KYBERNETIKA — VOLUME *48* (2012), NUMBER 6, PAGES 1136–1155

A LYAPUNOV-BASED DESIGN TOOL OF IMPEDANCE CONTROLLERS FOR ROBOT MANIPULATORS

Marco Mendoza, Isela Bonilla, Fernando Reyes
and Emilio González-Galván

This paper presents a design tool of impedance controllers for robot manipulators, based on the formulation of Lyapunov functions. The proposed control approach addresses two challenges: the regulation of the interaction forces, ensured by the impedance error converging to zero, while preserving a suitable path tracking despite constraints imposed by the environment. The asymptotic stability of an equilibrium point of the system, composed by full nonlinear robot dynamics and the impedance control, is demonstrated according to Lyapunov's direct method. The system's performance was tested through the real-time experimental implementation of an interaction task involving a two degree-of-freedom, direct-drive robot.

Keywords: impedance control, Lyapunov stability, robot manipulator

Classification: 68T40, 93C85, 93D05

1. INTRODUCTION

For decades, robotic systems have captured the attention of control researchers. The majority of these systems consider rigid robot manipulators, whose function is to perform specific industrial tasks using their end-effectors. A common feature in many robotic applications is the motion of the system in an unconstrained space i. e., the end-effector and the links of the manipulator are not in contact with the environment. However, several industrial and medical processes automated by robotic technology, require contact or interaction between the robot manipulator and its environment. The control of this interaction is crucial for the successful execution of many practical tasks where the robot's end-effector has to manipulate an object or perform some operation on a surface [25]. During interaction, manipulation is fundamental and requires the manipulator to be mechanically coupled to the object being manipulated, i. e., the manipulator may not be treated as an isolated system. In interaction tasks, the manipulator encounters environmental constraints and the interaction forces are not negligible [8].

Interaction control of a manipulator has been treated using, among others, two general strategies; the first, referred to as *hybrid position/force control* [22], divides the robot's workspace into orthogonal directions that are constrained either in force or position, and an appropriate force or position controller for each direction is designed [20]. The main problem with hybrid position/force control lies on its failure to recognize the

importance of the manipulator's impedance [1]. The second strategy, referred to as *impedance control* [8] is based on the control of the relationship between the interaction force and the position errors, resulting from this force. The dynamic interaction between the manipulator and its environment can be regulated and controlled by changing its mechanical impedance. Then, by means of an artificially-created end-effector compliance, the robot manipulator is able to maneuver in constrained environments maintaining adequate contact forces.

Impedance control represents a valuable tool in interaction tasks that do not require that the manipulator exerts a specific value of force. Instead, it is important that the robot adapts its dynamic behavior, during the contact, to achieve a desired compliant response. Systems whose motion is determined as a response to force inputs are described as admittances. When a robot manipulator is in contact with the environment, in order to ensure physical compatibility with the environmental admittance, it should assume the role of an impedance [8]. Due to the fact that the mechanical impedance is a relationship between force and motion, the impedance control schemes are widely used in industrial tasks, biological systems, and particularly in human-robot interaction such as: robotic surgery [6, 7], rehabilitation therapy [15, 16, 19], control of prosthetic devices [8], haptic devices [21], etc.

Impedance control has been implemented in many forms. In its simplest form it can be considered as a generalization of a damping and stiffness-control schemes [30]. In this form, it is essentially a PD position controller, with position and velocity feedback gains adjusted in order to obtain different apparent impedances. Extending the control of dynamic interaction, Hogan [8] presents a general and unified approach to implement, in a robot manipulator, an impedance controller and generate a linear mass-spring-damper, closed-loop system. Based on this approach, several control schemes have been developed for interaction control. Among them, a robust nonlinear impedance control was presented by Kazerooni [13]. The control approach introduced by Anderson and Spong [1], referred to as hybrid impedance control (HIC), combines the impedance control and the hybrid position/force control within a single strategy, ensuring a suitable interaction. In [5], a force-commanded impedance controller which results in an optimal combination of robustness and performance, is presented. In Carelli and Kelly [3], a solution to the adaptive impedance control of constrained robots with the presence of parametric uncertainties in the robot model, is presented. Recent contributions propose the use of visual information to determine the position of the object that interacts with the robot manipulator within an impedance and/or force control scheme [17, 18]. In reference [9], an accurate and robust impedance control technique, based on internal model control structure and time-delay estimation, is presented.

Inverse-dynamics control, presented in [24][26], is a known scheme to effectively decouple the manipulator's nonlinearly coupled degrees-of-freedom. Then, the control problem is reduced to design a suitable control acceleration according to the control issue to be solved. In this paper, a structure of such a control acceleration is proposed resulting in an impedance-control design tool based on the formulation of Lyapunov functions. The proposed approach represents a generalization of motion control to solve the constrained motion problem of robot manipulators. Typical impedance controllers are used to regulate the interaction between the manipulator and its environment, regardless of the

tracking errors caused by this interaction [8]. In order to reduce the errors, the value of the impedance parameters has to be increased and the compliant behavior of the robot end-effector is jeopardized. The present impedance control approach was designed in such a way that it solves both challenges: the regulation of the interaction forces and a suitable path tracking. The interaction forces are indirectly regulated by ensuring convergence to zero of the impedance error. The controller also allows for a suitable path tracking, despite the constraints imposed by the environment. The control scheme has two control loops. The first loop is an inner kinematic control loop where the reference position for the controller is modified, by generating an adjustment vector for the robot motion according to the contact forces measured via a wrist force/torque sensor. The second loop is a motion controller that allows the solution of the path tracking problem, even when the system is acted upon by environmental forces, enabling the system to maintain a compliant behavior. Further, this approach allows to lead the closed-loop system to an asymptotically-stable equilibrium point.

In addition to the theoretical issues of the proposed control approach, this paper also presents a real-time experimental implementation of an interaction task that makes use of a direct-drive robot manipulator of two degrees of freedom. Based on the experimental results, a comparison between two impedance controllers, derived from the proposed impedance-control approach, and the control proposed by Hogan in [8], is presented. The $\mathcal{L}_2$ norm of impedance errors, as suggested in [11, 12, 14, 29], and the index presented in [28], were used to evaluate the performance of each controller.

The layout of the paper is as follows. Section 2 summarizes the constrained, robot-dynamic model. The design tool of impedance controllers and its Lyapunov-stability analysis are presented in Section 3. Section 4 presents the most important aspects of the experimental implementation of an interaction task used to test the performance of the proposed impedance controllers. Finally, some concluding remarks are given in Section 5.

2. MODEL OF THE SYSTEM

2.1. Robot manipulator

The kinematic description of the position of robot's end-effector can be performed in joint-space or task-space. In the first case, $\boldsymbol{q}$ represents the joint coordinates while in task-space, $\boldsymbol{x}$ represents the posture (position and orientation) between the end-effector and a fixed cartesian coordinate frame [26].

Direct kinematics mapping $\mathcal{K} : \mathbb{R}^n \mapsto \mathbb{R}^m$, relates the joint position vector $\boldsymbol{q} \in \mathbb{R}^n$ with the task posture vector $\boldsymbol{x} \in \mathbb{R}^m$, and is given by

$$\boldsymbol{x} = \mathcal{K}(\boldsymbol{q}). \tag{1}$$

The number of degrees of freedom of the robot manipulator is represented by n while m is the dimension of its work space. First and second time derivatives of (1) correspond to the differential relationships between velocities and accelerations, respectively, of the joint and task spaces,

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{2}$$

$$\ddot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{3}$$

where $\boldsymbol{J}(\boldsymbol{q}) = \partial\mathcal{K}/\partial\boldsymbol{q} \in \mathbb{R}^{m\times n}$ is the analytical Jacobian matrix [24] and $\dot{\boldsymbol{J}}(\boldsymbol{q}) = \mathrm{d}\boldsymbol{J}/\mathrm{d}t \in \mathbb{R}^{m\times n}$ is its time derivative.

