5 Control and Manipulation

Bruno Siciliano and Luigi Villani

CONTENTS

5.1 Introduction 81
5.2 Motion Control 82
5.2.1 Joint Space Control 82
5.2.2 Control of Robots with Elastic Joints 85
5.2.3 Task Space Control 86
5.2.4 Multi-Priority Control 90
5.3 Force Control 91
5.3.1 Interaction of the End Effector with the Environment 92
5.3.2 Holonomic Constraints 93
5.3.3 Hybrid Force/Motion Control 95
5.3.4 Impedance Control 96
5.3.5 Multi-Priority Interaction 99
5.4 Future Directions and Recommended Reading 100
References 102

This chapter deals with fundamental control problems in robotic manipulation. The motion control problem is surveyed as the basis for more advanced algorithms designed for controlling lightweight robots with non-negligible joint elasticity as well as complex robotics systems with a large number of degrees of freedom. The problem of controlling the physical interaction of robots with the environment or a human is addressed as well.

5.1 INTRODUCTION

The early years of robotics—prior to the 1980s—were largely focused on manipulator arms and simple factory automation tasks: materials handling, welding, painting. Cost of computation, lack of good sensors, and lack of fundamental understanding of robot control were the primary barriers to progress.

Robotics today is a much richer field with far-ranging applications. Also, the definition of what constitutes a robot has broadened dramatically. A key role in this evolution was played by the synergy of robotics and control. On the one hand, technological advancements in sensors, computation, and actuators have motivated the development of new control algorithms. On the other hand, advancements in control have enabled solutions to challenging problems in robotics.

The purpose of this chapter is to focus on fundamental control problems in robotics with special emphasis on robotic manipulation. The motion control problem is surveyed as the basis for more advanced algorithms designed for controlling lightweight robots with non-negligible joint elasticity as well as complex robotics systems with a large number of degrees of freedom (DOF), able to execute multiple tasks at the same time. Moreover, the problem of controlling the physical interaction of robots with the environment or a human is addressed. Future directions are sketched at the end of the chapter, where a list of recommended reading is provided.

5.2 MOTION CONTROL

According to the Robot Institute of America, a robot manipulator is fundamentally a positioning device designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks. Thus, the motion control is a fundamental issue for the execution of manipulation tasks.

The problem of motion control is to determine the time behavior of the control inputs to achieve a desired motion. The control inputs are usually the motor currents but can be translated into torques or velocities for the purpose of control design. The desired motion is typically given by a reference trajectory generated by suitable motion planning algorithms. This problem is quite complex, since a manipulator is an articulated system, and, as such, the motion of one link influences the motion of the others. Moreover, the presence of non-negligible mechanical flexibility in the joints (and/or the links) of multi-DOF robots poses challenging control problems due to the possible excitation of vibrational phenomena.

In this section, we provide an overview of the main motion control techniques for rigid robot manipulators. Some hints about the control of robots with flexibility concentrated in the robot joints, which is a common situation in lightweight manipulators designed to work close to humans, are provided. Finally, a control approach based on coordinates suitable for specifying robotic tasks, i.e., the so-called task-space control, is presented.

5.2.1 JOINT SPACE CONTROL

The simplest control strategy for a n-DOF manipulator is to treat each robot joint as a single-input/single-output (SISO) system and design the controllers independently for each joint. Proportional, integral, derivative (PID) control is the most common method employed in this case. The objective of each joint controller is to ensure that the motor angle $q_m(t)$ follows a desired value $q_{md}(t)$. Coupling effects between joints due to varying configurations during motion are treated as disturbance inputs. This approach, known as *independent joint control*, works well for highly geared manipulators moving at relatively low speeds, since the large gear reduction and low speed tend to reduce the coupling effects.

Suitable linear decentralized feedforward compensation terms can be used to improve the tracking capability of desired trajectories with high values of speed and

acceleration. To further improve performance, the interaction between the joints can be compensated by a *nonlinear centralized feedforward* action. This solution still consists of n independent joint control servos; each joint controller elaborates references and measurements that refer to the single joint. The disturbance torque due to the interaction between the various joints is compensated by a centralized feedforward action that depends on the manipulator dynamic model computed using the joint desired positions, velocities, and accelerations.

Advanced control methods for robots generally aim to take into account the dynamic coupling between joints by using complete or partial knowledge of the dynamic model of the manipulator in the *feedback control* action.

By neglecting friction, elasticity in the joints or links, and other effects, the dynamic model of a n-link robot manipulator can be written in the so-called Euler–Lagrange form as

$$B(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)=\tau \tag{5.1}$$

where q is the $n\times 1$ joint vector, $B(q)$ is $n\times n$ inertia matrix, $C(q,\dot{q})\dot{q}$ is the $n\times 1$ vector of Coriolis and centrifugal torques, $g(q)$ is the $n\times 1$ vector of the gravitational torques, and τ is the $n\times 1$ vector of control torques.

The Euler–Lagrange equations fulfil a number of properties that can be suitably exploited for control design, namely:

1. Matrix $B(q)$ is symmetric and positive definite.
2. Matrix $C(q,\dot{q})$ can be chosen so that matrix $\dot{B}(q)-2C(q,\dot{q})$ is skew symmetric.
3. Equation 5.1 can be cast in a linear form with respect to a suitable $p\times 1$ vector π of dynamic parameters as $Y(q,\dot{q},\ddot{q})\pi=\tau$

An intuitive method of control, which ensures *tracking* of a desired time-varying joint trajectory $q_d(t)$, is the so-called *inverse dynamics*, or *feedback linearization* method, which computes the control torque τ as

$$\tau = B(q)r+C(q,\dot{q})\dot{q}+g(q) \tag{5.2}$$

$$r = \ddot{q}_d+K_d(\dot{q}_d-\dot{q})+K_p(q_d-q) \tag{5.3}$$

In view of Property 1, the closed loop equation in terms of the tracking error $e(t)=q_d(t)-q(t)$ satisfies the linear equation

$$\ddot{e}+K_d\dot{e}+K_pe=0$$

and therefore, the tracking error converges exponentially to zero for positive definite matrix gains K_d and K_p.

An interesting feature of the robot's dynamic model related to Property 2 is the so-called *passivity* property. Namely, the power supplied to the system $\dot{q}^T\tau$ is either stored as mechanical energy $E(t)$ or dissipated, which means that

$$\dot{q}^T\tau\geq\dot{E}(t)$$

This property can be exploited to design control laws with higher robustness in the presence of uncertainties in the dynamic model.

A first alternative to the inverse dynamics control law is the so-called *PD+ control*

$$\tau = B(q)\ddot{q}_d + C(q,\dot{q})\dot{q}_d + g(q) + K_d(\dot{q}_d - \dot{q}) + K_p(q_d - q) \tag{5.4}$$

containing a linear proportional derivative control action, which leads to a nonlinear closed loop equation. In this case, the global asymptotic convergence of the tracking error $e(t)$ can be proven by using the second Lyapunov stability method and exploits Properties 1 and 2. The advantage of the aforementioned control law is that in the case of *regulation* to a constant desired joint position q_d, Equation 5.4 reduces to

$$\tau = g(q) - K_d\dot{q} + K_p(q_d - q) \tag{5.5}$$

with positive definite matrices K_P and K_D, which is known as *PD control with gravity compensation*.

Another passivity-based motion tracking control law is the so-called *Slotine and Lie control*, defined by the equation

$$\tau = B(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + g(q) + K_D\sigma = Y(q,\dot{q},\dot{q}_r\ddot{q}_r)\pi + K_D\sigma \tag{5.6}$$

where

$$\dot{q}_r = \dot{q}_d + \Lambda_e e \quad \sigma = \dot{q}_r - \dot{q},$$

with diagonal positive definite matrices K_D and Λ_e.

