

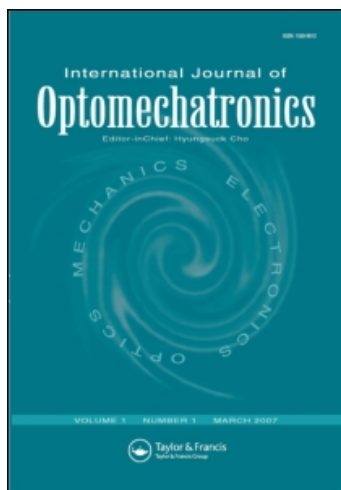
This article was downloaded by: [Baehr, Yvonne][informa internal users]

On: 7 May 2009

Access details: Access Details: [subscription number 755239602]

Publisher Taylor & Francis

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



International Journal of Optomechatronics

Publication details, including instructions for authors and subscription information:

<http://www.informaworld.com/smpp/title~content=t741771165>

Interaction Control of Robot Manipulators Using Force and Vision

Vincenzo Lippiello ^a; Bruno Siciliano ^a; Luigi Villani ^a

^a PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Napoli, Italy

Online Publication Date: 01 July 2008

To cite this Article Lippiello, Vincenzo, Siciliano, Bruno and Villani, Luigi(2008)'Interaction Control of Robot Manipulators Using Force and Vision',International Journal of Optomechatronics,2:3,257 — 274

To link to this Article: DOI: 10.1080/15599610802301599

URL: <http://dx.doi.org/10.1080/15599610802301599>

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: <http://www.informaworld.com/terms-and-conditions-of-access.pdf>

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

INTERACTION CONTROL OF ROBOT MANIPULATORS USING FORCE AND VISION

Vincenzo Lippiello, Bruno Siciliano, and Luigi Villani

PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Napoli, Italy

An approach to force and visual control of robot manipulators in contact with a partially known environment is proposed in this article. The environment is modeled as a rigid object of known geometry but of unknown and time-varying position and orientation. An algorithm for online estimation of the object pose is adopted, based on visual data provided by a camera as well as on forces measured during the interaction. This information is used by a force/position control scheme, in charge of managing the interaction. Simulation and experimental results are presented for the case of an industrial robot manipulator.

1. INTRODUCTION

The autonomy of a robotic system is strictly connected to the availability of sensing information on the external environment; among the various sensing capabilities, vision and force play a fundamental role.

In fact, visual perception allows achieving global information on the surrounding environment to be used for task planning and obstacle avoidance. On the other hand, the perception of the force applied to the end effector of a robot manipulator allows adjusting the end-effector motion so that the local constraints imposed by the environment during the interaction are satisfied.

Several approaches to control of robot manipulators combining force and vision feedback have been developed so far; e.g., hybrid visual/force control (Hosoda et al. 1998), shared and traded control (Nelson et al. 1995; Baeten and De Schutter 2004), or visual impedance control (Morel et al. 1998; Olsson et al. 2004). These algorithms improve classical interaction control schemes (Siciliano and Villani 1999), e.g., impedance control, hybrid force/position control and parallel force/position control, where only force and joint position measurements are used. A comparison of different control algorithms combining force and vision can be found in (Deng et al. 2005).

In general, interaction control approaches require knowledge of the geometry of the environment in the form of constraints imposed to the end-effector motion. Some of these approaches, like hybrid force/position control (Yoshikawa et al.

Address correspondence to Luigi Villani, PRISMA Lab, Dipartimento di Informatica e Sistemistica, Università degli Studi di Napoli Federico II, Via Claudio 21, 80125 Napoli, Italy. E-mail: lvillani@unina.it

NOMENCLATURE

f	$(k \times 1)$ feature vector	p	(3×1) position vector
h	(3×1) force vector	q	$(n \times 1)$ joint variables vector
H	(4×4) homogeneous transformation matrix	R	(3×3) rotation matrix
I_m	$(m \times m)$ identity matrix	v	(3×1) linear velocity
m	(3×1) moment vector	x	$(m \times 1)$ pose vector
n	(3×1) normal unit vector	ν	(6×1) velocity screw
o	(3×1) position vector of the origin	τ	$(n \times 1)$ joint torques vector
O_m	$(m \times m)$ null matrix	φ	$(p \times 1)$ orientation vector
		ω	(3×1) angular velocity

1988), make explicit use of the constraint equations for the design of the control law; other approaches, like impedance control (Hogan 1985) and parallel force/position control (Chiaverini and Sciacivco 1993), exploit the available knowledge of the geometry of the environment for the selection of the control gains and the choice of the reference force and position trajectories.

In the framework presented in this article, the geometry of the environment is assumed to be known, but its position and orientation with respect to the robot end-effector are unknown. The relative pose is estimated online from all the available sensor data, i.e., visual, force and joint position measurements, using the extended Kalman filter (EKF). The estimated pose is then exploited for computing online the constraint equations used by standard interaction control laws. Differently from the approaches reported in Deng et al. (2005), the resulting scheme has an inner/outer control structure where the inner control loop is a standard interaction control law based of joint positions and force feedback, while the outer loop exploits all the available measurements, i.e., joint positions, force and vision, to estimate the relative pose of the environment with respect to the robot.

The pose estimation algorithm is an extension of the visual tracking scheme proposed in Lippiello and Villani (2003) to the case that force and joint position measurements are also used. Remarkably, the same algorithm can be adopted both in free space and during the interaction, simply by modifying the measurements set of the EKF. The approach presented here was developed in previous conference papers (Lippiello et al. 2007a, b), for the case of impedance control and hybrid force/position control, respectively.

Two case studies on an industrial robot are considered. Simulation results are reported for the case of an hybrid force/position control scheme, while experimental tests are presented for the case of impedance control.

2. MODELING

Consider a robot in contact with an object, a wrist force sensor, and a camera mounted on the end effector (eye-in-hand) or fixed in the workspace (eye-to-hand). In this section, some modeling assumptions concerning the object, the robot, and the camera are presented.

2.1. Object

The position and orientation of a frame $O_o - x_o y_o z_o$ attached to a rigid object with respect to a base coordinate frame $O - xyz$ can be expressed in terms of the coordinate vector of the origin $\mathbf{o}_o$ and of the rotation matrix $\mathbf{R}_o(\boldsymbol{\varphi}_o)$, where $\boldsymbol{\varphi}_o$ is a $(p \times 1)$ vector corresponding to a suitable parametrization of the orientation. In the case that a minimal representation of the orientation is adopted, e.g., Euler angles, it is $p = 3$, while it is $p = 4$ if unit quaternions are used. Hence, the $(m \times 1)$ vector $\mathbf{x}_o = [\mathbf{o}_o^T \ \boldsymbol{\varphi}_o^T]^T$ defines a representation of the object pose with respect to the base frame in terms of $m = 3 + p$ parameters.

The homogeneous coordinate vector $\tilde{\mathbf{p}} = [\mathbf{p}^T \ 1]^T$ of a point P of the object with respect to the base frame can be computed as $\tilde{\mathbf{p}} = \mathbf{H}_o(\mathbf{x}_o) {}^o\tilde{\mathbf{p}}$, where ${}^o\tilde{\mathbf{p}}$ is the homogeneous coordinate vector of P with respect to the object frame and $\mathbf{H}_o$ is the homogeneous transformation matrix representing the pose of the object frame referred to the base frame.

