ROBOT CONTROL 1991
(SYROCO'91)

Selected Papers from the 3rd IFAC/IFIP/IMACS Symposium, Vienna, Austria, 16 - 18 September 1991

Edited by

I. TROCH, K. DESOYER and P. KOPACEK
Technical University of Vienna, Vienna, Austria

Published for the

INTERNATIONAL FEDERATION OF AUTOMATIC CONTROL

by

PERGAMON PRESS

OXFORD • NEW YORK • SEOUL • TOKYO

HYBRID FORCE-POSITION CONTROL FOR ROBOTS IN CONTACT WITH DYNAMIC ENVIRONMENTS

A. De Luca and C. Manes

Dipartimento di Informatica e Sistemistica, Università degli Studi di Roma "La Sapienza", Via Eudossiana 18, 00184 Roma, Italy

Abstract. *A generalization of the hybrid force-position approach for controlling robots in contact with dynamic environments is presented. Classical hybrid control applies only when the environment is supposed to be stiff, acting as a kinematic constraint on the robot end-effector, and relies on the orthogonality of admissible end-effector velocities with respect to reaction forces. When considering dynamic environments, this orthogonality does not hold anymore. In order to extend the hybrid strategy to cases when the environment, besides possibly imposing kinematic constraints on the end-effector, exhibits also a dynamic behavior, a suitable description of the robot-environment interaction is introduced. In this setting, the design of a hybrid controller can be carried out so to independently handle quantities in complementary subspaces. As an interesting result, two alternative hybrid schemes arise, since either forces or positions can be controlled along properly defined dynamic directions. Issues related to planning of nominal trajectories and filtering of actual measures so to fit them with the available interaction model are also discussed.*

Keywords. *Robots; force control; hybrid control; compliant motion; task-space; decoupling.*

Introduction

Several control strategies are being investigated so to make industrially feasible all those tasks in which the robot is required to keep stable contact and to exchange forces with the environment. Two approaches have received most of the interest: *impedance control* (Hogan, 1985, 1988; Kazerooni *et al.*, 1986) and *hybrid control* (Raibert and Craig, 1981; Khatib, 1987; Yoshikawa, 1987; McClamroch and Wang, 1988; De Luca *et al.*, 1988).

In general, impedance control is suitable only for those tasks where contact forces must be kept small, typically to avoid jamming among parts in assembly or insertion operations. Contact between the robot end-effector and the environment is described as a generalized *dynamic* impedance, i.e. by a complete set of six linear or nonlinear second order differential equations. Insensitivity to unexpected interactions with the environment is obtained by compromising between contact force and position accuracy, using programmable stiffness and damping matrices. An estimate of the directions along which systematic position errors occurr is needed in order to correctly design the stiffness matrix.

Instead, hybrid control is intended for cases where the robot has to exert specified forces of arbitrary intensity against the workpiece. Using information about the environment geometry, hybrid control has the additional feature of accurate surface following, typical of tasks such as cutting, deburring, polishing or grinding. Some robustness to modeling errors is achieved by proper handling force and displacement signals at the end-effector level. As originally conceived, hybrid control strictly applies only when assuming the end-effector in contact with a stiff environment, treated as a *kinematic* constraint on the robot motion. Neglecting friction forces, an useful orthogonality exists between admissible motions and generalized contact forces.

Thus, the robot-environment interaction is usually modeled as being completely dynamic or completely kinematic. In a recent work, we have proposed a control-oriented modeling approach for robots kinematically constrained and/or dynamically coupled with the external world (De Luca and Manes, 1991a). In particular, the mechanical nature of a *dynamic environment* can be taken fully into account, while the two previous interaction classes are included as special cases. A paradigmatic example of this situation is the robotic task of turning a crank whose dynamics is not negligible.

This paper describes how to use this modeling technique for the design of hybrid control laws in the presence of environment dynamics. The basic issue is that orthogonality between end-effector motions and contact forces (Mason, 1981) does not hold anymore and the original hybrid approach cannot be

This work is supported by the *Consiglio Nazionale delle Ricerche*, grant no. 90.00381.PF67 (*Progetto Finalizzato Robotica*).

applied as such. The space of admissible motions and the space of contact forces will be determined in the case of a general robot-environment interaction. Orthogonality can then be restated between two certain reduced subspaces, in one of which it will be possible to control only positions while in the other only forces. Moreover, there exist further *dynamic directions*, in which orthogonality between forces and motions cannot be asserted. Along these dynamic directions, either a force or a position can be chosen as output to be controlled by a hybrid scheme.

The paper is organized as follows. The key ideas of hybrid control are briefly revisited so to highlight the following generalizations. Next, the modeling approach for robots in contact with a possibly dynamic environments is summarized. The planning of nominal force and position trajectories using the proposed model variables is then presented. The filtering problem that underlies any hybrid control scheme is discussed in detail in the context of dynamic environments. Finally, two alternative decoupling and linearizing hybrid control laws are given in terms of desired contact variables.

