Chapter 14

Motion control

14.1. Introduction

The problem of controlling robots has been extensively addressed in the literature. A great variety of control approaches have been proposed. The most common in use with present industrial robots is a decentralized "proportional, integral, derivative" (PID) control for each degree of freedom. More sophisticated nonlinear control schemes have been developed, such as so-called *computed torque control*, termed *inverse dynamic control*, which linearizes and decouples the equation of motion of the robot. Owing to the modeling uncertainties, nonlinear adaptive techniques have been considered in order to identify on-line the dynamic parameters. More recently, properties of the dynamic model have led Lyapunov-based and passivity-based controls to be proposed.

In this chapter, we first study the classical PID control, then the nonlinear linearizing and decoupling control, which is considered to be the best theoretical solution for the control of robots. Finally, we present some advanced methods related to passivity-based and adaptive controls. Detailed surveys on robot control can be found in [Spong 89], [Samson 91], [Lewis 93], [Zodiac 96].

For simplicity, we will only consider serial robots. The methods presented can easily be generalized to robots with complex structures by employing the results of Chapter 10.

14.2. Equations of motion

In order to understand the basic problem of robot control, it is useful to recall the dynamic model (Chapter 9) whose general form for a robot with n degrees of freedom is the following:

$$\Gamma = \mathbf{A}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \mathbf{diag}(\dot{\mathbf{q}})\,\mathbf{F}_v + \mathbf{diag}(\mathrm{sign}(\dot{\mathbf{q}}))\,\mathbf{F}_c \qquad [14.1]$$

or, in a more compact form:

$$\Gamma = \mathbf{A}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \qquad [14.2]$$

and, since the model is linear in the dynamic parameters (equation [12.18]), we can write:

$$\Gamma = \Phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\chi \qquad [14.3]$$

where Γ is the (nx1) vector of joint torques; $\mathbf{A}(\mathbf{q})$ is the (nxn) inertia matrix of the robot; $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}}$ is the (nx1) vector of Coriolis and centrifugal torques; $\mathbf{Q}(\mathbf{q})$ is the vector of gravity torques; $\mathbf{F}_v$ and $\mathbf{F}_c$ are the vectors of the viscous friction and Coulomb friction parameters respectively; χ is the vector of the dynamic parameters (inertial parameters and friction parameters).

The torque transmitted to joint j by a current-driven electrical actuator (continuous or synchronous), assuming that the transmissions introduce neither backlash nor flexibility, is expressed by (equation [12.17]):

$$\Gamma_j = N_j\,K_{aj}\,K_{Tj}\,u_j \qquad [14.4]$$

where N_j is the gear transmission ratio, K_{aj} is the current amplifier gain, K_{Tj} is the torque constant of actuator j, and u_j is the control input of the amplifier.

The design of the control consists of computing the joint actuator torques (Γ_j, then u_j) in order to track a desired trajectory or to reach a given position.

14.3. PID control

14.3.1. *PID control in the joint space*

The dynamic model is described by a system of n coupled nonlinear second order differential equations, n being the number of joints. However, for most of today's industrial robots, a local decentralized "proportional, integral, derivative" (PID) control with constant gains is implemented for each joint. The advantages of such a technique are the simplicity of implementation and the low computational cost. The drawbacks are that the dynamic performance of the robot varies according to its configuration, and poor dynamic accuracy when tracking a high velocity trajectory. In many applications, these drawbacks are not of much significance.

Practically, the block diagrams of such a control scheme in the joint space is shown in Figure 14.1. The control law is given by:

$$\Gamma = K_p (q^d - q) + K_d (\dot{q}^d - \dot{q}) + K_I \int_{t0}^{t} (q^d - q) d\tau \qquad [14.5]$$

where $q^d(t)$ and $\dot{q}^d(t)$ denote the desired joint positions and velocities, and where K_p, K_d and K_I are (nxn) positive definite diagonal matrices whose generic elements are the proportional K_{pj}, derivative K_{dj} and integral K_{Ij} gains respectively.

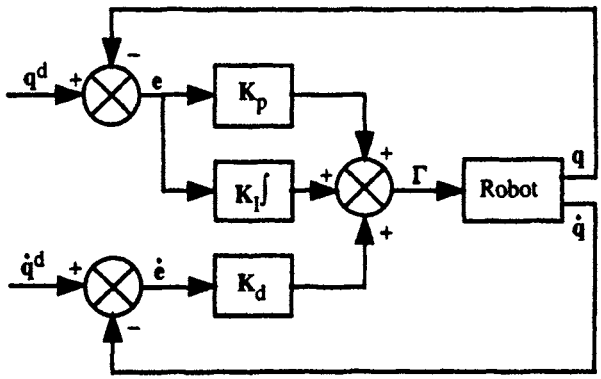

Figure 14.1. *Block diagram of a PID control scheme in the joint space*

The computation of K_{pj}, K_{dj} and K_{Ij} is carried out by considering that joint j is modeled by a linear second order differential equation such that:

$$\Gamma_j = a_j \ddot{q}_j + F_{vj} \dot{q}_j + \gamma_j \qquad [14.6]$$

where $a_j = A_{jj_{max}}$ is the maximum magnitude of the A_{jj} element of the inertia matrix of the robot and γ_j represents a disturbance torque.

Hence, assuming $\gamma_j = 0$, the closed-loop transfer function is given by:

$$\frac{q_j(s)}{q_j^d(s)} = \frac{K_{dj}\, s^2 + K_{pj}\, s + K_{Ij}}{a_j\, s^3 + (K_{dj} + F_{vj})\, s^2 + K_{pj}\, s + K_{Ij}} \qquad [14.7]$$

and the characteristic equation is:

$$\Delta(s) = a_j s^3 + (K_{dj} + F_{vj}) s^2 + K_{pj} s + K_{Ij} \qquad [14.8]$$

The most common solution in robotics consists of adjusting the gains in order to obtain a negative real triple pole. This yields the fastest possible response without overshoot. Thus, the characteristic equation is written as:

$$\Delta(s) = a_j (s + \omega_j)^3 \qquad [14.9]$$

with $\omega_j > 0$, and after solution, we obtain:

$$\begin{cases} K_{pj} = 3\, a_j\, \omega_j^2 \\ K_{dj} + F_{vj} = 3\, a_j\, \omega_j \\ K_{Ij} = a_j\, \omega_j^3 \end{cases} \qquad [14.10]$$

NOTES.–

- high gains $\mathbf{K_p}$ and $\mathbf{K_d}$ decrease the tracking error but bring the system to the neighborhood of the instability domain. Thus, the frequency ω_j should not be greater than the structural resonance frequency ω_{rj}. A reasonable trade-off is that $\omega_j = \omega_{rj} / 2$;
- in the absence of integral action, a static error due to gravity may affect the final position. Practically, the integral action can be deactivated when the position error is very large, since the proportional action is sufficient. It should also be deactivated if the position error becomes too small in order to avoid oscillations that could be caused by Coulomb frictions;
- the predictive action $\mathbf{K_d}\ \dot{\mathbf{q}}^d$ of equation [14.5] reduces significantly the tracking errors. In classical control engineering, this action is not often used;
- the gain $\mathbf{K_d}$ is generally integrated within the servo amplifier, whereas gain $\mathbf{K_p}$ is numerically implemented;
- the performance of a robot controlled in this way is acceptable if high-gear transmission ratios are used (scaling down the time-varying inertias and the coupling torques), if the robot is moving at low velocity, and if high position gains are assigned [Samson 83].

14.3.2. *Stability analysis*

If gravity effects are compensated by an appropriate mechanical design as for the SCARA robot, or by the control software, it can be shown that a PD control law is asymptotically stable for the regulation control problem [Arimoto 84]. The

demonstration is based on the definition of the following Lyapunov function candidate (Appendix 9):

$$V = \frac{1}{2}\dot{\mathbf{q}}^T \mathbf{A}(\mathbf{q})\,\dot{\mathbf{q}} + \frac{1}{2}\mathbf{e}^T \mathbf{K}_p\,\mathbf{e} \qquad [14.11]$$

where $\mathbf{e} = \mathbf{q}^d - \mathbf{q}$ is the position error, and where $\mathbf{q}^d$ is the desired joint position.

