• Nonlinear Control
  – Joint space control
    + Decoupled control
      ♦ PID control in joint space
    + Centralized control
      ♦ Compute torque control
      ♦ Inverse dynamics control
  – Operational space control

• Reading Assignment for Next Class
  ♦ See http://www-slab.usc.edu/courses/CS545
Two Control Spaces

• Joint Space:
  – Solve two separate subproblems
   + Inverse kinematics to transform desired trajectories in operational space into joint space (including higher derivatives!)
   + Joint space controller tracks desired trajectories in joint space
   + Quality of control depends on quality of kinematics model (indirect control method)

• Operational Space:
  + Formulate the controller directly in operational space, e.g., a PD controller in Cartesian space.
  + Inverse kinematics is somehow included in the operational space controller
  + Quality of control is independent of kinematics if sensor measurements are taken directly in operational space (but this is rarely the case)
  + Controller can break if inverse kinematics is ill-defined.
Joint Space Control

\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \]

- **Independent (de-centralized) Joint Space Control**
  - Is appropriate when:
    + Coupling terms are negligible and can be treated as disturbances
    + Robot is really decoupled (i.e., like a set of 1DOF robots)
    + Gains can be chosen really high
    + No computational power exists

- **Dependent (centralized) Joint Space Control**
  - Is appropriate when:
    + Coupling terms cannot be neglected anymore
Independent Joint Space Control

- Negative Feedback Control in Joint Space
  - Coupling terms are treated as disturbances

- In order to deal with disturbances from coupling terms more effectively, an acceleration-based feedback term is often added (note that good acceleration signals are not easy to obtain from real sensors)
Independent Joint Space Control (cont’d)

• Feedforward Compensation in Joint Space

• What is the right choice of a de-centralized feedforward controller?
  – Need de-centralized inertial, damping, and spring term
Computation Torque
Feedforward Control

• A hybrid centralized–decentralized control strategy
  + Use de-centralized PID controller for stabilization
  + Use (centralized) inverse dynamics model to add feedforward command

\[
B(q)\dddot{q} + C(q, \dot{q})\ddot{q} + G(q) = \tau
\]
\[
B(q)\dddot{q} + C(q, \dot{q})\ddot{q} + G(q) =
\]
\[
B(q_d)\dddot{q}_d + C(q_d, \dot{q}_d)\ddot{q}_d + G(q_d) + K_P (q_d - q) + K_D (\dot{q}_d - \dot{q})
\]

In approximation:
\[
B(q)\dddot{e} + C(q, \dot{q})\ddot{e} + G(q)e = K_P e + K_D \dot{e}
\]
\[
B(q)\dddot{e} + (C(q, \dot{q}) - K_D) + (G(q) - K_P) = 0
\]

• The error dynamics forms a second order linear time variant dynamical system that can be stabilized with a suitable choice of the feedback gains (assuming the model is accurate)
Computed Torque
Feedforward Control (cont’d)

• Remarks:
  – Feedforward command is only based on DESIRED states
    + Higher derivatives of desired states are usually clean
  – Feedforward commands are not very accurate if the system deviates too much from the desired trajectory
  – Feedforward commands can be computed off-line if necessary (computational burden)
Nonlinear Centralized Control

• Decentralized and hybrid approaches are not really proper centralized nonlinear controllers since the negative feedback control law assumes decoupling

• Goal: Develop a nonlinear negative feedback controller

• Most common approach: Feedback linearization
A Simple Example of a Nonlinear Control Law

- PD Control with Gravity Compensation (S&S, Ch.6.5.1)
Proper Inverse Dynamics Control

\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau \]
\[ u = B(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + K_P(q_d - q) + K_D(\dot{q}_d - \dot{q}) \]

• Note:
  – Online computation of inverse dynamics is always required
  – Higher derivatives of real system are required (not always possible with reasonable accuracy and acceptable phase lags)
  – Perfect knowledge of the dynamics model is assumed
  – High servo loop frequencies are required
  – Further improvements: Robust Control (S&S 6.5.3)
Operational Space Control

• Joint space:
  + Negative feedback control requires inverse kinematics, then joint space error can be computed

• Operational space:
  + Use direct kinematics to transform joint space variables into operational space, and compute operational space error (much cheaper computation)
  – New problems
    + Motor command in operational space needs to be transformed into joint space
    + System dynamics in operational space needs to be considered
    + Usually computationally expensive
Operational Space Control Schemes

• Jacobian Inverse Control (S&S 6.6.1)
• Jacobian Transpose Control (S&S 6.6.1)
In the previous chapter, trajectory planning techniques have been presented which allow generating the reference inputs to the motion control system. Generally speaking, the problem of controlling a manipulator is to determine the time history of the generalized forces (forces or torques) to be developed by the joint actuators so as to guarantee execution of the commanded task while satisfying given transient and steady-state requirements. The task may regard either the execution of specified motions for a manipulator operating in free space, or the execution of specified motions and contact forces for a manipulator whose end effector is constrained by the environment. In view of problem complexity, the two aspects will be treated separately; first, motion control in free space, and then interaction control in constrained space. The problem of motion control of a manipulator is the topic of this chapter. A number of joint space control techniques are presented. These can be distinguished between decentralized control schemes, i.e., when the single manipulator joint is controlled independently of the others, and centralized control schemes, i.e., when the dynamic interaction effects between the joints are taken into account. Finally, as a premise to the interaction control problem, the basic features of operational space control schemes are illustrated.

6.1 THE CONTROL PROBLEM

Several techniques can be employed for controlling a manipulator. The technique followed, as well as the way it is implemented, may have a significant influence on the manipulator performance and then on the possible range of applications. For instance, the need for trajectory tracking control in the operational space may lead to hardware/software implementations which differ from those allowing point-to-point control where only reaching of the final position is of concern.

On the other hand, the manipulator mechanical design has an influence on the kind of control scheme utilized. For instance, the control problem of a Cartesian manipulator is substantially different from that of an anthropomorphic manipulator.

The driving system of the joints has also an effect on the type of control strategy used. If a manipulator is actuated by electric motors with reduction gears of high ratios,
the presence of gears tends to linearize system dynamics and thus to decouple the joints in view of the reduction of nonlinearity effects. The price to pay, however, is the occurrence of joint friction, elasticity and backlash that may limit system performance more than it is due to configuration-dependent inertia, Coriolis forces, and so forth. On the other hand, a robot actuated with direct drives eliminates the drawbacks due to friction, elasticity and backlash but the weight of nonlinearities and couplings between the joints becomes relevant. As a consequence, different control strategies have to be thought of to obtain high performance.

Without any concern to the specific type of mechanical manipulator, it is worth remarking that task specification (end-effector motion and forces) is usually carried out in the operational space, whereas control actions (joint actuator generalized forces) are performed in the joint space. This fact naturally leads to considering two kinds of general control schemes; namely, a joint space control scheme (Fig. 6.1) and an operational space control scheme (Fig. 6.2). In both schemes, the control structure has closed loops to exploit the good features provided by feedback, i.e., robustness to modeling uncertainties and reduction of disturbance effects. In general terms, the following considerations shall be made.

The joint space control problem is actually articulated in two subproblems. First, manipulator inverse kinematics is solved to transform motion requirements from the operational space into the joint space. Then, a joint space control scheme is designed that allows tracking of the reference inputs. However, this solution has the drawback that a joint space control scheme does not influence the operational space variables which are controlled in an open-loop fashion through the manipulator mechanical structure. It is then clear that any uncertainty of the structure (construction tolerance, lack of calibration, gear backlash, elasticity) or any imprecision on the knowledge of the end-effector position relative to an object to manipulate causes a loss of accuracy on the operational space variables.

The operational space control problem follows a global approach that requires a greater algorithmic complexity; notice that inverse kinematics is now embedded into the feedback control loop. Its conceptual advantage regards the possibility of acting directly on operational space variables; this is somewhat only a potential advantage, since measurement of operational space variables is often performed not directly, but through the evaluation of direct kinematics functions starting from measured joint space
variables.

On the above premises, in the following, joint space control schemes for manipulator motion in the free space are presented first. In the sequel, operational space control schemes will be illustrated which are logically at the basis of interaction control in constrained manipulator motion.

### 6.2 JOINT SPACE CONTROL

In Chapter 4, it was shown that the equations of motion of a manipulator in the absence of external end-effector forces and, for simplicity, of static friction (difficult to model accurately) are described by

\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + F_v \dot{q} + g(q) = \tau \]  \hspace{1cm} (6.1)

with obvious meaning of the symbols. To control the motion of the manipulator in free space means to determine the \( n \) components of generalized forces—torques for revolute joints, forces for prismatic joints—that allow execution of a motion \( q(t) \) so that

\[ q(t) = q_d(t) \]

as closely as possible, where \( q_d(t) \) denotes the vector of desired joint trajectory variables.

The generalized forces are supplied by the actuators through proper transmissions to transform the motion characteristics. Let \( q_m \) denote the vector of joint actuator displacements; the transmissions—assumed to be rigid and with no backlash—establish the following relationship

\[ K_r q = q_m, \]  \hspace{1cm} (6.2)

where \( K_r \) is an \( (n \times n) \) matrix, usually diagonal in the absence of induced motions, whose elements are much greater than unity.

In view of (6.2), if \( \tau_m \) denotes the vector of actuator driving torques, one can write

\[ \tau_m = K_r^{-1} \tau. \]  \hspace{1cm} (6.3)
FIGURE 6.3
Block scheme of manipulator with drives.

By observing that the diagonal elements of $B(q)$ are formed by constant terms and configuration-dependent terms (functions of sine and cosine for revolute joints), one can set

$$B(q) = \tilde{B} + \Delta B(q)$$

where $\tilde{B}$ is the diagonal matrix whose constant elements represent the resulting average inertia at each joint. Substituting (6.2)–(6.4) into (6.1) yields

$$\tau_m = K_r^{-1} \tilde{B} K_r^{-1} \ddot{q}_m + F_m \dot{q}_m + d$$

where

$$F_m = K_r^{-1} F_v K_r^{-1}$$

represents the matrix of viscous friction coefficients about the motor axes, and

$$d = K_r^{-1} \Delta B(q) K_r^{-1} \ddot{q}_m + K_r^{-1} C(q, \dot{q}) K_r^{-1} \dot{q}_m + K_r^{-1} g(q)$$

represents the contribution depending on the configuration.

