
Chapter 11

Force and position control of flexible manipulators

B. Siciliano and L. Villani

While several control schemes have been proposed for force and position control of rigid robot manipulators, only a few related to flexible manipulators have been published so far. In this chapter the main force and position control strategies for flexible manipulators are surveyed and two different approaches are illustrated in depth. One achieves force and position regulation in an indirect way, by computing the joint and deflection variables in the presence of an external contact via a suitable closed-loop inverse kinematics scheme. The other exploits singular perturbation techniques to design force and position control schemes similar to those adopted for rigid robot manipulators, with an additional control action used to stabilise the fast dynamics related to link flexibility. A planar two-link flexible manipulator in contact with a compliant surface is considered, and simulation studies demonstrating the performance of the control techniques are presented and discussed.

11.1 Introduction

In a wide number of applications, such as polishing, debarring, machining or assembling, it is necessary to control the interaction between the robot manipulator and the environment. During the interaction, the environment sets constraints on the geometric paths that can be followed by the end-effector. In such a case, purely motion control strategies for controlling the interaction will fail. In fact, any motion planning or position tracking error may give rise to a contact force that, if not controlled, may produce an unstable behaviour and will inflict damage either to the robot or to the environment. The higher the environment stiffness and position control accuracy are, the easier such a situation can occur.

On the other hand, the intrinsic compliance of a flexible-link manipulator may contribute to reduction in the value of the forces that can be generated when the interaction task is executed by a rigid robot. This means that by using flexible robots to perform interaction tasks, some benefits may be gained (Sur and Murray, 1997), even though the distributed flexibility of the links makes the interaction control problem more complex than for rigid robots.

The most common solution to interaction control is to use a force/torque sensor, mounted between the last link and the end-effector, to provide force measurements that can be used to achieve force control.

Robot force control has attracted a wide number of researchers in the past two decades. Several control schemes have been proposed, and a state of the art on force control of rigid robot manipulators can be found in Gorinevski *et al.* (1997) and Siciliano and Villani (1999). For the specific case of flexible manipulators, however, only few papers have been published.

Interaction control strategies can be grouped into two categories: those performing indirect force controls and those performing direct force controls. The main difference between the two categories is that the former achieves force control via motion control, without explicit closure of a force feedback loop, and the latter offers the possibility of controlling the contact force to a desired value with the closure of a force feedback loop.

The first category includes compliance control (Salisbury, 1980) and impedance control (Hogan, 1985), where the end-effector is made compliant by relating the position error to the contact force through a static or dynamic relationship of adjustable parameters. If a detailed model of the environment is available then a widely adopted strategy, which belongs to the second category, known as the hybrid position/force control may be used. This strategy aims to control position along the unconstrained task directions and force along the constrained task directions. A selection matrix acting on both desired and feedback quantities serves this purpose for typically planar contact surfaces (Raibert and Craig, 1981), whereas the explicit constraint equations have to be taken into account for general curved contact surfaces (McClamroch and Wang, 1988; Yoshikawa, 1987).

In most practical situations, a detailed model of the environment is not available. In such a case, an effective strategy, still in the second category, that may be adopted is the inner/outer motion/force control where an outer force control loop is closed around the inner motion control loop, which is typically available in a robot manipulator (De Schutter and Van Brussel, 1988). In order to embed the possibility of controlling motion along the unconstrained task directions, the desired motion of the end-effector can be input to the inner loop of an inner/outer motion/force control scheme. The resulting parallel control is composed of a force control action and a motion control action, where the former is designed so as to dominate the latter in order to ensure force control along the constrained task directions (Chiaverini and Sciavicco, 1993).

For the case of flexible manipulators, early works addressing the stability problems of force control are reported in Chiou and Shahinpoor (1990) and Mills (1992). Hybrid force/position control is adopted in Matsuno and Yamamoto (1994), Matsuno *et al.* (1994) and Rocco and Book (1996), and is used in Hu and Ulsoy (1994) and

Yang *et al.* (1995) to design robust and adaptive control strategies, respectively. The problem of controlling the interaction of flexible macro-manipulators carrying a rigid micro-link is considered in Lew and Book (1993) and Yoshikawa *et al.* (1996) in the framework of hybrid control as well. Force and position control strategies conceived to manage the interaction with more or less compliant environments, without requiring a detailed model, are proposed in Siciliano and Villani, (1999, 2001).

The inherent difficulty of force control of flexible manipulators is due to problems similar to those arising in motion control (Book, 1993; Canudas de Wit *et al.*, 1996). Moreover, the kinematics and the dynamics of the robot cannot be stated independent of the forces acting on the robot tip (end-point).

In fact, when a robot interacts with the environment, the additional deflections caused by contact forces must be suitably taken into account for the computation of inverse kinematic solution. This can be done by adding a corrective term to the Jacobian, as in the solution based on the closed-loop inverse kinematics (CLIK) algorithm developed in Siciliano (1999) for the case of contact with an infinitely stiff environment and in Siciliano (1998) for the case of a compliant environment.

As for the dynamics of a flexible manipulator in contact with the environment, Matsuno and Yamamoto (1994) proposed a model derived using the Hamilton's principle, where the boundary condition of the link in contact with the environment depends on the contact force and input torque, as well as on the contact position. This makes the flexible manipulator equation very difficult to solve and simplification must be made. For example, in Matsuno *et al.* (1994), the boundary conditions are simplified by considering a quasi-static model derived on the basis of the static relationship between the elastic deformations and the contact force.

On the other hand, if the assumed mode technique is adopted to model the flexible manipulator, the mode functions must satisfy the geometric boundary conditions, which are not altered by the contact with the environment, while the natural boundary conditions (i.e. those involving the balance of forces and moments at the ends of the links) are automatically taken into account by the Lagrange formulation of the mathematical model (Book, 1984; Meirovitch, 1967; Rocco and Book, 1996). This modelling approach is also pursued in Hu and Ulsoy (1994), Kim *et al.* (1996), Lin (2003), Mills (1992), Siciliano and Villani (1999) and Yang *et al.* (1995).

Another difficulty in controlling flexible robots is the problem of damping the vibrations that are naturally excited during the task execution. An effective approach is based on singular perturbation theory (Kokotovic *et al.*, 1986). When the link stiffness is large, a two-timescale model of the flexible manipulator can be derived (Siciliano and Book, 1988) consisting of a slow subsystem corresponding to the rigid-body motion and a fast subsystem describing the flexible motion. A composite control strategy can then be applied, based on a slow control designed for the equivalent rigid manipulator and a fast control, which stabilises the fast subsystem. Further developments of perturbation techniques can be found in Fraser and Daniel (1991), Moallem *et al.* (1997), Siciliano *et al.* (1992) and Vandegrift *et al.* (1994) for the case of flexible manipulators moving in free space, and in Matsuno and Yamamoto (1994), Rocco and Book (1996), Siciliano and Villani (2000) and Yang *et al.* (1995) for the case of contact with the environment.

The focus of this chapter is on force control strategies for flexible manipulators, which are conceived to manage the interaction with a more or less compliant environment without requiring an accurate model thereof. Two different approaches are presented.

The first approach, based on Siciliano and Villani (2001) achieves force and position regulation in an indirect way as long as the arm kinematic model, the mass distribution and stiffness of the links as well as the environment stiffness and position are known. In detail, assuming a simple elastic model for the contact surface, a position set-point is assigned, corresponding to the desired force applied to the desired point on the surface. Then a closed-loop inverse kinematics algorithm based on a Jacobian transpose scheme described in (Siciliano, 1998) is adopted to compute the joint and deflection variables. These are input to a simple proportional, derivative (PD) joint regulator (De Luca and Siciliano, 1993).

The second approach, based on Siciliano and Villani (2000), exploits singular perturbation techniques to design interaction control schemes in the framework of parallel force and position control with an additional control action used to stabilise the fast dynamics related to link flexibility.

Simulation results are presented for a two-link planar manipulator under gravity in contact with an elastically compliant surface.

The chapter is organized as follows. In Section 11.2 the model of a planar n -link flexible manipulator in contact with an elastic environment is presented. The two-stage algorithm achieving indirect force and position control is presented in Section 11.3. In Section 11.4 a singular perturbed model for the flexible manipulator is developed and two different parallel control schemes are considered for the slow dynamics: the first ensures force and position regulation and the second guarantees force regulation and position tracking. Section 11.5 provides concluding remarks.