Since $\mathbf{q}^d$ is constant, the PD control law is given by:

$$\Gamma = \mathbf{K}_p\,\mathbf{e} - \mathbf{K}_d\,\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) \qquad [14.12]$$

From equations [14.1] and [14.12], and in the absence of friction, we obtain the following closed-loop equation:

$$\mathbf{K}_p\,\mathbf{e} - \mathbf{K}_d\,\dot{\mathbf{q}} = \mathbf{A}\,\ddot{\mathbf{q}} + \mathbf{C}\,\dot{\mathbf{q}} \qquad [14.13]$$

Differentiating the Lyapunov function [14.11] with respect to time yields:

$$\dot{V} = \frac{1}{2}\dot{\mathbf{q}}^T \dot{\mathbf{A}}\,\dot{\mathbf{q}} + \dot{\mathbf{q}}^T \mathbf{A}\,\ddot{\mathbf{q}} - \mathbf{e}^T \mathbf{K}_p\,\dot{\mathbf{q}} \qquad [14.14]$$

and substituting $\mathbf{A}\,\ddot{\mathbf{q}}$ from equation [14.13] yields:

$$\dot{V} = \frac{1}{2}\dot{\mathbf{q}}^T (\dot{\mathbf{A}} - 2\,\mathbf{C})\,\dot{\mathbf{q}} - \dot{\mathbf{q}}^T \mathbf{K}_d\,\dot{\mathbf{q}} \qquad [14.15]$$

Since the matrix $[\dot{\mathbf{A}} - 2\,\mathbf{C}]$ is skew-symmetric [Koditschek 84], [Arimoto 84] (§ 9.3.3.3), the term $\dot{\mathbf{q}}^T [\dot{\mathbf{A}} - 2\,\mathbf{C}]\,\dot{\mathbf{q}}$ is zero, giving:

$$\dot{V} = -\dot{\mathbf{q}}^T \mathbf{K}_d\,\dot{\mathbf{q}} \leq 0 \qquad [14.16]$$

This result shows that $\dot{V}$ is negative semi-definite, which is not sufficient to demonstrate that the equilibrium point ($\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{q}} = \mathbf{0}$) is asymptotically stable (Appendix 9). We have now to prove that as $\dot{\mathbf{q}} = \mathbf{0}$, the robot does not reach a configuration $\mathbf{q} \neq \mathbf{q}^d$. This can be done, thanks to the La Salle invariant set theorem [Hahn 67] (Appendix 9). The set $\mathcal{R}$ of points in the neighborhood of the equilibrium that satisfies $\dot{V} = 0$ is such that $\dot{\mathbf{q}} = \mathbf{0}$ and thus $\ddot{\mathbf{q}} = \mathbf{0}$. From equation [14.13], we conclude that necessarily $\mathbf{e} = \mathbf{0}$. Consequently, the equilibrium point ($\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{q}} = \mathbf{0}$) is the only possible equilibrium for the system and is also the largest invariant set in $\mathcal{R}$. Therefore, the equilibrium point is asymptotically stable.

Furthermore, it has been demonstrated that the system is asymptotically stable if in equation [14.12] we replace $\mathbf{Q(q)}$ by the constant term $\mathbf{Q(q^d)}$, corresponding to gravity torque at the desired position $\mathbf{q^d}$. The stability is also proven if one takes $K_{pj} > \|\partial \mathbf{Q(q)}/\partial \mathbf{q}\|$, which represents the 2-norm of the Jacobian matrix of gravity torques with respect to the joint variables [Korrami 88], [Tomei 91]. For more details on the computation of the gains when considering the robot dynamics, interested readers should refer to [Qu 91], [Kelly 95], [Rocco 96], [Freidovich 97].

14.3.3. *PID control in the task space*

When the motion is specified in the task space, one of the following schemes can be used to control the system:

- the control law is designed in the task space;
- the specified trajectory in the task space is transformed into a trajectory in the joint space, then a control in the joint space is performed.

For PID control in the task space, the control law is obtained by replacing $\mathbf{q}$ by $\mathbf{X}$ in equation [14.5] and by transforming the task space error signal into the joint space by multiplying it by $\mathbf{J}^T$(Figure 14.2):

$$\Gamma = \mathbf{J}^T [\mathbf{K}_p (\mathbf{X}^d - \mathbf{X}) + \mathbf{K}_d (\dot{\mathbf{X}}^d - \dot{\mathbf{X}}) + \mathbf{K}_I \int_{t0}^{t} (\mathbf{X}^d - \mathbf{X}) \, d\tau] \qquad [14.17]$$

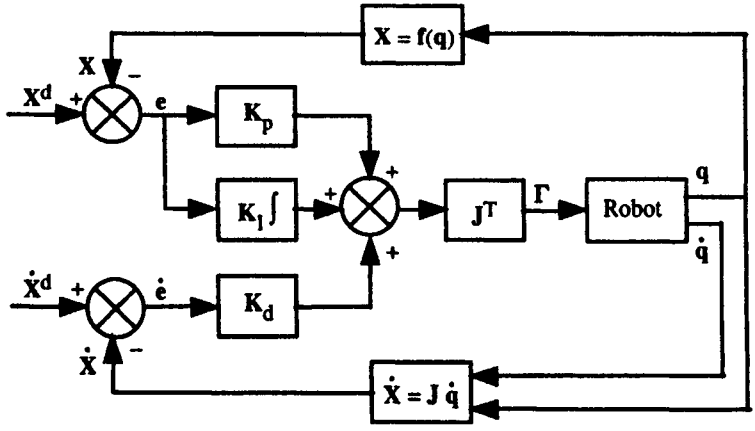

Figure 14.2. *Block diagram of a PID control scheme in the task space*

Two solutions are possible to transform the desired task space trajectory into the joint space: either we use the IGM to compute the joint positions then we compute the velocities and accelerations by differentiating the positions; or we compute the joint positions, velocities and accelerations as indicated below:

i) using the IGM (Chapter 4) to compute the joint positions:

$$\mathbf{q}^d = \mathbf{g}(\mathbf{X}^d) \qquad [14.18]$$

ii) using the IKM (Chapter 6) to compute the joint velocities. In the regular positions:

$$\dot{\mathbf{q}}^d = \mathbf{J}(\mathbf{q}^d)^{-1}\, \dot{\mathbf{X}}^d \qquad [14.19]$$

In singular positions or for redundant robots, the matrix $\mathbf{J}^{-1}$ should be replaced by a generalized inverse as described in Chapter 6;

iii) using the second order IKM (§ 5.10) to compute the joint accelerations (if desired):

$$\ddot{\mathbf{q}}^d = \mathbf{J}^{-1}\,(\ddot{\mathbf{X}}^d - \dot{\mathbf{J}}\,\dot{\mathbf{q}}^d) \qquad [14.20]$$

with:

$$\dot{\mathbf{J}}(\mathbf{q}^d, \dot{\mathbf{q}}^d) = \frac{d}{dt}\mathbf{J}(\mathbf{q}^d) \qquad [14.21]$$

14.4. Linearizing and decoupling control

14.4.1. *Introduction*

When the task requires fast motion of the robot and high dynamic accuracy, it is necessary to improve the performance of the control by taking into account, partially or totally, the dynamic interaction torques. Linearizing and decoupling control is based on canceling the nonlinearities in the robot dynamics [Khalil 78], [Zabala 78], [Raibert 78], [Khatib 80], [Luh 80a], [Freund 82], [Bejczy 85]... Such a control is known as *computed torque control* or *inverse dynamic control* since it is based on the utilization of the dynamic model. Theoretically, it ensures the linearization and the decoupling of the equations of the model, providing a uniform behavior whatever the configuration of the robot.

Implementing this method requires on-line computation of the inverse dynamic model, as well as knowledge of the numerical values of the inertial parameters and

friction parameters. Efficient modeling approaches to minimize the computational burden have been presented in § 9.6. With current computer performance, the computation can be handled on-line at a sufficiently high rate and is not anymore a limiting problem. The inertial parameters can be determined off-line with good accuracy by identification techniques as described in Chapter 12.

The linearizing and decoupling techniques consist of transforming a nonlinear control problem into a linear one by using an appropriate feedback law. In the case of rigid robot manipulators, the design of a linearizing and decoupling control law is facilitated by the fact that the number of actuators is equal to the number of joint variables, and that the inverse dynamic model giving the control input Γ of the system as a function of the state vector $(\mathbf{q}, \dot{\mathbf{q}})$ and of $\ddot{\mathbf{q}}$ is naturally obtained. These features ensure that the equations of the robot define a so-called *flat system* whose *flat outputs* are the joint variables $\mathbf{q}$ [Fliess 95]. Since the control law only involves the state variables $\mathbf{q}$ and $\dot{\mathbf{q}}$, it is termed a *static decoupling control law*. In the following, we describe this method both in the joint space and in the task space.

14.4.2. *Computed torque control in the joint space*

14.4.2.1. *Principle of the control*

Let us assume that joint positions and velocities are measurable and that measurements are noiseless. Let $\hat{\mathbf{A}}$ and $\hat{\mathbf{H}}$ be the estimates of $\mathbf{A}$ and $\mathbf{H}$ respectively. Hence, from equation [14.2], if we define a control law Γ such that [Khalil 79]:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q})\mathbf{w}(t) + \hat{\mathbf{H}}(\mathbf{q}, \dot{\mathbf{q}}) \qquad [14.22]$$

then, after substituting [14.22] into [14.2], we deduce that in the ideal case of perfect modeling and in the absence of disturbances, the problem reduces to that of the linear control of n decoupled double-integrators:

$$\ddot{\mathbf{q}} = \mathbf{w}(t) \qquad [14.23]$$

$\mathbf{w}(t)$ is the new input control vector. In order to define $\mathbf{w}(t)$, we study in the following sections two schemes: the first one is suited for *tracking control* when the trajectory is fully specified, the second one is suited for *position (regulation) control* when just the final point is specified.

14.4.2.2. *Tracking control scheme*

Let $\ddot{\mathbf{q}}^d(t)$, $\dot{\mathbf{q}}^d(t)$ and $\mathbf{q}^d(t)$ be the desired acceleration, velocity and position in the joint space. If we define $\mathbf{w}(t)$ according to the following equation[1]:

$$\mathbf{w}(t) = \ddot{\mathbf{q}}^d + \mathbf{K}_d (\dot{\mathbf{q}}^d - \dot{\mathbf{q}}) + \mathbf{K}_p (\mathbf{q}^d - \mathbf{q}) \qquad [14.24]$$

where $\mathbf{K}_p$ and $\mathbf{K}_d$ are (nxn) positive definite diagonal matrices; hence, referring to equation [14.23], the closed-loop system response is determined by the following decoupled linear error equation:

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \mathbf{0} \qquad [14.25]$$

where $\mathbf{e} = \mathbf{q}^d - \mathbf{q}$.

