip deflections and the end-tip slopes. Using the Correction Factor Method, all of these outputs are recalculated. As a result, some of them will be used as a feed-back to the whole system to test if the rotations are accomplished, while others (the end-tip slope of link 1, for instance) will be added to the angular position of the rigid body of the second link, therefore quantifying the effect that the first link has on the second link.

4. CONTROLLER DEVELOPMENT

Control techniques for rigid manipulators are now reasonably well developed. Since the dynamics at the hub of such manipulators is the same as the dynamics at the end-tip, the magnitude of the driving torques and forces causing the motion of a rigid manipulator can be used to accurately predict the position of the end-tip at any time. The absence of flexibility enables the manipulator to move without the occurrence of vibrations. Therefore, control techniques are designed to optimise the transition time between the initial position and the desired position of the end-effector. However, in the case of flexible-link manipulators, optimisation of the transition time between two positions of the end-point of the manipulator can generate unwanted characteristics such as vibrations. Depending on the degree of flexibility of the manipulator and the time taken to reach the desired position, the amplitude of these vibrations can be very significant. In consequence, the time gained in increasing the speed of the end-effector between the initial and the final positions may be lost in waiting for the vibrations to settle down.

Early attempts to design an efficient control law that will allow an optimisation of both transition time (fast response) and settling time (decrease of the vibrations) involved linearising the equations of motion of the manipulator about a nominal configuration and applying several linear control schemes,[5–7] but the lack of accuracy in the model design and the high number of approximations needed in the linearisation produced a system which was far from being realistic. Shaped torque techniques have been proposed[8] to minimise the residual vibtrations in flexible manipulators. This technique has been further developed to suppress multiple mode variations.[9] The computed torque method which was originally developed for rigid manipulators[10] was also tried on flexible link systems. The complexity of the inverse dynamics makes a straightforward application of the computed torque method or feedback linearisation impossible: instead, some approximate schemes have been proposed[11] for open and closed-loop control.

The main drawback of all model-based controllers is the difficulty in obtaining the exact model necessary. Therefore, the robustness to parameter uncertainties has been a major concern in control design for flexible manipulators.[12] Another difficulty with the flexible systems is the so called "spillover" problem which occurs when one of the links is vertical. Since the actual system is a distributed parameter system, any designed controller based on finite dimensional models will generally suffer from an inability to control or observe these spillovers.[13] Independent joint PD (proportional plus derivative) controllers have been shown to be stable

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

for rigid manipulators.[12] The same strategy was experimented with on a two-link rigid-flexible manipulator[14] with satisfactory results, but, since their design was based on a linearised model, the manoeuver angles were restricted to small values. Recent research on one-link flexible manipulators has suggested separating the flexible system from the rigid system and controlling the oscillations of the link by the use of quadratic programming. Previous results[15] using this demonstrated serious inaccuracies for large angle motions. However, the modified form of this controller described in section 4.2, based on the improved-accuracy dynamic model developed in section 3, performs much better, as the simulations presented demonstrate.

Before engaging in any trajectory planning, and therefore control strategy, for the two-link flexible manipulator, it is necessary to initially compensate for the two static deflections present at the end-tip of each link. This depends on the link characteristics such as length, cross-sectional area, flexural stiffness and mass, plus characteristics linked to the payload carried by the link such as its mass and position on the link. A correction procedure was designed to obtain the value of the deflection and the corresponding slope at any point of the link. Then, each link is rotated until its vertical coordinate coincides with the vertical coordinate of its rigid counterpart. In other words, if the rigid body of the manipulator is initially positioned on the horizontal axis ($y_{r1}(l_1, 0) = y_{r2}(l_2, 0) = 0$, where the subscript r relates to the rigid body), both flexible links are rotated upwards with an angle equal to:

$$\varphi = -\tan^{-1}\left(\frac{y_{stat1}(l_1)}{x_{stat1}(l_1)}\right) - \tan^{-1}\left(\frac{y_{stat2}(l_2)}{x_{stat2}(l_2)}\right) \qquad (46)$$

where the subscript *stat* relates to the end-tip coordinates when subject to static deflection.

4.1 Computed-torque (open-loop) control

Computed torque control is a well-established open-loop method for rigid manipulators. It can also be extended to flexible manipulators, but careful design is then necessary because of the tendancy of the link end-tip to vibrate with an inacceptable amplitude if the speed of rotation is too high. For this reason, the stability of the open-loop of both flexible subsystems must be carefully analysed.

