



Extended Model-Based Feedforward Compensation in \mathcal{L}_1 Adaptive Control for Mechanical Manipulators: Design and Experiments

Moussab Bennehar, Ahmed Chemori, François Pierrot and Vincent Creuze*

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM), Université de Montpellier-CNRS, Montpellier, France

OPEN ACCESS

Edited by:

Filippo Arrichiello,
University of Cassino and Southern
Lazio, Italy

Reviewed by:

Dongbin Lee,
Oregon Institute of Technology, USA
Francesco Pierri,
University of Basilicata, Italy
Elias Kosmatopoulos,
Democritus University of Thrace,
Greece

*Correspondence:

Vincent Creuze
vincent.creuze@lirmm.fr

Specialty section:

This article was submitted to Robotic Control Systems, a section of the journal *Frontiers in Robotics and AI*

Received: 03 August 2015

Accepted: 23 November 2015

Published: 09 December 2015

Citation:

Bennehar M, Chemori A, Pierrot F and Creuze V (2015) Extended Model-Based Feedforward Compensation in \mathcal{L}_1 Adaptive Control for Mechanical Manipulators: Design and Experiments. *Front. Robot. AI* 2:32. doi: 10.3389/frobt.2015.00032

This paper deals with a new control scheme for parallel kinematic manipulators (PKMs) based on the \mathcal{L}_1 adaptive control theory. The original \mathcal{L}_1 adaptive controller is extended by including an adaptive loop based on the dynamics of the PKM. The additional model-based term is in charge of the compensation of the modeled non-linear dynamics in the aim of improving the tracking performance. Moreover, the proposed controller is enhanced to reduce the internal forces, which may appear in the case of redundantly actuated PKMs (RA-PKMs). The generated control inputs are first regulated through a projection mechanism that reduces the antagonistic internal forces, before being applied to the manipulator. To validate the proposed controller and to show its effectiveness, real-time experiments are conducted on a new four degrees-of-freedom (4-DOFs) RA-PKM developed in our laboratory.

Keywords: parallel kinematic manipulators, \mathcal{L}_1 adaptive control, actuation redundancy, experiments, non-linear control

1. INTRODUCTION

Adaptive control of mechanical manipulators has gained an increased interest in the last few decades. Indeed, for an efficient control of such systems, uncertainties, external disturbances, and variations in the dynamics have to be considered in the control scheme design. Since conventional non-adaptive model-based controllers rely mainly on the dynamic model of the manipulator, they may fail if this last one is not sufficiently accurate. This is the case when some phenomena are not modeled (e.g., friction, actuators' dynamics, etc.) or when simplifying hypotheses are considered without careful analysis. In both cases, the controller will not be able to keep the desired performance and may even lead to instability of the closed-loop system. In such situation, adaptive control seems to be the appropriate solution to overcome these shortcomings. Indeed, adaptive controllers are mainly known for their ability to estimate unknown, varying, or uncertain parameters of the system and its environment and to adapt the controller accordingly.

The earlier designed adaptive controllers were mainly based on model reference adaptive control (MRAC). These controllers hold the advantage of not relying on any knowledge of the dynamics of the system. MARC-based strategies were mainly applied on serial manipulators. For instance, a decentralized control scheme based on MRAC has been proposed in Dubowsky and DesForges (1979) to control a 3-DOFs robotic arm. In this work, the manipulator was supposed to be controlled by a PD feedback loop with adjustable gains. The step response of each joint of the manipulator had to follow a user-defined step response generated by a second order reference model. However, a

very restrictive assumption was made in this study that the coupling between the different joints was neglected. A more sophisticated controller based on MRAC was proposed in Horowitz and Tomizuka (1986), where the coupled dynamics of the manipulator has been explicitly considered in the controller design. In this work, the proposed control law consists of a fixed PD feedback loop in addition to an adaptive loop intended to compensate for non-linearities of the system. However, this controller was not suitable for real-time setups. In fact, the number of the controller parameters to be estimated was too large (11 parameters for a simple 3-DOFs manipulator, after simplification). In addition to these issues, MRAC inherently suffers from a major drawback; the estimation and the control loops are coupled (Hovakimyan and Cao, 2010). This is mainly due to the fact that the adaptive gain in MRAC acts as a feedback gain, meaning that a trade-off has to be made between the speed of adaptation and the robustness of the controller. Recently, a new strategy called \mathcal{L}_1 adaptive control has been developed in order to overcome this problem (Cao and Hovakimyan, 2006a,b). The \mathcal{L}_1 adaptive control structure features a unique filtering technique that enables a decoupling between the estimation and the control loops. This means that the adaptation can be made arbitrarily fast while guaranteeing sufficient robustness margins.

With the advances in modeling techniques, obtaining an accurate dynamic model for a robot manipulator became relatively an easy task (Merlet, 2006). Consequently, model-based controllers that take advantage of the dynamics knowledge in the control design captured an increasing attention. Craig et al. proposed in Craig et al. (1987) an adaptive version of the computed-torque control scheme (Markiewicz, 1973). However, the proposed controller showed several limitations that restricted its real-time implementation (e.g., joint accelerations had to be measured). Later on, several improvements have been brought to model-based adaptive control resulting in more reliable, simpler, and more efficient adaptive schemes (Slotine and Weiping, 1987; Ortega and Spong, 1989; Sadegh and Horowitz, 1990; Shang and Cong, 2010). One particular control strategy that caught our attention is the desired compensation adaptation law (DCAL) proposed in Sadegh and Horowitz (1990). DCAL features many advantageous characteristics that make it suitable for real-time implementation. One interesting feature, among others, is that it relies on the desired trajectories of the manipulator instead of the measured ones, which means that it is less sensitive to measurement noise and enables offline computation of some terms in the control law. Nevertheless, one common drawback of model-based controllers is that the non-modeled dynamics are not taken into consideration by the adaptive loop. Instead, the feedback loop (usually a PD) is in charge of rejecting the eventual residual non-linearities. Though a PD feedback loop is usually sufficient for slow accelerations, the performance on high speeds may be improved by using more sophisticated feedback actions (Bennehar et al., 2014a,b).

In this paper, a new adaptive control scheme is proposed, it combines the benefits of both \mathcal{L}_1 and model-based adaptive control. It was shown in (Bennehar et al., 2015) that \mathcal{L}_1 adaptive control applied to a 4-DOFs PKM outperforms the PD controller in terms of tracking performance. In this work, we further

improve \mathcal{L}_1 adaptive control by augmenting it with a model-based adaptive term that accounts for the modeling uncertainties with known structure. We compare the performance of both \mathcal{L}_1 adaptive controller and the augmented one through real-time experiments on a new 4-DOFs RA-PKM developed in our laboratory. The remainder of this paper is organized as follows. In section 2, the dynamic modeling, required for the proposed controller design, is addressed. In section 3, a background on \mathcal{L}_1 adaptive control for mechanical manipulators is provided. The main contribution of the actual paper, being a new controller based on \mathcal{L}_1 and model-based adaptive control and its application to RA-PKMs are detailed in section 4. Real-time experiments performed on ARROW and the obtained results are presented and discussed in section 5. Finally, conclusions are drawn in section 6.