The solution $\mathbf{e}(t)$ of the error equation is globally exponentially stable. The gains K_{pj} and K_{dj} are adjusted to provide the axis j, over the whole set of configurations of the robot, the desired dynamics with a given damping coefficient ξ_j, and a given control bandwidth fixed by a frequency ω_j:

$$\begin{cases} K_{pj} = \omega_j^2 \\ K_{dj} = 2\,\xi_j\,\omega_j \end{cases} \qquad [14.26]$$

Generally, one seeks a critically damped system ($\xi_j = 1$) to obtain the fastest response without overshoot. The block diagram of this control scheme is represented in Figure 14.3. The control input torque to the actuators includes three components: the first compensates for Coriolis, centrifugal, gravity, and friction effects; the second is a proportional and derivative control with variable gains $\hat{\mathbf{A}}\,\mathbf{K}_p$ and $\hat{\mathbf{A}}\,\mathbf{K}_d$ respectively; and the third provides a predictive action of the desired acceleration torques $\hat{\mathbf{A}}\,\ddot{\mathbf{q}}^d$.

In the presence of modeling errors, the closed loop equation corresponding to Figure 14.3 is obtained by combining equations [14.22] and [14.2]:

$$\hat{\mathbf{A}} (\ddot{\mathbf{q}}^d + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e}) + \hat{\mathbf{H}} = \mathbf{A}\,\ddot{\mathbf{q}} + \mathbf{H} \qquad [14.27]$$

yielding:

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \hat{\mathbf{A}}^{-1} [(\mathbf{A} - \hat{\mathbf{A}})\,\ddot{\mathbf{q}} + \mathbf{H} - \hat{\mathbf{H}}] \qquad [14.28]$$

[1] An integral action on $\mathbf{w}(t)$ can also be added.

In this equation, the modeling errors constitute an excitation for the error equation. When these errors are too large, it is necessary to increase the proportional and derivative gains, but their magnitudes are limited by the stability of the system. The robustness and the stability of this control are addressed by Samson et al. [Samson 87]. It is shown namely that the matrix $\hat{\mathbf{A}}$ must be positive definite. It is shown as well that the errors $\mathbf{e}$ and $\dot{\mathbf{e}}$ decrease while the gains increase.

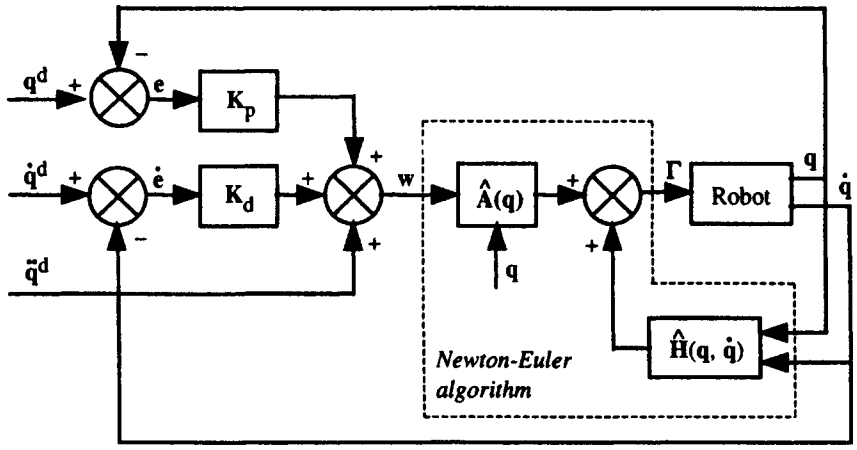

Figure 14.3. *Computed torque: block diagram of the tracking control scheme in the joint space*

14.4.2.3. *Position control scheme*

Let $\mathbf{q}^d$ be the desired position. A possible choice for $\mathbf{w}(t)$ is as follows (Figure 14.4):

$$\mathbf{w}(t) = \mathbf{K}_p(\mathbf{q}^d - \mathbf{q}) - \mathbf{K}_d\,\dot{\mathbf{q}} \qquad [14.29]$$

From equations [14.23] and [14.29], we obtain the closed-loop equation of the system:

$$\ddot{\mathbf{q}} + \mathbf{K}_d\,\dot{\mathbf{q}} + \mathbf{K}_p\,\mathbf{q} = \mathbf{K}_p\,\mathbf{q}^d \qquad [14.30]$$

which describes a decoupled linear system of second order differential equations. The solution $\mathbf{q}(t)$ is globally exponentially stable by properly choosing $\mathbf{K}_p$ and $\mathbf{K}_d$.

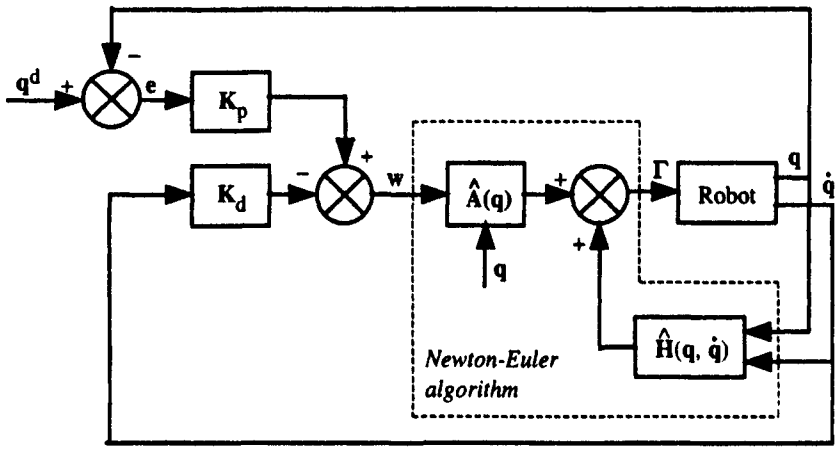

Figure 14.4. *Computed torque: block diagram of the position control scheme in the joint space*

14.4.2.4. *Predictive dynamic control*

Another scheme has been proposed by [Khalil 78] based on a predictive dynamic control: the estimates $\hat{\mathbf{A}}$ and $\hat{\mathbf{H}}$ are no longer computed with the current values of $\mathbf{q}$ and $\dot{\mathbf{q}}$, but rather with the desired values $\mathbf{q}^d$ and $\dot{\mathbf{q}}^d$. Thus, the control law is written as:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q}^d)\,\mathbf{w}(t) + \hat{\mathbf{H}}(\mathbf{q}^d, \dot{\mathbf{q}}^d) \qquad [14.31]$$

where $\mathbf{w}(t)$ is given by equation [14.24] or [14.29] according to the desired scheme.

In the case of exact modeling, we can assume that $\hat{\mathbf{A}}(\mathbf{q}) = \hat{\mathbf{A}}(\mathbf{q}^d)$ and $\hat{\mathbf{H}}(\mathbf{q}, \dot{\mathbf{q}}) = \hat{\mathbf{H}}(\mathbf{q}^d, \dot{\mathbf{q}}^d)$. The control law [14.31] linearizes and decouples the equations of the system as in the previous case. The main advantage of this scheme is that the computation of $\hat{\mathbf{A}}(\mathbf{q}^d)$ and $\hat{\mathbf{H}}(\mathbf{q}^d, \dot{\mathbf{q}}^d)$ is not corrupted by noisy variables.

14.4.2.5. *Practical computation of the computed torque control laws*

The control laws [14.22] and [14.31] can be computed by the inverse dynamic Newton-Euler algorithm (§ 9.5) without requiring explicit knowledge of $\mathbf{A}$ and $\mathbf{H}$. The algorithm provides the joint torques as a function of three arguments, namely the vectors of joint positions, velocities and accelerations. By comparing equations [14.2] and [14.22], we can conclude that:

- to compute the control law [14.22] (Figures 14.3 and 14.4), the input arguments of the Newton-Euler algorithm should be:
  - the joint position is equal to the current joint position $\mathbf{q}$;
  - the joint velocity is equal to the current joint velocity $\dot{\mathbf{q}}$;
  - the joint acceleration is equal to $\mathbf{w}(t)$;
- to compute the control law [14.31], the input arguments of the Newton-Euler algorithm should be:
  - the joint position is equal to the desired joint position $\mathbf{q}^d$;
  - the joint velocity is equal to the desired joint velocity $\dot{\mathbf{q}}^d$;
  - the joint acceleration is equal to $\mathbf{w}(t)$.

The computational cost of the computed torque control in the joint space is therefore more or less equal to the number of operations of the inverse dynamic model. As we stated in Chapter 9, the problem of on-line computation of this model at a sufficient rate is now considered solved (Chapter 9). Some industrial robot controllers offer a partial implementation of the computed torque control algorithm.