Interaction tasks require the relationship between joint torques and corresponding externally-applied forces and torques. In the joint-space representation, the generalized joint torques are given by $\boldsymbol{\tau}$. On the other hand, in task-space, the vector $\boldsymbol{f}$ consists of the force and torque components operating at the end-effector with respect to the fixed Cartesian coordinate system. Based on the principle of virtual work, the joint applied torques $\boldsymbol{\tau} \in \mathbb{R}^n$, needed to generate the forces and torques $\boldsymbol{f} \in \mathbb{R}^m$ on the end-effector, are given by the following relationship [27]

$$\boldsymbol{\tau} = \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{f}. \tag{4}$$

The joint-space dynamics of a n-link constrained rigid robot manipulator, interacting with the environment, can be written as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{f}_r(\dot{\boldsymbol{q}}) = \boldsymbol{\tau} - \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{f}_e \tag{5}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the symmetric positive definite inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ is the Coriolis and centripetal acceleration matrix, $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ is the vector of gravitational torques, $\boldsymbol{f}_r(\dot{\boldsymbol{q}}) \in \mathbb{R}^n$ is the vector of viscous-friction torques, and $\boldsymbol{f}_e \in \mathbb{R}^m$ denotes the vector of contact forces and torques [2].

2.2. Environment

Two main groups encompassing several environment models, have been considered:

- **Constrained.** Interaction between the robot and a rigid environment includes the models of impact, contact, Coulomb friction, and associated constraints.

- **Compliant.** The environment also can be modeled as a high-stiffness spring:

$$\boldsymbol{f}_e = \boldsymbol{K}_e(\boldsymbol{x} - \boldsymbol{x}_e) \tag{6}$$

  where $\boldsymbol{K}_e = \mathrm{diag}\{[k_e \cdots k_e]\} \in \mathbb{R}^{m\times m}$ represents the stiffness of an object located at $\boldsymbol{x}_e \in \mathbb{R}^m$. This linear relationship between force and deformation represents a first approach to robot-environment modeling. Hogan [8] proposed a dynamic-interaction model consisting of a generalized mass-spring-damper system.

In many applications, the interaction forces and torques between the end-effector and a compliant environment can be approximated by the ideal elastic model of the form (6). If the stiffness matrix $\boldsymbol{K}_e$ is positive definite, this model corresponds to a fully constrained case and the environment deformation coincides with the elementary end-effector displacement [25]. With high spring stiffness, the effective range of displacement decreases and force is a more appropriate measurement. On the other hand, with lower spring stiffness, the effective range of applied force decreases and the effector distance is the more appropriate measurement.

3. DESIGN TOOL FOR IMPEDANCE CONTROL

The impedance-control approach presented in this paper corresponds to a generalization of motion control in task-space, by choosing a desired trajectory $\boldsymbol{x}_d$. It attempts to maintain the following dynamic relationship [26],

$$\boldsymbol{x}_d - \boldsymbol{x} = \mathcal{F}(s)\boldsymbol{f}_e \tag{7}$$

where

$$\mathcal{F}(s) = [s^2\boldsymbol{M}_d + s\boldsymbol{B}_d + \boldsymbol{K}_d]^{-1} \tag{8}$$

while the diagonal, positive definite, stiffness, damping and inertia matrices are respectively $\boldsymbol{K}_d \in \mathbb{R}^{m\times m}$, $\boldsymbol{B}_d \in \mathbb{R}^{m\times m}$ and $\boldsymbol{M}_d \in \mathbb{R}^{m\times m}$. Then $\mathcal{F}(s)$ is a stable, second-order linear filter.

The impedance error $\tilde{\boldsymbol{\xi}} \in \mathbb{R}^m$ can be defined as [3]:

$$\tilde{\boldsymbol{\xi}} = \tilde{\boldsymbol{x}} - \boldsymbol{x}_{f_e} \tag{9}$$

where $\tilde{\boldsymbol{x}} = \boldsymbol{x}_d - \boldsymbol{x}$ represents the unconstrained tracking error and $\boldsymbol{x}_{f_e} = \mathcal{F}(s)\boldsymbol{f}_e$ is the adjustment vector obtained from the force filtering.

Formally, the aim of our impedance-control approach consists of selecting $\boldsymbol{\tau}$ in such a way that:

$$\lim_{t\to\infty} \tilde{\boldsymbol{\xi}}(t) = \boldsymbol{0} \tag{10}$$

$$\lim_{t\to\infty} \dot{\tilde{\boldsymbol{\xi}}}(t) = \boldsymbol{0}. \tag{11}$$

Note that, in absence of contact, i. e., if $\boldsymbol{f}_e \equiv \boldsymbol{0}$, the objective of impedance control is equivalent to the objective of motion control in task-space. Therefore, this approach to impedance control can be seen as a motion-control approach allowing a tolerance to error in the tracking of the desired trajectory, in the presence of interaction forces.

Considering the definition of impedance error (9), the vectors

$$\dot{\tilde{\boldsymbol{\xi}}} = \dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}} - \dot{\boldsymbol{x}}_{f_e} \tag{12}$$

$$\ddot{\tilde{\boldsymbol{\xi}}} = \ddot{\boldsymbol{x}}_d - \ddot{\boldsymbol{x}} - \ddot{\boldsymbol{x}}_{f_e} \tag{13}$$

represent the first and second time derivatives of the impedance error, respectively.

The impedance-control structure proposed in this paper is based on the concept of inverse dynamics [4, 24, 26], where the robot torques are selected as

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{J}^{-1}(\boldsymbol{q})[\boldsymbol{a} - \dot{\boldsymbol{J}}(\boldsymbol{q})\dot{\boldsymbol{q}}] + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{f}_r(\dot{\boldsymbol{q}}) + \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{f}_e. \tag{14}$$

In the case of a kinematically redundant manipulator, i. e., with nonsquare Jacobian matrix, a dynamically consistent generalized inverse of the Jacobian [2] can be adopted as,

$$\boldsymbol{J}^{\dagger}(\boldsymbol{q}) = \boldsymbol{J}^T(\boldsymbol{q})[\boldsymbol{J}(\boldsymbol{q})\boldsymbol{J}^T(\boldsymbol{q})]^{-1} \tag{15}$$

which is referred to as right pseudoinverse. If $\boldsymbol{J}(\boldsymbol{q})$ is square, the pseudoinverse (15) is reduced to the standard inverse Jacobian matrix $\boldsymbol{J}^{-1}(\boldsymbol{q})$.

By substituting the impedance-control law (14) in the dynamic model (5) and considering the time derivatives (2) – (3), the following is obtained,

$$\ddot{\boldsymbol{x}} = \boldsymbol{a}. \tag{16}$$

In order to solve the impedance-control problem, several controllers can be derived from a design tool given by the following theorem.