As illustrated by the block scheme of Fig. 6.3, the system of manipulator with drives is actually constituted by two subsystems; one has $\tau_m$ as input and $q_m$ as output, the other has $q_m$, $\dot{q}_m$, and $\ddot{q}_m$ as inputs, and $d$ as output. The former is linear and decoupled, since each component of $\tau_m$ influences only the corresponding component
of \( q_m \). The latter is nonlinear and coupled, since it accounts for all those nonlinear and coupling terms of manipulator joint dynamics.

On the basis of the above scheme, several control algorithms can be derived with reference to the detail of knowledge of the dynamic model. The simplest approach that can be followed, in case of high gear reduction ratios and/or limited performance in terms of required velocities and accelerations, is to consider the component of the nonlinear interacting term \( d \) as a disturbance for the single joint servo.

The design of the control algorithm leads to a decentralized control structure, since each joint is considered independently of the others. The joint controller must guarantee good performance in terms of high disturbance rejection and enhanced trajectory tracking capabilities. The resulting control structure is substantially based on the error between the desired and actual output, while the input control torque at actuator \( i \) depends only on the error of output \( i \).

On the other hand, when large operational speeds are required to a direct-drive manipulator (\( K_v = I \)), the nonlinear coupling terms strongly influence system performance. Therefore, considering the effects of the components of \( d \) as a disturbance may generate large tracking errors. In this case, it is advisable to design control algorithms that take advantage of a detailed knowledge of manipulator dynamics so as to compensate for the nonlinear coupling terms of the model. In other words, it is necessary to eliminate the causes rather than to reduce the effects induced by them; that is, to generate compensating torques for the nonlinear terms in (6.7). This leads to centralized control algorithms that are based on the (partial or complete) knowledge of the manipulator dynamic model.

Nevertheless, it should be pointed out that these techniques still require the use of error contributions between the desired and the actual trajectory, no matter whether they are implemented in a feedback or in a feedforward fashion. This is a consequence of the fact that the considered dynamic model, even though a quite complex one, is anyhow an idealization of reality which does not include effects, such as joint Coulomb friction, gear backlash, dimension tolerance, and the simplifying assumptions in the model, e.g., link rigidity, and so on.

As pointed out above, the role of the drive system is relevant for the type of control chosen. In the case of decentralized control, the actuator will be described in terms of its own model as a velocity-controlled generator. Instead, in the case of centralized control, the actuator will have to generate torque contributions computed on the basis of a complete or reduced manipulator dynamic model; it will be then considered as a torque-controlled generator which is representative of the actuator/power amplifier system satisfying the above requirement.

### 6.3 INDEPENDENT JOINT CONTROL

The simplest control strategy that can be thought of is one that regards the manipulator as formed by \( n \) independent systems (the \( n \) joints) and controls each joint axis as a single-input/single-output system. Coupling effects between joints due to varying configurations during motion are treated as disturbance inputs.

In the case of interest, the system to control is joint drive \( i \) corresponding to
FIGURE 6.4
Block scheme of joint drive system.

the single-input/single-output system of the decoupled and linear part of the scheme in Fig. 6.3. The interaction with the other joints is described by component $i$ of the vector $d$ in (6.7).

Without loss of generality, the actuator is taken as a rotary electric dc motor. Hence, the block scheme of joint $i$ can be represented in the domain of the complex variable $s$ as in Fig. 6.4. In this scheme $\theta$ is the angular variable of the motor, $I$ is the average inertia reported to the motor axis ($I_i = \bar{b}_{ii} / k_t^2$), $R_a$ is the armature resistance (auto-inductance has been neglected), and $k_t$ and $k_v$ are respectively the torque and motor constants. Further, $G_v$ denotes the voltage gain of the power amplifier, and then the reference input is not the armature voltage $V_a$ but the input voltage $V_c$ of the amplifier; note that the amplifier bandwidth has been assumed to be much larger than that of the controlled system. In the scheme of Fig. 6.4, it has been assumed also that

$$F_m \ll \frac{k_v k_t}{R_a},$$

i.e., the mechanical viscous friction coefficient has been neglected with respect to the electrical friction coefficient.

The input/output transfer function of the motor can be written as

$$M(s) = \frac{k_m}{s(1 + sT_m)},$$  \hspace{1cm} (6.8)

where

$$k_m = \frac{1}{k_v}, \quad T_m = \frac{R_a I}{k_v k_t}$$

are respectively the velocity-to-voltage gain and time constant of the motor.

### 6.3.1 Feedback Control

To guide selection of the controller structure, start by noticing that an effective rejection of the disturbance $d$ on the output $\theta$ is ensured by:

---

1. Subscript $i$ has been dropped for notation compactness. Also, Laplace transforms of time-dependent functions are indicated by capital letters without specifying dependence on $s$.

2. A complete treatment of actuators is deferred to Chapter 8.
FIGURE 6.5
Block scheme of general independent joint control.

- a large value of the amplifier gain before the point of intervention of the disturbance,
- the presence of an integral action in the controller so as to cancel the effect of the gravitational component on the output at steady state (constant $\theta$).

These requisites clearly suggest the use of a proportional-integral (PI) control action in the forward path whose transfer function is

$$C(s) = K_c \frac{1 + sT_c}{s};$$

this yields zero error at steady state for a constant disturbance, and the presence of the real zero at $s = -1/T_c$ offers a stabilizing action. To improve dynamic performance, it is worth choosing the controller as a cascade of elementary actions with local feedback loops closed around the disturbance.

Besides closure of a position feedback loop, the most general solution is obtained by closing inner loops on velocity and acceleration. This leads to the scheme in Fig. 6.5, where $C_P(s)$, $C_V(s)$, and $C_A(s)$ respectively represent position, velocity, and acceleration controllers, and the inmost controller shall be of PI type as in (6.9) so as to obtain zero error at steady state for a constant disturbance. Further, $k_{TP}$, $k_{TV}$, and $k_{TA}$ are the respective transducer constants, and the amplifier gain has been embedded in the gain of the inmost controller. In the scheme of Fig. 6.5, notice that the disturbance torque $D$ has been suitably transformed into a disturbance voltage by the factor $R_{ca}/k_t$.

In the following, a number of possible solutions that can be derived from the general scheme of Fig. 6.5 are presented; at this stage, the issue arising from possible lack of measurement of physical variables is not considered yet. Three case studies are considered which differ in the number of active feedback loops.
Position Feedback. In this case, the control action is characterized by:

\[ C_P(s) = K_P \frac{1}{s}, \quad C_V(s) = 1, \quad C_A(s) = 1 \]

\[ k_{TV} = k_{TA} = 0. \]

The scheme of Fig. 6.6 shows that the transfer function of the forward path is

\[ P(s) = \frac{k_m K_P (1 + s T_P)}{s^2 (1 + s T_m)}, \]

while that of the return path is

\[ H(s) = k_{TP}. \]

A root locus analysis can be performed as a function of the gain of the position loop \( k_m K_P k_{TP} T_P / T_m \). Three situations are illustrated for the poles of the closed-loop system with reference to the relation between \( T_P \) and \( T_m \) (Fig. 6.7). Stability of the closed-loop feedback system imposes some constraints on the choice of the parameters of the PI controller. If \( T_P < T_m \), the system is inherently unstable (Fig. 6.7a). Then, it must be \( T_P > T_m \) (Fig. 6.7b). As \( T_P \) increases, the absolute value of the real part of the two roots of the locus tending towards the asymptotes increases too, and the system has faster time response. Hence, it is convenient to render \( T_P \gg T_m \) (Fig. 6.7c). In any case, the real part of the dominant poles cannot be less than \(-1/2T_m\).

The closed-loop input/output transfer function is

\[ \frac{\Theta(s)}{\Theta_T(s)} = \frac{1}{k_{TP} s^2 (1 + s T_m)} \left( \frac{1}{1 + k_m K_P k_{TP} (1 + s T_P)} \right) \]

which can be expressed in the form

\[ W(s) = \frac{1}{k_{TP} (1 + s T_P)} \left( \frac{2\zeta s}{\omega_n} + \frac{s^2}{\omega_n^2} \right) \left( 1 + s \tau \right), \]
where $\omega_n$ and $\zeta$ are respectively the natural frequency and damping ratio of the pair of complex poles and $-1/\tau$ locates the real pole. These values are assigned to define the joint drive dynamics as a function of the constant $T_P$; if $T_P > T_m$, then $1/\zeta \omega_n > T_P > \tau$ (Fig. 6.7b); if $T_P \gg T_m$ (Fig. 6.7c), for large values of the loop gain, then $\zeta \omega_n > 1/\tau \approx 1/T_P$ and the zero at $-1/T_P$ in the transfer function $W(s)$ tends to cancel the effect of the real pole.

The closed-loop disturbance/output transfer function is

$$\frac{\theta(s)}{D(s)} = \frac{s R_a}{k_l K_p k_{TP} (1 + s T_P)}, \quad (6.11)$$

which shows that it is worth increasing $K_p$ to reduce the effect of disturbance on the output during the transient. The function in (6.11) has two complex poles ($-\zeta \omega_n, \pm j \sqrt{1 - \zeta^2 \omega_n}$), a real pole ($-1/\tau$), and a zero at the origin. The zero is due to the PI controller and allows canceling the effects of gravity on the angular position when $\theta$ is a constant.