If the vector of the robot's dynamic parameters π is uncertain, it can be replaced by an estimate $\hat{\pi}$.

The tracking capabilities of the control law (Equation 5.6) can be improved by adopting a *robust control* algorithm, where the estimate $\hat{\pi}$ is chosen as $\hat{\pi} = \pi_0 + \delta\pi$, with

$$Y\delta\pi = \begin{cases} -\rho\dfrac{Y^T\sigma}{\|Y^T\sigma\|}, & \text{if } \|Y^T\sigma\| > \varepsilon \\[2ex] -\rho\dfrac{Y^T\sigma}{\varepsilon}, & \text{if } \|Y^T\sigma\| \leq \varepsilon \end{cases}$$

where $\rho > 0$ is a bound on the parameter's uncertainty. In this case, uniform ultimate boundedness of the tracking error $e(t)$ can be proven, where the size of the ultimate boundedness set depends on $\varepsilon > 0$.

Alternatively an *adaptive control* law can be adopted, where the estimate $\hat{\pi}$ can be computed by using the update law

$$\dot{\hat{\pi}} = K_\pi Y^T(q,\dot{q},\dot{q}_r\ddot{q}_r)\sigma$$

with K_π positive definite matrix gain. Global convergence of the tracking errors to zero and boundedness of the parameter estimates can be proven using a Lyapunov analysis. Moreover, if the desired trajectory satisfies a suitable condition of *persistency of excitation*, the estimated parameters converge to the true parameters.

5.2.2 CONTROL OF ROBOTS WITH ELASTIC JOINTS

The assumption that robot manipulators are rigid multi-body mechanical systems simplifies the control design but in some cases, may lead to performance degradation and even unstable behavior, if mechanical flexibility cannot be neglected. In dynamic modeling, flexibility can be assumed to be concentrated at the robot joints or distributed along the robot links. In this chapter, we consider only the case of manipulators with elastic joints, which is a common situation of many lightweight robots designed to work close to humans.

A robot with elastic joints can be modeled in the Euler–Lagrange form as

$$B(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)+K(q-\theta) \;=\; 0 \tag{5.7}$$

$$M\ddot{\theta}+K(\theta-q)=\tau \tag{5.8}$$

where q is the $n\times 1$ vector of the link angular positions and θ is the $n\times 1$ vector of the motor angular positions. The quantity $\tau_J = K(\theta - q)$ is the vector of the elastic torques due to joint deformations, with K positive definite, diagonal joint stiffness matrix. Equation 5.7 is the so-called link equation describing the rigid multi-link dynamics, analogously to Equation 5.1. Equation 5.8 is the motor equation, where M is the positive definite, diagonal matrix of motor inertias (reflected through the gear ratio) and τ is the vector of the motor torques.

In the absence of gravity, the basic robotic task of moving between two arbitrary equilibrium configurations can be realized by using a simple decentralized PD control law using only feedback from the *motor variables*

$$\tau = -K_d\dot{\theta}+K_p(\theta_d-\theta) \tag{5.9}$$

with diagonal positive definite matrices K_P and K_D. In this case the equilibrium state is $\dot{q}=\dot{\theta}=0$, and $q=q_d$, $\theta=\theta_d=q_d$, i.e., with no joint deflection at steady state. Global asymptotic stability of the equilibrium can be proven as a consequence of the passivity of the system.

In the presence of gravity, the equilibrium position of the motor associated to a desired link position q_d is $\theta_d = q_d + K^{-1}g(q_d)$. In this case, global asymptotic stability is obtained by adding an extra gravity-dependent term τ_g to the PD control law (Equation 5.9). This term must match the gravity load $g(q_d)$ at steady state, and the following choices are possible, with progressively better transient performance, in the hypothesis that the joint elastic torque dominates the gravity torque:

- *Constant* compensation $\tau_g = g(q_d)$, ensuring global regulation for large enough K_p
- *On-line* compensation $\tau_g = g(\tilde{\theta}), \tilde{\theta} = \theta - K^{-1}g(q_d)$, which allows approximate compensation of the gravity effects on the links during the transient
- *Quasi-static* compensation $\tau_g = \overline{g}(\theta)$, where $\overline{g}(\theta)$ is the numerical solution of the equation $g(q)+K(q-\theta)=0$, which allows the lower bound on K_p to be removed.

It is important to notice that due to the presence of the joint elasticity, the state of the system is represented by four variables for each joint: the motor position and velocity, and the link position and velocity, i.e., quantities that are measured before and after the joint deformation. The use of the full state $(q, \dot{q}, \theta, \dot{\theta})$ in the feedback control laws allows, of course, faster and damped transient performance to be achieved. On the other hand, the use of controllers with a reduced set of measurements is particularly attractive. For this purpose, it can be shown that the only convenient partial feedback solution is that considered earlier, i.e., *feedback of the motor variables*. The other solutions, i.e. feedback of the link variables alone or the combination of motor positions and link velocities, and vice versa, must be avoided. The reason is related to the fact that the link variables and the control actuation are not physically *co-located*.

If joint *torque sensors* able to measure the elastic torque τ_J are available, the following torque feedback law can be used:

$$\tau = MM_\theta^{-1}u + (I - MM_\theta^{-1})\tau_J \tag{5.10}$$

where M_θ is a diagonal positive definite matrix and u is a new control input. Using Equation 5.10, the motor equation (5.8) becomes

$$M_\theta \ddot{\theta} + K(\theta - q) = u$$

where the matrix M_θ represents an apparent motor inertia that can be arbitrarily set. It can be easily understood that the smaller M_θ with relatively large K, the smaller the deviation from the rigid joint dynamics. In turn, torque feedback allows to neglect joint elasticity, and the control input u can be set as for a rigid robot with $\theta = q$.

5.2.3 TASK SPACE CONTROL

While the motion control problem for a robot manipulator is naturally formulated in terms of the n joint variables, i.e., in the joint space defined by the $n \times 1$ vector q, the robot's *task* is conveniently specified in terms of a vector of m coordinates x, which typically define the location of the manipulator's end effector or sometimes, of other points of interest of the robot's body. Typically, one has $n \geq m$, so that the joints can provide at least the DOF required for the end-effector task. If $n > m$ strictly, the manipulator is kinematically *redundant*. The mappings

$$x = k(q) \tag{5.11}$$

$$\dot{x} = J(q)\dot{q} \tag{5.12}$$

where Equation 5.12 can be obtained by differentiating Equation 5.11 with respect to time, are known as direct kinematics and differential kinematics, respectively. The matrix J is the $m \times n$ *task Jacobian*.

In the presence of redundant DOF, the same task can be executed with different joint motions, giving the possibility of better exploiting the workspace of the manipulator and ultimately resulting in a more versatile robotic arm.

In order to accomplish a given task, two kinds of control approaches can be considered, namely, joint space control and a task space control.

The *joint space control* approach is actually articulated in two subproblems. First, the manipulator inverse kinematics is solved to transform the desire task x_d into the corresponding joint motion q_d. Then, a joint space control scheme is designed that allows the actual motion q to track q_d.

In the *task space control* approach, the control torques are computed on the basis of the comparison between the desired task vector x_d and actual task vector x, and some kind of kinematic inversion is embedded in the feedback control loop. This implies that the algorithmic complexity of task space control schemes is usually higher than that of joint space control schemes, especially in the case of redundant robots.

Task space control approaches are usually preferred when the manipulator's task, usually defined in terms of end effector coordinates, may be subject to online modifications to accomodate unexpected events or to respond to sensor inputs. In particular, they are essential when physical interaction of the manipulator with the environment is of concern.

Task space control schemes can be split into two main categories: acceleration-based control and force-based control.