Theorem 3.1. Let $V(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}})$ be a positive-definite Lyapunov function (Appendix A) given by

$$V(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) = \frac{1}{2} \begin{bmatrix} \tilde{\boldsymbol{\xi}} \\ \dot{\tilde{\boldsymbol{\xi}}} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{K}_v & \boldsymbol{M}_d \\ \boldsymbol{M}_d & \boldsymbol{M}_d \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{\xi}} \\ \dot{\tilde{\boldsymbol{\xi}}} \end{bmatrix} + \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \tag{17}$$

where $\tilde{\boldsymbol{\xi}} \in \mathbb{R}^m$ represents the impedance-error vector, $\boldsymbol{K}_v \in \mathbb{R}^{m \times m}$ and $\boldsymbol{M}_d \in \mathbb{R}^{m \times m}$ are diagonal positive-definite matrices, and $\mathcal{U}_p : \mathbb{R}^m \mapsto \mathbb{R}$ is an artificial potential energy function such that

- $\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) > 0$ and $\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \equiv 0 \Longleftrightarrow \tilde{\boldsymbol{\xi}} = \boldsymbol{0}$, $\forall \tilde{\boldsymbol{\xi}} \in \mathbb{R}^m$.
- $\tilde{\boldsymbol{\xi}}^T \nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) > 0$ and $\tilde{\boldsymbol{\xi}}^T \nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \equiv 0 \Longleftrightarrow \tilde{\boldsymbol{\xi}} = \boldsymbol{0}$, $\forall \tilde{\boldsymbol{\xi}} \in \mathbb{R}^m$.
- $\nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \equiv \boldsymbol{0} \Longleftrightarrow \tilde{\boldsymbol{\xi}} = \boldsymbol{0}$.

where $\nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}})$ is the gradient of $\mathcal{U}_p$. Then, there exists some $\boldsymbol{a} \in \mathbb{R}^m$ given by

$$\boldsymbol{a} = \ddot{\boldsymbol{x}}_d - \ddot{\boldsymbol{x}}_{f_e} + \boldsymbol{M}_d^{-1}[\nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) + \boldsymbol{K}_v \dot{\tilde{\boldsymbol{\xi}}}] \tag{18}$$

such that the origin of the closed-loop equation (16) is asymptotically stable in Lyapunov sense.

P r o o f. First, by introducing (18) in (16) and considering the time derivative (13), the following closed-loop system equation, by combining the robot model (5) and the control scheme (14), can be obtained

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} \tilde{\boldsymbol{\xi}} \\ \dot{\tilde{\boldsymbol{\xi}}} \end{bmatrix} = \begin{bmatrix} \dot{\tilde{\boldsymbol{\xi}}} \\ -\boldsymbol{M}_d^{-1}[\nabla \mathcal{U}_p(\tilde{\boldsymbol{\xi}}) + \boldsymbol{K}_v \dot{\tilde{\boldsymbol{\xi}}}] \end{bmatrix}. \tag{19}$$

The equilibrium point of the system (19) exists under the following conditions:

a) A singularity-free work space, i. e. $\mathrm{rank}[\boldsymbol{J}(\boldsymbol{q})] = n \ \forall \boldsymbol{q} \in \mathbb{R}^n$, is considered.

b) From the upper term of (19) follows directly that $\dot{\tilde{\boldsymbol{\xi}}} = \boldsymbol{0}$.

c) Since $\boldsymbol{M}_d > 0 \Rightarrow \exists\ \boldsymbol{M}_d^{-1} > 0$, then

$$\begin{aligned} \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} \equiv \boldsymbol{0} &\iff \dot{\tilde{\boldsymbol{\xi}}} = \boldsymbol{0} \\ \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \equiv \boldsymbol{0} &\iff \tilde{\boldsymbol{\xi}} = \boldsymbol{0}. \end{aligned}$$

Therefore, the origin of the state space is an equilibrium point of the closed-loop system.

In order to carry out the Lyapunov-stability analysis of the equilibrium point, the function (17) is considered. Then, the time derivative of (17) along the trajectories of the closed-loop equation (19), after some algebra (Appendix B), can be written as:

$$\dot{V}(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) = -\tilde{\boldsymbol{\xi}}^T \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) - \dot{\tilde{\boldsymbol{\xi}}}^T (\boldsymbol{K}_v - \boldsymbol{M}_d)\dot{\tilde{\boldsymbol{\xi}}} \tag{20}$$

where $\tilde{\boldsymbol{\xi}}^T \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}})$ is positive definite and $\boldsymbol{K}_v$ and $\boldsymbol{M}_d$ are selected in such a way that $\lambda_{\min}\{\boldsymbol{K}_v\} > \lambda_{\max}\{\boldsymbol{M}_d\}$ (Appendix A). Then, $\dot{V}(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) < 0$ and therefore it is possible to conclude that the origin of the state space is asymptotically stable in Lyapunov sense. □

In order to minimize the effect of jump condition, i. e. when the robot losses and reestablishes the contact, the environment is treated as a mechanical impedance and the impedance-controller parameters are suitably tuned to obtain a smooth transition. The force/torque filtering $\boldsymbol{x}_{f_e}$ is stable and the impedance parameters $\boldsymbol{K}_d = \text{diag}\{[k_d \cdots k_d]\}$, $\boldsymbol{B}_d = \text{diag}\{[b_d \cdots b_d]\}$ and $\boldsymbol{M}_d = \text{diag}\{[m_d \cdots m_d]\}$ can be selected according to

$$\omega_n = \sqrt{\frac{k_d + k_e}{m_d}} \tag{21}$$

$$\zeta = \frac{b_d}{2\sqrt{m_d(k_d + k_e)}} \tag{22}$$

where the natural frequency ω_n and the damping factor ζ determine the dynamic behavior during the phase transition and the constrained motion (Appendix C).

3.1. Impedance controllers

An extended family of control algorithms can be designed from the proposed impedance-control structure (18) presented in Theorem 3.1. In this paper, two examples of control algorithms were proposed and tested. These algorithms were selected based on its simplicity and performance, without restricting the possibility of deriving other control schemes based on the proposed design tool. The two examples are:

Example 3.1.1. By selecting,

$$\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) = \frac{1}{2}\tilde{\boldsymbol{\xi}}^T \boldsymbol{K}_p \tilde{\boldsymbol{\xi}}$$

where $\boldsymbol{K}_p \in \mathbb{R}^{m\times m}$ is a diagonal positive-definite matrix of proportional gains, it can be obtained that

$$\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) = \boldsymbol{K}_p \tilde{\boldsymbol{\xi}}$$

then

$$\boldsymbol{a} = \ddot{\boldsymbol{x}}_d - \ddot{\boldsymbol{x}}_{f_e} + \boldsymbol{M}_d^{-1}[\boldsymbol{K}_p\tilde{\boldsymbol{\xi}} + \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}}], \tag{23}$$

and thus it can be observed that $\boldsymbol{K}_p\tilde{\boldsymbol{\xi}} + \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}}$ represents a proportional-derivative (PD) control action.

Example 3.1.2. Saturated regulators have shown very good performance in robot-control tasks. The $\tanh(\cdot)$ is a saturation function with smooth transition and also attenuates the error, therefore, allows more versatility in the tuning of controller gains. In order to design an impedance controller with a tanh-structure, $\boldsymbol{f}_p$ is developed via the integral of $\tanh(\cdot)$ function:

$$\ln\{\cosh(\tilde{\xi}_i)\} = \int \tanh(\tilde{\xi}_i)\,\mathrm{d}\tilde{\xi}_i \qquad i = 1, \ldots, m.$$

If $\boldsymbol{K}_p$=diag$\{[k_{p_1} \cdots k_{p_m}]\}$ then

$$\frac{\partial[k_{p_i}\ln\{\cosh(\tilde{\xi}_i)\}]}{\partial\tilde{\xi}_i} = k_{p_i}\tanh(\tilde{\xi}_i).$$

Then, a saturated-proportional derivative controller can be designed by selecting,

$$\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) = \boldsymbol{f}_p(\tilde{\boldsymbol{\xi}})^T\boldsymbol{K}_p\boldsymbol{f}_p(\tilde{\boldsymbol{\xi}})$$

where

$$\boldsymbol{f}_p(\tilde{\boldsymbol{\xi}}) \quad = \quad \begin{bmatrix} \sqrt{\ln\{\cosh(\tilde{\xi}_1)\}} \\ \vdots \\ \sqrt{\ln\{\cosh(\tilde{\xi}_m)\}} \end{bmatrix}.$$