11.2 Modelling

For the purpose of this chapter, planar n -link flexible manipulators with revolute joints are considered. The links are subject to bending deformation in the plane of motion only, that is, torsional effects are neglected. A sketch of a two-link arm, with coordinate frame assignment, is shown in Figure 11.1. The rigid motion is described by the joint-angle θ_i , while $w_i(x_i)$ denotes the transversal deflection of link i at x_i , $0 \leq x_i \leq L_i$, with L_i as the length of the link.

Let ${}^i\mathbf{p}_i(x_i) = \begin{bmatrix} x_i & w_i(x_i) \end{bmatrix}^T$ be the position of a point along the deflected link i with respect to frame (X_i, Y_i) and $\mathbf{p}_i$ the position of the same point in the base frame. Also let ${}^i\mathbf{r}_{i+1} = {}^i\mathbf{p}_i(L_i)$ be the position of the origin of frame (X_{i+1}, Y_{i+1}) with respect to frame (X_i, Y_i) , and $\mathbf{r}_{i+1}$ its position in the base frame.

The joint (rigid) rotation matrix $\mathbf{R}_i$ and the rotation matrix $\mathbf{E}_i$ of the (flexible) link at the end-point are, respectively,

$$\mathbf{R}_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{bmatrix}$$

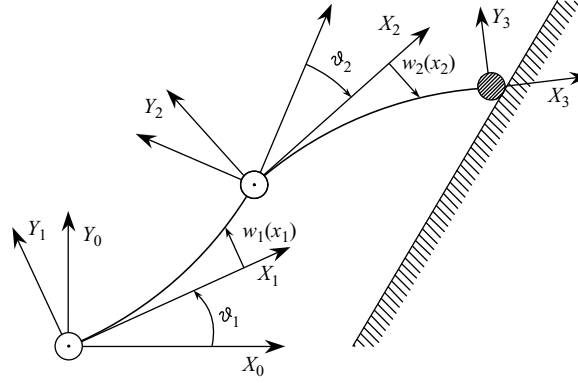


Figure 11.1 Planar two-link flexible manipulator

and

$$\mathbf{R}_i = \begin{bmatrix} 1 & -w'_{ie} \\ w'_{ie} & 1 \end{bmatrix}$$

where $w'_{ie} = (\partial w_i / \partial x_i) |_{x_i=L_i}$, and the small deflection approximation $\tan^{-1}(w'_{ie}) \cong w'_{ie}$ has been made. Hence the above absolute position vectors can be expressed as

$$\mathbf{p}_i = \mathbf{r}_i + \mathbf{W}_i^i \mathbf{p}_i$$

and

$$\mathbf{r}_{i+1} = \mathbf{r}_i + \mathbf{W}_i^i \mathbf{r}_{i+1}$$

where $\mathbf{W}_i$ is the global transformation matrix from the base frame to the frame (X_i, Y_i) given by the recursive equation,

$$\mathbf{W}_i = \mathbf{W}_{i-1} \mathbf{E}_{i-1} \mathbf{R}_i = \hat{\mathbf{W}}_{i-1} \mathbf{R}_i$$

with

$$\hat{\mathbf{W}}_0 = \mathbf{I}$$

On the basis of the above relations, the kinematics of any point along the manipulator is completely specified as a function of joint angles and link deflections.

A finite-dimensional model (of order m_i) of link flexibility can be obtained by the assumed modes technique. By exploiting the separability in time and space of solutions to the Euler–Bernoulli equation for flexible beams;

$$(EI)_i \frac{\partial^4 w_i(x_i, t)}{\partial x_i^4} + \frac{\partial^2 w_i(x_i, t)}{\partial t^2} = 0$$

for $i = 1, \dots, n$ where ρ_i is the uniform mass density and $(EI)_i$ is the constant flexural rigidity of link i , the link deflection can be expressed as

$$w_i(x_i, t) = \sum_{j=1}^{m_i} \phi_{ij}(x_i) \delta_{ij}(t) \quad (11.1)$$

where $\delta_{ij}(t)$ are the time-varying variables associated with the assumed spatial mode shapes $\phi_{ij}(x_i)$ of link i . The mode shapes have to satisfy proper boundary conditions at the base (clamped) and at the end of each link (mass).

In view of equation (11.1), a direct kinematics equation can be derived expressing the position $\mathbf{p}$ of the manipulator end-point as a function of the $(n \times 1)$ joint variable vector $\boldsymbol{\theta}$ and the $(m \times 1)$ deflection variable vector $\boldsymbol{\delta}$ that is,

$$\mathbf{p} = \mathbf{k}(\boldsymbol{\theta}, \boldsymbol{\delta}) \quad (11.2)$$

where $m = \sum_{i=1}^n m_i$. For later use in the inverse kinematics scheme, the differential kinematics is also needed. The absolute linear velocity of a point on the arm is

$$\dot{\mathbf{p}}_i = \mathbf{P}_i + \dot{\mathbf{W}}_i^i \mathbf{p}_i + \mathbf{W}_i^i \dot{\mathbf{p}}_i \quad (11.3)$$

with ${}^i\dot{\mathbf{r}}_{i+1} = {}^i\dot{\mathbf{p}}_i(L_i)$. Since the links are assumed to be inextensible ($\dot{x}_i = 0$), then ${}^i\dot{\mathbf{p}}_i(x_i) = [0 \quad \dot{w}_i(x_i)]^T$. The computation of equation (11.3) takes advantage of the recursion

$$\dot{\mathbf{W}}_i = \dot{\mathbf{W}}_{i-1} \mathbf{R}_i + \hat{\mathbf{W}}_{i-1}^i \dot{\mathbf{R}}_i \quad (11.4)$$

with

$$\dot{\mathbf{W}}_i = \dot{\mathbf{W}}_i \mathbf{E}_i + \mathbf{W}_i^i \dot{\mathbf{E}}_i \quad (11.5)$$

Also, note that

$$\dot{\mathbf{R}}_i = \mathbf{S} \mathbf{R}_i \dot{\theta}_i, \quad \dot{\mathbf{E}}_i = \mathbf{S} \dot{w}'_{ie} \quad (11.6)$$

with

$$\mathbf{S} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (11.7)$$

In view of equations (11.2)–(11.7), it is not difficult to show that the differential kinematics equation expressing the end-point velocity $\dot{\mathbf{p}}$ as a function of $\dot{\boldsymbol{\theta}}$ and $\dot{\boldsymbol{\delta}}$ can be written in the form:

$$\dot{\mathbf{p}} = \mathbf{J}_\theta(\boldsymbol{\theta}, \boldsymbol{\delta}) \dot{\boldsymbol{\theta}} + \mathbf{J}_\delta(\boldsymbol{\theta}, \boldsymbol{\delta}) \dot{\boldsymbol{\delta}} \quad (11.8)$$

where $\mathbf{J}_\theta = \partial \mathbf{k} / \partial \boldsymbol{\theta}$ and $\mathbf{J}_\delta = \partial \mathbf{k} / \partial \boldsymbol{\delta}$.

Assume that the manipulator is in contact with the environment. By virtue of the virtual work principle, the vector $\mathbf{f}$ of the forces exerted by the manipulator on the environment performing work on $\mathbf{p}$ has to be related to the $(n \times 1)$ vector $\mathbf{J}_\theta^T \mathbf{f}$ of joint torques performing work on $\boldsymbol{\theta}$ and the $(m \times 1)$ vector $\mathbf{J}_\delta^T \mathbf{f}$ of the elastic reaction forces performing work on $\boldsymbol{\delta}$.