It is assumed that the geometry of the object is known and that the interaction involves a portion of the external surface which satisfies a twice differentiable scalar equation $\varphi({}^o\mathbf{p}) = 0$. Hence, the unit vector normal to the surface at the point ${}^o\mathbf{p}$ and pointing outwards can be computed as

$${}^o\mathbf{n}({}^o\mathbf{p}) = \frac{(\partial\varphi({}^o\mathbf{p})/\partial {}^o\mathbf{p})^T}{\|(\partial\varphi({}^o\mathbf{p})/\partial {}^o\mathbf{p})\|} \quad (1)$$

where ${}^o\mathbf{n}$ is expressed in the object frame.

Notice that the object pose $\mathbf{x}_o$ is assumed to be unknown and may change during the task execution. As an example, a compliant contact can be modelled assuming that $\mathbf{x}_o$ changes during the interaction according to an elastic law.

A further assumption is that the contact between the robot and the object is of point type and frictionless. Therefore, when in contact, the tip point P_q of the robot instantaneously coincides with a point P of the object, so that the tip position ${}^q\mathbf{p}_q$ satisfies the surface equation $\varphi({}^o\mathbf{p}_q) = 0$. Moreover, the (3×1) contact force ${}^o\mathbf{h}$ is aligned to the normal unit vector ${}^o\mathbf{n}$.

2.2. Robot

The case of a n -joints robot manipulator is considered, with $n \geq 3$. The tip position $\mathbf{p}_q$ can be computed via the direct kinematics equation $\mathbf{p}_q = \mathbf{k}(\mathbf{q})$, where $\mathbf{q}$ is the $(n \times 1)$ vector of the joint variables. Also, the velocity of the robot's tip $\mathbf{v}_{P_q}$ can be expressed as $\mathbf{v}_{P_q} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}$ where $\mathbf{J} = \partial\mathbf{k}(\mathbf{q})/\partial\mathbf{q}$ is the robot Jacobian matrix. The vector $\mathbf{v}_{P_q}$ can be decomposed as

$${}^o\mathbf{v}_{P_q} = {}^o\dot{\mathbf{p}}_q + \mathbf{A}({}^o\mathbf{p}_q){}^o\boldsymbol{\nu}_o \quad (2)$$

with $\mathbf{A}(\cdot) = [\mathbf{I}_3 \ -\mathbf{S}(\cdot)]$, where $\mathbf{I}_3$ is the (3×3) identity matrix and $\mathbf{S}(\cdot)$ denotes the (3×3) skew-symmetric matrix operator. In Eq. (2), ${}^o\dot{\mathbf{p}}_q$ is the relative velocity of the tip point P_q with respect to the object frame while ${}^o\boldsymbol{\nu}_o = [{}^o\mathbf{v}_{O_o}^T \ {}^o\boldsymbol{\omega}_o^T]^T$ is the velocity screw characterizing the motion of the object frame with respect to the base frame in

terms of the translational velocity of the origin $\mathbf{v}_{O_o}$ and of the angular velocity $\boldsymbol{\omega}_o$. When the robot is in contact with the object, the normal component of the relative velocity ${}^o\dot{\mathbf{p}}_q$ is null, i.e., ${}^o\mathbf{n}^T({}^o\dot{\mathbf{p}}_q){}^o\dot{\mathbf{p}}_q = 0$.

2.3. Camera

A frame $O_c - x_c y_c z_c$ attached to the camera is considered. By using the classical pin-hole model, a point P of the object with coordinates ${}^c\mathbf{p} = [x \ y \ z]^T$ with respect to the camera frame is projected onto the point of the image plane with coordinates $[X \ Y]^T = \lambda_c [x/z \ y/z]^T$ where λ_c is the lens focal length.

The homogeneous coordinate vector of P with respect to the camera frame can be expressed as ${}^c\tilde{\mathbf{p}} = {}^c\mathbf{H}_o(\mathbf{x}_o, \mathbf{x}_c){}^o\tilde{\mathbf{p}}$. Notice that $\mathbf{x}_c$ is constant for eye-to-hand cameras; moreover, the matrix ${}^c\mathbf{H}_o$ does not depend on $\mathbf{x}_c$ and $\mathbf{x}_o$ separately but on the relative pose of the object frame with respect to the camera frame.

The velocity of the camera frame with respect to the base frame can be characterized in terms of the translational velocity of the origin $\mathbf{v}_{O_c}$ and of angular velocity $\boldsymbol{\omega}_c$. These vectors, expressed in camera frame, define the velocity screw ${}^c\boldsymbol{\nu}_c = [{}^c\mathbf{v}_{O_c}^T \ {}^c\boldsymbol{\omega}_c^T]^T$. It can be shown that the velocity screw ${}^c\boldsymbol{\nu}_o = [{}^c\mathbf{v}_{O_o}^T \ {}^c\boldsymbol{\omega}_o^T]^T$ corresponding to the absolute motion of the object frame can be expressed as

$${}^c\boldsymbol{\nu}_o = {}^c\boldsymbol{\nu}_{o,c} + \boldsymbol{\Gamma}({}^c\mathbf{o}_o){}^c\boldsymbol{\nu}_c \quad (3)$$

where ${}^c\boldsymbol{\nu}_{o,c} = [{}^c\dot{\mathbf{o}}_o^T \ {}^c\boldsymbol{\omega}_{o,c}^T]^T$ is the velocity screw corresponding to the relative motion of the object frame with respect to camera frame, and the matrix $\boldsymbol{\Gamma}(\cdot)$ is

$$\boldsymbol{\Gamma}(\cdot) = \begin{bmatrix} \mathbf{I}_3 & -\mathbf{S}(\cdot) \\ \mathbf{O}_3 & \mathbf{I}_3 \end{bmatrix}$$

being $\mathbf{O}_3$ the (3×3) null matrix.

The velocity screw ${}^r\boldsymbol{\nu}_s$ of a frame s with respect to a frame r can be expressed in terms of the time derivative of the vector $\mathbf{x}_s$ representing the pose of frame s through the equation

$${}^r\boldsymbol{\nu}_s = {}^r\mathbf{L}(\mathbf{x}_s)\dot{\mathbf{x}}_s \quad (4)$$

where ${}^r\mathbf{L}(\cdot)$ is a Jacobian matrix depending on the particular choice of coordinates for the orientation.

3. OBJECT POSE ESTIMATION

When the robot moves in free space, the unknown object pose can be estimated online by using vision; when the robot is in contact with the target object, also the force measurements and the joint position measurements can be used. In this section, the equations mapping the measurements to the unknown position and orientation of the object are derived. Then, the estimation algorithm based on the EKF is presented.