$\mathcal{U}_p$ has a quadratic form to ensure positivity of the function, therefore, the square root in $\boldsymbol{f}_p$ is only for compatibility with such a quadratic form. The gradient of $\mathcal{U}_p$ is given by

$$\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) = 2\boldsymbol{K}_p\frac{\partial\boldsymbol{f}_p(\tilde{\boldsymbol{\xi}})}{\partial\tilde{\boldsymbol{\xi}}}\boldsymbol{f}_p(\tilde{\boldsymbol{\xi}})$$

where

$$\frac{\partial\boldsymbol{f}_p(\tilde{\boldsymbol{\xi}})}{\partial\tilde{\boldsymbol{\xi}}} \quad = \quad \frac{1}{2}\begin{bmatrix} \frac{\tanh(\tilde{\xi}_1)}{\sqrt{\ln\{\cosh(\tilde{\xi}_1)\}}} & & 0 \\ & \ddots & \\ 0 & & \frac{\tanh(\tilde{\xi}_m)}{\sqrt{\ln\{\cosh(\tilde{\xi}_m)\}}} \end{bmatrix}.$$

Then

$$\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) = \boldsymbol{K}_p\tanh(\tilde{\boldsymbol{\xi}})$$

and therefore

$$\boldsymbol{a} = \ddot{\boldsymbol{x}}_d - \ddot{\boldsymbol{x}}_{f_e} + \boldsymbol{M}_d^{-1}[\boldsymbol{K}_p\tanh(\tilde{\boldsymbol{\xi}}) + \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}}]. \tag{24}$$

This controller will be referred to as Tanh-D impedance control.

3.2. Hogan's impedance-controller

The fundamental idea underlying impedance control [8] is that environmental constraints impose a relationship between the posture of a manipulator and the contact forces and torques exerted by the environment. The equation for the desired behavior may be regarded as a specification of the desired end-point acceleration, as a result of an external force and torque imposed on the manipulator. Then, the control input in (14) is chosen as

$$\boldsymbol{a} \;=\; \ddot{\boldsymbol{x}}_d + \boldsymbol{M}_d^{-1}\left[\boldsymbol{K}_d\tilde{\boldsymbol{x}} + \boldsymbol{B}_d\dot{\tilde{\boldsymbol{x}}} - \boldsymbol{f}_e\right] \tag{25}$$

where the parameters $\boldsymbol{M}_d$, $\boldsymbol{B}_d$ y $\boldsymbol{K}_d$ are, respectively, the mass, damping and stiffness of the desired mechanical impedance between the end-effector posture error $\tilde{\boldsymbol{x}}$ and the contact force and torque $\boldsymbol{f}_e$.

4. EXPERIMENTAL RESULTS

In order to provide experimental verification of the theoretical development presented herein, this section presents the real-time experimental implementation of a robotic interaction task, controlled by the proposed impedance-control structure. The experimental results presented in this section only consider the position and force components of $\boldsymbol{x}$ and $\boldsymbol{f}_e$, respectively.

Fig. 1. Experimental setup.

4.1. Experimental setup

The experimental setup depicted in Figure 1 consists of a two degree-of-freedom, direct-drive robot manipulator [23] with a force and torque sensor mounted at the end-effector. The robot is composed by two aluminum links and the joints are actuated by two Parker Compumotor direct-drive servomotors, whose torque limits are listed in Table 1. Servomotors are operated in torque mode, i. e. they act as torque sources and receive an analog voltage as a torque reference signal. Joint position data is obtained by using

incremental encoders installed in the motors while the velocities are obtained through numerical differentiation of the position measurements. In order to read the encoder's data and generate reference voltages, the robot includes a Precision MicroDynamics Inc. motion control card. The control algorithms were written in C language and run in real time, with a sample rate of 2.5 milliseconds, on a 166 MHz Pentium-I computer.

Joint	**Model**	**Max. Torque(Nm)**	**Resolution(ppr)**
1. Shoulder	DM-1150A	150	1,024,000
2. Elbow	DM-1015B	15	1,024,000

Tab. 1. Torque limits of the robot servo actuators.

The interaction forces $\boldsymbol{f}_e$ are measured explicitly by means of an appropriate force sensor which is mounted at the end of manipulator, therefore, the system can be decoupled during interaction with the environment. In the control approach presented herein, the impedance parameters are tuned according to a desired behavior. The tuned parameters are used in a discrete model that represents the force filtering defined in equations (7) – (8) and it was obtained by using a zero-order hold (Appendix C), in order to obtain a motion adjustment, $\boldsymbol{x}_{f_e}$ in the impedance error (9), that is processed by the controller. Force sensing was performed by using an ATI Industrial FT Gamma sensor, whose measurements limits are listed in Table 2. The acquisition card for the sensor was installed on a 3.6 GHz Pentium-IV computer and the signals were processed using a Visual C++ application. In order to communicate the robot with the force sensor, a communication protocol via parallel port was developed. With the purpose of guaranteeing the stability of the robot system and in order to synchronize both computers, sensor readings are sent by generating an interruption signal through the parallel port. Force data are floating-point, sharing memory location within an 8-nibble structure, enabling the sending of interaction-force information through the parallel port. Finally, the robot is to interact with a planar, wooden wall with an estimated stiffness of about $\boldsymbol{K}_e = \text{diag}\{[10^4\ 10^4]\}$ N/m.

Signal	**Operation range**	**Units**
f_x, f_y	$\pm$ 130	N
f_z	$\pm$ 400	N
τ_x, τ_y, τ_z	$\pm$ 10	Nm

Tab. 2. Force/torque sensor features.

4.2. Interaction task

The robot task consisted of tracking a pre-established trajectory without contact with the environment. In the presence of external forces, the manipulator must follow the trajectory imposed by the environment, regulating the contact forces in order to obtain a suitable tracking.

The desired trajectory was generated according to a cubic interpolation, with null initial and final velocities. The initial joint configuration of the robot manipulator was ($q_{d_1}(0) = 5^{\circ}$, $q_{d_2}(0) = -5^{\circ}$), corresponding to the Cartesian end-effector position (0.0392 m, –1.1283 m). The final joint configuration was defined as ($q_{d_1}(t_f) = 115^{\circ}$, $q_{d_2}(t_f) = -28^{\circ}$) which corresponds to the end-effector position (1.0869 m, 0.1545 m) on the x-y plane. The trajectory was planned in such a way that the robot must arrive to the final position in t_f=10 seconds. In joint-space, with the purpose of avoiding singularities of $J(\boldsymbol{q}_d)$, the planned trajectory is given by

$$\boldsymbol{q}_d(t) = \boldsymbol{q}_d(0) + 3\left(\frac{\Delta_q}{t_f^2}\right)t^2 - 2\left(\frac{\Delta_q}{t_f^3}\right)t^3 \tag{26}$$

where $\Delta_q = \boldsymbol{q}_d(t_f) - \boldsymbol{q}_d(0)$. Such a trajectory was transformed to Cartesian coordinates using direct kinematics, in order to obtain the desired trajectory $\boldsymbol{x}_d = [x_d\ y_d]^T$ as,

$$x_d(t) = l_1 \sin(q_{d_1}) + l_2 \sin(q_{d_1} + q_{d_2}) \tag{27}$$

$$y_d(t) = -l_1 \cos(q_{d_1}) - l_2 \cos(q_{d_1} + q_{d_2}). \tag{28}$$

The desired velocity and acceleration can be obtained by differentiating (27) and (28).

4.3. Performance Results

The experimental results of PD-type (23), Tanh-D (24) and Hogan's (25) impedance controllers are depicted in Figure 2–5. In order to compare the performance of the controllers, the interaction task described above was performed with the direct-drive manipulator. In all cases, the wall was placed approximately at $x_e = 0.98$ m.