A finite-dimensional Lagrangian dynamic model of the planar manipulator in contact with the environment can be obtained in terms of $\boldsymbol{\theta}$ and $\boldsymbol{\delta}$ in the form (De Luca and Siciliano, 1991):

$$\mathbf{M}_{\theta\theta}(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\theta}} + \mathbf{M}_{\theta\delta}(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\delta}} + \mathbf{c}_{\theta}(\boldsymbol{\theta}, \boldsymbol{\delta}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{\delta}}) + \mathbf{g}_{\theta}(\boldsymbol{\theta}, \boldsymbol{\delta}) = -\mathbf{J}_{\theta}^T(\boldsymbol{\theta}, \boldsymbol{\delta}) \mathbf{f} \quad (11.9)$$

$$\mathbf{M}_{\theta\delta}^T(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\theta}} + \mathbf{M}_{\delta\delta}(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\delta}} + \mathbf{c}_{\delta}(\boldsymbol{\theta}, \boldsymbol{\delta}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{\delta}}) + \mathbf{g}_{\delta}(\boldsymbol{\theta}, \boldsymbol{\delta}) + \mathbf{D}\dot{\boldsymbol{\delta}} + \mathbf{K}\boldsymbol{\delta} = -\mathbf{J}_{\delta}^T(\boldsymbol{\theta}, \boldsymbol{\delta}) \mathbf{f} \quad (11.10)$$

where $\mathbf{M}_{\theta\theta}$, $\mathbf{M}_{\theta\delta}$ and $\mathbf{M}_{\delta\delta}$ are the matrix blocks of the positive definite symmetric inertia matrix, $\mathbf{c}_{\theta}$, $\mathbf{c}_{\delta}$ are the vectors of Coriolis and centrifugal forces, $\mathbf{g}_{\theta}$, $\mathbf{g}_{\delta}$ are the vector of gravitational forces, $\mathbf{K}$ is the diagonal and positive definite link stiffness matrix, $\mathbf{D}$ is the diagonal and positive semi definite link damping matrix, and $\boldsymbol{\tau}$ is the vector of the input joint torques.

To analyse the performance of the position and force control algorithms, a model of the contact force is required. A real contact is a naturally distributed phenomenon in which the local characteristics of both the end-effector and the environment are involved. Moreover, friction effects between parts typically exist, which greatly complicate the nature of the contact itself. A simplified analysis can be pursued by considering a frictionless and planar surface, which is locally a good approximation to surfaces of regular curvature, and considering an elastic force given by

$$\mathbf{f} = k_e \mathbf{n} \mathbf{n}^T (\mathbf{p} - \mathbf{p}_e) = k_e \mathbf{n} \mathbf{n}^T (\mathbf{k}(\boldsymbol{\theta}, \boldsymbol{\delta}) - \mathbf{p}_e) \quad (11.11)$$

where k_e is the surface stiffness, $\mathbf{p}_e$ is the un-deformed (constant) position of the surface, $\mathbf{n}$ is the (constant) unit vector of the direction normal to the surface, and the direct kinematics equation (11.2) has been used to express the position of the contact point in terms of joint and deflection variables. Also, it is assumed that contact is not lost.

Notice that for the derivation of the dynamic model in equations (11.9) and (11.10), the presence of the contact with the environment does not affect the choice of the mode shapes $\phi_{ij}(x_i)$ in equation (11.1) and the force enters into the equations of motion through the Jacobian (Rocco and Book, 1996).

Also, in this chapter only the interaction with a more or less compliant environment described by relations of the form in equation (11.11) is considered. On the other hand, in case of interaction with an infinitely rigid environment, kinematic constraints are imposed on the coordinates of robot end-point and a constraint force must be considered in the dynamic model of the flexible manipulator, expressed in terms of Lagrange multipliers (McClamroch and Wang, 1988). As in the case of rigid robots, the presence of the constraints reduces the number of degrees of freedom of the system. Moreover, it is possible to reduce the number of differential equations by resorting to a coordinate partitioning procedure (Hu and Ulsoy, 1994; Matsuno and Yamamoto, 1994; Matsuno *et al.*, 1994; Mills, 1992; Rocco and Book, 1996; Yang *et al.*, 1995).

11.3 Indirect force and position regulation

The interaction of a flexible-link manipulator with a compliant environment can be managed by controlling both the contact force and the end-point position. In view of the model of the contact force in equation (11.11), the control objective can be specified in terms of a desired force $f_d \mathbf{n}$ aligned with $\mathbf{n}$ and a desired position $\mathbf{p}_d$ on the contact plane. Nevertheless, the quantities f_d and $\mathbf{p}_d$ cannot be assigned independently, since they have to be consistent with the model in equation (11.11). In other words, the desired value of the force f_d can be achieved only if the component normal to the plane of the desired position $\mathbf{p}_d$ is chosen as

$$p_{dn} = \mathbf{n}^T \mathbf{p}_d = k_e^{-1} f_d + p_{en} \quad (11.12)$$

Hence, force control can be realised indirectly via position control, provided that the surface stiffness k_e and the component p_{en} of the un-deformed position of the surface are known.

In this section, a force and position regulator is presented, which achieves a desired position on the contact plane as well as a desired force, provided that equation (11.12) is satisfied, without requiring direct measurement of the contact force.

The controller is based on a two-stage algorithm. The first stage is in charge of solving the inverse kinematics problem to compute the desired vectors of joint variables $\boldsymbol{\theta}_d$ and deflection variables $\boldsymbol{\delta}_d$ that place the end-point of the flexible arm at a desired position $\mathbf{p}_d$; the component p_{dn} of $\mathbf{p}_d$ is chosen according to equation (11.12) to achieve a desired force f_d , while the components of the desired position tangential to the plane can be freely chosen. In the second stage, which constitutes a joint regulator, the variables $\boldsymbol{\theta}_d$ and $\boldsymbol{\delta}_d$ are used as set-points.

11.3.1 First stage

The first stage of the algorithm computes the inverse kinematics solution. To derive a Jacobian-based inverse kinematics scheme, the differential kinematic equation accounting for link deflections caused by gravity and contact with the environment must be considered.

For the regulation problem, a static situation can be considered. By virtue of equation (11.10), in a static situation the deflections satisfy the equation

$$\mathbf{g}_\delta(\boldsymbol{\theta}, \boldsymbol{\delta}) + \mathbf{K}\boldsymbol{\delta} = -\mathbf{J}_\delta^T(\boldsymbol{\theta}, \boldsymbol{\delta}) \mathbf{f}$$

According to the small deflection approximation, it can be assumed that $\mathbf{g}_\delta$ is only a function of $\boldsymbol{\theta}$ (De Luca and Siciliano, 1993) and so is the case for $\mathbf{J}_\delta$ in equation (11.8) and $\mathbf{p}$ in equation (11.11). Hence, the deflection variables can be computed from equation (11.12) as

$$\boldsymbol{\delta} = -\mathbf{K}^{-1} (k_e \mathbf{j}_{\delta n}(\boldsymbol{\theta}) (p_n(\boldsymbol{\theta}) - p_{en}) + \mathbf{g}_\delta(\boldsymbol{\theta})) \quad (11.13)$$

where

$$\mathbf{j}_{\delta n}(\boldsymbol{\theta}) = \mathbf{J}^T \mathbf{n}, \quad p_n = \mathbf{n}^T \mathbf{p}, \quad p_{en} = \mathbf{n}^T \mathbf{p}_e$$

For later use in the inverse kinematics scheme, differentiating equation (11.13) with respect to time gives

$$\dot{\delta} = \mathbf{J}_{fg}(\boldsymbol{\theta}) \dot{\boldsymbol{\theta}} \quad (11.14)$$

where

$$\mathbf{J}_{fg} = -\mathbf{K}^{-1} (k_e \mathbf{J}_f(\boldsymbol{\theta}) + \mathbf{J}_g(\boldsymbol{\theta}))$$

with

$$\mathbf{J}_f = \frac{\partial \mathbf{j}_{\delta n}}{\partial \boldsymbol{\theta}} (p_n - p_{en}) + \mathbf{j}_{\delta n} \frac{\partial p_n}{\partial \boldsymbol{\theta}}$$

and $\mathbf{J}_g = \partial \mathbf{g}_\delta / \partial \boldsymbol{\theta}$. Substituting for $\dot{\delta}$ from equation (11.14) into equation (11.8) yields

$$\dot{\mathbf{p}} = \mathbf{J}_p(\boldsymbol{\theta}, \delta) \dot{\boldsymbol{\theta}} \quad (11.15)$$

where

$$\mathbf{J}_p = \mathbf{J}_\theta + \mathbf{J}_\delta \mathbf{J}_{fg}$$

is the overall Jacobian matrix relating joint velocity to end-point velocity. Notice that the Jacobian in equation (11.15) is obtained by modifying the rigid-body Jacobian $\mathbf{J}_\theta$ with two terms that account for the deflections induced by the contact force and gravity, respectively.

The attractive feature of the differential kinematics in equation (11.15) is its formal analogy with the differential kinematics equation for a rigid arm. Therefore, any Jacobian-based inverse kinematics scheme can be adopted in principle. In this respect, one of the most effective schemes is the CLIK scheme (Siciliano, 1990) that reformulates the inverse kinematics problem in terms of the convergence of a suitable closed-loop dynamic system.