Acceleration-based control schemes start from the the second-order kinematics equation

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

which can be solved in terms of the joint space acceleration $\ddot{q}$ as

$$\ddot{q} = J^{\dagger}(\ddot{x} - \dot{J}(q)\dot{q}) + N\eta \tag{5.13}$$

where

$$J^{\dagger}(q) = W^{-1}J^{T}(JW^{-1}J^{T})^{-1}$$

is any $n \times m$ right generalized inverse of J with symmetric and positive definite weight matrix W and

$$N = I - J^{\dagger}J$$

is a $n \times n$ matrix, which projects the arbitrary vector η to the null space of J. The last term of Equation 5.13 provides the null-space accelerations corresponding to the joint motions that produce a change in the configuration of the manipulator without affecting its task accelerations.

In view of Equation 5.13, a simple task space inverse dynamics control can be defined by using control law (Equation 5.2) with vector r chosen as

$$r = \ddot{q}_{\text{task}} + \ddot{q}_{\text{null}} \tag{5.14}$$

with

$$\ddot{q}_{\text{task}} = J^{\dagger}(a - \dot{J}(q)\dot{q}), \quad \ddot{q}_{\text{null}} = N\eta \tag{5.15}$$

where

$$a = \ddot{x}_d + K_d(\dot{x}_d - \dot{x}) + K_p(x_d - x) \tag{5.16}$$

and η can be set to assign a null space task, i.e., a task set to satisfy an additional goal, besides the main task, projected in the null space of the main task through matrix N.

In the case $n = m$, the closed loop dynamics can be written in terms of the task space tracking error $\Delta x(t) = x_d(t) - x(t)$ as

$$\Delta\ddot{x} + K_d\Delta\dot{x} + K_p\Delta x = 0 \tag{5.17}$$

and the task space error converges exponentially to zero for positive definite matrix gains K_d and K_p.

In the case $n > m$, the aforementioned equation describes only the task space dynamics, while the stability of the closed loop dynamics depends also on the null space dynamics

$$N(\ddot{q} - \eta) = 0$$

The stability analysis for the null space dynamics is not easy, and different choices of the vector η can be made, depending on how redundancy is used. A practical choice just aimed at stabilizing the internal motions is $\eta = -k_\phi\dot{q}$, with $k_\phi > 0$.

Torque-based control schemes define the control torque as the sum

$$\tau = \tau_{\text{task}} + \tau_{\text{null}} \tag{5.18}$$

with τ_{task} and τ_{null} usually set as

$$\tau_{\text{task}} = J^T h_c + C(q,\dot{q})\dot{q} + g(q), \quad \tau_{\text{null}} = P^T\gamma \tag{5.19}$$

where $P = I - J^{\#}J$ is a null space projector matrix and $J^{\#}$ the dynamically consistent generalized inverse of J, obtained by setting $W = B(q)$. Vector h_c is a control generalized force acting on the task variables, while $P^T\gamma$ is a torque in charge of controlling the null-space motion.

The role of h_c and γ can be better understood by plugging Equation 5.18 into Equation 5.1 and multiplying both sides of the resulting equation once by $J^{\#T}$ and once by P^T. The resulting equations are, respectively, the task space dynamics

$$\Lambda_x(q)J\ddot{q} = h_c$$

with $\Lambda_x = (JB^{-1}J^T)^{-1}$, and the null space dynamics

$$P^T(B(q)\ddot{q} - \gamma) = 0$$

Therefore, by choosing

$$h_c = \Lambda_x(q)(a - \dot{J}(q)\dot{q})$$

with a as in Equation 5.16, the exponentially stable closed loop dynamics (Equation 5.17) is recovered. It is worth observing that the matrix $\Lambda_x(q)$ represents the inertia of the robot reflected in the task space.

It is not difficult to show that the torque-based control law is equal to the acceleration-based control law written using the inertia-weighted generalized inverse. Anyway, each of the two formulations has pros and cons, which are still debated in the robotics literature.

In the case of regulation to a constant desired task vector x_d, a simpler torque-based control law can be adopted by choosing τ_{task} in Equation 5.18 as

$$\tau_{\text{task}} = J^T K_p (x_d - x) - K_d \dot{q} + g(q) \tag{5.20}$$

which is a proportional derivative (PD) control with gravity compensation in task space.

A drawback common to both the control approaches is that the stability of the *null space dynamics* cannot be determined easily. One of the reasons is that differently from the task space dynamics, the null space dynamics is not expressed by a minimum number of equations, which should be $r = n - m$. This problem can be overcome by considering a $n \times r$ matrix $Z(q)$, such that $JZ = O$, and introducing a $r \times 1$ velocity vector ν, such that

$$\dot{q}_n = N\dot{q} = Z\nu \tag{5.21}$$

A convenient choice of ν is given by left inertia-weighted generalized inverse $\nu = Z^{\#}\dot{q} = (Z^T M Z)^{-1} Z^T M \dot{q}$. By this choice, the extended Jacobian matrix $J_E(q)$ defined as

$$\begin{pmatrix} \dot{x} \\ \nu \end{pmatrix} = J_E(q)\dot{q} = \begin{pmatrix} J(q) \\ Z^{\#}(q) \end{pmatrix} \dot{q} \tag{5.22}$$

is non-singular for full rank matrix J, and the inverse is $J_E^{-1}(q) = \left[J^{\#}(q) \;\; Z(q) \right]$.

Because of the unique relationship between the task space and the null space variables with the joint space variables given by Equation 5.22, it is possible to rewrite the dynamics equation of the robot in the new variables as

$$\Lambda_E(q) \begin{pmatrix} \ddot{x} \\ \dot{\nu} \end{pmatrix} + \mu_E(q, \dot{q}) \begin{pmatrix} \dot{x} \\ \nu \end{pmatrix} + g_E(q) = J_E^{-T} \tau \tag{5.23}$$

where the quantities Λ_E, μ_E, and g_E have the same physical meaning of the corresponding quantities of the dynamic model in the joint space (Equation 5.1) and fulfil the noticeable properties of the Euler–Lagrange equations.

Therefore, control schemes similar to the joint space algorithms presented in Section 5.2.1 can be adopted. This approach is known as *extended task space control*. It can be shown that thanks to the particular choice of the weighted generalised inverses $J^{\#}$ and $Z^{\#}$, the inertial matrix Λ_E is block diagonal, i.e.,

$$\Lambda_E(q) = J_E^{-T} M J_E^{-1} = \begin{pmatrix} \Lambda_x & 0 \\ 0 & \Lambda_\nu \end{pmatrix} \tag{5.24}$$

with $\Lambda_x(q) = (JM^{-1}J^T)^{-1}$ and $\Lambda_v(q) = Z^T MZ$; thus, the task space dynamics and the null space dynamics are inertially decoupled. Moreover, the matrices $\dot{\Lambda}_x - 2\mu_x$ and $\dot{\Lambda}_v - 2\mu_v$ are skew symmetric, with

$$\mu_E = \begin{bmatrix} \mu_x & \mu_{xv} \\ -\mu_{xv}^T & \mu_v \end{bmatrix}$$

5.2.4 MULTI-PRIORITY CONTROL

The common feature of the task space control approaches considered here is that they allow the execution of a main task together with an additional goal projected in the null space of the main task. This *null space projection* guarantees that the additional goal has a lower priority with respect to the main task; i.e., the main task is fully executed while the additional goal is fulfilled only to the extent that it does not conflict with the main task.

In conventional manipulation structures, these additional goals or *secondary tasks* are typically used to improve the value of performance criteria during motion. The most frequently considered performance objective for trajectory tracking tasks is singularity avoidance, i.e., joint configurations where the Jacobian matrix is rank deficient. In these configurations, it is impossible to generate task velocities or accelerations in certain directions. Moreover, since task space control algorithms involve the inversion of the Jacobian matrix, very high joint velocities are produced in the vicinity of these configurations, which therefore, must be avoided. Another useful secondary task is that of keeping the robot away from joint limits, or from undesired regions of the workspace, due to the presence of obstacles.

