AN ADAPTIVE STIFFNESS CONTROL SCHEME FOR ROBOT MANIPULATORS IN TASK-SPACE

Berenice Maldonado-Fregoso, Marco Mendoza and Isela Bonilla
Facultad de Ciencias
Universidad Autónoma de San Luis Potosí
Av. Salvador Nava S/N, Zona Universitaria. San Luis Potosí, SLP, México
email: maldonadobere@gmail.com, marco.mendoza@uaslp.mx, isela.bonilla@uaslp.mx

ABSTRACT
In applications where robot manipulators are in contact with the environment, better known as constrained-motion applications, it is necessary to have a control algorithm that guarantees a suitable interaction between the robot and its environment. The main interaction control schemes in task-space require accurate knowledge of the dynamics of the system to be controlled; then, if the parameters of the environment and the robotic system are unknown, it is essential to use adaptive control schemes. This paper presents an adaptive stiffness control scheme for robot manipulators, that allows to solve the problem of interaction control in the face of parametric uncertainty about the stiffness of the environment and the gravitational forces acting on the robot. The control structure is based on a regressor to estimate the unknown parameters and it is supported by a stability analysis, in the Lyapunov sense, to demonstrate the asymptotic stability of the equilibrium point of the closed-loop system. Finally, some results obtained in simulation are presented in order to verify the correct performance of the proposed control structure.

KEY WORDS
Adaptive control, Parameter uncertainty, Robot manipulator, Stiffness.

1 Introduction
Nowadays robotic systems are employed in many areas where they could or not interact with the environment. In terms of the robot-environment interaction, there are two kind of applications: the constrained-motion applications, when the robot come in contact with the environment; and the unconstrained-motion applications, where the robot is freely moving. In constrained-motion applications is crucial to regulate the interaction forces [1].

In order to solve the constrained-motion control problem, different approaches have been developed and can be classified as direct or indirect methods. The direct methods explicitly use a force-feedback and among the most important schemes can be cited the force control [2, 3], the hybrid force-position control [4, 5, 6] and the parallel force-motion control [1]. In the indirect methods, the force-feedback is not explicit and the force is regulated in-directly using the force/motion relationship, the stiffness and impedance control are the most relevant approaches of this category. The impedance control is based on controlling the relationship between the contact force and the position error resulting from this force and, through the modification of the mechanical impedance, can regulate and control the interaction between the robot and its environment. In its simplest form it can be considered a generalization of damping and stiffness control schemes, in this way it is a PD position controller with adjusted position and velocity feedback gains to obtain different apparent impedances. A general approach of impedance control was introduced by Hogan [7] with a linear mass-spring-damper closed-loop system. This approach was the base to develop several control schemes for interaction applications [8, 9, 10, 11].

The stiffness control approach [12] is based on the concept of linear spring. This scheme use the difference between the position of the end-effector and a constant position, multiplied by a stiffness matrix that represents the environment, so the controller reproduces the behavior of a linear elastic material [13, 14]. The stiffness control problem was recently addressed in [15], where a family of stiffness controllers in task-space is proposed and the corresponding Lyapunov stability analysis is presented; and in [16], where a saturating stiffness control scheme for robots with bounded inputs is proposed.

The aforementioned approaches are useful in cases where accurate measurements of the system parameters are available; however, in cases where these parameters are not available, are unknown or variable, it is necessary to consider adaptive control schemes. In the literature of robot control can be found several adaptive controllers as: a regulation scheme adaptable to the parameters of manipulators with bounded inputs [17]; a joint-space adaptive controller for kinematic and dynamic parameter uncertainty of robot manipulators [18]; an adaptive impedance control for a robotic exoskeleton, which estimates the operator stiffness through surface electromyography to design an impedance model and uses a neural network to estimate the dynamic parameters of the robot [19]; an adaptive control, based on nonlinear sliding modes, with estimation of the kinematic and dynamic parameters of an exoskeleton and that uses the Lyapunov theory to demonstrate the asymptotic stability of the equilibrium point [20]; and an adaptive command-
filtered control to handle dynamic uncertainty, by using a barrier Lyapunov function to deal with the joint-space constraints and the Lyapunov approach to demonstrate the stability of the closed-loop system [21].