According to the Jacobian transpose scheme, the joint variables vector is computed by integrating the joint velocity vector chosen as

$$\dot{\boldsymbol{\theta}} = \mathbf{J}_p^T(\boldsymbol{\theta}, \delta) \mathbf{K}_p (\mathbf{p}_d - \mathbf{p}) \quad (11.16)$$

Using a Lyapunov argument (Siciliano, 1990) it can be shown that, as long as the vector $\mathbf{K}_p (\mathbf{p}_d - \mathbf{p})$ in equation (11.16) is outside the null space of $\mathbf{J}_p^T$, the end-point position error $\mathbf{p}_d - \mathbf{p}$ asymptotically tends to zero. In fact, a suitable choice of the matrix $\mathbf{K}_p$ can be made to avoid that the scheme gets stuck with $\mathbf{p}_d - \mathbf{p} \neq \mathbf{0}$ and $\dot{\boldsymbol{\theta}} = \mathbf{0}$. In summary, $\boldsymbol{\theta}$ and δ tend asymptotically to the constant values $\boldsymbol{\theta}_d$ and δ_d such that $\mathbf{p}_d = \mathbf{k}(\boldsymbol{\theta}_d, \delta_d)$.

Notice that one of the attractive features of this approach is that, similar to the rigid arm case, any Jacobian-based inverse kinematics scheme can be adopted in principle, as well as any joint-space control law. The solution chosen in this work for kinematic inversion does not require the inverse of the Jacobian and thus it works well in the neighbourhood of singularities.

11.3.2 Second stage

The second stage of the algorithm is in charge of regulating the joint and deflection variables to the values θ_d and δ_d computed in the first stage. To this aim, the simple PD regulator presented in (De Luca and Siciliano, 1993) can be adopted:

$$\tau = \mathbf{K}_1 (\theta_d - \theta) - \mathbf{K}_2 \dot{\theta} + \mathbf{g}_\theta (\theta_d, \delta_d) + \mathbf{J}_\theta^T (\theta_d, \delta_d) f_d \mathbf{n} \quad (11.17)$$

where $\mathbf{K}_1$ and $\mathbf{K}_2$ are suitable positive definite matrix gains. The feedforward terms $\mathbf{g}_\theta (\theta_d, \delta_d)$ and $\mathbf{J}_\theta^T (\theta_d, \delta_d) f_d \mathbf{n}$ are required to compensate for the gravity torque and contact force respectively, at steady state.

The control law in equation (11.17) ensures asymptotic convergence of θ and δ to the corresponding set-points. Hence, the two-stage control scheme in equations (11.16) and (11.17) guarantees that $\mathbf{p} \rightarrow \mathbf{p}_d$ and $\mathbf{f} \rightarrow f_d \mathbf{n}$ as $t \rightarrow \infty$.

Notice that the PD regulator ensures asymptotic stability only in the presence of significant damping. When passive damping is too low, active vibration damping can be achieved by using full state-feedback (Canudas de Wit *et al.*, 1996).

It is also worth noting that the scheme only makes use of joint position and velocity measurements. Obviously, any joint position control law for flexible arms may be used in the second stage of the scheme in lieu of the simple PD regulator in equation (11.17). In any case, the overall performance in terms of end-point position and force errors strongly depends on the accuracy of the static model of the flexible arm, as well as on the accuracy of the available estimates of the stiffness and position of the contact surface.

11.3.3 Simulation

To illustrate the performance of the two-stage algorithm, a planar two-link flexible manipulator is considered (see Fig. 11.1):

$$\theta = [\theta_1 \quad \theta_2]^T$$

and a payload of 0.1 kg is assumed to be placed at the end-point of the manipulator. An expansion with two clamped-mass assumed modes is taken for each link:

$$\delta = [\delta_{11} \quad \delta_{12} \quad \delta_{21} \quad \delta_{22}]^T$$

The parameters of the manipulator are given in Table 11.1.

The resulting natural frequencies of vibration are

$$\begin{aligned} f_{11} &= 1.40 \text{ Hz}, & f_{12} &= 5.10 \text{ Hz} \\ f_{21} &= 536.09 \text{ Hz}, & f_{22} &= 20792.09 \text{ Hz}. \end{aligned}$$

The stiffness matrix $\mathbf{K}$ is

$$\mathbf{K} = \begin{bmatrix} 38.79 & 513.37 \\ 536.09 & 20792.09 \end{bmatrix}$$

Table 11.1 Link parameters

Parameter (unit)	Link 1	Link 2
Density (kg/m)	1	1
Length (m)	0.5	0.5
Centre of mass (m)	0.25	0.25
Mass (kg)	0.5	0.5
Hub mass (kg)	1	1

The dynamic model of the manipulator and the missing numerical data can be found in (De Luca and Siciliano, 1991) while the direct and differential kinematics equations are reported in (Siciliano and Villani, 2001).

The contact surface is a vertical plane, thus the normal vector in equation (11.11) is $\mathbf{n} = [1 \ 0]^T$; a point of the un-deformed plane is

$$\mathbf{p}_e = [0.55 \ 0]^T \text{ m}$$

and the contact stiffness is $k_e = 50\text{N/m}$.

The feedback gain matrix $\mathbf{K}_p$ of the CLIK algorithm in equation (11.16) is chosen as

$$\mathbf{K}_p = \text{diag} \{ 500 \ 500 \}$$

and the inverse kinematics scheme is discretised at a sampling time $T_c = 1\text{ms}$, using the Euler integration rule. In particular, according to equation (11.16), the joint variables vector $\boldsymbol{\theta}_d$ is computed as

$$\boldsymbol{\theta}_d(t_{k+1}) = \boldsymbol{\theta}_d(t_k) + T_c \mathbf{J}_p^T(\boldsymbol{\theta}_d(t_k), \boldsymbol{\delta}_d(t_k)) \mathbf{K}_p (\mathbf{p}_d(t_k) - \mathbf{p}(t_k))$$

and, according to equation (11.13), the deflection variables vector $\boldsymbol{\delta}_d$ is computed as

$$\boldsymbol{\delta}_d(t_{k+1}) = -\mathbf{K}^{-1} (k_e \mathbf{j}_{\delta n}(\boldsymbol{\theta}_d(t_k)) (p_n(\boldsymbol{\theta}(t_k)) - p_{en}) + \mathbf{g}_{\delta}(\boldsymbol{\theta}_d(t_k)))$$

The feedback matrix gains in (11.17) are chosen as:

$$\mathbf{K}_1 = \text{diag} \{ 25 \ 25 \}, \quad \mathbf{K}_2 = \text{diag} \{ 3 \ 3 \}$$

Numerical simulations were performed using Matlab with Simulink. The arm was initially placed with the end-point in contact with the un-deformed plane at the position

$$\mathbf{p}_e = [0.55 \ -0.55]^T \text{ m}$$

with null contact force; the corresponding generalized coordinates of the manipulator, computed using the CLIK algorithm in equation (11.16), are

$$\boldsymbol{\theta} = [-1.396 \ 1.462]^T \text{ rad}$$

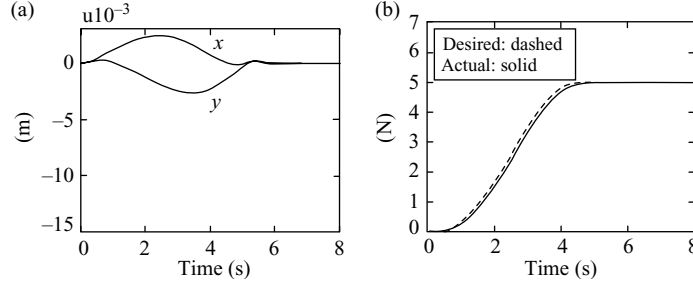


Figure 11.2 Time histories of the position error and of the contact force for the first example: (a) position error; (b) control force

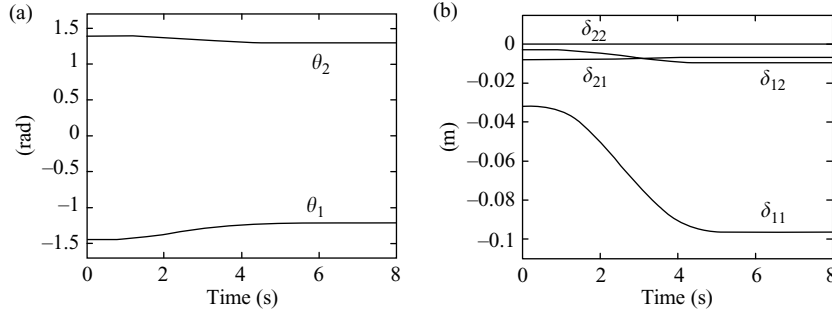


Figure 11.3 Time histories of the joint angles and of the link deflections for the first example: (a) joint angle; (b) link deflection

$$\delta = [-0.106 \quad 0.001 \quad -0.009 \quad -0.0001]^T \text{ m}$$

It is desired to reach the end-point position

$$\mathbf{p}_d = [0.55 \quad -0.50]^T \text{ m}$$

and a fifth-order polynomial trajectory with null initial and final velocity and acceleration is imposed from the initial to the final position with a duration of 5 s.