For highly redundant systems, like humanoids or multi-legged robots, multiple tasks could be arranged in priority in order to try to fulfil most of them, hopefully all of them, simultaneously. For example, in the case of humanoid robots, we may want to give higher priority to the task of respecting joint limit constraints over the task of controlling the robot's center of gravity for balance and in the end, the task of controlling the hands' motion.

The priority order among the tasks can be guaranteed by using projection matrices, like those used in the previous subsection. In fact, all the task space control approaches presented earlier can be extended to handle a task hierarchy, although the generalization is not straightforward.

In the case of acceleration-based control, let x_k be the $m_k \times 1$ vector defining the k-th priority task, where $m_k < n$ and $\dot{x}_k = J_k(q)\dot{q}$, for $k = 1, \ldots, L$ are the corresponding differential kinematics equations. The task hierarchy is defined such that $k = 1$ is top priority, and $k_a < k_b$ implies that k_a is located higher in the priority order than k_b.

A multi-priority control can be designed by using control law (Equation 5.2) with vector r defined as

$$r = \sum_{k=1}^{L} \ddot{q}_k, \quad \ddot{q}_k = \overline{J}_k^{\dagger}(a_k - \dot{J}_k\dot{q} - b_k) \tag{5.25}$$

with $a_k = \ddot{x}_{k,d} + K_{k,d}(\dot{x}_{k,d} - \dot{x}_k) + K_{k,p}(x_{k,d} - x_k)$, $x_{k,d}$ being the desired value of x_k. The quantities b_k and $\overline{J}_k$ are chosen so that the tasks x_j, with $j \geq k$, do not generate accelerations in the space of all higher-priority tasks. In detail,

$$\begin{aligned} b_k &= J_k \sum_{i=1}^{k-1} \ddot{q}_i \\ \overline{J}_k &= J_k N_{k-1} \\ N_k &= \prod_{j=1}^{k} (I - \overline{J}_j^{\dagger} \overline{J}_j) \end{aligned}$$

with $b_1 = 0$ and $N_0 = I$.

It can be verified that any symmetric positive definite weight matrix W can be used for the computation of the generalized inverse $\overline{J}_k^{\dagger}$. Moreover, if the inertia-weighted generalized inverse is used, the acceleration-based multi-priority control becomes equivalent to a torque-based multi-priority control. Similarly, the extended task space formulation generalized to the case of multiple priority tasks results in an inertially decoupled dynamic system, as in Equation 5.24.

5.3 FORCE CONTROL

Control of the physical interaction between a robot manipulator and the environment is crucial for the successful execution of a number of practical tasks in both industrial and service robotics. Typical examples in industrial settings include polishing, deburring, machining, and assembly. A variety of examples can be found also in service robotics, as the case of a humanoid robot opening a door or pulling out a drawer or of a rehabilitative robot performing assisted training of patients with movement disorders.

During contact, the environment may set constraints on the geometric paths that can be followed by the robot's end effector (kinematic constraints), as in the case of cleaning a window or turning a door handle. In other situations, the interaction occurs with a dynamic environment, as in the case of collaboration with a human. In all cases, a pure motion control strategy is not recommended.

In the presence of interaction, the higher the environment stiffness and position control accuracy are, the more easily the contact forces may rise and reach unsafe values. This drawback can be overcome by introducing compliance, either in a passive or in an active fashion, to accommodate the robot motion in response to interaction forces.

Passive compliance may be due to the structural compliance of the links, joints, or end effector or to the compliance of the position servo. Soft robot arms with elastic joints or links are purposely designed for intrinsically safe interaction with humans. In contrast, active compliance is entrusted to the control system, denoted *interaction control* or *force control*. In same cases, the measurement of the contact force and moment is required, which is fed back to the controller and used to modify or even generate online the desired motion of the robot.

The passive solution is faster than active reaction commanded by a computer control algorithm. However, the use of passive compliance alone lacks flexibility and cannot guarantee that high contact forces will never occur. Hence, the most effective solution is to use active force control (with or without force feedback) in combination with some degree of passive compliance.

Often, a *force/torque sensor* is mounted at the robot wrist to measure the exchanged forces, but other possibilities exist; for example, force sensors can be placed on the fingertips of robotic hands; also, external forces and moments can be estimated via shaft torque measurements of joint torque sensors or from the currents of the joint actuators.

The force control strategies can be grouped into two categories: those performing direct force control and those performing indirect force control. The main difference between the two categories is that the former offer the possibility of controlling the contact force and moment to a desired value, thanks to the closure of a force feedback loop; the latter instead achieve interaction control via motion control without explicit closure of a force feedback loop.

5.3.1 INTERACTION OF THE END EFFECTOR WITH THE ENVIRONMENT

The case of interaction of the end effector of a robot manipulator with the environment is considered first, which is the most common situation in the applications.

The end effector pose can be represented by the position vector p_e and the rotation matrix R_e, corresponding to the position and orientation of a frame attached to the end effector with respect to a fixed base frame.

The end effector velocity is denoted by the 6×1 *twist* vector $v_e = \left(\dot{p}_e^T \, \omega_e^T\right)^T$, where $\dot{p}_e$ is the translational velocity and ω_e the angular velocity, and can be computed from the joint velocity vector $\dot{q}$ using the linear mapping

$$v_e = J_e(q)\dot{q}$$

The matrix J_e is the end effector Jacobian, also known as *geometric Jacobian*.

The force f_e and moment m_e applied by the end effector to the environment are the components of the *wrench* vector $h_e = \left(f_e^T \; m_e^T\right)^T$. The joint torques τ_e corresponding to h_e can be computed as

$$\tau_e = J_e(q)^T h_e$$

These torques contribute to the manipulator dynamics, which can be rewritten as

$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau - J_e(q)^T h_e \tag{5.26}$$

To control the interaction of the end-effector with the environment, it is useful to define the control torque τ in terms of generalized control forces acting on the end-effector and for redundant robots, control torques that do not contribute to the end-effector motion. Namely, the torque-based control scheme (Equation 5.18) can be adopted, with

$$\tau = J_e^T h_c + P_e^T \gamma \tag{5.27}$$

where P_e is the dynamically consistent null space projection matrix. Replacing Equation 5.27 in Equation 5.26 and multiplying both sides of the resulting equation by $J_e^{\#T}$ gives

$$\Lambda_e(q)\dot{v}_e + \mu_e(q,\dot{q})v_e + \eta_e(q) = h_c - h_e \tag{5.28}$$

known as dynamic model in the *operational space*, where $\Lambda_e(q) = (J_e M^{-1} J_e^T)^{-1}$ is the 6×6 equivalent inertia matrix seen by the end-effector, $\mu_e(q,\dot{q})v_e$ is the wrench including centrifugal and Coriolis effects, and $\eta_e(q)$ is the wrench of the gravitational effects.

Equation 5.28 can be seen as a representation of the Newton's Second Law of Motion where all the generalized forces acting on the joints of the robot are reported at the end-effector. In the case of redundant robots, this equation does not represent the overall robot dynamics but must be completed by the equations of the null space dynamics.

Moreover, the full specification of the system dynamics would require also the analytic description of the interaction force and moment h_e. This is a very demanding task from a modeling viewpoint.

The design of the interaction control and the performance analysis are usually carried out under simplifying assumptions. The following two cases are considered:

1. The robot and the environment are perfectly rigid, and purely kinematics constraints are imposed by the environment, known as *holonomic constraints*.
2. the robot—rigid or elastic—interacts with a passive environment.