In Eq. (6.11), it can be recognized that the term $K_p k_{TP}$ is the reduction factor imposed by the feedback gain on the amplitude of the output due to disturbance; hence,
the quantity

\[ X_R = K_P k_{TP} \quad (6.12) \]

can be interpreted as the disturbance rejection factor, which in turn is determined by the gain \( K_P \). However, it is not advisable to increase \( K_P \) too much, because small damping ratios would result leading to unacceptable oscillations of the output. An estimate \( T_R \) of the output recovery time needed by the control system to recover the effects of the disturbance on the angular position can be evaluated by analyzing the modes of evolution of (6.11). Since \( \tau \approx T_P \), such estimate is expressed by

\[ T_R = \max \left\{ T_P, \frac{1}{\zeta \omega_n} \right\}. \quad (6.13) \]

**Position and Velocity Feedback.** In this case, the control action is characterized by:

\[ C_P(s) = K_P, \quad C_V(s) = K_V \frac{1 + sT_V}{s}, \quad C_A(s) = 1, \quad k_{TA} = 0. \]

To carry out a root locus analysis as a function of the velocity feedback loop gain, it is worth reducing the velocity loop in parallel to the position loop by following the usual rules for moving blocks. From the scheme in Fig. 6.8 the transfer function of the forward path is

\[ P(s) = \frac{k_m K_P K_V (1 + sT_V)}{s^2 (1 + sT_m)}, \]

while that of the return path is

\[ H(s) = k_{TP} \left( 1 + s \frac{k_{TV}}{K_P k_{TP}} \right). \]
FIGURE 6.9
Root locus for the position and velocity feedback control scheme.

The zero of the controller at \( s = -1/T_V \) can be chosen so as to cancel the effects of the real pole of the motor at \( s = -1/T_m \). Then, by setting

\[ T_V = T_m, \]

the poles of the closed-loop system move on the root locus as a function of the loop gain \( k_m K_V k_{TV} \), as shown in Fig. 6.9. By increasing the position feedback gain \( K_P \), it is possible to confine the closed-loop poles into a region of the complex plane with large absolute values of the real part. Then, the actual location can be established by a suitable choice of \( K_V \).

The closed-loop input/output transfer function is

\[ \frac{\Theta(s)}{\Theta_r(s)} = \frac{1}{k_{TP}} \frac{\frac{1}{k_{TP}}}{1 + \frac{s k_{TV}}{K_P k_{TP}} + \frac{s^2}{k_m K_P k_{TP} K_V}}, \quad (6.14) \]

which can be compared with the typical transfer function of a second-order system

\[ W(s) = \frac{1}{k_{TP}} \frac{\frac{1}{k_{TP}}}{1 + \frac{2\zeta s}{\omega_n} + \frac{s^2}{\omega_n^2}}, \quad (6.15) \]

It can be recognized that, with a suitable choice of the gains, it is possible to obtain any value of natural frequency \( \omega_n \) and damping ratio \( \zeta \). Hence, if \( \omega_n \) and \( \zeta \) are given as design requirements, the following relations can be found:

\[ K_V k_{TV} = \frac{2\zeta \omega_n}{k_m} \quad (6.16) \]

\[ K_P k_{TP} K_V = \frac{\omega_n^2}{k_m}. \quad (6.17) \]
FIGURE 6.10
Block scheme of position, velocity, and acceleration feedback control.

For given transducer constants \( k_{TP} \) and \( k_{TV} \), once \( K_V \) has been chosen to satisfy (6.16), the value of \( K_P \) is obtained from (6.17).

The closed-loop disturbance/output transfer function is

\[
\frac{\Theta(s)}{D(s)} = \frac{sR_a}{k_t K_P k_{TP} K_V (1 + sT_m)} (1 + \frac{sk_{TV}}{K_P k_{TP}} + \frac{s^2}{k_m K_P k_{TP} K_V}),
\]

which shows that the disturbance rejection factor is

\[ X_R = K_P k_{TP} K_V \]

and is fixed, once \( K_P \) and \( K_V \) have been chosen via (6.16) and (6.17). Concerning disturbance dynamics, the presence of a zero at the origin introduced by the PI, of a real pole at \( s = -1/T_m \), and of a pair of complex poles having real part \(-\zeta \omega_n\) should be noticed. Hence, in this case, an estimate of the output recovery time is given by the time constant

\[ T_R = \max \left\{ T_m, \frac{1}{\zeta \omega_n} \right\}, \]

which reveals an improvement with respect to the previous case in (6.13), since \( T_m \ll T_p \) and the real part of the dominant poles is not constrained by the inequality \( \zeta \omega_n < 1/2T_m \).

**Position, Velocity, and Acceleration Feedback.** In this case, the control action is characterized by:

\[ C_P(s) = K_P \quad C_V(s) = K_V \quad C_A(s) = K_A \frac{1 + sT_A}{s} \]
FIGURE 6.11
Root locus for the position, velocity, and acceleration feedback control scheme.

After some manipulation, the block scheme of Fig. 6.5 can be reduced to that of Fig. 6.10 where $G'(s)$ indicates the following transfer function

$$G'(s) = \frac{k_m}{1 + k_m K_A k_{TA}} \left( \frac{s T_m}{1 + k_m K_A k_{TA}} \frac{T_A}{T_m} \right) \left( 1 + \frac{s T_m}{1 + k_m K_A k_{TA}} \right).$$

The transfer function of the forward path is

$$P(s) = \frac{K_P K_Y K_A (1 + s T_A)}{s^2} G'(s),$$

while that of the return path is

$$H(s) = k_{TP} \left( 1 + \frac{s k_{TY}}{K_P k_{TP}} \right).$$

Also in this case, a suitable pole cancellation is worthy which can be achieved either by setting

$$T_A = T_m,$$

or by making

$$k_m K_A k_{TA} T_A \gg T_m \quad k_m K_A k_{TA} \gg 1.$$ 

The two solutions are equivalent as regards dynamic performance of the control system. In both cases, the poles of the closed-loop system are constrained to move on the root locus as a function of the loop gain $k_m K_P K_Y K_A / (1 + k_m K_A k_{TA})$ (Fig. 6.11). A close analogy with the previous scheme can be recognized, in that the resulting closed-loop system is again of second-order type.
The closed-loop input/output transfer function is

\[
\frac{\Theta(s)}{\Theta_r(s)} = \frac{1}{k_{TP}} \frac{k_{TV}}{K_P k_{TP}} + \frac{s^2(1 + k_m K_A k_{TA})}{k_m K_P k_{TP} K_V K_A},
\]

(6.21)

while the closed-loop disturbance/output transfer function is

\[
\frac{\Theta(s)}{D(s)} = -\frac{s R_a}{k_{TP} K_P K_V K_A (1 + s T_A)} \frac{s k_{TV}}{K_P k_{TP}} + \frac{s^2(1 + k_m K_A k_{TA})}{k_m K_P k_{TP} K_V K_A}.
\]

(6.22)

The resulting disturbance rejection factor is given by

\[X_R = K_P k_{TP} K_V K_A,\]

(6.23)

while the output recovery time is given by the time constant

\[T_R = \max \left\{ T_A, \frac{1}{\zeta \omega_n} \right\},\]

(6.24)

where \(T_A\) can be made less than \(T_m\), as pointed out above.

With reference to the transfer function in (6.15), the following relations can be established for design purposes, once \(\zeta, \omega_n,\) and \(X_R\) have been specified:

\[
2 K_P k_{TP} = \frac{\omega_n}{\zeta},
\]

(6.25)

\[k_m K_A k_{TA} = k_m X_R \frac{\omega_n^2}{\omega_n^2} - 1,
\]

(6.26)

\[K_P k_{TP} K_V K_A = X_R.
\]

(6.27)

For given \(k_{TP}\), \(k_{TV}\), and \(k_{TA}\), \(K_P\) is chosen to satisfy (6.25), \(K_A\) is chosen to satisfy (6.26), and then \(K_V\) is obtained from (6.27). Notice how admissible solutions for the controller typically require large values for the rejection factor \(X_R\). Hence, in principle, not only does the acceleration feedback allow achieving any desired dynamic behavior but, with respect to the previous case, it also allows prescribing the disturbance rejection factor as long as \(k_m X_R / \omega_n^2 > 1\).

In deriving the above control schemes, the issue of measurement of feedback variables was not considered explicitly. With reference to the typical position control servos that are implemented in industrial practice, there is no problem to measure position and velocity, while a direct measurement of acceleration, in general, either is not available or is too expensive to obtain. Therefore, for the scheme of Fig. 6.10, an indirect measure can be obtained by reconstructing acceleration from direct velocity.
measurement through a first-order filter (Fig. 6.12). The filter is characterized by a bandwidth $\omega_f = k_f$. By choosing this bandwidth wide enough, the effects due to measurement lags are not appreciable, and then it is feasible to take the acceleration filter output as the quantity to feed back. Some problem may occur concerning the noise superimposed on the filtered acceleration signal, though.

Resorting to a filtering technique may be useful when only the direct position measurement is available. In this case, by means of a second-order state variable filter, it is possible to reconstruct velocity and acceleration. However, the greater lags induced by the use of a second-order filter typically degrade the performance with respect to the use of a first-order filter, because of limitations imposed on the filter bandwidth by numerical implementation of the controller and filter.

Notice that the above derivation is based on an ideal dynamic model, i.e., when the effects of transmission elasticity as well as those of amplifier and motor electrical time constants are neglected. This implies that satisfaction of design requirements imposing large values of feedback gains may not be verified in practice, since the existence of unmodeled dynamics—such as electric dynamics, elastic dynamics due to non perfectly rigid transmissions, filter dynamics for the third scheme—might lead to degrading the system and eventually driving it to instability. In sum, the above solutions constitute design guidelines whose limits shall be emphasized with regard to the specific application.

