CHAPTER 11

Force control of manipulators

11.1 INTRODUCTION

Position control is appropriate when a manipulator is following a trajectory through space, but when any contact is made between the end-effector and the manipulator's environment, mere position control might not suffice. Consider a manipulator washing a window with a sponge. The compliance of the sponge might make it possible to regulate the force applied to the window by controlling the position of the end-effector relative to the glass. If the sponge is very compliant or the position of the glass is known very accurately, this technique could work quite well.

If, however, the stiffness of the end-effector, tool, or environment is high, it becomes increasingly difficult to perform operations in which the manipulator presses against a surface. Instead of washing with a sponge, imagine that the manipulator is scraping paint off a glass surface, using a rigid scraping tool. If there is any uncertainty in the position of the glass surface or any error in the position of the manipulator, this task would become impossible. Either the glass would be broken, or the manipulator would wave the scraping tool over the glass with no contact taking place.

In both the washing and scraping tasks, it would be more reasonable not to specify the position of the plane of the glass, but rather *to specify a force that is to be maintained normal to the surface.*

More so than in previous chapters, in this chapter we present methods that are not yet employed by industrial robots, except in an extremely simplified way. The major thrust of the chapter is to introduce the **hybrid position/force controller**, which is one formalism through which industrial robots might someday be controlled in order to perform tasks requiring force control. However, regardless of which method(s) emerge as practical for industrial application, many of the concepts introduced in this chapter will certainly remain valid.

11.2 APPLICATION OF INDUSTRIAL ROBOTS TO ASSEMBLY TASKS

The majority of the industrial robot population is employed in relatively **simple applications**, such as spot welding, spray painting, and pick-and-place operations. Force control has already appeared in a few applications; for example, some robots are already capable of simple force control that allows them to do such tasks as grinding and deburring. Apparently, the next big area of application will be to assembly-line tasks in which one or more parts are mated. In such **parts-mating** tasks, monitoring and control of the forces of contact are extremely important.

Precise control of manipulators in the face of uncertainties and variations in their work environments is a prerequisite to application of robot manipulators to assembly operations in industry. It seems that, by providing manipulator hands with sensors that can give information about the state of manipulation tasks, important progress can be made toward using robots for assembly tasks. Currently, the dexterity of manipulators remains quite low and continues to limit their application in the automated assembly area.

The use of manipulators for assembly tasks requires that the precision with which parts are positioned with respect to one another be quite high. Current industrial robots are often not accurate enough for these tasks, and building robots that are might not make sense. Manipulators of greater precision can be achieved only at the expense of size, weight, and cost. The ability to measure and control contact forces generated at the hand, however, offers a possible alternative for extending the effective precision of a manipulation. Because relative measurements are used, absolute errors in the position of the manipulator and the manipulated objects are not as important as they would be in a purely position-controlled system. Small variations in relative position generate large contact forces when parts of moderate stiffness interact, so knowledge and control of these forces can lead to a tremendous increase in effective positional accuracy.

11.3 A FRAMEWORK FOR CONTROL IN PARTIALLY CONSTRAINED TASKS

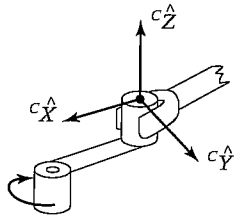
The approach presented in this chapter is based on a framework for control in situations in which motion of the manipulator is partially constrained by contact with one or more surfaces [1–3]. This framework for understanding partially constrained tasks is based on a simplified model of interaction between the manipulator's end-effector and the environment: We are interested in describing contact and freedoms, so we consider only the forces due to contact. This is equivalent to doing a quasi-static analysis and ignoring other static forces, such as certain friction components and gravity. The analysis is reasonable where forces due to contact between relatively stiff objects are the dominant source of forces acting on the system. Note that the methodology presented here is somewhat simplistic and has some limitations, but it is a good way to introduce the basic concepts involved and do so at a level appropriate for this text. For a related, but more general and rigorous methodology, see [19].

Every manipulation task can be broken down into subtasks that are defined by a particular contact situation occurring between the manipulator end-effector (or tool) and the work environment. With each such subtask, we can associate a set of constraints, called the **natural constraints**, that result from the particular

mechanical and geometric characteristics of the task configuration. For instance, a hand in contact with a stationary, rigid surface is not free to move through that surface; hence, a *natural* position constraint exists. If the surface is frictionless, the hand is not free to apply arbitrary forces tangent to the surface; thus, a *natural* force constraint exists.

In our model of contact with the environment, for each subtask configuration, a **generalized surface** can be defined with position constraints along the normals to this surface and force constraints along the tangents. These two types of constraint, force and position, partition the degrees of freedom of possible end-effector motions into two orthogonal sets that must be controlled according to different criteria. Note that this model of contact does not include all possible contacting situations. (See [19] for a more general scheme.)

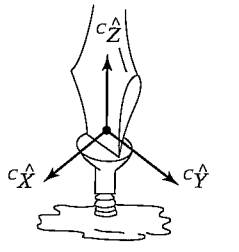
Figure 11.1 shows two representative tasks along with their associated natural constraints. Notice that, in each case, the task is described in terms of a frame $\{C\}$, the so-called **constraint frame**, which is located in a task-relevant location. According to the task, $\{C\}$ could be fixed in the environment or could move with the end-effector of the manipulator. In Fig. 11.1(a), the constraint frame is attached to the crank as shown and moves with the crank, with the $\hat{X}$ direction always directed toward the pivot point of the crank. Friction acting at the fingertips ensures a secure grip on the handle, which is on a spindle so that it can rotate relative to the crank arm. In Fig. 11.1(b), the constraint frame is attached to the tip of the screwdriver and moves with it as the task proceeds. Notice that, in the $\hat{Y}$ direction, the force is constrained to be zero, because the slot of the screw would allow the screwdriver to slip out in that direction. In these examples, a given set of constraints remains true throughout the task. In more complex situations, the task is broken into subtasks for which a constant set of natural constraints can be identified.



Natural constraints

$v_x = 0$	$f_y = 0$
$v_z = 0$	$n_z = 0$
$v_x = 0$	
$\omega_y = 0$	

(a) Turning crank



Natural constraints

$v_x = 0$	$f_y = 0$
$\omega_x = 0$	$n_z = 0$
$\omega_y = 0$	
$v_z = 0$	

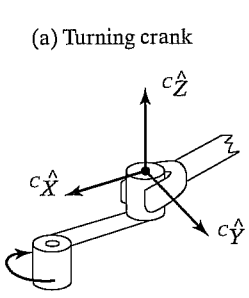
(b) Turning screwdriver

FIGURE 11.1: The natural constraints for two different tasks.