In the first example, it is assumed that the stiffness of the environment is known, hence the desired force corresponding to the desired position is

$$\mathbf{f}_d = [5 \quad 0]^T \text{ N}$$

The time histories of the position errors and of the actual and desired contact forces are shown in Figure 11.2, and the time histories of the joint angles and link deflections are shown in Figure 11.3. It can be seen that the tracking error along the trajectory is small, although the scheme was conceived as a regulator. Moreover, both the desired force and position are achieved at steady state. Note also that, because of gravity and contact force, the arm has to bend to reach the desired end-point position properly.

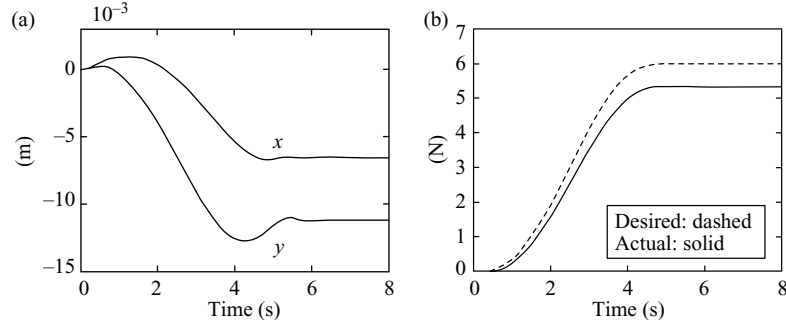


Figure 11.4 Time histories of the position error and of the contact force for the second example: (a) position error, (b) contact force

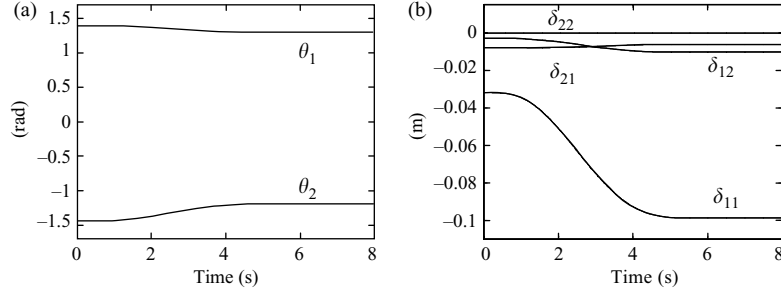


Figure 11.5 Time histories of the joint angles and of the link deflections for the second example: (a) joint angle, (b) link deflection

Actually the bending is much larger on the first link as expected (the links have the same parameters).

In the second numerical example, all the data are the same except for the estimated contact stiffness, which is assumed to be 60 N/m in lieu of the true value of 50 N/m. Hence, the desired force

$$\mathbf{f}_d = \begin{bmatrix} 6 & 0 \end{bmatrix}^T \text{ N}$$

is expected, with the same desired position.

The resulting time histories of the position errors and of the actual and desired contact forces are shown in Figure 11.4, and the time histories of the joint angles and link deflections are shown in Figure 11.5. It can be seen that the tracking error along the trajectory is limited, but a constant offset remains at steady state. Accordingly, the contact force reaches a constant value that is lower than the desired one, due to the fact that the contact stiffness was overestimated.

11.4 Direct force and position control

If a force sensor is available at the end-point of the manipulator, it is possible to achieve direct force control without requiring an exact estimate of the stiffness and of the position of the environment at rest. Moreover, if the dynamics related to link flexibility are suitably taken into account, tracking of a time-varying desired position can be achieved as well as regulation to a constant force.

In this section two different force and position control algorithms are presented, based on the parallel force and position approach of Chiaverini and Sciavicco (1993). The first scheme only requires partial knowledge of the model of the manipulator and guarantees force and position regulation. The second scheme achieves force regulation and position tracking by using more modelling information. Both schemes are part of a composite control strategy based on a two-time scale model of the flexible manipulator.

11.4.1 Composite control strategy

When stiffness of the link is large, it is reasonable to expect that the dynamics related to link flexibility are much faster than the dynamics associated with rigid motion of the robot so that the system naturally exhibits a two-timescale dynamic behaviour in terms of rigid and flexible variables. This feature can be conveniently exploited for control design (Matsuno and Yamamoto, 1994; Mills, 1992; Rocco and Book, 1996; Siciliano and Book, 1988; Yang *et al.*, 1995).

Following the approach proposed in (Siciliano and Book, 1988), the system can be decomposed into slow and fast subsystems by using singular perturbation theory; this leads to a composite control strategy for the full system based on separate control designs for the two reduced-order subsystems.

Assuming that full-state measurement is available and that a force sensor is mounted at the end-point of the manipulator, the joint torques can be conveniently chosen as

$$\boldsymbol{\tau} = \mathbf{g}_\theta(\boldsymbol{\theta}, \boldsymbol{\delta}) + \mathbf{J}_\theta^T(\boldsymbol{\theta}_d, \boldsymbol{\delta}_d) \dot{\mathbf{f}} + \mathbf{u} \quad (11.18)$$

to cancel out the effects of the static torques acting on the rigid part of the manipulator dynamics; the vector $\mathbf{u}$ is the new control input to be designed on the basis of the singular perturbation approach.

The timescale separation between the slow and fast dynamics can be determined by defining the singular perturbation parameter $\varepsilon = 1/\sqrt{k_m}$, where k_m is the smallest coefficient of the diagonal stiffness matrix $\mathbf{K}$, and the new variable

$$\mathbf{z} = \mathbf{K}\boldsymbol{\delta} = \frac{1}{\varepsilon^2} \hat{\mathbf{K}}\boldsymbol{\delta}$$

corresponds to the elastic force, where $\mathbf{K} = k_m \hat{\mathbf{K}}$. Considering the inverse $\mathbf{H}$ of the inertia matrix $\mathbf{M}$, the dynamic model in equations (11.9) and (11.10) with control law

in equation (11.18), can be rewritten in terms of the new variable $\mathbf{z}$ as

$$\begin{aligned} \ddot{\boldsymbol{\theta}} = & \mathbf{H}_{\theta\theta}^T(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \left(\mathbf{u} - \mathbf{c}_\theta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}, \dot{\boldsymbol{\theta}}, \varepsilon^2 \dot{\mathbf{z}}) \right) - \mathbf{H}_{\theta\delta}(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \left[\mathbf{c}_\delta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}, \dot{\boldsymbol{\theta}}, \varepsilon^2 \dot{\mathbf{z}}) \right. \\ & \left. + \mathbf{g}_\delta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) + \varepsilon^2 \mathbf{D} \hat{\mathbf{K}}^{-1} \dot{\mathbf{z}} + \mathbf{z} + \mathbf{J}_\delta^T(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \mathbf{f} \right] \end{aligned} \quad (11.19)$$

$$\begin{aligned} \varepsilon^2 \mathbf{z} = & \hat{\mathbf{K}} \mathbf{H}_{\theta\delta}^T(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \left(\mathbf{u} - \mathbf{c}_\theta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}, \dot{\boldsymbol{\theta}}, \varepsilon^2 \dot{\mathbf{z}}) \right) - \hat{\mathbf{K}} \mathbf{H}_{\delta\delta}(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \left[\mathbf{c}_\delta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}, \dot{\boldsymbol{\theta}}, \varepsilon^2 \dot{\mathbf{z}}) \right. \\ & \left. + \mathbf{g}_\delta(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) + \varepsilon^2 \mathbf{D} \hat{\mathbf{K}}^{-1} \dot{\mathbf{z}} + \mathbf{z} + \mathbf{J}_\delta^T(\boldsymbol{\theta}, \varepsilon^2 \mathbf{z}) \mathbf{f} \right] \end{aligned} \quad (11.20)$$

where a suitable partition of $\mathbf{H}$ has been considered:

$$\mathbf{H} = \mathbf{M}^{-1} = \begin{bmatrix} \mathbf{H}_{\theta\theta} & \mathbf{H}_{\theta\delta} \\ \mathbf{H}_{\theta\delta}^T & \mathbf{H}_{\delta\delta} \end{bmatrix}$$