Although that in literature exist adaptive controllers, they focus only on the part that involves the parameters of the robot and they do not include the environment with which they interact, besides being few that combine the adaptable part with parametric estimate of the environment, so that, in order to control the robot-environment interaction despite the parametric uncertainty, an adaptive stiffness-control scheme is proposed herein; which is adaptable to the gravitational parameters of the system as well as the stiffness of the environment, what is considered the main contribution of this article. By last the process of updating the parameters is based on a regression matrix and the asymptotic stability of the equilibrium point is proved with the Lyapunov theory.

On the other hand, a highly adaptable and reliable control system is required for robot-assisted therapy, in order to ensure the user’s safety during human-robot interaction [7, 16, 19]. The spasticity present in people who have suffered a stroke is the condition most important to be treated, since is necessary to consider the muscular stiffness of each patient to offer therapies in a personal way; so that, the proposed scheme would be an ideal tool for passive robot-assisted therapies, since it would adapt therapy exercises according with the user's estimated muscular stiffness, thus avoiding possible injuries.

2 Preliminaries

Let \( A \in \mathbb{R}^{m \times n} \) and \( y \in \mathbb{R}^n \). \( \lambda_{\text{min}} \{ A \} \) and \( \lambda_{\text{max}} \{ A \} \) represent the minimum and maximum eigenvalues of \( A \), respectively. \( a_{ij} \) represents the element of \( A \) at its \( i \)-th row and \( j \)-th column, and \( y_i \) is the \( i \)-th element of \( y \). \( \mathbb{R}^n \) denotes the origin of \( \mathbb{R}^n \) and \( I_n \) is the \( n \times n \) identity matrix. Throughout this work, \( \| \cdot \| \) represents the standard Euclidean norm for vectors, i.e., \( \| y \| = \sqrt{y^T y} = \sqrt{\sum_{i=1}^{n} y_i^2} \), and the induced norm for matrices, i.e., \( \| A \| = \sqrt{\lambda_{\text{max}} \{ A^T A \}} \).

Let us consider the dynamic model of a robot manipulator of \( n \) degrees of freedom, interacting with the environment in the \( m \)-dimensional task-space, is given by

\[
M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + g(q) = \tau - J^T(q) f_e, \tag{1}
\]

where \( q, \dot{q}, \ddot{q} \in \mathbb{R}^n \) are the joint position, velocity and acceleration vectors, respectively, \( M(q) \in \mathbb{R}^{n \times n} \) is the inertia matrix, \( C(q, \dot{q}) \in \mathbb{R}^{n \times n} \) is the vector of centripetal and Coriolis torques, and \( g(q) \in \mathbb{R}^n \) is the vector of gravitational torques. The vector of control torques is \( \tau \in \mathbb{R}^n \), \( J(q) \in \mathbb{R}^{m \times n} \) represents the Jacobian matrix of the manipulator, and \( f_e \in \mathbb{R}^n \) is the interaction force vector \[15\].

For simplicity, the interaction forces can be modeled as

\[
f_e = K_c [x - x_e] \tag{2}
\]

where \( K_c \in \mathbb{R}^{m \times m} \) is a symmetric and positive definite stiffness matrix, \( x \in \mathbb{R}^m \) denotes the position vector, and \( x_e \in \mathbb{R}^m \) is the location of the interaction object.

The relationship between the forces applied to the end effector of the robot in the task-space \( f_e \) and the joint torques \( \tau \), is given by

\[
\tau = J^T(q) f_e \tag{3}
\]

Then, the model (1) can be rewritten in task-space as

\[
M_x \ddot{x} + C_x \dot{x} + g_x = f_x - f_e \tag{4}
\]

where \( x \in \mathbb{R}^m \) and \( \dot{x} \in \mathbb{R}^m \) represent the acceleration and velocity of the robot, respectively \[16\]. While

\[
M_x = [J^{-1}(q)]^T M(q) J^{-1}(q) \]
\[
C_x = \{ [J^{-1}(q)]^T C(q, \dot{q}) - M_x J(q, \dot{q}) \} J^{-1}(q) \]
\[
g_x = [J^{-1}(q)]^T g(q)
\]

The next properties of the dynamic model will be useful for the further analysis.