### 6.3.2 Decentralized Feedforward Compensation

The above schemes have been derived according to the purpose of achieving good disturbance rejection. When the joint control servos are required to track reference trajectories with high values of speed and acceleration, the tracking capabilities of the scheme in Fig. 6.5 are unavoidably degraded. The adoption of a decentralized feedforward compensation allows reducing the tracking error. Therefore, if the reference inputs to the three control structures analyzed in the previous section are respectively modified into:

$$\Theta_r'(s) = \left( k_{TP} + \frac{s^2(1 + sT_m)}{k_m K_P (1 + sT_P)} \right) \Theta_d(s)$$

(6.28)
FIGURE 6.13
Block scheme of position feedback control with decentralized feedforward compensation.

FIGURE 6.14
Block scheme of position and velocity feedback control with decentralized feedforward compensation.

\[
\Theta'(s) = \left( k_{TP} + \frac{s k_{TV}}{K_P} + \frac{s^2}{k_m K_P K_V} \right) \Theta_d(s) \\
\Theta''(s) = \left( k_{TP} + \frac{s k_{TV}}{K_P} + \frac{(1 + k_m K_A k_{TA}) s^2}{k_m K_P K_V K_A} \right) \Theta_d(s),
\]

(6.29)  
(6.30)

tracking of the desired joint position trajectory is achieved, if not for the effect of disturbances. Notice that computing time derivatives of the desired trajectory is not a problem, once \( \Theta_d(t) \) is known analytically. The tracking control schemes, resulting from simple manipulation of Eqs. (6.28), (6.29), and (6.30) are reported respectively in Figs. 6.13, 6.14, and 6.15, where \( M(s) \) indicates the motor transfer function in (6.8).

All the solutions allow tracking of the input trajectory within the range of validity
and linearity of the employed models. It is worth noticing that, as the number of nested feedback loops increases, a less accurate knowledge of the system model is required to perform feedforward compensation. In fact, $T_m$ and $k_m$ are required for the scheme of Fig. 6.13, only $k_m$ is required for the scheme of Fig. 6.14, and $k_m$ again—but with reduced weight—for the scheme of Fig. 6.15.

It is worth recalling that *perfect* tracking can be obtained only on the assumption of exact matching of the controller and feedforward compensation parameters with the process parameters, as well as of exact modeling and linearity of the physical system. Deviations from the ideal values cause a performance degradation that shall be analyzed case by case.

The presence of saturation blocks in the schemes of Figs. 6.13, 6.14, and 6.15 is to be intended as intentional nonlinearities whose function is to limit relevant physical quantities during transients; the greater the number of feedback loops, the greater the number of quantities that can be limited (velocity, acceleration, and motor voltage). To this purpose, notice that trajectory tracking is obviously lost whenever any of the above quantities saturates. This situation often occurs for industrial manipulators required to execute point-to-point motions; in this case, there is less concern about the actual trajectories followed, and the actuators are intentionally taken to operate at the current limits so as to realize the fastest possible motions.

After simple block reduction on the above schemes, it is possible to determine equivalent control structures that utilize position feedback only and *regulators with standard actions*. It should be emphasized that the two solutions are equivalent in terms of disturbance rejection and trajectory tracking. However, tuning of regulator parameters is less straightforward, and the elimination of inner feedback loops prevents the possibility of setting saturations on velocity and/or acceleration. The control structures equivalent to those of Figs. 6.13, 6.14, and 6.15 are illustrated in Figs. 6.16, 6.17, and 6.18, respectively; control actions of PI, PID, and PIDD² type are illustrated which are respectively equivalent to the cases of: position feedback; position and velocity...
feedback; position, velocity, and acceleration feedback.

The above schemes can incorporate the typical structure of the controllers actually implemented in the control architectures of industrial robots. In these systems it is important to choose the largest possible gains so that model inaccuracy and interaction terms do not appreciably affect positions of the single joints. As pointed out above, the upper limit on the gains is imposed by all those factors that have not been modeled, such as implementation of discrete-time controllers in lieu of the continuous-time controllers analyzed in theory, presence of finite sampling time, neglected dynamic effects (e.g., joint elasticity, structural resonance, finite transducer bandwidth), and sensor noise. In
fact, the influence of such factors in the implementation of the above controllers may cause a severe system performance degradation for much too large values of feedback gains.

6.4 COMPUTED TORQUE FEEDFORWARD CONTROL

Define the tracking error $e(t) = \theta_d(t) - \theta(t)$. With reference to the most general scheme (Fig. 6.18), the output of the PIDD$^2$ regulator can be written as

$$a_2 \ddot{e} + a_1 \dot{e} + a_0 e + a_{-1} \int e(\zeta) d\zeta$$

which describes the time evolution of the error. The constant coefficients $a_2, a_1, a_0, a_{-1}$ are determined by the particular solution adopted. Summing the contribution of the feedforward actions and of the disturbance to this expression yields

$$\frac{1}{k_m} \dot{\theta}_d + \frac{T_m}{k_m} \ddot{\theta}_d - \frac{R_a}{k_t} d,$$

where

$$\frac{T_m}{k_m} = \frac{IR_a}{k_t}, \quad k_m = \frac{1}{k_v}.$$

The input to the motor (Fig. 6.5) has then to satisfy the following equation

$$a_2 \ddot{e} + a_1 \dot{e} + a_0 e + a_{-1} \int e(\zeta) d\zeta + \frac{T_m}{k_m} \ddot{\theta}_d + \frac{1}{k_m} \dot{\theta}_d - \frac{R_a}{k_t} d = \frac{T_m}{k_m} \ddot{\theta} + \frac{1}{k_m} \dot{\theta}.$$
With a suitable change of coefficients, this can be rewritten as

\[ a_2' \ddot{e} + a_1' \dot{e} + a_0' e + a_{-1}' \int e(\zeta) d\zeta = \frac{R_a}{k_t} d. \]

This equation describes the error dynamics and shows that any physically executable trajectory is asymptotically tracked only if the disturbance term \( d(t) = 0 \). With the term physically executable it is meant that the saturation limits on the physical quantities, e.g., current and voltage in electric motors, are not violated in the execution of the desired trajectory.

The presence of the term \( d(t) \) causes a tracking error whose magnitude is reduced as much as the disturbance frequency content is located off to the left of the lower limit of the bandwidth of the error system. The disturbance/error transfer function is given by

\[ \frac{E(s)}{D(s)} = \frac{\frac{R_a}{k_t} s}{a_2's^3 + a_1's^2 + a_0's + a_{-1}'}, \]

and thus the adoption of loop gains which are not realizable for the above discussed reasons is often required.

Nevertheless, even if the term \( d(t) \) has been introduced as a disturbance, its expression is given by (6.7). It is then possible to add a further term to the previous feedforward actions which is able to compensate the disturbance itself rather than its effects. In other words, by taking advantage of model knowledge, the rejection effort of an independent joint control scheme can be lightened with notable simplification from the implementation viewpoint.

Let \( q_d(t) \) be the desired joint trajectory and \( q_{md}(t) \) the corresponding actuator trajectory as in (6.2). By adopting an inverse model strategy, the feedforward action \( R_a K_t^{-1} d \) can be introduced with

\[ d_d = K_r^{-1} \Delta B(q_d) K_r^{-1} \dot{q}_{md} + K_r^{-1} C(q_d, \dot{q}_d) K_r^{-1} \dot{q}_{md} + K_r^{-1} g(q_d), \quad (6.31) \]

where \( R_a \) and \( K_t \) denote the diagonal matrices of armature resistances and torque constants of the actuators. This action tends to compensate the actual disturbance expressed by (6.7) and in turn allows the control system to operate in a better condition.

This solution is illustrated in the scheme of Fig. 6.19, which conceptually describes the control system of a manipulator with computed torque control. The feedback control system is representative of the \( n \) independent joint control servos; it is decentralized, since controller \( i \) elaborates references and measurements that refer to single joint \( i \). The interactions between the various joints, expressed by \( d \), are compensated by a centralized action whose function is to generate a feedforward action that depends on the joint references as well as on the manipulator dynamic model. This action compensates the nonlinear coupling terms due to inertial, Coriolis, centrifugal, and gravitational forces that depend on the structure and, as such, vary during manipulator motion.

Although the residual disturbance term \( \tilde{d} = d_d - d \) vanishes only in the ideal case of perfect tracking \( (q = q_d) \) and exact dynamic modeling, \( \tilde{d} \) is representative of
interaction disturbances of considerably reduced magnitude with respect to \( d \). Hence, the computed torque technique has the advantage to alleviate the disturbance rejection task for the feedback control structure and in turn allows limited gains. Notice that expression (6.31) in general imposes a computationally demanding burden on the centralized part of the controller. Therefore, in those applications where the desired trajectory is generated in real time with regard to hetroceptive sensory data and commands from higher hierarchical levels of the robot control architecture\(^3\), on-line computation of the centralized feedforward action may require too much time\(^4\).

Since the actual controller is to be implemented on a computer with a finite sampling time, torque computation has to be carried out during this interval of time; in order not to degrade dynamic system performance, typical sampling times are of the order of the millisecond.

Therefore, it may be worth performing only a partial feedforward action so as to compensate those terms of (6.31) that give the most relevant contributions during manipulator motion. Since inertial and gravitational terms dominate velocity-dependent terms (at operational joint speeds not greater than a few radians per second), a partial compensation can be achieved by computing only the gravitational torques and the inertial torques due to the diagonal elements of the inertia matrix. In this way, only the terms depending on the global manipulator configuration are compensated while those deriving from motion interaction with the other joints are not.

Finally, it should be pointed out that, for repetitive trajectories, the above com-

\(^3\) See also Chapter 9.

\(^4\) In this regard, the problem of real-time computation of compensating torques can be solved by resorting to efficient recursive formulations of manipulator inverse dynamics, such as the Newton-Euler algorithm presented in Chapter 4.
pensating contributions can be computed off-line and properly stored on the basis of a trade-off solution between memory capacity and computational requirements of the control architecture.

6.5 CENTRALIZED CONTROL

In the previous sections several techniques have been discussed that allow designing independent joint controllers. These are based on a single-input/single-output approach, since interaction and coupling effects between the joints have been considered as disturbances acting on each single joint drive system. However, as shown by the dynamic model (6.1), the manipulator is not a set of \( n \) decoupled systems but it is a multivariable system with \( n \) inputs (joint torques) and \( n \) outputs (joint positions) interacting between them by means of nonlinear relations.