Equations (11.19) and (11.20) represent a singularly perturbed form of the flexible manipulator model; when $\varepsilon \rightarrow \infty$, the model of an equivalent rigid manipulator is recovered. In fact, setting $\varepsilon = 0$ and solving for $\mathbf{z}$ in equation (11.20) gives

$$\mathbf{z}_s = \bar{\mathbf{H}}_{\delta\delta}^{-1}(\boldsymbol{\theta}_s) \bar{\mathbf{H}}_{\theta\delta}^T(\boldsymbol{\theta}_s) (\mathbf{u}_s - \bar{\mathbf{c}}_\theta(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s)) - \bar{\mathbf{c}}_\delta(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s) - \bar{\mathbf{g}}_\delta(\boldsymbol{\theta}_s) - \bar{\mathbf{J}}_\delta^T(\boldsymbol{\theta}_s) \mathbf{f}_s \quad (11.21)$$

where the subscript s indicates that the system is considered in the *slow* timescale and the bar denotes that a quantity is computed with $\varepsilon = 0$. Substituting equation (11.21) into equation (11.19) with $\varepsilon = 0$ yields

$$\ddot{\boldsymbol{\theta}}_s = \bar{\mathbf{M}}_{\theta\theta}^{-1}(\boldsymbol{\theta}_s) (\mathbf{u}_s - \bar{\mathbf{c}}_\theta(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s)) \quad (11.22)$$

where the equality:

$$\bar{\mathbf{M}}_{\theta\theta}^{-1}(\boldsymbol{\theta}_s) = \left(\bar{\mathbf{M}}_{\theta\theta}(\boldsymbol{\theta}_s) - \bar{\mathbf{M}}_{\theta\delta}(\boldsymbol{\theta}_s) \bar{\mathbf{M}}_{\delta\delta}^{-1}(\boldsymbol{\theta}_s) \bar{\mathbf{M}}_{\theta\delta}^T(\boldsymbol{\theta}_s) \right)$$

has been exploited, where $\bar{\mathbf{M}}_{\theta\theta}(\boldsymbol{\theta}_s)$ represents the inertia matrix of the equivalent rigid manipulator and $\bar{\mathbf{c}}_\theta(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s)$ the vector of the corresponding Coriolis and centrifugal torques.

The dynamics of the system in the *fast* timescale can be obtained by setting $t_f = t/\varepsilon$, treating the slow variables as constants in the fast timescale, and introducing the fast variables $\mathbf{z}_f = \mathbf{z} - \mathbf{z}_s$. Thus, the fast system in equation (11.20) is

$$\frac{d^2 \mathbf{z}_f}{dt_f^2} = -\hat{\mathbf{K}} \bar{\mathbf{H}}_{\delta\delta}(\boldsymbol{\theta}_s) \mathbf{z}_f + \hat{\mathbf{K}} \mathbf{H}_{\theta\delta}^T(\boldsymbol{\theta}_s) \mathbf{u}_f \quad (11.23)$$

where the fast control $\mathbf{u}_f = \mathbf{u} - \mathbf{u}_s$ has been introduced accordingly.

On the basis of the above two-timescale model, the design of a feedback controller for the system in equations (11.19) and (11.20) can be performed according to a composite control strategy, that is,

$$\mathbf{u} = \mathbf{u}_s(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s) + \mathbf{u}_f(\mathbf{z}_f, d\mathbf{z}_f/dt_f) \quad (11.24)$$

with the constraint that $\mathbf{u}_f(\mathbf{0}, \mathbf{0}) = \mathbf{0}$ so that $\mathbf{u}_f$ is inactive along the equilibrium manifold specified by equation (11.21).

To design the slow control for the rigid nonlinear system in equation (11.22), it is useful to derive the slow dynamics corresponding to the end-point position. Differentiating equation (11.8) gives the end-point acceleration;

$$\dot{\mathbf{p}} = \mathbf{J}_\theta(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\theta}} + \mathbf{J}_\delta(\boldsymbol{\theta}, \boldsymbol{\delta}) \ddot{\boldsymbol{\delta}} + \mathbf{h}(\boldsymbol{\theta}, \boldsymbol{\delta}, \dot{\boldsymbol{\theta}}, \dot{\boldsymbol{\delta}})$$

where $\mathbf{h} = \dot{\mathbf{J}}_\theta \dot{\boldsymbol{\theta}} + \dot{\mathbf{J}}_\delta \dot{\boldsymbol{\delta}}$. Hence, the corresponding slow system is

$$\dot{\mathbf{p}}_s = \bar{\mathbf{J}}_\theta(\boldsymbol{\theta}_s) \bar{\mathbf{M}}_{\theta\theta}^{-1}(\boldsymbol{\theta}_s) (\mathbf{u}_s - \bar{\mathbf{c}}_\theta(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s)) + \bar{\mathbf{h}}(\boldsymbol{\theta}_s, \dot{\boldsymbol{\theta}}_s) \quad (11.25)$$

where equation (11.22) has been used. The slow dynamic models in equations (11.22) and (11.25) enjoy the same notable properties of the rigid robot dynamic models (Canudas de Wit *et al.*, 1996), hence the control strategies used for rigid manipulators can be adopted.

The fast system in equation (11.23) is a marginally stable linear slowly time-varying system that can be stabilised to the equilibrium manifold $\dot{\mathbf{z}}_f = \mathbf{0}$ ($\dot{\mathbf{z}} = \mathbf{0}$) and $\mathbf{z}_f = \mathbf{0}$ ($\mathbf{z} = \mathbf{z}_s$) by a proper choice of the control input $\mathbf{u}_f$. A reasonable way to achieve this goal is to design a state-space control law of the form:

$$\mathbf{u}_f = \mathbf{K}_1 \dot{\mathbf{z}}_f + \mathbf{K}_2 \mathbf{z}_f \quad (11.26)$$

where, in principle, the matrices $\mathbf{K}_1$ and $\mathbf{K}_2$ should be tuned for every configuration $\boldsymbol{\theta}_s$. However, the computational burden necessary to perform this strategy can be avoided by using constant matrix gains tuned with reference to a given robot configuration (Siciliano and Book, 1988); any state-space technique can be used, for example, based on classical pole placement algorithms.

11.4.2 Force and position regulation

The control objective consists of simultaneous regulation of the contact force $\mathbf{f}$ to a constant set point $\mathbf{f}_d$ and of the position $\mathbf{p}$ to a constant set-point $\mathbf{p}_d$.

In case of contact with an elastically compliant surface, a viable strategy is the parallel control approach (Chiaverini and Sciavicco, 1993), which is especially effective in the case of inaccurate contact modelling. The key feature is to have a force control loop working in parallel to a position control loop along each task space direction. The logical conflict between the two loops is managed by imposing dominance of the force control action over position control, that is, force regulation is always guaranteed at the expense of a position error along the constrained directions.

A force/position parallel regulator controller for rigid robots was proposed in Chiaverini *et al.* (1994), based on position PD position control, gravity compensation, desired force feedforward and PI force control.

For the case of the flexible-link manipulator in equations (11.9) and (11.10), with reference to the slow system in equation (11.25), the following parallel regulator can be adopted:

$$\mathbf{u}_s = \bar{\mathbf{J}}_\theta^T(\boldsymbol{\theta}_s) k_P (\mathbf{p}_r - \mathbf{p}_s) - k_D \dot{\boldsymbol{\theta}}_s \quad (11.27)$$

where $\mathbf{p}_r$ is defined as:

$$\mathbf{p}_r = \mathbf{p}_d + k_p^{-1} \left(k_F (\mathbf{f}_d - \mathbf{f}_s) + k_I \int_0^t (\mathbf{f}_d - \mathbf{f}_s) d\tau \right) \quad (11.28)$$

and $k_P, k_D, k_F, k_I > 0$ are suitable feedback gains.