3.1. Vision

Vision is used to measure the image features, characterized by a set of scalar parameters f_j grouped in a vector $\mathbf{f} = [f_1, \dots, f_k]^T$. The mapping from the position and orientation of the object to the corresponding image feature vector can be computed using the projective geometry of the camera and can be written in the form

$$\mathbf{f} = \mathbf{g}_f({}^c\mathbf{H}_o(\mathbf{x}_o, \mathbf{x}_c)) \quad (5)$$

where only the dependence from the relative pose of the object frame with respect to camera frame has been explicitly evidenced. For the estimation of the object pose using EKF, it is required the computation of the Jacobian matrix $\mathbf{J}_f = \partial \mathbf{g}_f / \partial \mathbf{x}_o$. To this purpose, the time derivative of Eq. (5) can be computed in the form

$$\dot{\mathbf{f}} = \frac{\partial \mathbf{g}_f}{\partial \mathbf{x}_o} \dot{\mathbf{x}}_o + \frac{\partial \mathbf{g}_f}{\partial \mathbf{x}_c} \dot{\mathbf{x}}_c \quad (6)$$

where the second term in the right hand side is null for eye-to-hand cameras. On the other hand, the time derivative of Eq. (5) can be expressed also in the form $\dot{\mathbf{f}} = \mathbf{J}_{o,c} {}^c\boldsymbol{\nu}_{o,c}$ where the matrix $\mathbf{J}_{o,c}$ is the Jacobian mapping the relative velocity screw of the object frame with respect to the camera frame into the variation of the image feature parameters. The expression of $\mathbf{J}_{o,c}$ depends on the choice of the image features; examples of computation can be found in Espiau et al. (1992). Taking into account the velocity composition Eq. (3), vector $\dot{\mathbf{f}}$ can be rewritten in the form

$$\dot{\mathbf{f}} = \mathbf{J}_{o,c} {}^c\boldsymbol{\nu}_o - \mathbf{J}_c {}^c\boldsymbol{\nu}_c \quad (7)$$

where $\mathbf{J}_c = \mathbf{J}_{o,c} \boldsymbol{\Gamma}({}^c\mathbf{o}_o)$ is the Jacobian corresponding to the contribution of the absolute velocity screw of the camera frame, known in the literature as interaction matrix (Espiau et al. 1992). Considering Eq. (4), the comparison of Eq. (7) with Eq. (6) yields

$$\mathbf{J}_f = \mathbf{J}_{o,c} {}^c\mathbf{L}(\mathbf{x}_o) \quad (8)$$

3.2. Force

In the case of frictionless point contact, the measure of the force $\mathbf{h}$ at the robot tip during the interaction can be used to compute the unit vector normal to the object surface at the contact point ${}^o\mathbf{p}_q$, i.e.,

$$\mathbf{n}_h = \mathbf{h} / \|\mathbf{h}\| \quad (9)$$

On the other hand, vector $\mathbf{n}_h$ can be expressed as a function of the object pose $\mathbf{x}_o$ and of the robot position $\mathbf{p}_q$ in the form

$$\mathbf{n}_h = \mathbf{R}_o {}^o\mathbf{n}({}^o\mathbf{p}_q) = \mathbf{g}_h(\mathbf{x}_o, \mathbf{p}_q) \quad (10)$$

being ${}^o\mathbf{p}_q = \mathbf{R}_o^T(\mathbf{p}_q - \mathbf{o}_o)$. For the estimation of the object pose, it is required the computation of the Jacobian matrix $\mathbf{J}_h = \partial \mathbf{g}_h / \partial \mathbf{x}_o$. To this purpose, the time

derivative of Eq. (10) can be expressed as

$$\dot{\mathbf{n}}_h = \frac{\partial \mathbf{g}_h}{\partial \mathbf{x}_o} \dot{\mathbf{x}}_o + \frac{\partial \mathbf{g}_h}{\partial \mathbf{p}_q} \dot{\mathbf{p}}_q \quad (11)$$

but also as

$$\dot{\mathbf{n}}_h = \dot{\mathbf{R}}_o {}^o\mathbf{n}({}^o\mathbf{p}_q) + \mathbf{R}_o {}^oN({}^o\mathbf{p}_q) {}^o\dot{\mathbf{p}}_q \quad (12)$$

where ${}^oN({}^o\mathbf{p}_q) = \partial {}^o\mathbf{n} / \partial {}^o\mathbf{p}_q$ depends on the surface curvature and ${}^o\dot{\mathbf{p}}_q$ can be computed as

$${}^o\dot{\mathbf{p}}_q = \mathbf{R}_o^T (\dot{\mathbf{p}}_q - \dot{\mathbf{o}}_o + \mathbf{S}(\mathbf{p}_q - \mathbf{o}_o) \boldsymbol{\omega}_o)$$

Hence, replacing the above expression in Eq. (12) and taking into account the equality $\mathbf{R}_o {}^o\mathbf{n}({}^o\mathbf{p}_q) = -\mathbf{S}(\mathbf{n}_h) \boldsymbol{\omega}_o$, the following equality holds:

$$\dot{\mathbf{n}}_h = N \dot{\mathbf{p}}_q - N \dot{\mathbf{o}}_o - (\mathbf{S}(\mathbf{n}_h) - N \mathbf{S}(\mathbf{p}_q - \mathbf{o}_o)) \boldsymbol{\omega}_o \quad (13)$$

where $N = \mathbf{R}_o {}^oN({}^o\mathbf{p}_q) \mathbf{R}_o^T$. Considering Eq. (4), the comparison of Eq. (11) with Eq. (13) yields

$$\mathbf{J}_h = -[N \quad \mathbf{S}(\mathbf{n}_h) - N \mathbf{S}(\mathbf{p}_q - \mathbf{o}_o)] \mathbf{L}(\mathbf{x}_o) \quad (14)$$

3.3. Joint Positions

The measurement of the joint position vector $\mathbf{q}$ can be used to evaluate the position of the point P of the object when in contact with the robot's tip point P_q , using the direct kinematics equation. In particular, it is significant computing the scalar

$$\delta_{hq} = \mathbf{n}_h^T \mathbf{p}_q = g_{hq}(\mathbf{x}_o, \mathbf{p}_q) \quad (15)$$

using also the force measurements via Eq. (9). This quantity represents the component of the position vector of the robot's tip point along the constrained direction $\mathbf{n}_h$. The corresponding infinitesimal displacement is the same for the robot and for the object (assuming that contact is preserved). This is not true for the other components of the tip position, which do not provide any useful information on the object motion because the robot's tip may slide on the object surface; therefore, these components are not useful to estimate the pose of the object.

The EKF requires the computation of the Jacobian matrix $\mathbf{J}_{hq} = \partial g_{hq} / \partial \mathbf{x}_o$. The time derivative of δ_{hq} can be expressed as

$$\dot{\delta}_{hq} = \frac{\partial g_{hq}}{\partial \mathbf{x}_o} \dot{\mathbf{x}}_o + \frac{\partial g_{hq}}{\partial \mathbf{p}_q} \dot{\mathbf{p}}_q \quad (16)$$

but also as $\dot{\delta}_{hq} = \dot{\mathbf{n}}_h^T \mathbf{p}_q + \mathbf{n}_h^T \mathbf{R}_o ({}^o\dot{\mathbf{p}}_q + \mathbf{A}({}^o\mathbf{p}_q) {}^o\boldsymbol{\nu}_o)$ where the expression of the absolute velocity of the point P_q in Eq. (2) has been used. Using the identity ${}^o\mathbf{n}^T ({}^o\mathbf{p}_q) {}^o\dot{\mathbf{p}}_q = 0$,

this equation can be rewritten as

$$\dot{\delta}_{hq} = \mathbf{p}_q^T \dot{\mathbf{h}}_h + \mathbf{n}_h^T \mathbf{A}(\mathbf{p}_q - \mathbf{o}_o) \boldsymbol{\nu}_o \quad (17)$$

Hence, comparing Eq. (16) with Eq. (17) and taking into account Eqs. (4), (12), and (14), the following expression can be found.

$$\mathbf{J}_{hq} = \mathbf{p}_q^T \mathbf{J}_h + \mathbf{n}_h^T \mathbf{A}(\mathbf{p}_q - \mathbf{o}_o) \mathbf{L}(\mathbf{x}_o) \quad (18)$$

3.4. Extended Kalman Filter

The pose vector $\mathbf{x}_o$ of the object with respect to the base frame can be estimated using an extended Kalman filter.