2. DYNAMIC MODELING OF THE PLANT: THE ARROW ROBOT

ARROW (acronym for Accurate and Rapid Robot with large Operational Workspace) is a 4-DOF RA-PKM intended to be used for fast lightweight machining tasks. It has two degrees of actuation redundancy since it is equipped with six linear actuators and has four degrees of freedom. All the actuators are aligned along the same axis and are split into two sets, each set consists of three actuators (cf. **Figure 1**). The actuators are connected to the end effector through two pairs of simple arms and one pair of parallelograms. As its acronym indicates, the ARROW robot, developed in the framework of the French ANR-ARROW project, has been designed in the aim of creating a very fast and accurate PKM with large workspace and homogeneous performance throughout its workspace. Moreover, its platform is capable of performing $\pm 90^\circ$ rotations free of any singularities. The CAD design of ARROW is shown in **Figure 1** and its main geometrical and dynamical parameters are summarized in **Table 1**. A more detailed description about the design theory and the development of ARROW robot can be found in Shayya (2015).

For the subsequent control development, the establishment of the dynamic model of ARROW is required. The latter will be included in the control scheme design in order to enhance the dynamic performance and tracking capabilities of the robot. First, let us enumerate the considered hypotheses, mainly common to PKMs (Corbel et al., 2010), in order to simplify the establishment of the dynamic model:

1. Both dry and viscous friction forces are neglected since all active and passive joints are designed such as to minimize friction effects, even though, their insignificant effect could be considered in control as it will be explained later on.
2. The mass of each arm is split up into two halves being located at the end tips of the arm. Hence, the influence of the mass of the arms is considered with the actuators and the platform. This assumption can be justified by the small mass of the arms compared to the other components. Similarly, the mass of the parallelograms is also split up into two halves and considered in both the dynamics of the actuators and the moving platform.

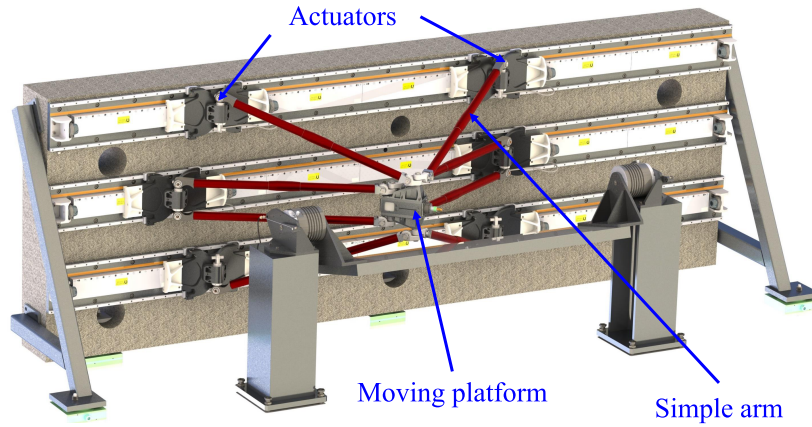


FIGURE 1 | CAD schematic view of ARROW PKM.

TABLE 1 | Main parameters of ARROW PKM.

Item	Value	Item	Value
Slider cart's mass	9.7 kg	Simple arm's mass	1.4 kg
Platform's mass	9.2 kg	Parallelogram's length	0.61 m
Parallelogram's mass	2.2 kg	Simple arm's length	0.96 m
Platform's inertia	0.514 kg m ²		

Applying the law of motion, the dynamics of the actuators are given by the following equation (Shayya et al., 2014):

$$M_a \ddot{q} = \Gamma - J_q^T f \quad (1)$$

being $M_a = \text{diag}(m_{as}, m_{as}, m_{ap}, m_{ap}, m_{as}, m_{as})$, where m_{as} includes the mass of the actuator, the moving cart, and the half-mass of the simple arm. m_{ap} includes the mass of the actuator, the moving cart, and the half-mass of the parallelogram (i.e., two simple arms). $\ddot{q} \in \mathbb{R}^6$ is the joint acceleration vector, $\Gamma \in \mathbb{R}^6$ is the input force vector, $J_q \in \mathbb{R}^{6 \times 6}$ is the joint Jacobian matrix, and $f \in \mathbb{R}^6$ is the force vector resulting from acceleration and gravitational forces acting on the moving platform. From the platform side, the dynamics are governed by Shayya et al. (2014)

$$M_p \ddot{X} + \Lambda_c \dot{X} = J_x^T f + m_p G \quad (2)$$

where $M_p \in \mathbb{R}^{4 \times 4}$ is the inertia matrix of the moving platform, $\ddot{X} \in \mathbb{R}^4$ is its acceleration vector, $\Lambda_c \in \mathbb{R}^{4 \times 4}$ is the matrix of Coriolis and centrifugal effects, $\dot{X} \in \mathbb{R}^4$ is the velocity vector of the moving platform, $J_x \in \mathbb{R}^{6 \times 4}$ is the Cartesian Jacobian matrix, m_p is the total mass of the platform, including the half-masses from the arms and parallelograms and $G = [0, 0, -10, 0]^T$ m/s² is the gravity wrench acting on the platform. The Jacobian matrices J_q and J_x links the joint velocities and those of the moving platform according to the following relationship:

$$J_q \dot{q} = J_x \dot{X} \quad (3)$$

where $\dot{q} \in \mathbb{R}^6$ is the joint velocities and $\dot{X} \in \mathbb{R}^4$ denotes the velocities of the moving platform. Solving for f in (1) and substituting the resulting equation in (2) while replacing $\ddot{q} = J_m^{-1} \ddot{X} + \dot{J}_m \dot{X}$,

where $J_m = J_q^{-1} J_x$ the inverse Jacobian matrix, yields the following equation:

$$\ddot{X} = H \Gamma - \Lambda \dot{X} + A_G \quad (4)$$

where $H \in \mathbb{R}^{4 \times 6}$, $\Lambda \in \mathbb{R}^{4 \times 4}$, and $A_G \in \mathbb{R}^4$ are expressed by

$$\begin{cases} H = (M_p + J_m^T M_a J_m)^{-1} J_m^T \\ \Lambda = H M_a \dot{J}_m + (M_p + J_m^T M_a J_m)^{-1} \Lambda_c \\ A_G = (M_p + J_m^T M_a J_m)^{-1} m_p G \end{cases}$$