A better insight into the behaviour of the system during the interaction can be achieved by considering a model of the compliant environment. For the purpose of this work, it is assumed that the same equation can be established in terms of the slow variables, that is,

$$\mathbf{f}_s = k_e \mathbf{n} \mathbf{n}^T (\mathbf{p}_s - \mathbf{p}_o)$$

The above elastic model shows that the contact force is normal to the plane, and thus a null force error can be obtained only if the desired force $\mathbf{f}_d$ is aligned with $\mathbf{n}$. Also, it can be recognised that null position errors can be obtained only on the contact plane while the component of the position along $\mathbf{n}$ has to accommodate the force requirement specified by $\mathbf{f}_d$.

The stability analysis for the slow system in equation (11.25) with the control law in equations (11.27) and (11.28) can be carried out with the same arguments used in Chiaverini *et al.* (1994) for the case of rigid robots. In particular, it can be shown that if the Jacobian $\bar{\mathbf{J}}_\theta(\theta_s)$ of the equivalent rigid manipulator is full-rank, then the closed-loop system has an exponentially stable equilibrium at

$$\mathbf{p}_{s\infty} = (\mathbf{I} - \mathbf{n} \mathbf{n}^T) \mathbf{p}_d + \mathbf{n} \mathbf{n}^T (k_e^{-1} \mathbf{f}_d + \mathbf{p}_o) \quad (11.29)$$

$$\mathbf{f}_{s\infty} = k_e \mathbf{n} \mathbf{n}^T (\mathbf{p}_{s\infty} - \mathbf{p}_o) = \mathbf{f}_d \quad (11.30)$$

where the matrix $(\mathbf{I} - \mathbf{n} \mathbf{n}^T)$ projects the vectors on the contact plane. The equilibrium position is depicted in Figure 11.6. It can be seen that $\mathbf{p}_{s\infty}$ differs from $\mathbf{p}_d$ by a vector aligned along the normal to the contact plane whose magnitude is that necessary to

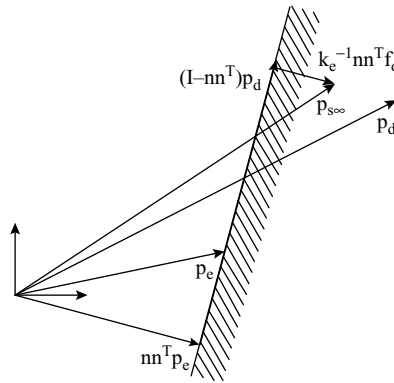


Figure 11.6 Equilibrium position with parallel force and position control

guarantee $\mathbf{f}_{s\infty} = \mathbf{f}_d$ in view of equation (11.30). Therefore, (for the slow system) force regulation is ensured while a null position error is achieved only for the component parallel to the contact plane.

If $\mathbf{f}_d$ is not aligned with $\mathbf{n}$, then it can be found that a drift motion of the end-point of the manipulator is generated along the plane; for this reason, if the contact geometry is unknown, it is advisable to set $\mathbf{f}_d = \mathbf{0}$.

As a final step, the full-order system in equations (11.9) and (11.10) and the composite control law in equation (11.27) with $\mathbf{u}_s$ in equation (11.27) and $\mathbf{u}_f$ in equation (11.26) have to be analysed. By virtue of Tikhonov's theorem it can be shown that regulation of the force $\mathbf{f}$ and of the position $\mathbf{p}$ is achieved with an order ε approximation.

11.4.3 Force regulation and position tracking

If tracking of a time-varying position $\mathbf{p}_d(t)$ on the contact plane is desired (with an order ε approximation), an inverse dynamics parallel control scheme can be adopted for the slow system, that is,

$$\mathbf{u}_s = \bar{\mathbf{B}}_{\theta\theta}(\theta_s) \bar{\mathbf{J}}_{\theta}^{-1}(\theta_s) (\mathbf{a}_s - \bar{\mathbf{h}}(\theta_s, \dot{\theta}_s)) + \bar{\mathbf{c}}_{\theta}(\theta_s, \dot{\theta}_s) \quad (11.31)$$

where $\mathbf{a}_s$ is a new control input and a non-redundant manipulator has been considered.

Substituting equation (11.31) into equation (11.25) gives

$$\dot{\mathbf{p}}_s = \mathbf{a}_s$$

Hence, the control input $\mathbf{a}_s$ can be chosen as

$$\mathbf{a}_s = \dot{\mathbf{p}}_r + k_D(\dot{\mathbf{p}}_r - \dot{\mathbf{p}}_s) + k_P(\mathbf{p}_r - \mathbf{p}_s) \quad (11.32)$$

where

$$\mathbf{p}_r = \mathbf{p}_d + \mathbf{p}_C \quad (11.33)$$

and $\mathbf{p}_C$ is the solution of the differential equation

$$k_A \dot{\mathbf{p}}_C + k_V \mathbf{p}_C = \mathbf{f}_d - \mathbf{f}_s \quad (11.34)$$

where $k_P, k_D, k_A, k_V > 0$ are suitable feedback gains.

Using the same arguments developed in Siciliano and Villani (1999) for rigid robots, it can be easily shown for the slow system that the control law in equations (11.31)–(11.34) ensures regulation of the contact force to the desired set-point $\mathbf{f}_d$ and tracking of the time-varying component of the desired position on the contact plane $(\mathbf{I} - \mathbf{n}\mathbf{n}^T) \mathbf{p}_d(t)$.

As before, Tikhonov's theorem has to be applied to the full-order system in equations (11.9) and (11.10) with the composite control law in equations (11.24), (11.26) and (11.31)–(11.34), and it can be shown that force regulation and position tracking are achieved with an order ε approximation.

11.4.4 Simulation

The above control laws are tested in simulation on the planar two-link flexible manipulator considered in Section 11.3, placed in the same initial position with the end-point in contact with the plane and null contact force. It is desired to reach the end-point position

$$\mathbf{p}_d = [0.55 \quad -0.35]^T \text{ m}$$

and a fifth-order polynomial trajectory with null initial and final velocity and acceleration is imposed from the initial to the final position with a duration of 5 s.

The desired force is taken from zero to the desired value

$$\mathbf{f}_d = [5 \quad 0]^T \text{ N}$$

according to a fifth-order polynomial trajectory with null initial and final velocity and acceleration and duration of 1 s.

The fast control law $\mathbf{u}_f$ has been implemented with $\varepsilon = 0.1606$ and the matrix gains in equation (11.26) were tuned by solving a linear quadratic (LQ) problem for the system in equation (11.23) with the configuration dependent terms computed in the initial manipulator configuration. The matrix weights of the index performance have been chosen so that to preserve the timescale separation between slow and fast dynamics for both the control schemes. The resulting matrix gains are

$$\mathbf{K}_1 = \begin{bmatrix} -0.0372 & -0.0204 & -0.0375 & 0.1495 \\ 0.0573 & 0.0903 & 0.0080 & -0.7856 \end{bmatrix}$$

$$\mathbf{K}_2 = \begin{bmatrix} -0.1033 & -0.0132 & -0.0059 & -0.0053 \\ -0.0882 & 0.0327 & -0.0537 & -0.0217 \end{bmatrix}$$

In order to reproduce in simulation a real situation of a continuous-time system with a digital controller, the control laws are discretised with 5 ms sampling time, while the equations of motion are integrated using a variable step Runge–Kutta method with a minimum step size of 1 ms.

In the first case study, the slow controller in equations (11.27) and (11.28) is considered with the composite control law in equation (11.24). The actual force $\mathbf{f}$ and position $\mathbf{p}$ are used in the slow control law instead of the corresponding slow values, assuming that direct force measurement is available and that the end-point position is computed from joint angles and link deflection measurements via the direct kinematics equation (11.2). The control gains were set to $k_P = 100$ and $k_D = 4$.