**Property 1.** The task-space dynamic model (4) is linear respect to the robot parameters (mass, inertia, etc.) and the parameters of the environment (stiffness), then, this model can be expressed as

\[
M_x \ddot{x} + C_x \dot{x} + Y(x) \theta = f_x \tag{5}
\]

where \( Y(x) \theta = g_x + f_e \), with \( \theta \in \mathbb{R}^p \) representing the parameters associated with gravitational forces and the stiffness and \( Y(x) \in \mathbb{R}^{m \times p} \) being a regression matrix.

**Property 2.** The inertia matrix \( M_x \) is a symmetric and positive definite matrix

\[
M_x > 0 \quad \Rightarrow \quad M_x = M_x^T \tag{6}
\]

**Property 3.** The inertia matrix \( M_x \) is bounded by

\[
\lambda_{\text{min}} \{ M_x \} I_m \leq M_x \leq \lambda_{\text{max}} \{ M_x \} I_m, \quad \forall x \in \mathbb{R}^m \tag{7}
\]

for some positive constants \( \lambda_{\text{min}} \{ M_x \} \leq \lambda_{\text{max}} \{ M_x \} \).

**Property 4.** The matrix \( C_x \) is related to \( M_x \) by

\[
\dot{x}^T \left[ \frac{1}{2} M_x - C_x \right] \dot{x} = 0, \quad \forall (x, \dot{x}) \in \mathbb{R}^m \times \mathbb{R}^m \tag{8}
\]

and actually

\[
M_x = C_x + C_x^T, \quad \forall (x, \dot{x}) \in \mathbb{R}^m \times \mathbb{R}^m \tag{9}
\]

**Property 5.** The matrix \( C_x \) satisfies \( \| C_x \| \leq k_c \| \dot{x} \|^2 \), \( \forall (x, \dot{x}) \in \mathbb{R}^m \times \mathbb{R}^m \) for some constant \( k_c > 0 \).

The next lemma is used to derive sufficient conditions for a function to be positive definite.

**Lemma 1.** Let \( f : \mathbb{R}^n \to \mathbb{R} \) be a continuously differentiable function with continuous partial derivatives up to at least second order. Assume that
with derivative gains; and \( \hat{\theta} \) during passive therapy exercises \[23\].

This behavior is desirable where the stiffness of the patient's joints is unknown and it imposes an interaction force on the robot. Therefore, the control algorithm must counteract this interaction force to try to reach the desired position. This behavior is desirable during passive therapy exercises \[23\].

In formal terms, the control objective is to design \( f_c \) such that

\[
\lim_{t \to \infty} \tilde{x}(t) = 0
\]

where \( \tilde{x} = x_d - x(t) \in \mathbb{R}^m \) represents the position error; and this goal is fulfilled even though the accurate values of the elements of \( \theta \) in (5) are unknown.

In order to solve this control problem, a nonlinear PD-type algorithm based on the \( \tanh \) function is presented.

### 3.2 Adaptive nonlinear PD-type control structure

The nonlinear PD structure corresponds to a stiffness controller with adaptive gravity/interaction-force compensation, given by

\[
f_c = K_P \tanh(\tilde{x}) - K_D \tilde{x} + Y(x) \dot{\theta}
\]

where \( K_P \in \mathbb{R}^{m \times m} \) is a symmetric positive definite matrix of proportional gains; \( K_D \) \( \dot{x} \) is referred to a dissipative term, with \( K_D \in \mathbb{R}^{m \times m} \) being a symmetric positive definite matrix of derivative gains; and \( \dot{\theta} \) is the estimate vector coming from

\[
\dot{\theta} = -\Gamma \gamma^T(x)[\tilde{x} - \epsilon \tanh(\tilde{x})]
\]

with \( \Gamma \in \mathbb{R}^{P \times P} \) being a positive definite diagonal constant matrix, and \( \epsilon \) being a positive constant satisfying

\[
\epsilon < \min\{\epsilon_1, \epsilon_2\}
\]

where

\[
\epsilon_1 = \sqrt{\frac{\lambda_{\min} \{K_P\}}{\lambda_{\max} \{M_x\}}}
\]

\[
\epsilon_2 = \frac{\lambda_{\min} \{K_P\} \lambda_{\min} \{K_D\}}{\lambda_{\min} \{K_P\} \sqrt{mk_c} + \lambda_{\max} \{M_x\} + \frac{1}{2} \lambda_{\max}^2 \{K_D\}}
\]