14.4.3. *Computed torque control in the task space*

The dynamic control in the task space is also known as *resolved acceleration control* [Luh 80a]. The dynamic behavior of the robot in the task space is described by the following equation, obtained after substituting $\ddot{\mathbf{q}}$ from equation [5.44] into equation [14.2]:

$$\Gamma = \mathbf{A}\,\mathbf{J}^{-1}(\ddot{\mathbf{X}} - \dot{\mathbf{J}}\,\dot{\mathbf{q}}) + \mathbf{H} \qquad [14.32]$$

As in the case of the joint space decoupling control, a control law that linearizes and decouples the equations in the task space is formulated as:

$$\Gamma = \hat{\mathbf{A}}\,\mathbf{J}^{-1}(\mathbf{w}(t) - \dot{\mathbf{J}}\,\dot{\mathbf{q}}) + \hat{\mathbf{H}} \qquad [14.33]$$

Assuming an exact model, the system is governed by the following equation of a double integrator in the task space:

$$\ddot{\mathbf{X}} = \mathbf{w}(t) \qquad [14.34]$$

Several schemes may be considered for defining $\mathbf{w}$ [Chevallereau 88]. For a tracking control scheme with a PD controller, the control law has the form:

$$w(t) = \ddot{X}^d + K_d (\dot{X}^d - \dot{X}) + K_p (X^d - X) \qquad [14.35]$$

The closed-loop behavior of the robot is described by the following error equation:

$$\ddot{e}_x + K_d \dot{e}_x + K_p e_x = 0 \qquad [14.36]$$

with:

$$e_x = X^d - X \qquad [14.37]$$

The corresponding block diagram is represented in Figure 14.5. The control input Γ can be computed by the inverse dynamic algorithm of Newton-Euler with the following arguments:

- the joint position is equal to the current joint position q;
- the joint velocity is equal to the current joint velocity $\dot{q}$;
- the joint acceleration is equal to $J^{-1}(w(t) - \dot{J}\,\dot{q})$.

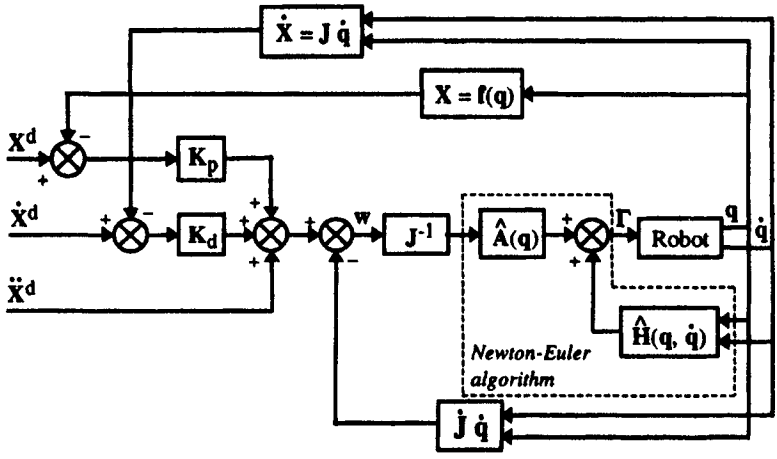

Figure 14.5. *Computed torque control in the task space*

In Appendix 10, we present an efficient algorithm to implement the computed torque control in the task space [Khalil 87a], [Chevallereau 88]. The proposed inverse dynamic algorithm uses many variables that must also be computed for the kinematic models. The computation of $\dot{J}\,\dot{q}$ is achieved with a recursive algorithm without differentiating J. In the case of the Stäubli RX-90 robot, the computational

cost of such an algorithm is 316 multiplications and 237 additions if we use the simplified base inertial parameters of Table 9.4.

NOTE.– If the robot is redundant, we replace the matrix $\mathbf{J}^{-1}$ in equation [14.33] by a generalized inverse. It can be shown that the robot is also governed by equation [14.36] in non-singular configurations. The homogeneous term of the generalized inverse must be chosen appropriately in order to avoid self joint motions in the null space of $\mathbf{J}$ [Hsu 88], [de Luca 91a], [Ait Mohamed 95].

14.5. Passivity-based control

14.5.1. *Introduction*

In the previous section, it is shown that the computed torque control exploits the inverse dynamic model to cancel the nonlinearities in the robot dynamics. In this section, we investigate another approach that uses the property of passivity of the robot (system that dissipates energy). Such control laws modify the natural energy of the robot so that it satisfies the desired objectives (position control or tracking control). In what follows, we first describe the robot dynamics with the Hamiltonian formalism, then we introduce the concept of passivity in the case of regulation control (fixed desired point). Finally, we show how to design a tracking controller by using properties of the passive feedback systems (Appendix 11). This section is largely based on the work of [Berguis 93] and [Landau 88].

14.5.2. *Hamiltonian formulation of the robot dynamics*

The Hamiltonian gives the total energy of the robot:

$$H = E + U \qquad [14.38]$$

where:

- $E(\mathbf{q}, \dot{\mathbf{q}})$ is the kinetic energy of the robot equal to $\frac{1}{2}\dot{\mathbf{q}}^T \mathbf{A}(\mathbf{q})\dot{\mathbf{q}}$;
- $U(\mathbf{q})$ is the potential energy of the robot;
- $\mathbf{A}(\mathbf{q})$ is the inertia matrix of the robot.

The (nx1) vector of *generalized momenta* is defined as:

$$\mathbf{p} = \mathbf{A}(\mathbf{q})\dot{\mathbf{q}} \qquad [14.39]$$

Therefore, equation [14.38] becomes:

$$H = \mathbf{p}^T \dot{\mathbf{q}} - L(\mathbf{q}, \dot{\mathbf{q}}) \qquad [14.40]$$

where $L = E - U$ is the Lagrangian of the robot. From equation [14.39], it also follows that:

$$E(\mathbf{p}, \mathbf{q}) = \frac{1}{2} \mathbf{p}^T \mathbf{A}^{-1} \mathbf{p} \qquad [14.41]$$

Defining the state variables by the vectors $\mathbf{q}$ and $\mathbf{p}$, we obtain the Hamiltonian equations of motion in state space form as:

$$\dot{\mathbf{q}} = \frac{\partial H(\mathbf{p}, \mathbf{q})}{\partial \mathbf{p}} = \frac{\partial E(\mathbf{p}, \mathbf{q})}{\partial \mathbf{p}} \qquad [14.42]$$

$$\dot{\mathbf{p}} = -\frac{\partial H(\mathbf{p}, \mathbf{q})}{\partial \mathbf{q}} + \Gamma = -\frac{\partial E(\mathbf{p}, \mathbf{q})}{\partial \mathbf{q}} - \frac{\partial U(\mathbf{q})}{\partial \mathbf{q}} + \Gamma \qquad [14.43]$$

Equation [14.43] is obtained from the Lagrangian equation [9.4], noting that:

$$\frac{\partial E(\mathbf{p}, \mathbf{q})}{\partial \mathbf{q}} = -\frac{\partial E(\mathbf{q}, \dot{\mathbf{q}})}{\partial \mathbf{q}}$$

The time derivative of H is such that:

$$\dot{H} = \frac{dH(\mathbf{p}, \mathbf{q})}{dt} = \left[\frac{\partial H(\mathbf{p}, \mathbf{q})}{\partial \mathbf{q}}\right]^T \dot{\mathbf{q}} + \left[\frac{\partial H(\mathbf{p}, \mathbf{q})}{\partial \mathbf{p}}\right]^T \dot{\mathbf{p}} = \dot{\mathbf{q}}^T \Gamma \qquad [14.44]$$

which yields:

$$\int_0^{t1} \dot{\mathbf{q}}^T(t)\, \Gamma(t)\, dt = H[\mathbf{p}(t1), \mathbf{q}(t1)] - H[\mathbf{p}(0), \mathbf{q}(0)] \qquad [14.45]$$

A rigid robot is defined as *passive* from the input Γ to the output $\dot{\mathbf{q}}$ when there exists a constant $0 < \beta < \infty$ such that:

$$\int_0^{t1} \dot{\mathbf{q}}^T(t)\, \Gamma(t)\, dt \geq -\beta$$

which is true, from equation [14.45], when $\beta = H[\mathbf{p}(0), \mathbf{q}(0)]$. This means that the total energy has a bounded minimum.

14.5.3. *Passivity-based position control*

Let us assume that we want to drive the robot to a desired position $\mathbf{q}^d$. Intuitively, this can be achieved by shifting the open-loop energy minimum from ($\dot{\mathbf{q}} = \mathbf{0}$, $\mathbf{q} = \mathbf{0}$) towards ($\dot{\mathbf{q}} = \mathbf{0}$, $\mathbf{e} = \mathbf{0}$) for the closed-loop system, where $\mathbf{e} = \mathbf{q}^d - \mathbf{q}$ is the position error. This shifting can be obtained by reshaping the potential energy of the system such that it attains the desired minimum at $\mathbf{e} = \mathbf{0}$. To this end, let $U^*(\mathbf{q})$ be an arbitrary function of the desired potential energy for the closed-loop system. Let us define the following control law:

$$\Gamma = -\frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} + \frac{\partial U(\mathbf{q})}{\partial \mathbf{q}} + \mathbf{v} \qquad [14.46]$$

where $\mathbf{v}$ is the (nx1) new input control vector [Takegaki 81b]. The Hamiltonian equations become:

$$\dot{\mathbf{q}} = \frac{\partial E(\mathbf{p}, \mathbf{q})}{\partial \mathbf{p}} \qquad [14.47]$$

$$\dot{\mathbf{p}} = -\frac{\partial E(\mathbf{p}, \mathbf{q})}{\partial \mathbf{q}} - \frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} + \mathbf{v} \qquad [14.48]$$

Hence, by using the control law [14.46], the initial Hamiltonian $H(\mathbf{p}, \mathbf{q})$ is modified into the desired one $H^*(\mathbf{p}, \mathbf{q})$ such that:

$$H^* = E + U^* \qquad [14.49]$$

and we can verify that:

$$\dot{H}^* = \dot{\mathbf{q}}^T \mathbf{v} \qquad [14.50]$$

This implies that the robot is passive from the new input $\mathbf{v}$ to the output $\dot{\mathbf{q}}$. To asymptotically stabilize the system, we add a damping in the loop such that:

$$\mathbf{v} = -\mathbf{K}_d \dot{\mathbf{q}} \qquad [14.51]$$

where $\mathbf{K}_d > 0$ is a diagonal matrix. Equation [14.50] becomes:

$$\dot{H}^* = - \dot{\mathbf{q}}^T \mathbf{K}_d \dot{\mathbf{q}} \qquad [14.52]$$

This expression is negative semi-definite. However, it can be verified that the equilibrium point ($\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{q}} = \mathbf{0}$) is the largest invariant set within the set $\dot{H}^*(\mathbf{q}, \mathbf{p}) = 0$. Hence, using the La Salle invariance theorem [Hahn 67], the asymptotic stability of the equilibrium can be proven.