Figure 11.7 shows the position error together with the time histories of the desired and actual contact force. It is easy to see that the contact force remains close to the desired value during the end-point motion (notice that the commanded position trajectory has a duration of 5 s) and reaches the desired set-point at steady state. The y -component of the desired position, which corresponds to a direction parallel to the contact plane, is regulated to the desired value. On the other hand, significant error occurs for the x -component; which corresponds to the direction normal to the contact plane, as expected. Notice that the steady-state value of the position error is exactly

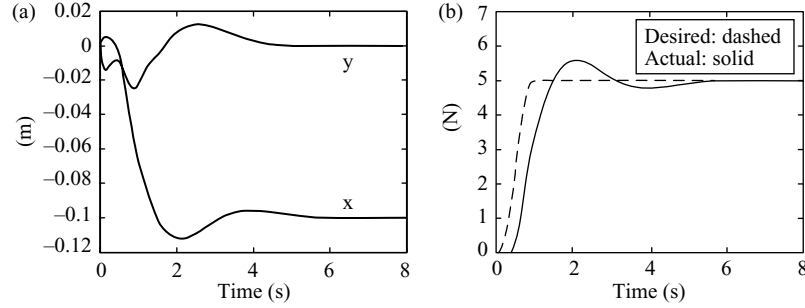


Figure 11.7 Time histories of the position error and of the contact force for the first case study: (a) position error, (b) contact force

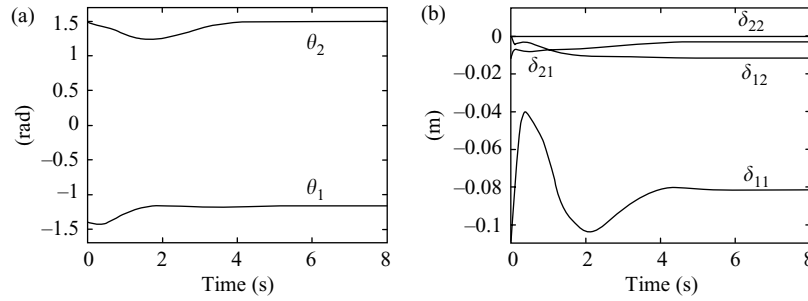


Figure 11.8 Time histories of the joint angles and of the link deflections for the first case study: (a) joint, angle, (b) link deflection

that required to achieve null force error, according to the equilibrium equations (11.29) and (11.30).

It can be seen from the time histories of the joint angles and link deflections shown in Figure 11.8 that the link response oscillations are well damped; moreover, because of gravity and contact force, the manipulator has to bend to reach the desired force and position.

Figure 11.9 shows the time history of the joint torque $\mathbf{u}$ and the first 0.5 s of the time history of the fast torque $\mathbf{u}_f$. It can be observed that the control effort keeps limited values during task execution; remarkably, the control torque $\mathbf{u}_f$ converges to zero with a transient much faster than the transient of $\mathbf{u}$ as expected.

In the second case study, the slow system in equations (11.31)–(11.34) is considered with the composite control law in equation (11.24). As before, the actual force $\mathbf{f}$ and position $\mathbf{p}$ are used in the controller in lieu of the corresponding slow variables. The control gains were set to $k_P = 100$, $k_D = 22$, $k_A = 0.7813$ and $k_V = 13.75$.

The time histories of the contact force and position errors are shown in Figure 11.10. This time the desired force set-point is reached after about 3 s, before

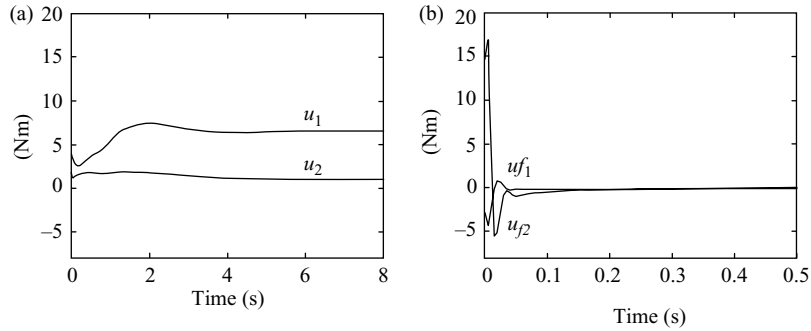


Figure 11.9 Time histories of the joint torques and fast control for the first case study: (a) joint torque, (b) fast control

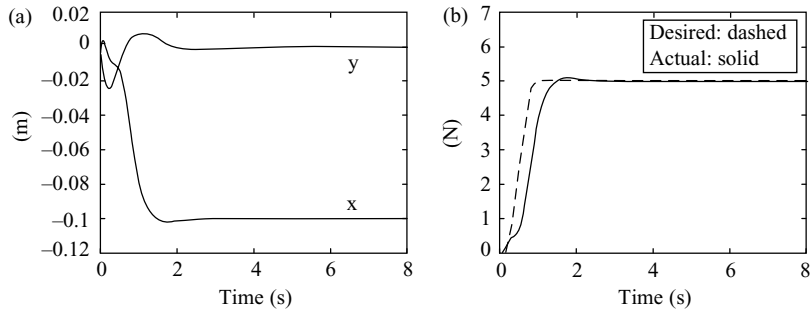


Figure 11.10 Time histories of the position error and of the contact force for the second case study: (a) position error, (b) contact force

the completion of the end-point motion. Moreover, the tracking performance for the y -component of the desired position is better than that in the previous case study.

The time histories of the joint angles and of the link deflections are shown in Figure 11.11, while the time histories of the components of the joint torque vector $\mathbf{u}$ and of the fast torque vector $\mathbf{u}_f$ are shown in Figure 11.12. It can be seen that although the performance is better than that in the previous case study, a similar control effort is required.

It is worth pointing out that the simulation of both the slow control laws without the fast control action in equation (11.26) has revealed an unstable behaviour; the results have not been reported for brevity.

11.5 Summary

The force control problem for a flexible manipulator in contact with a compliant environment has been considered. A dynamic model for a planar manipulator has been presented, which takes into account the forces acting on the end-point of the

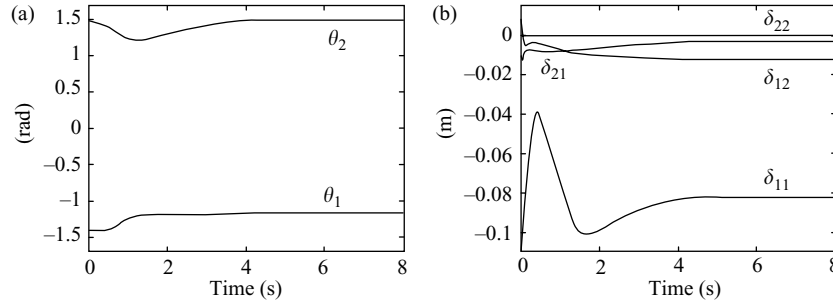


Figure 11.11 Time histories of the joint angles and of the link deflections for the second case study: (a) joint angle, (b) link deflection

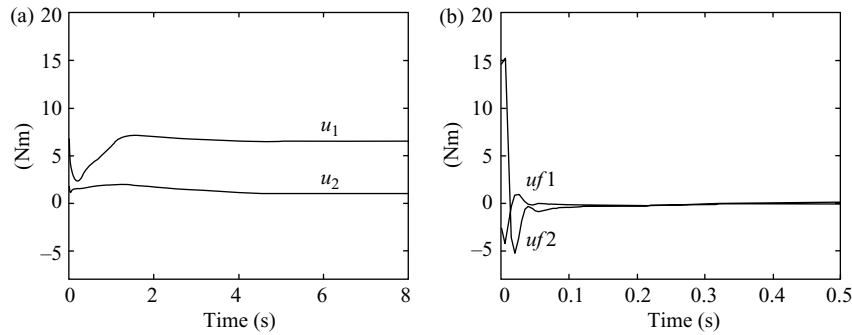


Figure 11.12 Time histories of the joint torques and fast control for the second case study: (a) joint torque, (b) fast control

robot. Two different approaches have been presented: a two-stage algorithm and a composite control law.

The attractive feature of the two-stage scheme is that it does not require force and deflection measurements. The price to pay is that an exact knowledge of the arm kinematics as well as the stiffness and position of the environment at rest are required to guarantee regulation of the force and of end-point position to a constant value.

In the composite control law, on the other hand, the additional objective of damping the vibrations that are naturally excited during the task execution is explicitly considered. By using the singular perturbation theory, the system has been split into a slow subsystem describing the rigid motion dynamics and a fast subsystem describing the flexible dynamics. This allows designing a fast control action for vibration damping as well as adopting algorithms designed for the rigid motion to achieve force and position control. Two different controllers have been considered, in the framework of parallel force and position control.

The simulation case studies developed on the model of a two-link flexible manipulator under gravity in contact with a compliant environment have confirmed the theoretical results.