Parameter	PD	Tanh-D	Hogan	Units
$\boldsymbol{K}_p$	diag{[600 600]}	diag{[600 600]}	-	N/m
$\boldsymbol{K}_v$	diag{[60 60]}	diag{[60 60]}	-	N·s/m
$\boldsymbol{K}_d$	diag{[10 10]}	diag{[10 10]}	diag{[10 10]}	N/m
$\boldsymbol{B}_d$	diag{[25 25]}	diag{[25 25]}	diag{[25 25]}	N·s/m
$\boldsymbol{M}_d$	diag{[2 2]}	diag{[2 2]}	diag{[2 2]}	kg

Tab. 3. Parameters of the impedance controllers.

Figure 2 shows the desired and actual end-effector paths. It is possible to observe that the desired trajectory is accurately followed by the end-effector until it encounters the wall, which imposes a vertical trajectory in $x = 0.98$ m. During the interaction with the wall, the contact forces are regulated by the impedance parameters $\boldsymbol{K}_d$, $\boldsymbol{B}_d$ and $\boldsymbol{M}_d$.

Tuned parameters for the impedance controllers are listed in Table 3. In order to compare the proposed controllers, similar tuned parameters are used in every case; $\boldsymbol{K}_p$ was chosen on the basis of a trade-off between tracking accuracy during the free motion and compliant behavior of the end-effector during the constrained motion, while $\boldsymbol{K}_v$ was chosen in order to guarantee a well-damped behavior.

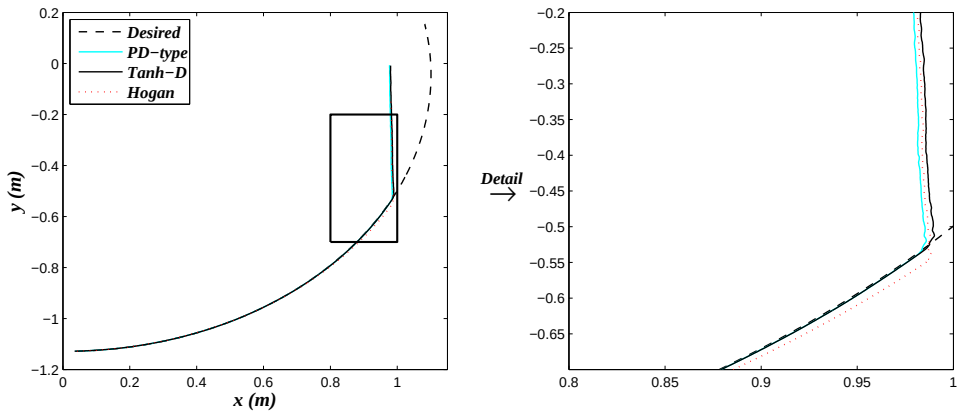

Fig. 2. Trajectory on the x-y plane.

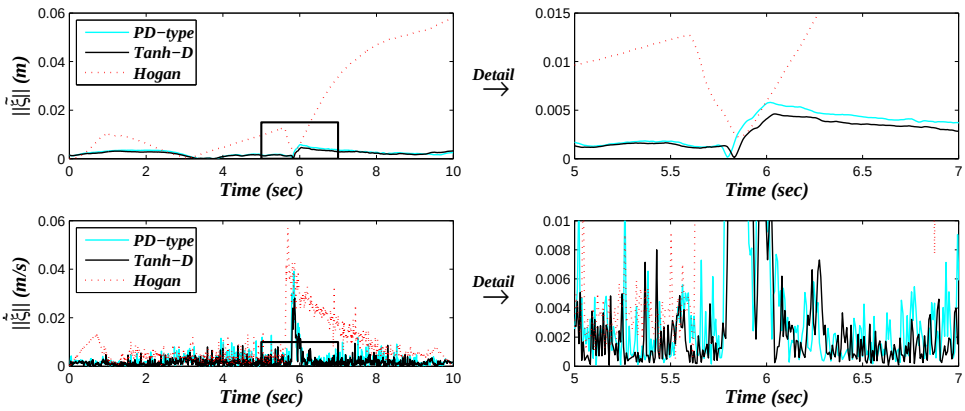

Fig. 3. Impedance error and its time-derivative, respectively.

Figure 3 shows that, as time evolves, the impedance error $\tilde{\boldsymbol{\xi}}$ tends to zero, in agreement with the impedance control objective defined in eq. (10). Similarly, as shown at the bottom of Figure 3, $\dot{\tilde{\boldsymbol{\xi}}}$ tends to zero, in agreement with the impedance control objective presented in eq. (11). However, it can be observed that in the case of Hogan's controller, the impedance-error components increase during the interaction with the environment, contrasting with those obtained with the controllers derived from the approach proposed herein.

The forces generated by the contact between the end-effector and the wall in the x- and y-direction, over time, are depicted in Figure 4. Notice that the contact forces are null during the motion in free space and become different from zero after contact. Such force components are due to impact and friction phenomena, which appears when the

end-effector is moving over the surface. The maximum force component appears when the end-effector contacts the wall, however, as detailed in eq. (7), the robot follows the trajectory imposed by the environment with a compliant behavior of the end-effector. Hence, the force values decrease when the interaction is regulated.

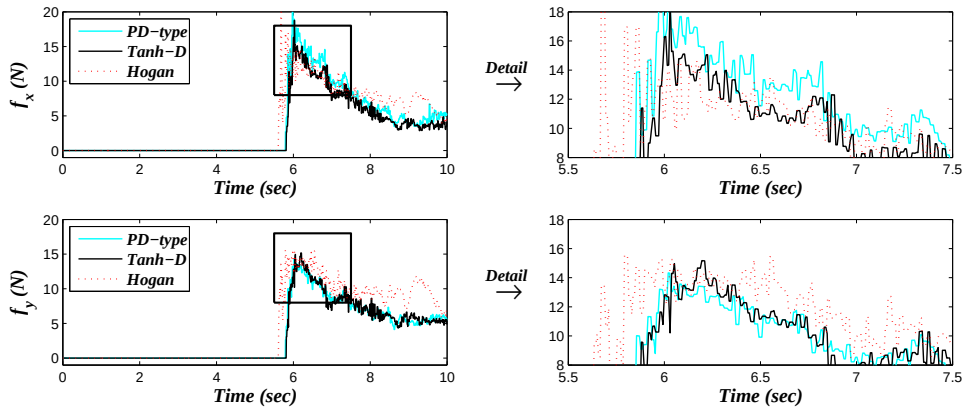

Fig. 4. Interaction forces in x and y directions, respectively.

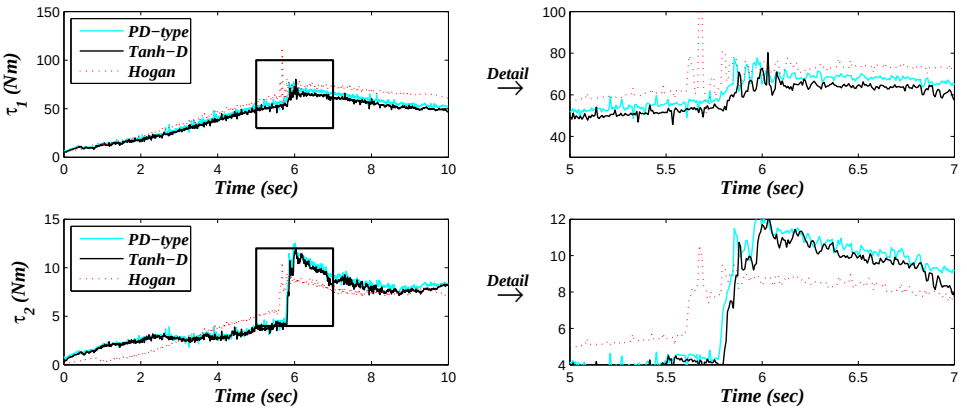

Fig. 5. Applied torques to the manipulator's joints: shoulder and elbow, respectively.