Hybrid Control Strategy

In any hybrid control scheme two basic steps can be recognized. The first one consists of processing the cartesian force and position measures, so to let them fit with the current available model of the task. In this spirit, mismatchings with the model are interpreted as 'noise' to be eliminated. Based on these 'filtered' versions of measurements, error signals are then computed with respect to desired evolutions. The second step is the design of a controller counteracting via feedback these errors, and eventually keeping them to zero with a feedforward action.

In both steps, a suitable and accurate model of the interaction task — including geometric, kinematic and dynamic aspects — greatly simplifies the problem. A general hybrid control architecture is shown in Fig. 1.

While the desired evolution of variables characterizing the task is always planned so to be compatible with the model, measured positions, velocities and forces are never consistent in practice, due to errors and approximations in modeling the real set-up. Raibert and Craig (1981) first proposed to filter the measured quantities by simply neglecting velocity components along force controlled directions, and viceversa. This operation was performed by 0/1 selection matrices. The 'core' of this approach, and then of any other hybrid strategy, is that no attempt will be made to improperly react to measures that do not match *in structure* with the task model. On the other hand, a well-designed hybrid controller will correct (force or position) errors only along 'expected' directions. Note that no identification is usually attempted. If measures are used to identify the real environment, the task model should be adapted during execution.

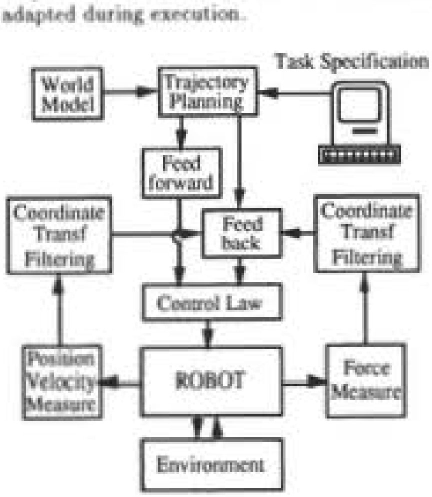

Fig. 1 – Hybrid control architecture

After the filtering and the error definition, any control law could be implemented which guarantees steady-state performance. However, schemes that avoid dynamic cross-coupling between velocity and force loops are preferred. At this level, different techniques have been proposed: local joint level control (Raibert and Craig, 1981), decoupling and linearizing control (Khatib, 1987; De Luca *et al.*, 1988), sliding mode control (De Luca et *al.*, 1989), adaptive control (Slotine and Li, 1987).

In the sequel it will be shown how a filtering procedure can be devised also in the case of a task model that includes explicitly the interaction with dynamic environments. Accordingly, a generalized decoupling and linearizing control technique will be used to impose desired force and position trajectories.

The Robot-Environment Model

The modeling approach proposed in (De Luca and Manes, 1991a), suitable for the design of hybrid control laws, is briefly summarized here.

The configurations of a rigid robot arm are identified by the joint variables vector $\mathbf{q} \in \mathbb{R}^n$. A world reference frame ${}^0S$ is fixed at the robot base, while a frame ${}^nS$ is attached to the arm tip. Let $\mathbf{r}$ be the position vector of the origin of ${}^nS$ with respect to ${}^0S$. A minimal representation is used for the orientation of ${}^nS$ with respect to ${}^0S$, e.g. Euler angles $\mathbf{o} = (\phi, \theta, \psi)$. Position and orientation can be organized in a single 6-dimensional pose vector $\mathbf{p} = (\mathbf{r}, \mathbf{o})$. As a consequence, end-effector direct and differential kinematics are defined *from the robot side* by

$$ {}^0\mathbf{p} = \mathbf{k}(\mathbf{q}), \qquad {}^0\dot{\mathbf{p}} = \frac{\partial \mathbf{k}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} = \mathbf{J}_k(\mathbf{q})\dot{\mathbf{q}}. \tag{1} $$

The superscript 0 indicates a vector expressed in ${}^0\mathcal{S}$ coordinates. The generalized end-effector velocity ${}^0\mathbf{v} = ({}^0\dot{\mathbf{r}}, {}^0\boldsymbol{\omega})$, composed of linear and angular terms, is related to ${}^0\dot{\mathbf{p}}$ by means of the transformation

$$ {}^0\mathbf{v} = \mathbf{G}({}^0\mathbf{p}){}^0\dot{\mathbf{p}}, \quad \mathbf{G}({}^0\mathbf{p}) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{O} \\ \mathbf{O} & \hat{\mathbf{G}}(\phi,\theta,\psi) \end{bmatrix}. \quad (2) $$

As a result,

$$ {}^0\mathbf{v} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}, \qquad \mathbf{J}(\mathbf{q}) = \mathbf{G}(\mathbf{k}(\mathbf{q}))\mathbf{J}_{\mathbf{k}}(\mathbf{q}), \quad (3) $$

where $\mathbf{J}(\mathbf{q})$ is the standard robot Jacobian. End-effector orientations are supposed to be nonsingular for the chosen set of angles ($\det \mathbf{G}(\mathbf{k}(\mathbf{q})) \neq 0$).