The closed-loop system, which is obtained by combining (5), (11) and (12), is given by

\[
\frac{dt}{dt} \begin{bmatrix} \dot{x} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} M_x^{-1} [K_P \tanh(\tilde{x}) - K_D \tilde{x} - C_x \dot{x} - Y(x) \dot{\theta}] \\ \Gamma \gamma^T(x)[\tilde{x} - \epsilon \tanh(\tilde{x})] \end{bmatrix}
\]

where \( \dot{\theta} = \theta - \hat{\theta} \). Let us note that, from Eq. (16) under stationary conditions, i.e. by considering \( \tilde{x} = \bar{x} = 0 \) and \( \dot{\theta} = \bar{\theta} = 0 \), \( 0 \) is the unique equilibrium position error of the closed loop, while the parameter estimation error equilibrium vector \( \tilde{\theta} \in \ker(Y^T(x_d)) \).

In order to perform the corresponding stability analysis in Lyapunov sense, consider the following scalar function

\[
V(\bar{x}, \dot{\bar{x}}, \bar{\theta}) = V_0(\bar{x}, \dot{\bar{x}}) + \frac{1}{2} \dot{\bar{\theta}} \Gamma^{-1} \dot{\bar{\theta}}
\]

where

\[
V_0(\bar{x}, \dot{\bar{x}}) = \frac{1}{2} \bar{x}^T M_x \dot{\bar{x}} - \epsilon \bar{x}^T M_x \tanh(\bar{x})
\]

\[
+ \begin{bmatrix} \sqrt{\ln \cosh(\bar{x}_1)} \\ \vdots \\ \sqrt{\ln \cosh(\bar{x}_m)} \end{bmatrix}^T K_P \begin{bmatrix} \sqrt{\ln \cosh(\bar{x}_1)} \\ \vdots \\ \sqrt{\ln \cosh(\bar{x}_m)} \end{bmatrix}
\]

Observe that from Lemma 1, \( V_0(0, 0, 0) = 0 \) and

\[
\frac{\partial}{\partial \bar{z}} V_0(\bar{x}, \dot{\bar{x}}) = \left[ K_P \tanh(\bar{x}) - \epsilon \bar{S}(\bar{x}) M_x \dot{\bar{x}} \right] M_x \dot{\bar{x}} - \epsilon M_x \tanh(\bar{x})
\]

where

\[
z = \begin{bmatrix} \bar{x} \\ \dot{\bar{x}} \end{bmatrix}
\]

\[
S(\bar{x}) = \text{diag}\{\text{sech}^2(\bar{x}_i)\} \forall i = 1, ..., m
\]

then

\[
\frac{\partial}{\partial \bar{z}} V_0(0, 0, 0) = \begin{bmatrix} 0_m \\ 0_m \end{bmatrix}
\]

On the other hand,

\[
H(z) = \begin{bmatrix} K_P S(\bar{x}) + 2 \epsilon T(\bar{x}) M_x \dot{\bar{x}} - \epsilon S(\bar{x}) M_x \\ -\epsilon S(\bar{x}) M_x \end{bmatrix}
\]

where

\[
T(\bar{x}) = \text{diag}\{\tanh(\bar{x}_i) \text{sech}(\bar{x}_i)\} \forall i = 1, ..., m
\]

then

\[
H(0_{2m}) = \begin{bmatrix} K_P - \epsilon M_x \\ -\epsilon M_x \end{bmatrix}
\]
and by using the Schur complement [24]

\[ H(0_{2m}) > 0 \iff M_k > 0 \quad \text{and} \quad K_P - \varepsilon^2 M_k > 0 \]

Therefore, by considering Properties 2 and 3, and satisfying (13), \( V_0(\bar{x}, \bar{\theta}) \) is positive definite, thus \( V(\bar{x}, \bar{\theta}) \) is concluded to be positive definite.