In Fig. 11.1, position constraints have been indicated by giving values for components of velocity of the end-effector, $\mathcal{V}$, described in frame $\{C\}$. We could just as well have indicated position constraints by giving expressions for position, rather than velocities; however, in many cases, it is simpler to specify a position constraint as a "velocity equals zero" constraint. Likewise, force constraints have been specified by giving values to components of the force-moment vector, $\mathcal{F}$, acting on the end-effector described in frame $\{C\}$. Note that when we say *position constraints*, we mean position or orientation constraints, and when we say *force constraints*, we mean force or moment constraints. The term *natural constraints* is used to indicate that these constraints arise naturally from the particular contacting situation. They have nothing to do with the desired or intended motion of the manipulator.

Additional constraints, called **artificial constraints**, are introduced in accordance with the natural constraints to specify desired motions or force application. That is, each time the user specifies a desired trajectory in either position or force, an artificial constraint is defined. These constraints also occur along the tangents and normals of the generalized constraint surface, but, unlike natural constraints, artificial force constraints are specified along surface normals, and artificial position constraints along tangents—hence, consistency with the natural constraints is preserved.

Figure 11.2 shows the natural and artificial constraints for two tasks. Note that when a natural position constraint is given for a particular degree of freedom in $\{C\}$,



(a) Turning crank

Natural constraints

$v_x = 0$ $f_y = 0$
$v_z = 0$ $n_z = 0$
$\omega_x = 0$
$\omega_y = 0$

Artificial constraints

$v_y = r\alpha_1$ $f_x = 0$
$\omega_z = \alpha_1$ $f_z = 0$
$n_x = 0$
$n_y = 0$

(b) Turning screwdriver

Natural constraints

$v_x = 0$ $f_y = 0$
$\omega_x = 0$ $n_z = 0$
$\omega_y = 0$
$v_z = 0$

Artificial constraints

$v_y = 0$ $f_x = 0$
$\omega_z = \alpha_2$ $n_x = 0$
$n_y = 0$
$f_z = \alpha_3$

FIGURE 11.2: The natural and artificial constraints for two tasks.

an artificial force constraint should be specified, and vice versa. At any instant, any given degree of freedom in the constraint frame is controlled to meet either a position or a force constraint.

Assembly strategy is a term that refers to a sequence of planned artificial constraints that will cause the task to proceed in a desirable manner. Such strategies must include methods by which the system can detect a change in the contacting situation so that transitions in the natural constraints can be tracked. With each such change in natural constraints, a new set of artificial constraints is recalled from the set of assembly strategies and enforced by the control system. Methods for automatically choosing the constraints for a given assembly task await further research. In this chapter, we will assume that a task has been analyzed in order to determine the natural constraints and that a human planner has determined an **assembly strategy** with which to control the manipulator.

Note that we will usually ignore friction forces between contacting surfaces in our analysis of tasks. This will suffice for our introduction to the problem and in fact will yield strategies that work in many cases. Usually friction forces of sliding are acting in directions chosen to be position controlled, and so these forces appear as disturbances to the position servo and are overcome by the control system.

EXAMPLE 11.1

Figure 11.3(a)–(d) shows an assembly sequence used to put a round peg into a round hole. The peg is brought down onto the surface to the left of the hole and then slid along the surface until it drops into the hole. It is then inserted until the peg reaches the bottom of the hole, at which time the assembly is complete. Each of the four indicated contacting situations defines a subtask. For each of the subtasks shown, give the natural and artificial constraints. Also, indicate how the system senses the change in the natural constraints as the operation proceeds.

First, we will attach the constraint frame to the peg as shown in Fig. 11.3(a). In Fig. 11.3(a), the peg is in free space, and so the natural constraints are

$$^{C}\mathcal{F} = 0. \tag{11.1}$$

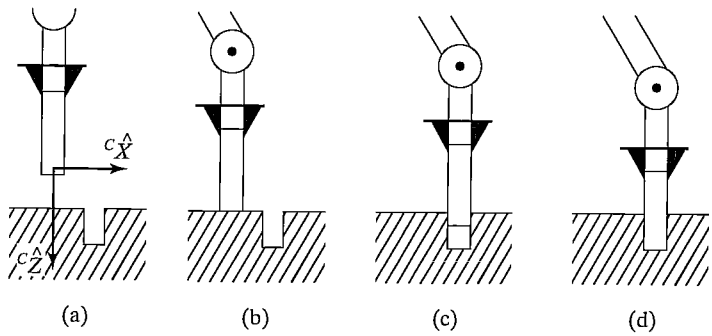


FIGURE 11.3: The sequence of four contacting situations for peg insertion.

Therefore, the artificial constraints in this case constitute an entire position trajectory, which moves the peg in the ${}^{C}\hat{Z}$ direction toward the surface. For example,

$$ {}^{C}v = \begin{bmatrix} 0 \\ 0 \\ v_{\text{approach}} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{11.2} $$

where v_{approach} is the speed with which to approach the surface.

In Fig. 11.3(b), the peg has reached the surface. To detect that this has happened, we observe the force in the ${}^{C}\hat{Z}$ direction. When this sensed force exceeds a threshold, we sense contact, which implies a new contacting situation with a new set of natural constraints. Assuming that the contacting situation is as shown in Fig. 11.3(b), the peg is not free to move in ${}^{C}\hat{Z}$, or to rotate about ${}^{C}\hat{X}$ or ${}^{C}\hat{Y}$. In the other three degrees of freedom, it is not free to apply forces; hence, the natural constraints are

$$ \begin{aligned} {}^{C}v_z &= 0, \\ {}^{C}\omega_x &= 0, \\ {}^{C}\omega_y &= 0, \\ {}^{C}f_x &= 0, \\ {}^{C}f_y &= 0, \\ {}^{C}n_z &= 0. \end{aligned} \tag{11.3} $$

The artificial constraints describe the strategy of sliding along the surface in the ${}^{C}\hat{X}$ direction while applying small forces to ensure that contact is maintained. Thus, we have

$$ \begin{aligned} {}^{C}v_x &= v_{\text{slide}}, \\ {}^{C}v_y &= 0, \\ {}^{C}\omega_z &= 0, \\ {}^{C}f_z &= f_{\text{contact}}, \\ {}^{C}n_x &= 0, \\ {}^{C}n_y &= 0. \end{aligned} \tag{11.4} $$

where f_{contact} is the force applied normal to the surface as the peg is slid, and v_{slide} is the velocity with which to slide across the surface.

In Fig. 11.3(c), the peg has fallen slightly into the hole. This situation is sensed by observing the velocity in the ${}^{C}\hat{Z}$ direction and waiting for it to cross a threshold (to become nonzero, in the ideal case). When this is observed, it signals that once again the natural constraints have changed, and thus our strategy (as embodied in

the artificial constraints) must change. The new natural constraints are

$$\begin{aligned} {}^{C}v_x &= 0, \\ {}^{C}v_y &= 0, \\ {}^{C}\omega_x &= 0, \\ {}^{C}\omega_y &= 0, \\ {}^{C}f_x &= 0, \\ {}^{C}n_z &= 0. \end{aligned} \tag{11.5}$$

We choose the artificial constraints to be

$$\begin{aligned} {}^{C}v_z &= v_{\text{insert}}, \\ {}^{C}\omega_z &= 0, \\ {}^{C}f_x &= 0, \\ {}^{C}f_y &= 0, \\ {}^{C}n_x &= 0, \\ {}^{C}n_y &= 0, \end{aligned} \tag{11.6}$$

where v_{insert} is the velocity at which the peg is inserted into the hole. Finally, the situation shown in Fig. 11.3(d) is detected when the force in the ${}^{C}\hat{Z}$ direction increases above a threshold.