To this purpose, a discrete-time state space dynamic model has to be considered, describing the object motion. The state vector of the dynamic model is chosen as $\mathbf{w} = [\mathbf{x}_o^T \quad \dot{\mathbf{x}}_o^T]^T$. For simplicity, the object velocity is assumed to be constant over one sample period T_s . This approximation is reasonable in the hypothesis that T_s is sufficiently small. Hence, the discrete-time dynamic model can be written as $\mathbf{w}_k = \mathbf{A}\mathbf{w}_{k-1} + \boldsymbol{\gamma}_k$, where $\boldsymbol{\gamma}$ is the dynamic modelling error described by zero mean Gaussian noise with covariance $\mathbf{Q}$ and $\mathbf{A}$ is the $(2m \times 2m)$ block matrix

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & T_s \mathbf{I}_m \\ \mathbf{O}_m & \mathbf{I}_m \end{bmatrix}$$

The output of the EKF, in the case that all the available data can be used, is the vector $\boldsymbol{\zeta}_k = [\boldsymbol{\zeta}_{f,k}^T \quad \boldsymbol{\zeta}_{h,k}^T \quad \boldsymbol{\zeta}_{hq,k}^T]^T$ of the measurements at time kT_s , where $\boldsymbol{\zeta}_{f,k} = \mathbf{f}_k + \boldsymbol{\mu}_{f,k}$, $\boldsymbol{\zeta}_{h,k} = \mathbf{h}_k + \boldsymbol{\mu}_{h,k}$, and $\boldsymbol{\zeta}_{hq,k} = \delta_k + \boldsymbol{\mu}_{hq,k}$, being $\boldsymbol{\mu}$ the measurement noise, which is assumed to be zero mean Gaussian noise with covariance $\boldsymbol{\Pi}$.

Taking into account Eqs. (5), (10), and (15), the output equation of the EKF can be written as $\boldsymbol{\zeta}_k = \mathbf{g}(\mathbf{w}_k) + \boldsymbol{\mu}_k$, with $[\boldsymbol{\mu}_{f,k}^T \quad \boldsymbol{\mu}_{h,k}^T \quad \boldsymbol{\mu}_{hq,k}^T]^T$ and $\mathbf{g}(\mathbf{w}_k) = [\mathbf{g}_f^T(\mathbf{w}_k) \quad \mathbf{g}_h^T(\mathbf{w}_k) \quad \mathbf{g}_{hq}^T(\mathbf{w}_k)]^T$, where only the explicit dependence on the state vector $\mathbf{w}_k$ has been evidenced. The EKF requires the computation of the Jacobian matrix of the output equation

$$\mathbf{C}_k = \left. \frac{\partial \mathbf{g}(\mathbf{w})}{\partial \mathbf{w}} \right|_{\mathbf{w}=\hat{\mathbf{w}}_{k,k-1}} = \left[\frac{\partial \mathbf{g}(\mathbf{w})}{\partial \mathbf{x}_o} \quad \mathbf{O} \right]_{\mathbf{w}=\hat{\mathbf{w}}_{k,k-1}}$$

where $\mathbf{O}$ is a null matrix of proper dimensions corresponding to the partial derivative of $\mathbf{g}$ with respect to the velocity variables. The Jacobian matrix $\partial \mathbf{g}(\mathbf{w}) / \partial \mathbf{x}_o$, in view of Eqs. (8), (14), and (18) has the expression $\partial \mathbf{g}(\mathbf{w}) / \partial \mathbf{x}_o = [\mathbf{J}_f^T \quad \mathbf{J}_h^T \quad \mathbf{J}_{hq}^T]^T$. The equations of the recursive form of the EKF are standard and are omitted here and can be found, e.g., in Lippiello and Villani (2003).

Notice that the adoption of the unit quaternion to represent the object orientation allows avoiding the problems of representation singularities typical of three-parameters representations. However, in the implementation of the Kalman filter, the unit norm constraint must be enforced to the estimated quaternion using, e.g., simple vector normalization (Choukroun et al. 2006).

4. INTERACTION CONTROL

The proposed algorithm can be used to estimate online the pose of an object in the workspace; this allows computing the surface equation with respect to the base frame in the form $\varphi(\mathbf{R}_o^T(\mathbf{p}_q - \mathbf{o}_o)) = \varphi(\mathbf{q}, t) = 0$, where the last equality holds in view of the direct kinematic equation of the robot. In the following, it is assumed that the object does not move; the general case of moving object is more complex but can be analyzed in a similar way. Hence, the constraint equation is $\varphi(\mathbf{q}) = 0$; moreover $\mathbf{J}_\varphi(\mathbf{q})\dot{\mathbf{q}} = 0$, where $\mathbf{J}_\varphi = \partial\varphi/\partial\mathbf{q}$ is a $(1 \times n)$ vector.

The dynamic model of the manipulator in contact with the environment is

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} - \mathbf{J}_\varphi^T(\mathbf{q})\lambda$$

where $\mathbf{B}$ is the $(n \times n)$ symmetric and positive definite inertia matrix, $\mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})$ is the $(n \times 1)$ vector taking into account Coriolis, centrifugal, friction and gravity torques, $\boldsymbol{\tau}$ is the $(n \times 1)$ vector of the joint torques, and λ is the lagrangian multiplier associated to the constraint.

The online computation of the constraint equations can be suitably exploited for interaction control. In the following, the case of the hybrid force/position control and of the impedance control are considered.

4.1. Hybrid Force/Position Control

Following the formulation presented in Villani and De Shutter (2008) the configuration of the robot subject to the constraint $\varphi(\mathbf{q}) = 0$ can be described in terms of a suitable vector $\mathbf{r}$ of $(n - 1)$ independent variables. From the implicit function theorem, this vector can be defined as $\mathbf{r} = \boldsymbol{\psi}(\mathbf{q})$, where $\boldsymbol{\psi}(\mathbf{q})$ is any $((n - 1) \times 1)$ twice differentiable vector function such that the $(n - 1)$ components of $\boldsymbol{\psi}(\mathbf{q})$ and function $\varphi(\mathbf{q})$ are linearly independent at least locally in a neighborhood of the operating point. This implies the existence of a $(n \times 1)$ vector function $\boldsymbol{\rho}(\mathbf{r})$ such that $\mathbf{q} = \boldsymbol{\rho}(\mathbf{r})$, at least locally. Moreover, the Jacobian $\mathbf{J}_\rho = \partial\boldsymbol{\rho}/\partial\mathbf{r}$ of the differential mapping $\dot{\mathbf{q}} = \mathbf{J}_\rho(\mathbf{r})\dot{\mathbf{r}}$ is a full-rank matrix. The matrices

$$\mathbf{S}_f = \mathbf{J}^{-T}(\mathbf{q})\mathbf{J}_\varphi^T(\mathbf{q}), \quad \mathbf{S}_v = \mathbf{J}(\mathbf{q})\mathbf{J}_\rho(\mathbf{r})$$

by virtue of the equalities $\mathbf{h} = \mathbf{S}_f\lambda$ and $\mathbf{v}_{p_q} = \mathbf{S}_v\dot{\mathbf{r}}$, play the role of selection matrices.