Figure 5 depicts the torques applied to each joint. As seen in these graphs, the torques generated for the impedance controllers do not exceed the saturation limits of the robot motors listed in Table 1, ensuring their correct operation.

Previous results do not allow us to compare the tracking performance of the two proposed controllers and the Hogan's controller. With this purpose, the $\mathcal{L}_2$ norm criterion has proven useful. The scalar-valued $\mathcal{L}_2$ norms of impedance error and its first derivative

are given by

$$\|\tilde{\boldsymbol{\xi}}\|_{\mathcal{L}_2} = \sqrt{\frac{1}{T_e}\int_0^{T_e} \|\tilde{\boldsymbol{\xi}}\|^2 \,\mathrm{d}t} \tag{29}$$

$$\|\dot{\tilde{\boldsymbol{\xi}}}\|_{\mathcal{L}_2} = \sqrt{\frac{1}{T_e}\int_0^{T_e} \|\dot{\tilde{\boldsymbol{\xi}}}\|^2 \,\mathrm{d}t} \tag{30}$$

where $T_e \in \mathbb{R}_+$ is the duration of the experimental test, in this case, 10 seconds. A small $\mathcal{L}_2$ value represents a better controller performance.

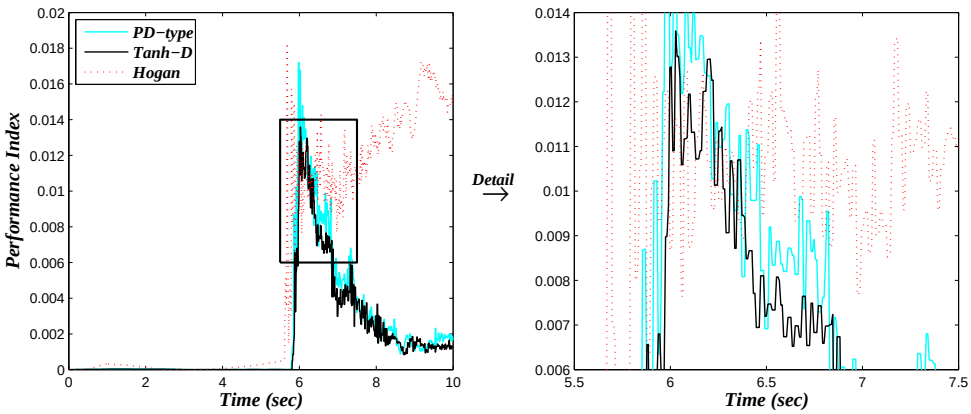

Fig. 6. Performance index including the interaction forces.

The performance results represent the RMS average of the impedance errors of five runs, under the same operation conditions. The performance indexes for the PD-type impedance controller were $\|\tilde{\boldsymbol{\xi}}\|_{\mathcal{L}_2} = 0.0054$ m and $\|\dot{\tilde{\boldsymbol{\xi}}}\|_{\mathcal{L}_2} = 0.0083$ m/sec, whereas for the Tanh-D controller were $\|\tilde{\boldsymbol{\xi}}\|_{\mathcal{L}_2} = 0.0047$ m and $\|\dot{\tilde{\boldsymbol{\xi}}}\|_{\mathcal{L}_2} = 0.0062$ m/sec. On the other hand, for the Hogan's controller were $\|\tilde{\boldsymbol{\xi}}\|_{\mathcal{L}_2} = 0.0664$ m, $\|\dot{\tilde{\boldsymbol{\xi}}}\|_{\mathcal{L}_2} = 0.0328$ m/sec. Considering these results, we can conclude that the Tanh-D impedance controller had the best tracking performance when compared using the scalar-valued $\mathcal{L}_2$ norm criterion.

The comparison criteria used above is useful only to measure the performance of controllers in path tracking, however, in an interaction task, the regulation of the forces generated during the contact between the end-effector and its environment is significant in the controller performance. In order to analyze the performance of the impedance controllers considering the contact forces, the interaction performance index presented in [28] was used, which is given by

$$J = \frac{\tilde{\boldsymbol{\xi}}^T \tilde{\boldsymbol{\xi}}}{\boldsymbol{x}_r^T \boldsymbol{x}_r} + \frac{\boldsymbol{f}_e^T \boldsymbol{f}_e}{\boldsymbol{f}_{\max}^T \boldsymbol{f}_{\max}} \tag{31}$$

where $\boldsymbol{x}_r$ is the reference trajectory defined as $\boldsymbol{x}_r = \boldsymbol{x}_d - \boldsymbol{x}_{f_e}$, and $\boldsymbol{f}_{\max}$ represents the maximum force.

The results of the performance evaluation obtained using the index (31) are presented in Figure 6. The data in this figure represent the average of the impedance errors and contact forces of five runs, under the same operation conditions. By considering the smallest value as the best performance, it can be observed that Than-D controller had, marginally, a better behavior during the development of the interaction task.

5. CONCLUSIONS

In this paper, an approach to impedance control was presented, which consists of an extension of motion control in task-space for robot manipulators. The proposed impedance-control approach allows for an accurate trajectory tracking in both interaction task phases, i. e. a correct unconstrained and/or constrained motion tracking. The impedance-control approach leads to a closed-loop equilibrium point locally, asymptotically stable in Lyapunov sense. This impedance-control structure represents a design tool that can be used to generate several impedance controllers. Two controllers were derived from the proposed approach and, in order to evaluate their performance, an experimental validation was implemented using a direct-drive, robot manipulator.

The experimental results enabled us to verify the ability of the proposed impedance-control approach to perform interaction tasks. The scalar-valued $\mathcal{L}_2$ norm provided a suitable index used to compare the tracking performance of the proposed impedance controllers which, for the cases presented herein, favors the Tanh-D controller over the PD-type controller and the controller proposed by Hogan, as it provides the smallest $\mathcal{L}_2$ norm. Additionally the performance J of equation (31) was used to evaluate the impedance controllers during the interaction, and the Tanh-D controller had the best behavior during the development of the task.

A. PROOF OF POSITIVITY OF LYAPUNOV FUNCTION

The first term of (17) is positive definite due to its quadratic form, where $\boldsymbol{K}_v$ and $\boldsymbol{M}_d$ are diagonal positive definite matrices, then

$$\begin{aligned} \boldsymbol{K}_v\boldsymbol{M}_d - \boldsymbol{M}_d\boldsymbol{M}_d &> 0 \\ (\boldsymbol{K}_v - \boldsymbol{M}_d)\boldsymbol{M}_d &> 0 \\ \boldsymbol{K}_v - \boldsymbol{M}_d &> 0 \end{aligned}$$

such that

$$\begin{aligned} \boldsymbol{K}_v &> \boldsymbol{M}_d \\ \boldsymbol{I} &> \boldsymbol{K}_v^{-1}\boldsymbol{M}_d \\ \rho(\boldsymbol{I}) &> \rho(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d) \quad \Longrightarrow \quad \rho(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d) < 1 \end{aligned} \tag{32}$$

where $\rho(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d)$ is the spectral radius of $\boldsymbol{K}_v^{-1}\boldsymbol{M}_d$, defined as $\rho(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d) = \max\{|\lambda| : \lambda \in \sigma(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d)\}$ where $\sigma(\boldsymbol{K}_v^{-1}\boldsymbol{M}_d)$ is the spectrum or set of all eigenvalues of $\boldsymbol{K}_v^{-1}\boldsymbol{M}_d$ [10]. Then, in order to fulfill the condition (32), the following is selected,

$$\lambda_{\min}\{\boldsymbol{K}_v\} > \lambda_{\max}\{\boldsymbol{M}_d\}. \tag{33}$$

While the second term of (17), $\mathcal{U}_p(\tilde{\boldsymbol{\xi}})$, is the artificial potential energy function induced by the control law, and also it is a positive definite function with respect to the impedance error $\tilde{\boldsymbol{\xi}}$, according to the design properties formulated in Theorem 3.1, and thus $V(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) > 0$ as well as a radially unbounded function.