It is interesting to note that changes in the natural constraints are always detected by observing the position or force variable that is *not* being controlled. For example, to detect the transition from Fig. 11.3(b) to Fig. 11.3(c), we monitor the velocity in ${}^{C}\hat{Z}$ while we are controlling force in ${}^{C}\hat{Z}$. To discover when the peg has hit the bottom of the hole, we monitor ${}^{C}f_z$, although we are controlling ${}^{C}v_z$.

The framework we have introduced is somewhat simplistic. A more general and rigorous method of "splitting" tasks into position-controlled and force-controlled components can be found in [19].

Determining assembly strategies for fitting more complicated parts together is quite complex. We have also neglected the effects of uncertainty in our simple analysis of this task. The development of automatic planning systems that include the effects of uncertainty and can be applied to practical situations has been a research topic [4–8]. For a good review of these methods, see [9].

11.4 THE HYBRID POSITION/FORCE CONTROL PROBLEM

Figure 11.4 shows two extreme examples of contacting situations. In Fig. 11.4(a), the manipulator is moving through free space. In this case, the natural constraints are all force constraints—there is nothing to react against, so all forces are constrained

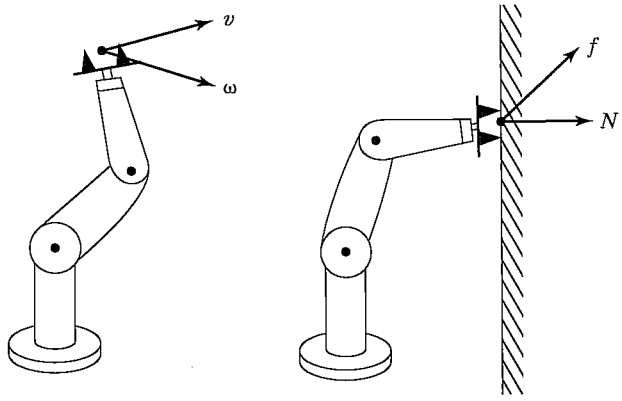


FIGURE 11.4: The two extremes of contacting situations. The manipulator on the left is moving in free space where no reaction surface exits. The manipulator on the right is glued to the wall so that no free motion is possible.

to be zero.[1] With an arm having six degrees of freedom, we are free to move in six degrees of freedom in position, but we are unable to exert forces in any direction. Figure 11.4(b) shows the extreme situation of a manipulator with its end-effector glued to a wall. In this case, the manipulator is subject to six natural position constraints, because it is not free to be repositioned. However, the manipulator is free to exert forces and torques to the object with six degrees of freedom.

In Chapters 9 and 10, we studied the position-control problem that applies to the situation of Fig. 11.4(a). The situation of Fig. 11.4(b) does not occur very often in practice; we usually must consider force control in the context of partially constrained tasks, in which some degrees of freedom of the system are subject to position control and others are subject to force control. Thus, in this chapter, we are interested in considering **hybrid position/force control** schemes.

The hybrid position/force controller must solve three problems:

1. Position control of a manipulator along directions in which a natural force constraint exists.
2. Force control of a manipulator along directions in which a natural position constraint exists.
3. A scheme to implement the arbitrary mixing of these modes along orthogonal degrees of freedom of an arbitrary frame, $\{C\}$.

11.5 FORCE CONTROL OF A MASS–SPRING SYSTEM

In Chapter 9, we began our study of the complete position-control problem with the study of the very simple problem of controlling a single block of mass. We were then able, in Chapter 10, to use a model of the manipulator in such a way that the problem of controlling the entire manipulator became equivalent to controlling n independent

[1] It is important to remember that we are concerned here with *forces of contact* between end-effector and environment, not inertial forces.

masses (for a manipulator with n joints). In a similar way, we begin our look at force control by controlling the force applied by a simple single-degree-of-freedom system.

In considering forces of contact, we must make some model of the environment upon which we are acting. For the purposes of conceptual development, we will use a very simple model of interaction between a controlled body and the environment. We model contact with an environment as a spring—that is, we assume our system is rigid and the environment has some stiffness, k_e.

Let us consider the control of a mass attached to a spring, as in Fig. 11.5. We will also include an unknown disturbance force, f_{dist}, which might be thought of as modeling unknown friction or cogging in the manipulator's gearing. The variable we wish to control is the force acting on the environment, f_e, which is the force acting in the spring:

$$f_e = k_e x. \tag{11.7}$$

The equation describing this physical system is

$$f = m\ddot{x} + k_e x + f_{\text{dist}}, \tag{11.8}$$

or, written in terms of the variable we wish to control, f_e,

$$f = mk_e^{-1}\ddot{f}_e + f_e + f_{\text{dist}}. \tag{11.9}$$

Using the partitioned-controller concept, as well as

$$\alpha = mk_e^{-1}$$

and

$$\beta = f_e + f_{\text{dist}}$$

we arrive at the control law,

$$f = mk_e^{-1}\left[\ddot{f}_d + k_{vf}\dot{e}_f + k_{pf}e_f\right] + f_e + f_{\text{dist}}, \tag{11.10}$$

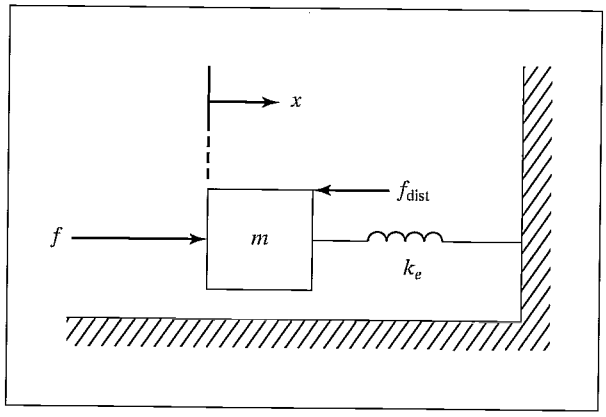


FIGURE 11.5: A spring–mass system.

where $e_f = f_d - f_e$ is the force error between the desired force, f_d, and the sensed force on the environment, f_e. If we could compute (11.10), we would have the closed-loop system

$$\ddot{e}_f + k_{vf}\dot{e}_f + k_{pf}e_f = 0. \tag{11.11}$$

However, we cannot use knowledge of f_{dist} in our control law, and so (11.10) is not feasible. We might leave that term out of the control law, but a steady-state analysis shows that there is a better choice, especially when the stiffness of the environment, k_e, is high (the usual situation).