An hybrid force/position control task can be assigned by specifying the desired force $\lambda_d(t)$ and the $n - 1$ components of the vector $\mathbf{r}_d(t)$. An inverse dynamics control law can be adopted, by choosing the control torque $\boldsymbol{\tau}$ as

$$\boldsymbol{\tau} = \mathbf{B}(\mathbf{q})\boldsymbol{\alpha} + \mathbf{n}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{J}_\varphi^T(\mathbf{q})\mathbf{S}_f\mathbf{h}_\lambda$$

with $\boldsymbol{\alpha} = \mathbf{J}^{-1}(\mathbf{q})(\mathbf{S}_v\boldsymbol{\alpha}_v + \dot{\mathbf{S}}_v\dot{\mathbf{r}} - \ddot{\mathbf{J}}\dot{\mathbf{q}})$,

$$\boldsymbol{\alpha}_v = \ddot{\mathbf{r}}_d + \mathbf{K}_{Dr}(\dot{\mathbf{r}}_d - \dot{\mathbf{r}}) + \mathbf{K}_{Pr}(\mathbf{r}_d - \mathbf{r}) \quad (19)$$

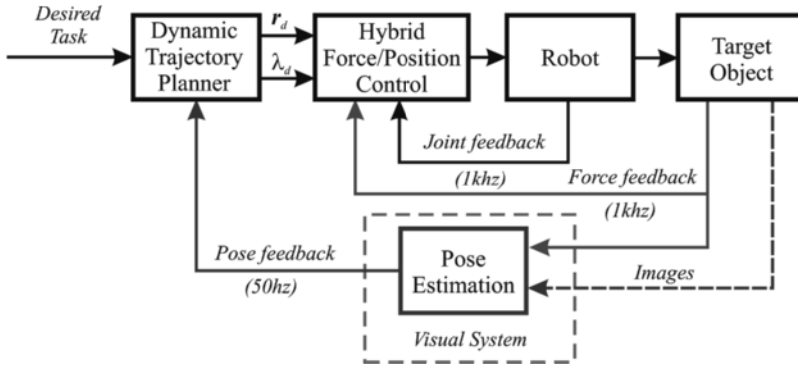


Figure 1. Hybrid force/position control.

and

$$h_\lambda = \lambda_d + k_{I\lambda} \int_0^t (\lambda_d(\tau) - \lambda(\tau)) d\tau \quad (20)$$

where K_{Dr} , K_{Pr} and $k_{I\lambda}$ are suitable feedback gains.

From the block scheme of Figure 1 it can be observed that the algorithm has a inner/outer structure, where the inner loop implements hybrid control whereas the outer loop computes the estimation of the object pose as well as the desired force and motion trajectories, on the basis of force, joint and visual measurements. Usually, the outer loop runs at a frequency lower than the inner loop, due to the limitation in the maximum camera frame rate (between 25 Hz and 60 Hz for low-cost cameras used in standard applications).

4.2. Impedance Control

The above scheme can be easily modified by replacing the inner hybrid force/position control with a different interaction control scheme. For instance, a position-based impedance control algorithm, based on the concept of compliant frame (Siciliano and Villani 1999), can be adopted (see Figure 2).

In detail, on the basis of the current estimate of the constraint equation, the dynamic trajectory planner generates a pose trajectory for a desired end-effector frame specified in terms of the position of the origin p_d and orientation matrix R_d . Moreover, a compliant frame r is introduced, specified in terms of p_r and R_r . Then, a mechanical impedance between the desired and the compliant frame is considered, so as to contain the values of the interaction force h and moment m . In other words, the desired position and orientation, together with the measured contact force and moment, are input to the impedance equation which, via a suitable integration, generates the position and orientation of the compliant frame to be used as a reference for the motion controller of the robot end effector.

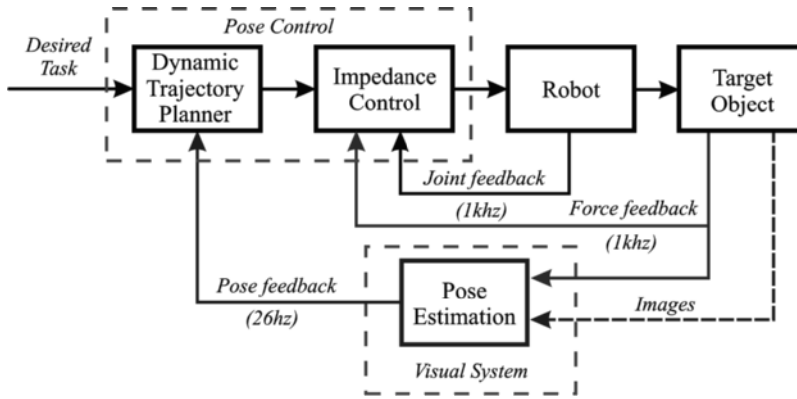


Figure 2. Position-based visual impedance control.

As far as the compliant frame is concerned, the position $\mathbf{p}_r$ can be computed via the translational impedance equation

$$\mathbf{M}_p \Delta \ddot{\mathbf{p}}_{dr} + \mathbf{D}_p \Delta \dot{\mathbf{p}}_{dr} + \mathbf{K}_p \Delta \mathbf{p}_{dr} = \mathbf{h} \quad (21)$$

where $\Delta \mathbf{p}_{dr} = \mathbf{p}_d - \mathbf{p}_r$, and $\mathbf{M}_p$, $\mathbf{D}_p$, and $\mathbf{K}_p$ are positive definite matrices representing the mass, damping, and stiffness characterizing the impedance.

The orientation of the reference frame $\mathbf{R}_r$ is computed via a geometrically consistent impedance equation similar to Eq. (21), in terms of an orientation error based on the (3×1) vector $\tau_{\epsilon_{dr}}$, defined as the vector part of the unit quaternion that can be extracted from ${}^{\tau}\mathbf{R}_d = \mathbf{R}_r^T \mathbf{R}_d$. The corresponding mass, damping and inertia matrices are $\mathbf{M}_o$, $\mathbf{D}_o$, and $\mathbf{K}_o$ respectively. More details about the geometrically consistent impedance based on the unit quaternion can be found in Siciliano and Villani (1999).

It is interesting noticing that the above scheme is able to manage both contact and free-space motion phases. In fact, during free space motion, the position-based impedance control is equivalent to a pose control and the whole scheme corresponds to a classical position-based visual servoing algorithm (Lippiello et al. 2007c). For this reason, this scheme can be classified as position-based visual impedance control.

5. CASE STUDIES

The proposed approach has been tested in a number of simulation and experimental case studies. The setup (Figure 3) is composed by a 6-DOF industrial robot Comau SMART-3 S with an open control architecture based on RTAI-Linux operating system. A six-axis force/torque sensor ATI FT30-100 with force range of ± 130 N and torque range of ± 10 N·m is mounted at the arm's wrist, providing readings of six components of generalized force at 1 ms. A visual system composed by a PC equipped with a Matrox GENESIS board and a Sony 8500CE B/W camera is available. The camera is fixed and calibrated with respect to the base frame of the robot.

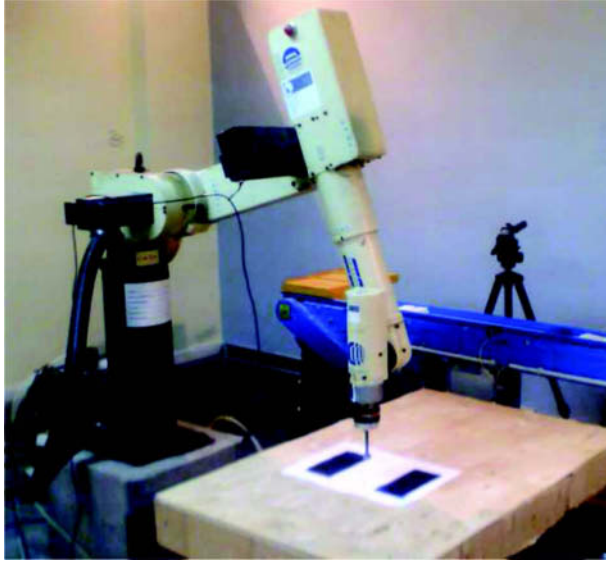


Figure 3. Experimental setup.