B. TIME DERIVATIVE OF LYAPUNOV FUNCTION

The time derivative of Lyapunov function (17) is given by

$$\begin{aligned}\dot{V}(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) &= \begin{bmatrix}\tilde{\boldsymbol{\xi}} \\ \dot{\tilde{\boldsymbol{\xi}}}\end{bmatrix}^T \begin{bmatrix}\boldsymbol{K}_v & \boldsymbol{M}_d \\ \boldsymbol{M}_d & \boldsymbol{M}_d\end{bmatrix} \begin{bmatrix}\dot{\tilde{\boldsymbol{\xi}}} \\ \ddot{\tilde{\boldsymbol{\xi}}}\end{bmatrix} + \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}})^T\dot{\tilde{\boldsymbol{\xi}}} \\ &= \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{M}_d\ddot{\tilde{\boldsymbol{\xi}}} + \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{M}_d\dot{\tilde{\boldsymbol{\xi}}} + \tilde{\boldsymbol{\xi}}^T\boldsymbol{M}_d\ddot{\tilde{\boldsymbol{\xi}}} + \tilde{\boldsymbol{\xi}}^T\boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}})^T\dot{\tilde{\boldsymbol{\xi}}}. \qquad (34)\end{aligned}$$

Along the trajectories of the closed-loop equation (19), the derivative (34) can be rewritten as

$$\begin{aligned}\dot{V}(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) &= \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{M}_d\{-\boldsymbol{M}_d^{-1}[\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) + \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}}]\} + \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{M}_d\dot{\tilde{\boldsymbol{\xi}}} + \tilde{\boldsymbol{\xi}}^T\boldsymbol{M}_d\{-\boldsymbol{M}_d^{-1}[\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) \\ &\quad + \boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}}]\} + \tilde{\boldsymbol{\xi}}^T\boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}})^T\dot{\tilde{\boldsymbol{\xi}}}, \\ &= -\dot{\tilde{\boldsymbol{\xi}}}^T\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) - \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \dot{\tilde{\boldsymbol{\xi}}}^T\boldsymbol{M}_d\dot{\tilde{\boldsymbol{\xi}}} - \tilde{\boldsymbol{\xi}}^T\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) - \tilde{\boldsymbol{\xi}}^T\boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} + \tilde{\boldsymbol{\xi}}^T\boldsymbol{K}_v\dot{\tilde{\boldsymbol{\xi}}} \\ &\quad + \nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}})^T\dot{\tilde{\boldsymbol{\xi}}}.\end{aligned}$$

Finally, after some algebra

$$\dot{V}(\tilde{\boldsymbol{\xi}}, \dot{\tilde{\boldsymbol{\xi}}}) = -\tilde{\boldsymbol{\xi}}^T\nabla\mathcal{U}_p(\tilde{\boldsymbol{\xi}}) - \dot{\tilde{\boldsymbol{\xi}}}^T(\boldsymbol{K}_v - \boldsymbol{M}_d)\dot{\tilde{\boldsymbol{\xi}}}. \qquad (35)$$

C. PRACTICAL ISSUES AND IMPLEMENTATION OF IMPEDANCE CONTROLLERS

C.1. Tuning of the impedance parameters

The selection of the impedance-parameter values can be carried out according to the desired dynamic behavior during interaction. By considering $\boldsymbol{K}_d = \text{diag}\{[k_d \cdots k_d]\}$, $\boldsymbol{B}_d = \text{diag}\{[b_d \cdots b_d]\}$ and $\boldsymbol{M}_d = \text{diag}\{[m_d \cdots m_d]\}$, equation (7) can be rewritten as

$$m_d(\ddot{\boldsymbol{x}}_d - \ddot{\boldsymbol{x}}) + b_d(\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}) + k_d(\boldsymbol{x}_d - \boldsymbol{x}) = \boldsymbol{f}_e. \qquad (36)$$

In addition, from the model (6), $\boldsymbol{f}_e = k_e(\boldsymbol{x} - \boldsymbol{x}_e)$. If $\boldsymbol{x}_d$ is constant, the dynamics of the manipulator and environment system along the operational space is described by

$$m_d\ddot{\boldsymbol{x}} + b_d\dot{\boldsymbol{x}} + (k_d + k_e)\boldsymbol{x} = k_d\boldsymbol{x}_d + k_e\boldsymbol{x}_e. \qquad (37)$$

Then, the following transfer function is obtained

$$\mathcal{F}(s) = \frac{1}{s^2 + \frac{b_d}{m_d}s + \frac{k_d+k_e}{m_d}}. \tag{38}$$

According to the known response of a second-order system

$$\omega_n^2 = \frac{k_d + k_e}{m_d} \qquad \omega_n = \sqrt{\frac{k_d + k_e}{m_d}} \tag{39}$$

$$\Longrightarrow$$

$$2\zeta\omega_n = \frac{b_d}{m_d} \qquad \zeta = \frac{b_d}{2\sqrt{m_d(k_d + k_e)}} \tag{40}$$

Therefore, the tuning of impedance parameters is determined by the type of desired response: underdamped, critically damped and overdamped.

C.2. Practical implementation of the force filtering

The force filtering defined in equations (7)–(8) was implemented using a discretized state-space model. By considering $\boldsymbol{K}_d = \text{diag}\{[k_d \cdots k_d]\} \in \mathbb{R}^{m\times m}$, $\boldsymbol{B}_d = \text{diag}\{[b_d \cdots b_d]\} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{M}_d = \text{diag}\{[m_d \cdots m_d]\} \in \mathbb{R}^{m\times m}$, the transfer-function model of force filtering is given by

$$\boldsymbol{x}_{f_e} = \frac{1}{m_d s^2 + b_d s + k_d}\boldsymbol{f}_e, \tag{41}$$

where $\boldsymbol{x}_{f_e} = [x_{f_1} \cdots x_{f_m}]^T$ and $\boldsymbol{f}_e = [f_{e_1} \cdots f_{e_m}]^T$.

The contact forces $\boldsymbol{f}_e$ are measured explicitly by means of an appropriate force sensor, therefore, the system can be decoupled during interaction with the environment, and the force filtering of each component can be written in state space as

$$\boldsymbol{z}_{f_i} = \boldsymbol{A}\boldsymbol{z}_{f_i} + \boldsymbol{B}f_{e_i} \tag{42}$$

where $i = 1, \ldots, m$ and

$$\boldsymbol{z}_{f_i} = \begin{bmatrix} x_{f_i} \\ \dot{x}_{f_i} \end{bmatrix} \tag{43}$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ -k_d/m_d & -b_d/m_d \end{bmatrix} \tag{44}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 \\ 1/m_d \end{bmatrix}. \tag{45}$$

By using a zero-order hold the force filtering can be represented by the following discrete model

$$\boldsymbol{z}_{f_i}(k+1) = \boldsymbol{\Phi}\boldsymbol{z}_{f_i}(k) + \boldsymbol{\Gamma}f_i(k) \tag{46}$$

where k represents the kth sample,

$$\boldsymbol{\Phi} = \mathrm{e}^{\boldsymbol{A}h} = \begin{bmatrix} \alpha_0 & \alpha_1 \\ -(k_d/m_d)\alpha_1 & \alpha_0 - (b_d/m_d)\alpha_1 \end{bmatrix} \tag{47}$$

$$\boldsymbol{\Gamma} = \int_0^h \mathrm{e}^{\boldsymbol{A}\sigma}\boldsymbol{B}\,\mathrm{d}\sigma = \frac{1}{m_d}\begin{bmatrix} \int_0^h \alpha_1\,\mathrm{d}\sigma \\ \int_0^h [\alpha_0 - (b_d/m_d)\alpha_1]\,\mathrm{d}\sigma \end{bmatrix} \tag{48}$$

with h as sampling period and

$$\alpha_0 = \frac{\lambda_1 \mathrm{e}^{\lambda_2 h} - \lambda_2 \mathrm{e}^{\lambda_1 h}}{\lambda_1 - \lambda_2} \tag{49}$$

$$\alpha_1 = \frac{\mathrm{e}^{\lambda_1 h} - \mathrm{e}^{\lambda_2 h}}{\lambda_1 - \lambda_2}. \tag{50}$$

The eigenvalues λ_1 and λ_2 are given by

$$\lambda_{1,2} = \frac{-b_d \pm \sqrt{b_d^2 - 4k_d m_d}}{2m_d}. \tag{51}$$

Therefore, by tuning the impedance parameters (k_d, b_d, m_d) and using the model (46), the adjustment vector $\boldsymbol{x}_{f_e}$ can be computed from $\boldsymbol{z}_{f_i}(k)$, in order to implement the proposed impedance controllers.