Various choices are possible for the desired potential energy function $U^*(\mathbf{q}, \mathbf{p})$ [Wen 88]. An obvious one that satisfies the constraint of a strict minimum at $\mathbf{e} = \mathbf{0}$ is:

$$U^* = \frac{1}{2} \mathbf{e}^T \mathbf{K}_p \mathbf{e} \qquad [14.53]$$

For this choice, the control law [14.46] becomes:

$$\Gamma = \mathbf{K}_p \mathbf{e} - \mathbf{K}_d \dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) \qquad [14.54]$$

which represents gravity compensation and a linear state-feedback loop [Takegaki 81b], as the one presented in § 14.3.

The following choice for $U^*(\mathbf{q}, \mathbf{p})$, under the condition that $\mathbf{K}_p$ is large enough, is also minimum when $\mathbf{e} = \mathbf{0}$ [Takegaki 81b]:

$$U^* = \frac{1}{2} \mathbf{e}^T \mathbf{K}_p \mathbf{e} + U(\mathbf{q}) - U(\mathbf{q}^d) + \mathbf{e}^T \mathbf{Q}(\mathbf{q}^d) \qquad [14.55]$$

Hence, the control law is:

$$\Gamma = \mathbf{K}_p \mathbf{e} - \mathbf{K}_d \dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}^d) \qquad [14.56]$$

14.5.4. *Passivity-based tracking control*

For a tracking task, it is necessary to modify the control law so that the strict energy minimum ($\mathbf{q} = \mathbf{0}$, $\dot{\mathbf{q}} = \mathbf{0}$) of the open-loop system is shifted towards ($\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{e}} = \mathbf{0}$) for the closed-loop. This can be achieved by modifying both the kinetic energy and the potential energy.

In this section, we analyze the passivity-based control laws of [Paden 88] and [Slotine 87] using the passive system feedback approach proposed by [Landau 88] (Appendix 11). Consider the following control law [Paden 88]:

$$\Gamma = \mathbf{A}(\mathbf{q})\,\ddot{\mathbf{q}}^{d} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}}^{d} + \mathbf{Q}(\mathbf{q}) + \mathbf{K}_p\,\mathbf{e} + \mathbf{K}_d\,\dot{\mathbf{e}} \qquad [14.57]$$

In the absence of friction, equations [14.57] and [14.1] lead to the closed-loop equation:

$$\mathbf{K}_p\,\mathbf{e} + \mathbf{K}_d\,\dot{\mathbf{e}} = \tau \qquad [14.58]$$

with :

$$\tau = -\mathbf{A}(\mathbf{q})\,\ddot{\mathbf{e}} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{e}} \qquad [14.59]$$

Equation [14.58] represents a system of two interconnected feedback blocks (Figure 14.6):

- a linear block B1 in the feedforward chain whose input and output are $\dot{\mathbf{e}}$ and τ respectively;
- a nonlinear block B2 in the feedback chain whose input and output are τ and $-\dot{\mathbf{e}}$ respectively.

In order to prove that the nonlinear block is passive, let us consider the integral of the input-output dot product:

$$\int_0^{t1} -\dot{\mathbf{e}}^{T}(t)\,\tau(t)\,dt = \int_0^{t1} [\dot{\mathbf{e}}^{T}\mathbf{A}(\mathbf{q})\,\ddot{\mathbf{e}} + \dot{\mathbf{e}}^{T}\,\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{e}}]\,dt \qquad [14.60]$$

Since:

$$\dot{\mathbf{e}}^{T}\,\mathbf{A}(\mathbf{q})\,\ddot{\mathbf{e}} = \frac{1}{2}\frac{d}{dt}[\dot{\mathbf{e}}^{T}\,\mathbf{A}(\mathbf{q})\,\dot{\mathbf{e}}] - \frac{1}{2}\,\dot{\mathbf{e}}^{T}\,\dot{\mathbf{A}}(\mathbf{q})\,\dot{\mathbf{e}} \qquad [14.61]$$

then:

$$\int_0^{t1} -\dot{\mathbf{e}}^{T}(t)\,\tau(t)\,dt = \int_0^{t1} [\frac{1}{2}\frac{d}{dt}[\dot{\mathbf{e}}^{T}\mathbf{A}(\mathbf{q})\dot{\mathbf{e}}] - \frac{1}{2}\,\dot{\mathbf{e}}^{T}\dot{\mathbf{A}}(\mathbf{q})\dot{\mathbf{e}} + \dot{\mathbf{e}}^{T}\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{e}}]\,dt \qquad [14.62]$$

Since $[\dot{\mathbf{A}}(\mathbf{q}) - 2\,\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})]$ is skew-symmetric, then $\dot{\mathbf{e}}^{T}\,[\dot{\mathbf{A}}(\mathbf{q}) - 2\,\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})]\,\dot{\mathbf{e}}$ is zero, which reduces equation [14.62] to:

$$\int_0^{t1} -\dot{e}^T(t)\,\tau(t)\,dt = \int_0^{t1} \frac{1}{2}\frac{d}{dt}[\dot{e}^T A(q)\,\dot{e}]\,dt$$

$$= \frac{1}{2}[\dot{e}^T(t1)\,A(q(t1))\,\dot{e}(t1) - \dot{e}^T(0)\,A(q(0))\,\dot{e}(0)] \qquad [14.63]$$

and finally:

$$\int_0^{t1} -\dot{e}^T(t)\,\tau(t)\,dt \geq -\gamma 0^2 = -\frac{1}{2}\dot{e}^T(0)\,A(q(0))\,\dot{e}(0) \qquad [14.64]$$

and, since $\gamma 0^2 < \infty$, it follows that the block B2 is passive.

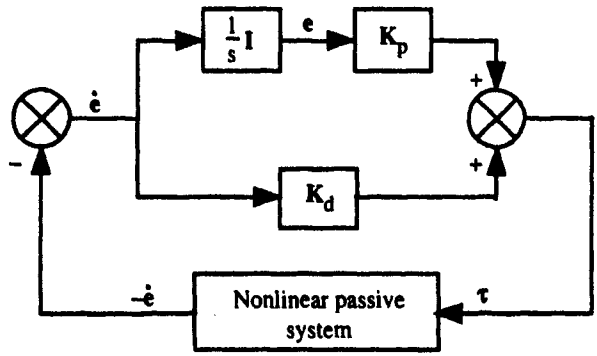

Figure 14.6. *Equivalent feedback representation of the closed-loop equation [14.58] (from [Landau 88])*

The linear block of the feedforward chain is characterized by a positive real transfer matrix:

$$H(s) = K_d + \frac{1}{s}K_p \qquad [14.65]$$

which proves (Appendix 11) that the system represented by equation [14.58] is stable, and more precisely that the error e of the linear system is bounded.

In order to ensure that $\mathbf{e} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$, the control law should be modified so that the transfer function of the feedforward chain be *strictly positive real*[2]. This can be done by removing the pole from the origin, choosing for example H(s) such that:

$$\mathbf{H(s)} = \mathbf{K_d} + \mathbf{K_p}\,[s\mathbf{I} + \Lambda]^{-1} \qquad [14.66]$$

where Λ is a positive definite matrix. Modifying the control law accordingly yields:

$$\Gamma = \mathbf{A(q)}\,\ddot{\mathbf{q}}^d + \mathbf{C(q, \dot{q})}\,\dot{\mathbf{q}}^d + \mathbf{Q(q)} + \mathbf{K_p}\,\tilde{\mathbf{e}} + \mathbf{K_d}\,\dot{\mathbf{e}} \qquad [14.67]$$

with $\frac{d}{dt}\tilde{\mathbf{e}} = -\Lambda\,\tilde{\mathbf{e}} + \dot{\mathbf{e}}$.

The closed-loop equation becomes:

$$\mathbf{K_p}\,\tilde{\mathbf{e}} + \mathbf{K_d}\,\dot{\mathbf{e}} = \tau \qquad [14.68]$$

The corresponding system is shown in Figure 14.7. As the transfer function of the feedforward chain is strictly positive definite, we can conclude that $\tilde{\mathbf{e}}(t) \rightarrow \mathbf{0}$ as $t \rightarrow \infty$ and $\dot{\mathbf{e}} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$, but unfortunately we cannot conclude that $\mathbf{e} \rightarrow \mathbf{0}$.