Equation (4) is referred to as the simplified direct dynamic model (SDDM) of ARROW robot, which is unique. However, the inverse dynamic model (IDM), required for model-based control schemes, is not unique due to actuation redundancy. It is usually obtained by using the pseudo-inverse of the matrix H as follows

$$H^+ (\ddot{X} + \Lambda \dot{X} - A_G) = \Gamma \quad (5)$$

where H^+ denotes the pseudo-inverse of H . The inverse dynamic model of ARROW robot, given by (5), is expressed in terms of the Cartesian coordinates of the moving platform and is suitable for Cartesian-space control. To express these dynamics in terms of joint coordinates instead of Cartesian ones, which is suitable for joint-space control, the inverse Jacobian matrix can be used to yield

$$H^+ (J_m^+ \ddot{q} + (\Lambda J_m^+ - J_m^+ J_m J_m^+) \dot{q} - A_G) = \Gamma \quad (6)$$

where J_m^+ is the pseudoinverse of J_m . The dynamics of ARROW robot in equation (6) can be rewritten into the standard joint space form of rigid multibody manipulators as follows:

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = \Gamma \quad (7)$$

being the dynamic matrices $M(q)$, $C(q, \dot{q})$ and $G(q)$ given by

$$\begin{cases} M(q) = H^+ J_m^+ \\ C(q, \dot{q}) = H^+ \Lambda J_m^+ - J_m^+ J_m J_m^+ \\ G(q) = -H^+ A_G \end{cases}$$

For more details about the kinematic and dynamic modeling of ARROW robot, the reader is referred to Shayya (2015).

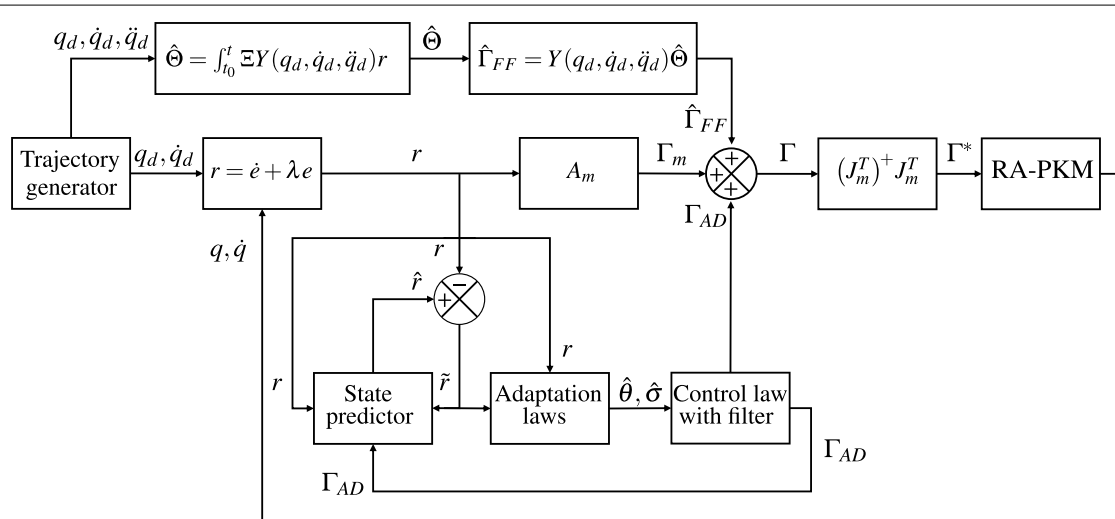


FIGURE 2 | Block diagram of the proposed controller.



FIGURE 3 | View of the experimental setup of ARROW PKM.

3. BACKGROUND ON \mathcal{L}_1 ADAPTIVE CONTROL

Consider the set of desired joint trajectories $q_d \in \mathbb{R}^n$ to be tracked by the active joints (linear actuators in the case of ARROW robot) of the manipulator to be controlled. These trajectories are issued from a trajectory generator in Cartesian space according to the task to be performed. Then the corresponding joint trajectories are computed by solving the inverse kinematics problem (IK), which is trivial in the case of PKMs. For the subsequent control development and to quantify the tracking errors, consider the following position-velocity combined tracking error:

$$r(t) = (\dot{q}(t) - \dot{q}_d(t)) + \lambda (q(t) - q_d(t)) \quad (8)$$

where $\lambda \in \mathbb{R}^+$ is a control design parameter. Let us now introduce the following control input that is the combination of two distinct terms:

$$\Gamma(t) = \Gamma_m(t) + \Gamma_{AD}(t), \quad \Gamma_m(t) \triangleq A_m r(t) \quad (9)$$

where $A_m \in \mathbb{R}^{n \times n}$ is a user-defined Hurwitz matrix that characterizes the desired transient response of the system. In equation (9), the first term [i.e., $A_m r(t)$] is a stabilizing state feedback term while $\Gamma_{AD}(t)$ is an adaptive term and will be explained later on. Substituting the control law (9) in (7) yields the following equation that governs the evolution of the combined tracking error:

$$\dot{r}(t) = A_m r(t) + \Gamma_{AD}(t) - \eta(t, \zeta(t)), \quad r(0) = r_0 \quad (10)$$

where the non-linear function $\eta(t, \zeta(t))$ with $\zeta = [r, q]^T$ gathers all the remaining non-linearities of the system that result from

the application of the control law in (9). These non-linearities may include both known and unknown quantities such as modeling non-linearities, uncertainties, external disturbances, and so on. Under some reasonable assumptions, the non-linear function $\eta(t, \zeta(t))$ can be parameterized using some parameters $\theta(t)$, $\sigma(t) \in \mathbb{R}^n$ and the Λ_∞ norm of the combined tracking error as follows (Hovakimyan and Cao, 2010):

$$\eta(t, \zeta(t)) = \theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t). \quad (11)$$

Substituting equation (11) in equation (10), the combined tracking error becomes

$$\dot{r}(t) = A_m r(t) + \Gamma_{AD}(t) - (\theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t)). \quad (12)$$

In an ideal scenario, where all the non-linearities considered in $\eta(t, \zeta(t))$ are perfectly known (i.e., perfect knowledge of the dynamic model, no immeasurable disturbances, no uncertainties, etc.), replacing $\Gamma_{AD}(t) = \eta(t, \zeta(t)) = (\theta(t) \|r_t\|_{\mathcal{L}_\infty} + \sigma(t))$ in equation (9) should compensate for all the non-linearities resulting in the desired error dynamics specified by A_m as follows:

$$\dot{r}(t) = A_m r(t). \quad (13)$$

However, in practical situations, the dynamics of the robot is either uncertain or unknown. Furthermore, the external disturbances acting on the system are often immeasurable, and hence, could not be accounted for by a specific control term. Consequently, the adaptive control component $\Gamma_{AD}(t)$ should be carefully designed in order to appropriately estimate the non-linear function $\eta(t, \zeta(t))$ in real-time. To that end, consider the following state predictor of the dynamics of the combined tracking error:

$$\begin{aligned} \dot{\hat{r}}(t) &= A_m \hat{r}(t) + \Gamma_{AD}(t) - (\hat{\theta}(t) \|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t)) \\ &\quad - K\tilde{r}(t), \quad \hat{r}(0) = r_0 \end{aligned} \quad (14)$$

where $\hat{r}(t) \in \mathbb{R}^n$ is the prediction of $r(t)$, $\tilde{r}(t) \triangleq \hat{r}(t) - r(t)$ is the prediction error, and $K \in \mathbb{R}^{n \times n}$ is a design matrix introduced to reject high frequency noises (Nguyen et al., 2013). $\hat{\theta}(t)$ and $\hat{\sigma}(t)$ are the estimates of $\theta(t)$ and $\sigma(t)$, respectively, generated according to the following projection-based adaptation rules:

$$\dot{\hat{\theta}}(t) = \Sigma \text{Proj} \left(\hat{\theta}(t), P\tilde{r}(t) \|r_t\|_{\mathcal{L}_\infty} \right), \quad \hat{\theta}(0) = \hat{\theta}_0 \quad (15)$$

$$\dot{\hat{\sigma}}(t) = \Sigma \text{Proj} (\hat{\sigma}(t), P\tilde{r}(t)), \quad \hat{\sigma}(0) = \hat{\sigma}_0 \quad (16)$$

where Σ is a positive adaptation gain and $P = P^T > 0$ is the solution to the algebraic Lyapunov equation $A_m^T P + P A_m = -Q$, for some arbitrary matrix $Q = Q^T > 0$. The projection operator Proj in equations (15) and (16) prohibits the estimated values from exceeding their allowable range specified in the control design (i.e., $\|\hat{\theta}(t)\|_\infty < \theta_b$, $\|\hat{\sigma}(t)\|_\infty < \sigma_b$, $\forall t > 0$, $\theta_b, \sigma_b \in \mathbb{R}^+$). The adaptive control term in (9) is the output of

$$\Gamma_{AD}(s) = C(s)\hat{\eta}(s) \quad (17)$$

where $\hat{\eta}(s)$ is the Laplace transform of $\hat{\eta}(t) = (\hat{\theta}(t) \|r_t\|_{\mathcal{L}_\infty} + \hat{\sigma}(t))$ and $C(s)$ is a diagonal matrix of BIBO-stable¹ strictly proper transfer functions with DC gain (Khalil, 2002). This particular structure, according to \mathcal{L}_1 adaptive control theory, decouples the estimation loop from the control loop that allows for arbitrarily large gains without hurting robustness.

¹ BIBO: Bounded Input Bounded Output.

TABLE 2 | Summary of the control parameters used in real-time experiments for both controllers.

Parameter	Description	Value
λ	Position error weight	725
A_m	Transient response matrix	$-235 \times I_{6 \times 6}$
θ_b	Upper bound on θ	35
σ_b	Upper bound on σ	35
Σ	Adaptation gain	10^6
Ξ	Adap. gain for model-based term	$\text{diag}(1, 1, 1, 0.005)$
K	Noise rejection gain	$5765 \times I_{6 \times 6}$
$C(s)$	\mathcal{L}_1 design filter	$144/(s^2 + 16.8s + 144)$
Q	Arbitrary control design matrix	$I_{6 \times 6}$
P	Solution of Lyapunov algebraic equation	$21 \times 10^{-4} \times I_{6 \times 6}$

4. PROPOSED SOLUTION: AN EXTENDED \mathcal{L}_1 ADAPTIVE CONTROL

In order to improve the performance of \mathcal{L}_1 adaptive control, we propose to include the dynamics of the manipulator in the control scheme design. Furthermore, we want the additional dynamic-based term to be able to adapt its parameters to the possible variations in the robot and its environment (e.g., handling a payload with unknown dynamic properties, interaction with the environment, etc.). Though the dynamic parameters of the robot may be known, one should consider the scenario where these values may be varying or inaccurately known. The current section highlights the main contribution of this paper as it describes how the robot's dynamics knowledge is included in the control law and how it is supposed to improve the control performance of the manipulator.

4.1. Linear-in-the-Parameters Formulation of the Dynamics

For model-based adaptive control schemes, a reformulation of the dynamics is required (Craig et al., 1987). The unknown or uncertain parameters to be estimated in real-time have to be separated from the non-linear functions in the dynamic model.

Consider the following dynamic model of a n -DOFs mechanical manipulator in joint space:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = \Gamma(t) \quad (18)$$

where $M(q) \in \mathbb{R}^{n \times n}$ is the symmetric, positive definite inertia matrix, $N(q, \dot{q}) \in \mathbb{R}^n$ gathers Coriolis, centrifugal, and gravity forces, and $\Gamma \in \mathbb{R}^n$ is the input vector. $M(q)$ and $N(q, \dot{q})$ are usually composed of non-linear functions of the joint position and velocity vectors. These non-linear terms, however, appear in a linear form with respect to the robot's geometric and dynamic parameters. Consequently, the dynamics in (18) can be rewritten in the following linear form:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = Y(q, \dot{q}, \ddot{q})\Theta \quad (19)$$

where $Y(\cdot) \in \mathbb{R}^{n \times p}$ is called the regression matrix that is a non-linear function of q, \dot{q} and \ddot{q} , $\Theta \in \mathbb{R}^p$ is the vector of the parameters of the manipulator to be estimated. It is worth to note that the formulation in (19) is not unique, it actually depends on the set