From equations (42) and (45), the open-loop transfer function giving the relationship between the input torques $\tau_j(t)$ and the output variables $u_j(x, t)$ can be written as:

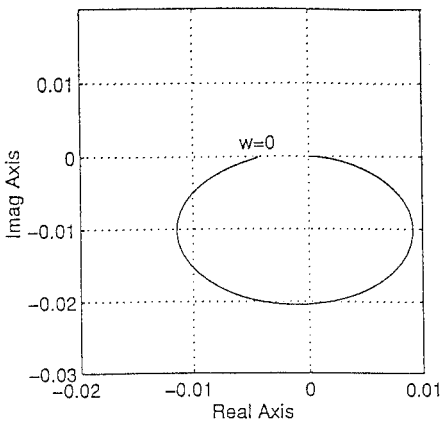
$$u_j(x, s) = C_j(x)(sI - A_j)^{-1}B_j\tau_j(s) + u_{0j}(x); \quad j = 1, 2. \qquad (47)$$

Concentrating only on the dynamic response of each subsystem, the corresponding transfer functions can be rewritten as:

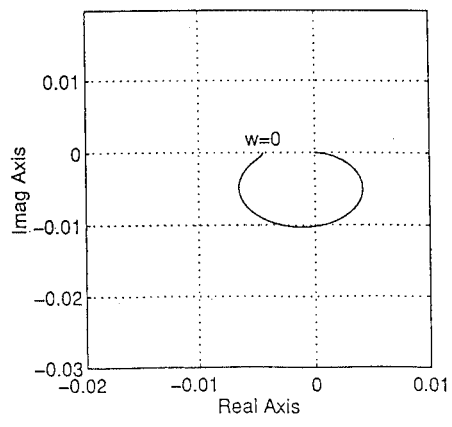
$$G_j(x, s) = \frac{u_j(x, s) - u_{0j}(x)}{\tau_j(s)} = C_j(x)(sI - A_j)^{-1}B_j; \quad j = 1, 2. \qquad (48)$$

The Matlab package was used to determine the system poles and zeros. This showed that the open-loop poles are $-1.12 \pm 10.24i$, $-2.81 \pm 52.65i$ and $-7.32 \pm 124.33i$ for the first link subsystem and $-2.25 \pm 10.06i$, $-5.62 \pm 52.42i$ and $-14.24 \pm 123.40i$ for the second subsystem. These correspond to the first, second and third flexible modes respectively for each link. The relative magnitude of these three pole-pairs for each link demonstrates the dominance of the first and second flexible modes in the stability of both flexible subsystems and in the response of the system as a whole. The pole-pair associated with the first mode of link one is particularly dominant since the payload is relatively high.

A Nyquist plot (Figure 3) shows the robustness of the open-loop transfer functions. The gain of the transfer function of the first link is obviously more important than that of the second link, proving that the same angle of rotation will produce vibrations of a larger amplitude for the first link, and therefore, a longer settling time. To achieve the control of the vibrations occurring towards the end of the rotation of each link, a straightforward computed torque technique is derived by producing a smooth trajectory for both flexible links in a way that the energy produced by each torque is conserved, and the rotation time is stretched to an optimal value, therefore



(a) First elastic sub-system



(b) Second elastic sub-system

Fig. 3. Nyquist diagrams for the open loop transfer functions.

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

limiting the amplitude of the residual vibrations and shortening the settling time for the links to regain their static positions.

The relationship between the applied torque $\tau_j(t)$ and the dynamic deflection $u_j(x, t)$ is given by:

$$u_j(x, s) = C_j(x)(sI - A_j)^{-1}B_j\tau_j(s) + u_{0j}(x)$$
$$= G_j(x, s)\tau_j(s) + u_{0j}(x) \quad j = 1, 2. \tag{49}$$

where $|G_j(x, i\omega)|$ is the gain of amplification. Control of the amplitude of vibration of the links can be achieved by limiting the magnitude of the input τ_j. The control procedure is then to calculate the torques according to:

$$\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} = [D(\theta_1, \theta_2)]\begin{bmatrix}\ddot{\theta}_1\\ \ddot{\theta}_2\end{bmatrix} + [h(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2)]$$
$$+ [c(\theta_1, \theta_2)] \tag{50}$$