Different types of contacts are allowed between the robot end-effector and the environment. In any case, the end-effector pose can be expressed *from the environment side* in terms of a parameter vector $\mathbf{s} \in \mathbb{R}^e$. Here, a non-redundant environment is considered, i.e. $e \leq 6$. Since the choice of the vector $\mathbf{s}$ is not unique, in order to determine a convenient parametrization we will proceed as follows. A first set of variables $\mathbf{s}_D \in \mathbb{R}^d$ is needed to describe environment dynamics, when present, and will appear in the associated equations of motion. An additional set of purely kinematic variables $\mathbf{s}_K \in \mathbb{R}^k$, $k = e - d$, may be required to specify uniquely the end-effector pose ${}^0\mathbf{p}$. Defining $\mathbf{s} = (\mathbf{s}_K, \mathbf{s}_D)$, it follows that

$$ {}^0\mathbf{p} = \Gamma(\mathbf{s}), \qquad {}^0\dot{\mathbf{p}} = \frac{\partial\Gamma(\mathbf{s})}{\partial\mathbf{s}}\dot{\mathbf{s}}, \quad (4) $$

while, using (2) and (4),

$$ {}^0\mathbf{v} = \mathbf{T}(\mathbf{s})\dot{\mathbf{s}}, \qquad \mathbf{T}(\mathbf{s}) = \mathbf{G}(\Gamma(\mathbf{s}))\frac{\partial\Gamma(\mathbf{s})}{\partial\mathbf{s}}. \quad (5) $$

The parametrization of end-effector pose transfers directly to admissible velocities, allowing to separate contributions due to *kinematic* and to *dynamic* degrees of freedom of the environment. Accordingly, $\mathbf{T}(\mathbf{s})$ is partitioned as $[\mathbf{T}_K(\mathbf{s})\ \mathbf{T}_D(\mathbf{s})]$. The environment is supposed to be such that matrix $\mathbf{T}$ is always full rank, at least in the region of interest. The exchanged forces between end-effector and workpiece are collected in a 6-dimensional vector $\mathbf{F}$ of forces and torques, along and about the coordinate axes of ${}^0\mathcal{S}$. Similarly to (5), all contact forces can be parametrized as ${}^0\mathbf{F} = \mathbf{Y}(\mathbf{s})\lambda$. Since work cannot be performed on kinematic degrees of freedom, $\mathbf{Y}$ must be such that $\mathbf{Y}^T\mathbf{T}_K = 0$. Being $\mathbf{Y}$ full rank just as $\mathbf{T}$, so $\lambda \in \mathbb{R}^{(6-k)}$. At each $\mathbf{s}$, the space of admissible contact forces, $span[\mathbf{Y}(\mathbf{s})]$, can be decomposed into two parts. The subspace of static reaction forces, $span[\mathbf{Y}_R(\mathbf{s})]$, is defined by $\mathbf{T}^T\mathbf{Y}_R = \mathbf{0}$ and has dimension $6-e$. The complement, $span[\mathbf{Y}_A(\mathbf{s})]$, is a d-dimensional space of active forces, *i.e.* those responsible for energy exchange between the robot and the environment, and is such that $\mathbf{T}_D^T\mathbf{Y}_A$ is always nonsingular. We define as *dynamic directions* at the contact those which are in the span of the columns of $\mathbf{T}_D$ or $\mathbf{Y}_A$. In synthesis

$$ {}^0\mathbf{v} = \mathbf{T}(\mathbf{s})\dot{\mathbf{s}} = \mathbf{T}_D(\mathbf{s})\dot{\mathbf{s}}_D + \mathbf{T}_K(\mathbf{s})\dot{\mathbf{s}}_K, \quad (6a) $$

$$ {}^0\mathbf{F} = \mathbf{Y}(\mathbf{s})\lambda = \mathbf{Y}_A(\mathbf{s})\lambda_A + \mathbf{Y}_R(\mathbf{s})\lambda_R, \quad (6b) $$

with

$$ \begin{aligned} &\mathbf{T}_{(6\times e)} = [\mathbf{T}_{D(6\times d)}\ \mathbf{T}_{K(6\times k)}], \\ &\mathbf{Y}_{[6\times(6-k)]} = [\mathbf{Y}_{A(6\times d)}\ \mathbf{Y}_{R(6\times(6-e))}], \\ &\mathbf{T}^T\mathbf{Y}_R = \mathbf{0}_{e\times(6-e)}, \quad \mathbf{Y}^T\mathbf{T}_K = \mathbf{0}_{(6-k)\times k}, \\ &\mathbf{T}_D^T\mathbf{Y}_A:\ d\times d \text{ nonsingular matrix.} \end{aligned} \quad (7) $$

Examples of application of the above modeling are worked out in (De Luca and Manes, 1991a)