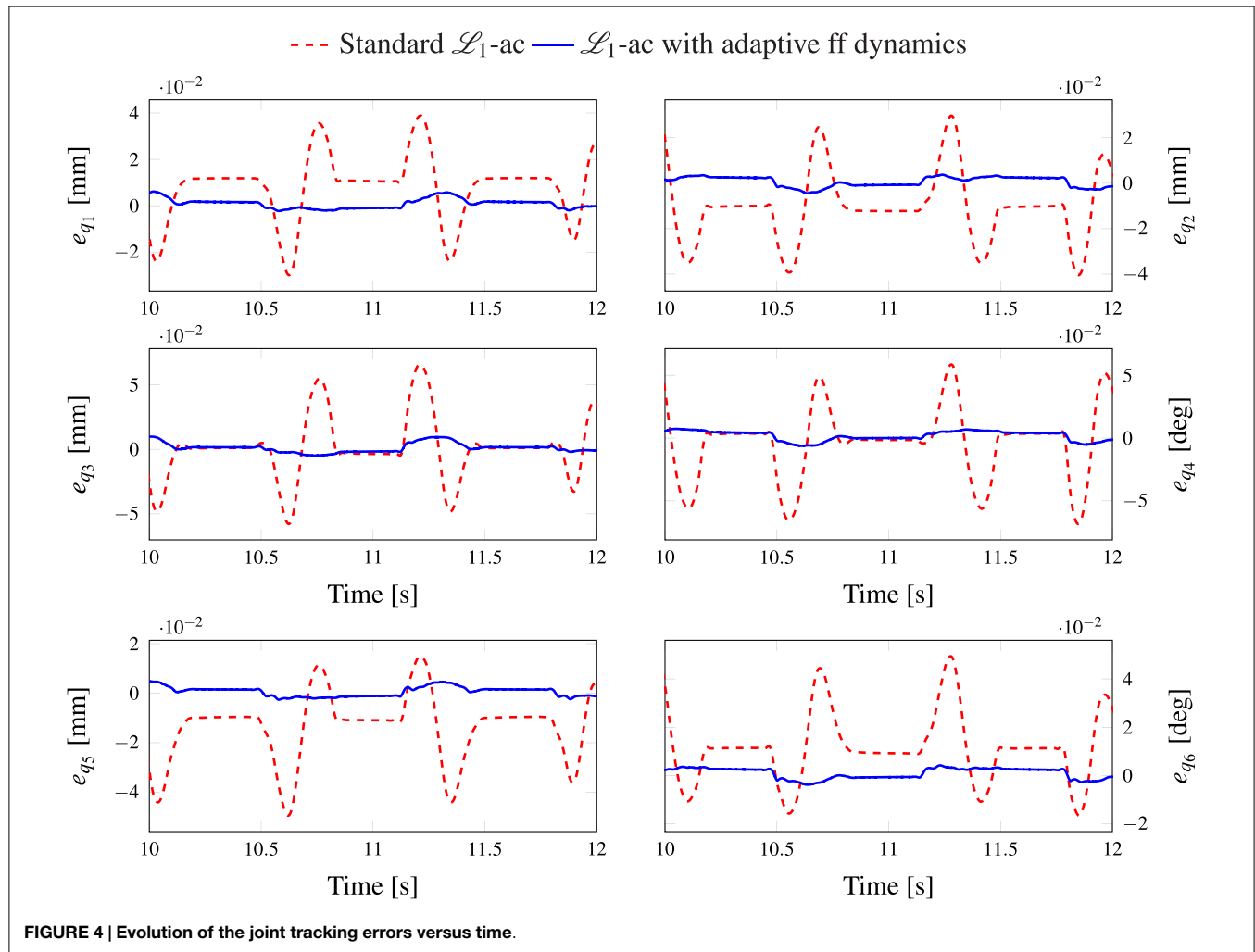


FIGURE 4 | Evolution of the joint tracking errors versus time.

of chosen parameters. Besides, the estimated elements within the parameters vector can either appear in a separate form or as a combination of different parameters. Examples of the dynamics reformulation for some manipulators can be found in Craig et al. (1987) and Sadegh and Horowitz (1990).

4.2. Proposed Control Solution: Augmented \mathcal{L}_1 Adaptive Control with Adaptive Feedforward

The \mathcal{L}_1 adaptive control law described in section 3 consists into two separate terms. The first one is a stabilizing state feedback term, while the second one is the adaptive term in charge of compensating the system uncertainties to achieve the desired performance. Thus, the adaptive term in \mathcal{L}_1 adaptive control is responsible for compensating both structured (i.e., modeled) and unstructured (i.e., not included in the model such as friction) uncertainties. Nevertheless, since we have in disposal a dynamic model of our manipulator, it can be used to compensate modeling uncertainties through a more sophisticated model-based term that takes into consideration the structure of the dynamics.

The proposed idea consists then in adding an adaptive feedforward term to the control law (9). Hence, the proposed control

action will consist of three distinct terms where each term has a specific role in the controlled closed-loop system. The new proposed control law is then expressed as follows:

$$\Gamma(t) = \Gamma_m(t) + \Gamma_{AD}(t) + \hat{\Gamma}_{FF}(t) \quad (20)$$

where $\hat{\Gamma}_{FF} \in \mathbb{R}^n$ is the model-based adaptive feedforward. It is obtained by replacing q , \dot{q} , and \ddot{q} in the inverse dynamic model by their desired values q_d , \dot{q}_d , and \ddot{q}_d , respectively. Besides, the dynamic matrices $M(\cdot)$ and $N(\cdot)$ are replaced by their estimates $\hat{M}(\cdot)$ and $\hat{N}(\cdot)$ to be adjusted in real-time. The adaptive feedforward term can then be expressed as follows:

$$\hat{\Gamma}_{FF}(t) = Y(q_d, \dot{q}_d, \ddot{q}_d) \hat{\Theta}(t) \quad (21)$$

where $\hat{\Theta}(t) \in \mathbb{R}^p$ is the estimate of the parameters vector Θ , its variation is governed by the following adaptation rule

$$\dot{\hat{\Theta}}(t) = -\Xi Y(q_d, \dot{q}_d, \ddot{q}_d)^T r(t) \quad (22)$$

where $\Xi = \text{diag}(\xi_1, \dots, \xi_p)$ is the adaptation gain matrix.

The proposed control law in (20) is expected to improve the tracking performance compared to original \mathcal{L}_1 adaptive control. In fact, thanks to the addition of the model-based adaptive