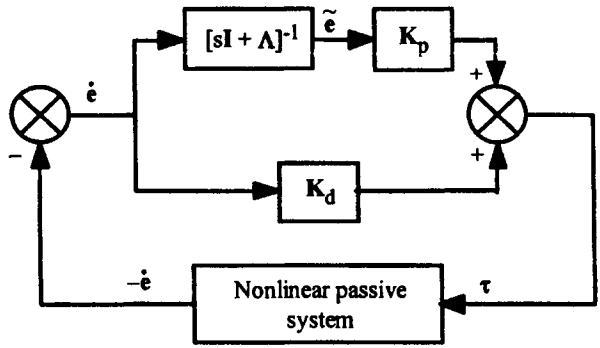

Figure 14.7. *Equivalent feedback representation of the closed-loop equation [14.68] (from [Landau 88])*

In order to ensure that $\mathbf{e} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$, e should be a state of the feedforward chain. This is achieved with the following control law:

---

[2] We can find in [Paden 88] another demonstration proving that this law is asymptotically stable, i.e. that ($\mathbf{e} = \mathbf{0}$, $\dot{\mathbf{e}} = \mathbf{0}$) for arbitrary values $\mathbf{K_p} = \mathbf{K_p}^T > 0$ and $\mathbf{K_d} = \mathbf{K_d}^T > 0$.

$$\Gamma = A(q)\,\ddot{q}^r + C(q, \dot{q})\,\dot{q}^r + Q(q) + K_p\,e + K_d\,\dot{e}_r \qquad [14.69]$$

with:

- $e = q^d - q$
- $\dot{e}_r = \dot{e} + \Lambda\,e = \dot{q}^r - \dot{q}$
- $\dot{q}^r = \dot{q}^d + \Lambda\,e$
- $\Lambda = \Lambda^T > 0$

which implies that e and $\dot{e}_r$ are related through the transfer function $[sI + \Lambda]^{-1}$. The $\dot{q}^r$ vector is called the *reference velocity*.

Combining equations [14.69] and [14.1], and assuming for convenience that friction torques are either compensated or neglected, leads to the following closed-loop equation:

$$K_p\,e + K_d\,\dot{e}_r = -A(q)\,\ddot{e}_r - C(q, \dot{q})\,\dot{e}_r = \tau \qquad [14.70]$$

The corresponding system is shown in Figure 14.8. Note that, in this case, $e(t)$ is the state of the feedforward chain that is strictly positive real. Thus, the system is globally asymptotically stable.

The tracking control law [14.69] presents several interesting analogies with the position control law [14.54]. First, the terms $Q(q)$ and $K_p\,e$ modify the potential energy as in the control law [14.54]. Then, the compensations for $A(q)$ and $C(q, \dot{q})$ modify the kinetic energy in the desired sense. Finally, the term $K_d\,\dot{e}_r$ introduces a damping that contributes to satisfy the tracking objective.

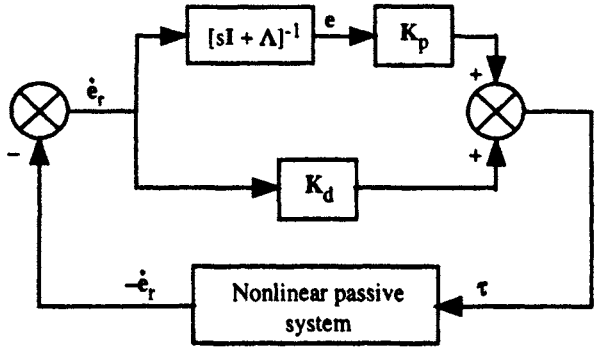

Figure 14.8. *Equivalent feedback representation of the closed-loop equation [14.70] (from [Landau 88])*

NOTE.– The computation of the passivity-based control law [14.69] cannot be achieved by the Newton-Euler inverse dynamic model due to the presence of both $\dot{\mathbf{q}}$ and $\dot{\mathbf{q}}^r$ in the expressions of the Coriolis and centrifugal forces. [Kawasaki 96] proposes an efficient algorithm for its computation.

14.5.5. *Lyapunov-based method*

In [Wen 88], we can find the demonstration of the stability of all the control laws presented in the previous sections with the definition of a suitable Lyapunov function. The exponential stability demonstrations of the following laws are also given:

$$\Gamma = \mathbf{A}(\mathbf{q})\,\ddot{\mathbf{q}}^d + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}^d)\,\dot{\mathbf{q}}^d - \frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \mathbf{K}_d\,\dot{\mathbf{e}} \qquad [14.71]$$

$$\Gamma = \mathbf{A}(\mathbf{q})\,\ddot{\mathbf{q}}^d + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}} - \frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} + \mathbf{Q}(\mathbf{q}) + \mathbf{K}_d\,\dot{\mathbf{e}} \qquad [14.72]$$

$$\Gamma = \mathbf{A}(\mathbf{q}^d)\,\ddot{\mathbf{q}}^d + \mathbf{C}(\mathbf{q}^d, \dot{\mathbf{q}}^d)\,\dot{\mathbf{q}}^d - \frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} + \mathbf{Q}(\mathbf{q}^d) + \mathbf{K}_d\,\dot{\mathbf{e}} \qquad [14.73]$$

Choosing $U^* = \frac{1}{2}\,\mathbf{e}^T\,\mathbf{K}_p\,\mathbf{e}$ results in $\frac{\partial U^*(\mathbf{q})}{\partial \mathbf{q}} = -\mathbf{K}_p\,\mathbf{e}$, but other choices are also possible. Note that equations [14.71] and [14.72] can be computed by the inverse dynamic algorithm of Newton Euler.

14.6. Adaptive control

14.6.1. *Introduction*

Since the dynamic model is often not exactly known (inaccuracies in the dynamic parameters of the robot, of the payload, high-frequency unmodeled dynamics...), the adaptive control theory has been investigated extensively as an interesting approach to estimate or adjust on-line the dynamic parameter values used in the control. The nonlinear adaptive control of rigid robot manipulators can be considered today to be mature, as is indicated by the large number of methods published over the last two decades [Bayard 88], [Ortega 89]. The different approaches of adaptive control can be classified as:

i) simplification of the dynamic model [Dubowsky 79], [Takegaki 81a];

ii) application of the adaptive techniques, which were developed for linear systems [Horowitz 80], [Nicosia 84], [Hsia 86];

iii) formulation of a nonlinear decoupling and linearizing adaptive control [Craig 86b];

iv) formulation of a nonlinear adaptive control based on the passivity property of the robot [Slotine 87], [Sadegh 87], [Landau 88], [Kelly 88];

v) formulation of parameter adaptation mechanisms that avoid joint acceleration computation as the filtered dynamic model [Middleton 88], [Li 89] or the energy-based model [El Serafi 91a].

The control laws proposed in the first two strategies are only valid for slow motion and do not take into account the full dynamics of the robot. The nonlinear adaptive control law of Craig requires joint accelerations and assumes that the estimated inertia matrix is invertible. The fourth and fifth schemes avoid the joint acceleration estimation and are, at least from a theoretical viewpoint, the most interesting.

In the next sections, we present the principles of the nonlinear linearizing adaptive control and of the passivity-based adaptive control.

14.6.2. *Adaptive feedback linearizing control*

The first version of an adaptive dynamic control has been formulated by Craig et al. [Craig 86b]. The control law has the same structure as the computed torque control law of equation [14.22], and can be written in the following form (Figure 14.9):

$$\Gamma = \mathbf{A}(\mathbf{q}, \hat{\chi})\, \mathbf{w}(t) + \mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}, \hat{\chi}) \qquad [14.74]$$

where $\hat{\chi}$ is the vector of the estimated base dynamic parameters, and:

$$\mathbf{w}(t) = \ddot{\mathbf{q}}^d + \mathbf{K}_d\, \dot{\mathbf{e}} + \mathbf{K}_p\, \mathbf{e} \qquad [14.75]$$

The control law [14.74] is associated with an on-line identification law, which provides $\hat{\chi}(t)$. For brevity, the control law will be noted:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q})\, \mathbf{w}(t) + \hat{\mathbf{H}}(\mathbf{q}, \dot{\mathbf{q}}) \qquad [14.76]$$

Combining equations [14.2], [14.3] and [14.76], leads to the closed-loop error equation (see equation [14.28]):

$$\ddot{\mathbf{e}} + \mathbf{K}_d \dot{\mathbf{e}} + \mathbf{K}_p \mathbf{e} = \hat{\mathbf{A}}^{-1}(\mathbf{q}) [\Phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \chi - \Phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \hat{\chi}] = \hat{\mathbf{A}}^{-1}(\mathbf{q}) \Phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \tilde{\chi} \quad [14.77]$$

with:

$$\tilde{\chi} = \chi - \hat{\chi} \quad [14.78]$$

Let us rewrite equation [14.77] under the state space form:

$$\dot{\mathbf{x}} = \mathbf{a}\,\mathbf{x} + \mathbf{b}\,\hat{\mathbf{A}}^{-1}(\mathbf{q})\,\Phi(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\tilde{\chi} \quad [14.79]$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{bmatrix}, \mathbf{a} = \begin{bmatrix} \mathbf{0}_n & \mathbf{I}_n \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{0}_n \\ \mathbf{I}_n \end{bmatrix} \quad [14.80]$$

where $\mathbf{0}_n$ and $\mathbf{I}_n$ are the (nxn) null matrix and identity matrix respectively.