such that the following conditions are satisfied:

$$|u_1(x, i\omega)| \le L\{u^d_{1\max}(x)\}; \quad |u_2(x, i\omega)| \le L\{u^d_{2\max}(x)\} \tag{51}$$

where $u^d_{1\max}(x)$ and $u^d_{2\max}(x)$ are the maximum desired amplitudes for the vibrations at the distance x from the origin of the respective link. Substituting equation (48) into equation (49), the necessary conditons become:

$$|\tau_1(x, i\omega)| \le L\{u^d_{1\max}(x)\} \cdot |G_1(x, i\omega)|^{-1};$$
$$|\tau_2(x, i\omega)| \le L\{u^d_{2\max}(x)\} \cdot |G_2(x, i\omega)|^{-1} \tag{52}$$

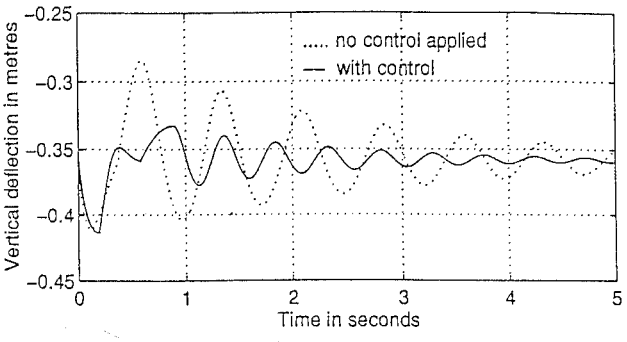
In order to obtain a dynamic response for the link vibrations within the desired scale set by the maximum desired values $u^d_{1\max}(x)$ and $u^d_{2\max}(x)$, the respective torques to be delivered to the actuators of both links should satisfy the control condition given by:

$$|\tau_j(x, i\omega)| \le L\{u^d_{j\max}(x)\} \cdot |G_j(x, i\omega)|^{-1}$$
$$= L\{\tau^d_{j\max}\}^d \quad j = 1, 2. \tag{53}$$

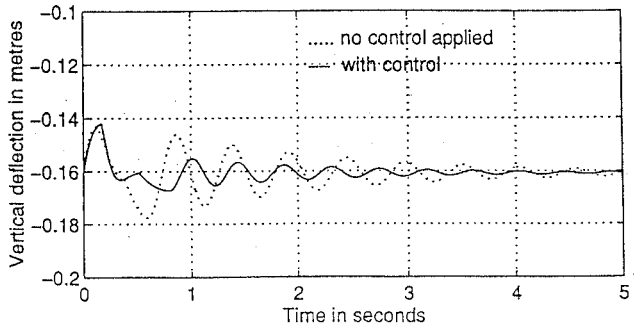
The aim of this control technique is to find the desired functions giving $\theta_1(t)$ and $\theta_2(t)$ that will produce a smooth trajectory for both end-tips and therefore dampen the residual vibrations of the flexible links. The nonlinearity of the inverse dynamics equations of the rigid mode make this control procedure very difficult to implement.[14] It can be seen from equation (50) that to obtain $\theta_1(t)$ and $\theta_2(t)$ from $\tau_1(t)$ and $\tau_2(t)$ requires parallel programming for a set of highly nonlinear differential equations. A solution adopted was to use a trial and error method, involving setting an initial trajectory for both links, calculating the respective torques, comparing these torques to the conditions provided by the control procedure, then, if the conditions are not satisfied, readjusting the trajectories by increasing the time of rotation.

The control strategy adopted was to apply the computed torque technique to the part of the motion where the braking procedure is occurring. The first phase of the motion (acceleration phase) was left intact to optimise on the time criterion. But as soon as the links start to decelerate, the control procedure limits the value of the braking torque to allow a smoother motion, and hence smaller oscillations. The first link rotates at a slower speed than the second link because, to satisfy the conditions listed in equation (50), each link has to be rotated according to a specific optimal time dictated by the control law applied on the braking torque. Since the first link is subjected to more inertia, its time of rotation is increased accordingly.