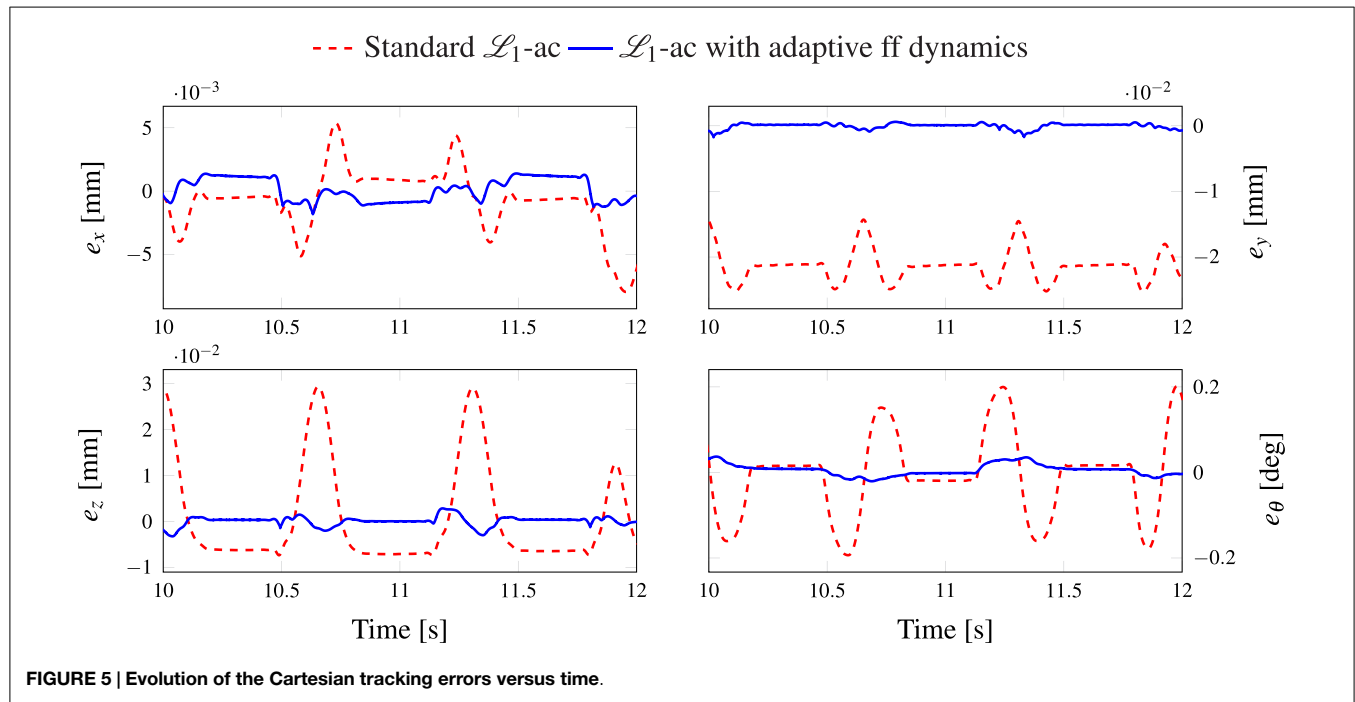


FIGURE 5 | Evolution of the Cartesian tracking errors versus time.

feedforward term, the structured uncertainties will be accounted for by this added term. The adaptive control component Γ_{AD} of conventional \mathcal{L}_1 adaptive control will be in charge of compensating the non-structured uncertainties such as frictions, possible disturbances, and any residual non-linearities.

The proposed controller in (20) can be applied for both serial and fully actuated² parallel manipulators without any modification. However, for the particular case of redundantly actuated parallel manipulators, a slight modification is required in order to deal with the actuation redundancy issues as it is explained in the following.

4.3. Application to PKMs with Actuation Redundancy

Redundantly actuated parallel manipulators have the peculiarity of the existence of internal forces (antagonistic to each other) that may deteriorate the control performance and cause energy losses. These internal efforts are mainly caused by decentralized control, calibration errors, and encoders' resolution (Andreas and Hufnagel, 2011). These internal forces, which do not generate any motion, may be used for secondary tasks such as backlash avoidance and stiffness modulation. However, if no secondary task is scheduled within the control strategy, one should consider reducing their effects in order to avoid vibrations that might deteriorate the control performance.

As it has been explained in Andreas and Hufnagel (2011), the internal forces are caused by the control inputs that are in the null space of the inverse Jacobian matrix J_m . Consequently, one solution to reduce their effects can be to project the generated

TABLE 3 | Tracking performance comparison in terms of RMS errors.

	RMS_J (mm)	RMS_T (mm)	RMS_R (deg)
\mathcal{L}_1 adaptive control	0.0496	0.0343	4.7×10^{-3}
Extended \mathcal{L}_1 adaptive control	0.0102	0.0074	1.2×10^{-3}
Improvement	79.43%	78.59%	74.1%

control inputs into the range space of J_m using the following projector:

$$R_{J_m^T} = \left(J_m^T \right)^+ J_m^T. \quad (23)$$

Hence, the actual control inputs that will be applied on the robot are the “regularized” ones obtained through the following projection

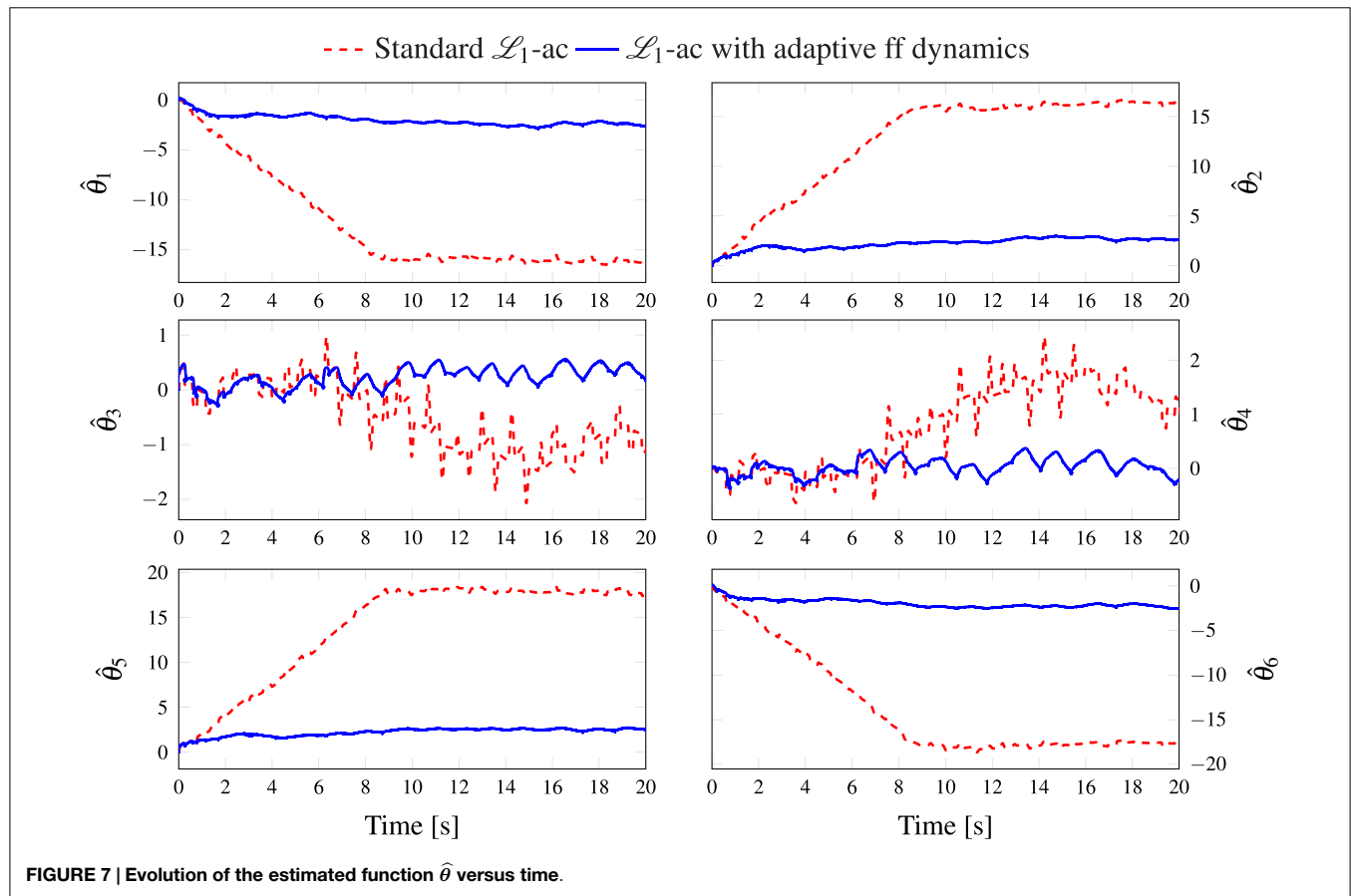
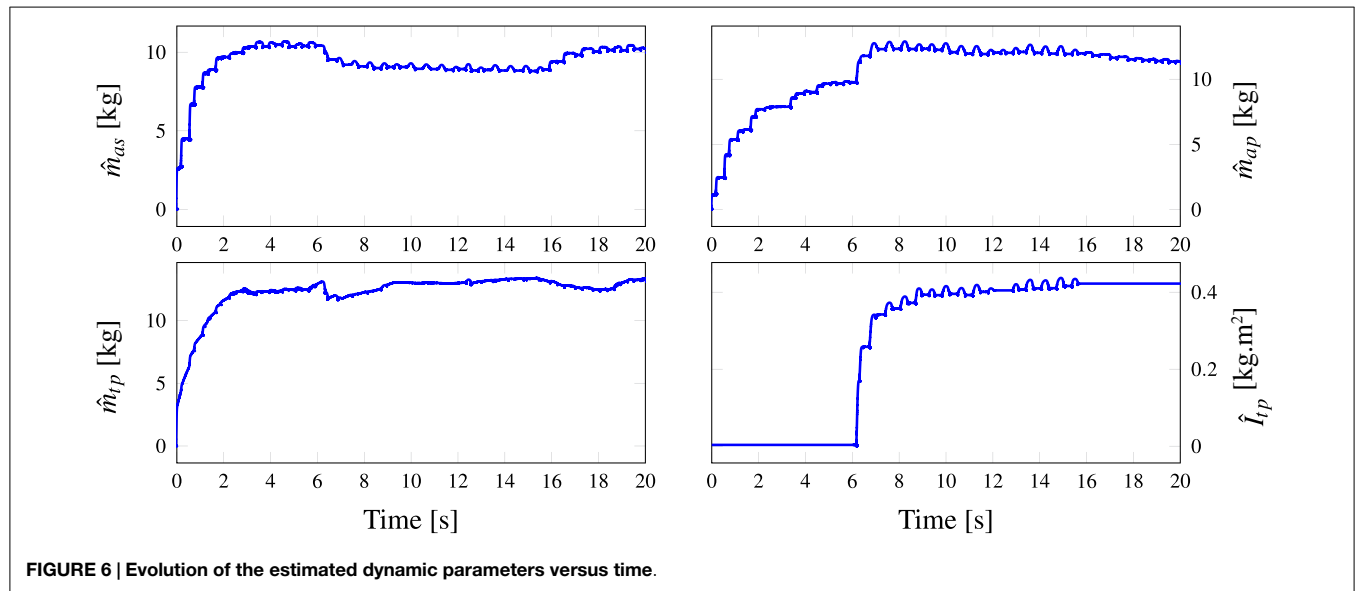
$$\Gamma^*(t) = R_{J_m^T} \Gamma(t). \quad (24)$$