5.1. Hybrid Force/Position Control

A simulation study has been developed to test the hybrid force/position control scheme. The MATLAB/Simulink kinematic and dynamic models of the Comau SMART-3 S can be found in Siciliano and Villani (1999).

A planar object surface is considered, described by the equation ${}^o\mathbf{n}^{T_o}\mathbf{p} = 0$, assuming that the origin O_o of the object frame is a point of the plane and the axis z_o is aligned to the normal ${}^o\mathbf{n}$. During the interaction with the robot, the normal vector $\mathbf{n}$ remains constant in the base frame while the plane is assumed to be elastically compliant along $\mathbf{n}$ according to a simple elastic law. The contact force of the object on the robot's tip at $\mathbf{p}_q$ is given by

$$\mathbf{h} = \begin{cases} k\mathbf{n}\mathbf{n}^T(\mathbf{p}_o - \mathbf{p}_q) & \text{if } \mathbf{n}^T(\mathbf{p}_o - \mathbf{p}_q) \geq 0 \\ \mathbf{0} & \text{if } \mathbf{n}^T(\mathbf{p}_o - \mathbf{p}_q) < 0 \end{cases}$$

where $\mathbf{p}_q$ is on the plane when $\mathbf{h} \neq \mathbf{0}$ while $\mathbf{p}_o$ is a constant vector representing the position of a point of the plane when $\mathbf{h} = \mathbf{0}$. The scalar k , representing the stiffness of the surface, has been set to 10000 N/m.

The end-effector tool is a rigid stick of 25 cm length. The robot has a force/torque sensor mounted at the wrist. Neglecting the weight and inertia of the tool, the force at the robot's tip $\mathbf{p}_q$ and that measured at the robot wrist by the force sensor are the same, while a moment is measured by the sensor if the stick is not aligned to ${}^o\mathbf{n}$.

A camera is mounted on the robot end effector. It is assumed that the intrinsic parameters of the camera are affected by a 3% error, while the extrinsic parameters are supposed to be known with a tolerance of ± 1 cm for the position and ± 3 deg for the orientation. The object features are 4 points at the corners of a square of 10 cm side.

The control parameters in Eqs. (19) and (20) are chosen as

$$\mathbf{K}_{Pr} = \text{diag}\{250, 250, 250, 15, 15, 15\}, \mathbf{K}_{Dr} = \text{diag}\{220, 220, 220, 10, 10, 10\}$$

and $k_{I\lambda} = 0.05$; a 1 ms sampling time has been selected. These values have been selected by imposing a given closed-loop bandwidth and damping.

The desired task is planned in the object frame and consists in a straight-line motion of the end effector along the z_o axis keeping a fixed orientation with the stick normal to the $x_o y_o$ plane. The final position is the estimated rest position of the plane. During this phase, only the end-effector position and orientation are controlled, according to the law Eq. (19) applied to all the task directions. When force is detected, the motion controller is replaced by the hybrid force/position control law Eqs. (19) and (20), with a constant position and orientation and a desired force of -20 N along the z_o axis. After 2 s, a desired circular trajectory on the contact plane with 10 cm radius and 8 s duration is commanded for the end-effector position, while the desired orientation is unchanged and the desired force is kept constant to -20 N.

In the EKF, the non null elements of the matrix $\mathbf{H}$ has been set equal to $625 \cdot 10^{-11}$ for $\mathbf{f}$, 10^{-5} for $\mathbf{n}_h$ and 10^{-6} for δ_{hq} . These values have been set on the basis of the calibration errors of the camera. The state noise covariance matrix has been selected so as to give a rough measure of the errors due to the simplification introduced in the model (constant velocity), by considering only velocity disturbance, i.e.,

$$\mathbf{Q} = \text{diag}\{0, 0, 0, 0, 0, 0, 0, 0, 5, 0.5, 10^2, 10^3, 10^3, 10^3\} \cdot 10^{-11}$$

Notice that the unit quaternion has been used for the orientation in the EKF. Moreover a 20 ms sampling time has been set for the estimation algorithm, corresponding to the typical camera frame rate of 50 Hz.

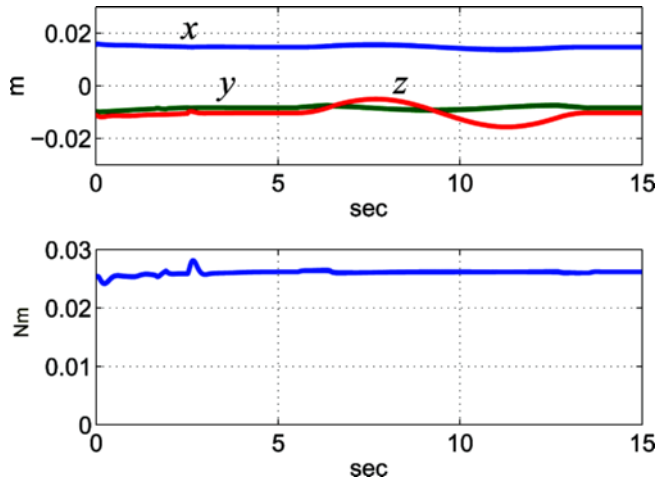


Figure 4. Pose estimation error in the first simulation. Top: position error; bottom: orientation error.

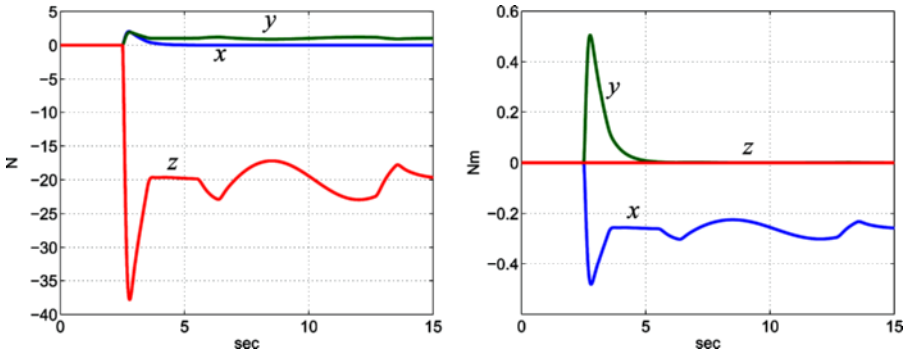


Figure 5. Measured force (top) and moment (bottom) in the first simulation.

Two different simulations are presented, to show the effectiveness of the use of force and joint position measurements, besides visual measurements.

In the first simulation, only the visual measurements are used. The object pose estimation errors are reported in Figure 4. The position error is computed as the difference between the real position of the origin of the object frame and the estimated position referred to the object frame; the orientation error is defined as the norm of the vector part of the quaternion extracted from the rotation matrix representing the mutual orientation of the real object frame with respect to the estimated frame. The task starts at time $t_o = 0$ s, when an estimate of the object pose is available from visual measurements; notice that the initial value of the pose estimation error is non null, due to camera calibration errors and remains different from zero during all the task.