The computed torque control strategy was tested by simulating a $+30°$ rotation of the first link and a $-30°$ rotation of the second link. For this simulation, the maximum amplitudes for vibration specified in equation (51) were set at $u^d_{1\max}(l) = 24$ mm and $u^d_{2\max}(l) = 8$ mm. In the absence of any control law, the rigid mode of the first link rotated by an amount of 30° in 1.18 seconds. After applying a computed torque control on the braking torque relative to the first link, the same rotation was effected in 1.72 seconds. For the second link, the rigid mode is rotated by an amount of $-30°$ in 1.18 seconds when uncontrolled, and 1.5 seconds when subject to the computed torque control. Figure 4 illustrates of the end-tip motion of the links, with and without the computed torque controller. It is clear that controlling the amplitude of the braking torque improves the behaviour of the end-tip motion of each link. The



(a) Link 1



(b) Link 2

Fig. 4. End-tip deflection with and without computed-torque control.

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

uncontrolled links are subjected to a very pronounced braking process and serious vibrations occur at the end of the rotation. However, with control applied, the amplitude of the residual link vibrations is much reduced and the settling time is correspondingly decreased.

Even though the computed torque control is an efficient method of control for flexible manipulators, the difficulty in solving the highly nonlinear equations describing the inverse dynamics of the system is a very important drawback. The computation burden does not favour an on-line real-time application for this method. As a result, research was directed to closed-loop control and quadratic optimal control in particular.

4.2 Quadratic optimal (closed-loop) control

Given a system described by the following state-space representation:

$$\dot{x} = Ax + Bu \tag{54}$$

minimising some function of the error signal will produce the following quadratic performance index:

$$J = \int_0^T [x^d(t) - x(t)]' Q [x^d(t) - x(t)]\, dt \tag{55}$$

where $x^d(t)$ represents the desired state, $x(t)$ the actual state (thus, $x^d(t) - x(t)$ is the error vector), Q a positive-definite matrix, and the time interval $0 \le t \le T$ is either finite or infinite. The superscript $'$ indicating here the transpose of a vector or matrix. In addition to considering errors as a measure of system performance, the energy required for the control action is usually added to the performance index. Since the control signal may have the dimension of force or torque, the control energy is proportional to the integral of such control signal squared. If the error function is minimised regardless of the energy required, then a design may result that calls for overly large values of the control signal. This is undesirable since all physical systems are subject to saturation. Large-amplitude control signals are ineffective outside the range determined by saturation. Thus, practical considerations place a constraint on the control vector:

$$\int_0^T u'(t) R u(t)\, dt = J_R \tag{56}$$

where R is a positive-definite matrix and J_R a positive constant. The performance index of a control system over the time interval $0 \le t \le T$ may then be written, with the use of a Lagrange multiplier λ, as

$$J = \int_0^T [x^d(t) - x(t)]' Q [x^d(t) - x(t)]\, dt + \lambda \int_0^T u'(t) R u(t)\, dt \tag{57}$$

The Lagrange multiplier λ is a positive constant indicating the weight of control cost with respect to minimising the error function. In this formulation $u(t)$ is unconstrainted. Design based on this performance index has a practical significance that the resulting system compromises between minimising the integral error squared and minimising the control energy. If $T = \infty$ and the desired state $x^d(t)$ is the origin ($x^d(t) = 0$), then the preceding performance index can be expressed as:

$$J = \int_0^\infty [x'(t) Q x(t) + u'(t) R u(t)]\, dt \tag{58}$$

where λ has been included to the matrix R. A choice of weighting matrices Q and R is in a sense arbitrary. Although minimising an "arbitrary" quadratic performance index may not seem to have much significance, the advantage of the quadratic optimal control approach is that the resulting system is a stable system. This approach is sometimes a better alternative to the pole placement approach. The optimisation can also be operated on the output vector ($y(t) = Cx(t)$) instead of the state vector $x(t)$, transforming the performance index into

$$J = \int_0^\infty [y'(t) Q y(t) + u'(t) R u(t)]\, dt = \int_0^\infty [x'(t) C' Q C x(t) + u'(t) R u(t)]\, dt \tag{59}$$