If we choose to leave the f_{dist} term out of our control law, equate (11.9) and (11.10), and do a steady-state analysis by setting all time derivatives to zero, we find that

$$e_f = \frac{f_{\text{dist}}}{\alpha}, \tag{11.12}$$

where $\alpha = mk_e^{-1}k_{pf}$, the effective force-feedback gain; however, if we choose to use f_d in the control law (11.10) in place of the term $f_e + f_{\text{dist}}$, we find the steady-state error to be

$$e_f = \frac{f_{\text{dist}}}{1+\alpha}. \tag{11.13}$$

When the environment is stiff, as is often the case, α might be small, and so the steady-state error calculated in (11.13) is quite an improvement over that of (11.12). Therefore, we suggest the control law

$$f = mk_e^{-1}\left[\ddot{f}_d + k_{vf}\dot{e}_f + k_{pf}e_f\right] + f_d. \tag{11.14}$$

Figure 11.6 is a block diagram of the closed-loop system using the control law (11.14).

Generally, practical considerations change the implementation of a force-control servo quite a bit from the ideal shown in Fig. 11.6. First, force trajectories are

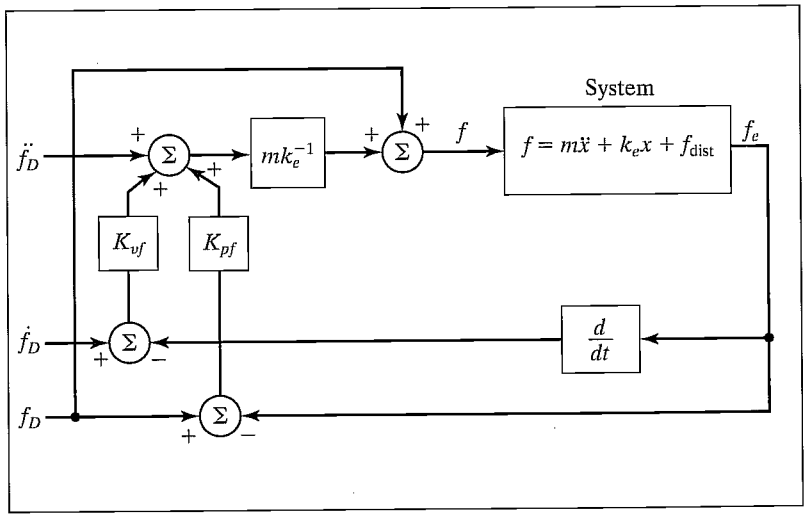


FIGURE 11.6: A force control system for the spring–mass system.

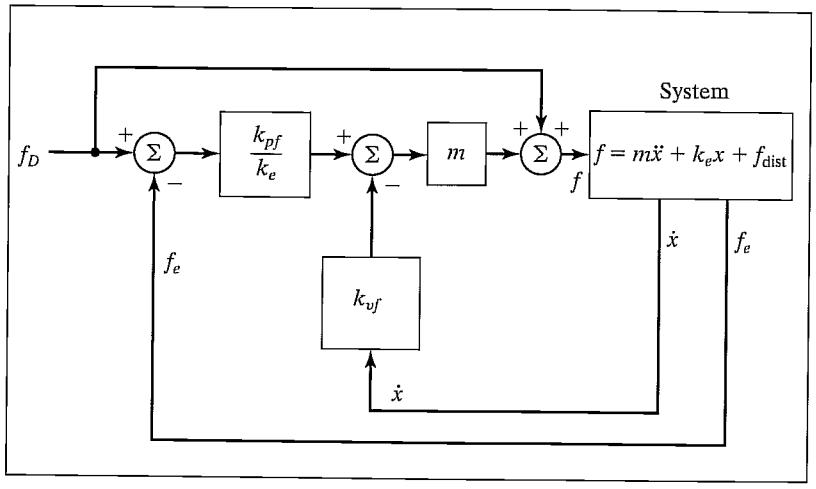


FIGURE 11.7: A practical force-control system for the spring–mass system.

usually constants—that is, we are usually interested in controlling the contact force to be at some constant level. Applications in which contact forces should follow some arbitrary function of time are rare. Therefore, the $\dot{f}_d$ and $\ddot{f}_d$ inputs of the control system are very often permanently set to zero. Another reality is that sensed forces are quite "noisy," and numerical differentiation to compute $\dot{f}_e$ is ill-advised. However, $f_e = k_e x$, so we can obtain the derivative of the force on the environment as $\dot{f}_e = k_e \dot{x}$. This is much more realistic, in that most manipulators have means of obtaining good measures of velocity. Having made these two pragmatic choices, we write the control law as

$$f = m\left[k_{pf} k_e^{-1} e_f - k_{vf}\dot{x}\right] + f_d, \tag{11.15}$$

with the corresponding block diagram shown in Fig. 11.7.

Note that an interpretation of the system of Fig. 11.7 is that force errors generate a set-point for an inner velocity-control loop with gain k_{vf}. Some force-control laws also include an integral term to improve steady-state performance.

An important remaining problem is that the stiffness of the environment, k_e, appears in our control law, but is often unknown and perhaps changes from time to time. However, often an assembly robot is dealing with rigid parts, and k_e could be guessed to be quite high. Generally this assumption is made, and gains are chosen such that the system is somewhat robust with respect to variations in k_e.

The purpose in constructing a control law to control the force of contact has been to show one suggested structure and to expose a few issues. For the remainder of this chapter, we will simply assume that such a force-controlling servo could be built and abstract it away into a black box, as shown in Fig. 11.8. In practice, it is not easy to build a high-performance force servo, and it is currently an area of active research [11–14]. For a good review of this area, see [15].

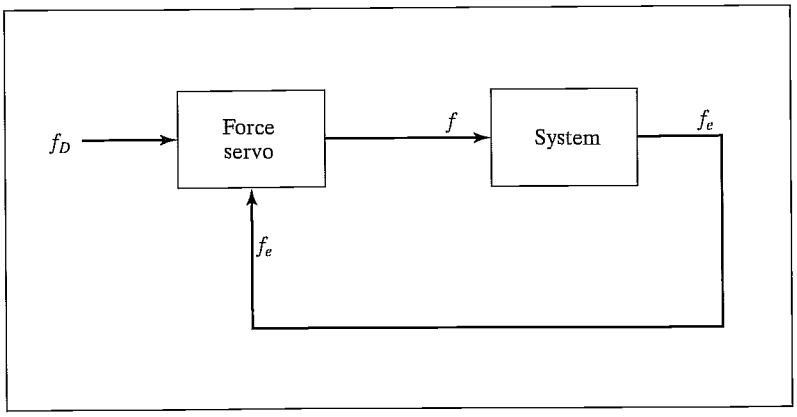


FIGURE 11.8: The force-control servo as a black box.