Following the Lagrangian approach, the robot-environment dynamic model can be written in the usual form as

$$ \mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{u} - \mathbf{J}^T(\mathbf{q}){}^0\mathbf{F}, \quad (8a) $$

$$ \mathbf{B}_E(\mathbf{s}_D)\ddot{\mathbf{s}}_D + \mathbf{n}_E(\mathbf{s}_D,\dot{\mathbf{s}}_D) = \mathbf{T}_D^T(\mathbf{s}){}^0\mathbf{F}. \quad (8b) $$

The symmetric and positive definite inertia matrices $\mathbf{B}(\mathbf{q})$ and $\mathbf{B}_E(\mathbf{s}_D)$ refer to the robot and the environment, and have dimensions $n\times n$ and $d\times d$. Vector $\mathbf{u}$ collects the joint input forces/torques. The coupling between robot and environment dynamics is given by the constraint on the end-effector pose, written in algebraic and differential form as

$$ \mathbf{k}(\mathbf{q}) = \Gamma(\mathbf{s}), \qquad \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{T}(\mathbf{s})\dot{\mathbf{s}}. \quad (9) $$

Equations (9) can be used in order to eliminate the explicit appearance of the force vector ${}^0\mathbf{F}$ in (8). This can be achieved in two different ways, depending on the handling of dynamic directions. In the first case, λ_A are explicited as directly depending on the robot input $\mathbf{u}$, while dynamic accelerations are seen as a consequence. In the second, $\mathbf{u}$ affects directly the accelerations $\ddot{\mathbf{s}}_D$, which cause in turn active forces parametrized by the vector λ_A. The resulting alternative dynamic relationships are (see Appendix)

$$ \mathbf{Q}(\mathbf{q},\mathbf{s})\begin{bmatrix}\lambda_A \\ \lambda_R \\ \ddot{\mathbf{s}}_K\end{bmatrix} = \mathbf{m}(\mathbf{q},\dot{\mathbf{q}},\mathbf{s},\dot{\mathbf{s}}) + \mathbf{J}\mathbf{B}^{-1}\mathbf{u}, \quad (10a) $$

$$ \ddot{\mathbf{s}}_D = -\mathbf{B}_E^{-1}\mathbf{n}_E + \mathbf{B}_E^{-1}\mathbf{T}_D^T\mathbf{Y}_A\lambda_A, \quad (10b) $$

and

$$ \hat{\mathbf{Q}}(\mathbf{q},\mathbf{s})\begin{bmatrix}\ddot{\mathbf{s}}_D \\ \lambda_R \\ \ddot{\mathbf{s}}_K\end{bmatrix} = \hat{\mathbf{m}}(\mathbf{q},\dot{\mathbf{q}},\mathbf{s},\dot{\mathbf{s}}) + \mathbf{J}\mathbf{B}^{-1}\mathbf{u}, \quad (11a) $$

$$ \lambda_A = (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{n}_E + (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{B}_E\ddot{\mathbf{s}}_D. \quad (11b) $$

Both matrices $\mathbf{Q}$ and $\hat{\mathbf{Q}}$ depend on robot and environment inertia matrices, $\mathbf{B}$ and $\mathbf{B}_E$, and jacobians, $\mathbf{J}$ and $\mathbf{T}$, while the vectors $\mathbf{m}$ and $\hat{\mathbf{m}}$ depend also on their time derivatives.

Usually, the components of $\mathbf{s}$ and λ have a physical meaning in terms of lengths or angles and, respectively, in terms of forces or torques. In particular, in the case of non-dynamic environments, $\dot{\mathbf{s}}_K$ and λ_R may directly represent tangential velocity and normal force with respect to the constraint. Therefore, the parameter space of $\mathbf{s}$ and λ can be viewed as a generalization of the *task-space* concept, often introduced in hybrid control schemes. This suggests to take these variables as controlled output for system (8).

Trajectory Planning

For hybrid tasks, the trajectory planner specifies a desired time evolution both for the end-effector position and for the contact force. Using the available model of the task, trajectories are planned so to satisfy the associated algebraic and differential constraints. In particular, the task specification is conveniently translated into a trajectory for the parameters $\mathbf{s}$ and λ, rather than for the robot variables.

For a purely kinematic environment, arbitrarily chosen time histories $\mathbf{s}_{K,des}(t)$ and $\lambda_{R,des}(t)$ give compatible cartesian position and force trajectories

$$\begin{aligned} {}^0\mathbf{p}(t) &= \Gamma(\mathbf{s}_{K,des}(t)), \\ {}^0\mathbf{F}_{des}(t) &= \mathbf{Y}_R(\mathbf{s}_{K,des}(t))\lambda_{R,des}(t). \end{aligned} \tag{12}$$

In the presence of environment dynamics, time evolutions can be assigned alternatively for the set $(\mathbf{s}_D, \mathbf{s}_K, \lambda_R)$ or for $(\lambda_A, \lambda_R, \mathbf{s}_K)$. Choosing $(\mathbf{s}_{des}(t), \lambda_{R,des}(t))$ — a specification of the end-effector position and only of static reaction forces — causes a $\lambda_{A,des}(t)$, defined according to (11b) with desired values used in the right hand terms. Conversely, assigning $(\mathbf{s}_{K,des}(t), \lambda_{des}(t))$ automatically yields a time evolution $\mathbf{s}_{D,des}(t)$ through integration of (10b), using desired values in the right hand side and starting from the actual initial conditions $\mathbf{s}_D(0), \dot{\mathbf{s}}_D(0)$.