The overall control strategy is summarized in the bloc diagram illustrated in Figure 2. From an academic point of view, it would be interesting to estimate all dynamic and geometric parameters in the adaptation process. However, in practical situations, the parameters that are less known or more likely to vary are the dynamic ones (i.e., the masses and inertias). Consequently, in this work, we choose the following parameters vector for ARROW robot: $\Theta = [m_{as}, m_{ap}, m_p, I_p]^T$, to be estimated online. These parameters are related to the masses and inertia of the different parts of the manipulator.

5. EXPERIMENTAL VALIDATION OF THE PROPOSED CONTROLLER

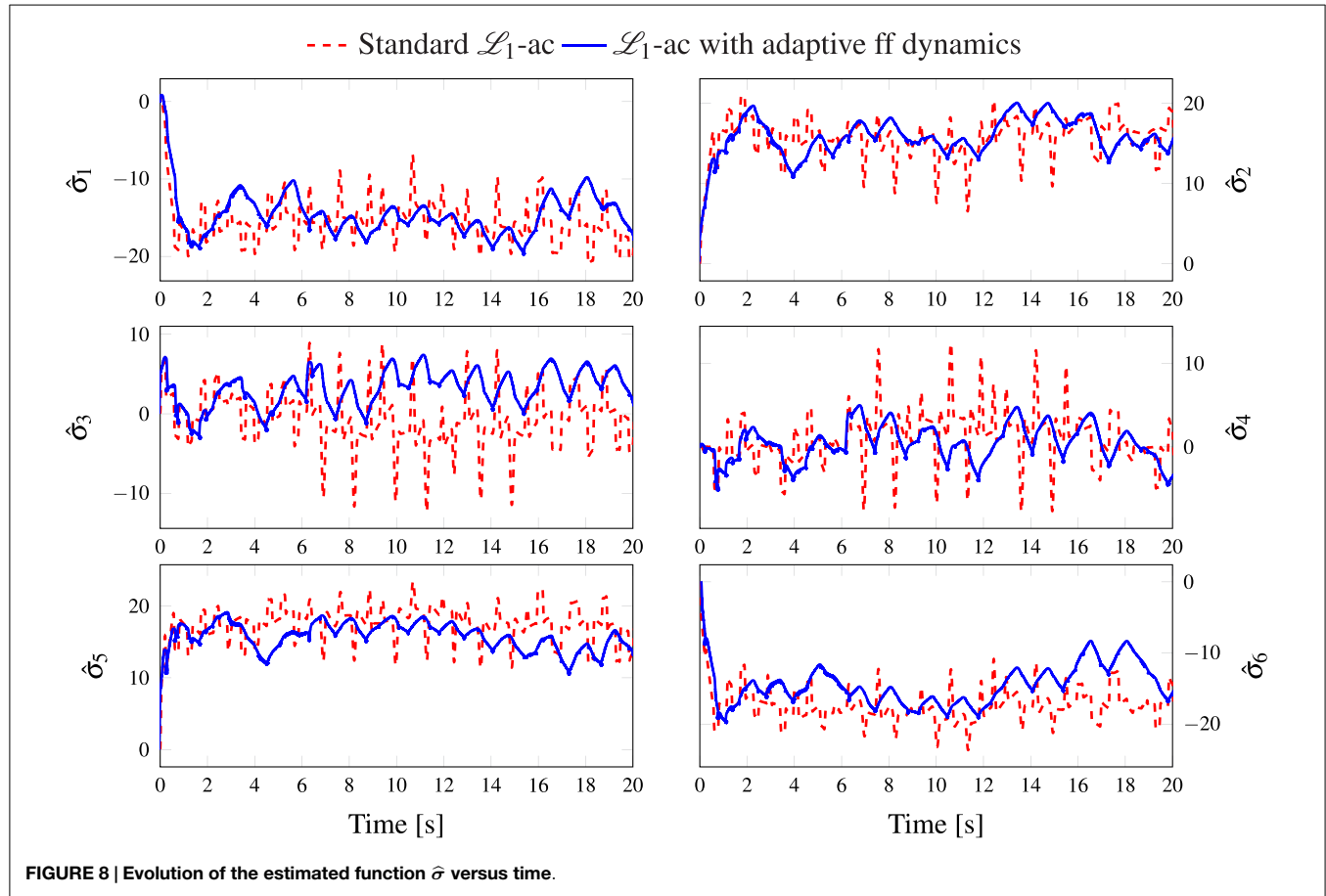
To show the relevance of the proposed controller, real-time experiments have been conducted on ARROW robot. The experimental