11.6 THE HYBRID POSITION/FORCE CONTROL SCHEME

In this section, we introduce an architecture for a control system that implements the hybrid position/force controller.

A Cartesian manipulator aligned with $\{C\}$

We will first consider the simple case of a manipulator having three degrees of freedom with prismatic joints acting in the $\hat{Z}$, $\hat{Y}$, and $\hat{X}$ directions. For simplicity, we will assume that each link has mass m and slides on frictionless bearings. Let us also assume that the joint motions are lined up exactly with the constraint frame, $\{C\}$. The end-effector is in contact with a surface of stiffness k_e that is oriented with its normal in the $-{}^C\hat{Y}$ direction. Hence, force control is required in that direction, but position control in the ${}^C\hat{X}$ and ${}^C\hat{Z}$ directions. (See Fig. 11.9.)

In this case, the solution to the hybrid position/force control problem is clear. We should control joints 1 and 3 with the position controller developed for a unit mass in Chapter 9. Joint 2 (operating in the $\hat{Y}$ direction) should be controlled with the force controller developed in Section 11.4. We could then supply a position trajectory in the ${}^C\hat{X}$ and ${}^C\hat{Z}$ directions, while independently supplying a force trajectory (perhaps just a constant) in the ${}^C\hat{Y}$ direction.

If we wish to be able to switch the nature of the constraint surface such that its normal might also be $\hat{X}$ or $\hat{Z}$, we can slightly generalize our Cartesian arm-control system as follows: We build the structure of the controller such that we may specify a complete position trajectory in all three degrees of freedom and also a force trajectory in all three degrees of freedom. Of course, we can't control so as to meet these six constraints at any one time—rather, we will set modes to indicate which components of which trajectory will be followed at any given time.

Consider the controller shown in Fig. 11.10. Here, we indicate the control of all three joints of our simple Cartesian arm in a single diagram by showing both the position controller and the force controller. The matrices S and S' have been introduced to control which mode—position or force—is used to control each joint of the Cartesian arm. The S matrix is diagonal, with ones and zeros on the diagonal.

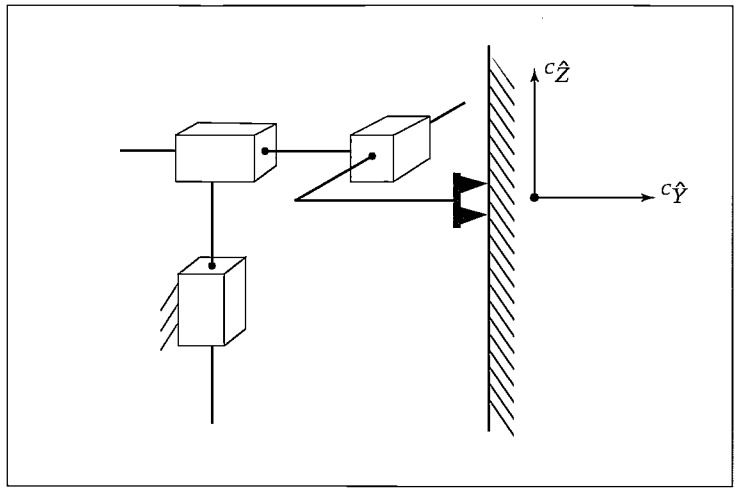


FIGURE 11.9: A Cartesian manipulator with three degrees of freedom in contact with a surface.

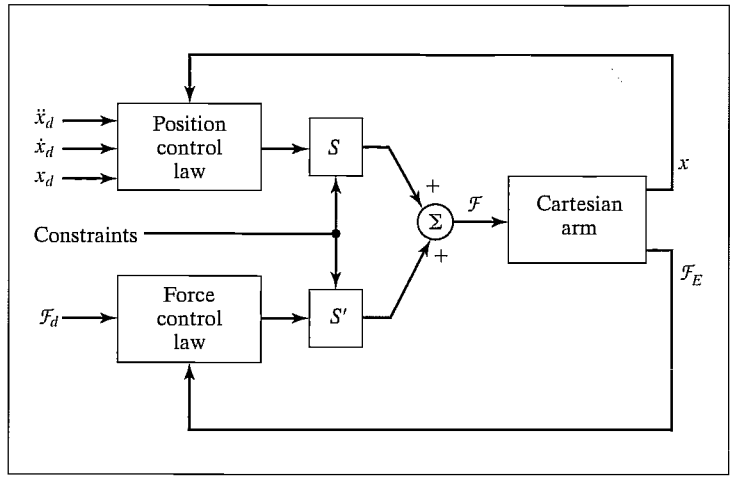


FIGURE 11.10: The hybrid controller for a 3-DOF Cartesian arm.

Where a one is present in S, a zero is present in S' and position control is in effect. Where a zero is present in S, a one is present in S' and force control is in effect. Hence the matrices S and S' are simply switches that set the mode of control to be used with each degree of freedom in $\{C\}$. In accordance with the setting of S, there are always three components of the trajectory being controlled, yet the mix between position control and force control is arbitrary. The other three components of desired trajectory and associated servo errors are being ignored. Hence, when a certain degree of freedom is under force control, position errors on that degree of freedom are ignored.

EXAMPLE 11.2

For the situation shown in Fig. 11.9, with motions in the ${}^{C}\hat{Y}$ direction constrained by the reaction surface, give the matrices S and S'.

Because the $\hat{X}$ and $\hat{Z}$ components are to be position controlled, we enter ones on the diagonal of S corresponding to these two components. This will cause the position servo to be active in these two directions, and the input trajectory will be followed. Any position trajectory input for the $\hat{Y}$ component will be ignored. The S' matrix has the ones and zeros on the diagonal inverted; hence, we have

$$S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$S' = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{11.16}$$

Figure 11.10 shows the hybrid controller for the special case that the joints line up exactly with the constraint frame, $\{C\}$. In the following subsection, we use techniques studied in previous chapters to generalize the controller to work with general manipulators and for an arbitrary $\{C\}$; however, in the ideal case, the system performs as if the manipulator had an actuator "lined up" with each of the degrees of freedom in $\{C\}$.

A general manipulator

Generalizing the hybrid controller shown in Fig. 11.10 so that a general manipulator may be used is straightforward with the concept of Cartesian-based control. Chapter 6 discussed how the equations of motion of a manipulator could be written in terms of Cartesian motion of the end-effector, and Chapter 10 showed how such a formulation might be used to achieve decoupled Cartesian position control of a manipulator. The major idea is that, through use of a dynamic model written in Cartesian space, it is possible to control so that the combined system of the actual manipulator and computed model appear as a set of independent, uncoupled unit masses. Once this decoupling and linearizing are done, we can apply the simple servo already developed in Section 11.4.