Thereafter, the time-derivative of (17), along the trajectories of the closed-loop system (16), is given by

\[
\dot{V}(\bar{x}, \bar{\theta}) = \dot{x}^T M_k \dot{x} + \varepsilon \dot{x}^T \dot{M}_k \tanh(\bar{x}) - \varepsilon \dot{x}^T M_k \tanh(\bar{x}) + \varepsilon \dot{x}^T S(\bar{x}) M_k \dot{x}
\]

\[
- \dot{x}^T K_P \tanh(\bar{x}) + \bar{\theta}^T \Gamma^{-1} \bar{\theta}
\]

\[
= - \dot{x}^T K_P \dot{x} - \varepsilon \tanh^T(\bar{x}) K_P \tanh(\bar{x}) + \varepsilon \tanh^T(\bar{x}) K_D \dot{x} - \varepsilon \dot{x}^T C \tanh(\bar{x}) + \varepsilon \dot{x}^T S(\bar{x}) M_k \dot{x}
\]

(22)

where Properties 2, 3 and 4 were used. The derivative (22) can be lower-bounded by

\[
\dot{V}(\bar{x}, \bar{\theta}) \leq -W_0(\bar{x}, \bar{\theta})
\]

(23)

where

\[
W_0(\bar{x}, \bar{\theta}) = \left[ \frac{|| \tanh(\bar{x}) ||}{|| \bar{x} ||} \right]^T \bar{Q}_0 \left[ \frac{|| \tanh(\bar{x}) ||}{|| \bar{x} ||} \right]
\]

(24)

with

\[
\bar{Q}_0 = \begin{bmatrix} \varepsilon \lambda_{\min}\{K_P\} & -\varepsilon \lambda_{\max}\{K_D\} \\ -\frac{\varepsilon}{2} \lambda_{\max}\{K_D\} & \lambda_{\min}\{K_D\} - \varepsilon \sqrt{\lambda_{\max}\{M_k\}} \end{bmatrix}
\]

(25)

From inequality (13), \( W_0(\bar{x}, \bar{\theta}) \) is positive definite and \( \dot{V}(\bar{x}, \bar{\theta}) \leq 0 \) with \( \dot{V}(\bar{x}, \bar{\theta}) = 0 \iff (\bar{x}, \bar{\theta}) = (0_m, 0_m) \). Therefore, by defining the following set

\[
\Omega = \{(\bar{x}, \bar{\theta}) \in \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^p : \dot{V}(\bar{x}, \bar{\theta}) = 0\}
\]

it can be concluded that the closed-loop equilibrium point is asymptotically stable by the invariance theory, because \((0_m, 0_m, \bar{\theta}_E)\) is the only initial state in \( \Omega \) whose corresponding solution remains forever in \( \Omega \) [22].

4 Simulation results

The performance of the proposed controller (11) was verified through numerical simulations. A robotic manipulator of two degrees of freedom interacting with a wall located in the workspace, were considered. The wall has a stiffness 10,000 N/m and Fig. 1 shows the simulation set-up, where the green bars represent the robot, the red line represents the wall and the yellow dot represents the desired position.

![Figure 1. Graphic representation of the simulation set-up.](image1)

Figure 2. Robot of two degrees of freedom.

4.1 Robot model

The dynamical model of the robot (Fig. 2) is characterized by the following inertia matrix

\[
M(q) = \begin{bmatrix} \theta_{11} & \theta_{12}\cos(q_1 - q_2) & \theta_{13} \\ \theta_{12}\cos(q_1 - q_2) & \theta_{12}\sin(q_1 - q_2) & 0 \\ \theta_{13} & 0 & \theta_{13} \end{bmatrix}
\]

(26)

where

\[
\theta_{11} = m_1 l_1^2 + m_2 l_2^2 + m_3 l_3^2 + m_4 l_4^2 + l_1 + l_3
\]

\[
\theta_{12} = m_3 l_2 l_3 + m_{14} l_4
\]

\[
\theta_{13} = m_2 l_2^2 + m_3 l_3^2 + m_{14} l_4 + l_2 + l_4
\]

the matrix of centripetal and Coriolis forces given by

\[
C(q, \dot{q}) = \begin{bmatrix} 0 & \theta_{12}\dot{q}_1 \sin(q_1 - q_2) \\ -\theta_{12}\dot{q}_1 \sin(q_1 - q_2) & 0 \\ \theta_{12}\dot{q}_2 \sin(q_1 - q_2) & 0 \end{bmatrix}
\]