The time history of the force and moments measured by the sensor are reported in Figure 5. Notice that the normal component of the measured force (the

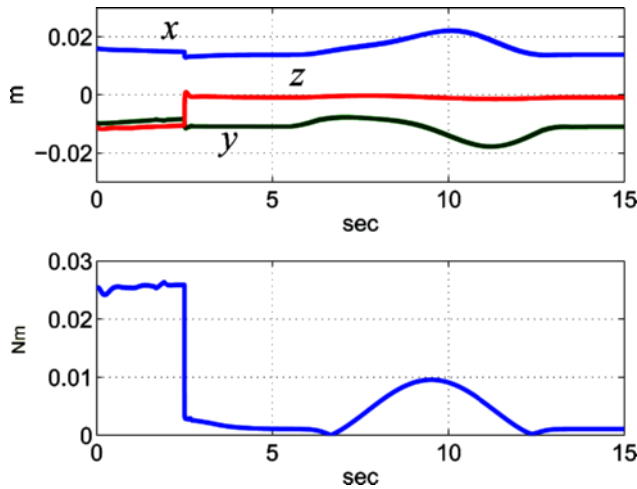


Figure 6. Pose estimation error in the second simulation. Top: position error; bottom: orientation error.

z component), after a transient, reaches a value close to the desired one. However, the y component of the force remains non null and a moment different from zero can be observed during the task execution, due to the error in the estimation of z_o .

The same task is repeated using also the contact force and the joint position measurements for object pose estimation; the results are reported in Figures 6 and 7. Before the contact (i.e., before time $t_c \simeq 2.2$ s), the estimation errors are the same as in the previous simulation. After the contact, the benefit of using additional measurements in the EKF results in a significant reduction of the pose estimation error, especially for the z component of the position (aligned to z_o) and for the orientation. The peak of the measured force during the transient is lower than before and the value of the normal component of the force, during the execution of the circle, remains very close to the desired value of -20 N. Finally, all the other components of the measured force, as well as the measured moment, become null after a short transient, meaning that the stick remains aligned to z_o .

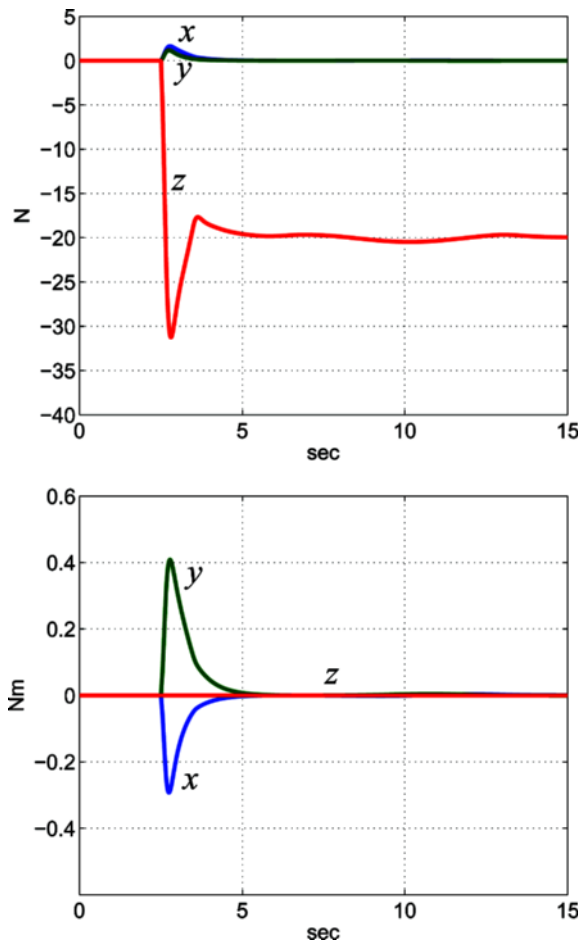


Figure 7. Measured force (top) and moment (bottom) in the second simulation.

5.2. Impedance Control

Experimental tests have been carried out to check the performance the impedance control scheme. The environment is a planar wooden horizontal surface, with an estimated stiffness (along $^o n$) of about 46000 N/m. The object features are 8 landmark points lying on the plane at the corners of two rectangles of 10×20 cm size (as in Figure 3).

The impedance parameters are chosen as: $M_p = 40I_3$, $D_p = 26.3I_3$ and $K_p = 1020I_3$, $M_o = 15I_3$, $D_o = 17.4I_3$ and $K_o = 3I_3$; a 1 ms sampling time has been selected for the impedance and pose control. Notice that the stiffness of the object is much higher than the positional stiffness of the impedance, so that the environment can be considered rigid.

The desired task is planned in the object frame and consists in a straight-line motion of the end effector along the z_o -axis keeping a fixed orientation with the stick

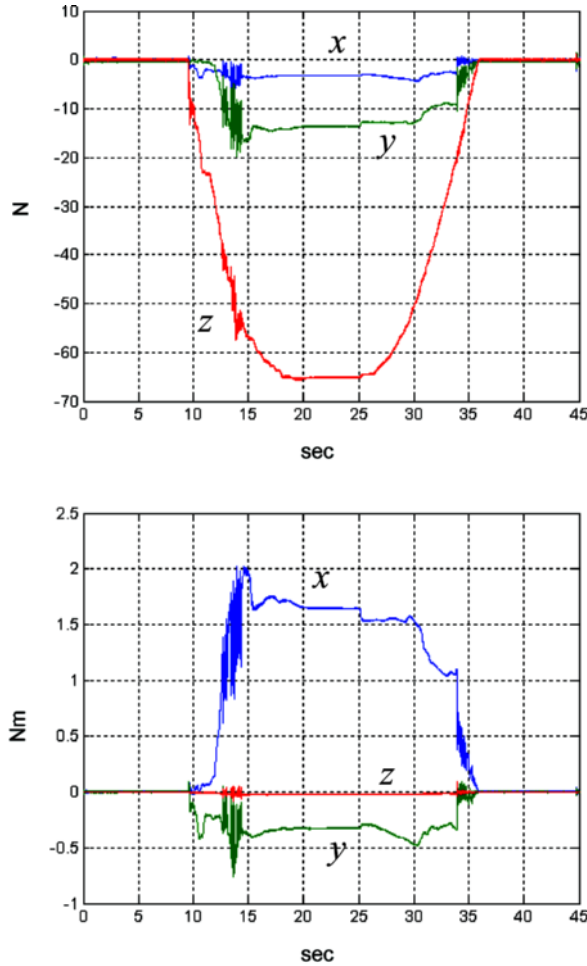


Figure 8. Measured force (top) and moment (bottom) in the first experiment.

normal to the $x_o y_o$ -plane. The final position is ${}^o p_f = [0 \ 0 \ 0.05]^T$ m, which is chosen to have a normal force of about 50 N at the equilibrium, with the selected value of the impedance positional stiffness. A trapezoidal velocity profile time-law is adopted, with a cruise velocity of 0.01 m/s. The absolute trajectory is computed from the desired relative trajectory using the current object pose estimation. The final position of the end effector is held for 2 s; after, a vertical motion in the opposite direction is commanded.