These operations, aimed at having the complete set of desired contact/environment quantities $(\mathbf{s}_{D,des}, \mathbf{s}_{K,des}, \lambda_{A,des}, \lambda_{R,des})$, are not required in the control law. The task needs only to be parametrized in one of the two ways and the control system will be concerned only with the tracking of the specified parameters.

Measure Filtering

Assume first that, at a given time instant, the end-effector and environment configurations are known through the vector $\mathbf{s}$. It is required to find actual parameter values $(\dot{\mathbf{s}}_m, \lambda_m)$ for given measures ${}^0\mathbf{v}_m$ and ${}^0\mathbf{F}_m$, under the full rank hypothesis for matrices $\mathbf{T}$ and $\mathbf{Y}$ in (6). The subscript m stands for variables computed *from measures*. This is equivalent to solve the problem of inverting the linear contact constraints (6). In the ideal case of a perfect matching between the model and the real environment, the measured velocities ${}^0\mathbf{v}_m$ and forces ${}^0\mathbf{F}_m$ will lie, for each $\mathbf{s}$, respectively in *span*[$\mathbf{T}$] and in *span*[$\mathbf{Y}$], and the solution of the inversion problem will be unique. The presence of a mismatch with the model, and in general of measurement errors, asks instead for a minimum error solution to

$${}^0\mathbf{v}_m = \mathbf{T}\dot{\mathbf{s}}_m = [\mathbf{T}_D \quad \mathbf{T}_K]\begin{bmatrix}\dot{\mathbf{s}}_{D,m}\\ \dot{\mathbf{s}}_{K,m}\end{bmatrix}, \tag{13a}$$

$${}^0\mathbf{F}_m = \mathbf{Y}\lambda_m = [\mathbf{Y}_A \quad \mathbf{Y}_D]\begin{bmatrix}\lambda_{A,m}\\ \lambda_{R,m}\end{bmatrix}. \tag{13b}$$

To this end, an error norm has to be defined but this involves some caution in view of the non-homogeneous nature of the components of generalized velocity and force vectors. Indeed, once a norm has been chosen, a well-defined metric is introduced for comparing elements of a given set. Consider an error vector $\mathbf{e} = (\mathbf{e}_{lin}, \mathbf{e}_{ang})$ and the following norm definition

$$\|\mathbf{e}\| \triangleq (\|\mathbf{e}_{lin}\|^2 + \alpha^2\|\mathbf{e}_{ang}\|^2)^{1/2}. \tag{14}$$

This corresponds to choosing as a metric tensor the positive definite diagonal matrix

$$\mathbf{W}_\alpha = \begin{bmatrix}\mathbf{I}_{3\times3} & \mathbf{O}_{3\times3}\\ \mathbf{O}_{3\times3} & \alpha^2\mathbf{I}_{3\times3}\end{bmatrix}. \tag{15}$$

Since there is no canonical way to compare angular quantities with translational ones, the factor α in (15) represents the designer 'confidence' attached to the angular matching relative to the translational one. More in general, choosing a $\mathbf{W} > 0$ as metric tensor for velocity, one can define a scalar product and a norm as

$$<{}^0\mathbf{v}_1, {}^0\mathbf{v}_2> \triangleq {}^0\mathbf{v}_1^T\mathbf{W}\,{}^0\mathbf{v}_2, \quad \|{}^0\mathbf{v}\|^2 \triangleq {}^0\mathbf{v}^T\mathbf{W}\,{}^0\mathbf{v}. \tag{16}$$

It follows that two velocity vectors are orthogonal if ${}^0\mathbf{v}_1^T\mathbf{W}\,{}^0\mathbf{v}_2 = 0$. Note that a 'natural' orthogonality is intrinsically defined only between velocity and force vectors, being the force space *dual* to the velocity space, since it expresses the concept of null power flux supplied by a force along a given direction. For consistency with the choice (16), the induced weighting matrix for forces is $\mathbf{W}^{-1}$.

With these considerations, the two solutions to (13) minimizing the associated weighted error norms are obtained using pseudoinversion as

$$\begin{aligned} \dot{\mathbf{s}}_m &= \mathbf{T}^{\dagger}\,{}^0\mathbf{v}_m = (\mathbf{T}^T\mathbf{W}\mathbf{T})^{-1}\mathbf{T}^T\mathbf{W}\,{}^0\mathbf{v}_m, \\ \lambda_m &= \mathbf{Y}^{\dagger}\,{}^0\mathbf{F}_m = (\mathbf{Y}^T\mathbf{W}^{-1}\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{W}^{-1}\,{}^0\mathbf{F}_m. \end{aligned} \tag{17}$$

It must be stressed that (17) parametrizes only those components of ${}^0\mathbf{v}_m$ and ${}^0\mathbf{F}_m$ that are, respectively, in $span[\mathbf{T}_D\ \mathbf{T}_K]$ and in $span[\mathbf{Y}_A\ \mathbf{Y}_R]$, i.e. only those components that were modeled. Measured components along 'orthogonal' directions are filtered out. Model adherence to the real world can be estimated from the pseudoinversion residuals $\|{}^0\mathbf{v}_m - \mathbf{T}\dot{\mathbf{s}}_m\|/\|{}^0\mathbf{v}_m\|$ and $\|{}^0\mathbf{F}_m - \mathbf{Y}\lambda_m\|/\|{}^0\mathbf{F}_m\|$.