In order to follow a methodological approach which is consistent with control design, it is necessary to treat the control problem in the context of nonlinear multivariable systems. This approach will obviously account for the manipulator dynamic model and lead to finding nonlinear centralized control laws, whose implementation is needed for high manipulator dynamic performance. On the other hand, the above computed torque control can be interpreted in this framework, since it provides a model-based nonlinear control term to enhance trajectory tracking performance. Notice, however, that this action is inherently performed off-line, as it is computed on the time history of the desired trajectory and not of the actual one.

For the following derivation, it is worth rewriting the mathematical model of the manipulator with drives in a more suitable form. The manipulator is described by Eq. (6.1)

\[
B(q)\ddot{\mathbf{q}} + C(q, \dot{\mathbf{q}})\dot{\mathbf{q}} + F_v \ddot{\mathbf{q}} + g(q) = \tau,
\]

while the transmissions are described by Eq. (6.2)

\[
K_r \mathbf{q} = \mathbf{q}_m.
\]

With reference to (6.3) and the block scheme of Fig. 6.4, the \( n \) driving systems can be described in compact matrix form by the equations:

\[
K_r^{-1} \tau = K_i i_a \tag{6.32}
\]

\[
v_a = R_a i_a + K_v \dot{q}_m \tag{6.33}
\]

\[
v_c = G_v v_c \tag{6.34}
\]

In (6.32), \( K_i \) is the diagonal matrix of torque constants and \( i_a \) is the vector of armature currents of the \( n \) motors; in (6.33), \( v_a \) is the vector of armature voltages, \( R_a \) is the diagonal matrix of armature resistances, and \( K_v \) is the diagonal matrix of voltage constants of the \( n \) motors; in (6.34), \( G_v \) is the diagonal matrix of gains of the \( n \) amplifiers and \( v_c \) is the vector of control voltages of the \( n \) servomotors.

On reduction of (6.1), (6.2), (6.32), (6.33), and (6.34), the dynamic model of the system given by the manipulator and drives is described by

\[
B(q)\ddot{\mathbf{q}} + C(q, \dot{\mathbf{q}})\dot{\mathbf{q}} + F_v \ddot{\mathbf{q}} + g(q) = \mathbf{u} \tag{6.35}
\]
where the following positions have been made:

\[
F = F_v + K_r K_t R_a^{-1} K_v K_r \\
u = K_r K_t R_a^{-1} G_v v_c.
\]  

(6.36) \hfill (6.37)

In (6.36), \(F\) is the diagonal matrix accounting for all viscous (mechanical and electrical) damping terms, and, in (6.37), \(u\) is the vector which is taken as control input to the system. Notice that the actual torques that determine the motion of the system of manipulator with drives can be obtained by subtracting to (6.37) the term \(K_r K_t R_a^{-1} K_v K_r \dot{q}\) due to electrical friction. The overall system is then \textit{voltage-controlled} and the corresponding block scheme is illustrated in Fig. 6.20.

If the actuators have to provide torque contributions computed on the basis of a complete or reduced manipulator model, the design of \(u\) in (6.35) depends on the matrices \(K_t\), \(K_v\), and \(R_a\) of the motors, which are influenced by the operating conditions. To reduce sensitivity to parameter variations, it is worth considering driving systems characterized by a torque (current) control rather than by a voltage control. In this case the actuators behave as \textit{torque-controlled generators}; Eq. (6.33) becomes meaningless and Eq. (6.34) is replaced with

\[
i_a = G_i v_c.
\]

(6.34')

which gives a proportional relation between the armature currents \(i_a\) (and thus the torques) and the control voltages \(v_c\) established by the constant matrix \(G_i\). As a consequence, Eqs. (6.36) and (6.37) become

\[
F = F_v \\
u = K_r K_t G_i v_c = \tau,
\]

(6.36') \hfill (6.37')

which show a reduced dependence of \(u\) and \(F\) on the motor parameters. The overall system is now \textit{torque-controlled} and the resulting block scheme is illustrated in Fig. 6.21.

In the remainder, the problem of finding control laws \(u\) that ensure a given performance for the system of manipulator with drives is considered. Since Eq. (6.37') can be considered as a constant proportional relation between \(v_c\) and \(u\), the centralized control algorithms that follow directly refer to the generation of control torques \(u\).
6.5.1 PD Control with Gravity Compensation

Let a constant equilibrium posture be assigned for the system as the vector of desired joint variables \( q_d \). It is desired to find the structure of the controller which ensures global asymptotic stability of the above posture.

The determination of the control input which stabilizes the system around the equilibrium posture is based on the Lyapunov direct method.

Take the vector \( [\ddot{q}^T \ \dot{q}^T]^T \) as the system state, where

\[
\ddot{q} = q_d - q
\]  

(6.38)

represents the error between the desired and the actual posture. Choose the following positive definite quadratic form as Lyapunov function candidate

\[
V(\dot{q}, \ddot{q}) = \frac{1}{2} \dot{q}^T B(q) \dot{q} + \frac{1}{2} \ddot{q}^T K_P \ddot{q} > 0 \quad \forall \dot{q}, \ddot{q} \neq 0
\]  

(6.39)

where \( K_P \) is an \((n \times n)\) symmetric positive definite matrix. An energy-based interpretation of (6.39) reveals a first term expressing the system kinetic energy and a second term expressing the potential energy stored in the system of equivalent stiffness \( K_P \) provided by the \( n \) position feedback loops.

Differentiating (6.39) with respect to time, and recalling that \( q_d \) is constant, yields

\[
\dot{V} = \dot{q}^T B(q) \dot{q} + \frac{1}{2} \ddot{q}^T \dot{B}(q) \dot{q} - \dot{q}^T K_P \ddot{q}.
\]  

(6.40)

Solving (6.35) for \( B \ddot{q} \) and substituting it in (6.40) gives

\[
\dot{V} = \frac{1}{2} \dot{q}^T (\ddot{B}(q) - 2C(q, \dot{q})) \dot{q} - \dot{q}^T F \ddot{q} + \dot{q}^T (u - g(q) - K_P \ddot{q}).
\]  

(6.41)

The first term on the right-hand side is null since the matrix \( N = \ddot{B} - 2C \) satisfies Eq. (4.48). The second term is negative definite. Then, the choice

\[
u = g(q) + K_P \ddot{q},
\]  

(6.42)

describing a controller with compensation of gravitational terms and a proportional action, leads to a negative semi-definite \( \dot{V} \) since

\[
\dot{V} = 0 \quad \dot{q} = 0, \forall \ddot{q}.
\]
This result can be obtained also by taking the control law

\[ u = g(q) + K_P \ddot{q} - K_D \dot{q}, \quad (6.43) \]

with \( K_D \) positive definite, corresponding to a nonlinear compensation action of gravitational terms with a linear proportional-derivative (PD) action. In fact, substituting (6.43) into (6.41) gives

\[ \dot{V} = -\dot{q}^T (F + K_D) \dot{q}, \quad (6.44) \]

which reveals that the introduction of the derivative term causes an increase of the absolute values of \( \dot{V} \) along the system trajectories, and then it gives an improvement of system time response. Notice that the inclusion of a derivative action in the controller, as in (6.43), is crucial when direct-drive manipulators are considered. In that case, in fact, mechanical viscous damping is practically null, and current control does not allow exploiting the electrical viscous damping provided by voltage-controlled actuators.

According to the above, the function candidate \( V \) decreases as long as \( \dot{q} \neq 0 \) for all system trajectories. It can be shown that the system reaches an equilibrium posture. To find such posture, notice that \( \dot{V} \equiv 0 \) only if \( \dot{q} \equiv 0 \). System dynamics under control (6.43) is given by

\[ B(q) \ddot{q} + C(q, \dot{q}) \dot{q} + F \dot{q} + g(q) = g(q) + K_P \ddot{q} - K_D \dot{q}. \quad (6.45) \]

At the equilibrium \( (\dot{q} \equiv 0, \ddot{q} \equiv 0) \) it is

\[ K_P \ddot{q} = 0, \quad (6.46) \]

and then

\[ \ddot{q} = q_d - q \equiv 0 \]

is the sought equilibrium posture. The above derivation rigorously shows that any manipulator equilibrium posture is globally asymptotically stable under a controller.
with a PD linear action and a nonlinear gravity compensating action. Stability is ensured for any choice of $K_P$ and $K_D$, as long as these are positive definite matrices. The resulting block scheme is shown in Fig. 6.22.

The control law requires the on-line computation of the term $g(q)$. If compensation is imperfect, the above discussion does not lead to the same result; this aspect will be revisited later with reference to robustness of controllers performing nonlinear compensation.

### 6.5.2 Inverse Dynamics Control

Consider now the problem of tracking a joint space trajectory. The reference framework is that of control of nonlinear multivariable systems. The dynamic model of an $n$-joint manipulator is expressed by (6.35) which can be rewritten as

$$B(q)\ddot{q} + n(q, \dot{q}) = u,$$  \hspace{1cm} (6.47)

where for simplicity it has been set

$$n(q, \dot{q}) = C(q, \dot{q})\dot{q} + F\dot{q} + g(q).$$  \hspace{1cm} (6.48)

The approach that follows is founded on the idea to find a control vector $u$, as a function of the system state, which is capable to realize an input/output relationship of linear type; in other words, it is desired to perform not a local linearization but a global linearization of system dynamics obtained by means of a nonlinear state feedback. The possibility of finding such a linearizing controller is guaranteed by the particular form of system dynamics. In fact, Eq. (6.47) is linear in the control $u$ and has a full-rank matrix $B(q)$ which can be inverted for any manipulator configuration.