(27)

and the following vector of gravitational forces

\[
g(q) = \begin{bmatrix} \theta_{G1} \sin(q_1) \\ \theta_{G2} \sin(q_2) \end{bmatrix}
\]

(28)

where

\[
\theta_{G1} = m_1 l_1 c_1 + m_3 l_3 c_2 + m_4 l_4 a_c
\]

\[
\theta_{G2} = m_2 l_2 c_2 + m_3 l_3 c_2 + m_{14} l_4 a_c
\]

with \( m_i, l_i, l_{ci} \) and \( l_i \) being the mass, length, center of mass and inertia moment of link \( i \), respectively; and \( a_c \) representing the acceleration of gravity.
On the other hand, the direct kinematics is
\[
\begin{bmatrix}
  x_1 \\
  x_2
\end{bmatrix} = \begin{bmatrix}
  l_1 \sin q_1 + l_4 \sin q_2 \\
  -l_1 \cos q_1 - l_4 \cos q_2
\end{bmatrix}
\]
(29)
where \( x_1 \) is the \( x \)-component and \( x_2 \) is the \( y \)-component of the end effector position (see Fig. 2). Finally, the corresponding Jacobian matrix is given by
\[
J = \begin{bmatrix}
  l_1 \cos q_1 & l_4 \cos q_2 \\
  l_1 \sin q_1 & l_4 \sin q_2
\end{bmatrix}
\]
(30)

Table 1. Parameter values.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Notation</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of link 1</td>
<td>( l_1 )</td>
<td>0.5</td>
<td>m</td>
</tr>
<tr>
<td>Length of link 2</td>
<td>( l_2 )</td>
<td>0.3</td>
<td>m</td>
</tr>
<tr>
<td>Length of link 3</td>
<td>( l_3 )</td>
<td>0.5</td>
<td>m</td>
</tr>
<tr>
<td>Length of link 4</td>
<td>( l_4 )</td>
<td>0.8</td>
<td>m</td>
</tr>
<tr>
<td>Inertial parameter 1</td>
<td>( \theta_1 )</td>
<td>3.48</td>
<td>kg m(^2)</td>
</tr>
<tr>
<td>Inertial parameter 2</td>
<td>( \theta_2 )</td>
<td>0.145</td>
<td>kg m(^2)</td>
</tr>
<tr>
<td>Inertial parameter 3</td>
<td>( \theta_3 )</td>
<td>1.16</td>
<td>kg m(^2)</td>
</tr>
<tr>
<td>Gravitational parameter 1</td>
<td>( \theta_{G1} )</td>
<td>34.14</td>
<td>kg m(^2)</td>
</tr>
<tr>
<td>Gravitational parameter 2</td>
<td>( \theta_{G2} )</td>
<td>11.38</td>
<td>kg m(^2)</td>
</tr>
</tbody>
</table>

Table 1 shows the values of the robot parameters. Therefore, for this robot manipulator, Properties 3 and 5 are satisfied with \( \lambda_{\text{max}} \{M_x\} = 2.38 \times 10^4 \) and \( k_c = 4.89 \times 10^8 \).

While to implement the adaptive compensation term \( Y(x) \dot{\hat{\theta}} \) we obtain that
\[
Y(x) = \begin{bmatrix}
  y_{11} & 0 & 0 & 0 \\
  0 & y_{22} & y_{23} & y_{24}
\end{bmatrix}
\]
(31)
where
\[
\begin{align*}
  y_{11} &= -[\sin q_1 \sin q_2]/\sin(q_1 - q_2) \\
  y_{22} &= -[\cos q_1 \sin q_2]/\sin(q_1 - q_2) \\
  y_{23} &= [\sin q_1 \cos q_2]/\sin(q_1 - q_2) \\
  y_{24} &= \begin{cases} x_2 - x_2 & \text{if } f_e \neq 0 \\ 0 & \text{if } f_e = 0 \end{cases}
\end{align*}
\]

and the corresponding vector of parameters \( \hat{\theta} \in \mathbb{R}^4 \), with \( \hat{\theta}_1, \hat{\theta}_2 \) and \( \hat{\theta}_3 \) being the parameters related to gravitational forces and \( \hat{\theta}_4 \) being the parameter related to environmental stiffness. The vector of real parameters is
\[
\theta = \begin{bmatrix}
  54.05 \\
  14.22 \\
  68.28 \\
  10000
\end{bmatrix}
\]

4.2 Results

For the implementation of adaptive controller (11), the initial values of the parameters were \( \hat{\theta}(0) = [50 10 60 8000]^T \).