The above procedure is an extension of the 0/1 selection matrices approach used in early hybrid control schemes, and has been already applied in non-dynamic cases (Wen, 1989; Duffy, 1990). In particular, Duffy (1990) questioned the non-invariance of this filtering technique w.r.t. translation and rotation. However, it can be shown that invariant scalar products cannot be associated with positive definite metric tensors, unless a configuration dependent norm is chosen (Abbati-Marescotti *et al.*, 1990). Note that the weighting matrix $\mathbf{W}$ plays no role if ${}^0\mathbf{v}_m \in span[\mathbf{T}]$ (and, dually, if ${}^0\mathbf{F}_m \in span[\mathbf{Y}]$).

The problem of inverting constraint (4) in terms of $\mathbf{s}$ is more involved, being $\mathbf{\Gamma}(\mathbf{s})$ in general a nonlinear function. In order to obtain the parameter $\mathbf{s}$ from the *measured* pose ${}^0\mathbf{p}_m$, one has to solve at each time instant

$$\mathbf{\Gamma}(\mathbf{s}_m) = {}^0\mathbf{p}_m(t). \tag{18}$$

This solution is to be used in all the previous formulas, in particular as argument in (17). The filtering problem can be formulated as a nonlinear least squares optimization problem as

$$\min_{\mathbf{s}}\ [{}^0\mathbf{p}_m - \mathbf{\Gamma}(\mathbf{s})]^T \mathbf{A}\, [{}^0\mathbf{p}_m - \mathbf{\Gamma}(\mathbf{s})], \tag{19}$$

with $\mathbf{A} > 0$, for which a numeric procedure has to be devised. This problem is typically solved in an iterative way, updating $\mathbf{s}_m$ from the parameter value computed at the previous time instant, and starting with an initial feasible value. Ad hoc simplified solution algorithms can also be derived for particular environment kinematic structures.

Decoupling and Linearizing Control

A model-based hybrid control law is presented here, realizing input-output linearization and decoupling. Such a state-feedback control law can be derived for each choice of controlled outputs, as implicitly defined in (10) and (11). In general, it is reasonable to control force along dynamic directions (i.e. to take λ_A as output) whenever there are elastic reaction terms in the environment dynamics. Otherwise, it may be more convenient to choose the position $\mathbf{s}_D$ as output. Working with the output set $(\lambda_A, \lambda_R, \mathbf{s}_K)$, it follows from (10) that the input

$$\mathbf{u} = (\mathbf{J}\mathbf{B}^{-1})^* \left\{ \mathbf{Q} \begin{bmatrix} \lambda_{A,ref} \\ \lambda_{R,ref} \\ \mathbf{a}_{K,ref} \end{bmatrix} - \mathbf{m} \right\}, \tag{20}$$

where $\mathbf{A}^*$ is any right inverse of $\mathbf{A}$ (Boullion and Odell, 1971), transforms the closed-loop system into

$$\begin{bmatrix} \lambda_A \\ \lambda_R \\ \ddot{\mathbf{s}}_K \end{bmatrix} = \begin{bmatrix} \lambda_{A,ref} \\ \lambda_{R,ref} \\ \mathbf{a}_{K,ref} \end{bmatrix}, \qquad \ddot{\mathbf{s}}_D = -\mathbf{B}_E^{-1}\mathbf{n}_E + \mathbf{B}_E^{-1}\mathbf{T}_D^T\mathbf{Y}_A\lambda_{A,ref}, \tag{21}$$

that is decoupled and input-output linearized in terms of the new input set $(\lambda_{A,ref}, \lambda_{R,ref}, \mathbf{a}_{K,ref})$.