Taking the control $u$ as a function of the manipulator state in the form

$$u = B(q)y + n(q, \dot{q})$$  \hspace{1cm} (6.49)

leads to the system described by

$$\ddot{q} = y,$$

where $y$ represents a new input vector whose expression is to be determined yet; the resulting block scheme is shown in Fig. 6.23. The nonlinear control law in (6.49) is termed inverse dynamics control since it is based on the computation of manipulator inverse dynamics. The system under control (6.49) is linear and decoupled with respect to the new input $y$. In other words, the component $y_i$ influences, with a double integrator relationship, only the joint variable $q_i$, independently of the motion of the other joints.

In view of the choice (6.49), the manipulator control problem is reduced to that of finding a stabilizing control law $y$. To this purpose, the choice

$$y = -K_Pq - K_D\dot{q} + r$$  \hspace{1cm} (6.50)

leads to the system of second-order equations

$$\ddot{q} + K_D\dot{q} + K_Pq = r$$  \hspace{1cm} (6.51)
which, on the assumption of positive definite matrices $K_P$ and $K_D$, is asymptotically stable. Choosing $K_P$ and $K_D$ as diagonal matrices of the type

$$K_P = \text{diag}\{\omega_{n1}^2, \ldots, \omega_{nn}^2\} \quad K_D = \text{diag}\{2\zeta_1\omega_{n1}, \ldots, 2\zeta_n\omega_{nn}\},$$

gives a decoupled system. The reference component $r_i$ influences only the joint variable $q_i$ with a second-order input/output relationship characterized by the natural frequency $\omega_{ni}$ and a damping ratio $\zeta_i$.

Given any desired trajectory $q_d(t)$, tracking of this trajectory for the output $q(t)$ is ensured by choosing

$$r = \ddot{q}_d + K_D\dot{q}_d + K_P q_d.$$  \hfill (6.52)

In fact, substituting (6.52) into (6.51) gives the homogeneous second-order differential equation

$$\ddot{\tilde{q}} + K_D\dot{\tilde{q}} + K_P\tilde{q} = 0$$  \hfill (6.53)

expressing the dynamics of position error (6.38) while tracking the given trajectory. Such error occurs only if $\tilde{q}(0)$ and/or $\dot{\tilde{q}}(0)$ are different from zero and converges to zero with a speed depending on the matrices $K_P$ and $K_D$ chosen.

The resulting block scheme is illustrated in Fig. 6.24, in which two feedback loops are represented; an inner loop based on the manipulator dynamic model and an outer loop operating on the tracking error. The function of the inner loop is to obtain a linear and decoupled input/output relationship, whereas the outer loop is required to stabilize the overall system. The controller design for the outer loop is simplified since it operates on a linear and time-invariant system. Notice that the implementation of this control scheme requires computation of the inertia matrix $B(q)$ and of the vector of Coriolis, centrifugal, gravitational, and damping terms $n(q, \dot{q})$ in (6.48). Differently from computed torque control, these terms must be computed on line since control is now based on nonlinear feedback of the current system state, and thus it is not possible to precompute the terms off line as for the previous technique.

The above technique of nonlinear compensation and decoupling is very attractive from a control viewpoint since the nonlinear and coupled manipulator dynamics is replaced with $n$ linear and decoupled second-order subsystems. Nonetheless, this technique is based on the assumption of perfect cancellation of dynamic terms, and
FIGURE 6.24
Block scheme of joint space inverse dynamics control.

then it is quite natural to raise questions about sensitivity and robustness problems due to unavoidably imperfect compensation.

Implementation of inverse dynamics control laws indeed requires that parameters of the system dynamic model are accurately known and the complete equations of motion are computed in real time. These conditions are difficult to verify in practice. On one hand, the model is usually known with a certain degree of uncertainty due to imperfect knowledge of manipulator mechanical parameters, existence of unmodeled dynamics, and model dependence on end-effector payloads not exactly known and thus not perfectly compensated. On the other hand, inverse dynamics computation is to be performed at sampling times of the order of the millisecond so as to ensure that the assumption of operating in the continuous time domain is realistic. This may pose severe constraints on the hardware/software architecture of the control system. In such cases, it may be advisable to lighten the computation of inverse dynamics and compute only the dominant terms.

On the basis of the above remarks, from an implementation viewpoint, compensation may be imperfect both for model uncertainty and for the approximations made in on-line computation of inverse dynamics. In the following, two control techniques are presented which are aimed at counteracting the effects of imperfect compensation. The first one consists of the introduction of an additional term to an inverse dynamics controller which provides robustness to the control system by counteracting the effects of the approximations made in on-line computation of inverse dynamics. The second one adapts the parameters of the model used for inverse dynamics computation to those of the true manipulator dynamic model.

6.5.3 Robust Control

In the case of imperfect compensation, it is reasonable to assume in (6.47) a control
6.6 OPERATIONAL SPACE CONTROL

In all the above control schemes, it was always assumed that the desired trajectory is available in terms of the time sequence of the values of joint position, velocity, and acceleration. Accordingly, the error for the control schemes was expressed in the joint space.

As often pointed out, motion specifications are usually assigned in the operational space, and then an inverse kinematics algorithm has to be utilized to transform operational space references into the corresponding joint space references. The process of kinematic inversion has an increasing computational load when, besides inversion of direct kinematics, also inversion of first-order and second-order differential kinematics is required to transform the desired time history of end-effector position, velocity and acceleration into the corresponding quantities at the joint level. It is for this reason that current industrial robot control systems compute the joint positions through kinematics inversion, and then perform a numerical differentiation to compute velocities and accelerations.

A different approach consists of considering control schemes developed directly in the operational space. If the motion is specified in terms of operational space variables, the measured joint space variables can be transformed into the corresponding operational space variables through direct kinematics relations. Comparing the desired input with the reconstructed variables allows designing feedback control loops where trajectory inversion is replaced with a suitable coordinate transformation embedded in the feedback loop.

All operational space control schemes present considerable computational requirements, in view of the necessity to perform a number of computations in the feedback loop which are somewhat representative of inverse kinematics functions. With reference to a numerical implementation, the presence of a computationally demanding load requires sampling times that may lead to degrading the performance of the overall control system.

In the face of the above limitations, it is worth presenting operational space control schemes, whose utilization becomes necessary when the problem of controlling interaction between the manipulator and the environment is of concern. In fact, joint space control schemes suffice only for motion control in the free space. When the manipulator's end effector is constrained by the environment, e.g., in the case of end-effector in contact with an elastic environment, it is necessary to control both positions and contact forces and it is convenient to refer to operational space control schemes. Hence, below some solutions are presented; these are worked out for motion control, but they constitute the premise for the interaction control strategies that will be illustrated in the next chapter.

6.6.1 General Schemes

As pointed out above, operational space control schemes are based on a direct comparison of the inputs, specifying operational space trajectories, with the measurements of the corresponding manipulator outputs. It follows that the control system shall incor-
porate some actions that allow passing from the operational space, in which the error is specified, to the joint space, in which control generalized forces are developed.

A possible control scheme that can be devised is the so-called Jacobian inverse control (Fig. 6.29). In this scheme, the end-effector location in the operational space is compared with the corresponding desired quantity, and then an operational space deviation \( \Delta x \) can be computed. Assuming that this deviation is sufficiently small for a good control system, \( \Delta x \) can be transformed into a corresponding joint space deviation \( \Delta q \) through the inverse of the manipulator Jacobian. Then, the control input generalized forces can be computed on the basis of this deviation through a suitable feedback matrix gain. The result is a presumbale reduction of \( \Delta q \) and in turn of \( \Delta x \).

In other words, the Jacobian inverse control leads to an overall system that intuitively behaves like a mechanical system with a generalized \( n \)-dimensional spring in the joint space, whose constant stiffness is determined by the feedback matrix gain. The role of such system is to take the deviation \( \Delta q \) to zero. If the matrix gain is diagonal, the generalized spring corresponds to \( n \) independent elastic elements, one for each joint.

A conceptually analogous scheme is the so-called Jacobian transpose control (Fig. 6.30). In this case, the operational space error is treated first through a matrix gain. The output of this block can then be considered as the elastic force generated by a generalized spring whose function in the operational space is that to reduce or to cancel the position deviation \( \Delta x \). In other words, the resulting force drives the end effector along a direction so as to reduce \( \Delta x \). This operational space force has then to be transformed into the joint space generalized forces, through the transpose of the Jacobian, so as to realize the described behavior.

Both Jacobian inverse and transpose control schemes have been derived in an intuitive fashion. Hence, there is no guarantee that such schemes are effective in terms of stability and trajectory tracking accuracy. These problems can be faced by presenting two mathematical solutions below, which will be shown to be substantially equivalent to the above schemes.

### 6.6.2 PD Control with Gravity Compensation

By analogy with joint space stability analysis, given a constant end-effector location \( x_d \), it is desired to find the control structure so that the operational space error

\[
\tilde{x} = x_d - x
\]  

(6.98)
tends asymptotically to zero. Choose the following positive definite quadratic form as a Lyapunov function candidate

\[ V(\dot{q}, \ddot{x}) = \frac{1}{2} \dot{q}^T \dot{B}(q) \dot{q} + \frac{1}{2} \ddot{x}^T K_P \ddot{x} > 0 \quad \forall \dot{q}, \ddot{x} \neq 0, \quad (6.99) \]

with \( K_P \) a symmetric positive definite matrix. Differentiating (6.99) with respect to time gives

\[ \dot{V} = \dot{q}^T \dot{B}(q) \dot{q} + \frac{1}{2} \dddot{x}^T K_P \ddot{x}. \]

Since \( \dddot{x}_d = 0 \), in view of (3.37) it is

\[ \dddot{x} = -J_A(q) \ddot{q} \]

and then

\[ \dot{V} = \dot{q}^T \dot{B}(q) \dot{q} + \frac{1}{2} \dddot{x}^T K_P \ddot{x}. \quad (6.100) \]

By recalling the expression of joint space manipulator dynamic model (6.35) and property (4.48), Eq. (6.100) becomes

\[ \dot{V} = -\dot{q}^T F \dot{q} + \dot{q}^T (u - g(q) - J_A^T(q) K_P \ddot{x}). \quad (6.101) \]

This equation suggests the structure of the controller; in fact, by choosing the control law

\[ u = g(q) + J_A^T(q) K_P \ddot{x} - J_A^T(q) K_D J_A(q) \dot{q}, \quad (6.102) \]

with \( K_D \) positive definite, Eq. (6.101) becomes

\[ \dot{V} = -\dot{q}^T F \dot{q} - \dot{q}^T J_A^T(q) K_D J_A(q) \dot{q}. \quad (6.103) \]

As can be seen from Fig. 6.31, the resulting block scheme reveals an analogy with the scheme of Fig. 6.30. Control law (6.102) performs a nonlinear compensating action of joint space gravitational forces and an operational space linear PD control action. The
last term has been introduced to enhance system damping; in particular, if measurement of \( \ddot{x} \) is deduced from that of \( \dot{q} \), one can simply choose the derivative term as \(-K_D \dot{q}\).