Let us consider the following Lyapunov function candidate:

$$V = \mathbf{x}^T \mathbf{P}\,\mathbf{x} + \tilde{\chi}^T \Lambda \tilde{\chi} \quad [14.81]$$

where $\Lambda = \mathbf{diag}(\lambda_1, \lambda_2, ..., \lambda_b)$ is a positive definite adaptation gain matrix.

P is the unique positive definite matrix, which is the solution of the Lyapunov equation such that:

$$\mathbf{a}^T \mathbf{P} + \mathbf{P}\,\mathbf{a} = -\mathbf{F} \quad [14.82]$$

Differentiating V with respect to time leads to:

$$\dot{V} = -\mathbf{x}^T \mathbf{F}\,\mathbf{x} + 2\,\tilde{\chi}^T [\Phi^T(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\hat{\mathbf{A}}^{-1}(\mathbf{q})\,\mathbf{b}^T \mathbf{P}\,\mathbf{x} + \Lambda \dot{\tilde{\chi}}] \quad [14.83]$$

Assuming the following adaptation law:

$$\dot{\tilde{\chi}} = -\Lambda^{-1}\,\Phi^T(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\,\hat{\mathbf{A}}^{-1}(\mathbf{q})\,\mathbf{b}^T \mathbf{P}\,\mathbf{x} = -\dot{\hat{\chi}} \quad [14.84]$$

the expression of $\dot{V}$ becomes:

$$\dot{V} = -\mathbf{x}^T \mathbf{F}\,\mathbf{x} \leq 0 \quad [14.85]$$

Therefore, the vector **x** is bounded and $\mathbf{x} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$. Since **x** is composed of **e** and $\dot{\mathbf{e}}$, then $\mathbf{e} \rightarrow \mathbf{0}$ and $\dot{\mathbf{e}} \rightarrow \mathbf{0}$. The adaptive dynamic control algorithm given by equations [14.76] and [14.84] is thus globally asymptotically stable.

This method has two major limitations: the first is that the joint accelerations are required for implementation; the second is that the inverse of the estimated inertia matrix has to be bounded. Craig et al. [Craig 86b] suggested projection of the estimated parameters in a bounded region of the parameter space. However, this projection does not guarantee that the inverse of the inertia matrix exists.

Spong and Ortega [Spong 90] proposed a new version of this algorithm in which the condition of the invertibility of the matrix $\hat{\mathbf{A}}$ is relaxed, but the joint accelerations are still required.

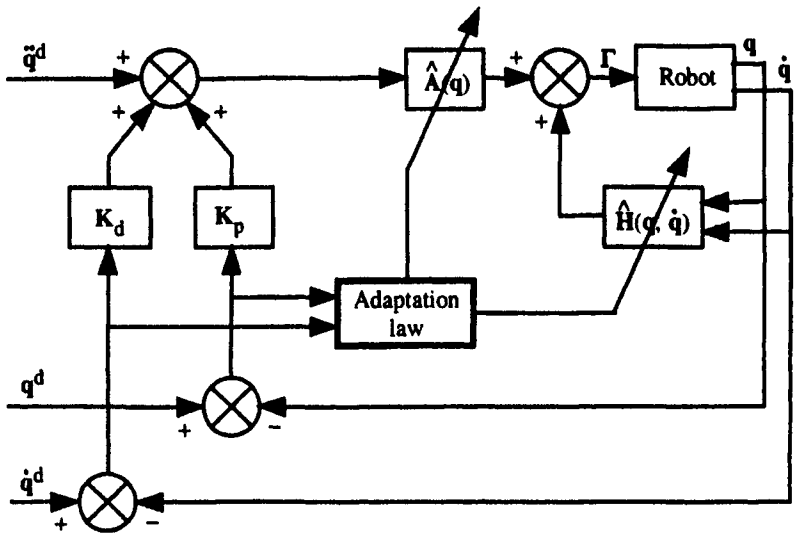

Figure 14.9. *Nonlinear adaptive control (from [Craig 86b])*

14.6.3. *Adaptive passivity-based control*

In order to develop an adaptive algorithm based on the full dynamic model, Slotine and Li [Slotine 87] exploited the property of skew-symmetry of the matrix $[\dot{\mathbf{A}} - 2\,\mathbf{C}]$. This property is a consequence of the passivity of the robot.

The control law is derived from equation [14.69] with $\mathbf{K_p} = \mathbf{0}$:

$$\Gamma = \mathbf{A}(\mathbf{q}, \hat{\chi})\,\ddot{\mathbf{q}}^r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}, \hat{\chi})\,\dot{\mathbf{q}}^r + \mathbf{Q}(\mathbf{q}, \hat{\chi}) + \mathbf{K_d}\,\dot{\mathbf{e}}_r \qquad [14.86]$$

rewritten as:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q})\,\ddot{\mathbf{q}}^r + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}}^r + \hat{\mathbf{Q}}(\mathbf{q}) + \mathbf{K_d}\,\dot{\mathbf{e}}_r \qquad [14.87]$$

with:

- $\mathbf{e} = \mathbf{q}^d - \mathbf{q}$
- $\dot{\mathbf{e}}_r = \dot{\mathbf{e}} + \Lambda \mathbf{e}$
- $\dot{\mathbf{q}}^r = \dot{\mathbf{q}} + \dot{\mathbf{e}}_r = \dot{\mathbf{q}}^d + \Lambda \mathbf{e}$

$\dot{\mathbf{e}}_r$ may be regarded as a sliding surface in the state space plane defined by $\mathbf{e}$ and $\dot{\mathbf{e}}$. To design the adaptation law, let us consider the following Lyapunov function candidate:

$$V = \frac{1}{2}[\dot{\mathbf{e}}_r^T \mathbf{A} \dot{\mathbf{e}}_r + \tilde{\chi}^T \mathbf{F} \tilde{\chi}] \qquad [14.88]$$

with:

- $\tilde{\chi} = \chi - \hat{\chi}$;
- $\hat{\chi}$: vector of the estimated base dynamic parameters;
- $\mathbf{F}$: positive definite gain adaptation matrix.

The differentiation of V with respect to time leads to:

$$\dot{V} = \frac{1}{2}\dot{\mathbf{e}}_r^T \dot{\mathbf{A}} \dot{\mathbf{e}}_r + \dot{\mathbf{e}}_r^T \mathbf{A} \ddot{\mathbf{e}}_r + \tilde{\chi}^T \mathbf{F} \dot{\tilde{\chi}} = \dot{\mathbf{e}}_r^T [\frac{1}{2}\dot{\mathbf{A}} \dot{\mathbf{e}}_r + \mathbf{A} (\ddot{\mathbf{q}}^r - \ddot{\mathbf{q}})] + \tilde{\chi}^T \mathbf{F} \dot{\tilde{\chi}} \qquad [14.89]$$

and after substitution of $\mathbf{A}\ddot{\mathbf{q}}$, using equation [14.1] while assuming no friction for sake of brevity, it becomes:

$$\dot{V} = \dot{\mathbf{e}}_r^T [\frac{1}{2}\dot{\mathbf{A}} \dot{\mathbf{e}}_r + \mathbf{A} \ddot{\mathbf{q}}^r - \Gamma + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) (\dot{\mathbf{q}}^r - \dot{\mathbf{e}}_r) + \mathbf{Q}(\mathbf{q})] + \tilde{\chi}^T \mathbf{F} \dot{\tilde{\chi}} \qquad [14.90]$$

Since $[\dot{\mathbf{A}} - 2\mathbf{C}]$ is skew-symmetric [Koditschek 84] (§ 9.3.3.3), we obtain:

$$\dot{V} = \dot{\mathbf{e}}_r^T [\mathbf{A} \ddot{\mathbf{q}}^r - \Gamma + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}^r + \mathbf{Q}(\mathbf{q})] + \tilde{\chi}^T \mathbf{F} \dot{\tilde{\chi}} \qquad [14.91]$$

Substituting the control law [14.87] in equation [14.91] yields:

$$\dot{V} = \dot{\mathbf{e}}_r^T [\tilde{\mathbf{A}} \ddot{\mathbf{q}}^r + \tilde{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}^r + \tilde{\mathbf{Q}}(\mathbf{q}) - \mathbf{K}_d \dot{\mathbf{e}}_r] + \tilde{\chi}^T \mathbf{F} \dot{\tilde{\chi}} \qquad [14.92]$$

where:

$$\tilde{\mathbf{A}} = \mathbf{A} - \hat{\mathbf{A}},\ \tilde{\mathbf{C}} = \mathbf{C} - \hat{\mathbf{C}},\ \tilde{\mathbf{Q}} = \mathbf{Q} - \hat{\mathbf{Q}} \qquad [14.93]$$

Since **A**, **C** and **Q** are linear in the dynamic parameters, we can write that:

$$\tilde{\mathbf{A}}\,\ddot{\mathbf{q}}^r + \tilde{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}}^r + \tilde{\mathbf{Q}}(\mathbf{q}) = \Phi(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}^r, \ddot{\mathbf{q}}^r)\,\tilde{\boldsymbol{\chi}} \qquad [14.94]$$

By combining equations [14.94] and [14.92], it follows that:

$$\dot{V} = -\dot{\mathbf{e}}_r^T \mathbf{K}_d\,\dot{\mathbf{e}}_r + \tilde{\boldsymbol{\chi}}^T [\mathbf{F}\,\dot{\tilde{\boldsymbol{\chi}}} + \Phi^T(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}^r, \ddot{\mathbf{q}}^r)\,\dot{\mathbf{e}}_r] \qquad [14.95]$$

Let us choose the adaptation law:

$$\dot{\tilde{\boldsymbol{\chi}}} = -\mathbf{F}^{-1}\,\Phi^T(\mathbf{q}, \dot{\mathbf{q}}, \dot{\mathbf{q}}^r, \ddot{\mathbf{q}}^r)\,\dot{\mathbf{e}}_r = -\dot{\hat{\boldsymbol{\chi}}} \qquad [14.96]$$

where the matrix $\mathbf{F}^{-1}$ is the adaptation gain. Equation [14.95] becomes:

$$\dot{V} = -\dot{\mathbf{e}}_r^T \mathbf{K}_d\,\dot{\mathbf{e}}_r \leq 0 \qquad [14.97]$$

From equation [14.97], we conclude that the control law [14.87] associated with the adaptation law [14.96] is stable.