Alternatively, applying

$$\mathbf{u} = (\mathbf{J}\mathbf{B}^{-1})^* \left\{ \hat{\mathbf{Q}} \begin{bmatrix} \mathbf{a}_{D,ref} \\ \lambda_{R,ref} \\ \mathbf{a}_{K,ref} \end{bmatrix} - \hat{\mathbf{m}} \right\} \tag{22}$$

yields, from (11), the system

$$\begin{bmatrix} \ddot{\mathbf{s}}_D \\ \lambda_R \\ \ddot{\mathbf{s}}_K \end{bmatrix} = \begin{bmatrix} \mathbf{a}_{D,ref} \\ \lambda_{R,ref} \\ \mathbf{a}_{K,ref} \end{bmatrix}, \qquad \lambda_A = (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{n}_E + (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{B}_E\mathbf{a}_{D,ref} \tag{23}$$

whose outputs are now decoupled and linearized w.r.t. the set of inputs $(\mathbf{a}_{D,ref}, \lambda_{R,ref}, \mathbf{a}_{K,ref})$. In the above decoupling laws $\mathbf{s} = \mathbf{s}_m$ and $\dot{\mathbf{s}} = \dot{\mathbf{s}}_m$, obtained by filtering the measures of robot end-effector position and velocity. The unobservable dynamics appearing in (21) coincides with the environment dynamics, which is then required to be stable (e.g. passive). A similar requirement is not present in (23), although it is usually desirable to keep λ_A at feasible levels.

The reference values to be plugged into (20) and (22) can be generated *on the linear side* in a variety of standard techniques. Using e.g. the nominal feedforward terms, plus a proportional-derivative control for position parameters and a proportional law for force parameters, leads to

$$\begin{aligned} \mathbf{a}_{K,ref} &= \ddot{\mathbf{s}}_{K,des} + \mathbf{K}_{PK}(\mathbf{s}_{K,des} - \mathbf{s}_{K,m}) \\ &\quad + \mathbf{K}_{VK}(\dot{\mathbf{s}}_{K,des} - \dot{\mathbf{s}}_{K,m}), \\ \lambda_{R,ref} &= \lambda_{R,des} + \mathbf{K}_{FR}(\lambda_{R,des} - \lambda_{R,m}), \end{aligned} \tag{24}$$

together with

$$\lambda_{A,ref} = \lambda_{A,des} + \mathbf{K}_{FA}(\lambda_{A,des} - \lambda_{A,m}) \tag{25}$$

for (21), and together with

$$\begin{aligned} \mathbf{a}_{D,ref} &= \ddot{\mathbf{s}}_{D,des} + \mathbf{K}_{PD}(\mathbf{s}_{D,des} - \mathbf{s}_{D,m}) \\ &\quad + \mathbf{K}_{VD}(\dot{\mathbf{s}}_{D,des} - \dot{\mathbf{s}}_{D,m}) \end{aligned} \tag{26}$$

for (23). The resulting control scheme allows to shape the dynamic behavior at the contact in any desired way, but its implementation requires an accurate knowledge of the kinematics and dynamics of both the robot and the environment. This motivates the search for simpler control laws that still provide no steady-state error, keeping limited transient errors on position and force. Stability of these strategies has to be shown also when the robot is in contact with a dynamic environment (De Luca and Manes, 1991b).

References

Abbati-Marescotti, A., C. Bonivento, and C. Melchiorri (1990). On the invariance of the hybrid position/force control. *J. of Intelligent & Robotic Systems*, vol. 3, no. 3, pp. 233–250.

Boullion, T.L. and P.L. Odell (1971). *Generalized Inverse Matrices*, John Wiley.

De Luca, A., C. Manes, and F. Nicolò (1988). A task space decoupling approach to hybrid control of manipulators. *2nd IFAC Symp. on Robot Control (SYROCO'88)*, Karlsruhe, D, pp. 157–162.

De Luca, A., C. Manes, and G. Ulivi (1989). Robust hybrid dynamic control of robot arms. *28th Conf. on Decision and Control*, Tampa, FL, pp. 2641–2646.

De Luca, A. and C. Manes (1991a). On the modeling of robots in contact with a dynamic environment. *5th Int. Conf. on Advanced Robotics (ICAR'91)*, Pisa, I.

De Luca, A. and C. Manes (1991b). Stable hybrid robot control laws in the presence of dynamic environments. In preparation.

Duffy, J. (1990). The fallacy of modern hybrid control theory that is based on 'orthogonal complements' of twist and wrench spaces. *J. of Robotic Systems*, vol. 7, no. 2, pp. 139–144.

Hogan, N. (1985). Impedance control: An approach to manipulation: Part I – Theory; Part II – Implementation; Part III – Applications. *Trans. ASME J. of Dynamic Systems, Measurement, and Control*, vol. 107, no. 3, pp. 1–24.

Hogan, N. (1988). On the stability of manipulators performing contact tasks. *IEEE J. of Robotics and Automation*, vol. 4, no. 6, pp. 677–686.

Kazerooni, H., T.B. Sheridan, and P.K. Houpt (1986). Robust compliant motion for manipulators, Part I: The fundamental concepts of compliant motion. *IEEE J. of Robotics and Automation*, vol. RA-2, no. 2, pp. 83–92.

Khatib, O. (1987). A unified approach to motion and force control of robot manipulators: the operational space formulation. *IEEE J. of Robotics and Automation*, vol. RA-3, pp. 43–53.

McClamroch, N.H. and D. Wang (1988). Feedback stabilization and tracking in constrained robots. *IEEE Trans. on Automatic Control*, vol. AC-33, no. 5, pp. 419–426.