In the case of a two-link flexible manipulator, the state-space representation of the flexible subsystems is given by equation (33), where $j = 1, 2$ represents the link number. The problem of determining the optimal control torques, $\tau_j(t)$, $j = 1, 2$, can be solved by minimising the following performance index for each flexible link

$$J_j = \int_0^\infty [u_j'(x, t) Q_j u_j(x, t) + \tau_j'(t) R_J \tau_j(t)]\, dt \quad j = 1, 2. \tag{60}$$

where $u_j(x, t) = C_j(x) \underline{q_j}(x, t)$, $\underline{q_j}(x, t)$ is the vector of state variables of each link at a distance x. Since the control operates on the end-tip response, x is replaced by l. Replacing the torques $\tau_j(t)$ by the feed-back command $-K_j(x) \underline{q_j}(x, t)$ gives the new performance index:

$$J_{lj} = \int_0^\infty \underline{q_{lj}'}(t) [C_{lj}' Q_j C_{lj} + K_{lj}' R_J K_{lj}] \underline{q_{lj}}(t)\, dt \quad j = 1, 2. \tag{61}$$

where the subscript lj relates to the end-tip of each flexible link. The performance index is then solved according to the second method of Liapunov by assuming that

$$\underline{q_{lj}'}(t) [C_{lj}' Q_j C_{lj} + K_{lj}' R_j K_{lj}] \underline{q_{lj}}(t) = -\frac{d}{dt} (\underline{q_{lj}'}(t) P_j \underline{q_{lj}}(t)) \quad j = 1, 2. \tag{62}$$

where P_j is a positive-definite matrix of the form

$$P_j = \begin{bmatrix} p_{j11} & p_{j12} & p_{j13} & p_{j14} & p_{j15} & p_{j16} \\ p_{j12} & p_{j22} & p_{j23} & p_{j24} & p_{j25} & p_{j26} \\ p_{j13} & p_{j23} & p_{j33} & p_{j34} & p_{j35} & p_{j36} \\ p_{j14} & p_{j24} & p_{j34} & p_{j44} & p_{j45} & p_{j46} \\ p_{j15} & p_{j25} & p_{j35} & p_{j45} & p_{j55} & p_{j56} \\ p_{j16} & p_{j26} & p_{j36} & p_{j46} & p_{j56} & p_{j66} \end{bmatrix}$$

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

Equation (62) can be analytically developed as follows:

$$\begin{aligned}&\underline{q}'_{lj}(t)[C'_{lj}Q_jC_{lj}+K'_{lj}R_jK_{lj}]\underline{q}_{lj}(t)\\&\quad=-\underline{\dot{q}}'_{lj}(t)P_j\underline{q}_{lj}(t)-\underline{q}'_{lj}(t)P_j\underline{\dot{q}}_{lj}(t)\\&\quad=-\underline{q}'_{lj}[(A_j-B_jK_{lj})'P_j+P_j(A_j-B_jK_{lj})]\underline{q}_{lj}\end{aligned}\quad(63)$$

Comparing both sides of this last equation and noting that this equation must hold true for any $\underline{q}_{ij}$, the requirement becomes:

$$\begin{aligned}&(A_j-B_jK_{ij})'P_j+P_j(A_j-B_jK_{lj})\\&\quad=-(C'_{lj}Q_jC_{lj}+K'_{lj}R_jK_{lj})\end{aligned}\quad(64)$$

The second method of Liapunov states that, if $A-BK$ is a stable matrix, there exists a positive-definite matrix P that satisfies the equation above. The performance index can then be evaluated as

$$\begin{aligned}J_{lj}&=\int_0^\infty \underline{q}'_{lj}(t)[C'_{lj}Q_jC_{lj}+K'_{lj}R_jK_{lj}]\underline{q}_{lj}(t)\,dt\\&=-[\underline{q}'_{lj}(t)P_j\underline{q}_{lj}(t)]_0^\infty\\&=-\underline{q}'_{lj}(\infty)P_j\underline{q}_{lj}(\infty)+\underline{q}'_{lj}(0)P_j\underline{q}_{lj}(0)\end{aligned}\quad(65)$$