It is obvious that these situations are only ideal. However, the robustness of the control should be able to cope with situations where some of the ideal assumptions are relaxed. In that case, the control laws may be adapted to deal with nonideal characteristics.

5.3.2 HOLONOMIC CONSTRAINTS

In this subsection, for simplicity, only the case of non-redundant robots, with square and non-singular geometric Jacobian matrix J_e, is considered.

It is assumed that the environment is rigid and frictionless and imposes kinematic constraints to the robot's end-effector motion. These holonomic constraints reduce the dimension of the space of the feasible end-effector velocities and of the contact forces and moments. In detail, in the presence of m independent constraints ($m < 6$), the end-effector velocity belongs to a subspace of dimension $6 - m$, while the end-effector wrench belongs to a subspace of dimension, and can be expressed in the form

$$v_e = S_v(q)\nu, \quad h_e = S_f(q)\lambda$$

where ν is a suitable $(6-m) \times 1$ vector and λ is a suitable $m \times 1$ vector. Moreover, the subspaces of forces and velocity are *reciprocal*, i.e.,

$$h_e^T v_e = 0, \quad S_f^T(q) S_v(q) = 0$$

The concept of reciprocity expresses the physical fact that in the hypothesis of rigid and frictionless contact, the wrench does not cause any work against the twist.

An interaction task can be assigned in terms of a desired end-effector twist v_D and wrench h_D that are computed as

$$v_D = S_v \nu_D, \quad h_D = S_f \lambda_D$$

by specifying vectors λ_D and ν_D.

In many robotic tasks, it is possible to set an orthogonal reference frame, usually referred as a *task frame*, in which the matrices S_v and S_f are constant. The task frame can be chosen to be attached either to the end-effector or to the environment. Moreover, the interaction task is specified by assigning a desired force/torque or a desired linear/angular velocity along/about each of the frame axes.

In more complex and general situations, the matrices S_v and S_f can be found analytically, starting from the analytic expression of the holonomic constraints in the joint space

$$\phi(q) = 0 \tag{5.29}$$

where ϕ is an $m \times 1$ twice differentiable vector function, whose components are linearly independent at least locally in a neighborhood of the operating point. Constraints of the form in Equation 5.29 involving only the generalized coordinates of the system, are known as *holonomic constraints*.

Differentiation of Equation 5.29 yields

$$J_\phi(q)\dot{q} = 0 \tag{5.30}$$

where $J_\phi(q)$ is the $m \times 6$ *constraint Jacobian*. In the absence of friction, the generalized interaction forces are represented by a reaction wrench that tends to violate the constraints. This end-effector wrench produces reaction torques at the joints that can be computed using the principle of virtual work as

$$\tau_e = J_\phi^T(q)\lambda$$

where λ is an $m \times 1$ vector of *Lagrange multipliers*. The end-effector wrench corresponding to τ_e can be computed as

$$h_e = J_e^{-T}(q)\tau_e = S_f(q)\lambda \tag{5.31}$$

where

$$S_f = J_e^{-T}(q)J_\phi^{-T}(q) \tag{5.32}$$

Using Equations 5.31 and 5.32, Equation 5.30 can be rewritten in the form

$$J_\phi(q)J_e^{-1}(q)J_e(q)\dot{q} = S_f^T v_e = 0 \tag{5.33}$$

which, by virtue of Equation 5.31, is equivalent to the reciprocity condition $h_e^T v_e$. At this point, matrix S_v can be computed from the equality $S_f^T(q)S_v(q) = 0$.

5.3.3 HYBRID FORCE/MOTION CONTROL

The reciprocity of the velocity and force subspaces naturally leads to a control approach, known as *hybrid force/motion control*, aimed at controlling simultaneously both the contact force and the end-effector motion in two reciprocal subspaces.

In the presence of holonomic constraints, the external wrench can be written in the form $h_e = S_f \lambda$, and it is possible to compute the dynamics of the constrained system in terms of $6-m$ second-order equations

$$\Lambda_v(q)\dot{v} = S_v^T \left[h_c - \beta_e(q,\dot{q})\right]$$

where $\Lambda_v = S_v^T \Lambda_e S_v$ and $\beta_e(q,\dot{q}) = \mu_e(q,\dot{q})v_e + \eta_e(q)$, assuming constant matrices S_v and S_f. Moreover, the vector λ can be computed as

$$\lambda = S_f^\dagger(q)\left[h_c - \beta_e(q,\dot{q})\right]$$

$S_f^\dagger$ being a suitable pseudoinverse of matrix S_f. The aforementioned equation reveals that the contact force is a constraint force, which instantaneously depends on the applied input wrench h_c.

An inverse dynamics inner control loop can be designed by choosing the control wrench h_c as

$$h_c = \Lambda_e(q)S_v\alpha_v + S_f f_\lambda + \beta_e(q,\dot{q})$$

where α_v and f_λ are properly designed control inputs, which leads to the equations

$$\dot{v} = \alpha_v, \quad \lambda = f_\lambda$$

showing a complete decoupling between motion control and force control.

Then, the desired force $\lambda_D(t)$ can be achieved by setting

$$f_\lambda = \lambda_D(t)$$

but this choice is very sensitive to disturbance forces, since it contains no force feedback. Alternative choices are

$$f_\lambda = \lambda_D(t) + K_{P\lambda}\left[\lambda_D(t) - \lambda(t)\right]$$

or

$$f_\lambda = \lambda_D(t) + K_{I\lambda}\int_0^t \left[\lambda_D(\tau) - \lambda(\tau)\right]\mathrm{d}\tau$$

where $K_{P\lambda}$ and $K_{I\lambda}$ are suitable positive-definite matrix gains. The proportional feedback is able to reduce the force error due to disturbance forces, while the integral action is able to compensate for constant bias disturbances.

Motion control is achieved by setting

$$\alpha_v = \dot{v}_D(t) + K_{Pv}\left[v_D(t) - v(t)\right] + K_{Iv}\int_0^t \left[v_D(\tau) - v(\tau)\right]\mathrm{d}\tau$$

where K_{Pv} and K_{Iv} are suitable matrix gains. It is straightforward to show that asymptotic tracking of $v_D(t)$ and $\dot{v}_D(t)$ is ensured with exponential convergence for any choice of positive definite matrices K_{Pv} and K_{Iv}.

Notice that the implementation of force feedback requires the computation of vector λ from the measurement of the end-effector wrench h_e as $S_f^\dagger v_e$. Analogously, vector v can be computed from v_e as $S_v^\dagger v_e$, $S_v^\dagger$ being a suitable pseudoinverse of matrix S_v.

The hypothesis of rigid contact can be removed, and this implies that along some directions, both motion and force are allowed, although they are not independent. Hybrid force/motion control schemes can be defined also in this case.

5.3.4 IMPEDANCE CONTROL

The interaction force of the ideal situation considered in the previous subsection, being a constraint reaction force, is not a state variable of the system. In that case, the stability of the controlled system can be analyzed by considering only the robot dynamics and the kinematic constrains.

In more general situations, the environment is a dynamic system itself, and the overall coupled robot–environment dynamics must be considered for stability analysis and control design.

A physical interaction can be modeled as an exchange of mechanical power at an *interaction port* in the form $p(t) = h^T v$, h being the generalized interaction force and v the generalized velocity at the port.

In this perspective, a mechanical *impedance* at an interaction port can be defined as a dynamic operator that determines an output force in response to an input velocity at the same port. Vice versa, a mechanical *admittance* is a dynamic operator that determines an output velocity in response to an input force.

For linear systems, admittance is the inverse of impedance, and both can be represented as transfer functions in Laplace domain. Moreover, impedance (admittance) is analogous to electrical impedance (admittance) if we replace force with voltage and velocity with current.

In modeling the physical interaction between the robot and the environment, if one system is modeled as an impedance, the other must be modeled as an admittance, and vice versa. The roles of impedance and admittance are interchangeable, with some exceptions.