Figure 3. Test 1: Components of the position vector \( x \), where \( x_1 \) corresponds to the \( x \)-component and \( x_2 \) corresponds to the \( y \)-component.

The system was tuned empirically looking for the best time-response and the gains were selected as
\[
K_P = \begin{bmatrix}
  6500 & 0 \\
  0 & 4500
\end{bmatrix}
\]
\[
K_D = \begin{bmatrix}
  1300 & 0 \\
  0 & 900
\end{bmatrix}
\]
\[
\Gamma = \begin{bmatrix}
  100 & 0 & 0 & 0 \\
  0 & 400 & 0 & 0 \\
  0 & 0 & 250 & 0 \\
  0 & 0 & 0 & 1 \times 10^6
\end{bmatrix}
\]
\[
\varepsilon = 1.17 \times 10^{-6}
\]

4.2.1 Test 1: Regulation to a single point

Test 1 consisted of moving the robot from the initial position \( x(0) = (-0.21, -0.92) \) to the desired position \( x_d = (0, -1) \) m and interacting with a wall, parallel to the \( x \) axis, located at a distance of 0.99 m.

The results for the test 1 are shown in Figures 3 - 6. In Figure 3, it can be observed the time-response of the position of the robot’s end effector, where both components tend to the desired position; however, during robot-environment interaction, a small steady-state error is observed due to the stiffness presented by the wall.

Figure 4 shows the time-response of the interaction force and it can be observed a zero force while there is no contact with the wall. Then, when there is robot-environment interaction, the force increase according to the wall stiffness and finally the \( y \)-component of force converges to a value close to 62 N.

Figures 5 and 6 show that, as the time evolves, the estimates of gravity and stiffness parameters remain bounded as expected, based on the stability analysis of the proposed controller.
4.2.2 Test 2: Point-to-point trajectory

Test 2 consisted in getting that the robot’s end effector tracks a point-to-point path over the same wall. By using linear interpolators, the path planning was as follows:

- Starting from the initial position $x(0) = (-0.11, -0.77)$ m, the robot’s end effector must reach the position $(-0.11, -1.0)$ m, in 4 seconds.
- Then, the robot must arrive to $(0.3, -1.0)$ m in 6 seconds.
- Finally, the robot must return to the initial position $x(0)$ in 4 seconds and stay there.

The results for the test 2 are shown in Figures 7-11. The Figure 7 shows the trajectory followed by the robot’s end effector. In Figure 8, it can be observed the time-response of the position of the robot’s end effector, where both components tend to the desired position; and in the same way that in test 1, during robot-environment interaction, a small steady-state error is observed due to the stiffness presented by the wall.

Figure 9 shows the time-response of the interaction force and also it can be observed a zero force while there is no contact with the wall. Then, when there is robot-environment interaction, the force increases according to the wall stiffness.

Figures 10 and 11 show that, the gravity and stiffness parameters are adapted as the robot tracks the trajectory and their values remain bounded. Therefore, it can be concluded that the control scheme works properly and the control objective is achieved in both simulation tests.

5 Conclusions

In this paper, an adaptive stiffness control scheme for robot-environment interaction with parametric uncertainty has been presented. The suitable performance of the control
scheme is supported by stability analysis in the Lyapunov sense and its efficiency was verified through simulation. The control scheme has a Proportional-Derivative structure, that combines a nonlinear proportional action with a linear dissipative action, and compensate gravity and interaction forces by a regressor-based adaptive term.

A potential area of application of this control scheme are the therapies of rehabilitation assisted by robots, because people who have suffered a stroke presents spasticity, so the proposed controller would be useful to provide rehabilitation therapies where the robot can adapt to the muscular stiffness of each patient, thus achieving personalized therapies.

Despite the suitable results obtained in simulation, as future work it is necessary to carry out experimental tests that allow to verify the proper performance of the adaptive control scheme presented herein.

Acknowledgements: The authors wish to thank CONACYT for the support provided through the project no. 222316.

References