Figure 11.11 shows the compensation based on the formulation of the manipulator dynamics in Cartesian space such that the manipulator appears as a set of uncoupled unit masses. For use in the hybrid control scheme, the Cartesian dynamics and the Jacobian are written in the constraint frame, $\{C\}$. Likewise, the kinematics are computed with respect to the constraint frame.

Because we have designed the hybrid controller for a Cartesian manipulator aligned with the constraint frame, and because the Cartesian decoupling scheme provides us with a system with the same input–output properties, we need only combine the two to generate the generalized hybrid position/force controller.

Figure 11.12 is a block diagram of the hybrid controller for a general manipulator. Note that the dynamics are written in the constraint frame, as is the Jacobian.

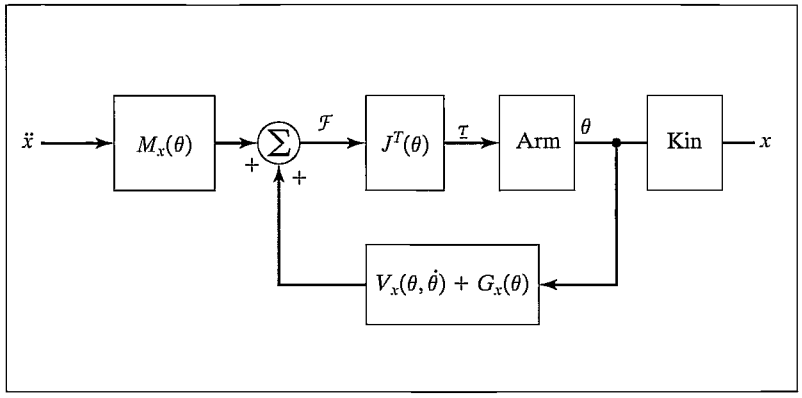


FIGURE 11.11: The Cartesian decoupling scheme introduced in Chapter 10.

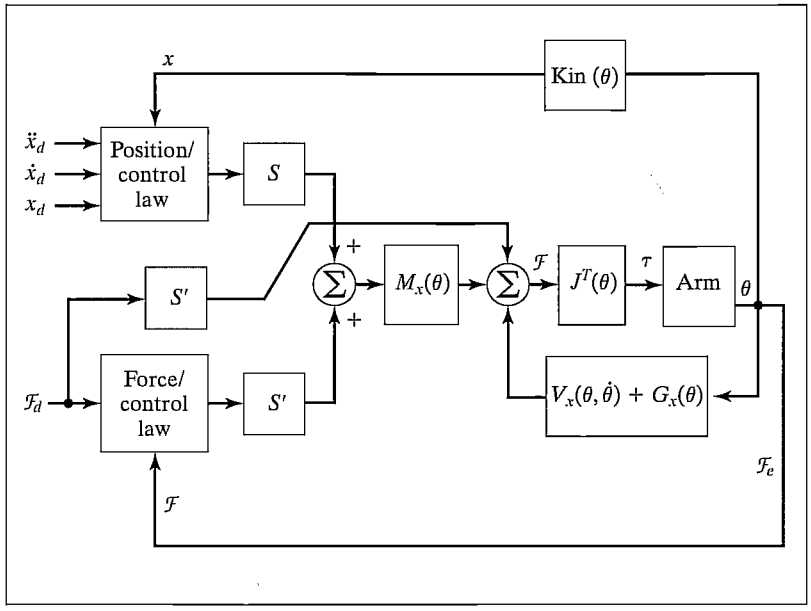


FIGURE 11.12: The hybrid position/force controller for a general manipulator. For simplicity, the velocity-feedback loop has not been shown.

The kinematics are written to include the transformation of coordinates into the constraint frame, and the sensed forces are likewise transformed into $\{C\}$. Servo errors are calculated in $\{C\}$, and control modes within $\{C\}$ are set through proper choice of S.[2] Figure 11.13 shows a manipulator being controlled by such a system.

[2]The partitioning of control modes along certain task-related directions has been generalized in [10] from the more basic approach presented in this chapter.

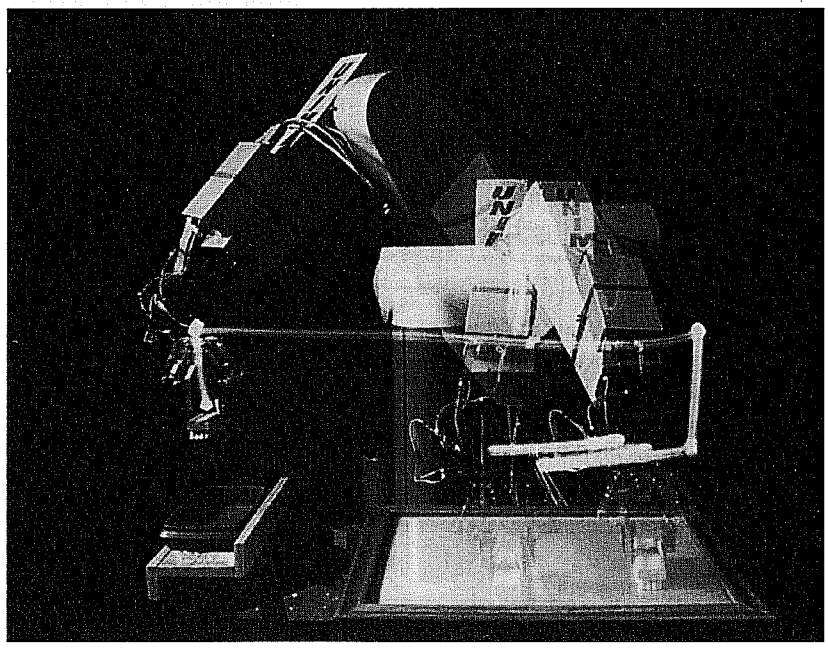


FIGURE 11.13: A PUMA 560 manipulator washes a window under control of the COSMOS system developed under O. Khatib at Stanford University. These experiments use force-sensing fingers and a control structure similar to that of Fig. 11.12 [10].

Adding variable stiffness

Controlling a degree of freedom in strict position or force control represents control at two ends of the spectrum of servo stiffness. An ideal position servo is infinitely stiff and rejects all force disturbances acting on the system. In contrast, an ideal force servo exhibits zero stiffness and maintains the desired force application regardless of position disturbances. It could be useful to be able to control the end-effector to exhibit stiffnesses other than zero or infinite. In general, we might wish to control the **mechanical impedance** of the end-effector [14, 16, 17].

In our analysis of contact, we have imagined that the environment is very stiff. When we contact a stiff environment, we use zero-stiffness force control. When we contact zero stiffness (moving in free space) we use high-stiffness position control. Hence, it appears that controlling the end-effector to exhibit a stiffness that is approximately the inverse of the local environment is perhaps a good strategy. Therefore, in dealing with plastic parts or springs, we could wish to set servo stiffness to other than zero or infinite.