It is important to notice that the value of the two port variables characterizing the interaction, i.e, the force h and the velocity v, depends on both robot and environment dynamics, as can be easily verified considering the electric circuit analogy. On the other hand, a given impedance (or admittance) dynamic behavior can be imposed on the robot by the control independently of the dynamics of the other interacting system.

The aim of *impedance control* is to control neither the force nor the velocity during the interaction, but the dynamic relationship between force and velocity, namely, the robot's impedance or admittance.

Therefore, a successful impedance control must satisfy the following goals:

- *Robust stability* regardless of the environment dynamics, or for a certain class of environments
- *Performance*, i.e., minimal deviation of the actual robot impedance (or admittance) from the desired one.

Concerning the first goal, from the passive control theory, it is known that if the dynamics of the environment is passive, the stability of the interaction can be guaranteed, provided that the dynamics imposed on the robot is passive as well. This property, which derives from the Nyquist stability criterion for feedback interconnected linear systems extended to nonlinear passive systems using a Lyapunov-like analysis, provides only a sufficient condition. Therefore, passivity is sought to ensure robustness, but sometimes it may be too conservative, and stability can be achieved also if the interacting systems are not passive.

With reference to the second goal, a number of different schemes have been proposed to reshape the robot's natural impedance or admittance.

The simplest approach is that of imposing a suitable static relationship between the deviation of the end-effector position and orientation from a desired pose and the force exerted on the environment by using the control law

$$h_c = K_P \Delta x_{de} - K_D v_e + \eta_e(q) \tag{5.34}$$

where K_P and K_D are suitable matrix gains, and Δx_{de} is a suitable error between a desired and the actual end-effector position and orientation. The position error component of Δx_{de} can be simply chosen as $p_D - p_e$. Concerning the orientation error component, different choices are possible, which are not all equivalent, but this issue is outside the scope of this chapter.

The control input (Equation 5.34) corresponds to a wrench (force and moment) applied to the end-effector, which includes a gravity compensation term $\eta_e(q)$, a viscous damping term $K_D v_e$, and an elastic wrench provided by a virtual spring with stiffness matrix K_P (or equivalently, compliance matrix K_P^{-1}) connecting the end-effector frame with a frame of desired position and orientation. This control law is known as *stiffness control* or *compliance control*, and when the viscous damping torque is chosen in the joint space as $-K_D \dot{q}$, it coincides with the task space control law (Equation 5.20).

Using passivity, it is possible to prove the stability in the case of interaction with a passive environment. Moreover, at steady-state, the robot's end-effector has the desired elastic behavior

$$K_P \Delta x_{de} = h_e$$

It is clear that if $h_e \neq 0$, then the end-effector deviates from the desired pose, which is usually denoted as *virtual pose*.

The closed loop system (Equations 5.35 with 5.34) can be written as

$$\Lambda_e(q)\dot{v}_e + \mu_e(q,\dot{q})v_e + K_D v_e - K_P \Delta x_{de} = h_e \tag{5.35}$$

corresponding to a 6-DOF nonlinear and configuration-dependent mechanical impedance of mass-spring-damper type, with inertia (mass) matrix $\Lambda_e(q)$, adjustable damping K_D, and stiffness K_P, producing the external wrench h_e.

One problem of the aforementioned impedance control law, which does not require feedback of the interaction force and moment h_e, is that is not possible to modify the natural inertia of the robot (seen from the end-effector) or, the friction torques which have not been considered here but which can be easily added to the term $\mu_e(q,\dot{q})v_e$. Hence, the performance of the control scheme is poor in the case that the natural robot inertia and friction are dominant with respect to the desired damping and stiffness. This happens, for example, when trying to impose a low impedance on a bulky industrial robot.

A partial solution to this problem is to use feedback of the interaction force and moment h_e in the following control law, known as *impedance control*:

$$h_c = \Lambda_e(q)\alpha + \mu_e(q,\dot{q})v_e + \eta_e(q) + h_e \tag{5.36}$$

where α is chosen as

$$\alpha = \dot{v}_d + K_M^{-1}(K_D\Delta v_{de} + K_P\Delta x_{de} - h_e)$$

The following expression can be found for the closed loop system:

$$K_M\Delta\dot{v}_{de} + K_D\Delta v_{de} + K_P\Delta x_{de} = h_e \tag{5.37}$$

representing the equation of a 6-DOF configuration independent mass-spring-damper system with adjustable inertia (mass) matrix K_M, damping K_D, and stiffness K_P. This ideal behavior can be obtained only in the hypothesis that friction forces, included into the term $\mu_e(q,\dot{q})v_e$, can be completely cancelled out using Equation 5.36, which is unlikely to happen.

An alternative control scheme able to cope with friction and unmodeled dynamics is known as *admittance control*. In this case, an inner motion control loop is used to ensure tracking of a reference motion trajectory, and thus, the robot becomes an impedance producing force in response to motion. The reference trajectory for the inner motion control loop is produced by an outer controller, which uses feedback of the interaction control force and moment h_e. This latter control system is, therefore, an admittance and can be designed so as to produce, in closed loop with the motion-controlled robot, the dynamic behavior described by Equation 5.37. The main difference, with respect to the impedance control scheme, is that friction rejection is not demanded to model-based cancellation but is robustly performed by the inner motion control loop.

One common feature of the impedance and admittance control schemes described before is that they allow the natural inertia of the robot reflected to the end-effector to be modified thanks to the feedback of the interaction force and moment h_e.

In the case of rigid robots, the closed loop stability is ensured provided that the impedance matrices in the impedance equation Equation 5.37 are positive semidefinite in case of contact with a generic passive environment. For robots with non-negligible joint elasticity, considering a simple linear 1-DOF case, it can be shown

that passivity is guaranteed only if the desired mass of the impedance is higher than the mass of the robot at the link side. The reason is related to the fact that the interaction force, used in the feedback control law, and the control actuation are not physically *co-located*. The same result can be extended to the case of multi-variable liner systems, but it is not easy to prove for nonlinear systems. In this latter case, the passivity can be easily shown only if the robot's inertia is left unchanged. Therefore, to guarantee passivity in the nonlinear multi-variable case, usually the mass of the desired impedance is set equal to the natural inertia of the robot reflected to the end-effector. This conservative choice may often limit the performance.

5.3.5 MULTI-PRIORITY INTERACTION

A significant problem for robots with a high number of DOF is that of achieving a decoupled behavior among a hierarchy of tasks with different priorities, also in the presence of interaction.

An external force applied anywhere to the robot's body produces an external torque τ_{ext}, which contributes to the robot's dynamics as follows:

$$B(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)=\tau-\tau_{ext} \tag{5.38}$$

By adopting the extended task space control formulation presented in Subsection 5.2.3, the inverse dynamics control law (Equation 5.2) with r chosen as

$$r=J_E^{-1}(\begin{pmatrix}\ddot{x}_c\\ \dot{v}_c\end{pmatrix}-\dot{J}_E\dot{q})=J^{\#}(\ddot{x}_c-\dot{J}\dot{q})+Z(\dot{v}_c-\dot{Z}^{\#}\dot{q}) \tag{5.39}$$

yields the closed-loop dynamics

$$\ddot{q}=J^{\#}(\ddot{x}_c-\dot{J}\dot{q})+Z(\dot{v}_c-\dot{Z}^{\#}\dot{q})-B^{-1}(q)\tau_{ext} \tag{5.40}$$