Since $\dot{V}$ is only negative semi-definite, we cannot conclude on the asymptotic stability of the closed-loop system. Unfortunately, the La Salle invariance theorem cannot be applied to non-autonomous systems, which is the case in tracking tasks. To complete the proof of asymptotic stability, the Barbalat lemma can be used (Appendix 9).

It is worth noting that adding a proportional gain $\mathbf{K}_p\,\mathbf{e}$ to the control law [14.87] makes it possible to use the results on passivity of § 14.5.4. and to prove asymptotic stability with a Lyapunov function. Let us consider the following law [Landau 88]:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q})\,\ddot{\mathbf{q}}^r + \hat{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\,\dot{\mathbf{q}}^r + \hat{\mathbf{Q}}(\mathbf{q}) + \mathbf{K}_p\,\mathbf{e} + \mathbf{K}_d\,\dot{\mathbf{e}}_r \qquad [14.98]$$

which can be rewritten as:

$$\Gamma = A(q, \chi)\,\ddot{q}^r + C(q, \dot{q}, \chi)\,\dot{q}^r + Q(q, \chi) + K_p\,e + K_d\,\dot{e}_r - A(q, \tilde{\chi})\,\ddot{q}^r - C(q, \dot{q}, \tilde{\chi})\,\dot{q}^r - Q(q, \tilde{\chi}) \qquad [14.99]$$

the adaptation law being given by equation [14.96].

From equations [14.1] and [14.99], in the absence of friction, we can represent the system by the three interconnected blocks of Figure 14.10. Blocks B1 and B2 represent the system of Figure 14.8 whose passivity has been demonstrated in § 14.5. To demonstrate the passivity of the block B3, we must verify that:

$$\int_0^{t1} -\dot{e}_r^T(t)\,\Phi\,\tilde{\chi}\,dt \geq -\gamma_0^2 \quad \text{with } \gamma_0^2 < \infty \qquad [14.100]$$

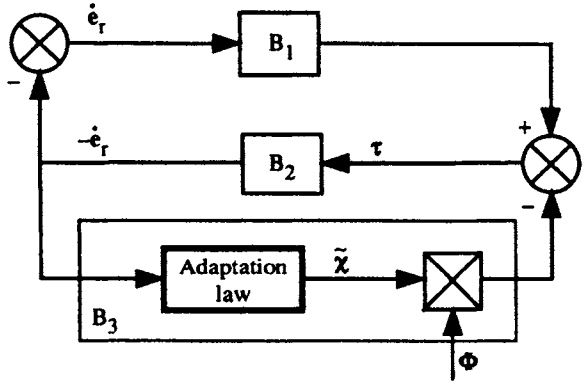

Figure 14.10. *Equivalent feedback representation of the closed-loop equation for the passivity-based adaptive control (from [Landau 88])*

From equation [14.96], and since **F** is symmetric, it follows that:

$$-\dot{e}_r^T(t)\,\Phi = \dot{\tilde{\chi}}^T\,F \qquad [14.101]$$

and using equation [14.100], we obtain:

$$\int_0^{t1} -\dot{e}_r^T(t)\,\Phi\,\tilde{\chi}\,dt = \int_0^{t1} \dot{\tilde{\chi}}^T F\,\tilde{\chi}\,dt = \int_0^{t1} \frac{1}{2}\frac{d}{dt}[\tilde{\chi}^T F\,\tilde{\chi}]\,dt \geq -\frac{1}{2}\tilde{\chi}^T(0)\,F\,\tilde{\chi}(0) \qquad [14.102]$$

Since $\mathbf{F}$ is positive definite, equation [14.100] is verified and the block B3 is passive, which demonstrates the asymptotic stability since the block B1 is strictly positive real.

To demonstrate the stability of the control law [14.98] with the adaptation law [14.96] using a Lyapunov analysis, let us choose the following Lyapunov function candidate [Sadegh 90]:

$$V = \frac{1}{2}\dot{\mathbf{e}}_r^T \mathbf{A}\, \dot{\mathbf{e}}_r + \frac{1}{2}\mathbf{e}^T \mathbf{K}_p\, \mathbf{e} + \frac{1}{2}\tilde{\boldsymbol{\chi}}^T \mathbf{F}\, \tilde{\boldsymbol{\chi}} \qquad [14.103]$$

Unlike the function [14.88], equation [14.103] is a function of the transformed state vector $[\mathbf{e}^T \quad \dot{\mathbf{e}}_r^T]^T$. It can be verified that:

$$\dot{V} = -\dot{\mathbf{e}}_r^T \mathbf{K}_d\, \dot{\mathbf{e}}_r - \mathbf{e}^T \Lambda\, \mathbf{K}_p\, \mathbf{e} \leq 0 \qquad [14.104]$$

Since $\dot{V}$ is a function of $\dot{\mathbf{e}}_r$ and $\mathbf{e}$, we can conclude that the closed-loop system is globally asymptotically stable. Note that the two control laws [14.87] and [14.98] are similar, but the second one is more practical for tuning since there is an additional gain $\mathbf{K}_p$.

The passivity-based adaptive control law does not require joint acceleration estimations. However, its drawback is that the inverse dynamics cannot be directly computed by the Newton-Euler algorithm, due to the presence of both $\dot{\mathbf{q}}$ and $\dot{\mathbf{q}}^r$ in the expressions of the Coriolis and centrifugal forces. Kawasaki [Kawasaki 96] proposes an efficient algorithm for its computation.

To avoid this computational problem, Sadegh and Horowitz [Sadegh 90] proposed to calculate both the control and adaptation laws in terms of the desired position, velocity and acceleration:

$$\Gamma = \hat{\mathbf{A}}(\mathbf{q}^d)\, \ddot{\mathbf{q}}^d + \hat{\mathbf{H}}(\mathbf{q}^d, \dot{\mathbf{q}}^d) + \mathbf{K}_d\, \dot{\mathbf{e}}_r + \mathbf{K}_p\, \mathbf{e} + \mathbf{K}_n \| \mathbf{e} \| \dot{\mathbf{e}}_r \qquad [14.105]$$

$$\dot{\hat{\boldsymbol{\chi}}} = \mathbf{F}^{-1}\, \boldsymbol{\Phi}^T(\mathbf{q}^d, \dot{\mathbf{q}}^d, \ddot{\mathbf{q}}^d)\, \dot{\mathbf{e}}_r \qquad [14.106]$$

where $\mathbf{K}_n \| \mathbf{e} \| \dot{\mathbf{e}}_r$ is an additional nonlinear feedback component to compensate for the errors introduced by the modification of the original adaptive control law.

The control law [14.105] can be computed by the Newton-Euler algorithm. The adaptation law [14.106] requires the computation of the elements of $\boldsymbol{\Phi}$ related to the dynamic parameters that have to be adapted. The corresponding computational cost of both laws for a six degree-of-freedom robot such as the Stäubli RX-90 is about 700 additions and 950 multiplications. These figures can considerably be reduced if only the parameters of the payload are adapted [El Serafi 91b].

14.7. Conclusion

Although the controllers of most present day industrial robots are merely designed from linear control theory, more advanced methods must be developed to cope with the nonlinear nature of the articulated structures, namely for applications requiring high dynamic performances (cycle time, dynamic accuracy...).

We presented in this chapter three methods achieving this objective: computed-torque or dynamic control, passivity-based control and adaptive control. The implementation of such controls requires on-line computation of the inverse dynamic model, which can be carried out according to the algorithms proposed in Chapter 9. In order to estimate the dynamic parameters, we make use of the techniques described in Chapter 12.

We assumed that the system and the controller are continuous. In practice, the control is achieved by a computer, which introduces time delays due to data acquisition and control law computation. The effect of these delays on the process performance is an issue of the sampling control theory and is out of the scope of this book. However, from an implementation viewpoint, the sampling period should be small enough with respect to the bandwidth of the mechanical system. Typically, a frequency close to 1000 Hz has been used for the controller of the Acma SR400 robot [Restrepo 96]. Note that the use of a high frequency allows us to increase the value of the feedback gains and results in a more robust control [Samson 87].

All the control laws presented in this chapter rely on the availability of joint positions and velocities. All the robots are equipped with high precision sensors for joint position measurements. On the other hand, the tachometers used for joint velocity measurements provide noisy signals. Therefore, it is better to generate the velocity signal by numerical differentiation of the position measurements. Other sophisticated techniques consist of designing a velocity observer from the input torque and the joint position data [Nicosia 90], [Canudas de Wit 92], [Berguis 93], [Khelfi 95], [Cherki 96].

In this chapter, we only considered rigid robots. For further reading about the control of robots with flexible joints, refer for example to [Benallegue 91], [Brogliato 91], [Zodiac 96].