Eq. (6.103) shows that, for any system trajectory, the Lyapunov function decreases as long as \( \dot{q} \neq 0 \). The system then reaches an \textit{equilibrium posture}. By an argument similar to that in the joint space (see Eqs. (6.44)–(6.46)), this posture is determined by

\[
J_A^T(q)K_P \ddot{x} = 0. \tag{6.104}
\]

From (6.104) it can be recognized that, on the assumption of \textit{full-rank} Jacobian, it is

\[ \ddot{x} = x_d - x = 0, \]

i.e., the sought result.

If measurements of \( x \) and \( \dot{x} \) are made directly in the operational space, \( k(q) \) and \( J_A(q) \) in the scheme of Fig. 6.31 are just indicative of direct kinematics functions; it is, however, necessary to measure \( q \) to update both \( J_A^T(q) \) and \( g(q) \) on line. If measurements of operational space quantities are indirect, the controller has to compute the direct kinematics functions, too.

### 6.6.3 Inverse Dynamics Control

Consider now the problem of tracking an operational space trajectory. Recall the manipulator dynamic model in the form (6.47)

\[
B(q)\ddot{q} + n(q, \dot{q}) = u
\]

where \( n \) is given by (6.48). As in (6.49), the choice of the \textit{inverse dynamics linearizing control}

\[
u = B(q)y + n(q, \dot{q})
\]

leads to the system of double integrators

\[
\ddot{q} = y. \tag{6.105}
\]
FIGURE 6.32
Block scheme of operational space inverse dynamics control.

The new control input $y$ is to be designed so as to allow tracking of a trajectory specified by $x_d(t)$. To this purpose, the second-order differential equation in the form (4.117)

$$\ddot{x} = J_A(q)\ddot{q} + \dot{J}_A(q, \dot{q})\dot{q}$$

suggests, for a nonredundant manipulator, the choice of the control law

$$y = J_A^{-1}(q)(\ddot{x} + K_D \ddot{x} + K_P \ddot{x} - \dot{J}_A(q, \dot{q})\dot{q})$$

(6.106)

with $K_P$ and $K_D$ positive definite (diagonal) matrices. In fact, substituting (6.106) into (6.105) gives

$$\ddot{x} + K_D \ddot{x} + K_P \ddot{x} = 0$$

(6.107)

which describes the operational space error dynamics, with $K_P$ and $K_D$ determining the error convergence rate to zero. The resulting inverse dynamics control scheme is reported in Fig. 6.32, which confirms the anticipated analogy with the scheme of Fig. 6.29. Again in this case, besides $x$ and $\dot{x}$, also $q$ and $\dot{q}$ are to be measured. If measurements of $x$ and $\dot{x}$ are indirect, the controller must compute the direct kinematics functions $k(q)$ and $J_A(q)$ on line.

A critical analysis of the schemes in Figs. 6.31 and 6.32 reveals that the design of an operational space controller always requires computation of manipulator Jacobian. As a consequence, controlling a manipulator in the operational space is in general more complex than controlling it in the joint space. In fact, the presence of singularities and/or redundancy influences the Jacobian, and the induced effects are somewhat difficult to handle with an operational space controller. For instance, if a singularity occurs for the scheme of Fig. 6.31 and the error enters the null space of the Jacobian, the manipulator gets stuck at a different configuration from the desired one. This problem is even more
critical for the scheme of Fig. 6.32 which would require the computation of a DLS-
inverse of the Jacobian. Yet, for a redundant manipulator, a joint space control scheme
is naturally transparent to this situation, since redundancy has been already solved by
inverse kinematics, whereas an operational space control scheme should incorporate a
redundancy handling technique inside the feedback loop.

6.7 A COMPARISON BETWEEN VARIOUS CONTROL SCHEMES

In order to make a comparison between the various control schemes presented, consider
the two-link planar arm with the same data of Example 4.2:

\[ a_1 = a_2 = 1 \text{ m} \quad \ell_1 = \ell_2 = 0.5 \text{ m} \quad m_{t_1} = m_{t_2} = 50 \text{ kg} \quad I_{t_1} = I_{t_2} = 10 \text{ kg} \cdot \text{m}^2 \]

\[ k_{r_1} = k_{r_2} = 100 \quad m_{m_1} = m_{m_2} = 5 \text{ kg} \quad I_{m_1} = I_{m_2} = 0.01 \text{ kg} \cdot \text{m}^2. \]

The arm is assumed to be driven by two equal actuators with the following data:

\[ F_{m_1} = F_{m_2} = 0.01 \text{ N} \cdot \text{m/s/rad} \quad R_{a_1} = R_{a_2} = 10 \text{ ohm} \]

\[ k_{t_1} = k_{t_2} = 2 \text{ N} \cdot \text{m/A} \quad k_{v_1} = k_{v_2} = 2 \text{ V/s/rad}; \]

it can be verified that \( F_{m_i} \ll k_{vi} k_{ti} / R_{ai} \) for \( i = 1, 2 \).

The desired tip trajectories have a typical trapezoidal velocity profile, and thus
it is anticipated that sharp torque variations will be induced. The tip path is a motion
of 1.6 m along the horizontal axis, as in the path of Example 4.2. In the first case (fast
trajectory), the acceleration time is 0.6 s and the maximum velocity is 1 m/s. In the
second case (slow trajectory), the acceleration time is 0.6 s and the maximum velocity
is 0.25 m/s. The motion of the controlled arm was simulated on a computer, by adopting
a discrete-time implementation of the controller with a sampling time of 1 ms.

The following control schemes in the joint space and in the operational space
have been utilized; an (analytic) inverse kinematics solution has been implemented to
generate the reference inputs to the joint space control schemes.

A. Independent joint control with position and velocity feedback (Fig. 6.8) with the
following data for each joint servo:

\[ K_P = 5 \quad K_V = 10 \quad k_{TP} = k_{TV} = 1, \]

Corresponding to \( \omega_n = 5 \text{ rad/s} \) and \( \zeta = 0.5 \).

B. Independent joint control with position, velocity, and acceleration feedback
(Fig. 6.10) with the following data for each joint servo:

\[ K_P = 5 \quad K_V = 10 \quad K_A = 2 \quad k_{TP} = k_{TV} = k_{TA} = 1, \]

Corresponding to \( \omega_n = 5 \text{ rad/s} \), \( \zeta = 0.5 \), and \( X_R = 100 \). To reconstruct
acceleration, a first-order filter has been utilized (Fig. 6.12) characterized by
\( \omega_{3f} = 100 \text{ rad/s}. \)
C. As in scheme A with the addition of a decentralized feedforward action (Fig. 6.14).

D. As in scheme B with the addition of a decentralized feedforward action (Fig. 6.15).

E. Joint space computed torque control (Fig. 6.19) with feedforward compensation of the diagonal terms of the inertia matrix and of gravitational terms, and decentralized feedback controllers as in scheme A.

F. Joint space PD control with gravity compensation (Fig. 6.22), modified by the addition of a feedforward velocity term $K_D \ddot{q}_d$, with the following data:

$$K_P = 3750I, \quad K_D = 750I.$$

G. Joint space inverse dynamics control (Fig. 6.24) with the following data:

$$K_P = 25I, \quad K_D = 5I.$$

H. Joint space robust control (Fig. 6.25), on the assumption of constant inertia ($\hat{\overline{B}} = \overline{B}$) and compensation of friction and gravity ($\hat{\tau} = F_v \dot{q} + g$), with the following data:

$$K_P = 25I, \quad K_D = 5I, \quad P = I, \quad \rho = 70, \quad \varepsilon = 0.004.$$

I. As in case H with $\varepsilon = 0.01$.

J. Joint space adaptive control (Fig. 6.28) with a parameterization of the arm dynamic model (4.81) as in (4.82) and (4.83). The initial estimate of the vector $\hat{\pi}$ is computed on the basis of the nominal parameters. The arm is supposed to carry a load which causes the following variations on the second link parameters:

$$\Delta m_2 = 10\text{ kg}, \quad \Delta m_2 \ell_{C_2} = 11\text{ kg} \cdot \text{m}, \quad \Delta \tilde{I}_2 = 12.12\text{ kg} \cdot \text{m}^2.$$

This information is obviously utilized only to update the simulated arm model.

Further the following data are set:

$$A = 5I, \quad K_P = 750I, \quad K_\pi = 0.01I.$$

K. Operational space PD control with gravity compensation (Fig. 6.31), modified by the addition of a feedforward velocity term $K_D \ddot{x}_d$, with the following data:

$$K_P = 16250I, \quad K_D = 3250I.$$

L. Operational space inverse dynamics control (Fig. 6.32) with the following data:

$$K_P = 25I, \quad K_D = 5I.$$
It is worth remarking that the adopted model of the dynamic system of arm with drives is that described by Eq. (6.35). In the decentralized control schemes A–E, the joints have been voltage-controlled as in the block scheme of Fig. 6.20, with unit amplifier gains \( G_v = I \). On the other hand, in the centralized control schemes F–L, the joints have been current-controlled as in the block scheme of Fig. 6.21, with unit amplifier gains \( G_i = I \).