Mason, M.T. (1981). Compliance and force control for computer controlled manipulators. *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-11, no. 6, pp. 418–432.

Raibert, M.H. and J.J. Craig (1981). Hybrid position/force control of manipulators. *Trans. ASME J. of Dynamic Systems, Measurement, and Control*, vol. 102, no. 3, pp. 126–133.

Slotine, J-J.E. and W. Li (1987). Adaptive manipulator control: parameter convergence and task-space strategies. *1987 American Control Conf.*, Minneapolis, MN, pp. 828–835.

Wen, J. (1989). Position and force control of robot arms. *4th IEEE Int. Symp. on Intelligent Control*, Albany, NY, pp. 251–256.

Yoshikawa, T. (1987). Dynamic hybrid position/force control of robot manipulators — Description of hand constraints and calculation of joint driving force. *IEEE J. of Robotics and Automation*, vol. RA-3, no. 5, pp. 386–392.

Appendix

Differentiate (9) to obtain

$$\mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} = \mathbf{T}_K(\mathbf{s})\ddot{\mathbf{s}}_K + \dot{\mathbf{T}}_K(\mathbf{s},\dot{\mathbf{s}})\dot{\mathbf{s}}_K + \mathbf{T}_D(\mathbf{s})\ddot{\mathbf{s}}_D + \dot{\mathbf{T}}_D(\mathbf{s},\dot{\mathbf{s}})\dot{\mathbf{s}}_D. \quad \text{(A1)}$$

Solving (8) for accelerations (dropping term dependence)

$$\begin{aligned} \ddot{\mathbf{q}} &= \mathbf{B}^{-1}\mathbf{u} - \mathbf{B}^{-1}\mathbf{J}^{T0}\mathbf{F} - \mathbf{B}^{-1}\mathbf{n}, \\ \ddot{\mathbf{s}}_D &= \mathbf{B}_E^{-1}\mathbf{T}_D^{T0}\mathbf{F} - \mathbf{B}_E^{-1}\mathbf{n}_E, \end{aligned} \quad \text{(A2)}$$

and substituting in (A1), yields

$$\mathbf{T}_K\ddot{\mathbf{s}}_K + \dot{\mathbf{T}}_K\dot{\mathbf{s}}_K + \mathbf{T}_D\mathbf{B}_E^{-1}\mathbf{T}_D^{T0}\mathbf{F} - \mathbf{T}_D\mathbf{B}_E^{-1}\mathbf{n}_E + \dot{\mathbf{T}}_D\dot{\mathbf{s}}_D = \mathbf{J}\mathbf{B}^{-1}\mathbf{u} - \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^{T0}\mathbf{F} - \mathbf{J}\mathbf{B}^{-1}\mathbf{n} + \dot{\mathbf{J}}\dot{\mathbf{q}}. \quad \text{(A3)}$$

Introducing the force parametrization (6b) in (A3) gives (10a), with

$$\mathbf{m} = \dot{\mathbf{J}}\dot{\mathbf{q}} - \dot{\mathbf{T}}_K\dot{\mathbf{s}}_K - \dot{\mathbf{T}}_D\dot{\mathbf{s}}_D + -\mathbf{J}\mathbf{B}^{-1}\mathbf{n} + \mathbf{T}_D\mathbf{B}_E^{-1}\mathbf{n}_E, \quad \text{(A4)}$$

and matrix $\mathbf{Q}$ defined as

$$\left[(\mathbf{T}_D\mathbf{B}_E^{-1}\mathbf{T}_D^T + \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T)\mathbf{Y}_A \quad \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T\mathbf{Y}_R \quad \mathbf{T}_K\right], \quad \text{(A5)}$$

where $\mathbf{T}_D^T\mathbf{Y}_R = \mathbf{0}$ was used. Solving from (3a)

$$\lambda_A = (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{n}_E + (\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{B}_E\ddot{\mathbf{s}}_D, \quad \text{(A6)}$$

and replacing in (10a), leads to (11a) with

$$\begin{aligned} \hat{\mathbf{m}} = \dot{\mathbf{J}}\dot{\mathbf{q}} &- \dot{\mathbf{T}}_K\dot{\mathbf{s}}_K - \dot{\mathbf{T}}_D\dot{\mathbf{s}}_D - \mathbf{J}\mathbf{B}^{-1}\mathbf{n} \\ &- \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T\mathbf{Y}_A(\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{n}_E, \end{aligned} \quad \text{(A7)}$$

and $\hat{\mathbf{Q}}$ defined as

$$\left[\mathbf{T}_D + \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T\mathbf{Y}_A(\mathbf{T}_D^T\mathbf{Y}_A)^{-1}\mathbf{B}_E \quad \mathbf{J}\mathbf{B}^{-1}\mathbf{J}^T\mathbf{Y}_R \quad \mathbf{T}_K\right]. \quad \text{(A8)}$$

It can be shown that both $\mathbf{Q}$ and $\hat{\mathbf{Q}}$ are nonsingular if $\mathbf{T}$ and $\mathbf{J}^T\mathbf{Y}_R$ are full rank.