Within the framework of the hybrid controller, this is done simply by using position control and lowering the position gain corresponding to the appropriate degree of freedom in $\{C\}$. Generally, if this is done, the corresponding velocity gain is lowered so that that degree of freedom remains critically damped. The ability to change both position and velocity gains of the position servo along the

degrees of freedom of $\{C\}$ allows the hybrid position/force controller to implement a generalized impedance of the end-effector [17]. However, in many practical situations we are dealing with the interaction of stiff parts, so that pure position control or pure force control is desired.

11.7 CURRENT INDUSTRIAL-ROBOT CONTROL SCHEMES

True force control, such as the hybrid position/force controller introduced in this chapter, does not exist today in industrial robots. Among the problems of practical implementation are the rather large amount of computation required, lack of accurate parameters for the dynamic model, lack of rugged force sensors, and the burden of difficulty placed on the user in specifying a position/force strategy.

Passive compliance

Extremely rigid manipulators with very stiff position servos are ill-suited to tasks in which parts come into contact and contact forces are generated. In such situations, parts are often jammed or damaged. Ever since early experiments with manipulators attempting to do assembly, it was realized that, to the extent that the robots could perform such tasks, it was only thanks to the compliance of the parts, of the fixtures, or of the arm itself. This ability of one or more parts of the system to "give" a little was often enough to allow the successful mating of parts.

Once this was realized, devices were specially designed to introduce compliance into the system on purpose. The most successful such device is the RCC or *remote center compliance* device developed at Draper Labs [18]. The RCC was cleverly designed so that it introduced the "right" kind of compliance, which allowed certain tasks to proceed smoothly and rapidly with little or no chance of jamming. The RCC is essentially a spring with six degrees of freedom, which is inserted between the manipulator's wrist and the end-effector. By setting the stiffnesses of the six springs, various amounts of compliance can be introduced. Such schemes are called **passive-compliance** schemes and are used in industrial applications of manipulators in some tasks.

Compliance through softening position gains

Rather than achieving compliance in a passive, and therefore fixed, way, it is possible to devise schemes in which the apparent stiffness of the manipulator is altered through adjustment of the gains of a position-control system. A few industrial robots do something of this type for applications such as grinding, in which contact with a surface needs to be maintained but delicate force control is not required.

A particularly interesting approach has been suggested by Salisbury [16]. In this scheme, the position gains in a joint-based servo system are modified in such a way that the end-effector appears to have a certain stiffness along Cartesian degrees of freedom: Consider a general spring with six degrees of freedom. Its action could be described by

$$\mathcal{F} = K_{px}\delta\chi, \tag{11.17}$$

where K_{px} is a diagonal 6×6 matrix with three linear stiffnesses followed by three torsional stiffnesses on the diagonal. How could we make the end-effector of a manipulator exhibit this stiffness characteristic?

Recalling the definition of the manipulator Jacobian, we have

$$\delta\chi = J(\Theta)\delta\Theta. \tag{11.18}$$

Combining with (11.17) gives

$$\mathcal{F} = K_{px}J(\Theta)\delta\Theta. \tag{11.19}$$

From static-force considerations, we have

$$\tau = J^T(\Theta)\mathcal{F}, \tag{11.20}$$

which, combined with (11.19), yields

$$\tau = J^T(\Theta)K_{px}J(\Theta)\delta\Theta. \tag{11.21}$$

Here, the Jacobian is usually written in the tool frame. Equation (11.21) is an expression for how joint torques should be generated as a function of small changes in joint angles, $\delta\Theta$, in order to make the manipulator end-effector behave as a Cartesian spring with six degrees of freedom.

Whereas a simple joint-based position controller might use the control law

$$\tau = K_pE + K_v\dot{E}, \tag{11.22}$$

where K_p and K_v are constant diagonal gain matrices and E is servo error defined as $\Theta_d - \Theta$, Salisbury suggests using

$$\tau = J^T(\Theta)K_{px}J(\Theta)E + K_v\dot{E}, \tag{11.23}$$

where K_{px} is the desired stiffness of the end-effector in Cartesian space. For a manipulator with six degrees of freedom, K_{px} is diagonal with the six values on the diagonal representing the three translational and three rotational stiffnesses that the end-effector is to exhibit. Essentially, through use of the Jacobian, a Cartesian stiffness has been transformed to a joint-space stiffness.

Force sensing

Force sensing allows a manipulator to detect contact with a surface and, using this sensation, to take some action. For example, the term **guarded move** is sometimes used to mean the strategy of moving under position control until a force is felt, then halting motion. Additionally, force sensing can be used to weigh objects that the manipulator lifts. This can be used as a simple check during a parts-handling operation—to ensure that a part was acquired or that the appropriate part was acquired.

Some commercially available robots come equipped with force sensors in the end-effector. These robots can be programmed to stop motion or take other action

when a force threshold is exceeded, and some can be programmed to weigh objects that are grasped in the end-effector.

BIBLIOGRAPHY

[1] M. Mason, "Compliance and Force Control for Computer Controlled Manipulators," M.S. Thesis, MIT AI Laboratory, May 1978.

[2] J. Craig and M. Raibert, "A Systematic Method for Hybrid Position/Force Control of a Manipulator," *Proceedings of the 1979 IEEE Computer Software Applications Conference*, Chicago, November 1979.

[3] M. Raibert and J. Craig, "Hybrid Position/Force Control of Manipulators," *ASME Journal of Dynamic Systems, Measurement, and Control*, June 1981.

[4] T. Lozano-Perez, M. Mason, and R. Taylor, "Automatic Synthesis of Fine-Motion Strategies for Robots," 1st International Symposium of Robotics Research, Bretton Woods, NH, August 1983.

[5] M. Mason, "Automatic Planning of Fine Motions: Correctness and Completeness," IEEE International Conference on Robotics, Atlanta, March 1984.

[6] M. Erdmann, "Using Backprojections for the Fine Motion Planning with Uncertainty," *The International Journal of Robotics Research*, Vol. 5, No. 1, 1986.

[7] S. Buckley, "Planning and Teaching Compliant Motion Strategies," Ph.D. Dissertation, Department of Electrical Engineering and Computer Science, MIT, January 1986.

[8] B. Donald, "Error Detection and Recovery for Robot Motion Planning with Uncertainty," Ph.D. Dissertation, Department of Electrical Engineering and Computer Science, MIT, July 1987.

[9] J.C. Latombe, "Motion Planning with Uncertainty: On the Preimage Backchaining Approach," in *The Robotics Review*, O. Khatib, J. Craig, and T. Lozano-Perez, Editors, MIT Press, Cambridge, MA, 1988.

[10] O. Khatib, "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 1, 1987.

[11] D. Whitney, "Force Feedback Control of Manipulator Fine Motions," Proceedings Joint Automatic Control Conference, San Francisco, 1976.

[12] S. Eppinger and W. Seering, "Understanding Bandwidth Limitations in Robot Force Control," *Proceedings of the IEEE Conference on Robotics and Automation*, Raleigh, NC, 1987.