Since all the poles of A_j have negative real parts (see section 4.1), $\underline{q}_{lj}(\infty)\to 0$,

$$J_{lj}=\underline{q}'_{lj}(0)P_j\underline{q}_{lj}(0);\quad j=1,2.\quad(66)$$

Since R_j is assumed to be a positive-definite matrix, it can be written as follows:

$$R_j=T'_jT_j;\quad j=1,2.\quad(67)$$

where T_j is a non-singular matrix. Then equation (64) can be written as

$$\begin{aligned}&(A'_j-K'_{lj}B'_j)P_j+P_j(A_j-B_jK_{lj})+C'_{lj}Q_jC_{lj}\\&\qquad+K'_{lj}T'_jT_jK_{lj}=0\end{aligned}\quad(68)$$

which can be rewritten as (by definition, $P'_j=P_j$)

$$\begin{aligned}&A'_jP_j+P_jA_j+[T_jK_{lj}-(T'_j)^{-1}B'_jP_j]'[T_jK_{lj}-(T'_j)^{-1}B'_jP_j]\\&\qquad-P_jB_jR_j^{-1}B'_jP_j+C'_{lj}Q_jC_{lj}=0\end{aligned}\quad(69)$$

The minimisation of J_{lj} with respect to K_{lj} requires the minimisation of

$$\underline{q}'_{lj}[T_jK_{lj}-(T'_j)^{-1}B'_jP_l]'[T_jK_{lj}-(T'_j)^{-1}B'_jP_j]\underline{q}_{lj}\quad(70)$$

with respect to K_{lj}. Since the last expression is nonnegative (a quantity squared), the minimum occurs when it is zero, or when:

$$T_jK_{lj}=(T'_j)^{-1}B'_jP_j;\quad j=1,2\quad(71)$$

Hence:

$$K_{lj}=T_j^{-1}(T'_j)^{-1}B'_jP_j=R_j^{-1}B'_jP_j;\quad j=1,2.\quad(72)$$

This last equation gives the optimal matrix K_{lj}. Thus, the optimal control law to the quadratic optimal control problem where the performance index is given by equation (60) becomes

$$\tau_j(t)=-K_{lj}\underline{q}_{lj}(t)=-R_j^{-1}B'_jP_j\underline{q}_{lj}(t)\quad j=1,2.\quad(73)$$

The matrix P_j must satisfy equation (68) or the following reduced-matrix Riccati equaiton:

$$\begin{aligned}&A'_jP+PA-PBR^{-1}B'P+C'QC\\&\qquad=A'_jP_j+P_jA_j-P_jB_jR_j^{-1}B'_jP_j\\&\qquad\quad+C'_{lj}Q_jC_{lj}=0\quad j=1,2.\end{aligned}\quad(74)$$

If equation (74) is solved for the matrix P_j, and P_j is then substituted into equation (72), the resulting matrix K_{lj} is the optimal feed-back gain. Since the output to the two flexible subsystems is the deflection at the respective end-tips, Q_j is a scalar used to weight the output $u_{lj}(t)$:

$$Q_j=\mu_j;\quad j=1,2.\quad(75)$$

and since the input is also a scalar (τ_j), R_j becomes:

$$R_j=\eta_j;\quad j=1,2.\quad(76)$$

This results in $K_j(x)$ being a vector of the following form:

$$K_j(x)=[k_{j1}\quad k_{j2}\quad k_{j3}\quad k_{j4}\quad k_{j5}\quad k_{j6}];\quad j=1,2.\quad(77)$$

Replacing Q_j and R_j in equation (74) yields:

$$A'_jP_j+P_jA_j-P_jB_j\eta_j^{-1}B'_jP_j+C'_{lj}\mu_jC_{lj}=0;\quad j=1,2.\quad(78)$$

Choosing $\mu_1=\mu_2=2$, and $\eta_1=\eta_2=\sqrt{2}$ will result in the matrices P_1 and P_2 [from equation (78)] being equal to:

$$P_1=\begin{bmatrix}6.90&0.14&-0.05&-0.01&0&0\\0.14&0.06&0.01&0&0&0\\-0.05&0.01&1.39&0.01&0&0\\-0.01&0&0.01&0.01&0&0\\0&0&0&0&10^{-3}&0\\0&0&0&0&0&2\times10^{-5}\end{bmatrix};$$

$$P_2=\begin{bmatrix}3.93&0.14&-0.10&-0.01&0&0\\0.14&0.03&0.01&0&0&0\\-0.10&0.01&0.72&2.8\times10^{-3}&0&0\\-0.01&0&2.8\times10^{-3}&2\times10^{-3}&0&0\\0&0&0&0&10^{-4}&0\\0&0&0&0&0&10^{-5}\end{bmatrix}$$