Regarding the parameters of the various controllers, these have been chosen in such a way as to allow a significant comparison of the performance of each scheme in response to congruent control actions. In particular, it can be observed that:

- The dynamic behavior of the joints is the same for schemes A–E.
- The gains of the PD actions in schemes F, J, and K have been tuned so as to obtain response times similar to those of schemes A–E.
- The gains of the PD actions in schemes G, H, I, and L have been chosen so as to obtain the same natural frequency and damping ratios as those of schemes A–E.

The results obtained with the various control schemes are illustrated in Figs. 6.33–6.41 for the fast trajectory and in Figs. 6.42–6.50 for the slow trajectory, respectively. In the case of two quantities represented in the same plot notice that:
FIGURE 6.34
Time history of the joint torques and of the norm of tip position error for the fast trajectory; left—with control scheme C, right—with control scheme D.

FIGURE 6.35
Time history of the joint torques and of the norm of tip position error for the fast trajectory with control scheme E.

- For the joint trajectories, the dashed line indicates the reference trajectory obtained from the tip trajectory via inverse kinematics, while the solid line indicates the actual trajectory followed by the arm.
- For the joint torques, the solid line refers to joint 1 while the dashed line refers to joint 2.
FIGURE 6.36
Time history of the joint positions and torques and of the norm of tip position error for the fast trajectory with control scheme F.

FIGURE 6.37
Time history of the joint torques and of the norm of tip position error for the fast trajectory with control scheme G.

- For the tip position error, the solid line indicates the error component along the horizontal axis while the dashed line indicates the error component along the vertical axis.

Finally, the representation scales have been made as uniform as possible in order to
allow a more direct comparison of the results.

Regarding performance of the various control schemes for the fast trajectory, the obtained results allow drawing the following considerations.

Deviation of the actual joint trajectories from the desired ones shows that tracking performance of scheme A is quite poor (Fig. 6.33). It should be noticed, however, that
FIGURE 6.40
Time history of the joint torques and of the norm of tip position error for the fast trajectory with control scheme K.

FIGURE 6.41
Time history of the joint torques and of the norm of tip position error for the fast trajectory with control scheme L.

the largest contribution to the error is caused by a time lag of the actual trajectory behind the desired one, while the distance of the tip from the geometric path is quite contained. Similar results were obtained with scheme B, and then they have not been reported.

With schemes C and D, an appreciable tracking accuracy improvement is observed (Fig. 6.34), with better performance for the second scheme, thanks to the outer acceleration feedback loop that allows prescribing a disturbance rejection factor twice as much as for the first scheme. Notice that the feedforward action allows obtaining a set of torques which are closer to the nominal ones required to execute the desired trajectory; the torque time history has a discontinuity in correspondence of the acceleration and deceleration fronts.

The tracking error is further decreased with scheme E (Fig. 6.35), by virtue of the additional nonlinear feedforward compensation.

Scheme F guarantees stable convergence to the final arm posture with a tracking performance which is better than that of schemes A and B, thanks to the presence of a velocity feedforward action, but worse than that of schemes C–E, in view of lack of an
FIGURE 6.42
Time history of the joint positions and torques and of the tip position errors for the slow trajectory with control scheme A.

acceleration feedforward action (Fig. 6.36).

As would be logical to expect, the best results are observed with scheme G for which the tracking error is practically zero, and it is mainly due to numerical discretization of the controller (Fig. 6.37).

It is then worth comparing the performance of schemes H and I (Fig. 6.38). In fact, the choice of a small threshold value for ε (scheme H) induces high-frequency components in joint 1 torque (see the thick portions of the torque plot) at the advantage of a very limited tracking error. As the threshold value is increased (scheme I), the torque assumes a smoother behavior at the expense of a doubled norm of tracking error, though.

For scheme J, a lower tracking error than that of scheme F is observed, thanks to the effectiveness of the adaptive action on the parameters of the dynamic model. Nonetheless, the parameters do not converge to their nominal values, as confirmed by the time history of the norm of the parameter error vector that reaches a nonnull steady-state value (Fig. 6.39).

Finally, the performance of schemes K and L is substantially comparable to that of corresponding schemes F and G (Figs. 6.40 and 6.41).

Performance of the various control schemes for the slow trajectory is globally better than that for the fast trajectory. Such improvement is particularly evident for the decentralized control schemes (Figs. 6.42–6.44), whereas the tracking error reduction
FIGURE 6.43
Time history of the joint torques and of the norm of tip position error for the slow trajectory; left—with control scheme C, right—with control scheme D.

FIGURE 6.44
Time history of the joint torques and of the norm of tip position error for the slow trajectory with control scheme E.

for the centralized control schemes is less dramatic (Figs. 6.45–6.50), in view of the small order of magnitude of the errors already obtained for the fast trajectory. In any case, as regards performance of each single scheme, it is possible to make a number of remarks analogous to those previously made.
FIGURE 6.45  
Time history of the joint positions and torques and of the norm of tip position error for the slow trajectory with control scheme F.

FIGURE 6.46  
Time history of the joint torques and of the norm of tip position error for the slow trajectory with control scheme G.

PROBLEMS

6.1. With reference to the block scheme with position feedback in Fig. 6.6, find the transfer functions of the forward path, the return path, and the closed-loop system.

6.2. With reference to the block scheme with position and velocity feedback in Fig. 6.8, find the transfer functions of the forward path, the return path, and the closed-loop system.
FIGURE 6.47
Time history of the joint torques and of the norm of tip position error for the slow trajectory; left—with control scheme H, right—with control scheme I.

FIGURE 6.48
Time history of the norm of tip position error and of the norm of parameter error vector for the slow trajectory with control scheme I.

6.3. With reference to the block scheme with position, velocity, and acceleration feedback in Fig. 6.10, find the transfer functions of the forward path, the return path, and the closed-loop system.

6.4. For a single joint drive system with the data: \( I = 6 \text{ kg} \cdot \text{m}^2 \), \( R_a = 0.3 \text{ ohm} \), \( k_t = 0.5 \text{ N} \cdot \text{m}/\text{A} \), \( k_v = 0.5 \text{ V} \cdot \text{s}/\text{rad} \), and \( F_m = 0.001 \text{ N} \cdot \text{m} \cdot \text{s}/\text{rad} \), find the parameters of the controller with position feedback (unit transducer constant) that allows obtaining a closed-
loop response with damping ratio $\zeta \geq 0.4$. Discuss disturbance rejection properties.

6.5. For the drive system of Problem 6.4, find the parameters of the controller with position and velocity feedback (unit transducer constants) that allows obtaining a closed-loop response with damping ratio $\zeta \geq 0.4$ and natural frequency $\omega_n = 20 \text{ rad/s}$. Discuss disturbance rejection properties.

6.6. For the joint drive system of Problem 6.4, find the parameters of the controller with position, velocity, and acceleration feedback (unit transducer constants) that allows obtaining a closed-loop response with damping ratio $\zeta \geq 0.4$, natural frequency $\omega_n = 20 \text{ rad/s}$ and disturbance rejection factor $X_R = 400$. Also, design a first-order filter that allows acceleration measurement reconstruction.

6.7. Verify that the control schemes in Figs. 6.13, 6.14, and 6.15 correspond to realize Eqs. (6.28), (6.29), and (6.30), respectively.

6.8. Verify that the standard regulation schemes in Figs. 6.16, 6.17, and 6.18 are equivalent to the schemes in Figs. 6.13, 6.14, and 6.15, respectively.


6.10. For the two-link planar arm with the same data as in Section 6.7, design a joint control of
PD type with gravity compensation. By means of a computer simulation, verify stability for the following postures $q = [\pi/4 \ -\pi/2]^T$ and $q = [-\pi \ -3\pi/4]^T$, respectively. Implement the control in discrete-time with a sampling time of 1 ms.

6.11. For the two-link planar arm with the same data as in Section 6.7, on the assumption of a concentrated tip payload of mass $m_L = 10$ kg, design an independent joint control with feedforward computed torque. Perform a computer simulation of the motion of the controlled arm along the joint space rectilinear path from $q_i = [0 \ \pi/4]^T$ to $q_f = [\pi/2 \ \pi/2]^T$ with a trapezoidal velocity profile and a trajectory duration $t_f = 1$ s. Implement the control in discrete-time with a sampling time of 1 ms.

6.12. For the two-link planar arm of Problem 6.11, design an inverse dynamics joint control. Perform a computer simulation of the motion of the controlled arm along the trajectory specified in Problem 6.11. Implement the control in discrete-time with a sampling time of 1 ms.

6.13. For the two-link planar arm of Problem 6.11, design a robust joint control. Perform a computer simulation of the motion of the controlled arm along the trajectory specified in Problem 6.11. Implement the control in discrete-time with a sampling time of 1 ms.

6.14. For the two-link planar arm of Problem 6.11, design an adaptive joint control, on the basis of a suitable parameterization of the arm dynamic model. Perform a computer simulation of the motion of the controlled arm along the trajectory specified in Problem 6.11. Implement the control in discrete-time with a sampling time of 1 ms.

6.15. For the two-link planar of Problem 6.11, design a PD control in the operational space with gravity compensation. By means of a computer simulation, verify stability for the following postures $p = [0.5 \ 0.5]^T$ and $p = [0.6 \ -0.2]^T$, respectively. Implement the control in discrete-time with a sampling time of 1 ms.

6.16. For the two-link planar arm of Problem 6.11, design an inverse dynamics control in the operational space. Perform a computer simulation of the motion of the controlled arm along the operational space rectilinear path from $p(0) = [0.7 \ 0.2]^T$ to $p(1) = [0.1 \ -0.6]^T$ with a trapezoidal velocity profile and a trajectory duration $t_f = 1$ s. Implement the control in discrete-time with a sampling time of 1 ms.

BIBLIOGRAPHY