In the EKF, the non null elements of the matrix Π have been set equal to 2.5 pixel^2 for f , $5 \cdot 10^{-4}$ for n_h and 10^{-6} m^2 for δ_{hq} . These values have been set on the basis of the calibration errors of the camera. The state noise covariance matrix has been selected so as to give a rough measure of the errors due to the simplification introduced in the model (constant velocity), by considering only velocity disturbance, i.e.,

$$Q = \text{diag}\{0, 0, 0, 0, 0, 0, 0, 0, 10, 10, 10, 1, 1, 1\} \cdot 10^{-9}$$

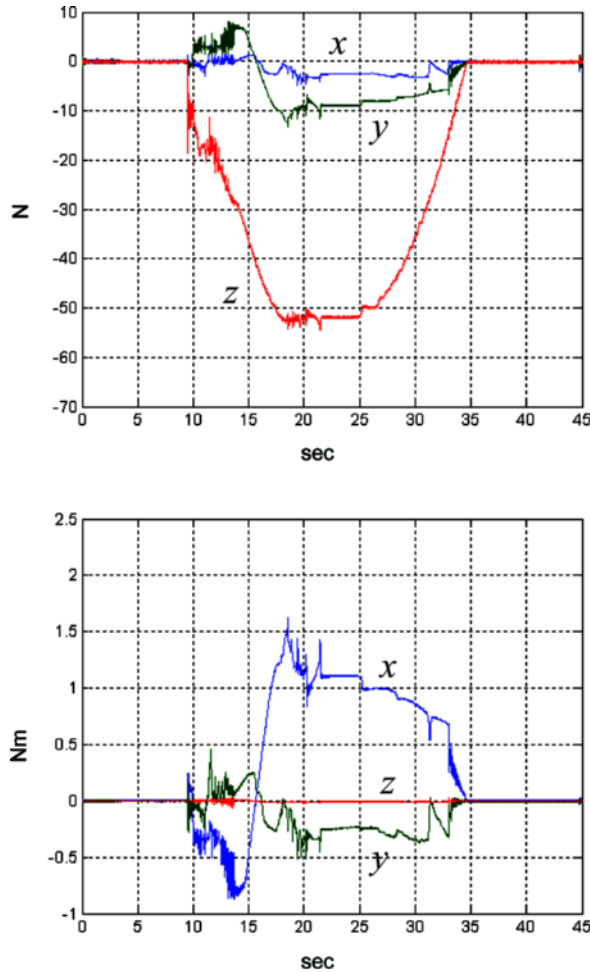


Figure 9. Measured force (top) and moment (bottom) in the second experiment.

Notice that the unit quaternion has been used for the orientation in the EKF. Moreover a 38 ms sampling time has been used for the estimation algorithm, corresponding to the typical camera frame rate of 26 Hz.

Two different experiments are presented, to show the effectiveness of the use of force and joint position measurements, besides visual measurements.

In the first experiment only the visual measurements are used. The time history of the measured force and moment in the sensor frame are reported in Figure 8. Notice that the force is null during the motion in free space and becomes different from zero after the contact. The impedance control keeps the force limited during the transient while, at steady state, the force component along the z axis reaches a value of about -65 N, which is different from the desired value of -50 N. This is caused by the presence of estimation errors on the position of the plane due to calibration errors of the camera. Moreover, the moment measured by the sensor is different from zero due to the misalignment between the estimated and the real normal direction of the plane.

The same task is repeated using also the contact force and the joint position measurements for object pose estimation; the results are reported in Figure 9. It can be observed that the benefit of using additional measurements in the EKF results in a force along the vertical direction which is very close to the desired value, due to the improved estimation of the position of the plane; moreover, the contact moment is also reduced because of the better estimation of the normal direction of the plane.

6. CONCLUSION

A framework for force and visual control of robot manipulators was proposed in this article. The environment was assumed as a rigid object of known geometry but unknown and possibly time-varying position and orientation. An estimation algorithm was adopted, based on visual, force and joint position data, which allows computing the actual position and orientation of the object so as to update in real time the constraint equations for the robot end effector. The resulting control has a inner/outer structure where the outer loop performs pose estimation and the inner loop is an interaction control scheme. Simulation and experimental results have demonstrated the superior performance of the proposed approach with respect to algorithms based on force measurements or visual measurements separately.

ACKNOWLEDGEMENTS

The research leading to these results has been supported in part by the PHRIENDS specific targeted research project, funded by the European Community's Sixth Framework Programme under contract IST-045359, and in part by the DEXMART large-scale integrating project, funded by the European Community's Seventh Framework Programme under grant agreement ICT-216239. The authors are solely responsible for its content. It does not represent the opinion of the European Community and the Community is not responsible for any use that might be made of the information contained therein.

REFERENCES

- Baeten, J. and J. De Schutter. 2004. *Integrated visual servoing and force control. The task frame approach*. Heidelberg, Germany: Springer.
- Chiaverini, S. and L. Sciavicco. 1993. The parallel approach to force/position control of robotic manipulators. *IEEE Trans. on Robotics and Automation* 9(4):361–373.
- Choukroun, D., Y. Itzhack, Y. Bar-Itzhack, and Y. Oshman. 2006. Novel quaternion Kalman filter. *IEEE Transactions on Aerospace and Electronic Systems* 42(1):174–190.
- Deng, L., F. Janabi-Sharifi, and I. Hassanzadeh. 2005. Comparison of combined vision/force control strategies for robot manipulators. *Proceedings of SPIE Int. Symp. Optomechatronic Technologies: Optomechatronic Systems Control Conference*, 605202-1-12.
- Espiau, B., F. Chaumette, and P. Rives. 1996. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation* 8(3):313–326.
- Hogan, N. 1985. Impedance control: An approach to manipulation, parts I-III. *ASME J. of Dynamic Systems, Measurement, and Control* 107(1):1–24.
- Hosoda, H., K. Igarashi, and M. Asada. 1998. Adaptive hybrid control for visual and force servoing in an unknown environment. *Robotics & Automation Magazine* 5(4):39–43.
- Lippiello, V. and L. Villani. 2003. Managing redundant visual measurements for accurate pose tracking. *Robotica* 21(5):511–519.
- Lippiello, V., B. Siciliano, and L. Villani. 2007a. A position-based visual impedance control for robot manipulators. *Proceedings of IEEE Int. Conf. on Robotics and Automation* 2068–2073.
- Lippiello, V., B. Siciliano, and L. Villani. 2007b. Robot force/position control with force and visual feedback. *Proceedings of European Control Conference* 3790–3795.
- Lippiello, V., B. Siciliano, and L. Villani. 2007c. Position-based visual servoing in industrial multi-robot cells using a hybrid camera configuration. *IEEE Trans. on Robotics* 23:73–86.
- Morel, G., E. Malis, and S. Boudet. 1998. Impedance based combination of visual and force control. *Proceedings of IEEE Int. Conf. on Robotics and Automation* 1743–1748.
- Nelson, B. J., J. D. Morrow, and P. K. Khosla. 1995. Improved force control through visual servoing. *Proceedings of American Control Conference* 380–386.
- Olsson, T., R. Johansson, and A. Robertsson. 2004. Flexible force-vision control for surface following using multiple cameras. *Proceedings of IEEE/RSJ Int. Conf. on Intelligent Robots and System* 798–803.
- Siciliano, B. and L. Villani. 1999. *Robot force control*. Boston: Kluwer Academic Publishers.
- Villani, L. and J. De Schutter. 2008. Force control. In *Springer Handbook of Robotics*, eds. B. Siciliano and O. Khatib. New York: Springer.
- Yoshikawa, T., T. Sugie, and M. Tanaka. 1998. Dynamic hybrid position/force control of robot manipulators—controller design and experiment. *IEEE J. of Robotics and Automation* 4(6):699–705.