(Received November 8, 2011)

REFERENCES

[1] R. Anderson and M. Spong: Hybrid impedance control of robotic manipulators. IEEE Trans. Robotic. Autom. *4* (1988), 5, 549–556.

[2] C. Canudas, B. Siciliano, and G. Bastin: Theory of Robot Control. Springer–Verlag, 1996.

[3] R. Carelli and R. Kelly: An adaptive impedance/force controller for robot manipulators. IEEE Trans. Automat. Control *36* (1991), 8, 967–971.

[4] S. Chiaverini, B. Siciliano, and L. Villani: A survey of robot interaction control schemes with experimental comparison. IEEE-ASME Trans. Mech. *4* (1999), 273–285.

[5] J. González and G. Widmann: A force commanded impedance control scheme for robots with hard nonlinearities. IEEE Trans. Control Syst. Theory *3* (1995), 4, 398–408.

[6] U. Hagn, T. Ortmaier, R. Konietschke, B. Kuebler, U. Seibold, A. Tobergte, M. Nickl, S. Joerg, and G. Hirzinger: Telemanipulators for remote minimally invasive surgery. IEEE Robot. Automat. Magazine *15* (2008), 4, 28–38.

[7] U. Hagn, M. Nickl, S. Jörg, G. Passig, T. Bahls, A. Nothhelfer, F. Hacker, L. Le-Tien, A. Albu-Schäffer, R. Konietschke, M. Grebenstein, R. Warpup, R. Haslinger, M. Frommberger, and G. Hirzinger: The DLR MIRO: A versatile lightweight robot for surgical applications. Ind. Robot *35* (2008), 4, 324–336.

[8] N. Hogan: Impedance control: An approach to manipulation: Part I – Theory, Part II – Implementation and Part III – Applications. J. Dyn. Syst-T ASME *107* (1985), 1–24.

[9] S. Hoon-Kang, M. Jin, and P. Hun-Chang: A solution to the accuracy/robustness dilemma in impedance control. IEEE-ASME Trans. Mech. *14* (2009), 3, 282–294.

[10] R. Horn and C. Johnson: Matrix Analysis. Cambridge University Press, New York 1985.

[11] A. de Jager and J. Banens: Experimental evaluation of robot controllers. In: Proc. 33rd Conf. Decision Control, Lake Buena Vista 1994, pp. 363–368.

[12] A. Jaritz and M. W. Spong: An experimental comparison of robust control algorithms on a direct drive manipulators. IEEE Trans. Control Syst. Theory *4* (1996), 627–640.

[13] H. Kazerooni: Robust nonlinear impedance control for robot manipulators. In: Proc. IEEE Int. Conf. Robotic. Autom. 1987, pp. 741–750.

[14] K. Kim and Y. Hori: Experimental evaluation of adaptive and robust schemes for robot manipulator control. IEEE Trans. Ind. Electron. *42* (1995), 653–662.

[15] H. I. Krebs, M. Ferraro, S. P, Buerger, M. J. Newbery, A. Makiyama, M. Sandmann, D. Lynch, B. T. Volpe, and N. Hogan: Rehabilitation robotics: Pilot trial of a spatial extension for MIT-manus. J. Neuroeng. Rehabil. *1* (2004), 5.

[16] H. I. Krebs, B. T. Volpe, M. L, Aisen, W. Hening, S. Adamovich, H. Poizner, K. Subrahmanyan, and N. Hogan: Robotic applications in neuromotor rehabilitation. Robotica *21* (2003), 3–11.

[17] V. Lippiello, B. Siciliano, and L. Villani: Robot interaction control using force and vision. In: Proc. IEEE-RSJ Int. Conf. Robot. Syst., 2006, pp. 1470–1475.

[18] V. Lippiello, B. Siciliano, and L. Villani: A position-based visual impedance control for robot manipulators. In: Proc. IEEE Int. Conf. Robotic. Autom., Roma 2007, pp. 2068–2073.

[19] L. Marchal-Crespo and D. J. Reinkensmeyer: Review of control strategies for robotic movement training after neurologic injury. J. Neuroeng. Rehabil. *6* (2009), 20.

[20] W. McCormick and H. Schwartz: An investigation of impedance control for robot manipulators. Internat. J. Robot. Res. *12* (1993), 5, 473–489.

[21] A. M. Okamura: Methods for haptic feedback in teleoperated robot-assisted surgery. Ind. Robot *31(6)* (2004), 499–508.

[22] M. Raibert and J. Craig: Hybrid position/force control of manipulators. J. Dyn. Syst-T ASME *102* (1981), 126–133.

[23] F. Reyes and R. Kelly: Experimental evaluation of identification schemes on a direct drive robot. Robotica *15* (1997), 563–571.

[24] L. Sciavicco and B. Siciliano: Modeling and Control of Robot Manipulators. McGraw-Hill, New York 1996.

[25] B. Siciliano and L. Villani: Robot Force Control. Kluwer Academic Publishers, Boston 1999.

[26] M. W. Spong and M. Vidyasagar: Robots Dynamics and Control. John Wiley and Sons, New York 1989.

[27] M. Takegaki and S. Arimoto: A new feedback method for dynamic control of manipulators. J. Dyn. Syst-T ASME *102* (1981), 119–125.

[28] Y. H. Tsoi and S. Q. Xie: Impedance control of ankle rehabilitation robot. In: Proc. IEEE Int. Conf. Robotic. Bio., Bangkok 2009.

[29] L. Whitcomb, A. Rizzi, and D. E. Koditschek: Comparative experiments with a new adaptive controller for robot arms. IEEE Trans. Robotic. Autom. *9* (1993), 59–70.

[30] D. Whitney: Historical perspective and state of the art in robot force control. In: Proc. IEEE Int. Conf. Robotic. Autom. 1985, pp. 262–268.

Marco Mendoza, Facultad de Ciencias, Universidad Autónoma de San Luis Potosí, Av. Salvador Nava S/N, Zona Universitaria, San Luis Potosí, S.L.P., 78290. Mexico.

e-mail: marco.robotica@gmail.com

Isela Bonilla, Facultad de Ciencias, Universidad Autónoma de San Luis Potosí, Av. Salvador Nava S/N, Zona Universitaria, San Luis Potosí, S.L.P., 78290. Mexico.
e-mail: ibonilla@fc.uaslp.mx

Fernando Reyes, Grupo de Robótica, Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, Av. San Claudio y 18 Sur, Col. San Manuel, Puebla, Pue., 72570. Mexico.
e-mail: freyes@ece.buap.mx

Emilio González-Galván, Centro de Investigación y Estudios de Posgrado, Facultad de Ingeniería, Universidad Autónoma de San Luis Potosí, Av. Manuel Nava 8, Zona Universitaria, San Luis Potosí, S.L.P., 78290. Mexico.
e-mail: egonzale@uaslp.mx