[13] W. Townsend and J.K. Salisbury, "The Effect of Coulomb Friction and Stiction on Force Control," *Proceedings of the IEEE Conference on Robotics and Automation*, Raleigh, NC, 1987.

[14] N. Hogan, "Stable Execution of Contact Tasks Using Impedance Control," *Proceedings of the IEEE Conference on Robotics and Automation*, Raleigh, NC, 1987.

[15] N. Hogan and E. Colgate, "Stability Problems in Contact Tasks," in *The Robotics Review*, O. Khatib, J. Craig, and T. Lozano-Perez, Editors, MIT Press, Cambridge, MA, 1988.

[16] J.K. Salisbury, "Active Stiffness Control of a Manipulator in Cartesian Coordinates," 19th IEEE Conference on Decision and Control, December 1980.

[17] J.K. Salisbury and J. Craig, "Articulated Hands: Force Control and Kinematic Issues," *International Journal of Robotics Research*, Vol. 1, No. 1.

[18] S. Drake, "Using Compliance in Lieu of Sensory Feedback for Automatic Assembly," Ph.D. Thesis, Mechanical Engineering Department, MIT, September 1977.

[19] R. Featherstone, S.S. Thiebaut, and O. Khatib, "A General Contact Model for Dynamically-Decoupled Force/Motion Control," *Proceedings of the IEEE Conference on Robotics and Automation*, Detroit, 1999.

EXERCISES

11.1 [12] Give the natural constraints present for a peg of square cross section sliding into a hole of square cross section. Show your definition of $\{C\}$ in a sketch.

11.2 [10] Give the artificial constraints (i.e., the trajectory) you would suggest in order to cause the peg in Exercise 11.1 to slide further into the hole without jamming.

11.3 [20] Show that using the control law (11.14) with a system given by (11.9) results in the error-space equation

$$\ddot{e}_f + k_{v_f}\dot{e}_f + (k_{pf} + m^{-1}k_e)e_f = m^{-1}k_e f_{\text{dist}},$$

and, hence, that choosing gains to provide critical damping is possible only if the stiffness of the environment, k_e, is known.

11.4 [17] Given

$${}^{A}_{B}T = \begin{bmatrix} 0.866 & -0.500 & 0.000 & 10.0 \\ 0.500 & 0.866 & 0.000 & 0.0 \\ 0.000 & 0.000 & 1.000 & 5.0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

if the force–torque vector at the origin of $\{A\}$ is

$${}^{A}v = \begin{bmatrix} 0.0 \\ 2.0 \\ -3.0 \\ 0.0 \\ 0.0 \\ 4.0 \end{bmatrix},$$

find the 6×1 force–torque vector with reference point at the origin of $\{B\}$.

11.5 [17] Given

$${}^{A}_{B}T = \begin{bmatrix} 0.866 & 0.500 & 0.000 & 10.0 \\ -0.500 & 0.866 & 0.000 & 0.0 \\ 0.000 & 0.000 & 1.000 & 5.0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

if the force–torque vector at the origin of $\{A\}$ is

$${}^{A}v = \begin{bmatrix} 6.0 \\ 6.0 \\ 0.0 \\ 5.0 \\ 0.0 \\ 0.0 \end{bmatrix},$$

find the 6×1 force–torque vector with reference point at the origin of $\{B\}$.

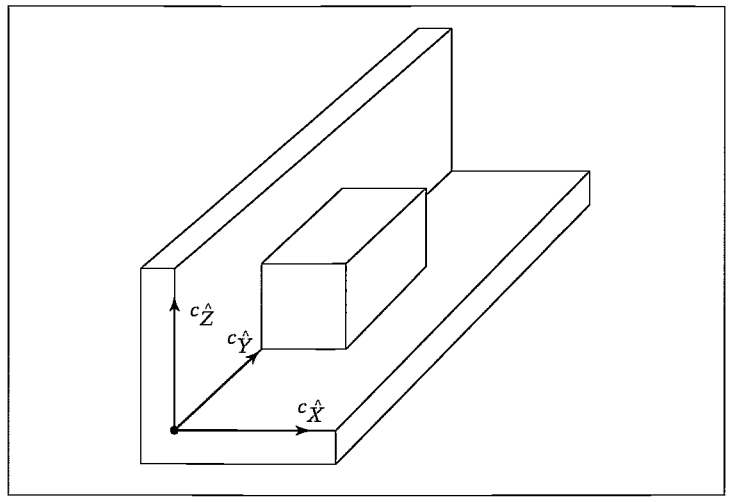


FIGURE 11.14: A block constrained by a floor below and a wall to the side.

11.6 [18] Describe in English how you accomplish the insertion of a book into a narrow crack between books on your crowded bookshelf.

11.7 [20] Give the natural and artificial constraints for the task of closing a hinged door with a manipulator. Make any reasonable assumptions needed. Show your definition of $\{C\}$ in a sketch.

11.8 [20] Give the natural and artificial constraints for the task of uncorking a bottle of champagne with a manipulator. Make any reasonable assumptions needed. Show your definition of $\{C\}$ in a sketch.

11.9 [41] For the stiffness servo system of Section 11.7, we have made no claim that the system is stable. Assume that (11.23) is used as the servo portion of a decoupled and linearized manipulator (so that the n joints appear as unit masses). Prove that the controller is stable for any K_v which is negative definite.

11.10 [48] For the stiffness servo system of Section 11.7, we have made no claim that the system is or can be critically damped. Assume that (11.23) is used as the servo portion of a decoupled and linearized manipulator (so that the n joints appear as unit masses). Is it possible to design a K_p that is a function of Θ and causes the system to be critically damped over all configurations?

11.11 [15] As shown in Fig. 11.14, a block is constrained below by a floor and to the side by a wall. Assuming this contacting situation is maintained over an interval of time, give the natural constraints that are present.

PROGRAMMING EXERCISE (PART 11)

Implement a Cartesian stiffness-control system for the three-link planar manipulator by using the control law (11.23) to control the simulated arm. Use the Jacobian written in frame {3}.

For the manipulator in position $\Theta = [60.0 \; -90.0 \; 30.0]$ and for K_{px} of the form

$$K_{px} = \begin{bmatrix} k_{\text{small}} & 0.0 & 0.0 \\ 0.0 & k_{\text{big}} & 0.0 \\ 0.0 & 0.0 & k_{\text{big}} \end{bmatrix},$$

simulate the application of the following static forces:

1. a 1-newton force acting at the origin of {3} in the $\hat{X}_3$ direction, and
2. a 1-newton force acting at the origin of {3} in the $\hat{Y}_3$ direction.

The values of k_{small} and k_{big} should be found experimentally. Use a large value of k_{big} for high stiffness in the $\hat{Y}_3$ direction and a low value of k_{small} for low stiffness in the $\hat{X}_3$ direction. What are the steady-state deflections in the two cases?