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

Substituting these matrices into equation (72) for $R_j = \eta_j = \sqrt{2}$ gives the vectors K_{l1} and K_{l2} of the optimal feedback gains:

$$K_{l1} = [-4.85 \quad 1.99 \quad 0.27 \quad 2 \times 10^{-3} \quad 0 \quad 0];$$
$$K_{l2} = [-4.83 \quad 1.05 \quad 0.25 \quad 2 \times 10^{-3} \quad 0 \quad 0] \tag{79}$$

Thus, the closed-loop state matrices $A_1 - B_1 K_{l1}$ and $A_2 - B_2 K_{l2}$ have the following poles for the first link,

$$s_{11,12} = -1.29 \pm 10.24i \quad s_{13,14} = -2.90 \pm 52.65i$$
$$s_{15,16} = -7.32 \pm 124.33i$$

and for the second link,

$$s_{21,22} = -2.48 \pm 10.06i \quad s_{23,24} = -5.83 \pm 52.42i$$
$$s_{25,26} = -7.32 \pm 124.40i$$

The first two pairs of poles for each link had their real part shifted further down the real axis (in comparison to the open-loop poles presented in section 4.1). The optimal control did not alter the third pole for each link since its contribution in the final response is negligible.[1] The imaginary parts of all the poles of the closed-loop transfer function were not affected by the optimal feed-back gains, the aim of any control law being mainly to stabilise the system by shifting the real part of the appropriate poles into the negative half-plane. The outputs $u_{l1}(t)$ and $u_{l2}(t)$ can now be obtained according to the following equation:

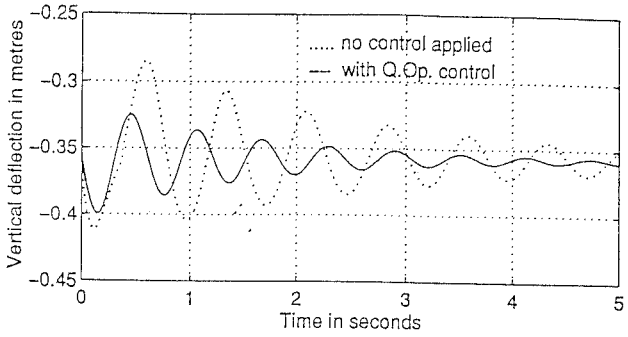
$$u_{lj}(t) = C_{lj} \exp\{(A_j - B_j K_{lj})t\}\underline{q_{lj}}(0) + u_{0lj} \quad j = 1, 2. \tag{80}$$

$\underline{q_{lj}}(0)$ being the vector of initial conditions at the end-tip for the state variables of each link, u_{0lj} are the static deflections at the end-tip of each link. This last equation could be rewritten using the closed-loop poles as:

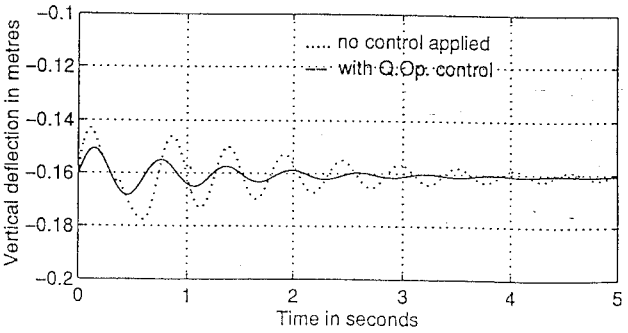
$$u_{lj}(t) = C_{lj} L^{-1}\{[s_{ji} I - A_j + B_j K_{lj}]^{-1}\}\underline{q_{lj}}(0) + u_{0lj}$$
$$i = 1, 6; j = 1, 2. \tag{81}$$

the state number is indicated by the subscript i, the link number is indicated by the subscript j. L^{-1} relates to the inverse Laplace transform.

Figure 5 shows the time history of the end-tip deflection of the first and second link, respectively, with and without optimal control. It can be seen that the quadratic optimal control allows the links to vibrate freely as soon as the rotation is started. The manoeuver is operated very smoothly and as soon as the torques reach a certain magnitude (determined by the controller) they are automatically decreased (or increased if the link is in a braking phase). This effect of damping, caused by the torques, allows the suppression of the forced vibrations, and the controlled links are then subject to a free vibration. Therefore, the settling time is decreased enormously. The desired torques $\tau_1(t)$ and $\tau_2(t)$ are obtained by using equation (73) for both flexible links. They were then used to calculate the angles of rotation $\theta_1(t)$ and $\theta_2(t)$ of the rigid mode of both links from eqn (50), by employing the Matlab© package and applying a Runge–Kutta method using 4*th* and 5*th* order functions.