Multiplying both sides of Equation 5.40 by $Z^{\#}$, and considering that $\dot{v}=\dot{Z}^{\#}\dot{q}+Z^{\#}\ddot{q}$, the null space closed-loop equation is obtained as

$$\dot{v}=\dot{v}_c-Z^{\#}B^{-1}\tau_{ext} \tag{5.41}$$

On the other hand, multiplying both sides of Equation 5.40 by J gives the task space closed loop equation

$$\ddot{x}=\ddot{x}_c-JB^{-1}\tau_{ext} \tag{5.42}$$

Notice that the null space velocity vector v is, in general, non-integrable, and thus, a null space position error cannot be easily defined. However, a projected joint space error $Z^T\tilde{q}$ can be used to define the null space command acceleration

$$\dot{v}_c=\dot{v}_d+\Lambda_v^{-1}((\mu_v+B_v)\tilde{v}+Z^Tk_q\tilde{q}) \tag{5.43}$$

with $k_q>0$, B_v symmetric and positive definite matrix, and $\tilde{v}=v_d-v$. The configuration-dependent quantities $\Lambda_v=Z^TMZ$ and μ_v are, respectively, the inertia matrix and the Coriolis/centrifugal matrix in the null space defined in Subsection 5.2.3. The corresponding closed loop equation is

$$\Lambda_v\dot{\tilde{v}}+(\mu_v+B_v)\tilde{v}+Z^Tk_q\tilde{q}=Z^T\tau_{ext} \tag{5.44}$$

where $Z^T \tau_{ext}$ is the projection of the external torque on the null space. Equation 5.44 can be interpreted as an impedance equation defined in the null space, with inertia Λ_v, damping B_v, and projected elastic torque $Z^T k_q \tilde{q}$.

On the other hand, the following choice for the task space command acceleration

$$\ddot{x}_c = \ddot{x}_d + \Lambda_x^{-1}((\mu_x + D)\dot{\tilde{x}} + K\tilde{x}) \tag{5.45}$$

produces the closed-loop equation

$$\Lambda_x \ddot{\tilde{x}} + (\mu_x + D)\dot{\tilde{x}} + K\tilde{x} = J^{\#T} \tau_{ext} \tag{5.46}$$

which also represents an impedance equation in task space with inertia Λ_x, damping D, and stiffness torque K.

The parameters of the two impedance equations (Equations 5.44 and 5.46) can be set depending on the expected behavior in response to forces and moments applied anywhere to the robot.

Consider, for example, the following scenario: a robot assistant composed by an anthropomorphic manipulator mounted on a mobile wheeled platform carries a glass with some water, and for some reason, a human applies a force to the robot's elbow. If the robot is compliant, and the compliance is not selective, the force may produce a change in the glass orientation, and the water can flow out. On the other hand, if the robot is controlled so that, in the task space defined as the glass (position and) orientation, the impedance is set high (i.e., low compliance), while in the null space, the impedance is set low (i.e., high compliance), then the robot's structure will comply mainly in the null space, keeping the (position and) orientation of the glass close to the desired one.

5.4 FUTURE DIRECTIONS AND RECOMMENDED READING

The motion control problem for robot manipulators was a challenging research topic from about the mid-1980s until the mid-1990s, when researchers started to exploit the structural properties of manipulator dynamics, such as feedback linearizability, passivity, linearity in the parameters, and others. Many of these methods are well assessed and can be found in standard robotics textbooks, such as Siciliano et al. [33]. An incomplete list of fundamental references where more detailed treatment of these topics can be found is: Arimoto and Miyazaki [4] for PD control with gravity compensation, Paden and Panja [27] for PD+ algorithm, Slotine and Li [35] and Ortega and Spong [24] for passivity-based adaptive control, and Abdallah et al. [1] and Spong [37] for robust control.

Research in control of robot manipulators with flexible transmissions started in the same years and is still continuing. The assumptions leading to the dynamic model for robots with elastic joints considered in this chapter were introduced in Spong [36]. The three versions of PD controller with gravity compensation were presented in Tomei [38], De Luca et al. [11] and Albu-Schäffer et al. [3]. To achieve robust control performance, special interest has been devoted to joint torque feedback

Albu-Schäffer and Hirzinger [2] and [25]. A detailed treatment of dynamic modeling and control issues for flexible robots can be found in De Luca and Book [10].

The presence of joint elasticity, at the beginning, was seen as a limiting factor for performance. Nowadays, joint elasticity is considered an explicit advantage for safe physical human–robot interaction and for locomotion. A relatively new and active area of research is the control of actuators with on-line controlled variable stiffness and damping, which exhibit properties that look appealing to build the next generation of robots and humanoids.

Another challenging research area is the control of robotic structures with a large number of joints, like hyper-redundant robots, cooperating manipulators, multifingered hands, vehicle-manipulator systems, or multiarm/multilegged robots, such as humanoid robots. The motion control problem in task space is particularly relevant for such robotic structures, because the high number of DOF at their disposal may be suitably exploited to satisfy a certain number of different tasks at the same time. This problem was initially addressed for simple manipulation structures [17] under the name of operational space control, using a torque-based approach, and then extended in more recent papers to more complex dynamics, like humanoids [18]. The multi-priority acceleration-based approach was considered in Sadeghian et al. [30], while an interesting comparison between the two methods was presented in Nakanishi et al. [21]. The decomposition of the robot dynamics in the task space and null space can be found in Oh et al. [23], and the extension to the multi-priority case is presented in Ott et al. [26].

The performance of a force-controlled robotic system or in general, of interaction control depends on the dynamics of the environment, which is very difficult to model and identify accurately. Hence, the standard performance indices used to evaluate a control system, i.e., stability, bandwidth, accuracy, and robustness, cannot be defined by considering the robotic system alone, as for the case of robot motion control, but must be always referred to the particular contact situation at hand. For this reason, although force control in industrial applications can be considered as a mature technology, standard design methodologies are not yet available.

Force control techniques are employed also in medical robotics, haptic systems, telerobotics, humanoid robotics, and micro and nano robotics. An interesting field of application is related to human-centered robotics, where control plays a key role in achieving adaptability, reaction capability, and safety. Robots and biomechatronic systems based on the novel variable impedance actuators, with physically adjustable compliance and damping, capable of reacting softly when touching the environment, necessitate the design of specific control laws [39]. The combined use of exteroceptive sensing (visual, depth, proximity, force, and tactile sensing) for reactive control in the presence of uncertainty represents another challenging research direction [28].

The two major paradigms of force control (impedance and hybrid force/motion control) presented in this section have a number of different implementations, deriving from the specific needs of the applications or exploiting more advanced control methods that have been developed in the last three decades. A description of the state of the art of the first decade is provided in Whitney [41], whereas later advancements

are presented in Gorinevsky et al. [14] and Siciliano and Villani [34]. An extensive treatment of this topic with related bibliography can be found in Villani and De Schutter [40].

The original hybrid force/position control concept was introduced in Raibert and Craig [29], based on the concepts of natural and artificial constraints proposed in Mason [19]. The inclusion on constraints in the manipulator dynamics was considered in Yoshikawa [42] and McClamroch and Wang [20], while the task frame formalism was systematically developed in De Schutter and Van Brussel [13], Bruyninckx and De Schutter [5] and De Schutter et al. [12].

The original idea of a mechanical impedance model used for controlling the interaction between the manipulator and the environment was presented in Hogan [15], while the use of controlled stiffness or compliance was introduced in Salisbury [32]. The stability of impedance and admittance control based on passivity for rigid and elastic joint robots is discussed in Colgate and Hogan [9], Newman [22], Hogan and Buerger [16] and Ott et al. Ott-08. A less conservative stability concept for interaction control, with the name of complementary stability, was proposed in Buerger and Hogan [6]. For spatial rotations and translations, the specification of task-relevant impedance (and especially stiffness) is particularly challenging [7,8].

Finally, the interaction control problem of redundant robots in a multi-priority framework is considered in Sadeghian et al. [31] and Ott et al. [26].