(a) Link 1



(b) Link 2

Fig. 5. End-tip deflection with and without quadratic optimal control.

5. CONCLUSIONS

The paper initially explained the necessity for having an accurate static and dynamic model of the two-flexible-link manipulator, which properly represents the coupling and interactions between the links. Such a model has been developed in previous work and the main steps involved in the construction of this model have been reviewed.

The functions to be fulfilled by the controller of a flexible manipulator system were outlined in section 1. Firstly, it must compensate for the static deflection of the flexible links under gravity forces and, secondly, it must act to reduce both the magnitude and time duration of link oscillations which arise naturally out of its flexibility. The paper has investigated the ability of two alternative controllers, a computed-torque controller and a quadratic optimal controller, to fulfill these functions. Simulation of their relative performance has shown that, whilst computed-torque control (which is essentially an open-loop method) can fulfill these functions to a limited extent; much better performance is obtained by the closed-loop, quadratic optimal controller.

References

1. A.S. Morris and A. Madani, "Static and Dynamic Modelling of a Two-Flexible-Link Robot Manipulator", *Robotica* **14**(3), 289–300 (1995).
2. R.H. Cannon Jr. and E. Schmitz, 1984, "Initial

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press

Experiments on the End-Point Control of a Flexible One-Link Robot", *Int. J. Rob. Res.* **3**(3), 62–75 (1984).
3. S. Timoshenko and J.M. Lessels, *Applied Elasticity* (First edition, Westinghouse Technical Night School Press, 1925).
4. F.S. Tse, I.E. Morse and R.T. Hinkle, *Mechanical Vibrations, Theory and Application* (Second ed., Allyn and Bacon, Boston, 1978).
5. W.J. Book, O. Maizza-Netto and D.E. Whitney, "Feedback Control of Two Joint Systems with Distributed Flexibility", *ASME J. of Dyn. Sys. Meas. and Cont.* **97,** 424–431 (1975).
6. R.H. Cannon and E. Schmitz, "Initial Experiments on the Control of a Flexible Manipulator", *Int. J. of Rob. Res.* **3,** 62–75 (1984).
7. T. Fukuda, "Flexibility Control of Elastic Robot Arms", *J. of Rob. Sys.* **2**(1), 73–88 (1985).
8. P.H. Meckl and W.P. Steering, "Reducing Residual Vibration in Systems with Uncertain Resonances", *IEEE Cont. Sys. Mag.* 73–76 (April, 1988).
9. M. Hyde and W. Seering, "Inhibiting Multiple Mode Vibration in Controlled Flexible Systems", *Proc 1991 Amer. Cont. Conf.,* Boston, MA (1991) pp. 2449–2454.
10. J.Y.S. Luh, M.W. Walker and R.P. Paul, "Resolved Motion Force Control of Robot Manipulators", *ASME J. of Dyn. Sys. Meas. and Cont.* **102,** 126–133 (1982).
11. H. Asada, Z. Ma and H. Tokumaru, "Inverse Dynamics of Flexible Robot Arms: Modeling and Computation for Trajectory Control", *ASME J. of Dyn. Sys. Meas. and Cont.* **112,** 177–185 (1990).
12. H. Asada and J.J.E. Stoline, *Robot Analysis and Control* (Wiley, New York, 1986).
13. M.J. Balas, "Feedback Control of Flexible Systems", *IEEE Trans.* **AC-23**(4), 673–679 (1978).
14. C.M. Oakley and R.H. Cannon, "Initial Experiments on the End-Point control of a Two-Link Manipulator with a Very Flexible Forearm", *Proc. Amer. Cont. Conf.* Atlanta, GA (1988) pp. 996–1002.
15. G.R. Eisler, R.D. Robinett, D.J. Segalman and J.D. Feddema, "Approximate Optimal Trajectories for Flexible-Link Manipulator Slewing Using Recursive Quadratic Programming", *ASME J. of Dyn. Sys. Meas. and Cont.* **115,** 405–410 (1993).

https://doi.org/10.1017/S0263574798000186 Published online by Cambridge University Press