REFERENCES

1. C. Abdallah, D. Dawson, P. Dorato, and M. Jamshidi. Survey of robust control for rigid robots. *IEEE Control Syst. Mag.*, 11(2):24–30, 1991.
2. A. Albu-Schäffer and G. Hirzinger. A globally stable state feedback controller for flexible joint robots. *Adv. Robot.*, 15(8):799–814, 2001.
3. A. Albu-Schäffer, C. Ott, and G. Hirzinger. A unified passivity based control framework for position, torque and impedance control of flexible joint robots. *Int. J. Robot. Res.*, 26:23–39, 2007.
4. S. Arimoto and F. Miyazaki. Stability and robustness of PID feedback control for robot manipulators of sensory capability. In M. Brady and R. Paul, editors, *Robotics Research: The First International Symposium*, pages 783–799, Cambridge, MA, MIT Press, 1984.
5. H. Bruyninckx and J. De Schutter. Specification of force-controlled actions in the "task frame formalism" – a synthesis. *IEEE Trans. Robot. Autom.*, 12(4):581–589, 1996.
6. S.P. Buerger and N. Hogan. Complementary stability and loop shaping for improved human-robot interaction. *IEEE Trans. Rob.*, 23(2):232–244, 2007.
7. F. Caccavale, P. Chiacchio, A. Marino, and L. Villani. Six-DOF impedance control of dual-arm cooperative manipulators. *IEEE/ASME Trans. Mechatronics*, 13(5):576–586, 2008.
8. F. Caccavale, C. Natale, B. Siciliano, and L. Villani. Six-DOF impedance control based on angle/axis representations. *IEEE Trans. Robot. Autom.*, 15:289–300, 1999.
9. E. Colgate and N. Hogan. The interaction of robots with passive environments: Application to force feedback control. In K.J. Waldron, editor, *Advanced Robotics: 1989*, pages 465–474. Springer, Berlin Heidelberg, D, 1989.
10. A. De Luca and W. Book. Robots with flexible elements. In B. Siciliano and O. Khatib, editors, *Springer Handbook of Robotics*, pages 287–319. Springer, Berlin, D, 2008.

11. A. De Luca, B. Siciliano, and L. Zollo. PD control with on-line gravity compensation for robots with elastic joints: Theory and experiments. *Automatica*, 41(10):1809–1819, 2005.
12. J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aerbeliën, K. Claes, and H. Bruyninckx. Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty. *Int. J. Robot. Res.*, 26(5):433–455, 2007.
13. J. De Schutter and H. Van Brussel. Compliant robot motion I. A formalism for specifying compliant motion tasks. *Int. J. Robot. Res.*, 7(4):3–17, 1988.
14. D. Gorinevsky, A. Formalsky, and A. Schneider. *Force Control of Robotics Systems*. CRC Press, Boca Raton, FL, 1997.
15. N. Hogan. Impedance control: An approach to manipulation. Parts I–III. *ASME J. Dyn. Syst. Meas. Contr.*, 107:1–24, 1985.
16. N. Hogan and S.P. Buerger. Impedance and interaction control. In T.R. Kurfess, editor, *Robotics and Automation Handbook*, pages 19.1–19.24. CRC Press, New York, NY, 2004.
17. O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robot. Autom.*, 3:43–53, 1987.
18. O. Khatib, L. Sentis, J.H. Park, and J. Warren. Whole-body dynamic behavior and control of human-like robots. *Int. J. Hum. Robot.*, 1:29–43, 2004.
19. M.T. Mason. Compliance and force control for computer controlled manipulators. *IEEE Trans. Syst. Man Cybern.*, 11:418–432, 1981.
20. N.H. McClamroch and D. Wang. Feedback stabilization and tracking of constrained robots. *IEEE Trans. Autom. Contr.*, 33(5):419–426, 1988.
21. J. Nakanishi, R. Cory, M.J. Peters, and S. Schaal. Operational space control: A theoretical and empirical comparison. *Int. J. Robot. Res.*, 27:737–757, 2008.
22. W.S. Newman. Stability and performance limits of interaction controllers. *ASME J. Dyn. Syst. Meas. Control*, 114(4):563–570, 1992.
23. Y. Oh, W. Chung, and Y. Youm. Extended impedance control of redundant manipulators based on weighted decomposition of joint space. *J. Rob. Syst.*, 15(5):231–258, 1998.
24. R. Ortega and M.W. Spong. Adaptive motion control of rigid robots: A tutorial. *Auomatica*, 25:877–888, 1989.
25. C. Ott, A. Albu-Schäffer, A. Kugi, and G. Hirzinger. On the passivity based impedance control of flexible joint robots. *IEEE Trans. Rob.*, 24:416–429, 2008.
26. C. Ott, A. Dietrich, and A. Albu-Schäffer. Prioritized multi-task compliance control of redundant manipulators. *Automatica*, 53(0):416–423, 2015.
27. B. Paden and R. Panja. Globally asymptotically stable PD+ controller for robot manipulators. *Int. J. Control*, 47(6):1697–1712, 1988.
28. M. Prats, A.P. Del Pobil, and P.J. Sanz. *Robot Physical Interaction through the combination of Vision, Tactile and Force Feedback - Applications to Assistive Robotics*, volume 84 of *Springer Tracts in Advanced Robotics*. Springer, Berlin Heidelberg, D, 2013.
29. M.H. Raibert and J.J. Craig. Hybrid position/force control of manipulators. *ASME J. Dyn. Syst. Meas. Contr.*, 103:126–133, 1981.
30. H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano. Dynamic multi-priority control in redundant robotic systems. *Robotica*, 31(7):1155–1167, 2013.
31. H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano. Task-space control of robot manipulators with null-space compliance. *IEEE Trans. Robot. Autom.*, 30(2):493–506, 2014.
32. J.K. Salisbury. Active stiffness control of a manipulator in Cartesian coordinates. In *19th IEEE Conf. Decis. Contr.*, Albuquerque, NM, pages 95–100, 1980.

33. B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer, London, 2009.
34. B. Siciliano and L. Villani. *Robot Force Control*. Kluwer Academic Publishers, New York, 1999.
35. J.-J.E. Slotine and W. Li. On the adaptive control of robot manipulators. *Int. J. Robot. Res.*, 6(3):49–59, 1987.
36. M.W. Spong. Modeling and control of elastic joint robots. *ASME J. Dyn. Syst. Meas. Control*, 109(4):310–319, 1987.
37. M.W. Spong. On the robust control of robot manipulators. *IEEE Trans. Autom. Contr.*, 37:1782–1786, 1992.
38. P. Tomei. A simple pd controller for robots with elastic joints. *IEEE Trans. Autom. Contr.*, 36(10):1208–1213, 1991.
39. B. Vanderborght, A. Schäffer, A. Bicchi, E. Burdet, D.G. Caldwell, R. Carloni, M. Catalano, O. Eiberger, W. Friedl, G. Ganesh, M. Garabini, M. Grebenstein, G. Grioli, S. Haddadin, H. Hoppner, A. Jafari, M. Laffranchi, D. Lefeber, F. Petit, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, L.C. Visser, and S. Wolf. Variable impedance actuators: A review. *Robot. Auton. Syst.*, 61(12):1601–1614, 2013.
40. L. Villani and J. De Schutter. Force control. In B. Siciliano and O. Khatib, editors, *Springer Handbook of Robotics*, pages 161–185. Springer, Berlin, D, 2008.
41. D.E. Whitney. Force feedback control of manipulator fine motions. *ASME J. Dyn. Syst. Meas. Cont.*, 99:91–97, 1977.
42. T. Yoshikawa. Dynamic hybrid position/force control of robot manipulators – description of hand constraints and calculation of joint driving force. *IEEE J. Robot. Aut.*, 3:386–392, 1987.