²Parallel manipulators for which the number of actuators is equal to the number of degrees of freedom.



setup is illustrated in **Figure 3** and the control scheme is developed using Simulink software from Mathworks Inc. The controller is compiled using the Real-Time Workshop toolbox before being loaded on the target PC. The latter is an industrial computer running xPC Target (also from Mathworks Inc.) in real-time with a sampling frequency of 5 kHz.

The moving platform of the robot has to perform several point-to-point displacements and rotations inside the workspace. The estimated parameters vector is initialized to $\hat{\Theta}(0) = [0, 0, 0, 0]^T$. The controller parameters of both controllers (the original \mathcal{L}_1 adaptive controller and the augmented one) are summarized in **Table 2**.



A comparison between the \mathcal{L}_1 adaptive controller and the proposed augmented one in terms of joint tracking errors is illustrated in **Figure 4**. For clarity of the results, the plots were zoomed to the interval (Horowitz and Tomizuka, 1986; Khalil, 2002) seconds. It can be seen that the proposed model-based augmented controller clearly provides much better tracking than the original controller. Thanks to the model-based adaptive term, the controller successfully compensates the modeled non-linear dynamics with initially unknown parameters. It can also be seen that there exists an offset in the joint tracking errors in the case of the standard controller corresponding to the simple arms (i.e., q_1 , q_2 , q_5 , and q_6). This is mainly due to the better handling of gravity effects in the proposed controller compared with the standard one. This result is further highlighted in **Figure 5** where the Cartesian tracking errors are plotted. The plots are zoomed to the interval (Horowitz and Tomizuka, 1986; Khalil, 2002) seconds for clarity. An offset of the tracking errors corresponding to the y -axis (vertical axis) in the standard \mathcal{L}_1 adaptive controller, due to the previously mentioned reason. Since the robot is not equipped with external sensors, the Cartesian tracking errors $e_c = [e_x, e_y, e_z, e_\theta]^T \in \mathbb{R}^4$ were estimated using the following equation (Sartori Natal et al., 2015):

$$e_c(t) = J_m^+ e_q(t) \quad (25)$$

where $e_q(t) \triangleq q_d(t) - q(t)$ is the joint tracking error vector. In order to quantify the enhancement brought by the proposed

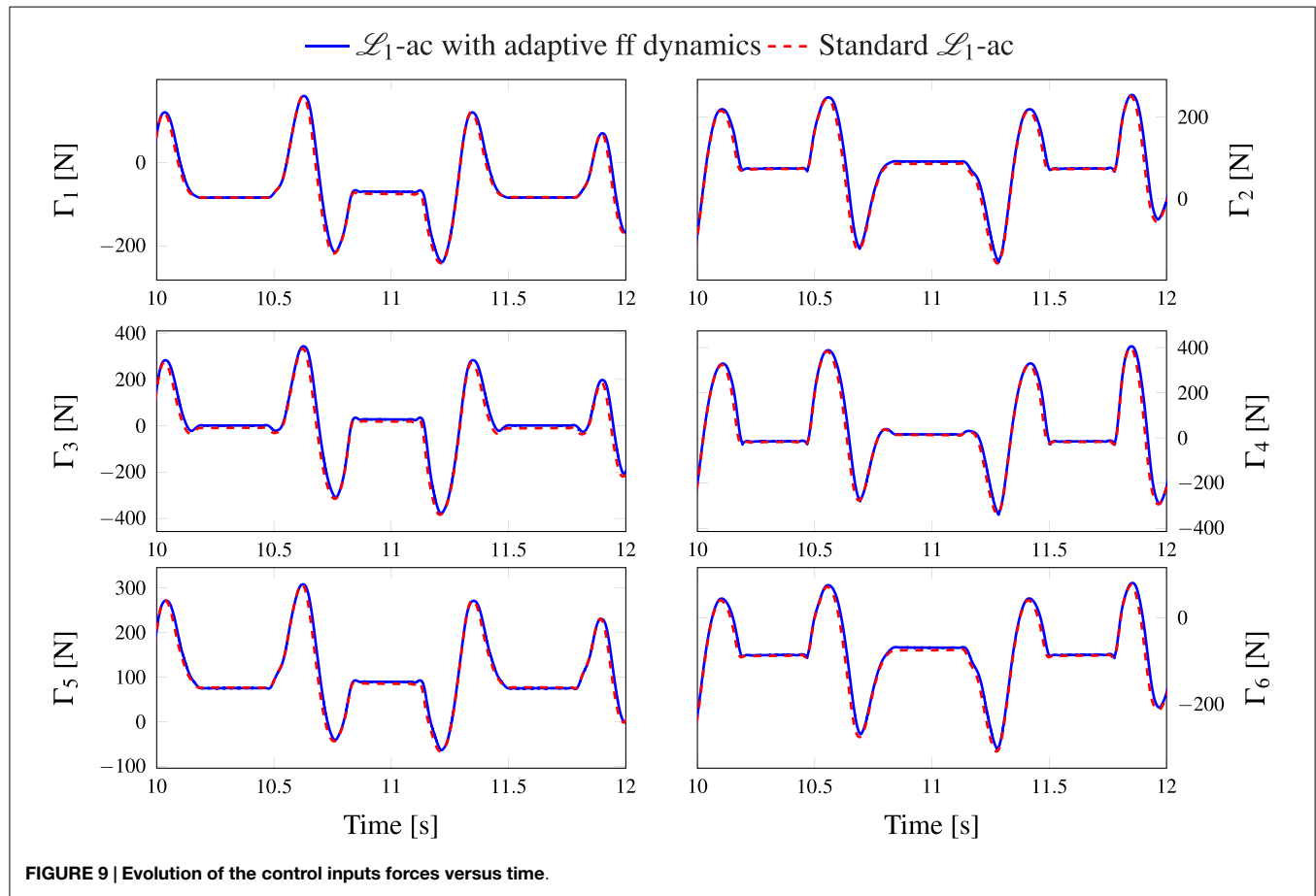
controller, the tracking performance is evaluated based on the root mean square of the tracking errors using the following criteria:

$$\text{RMS}_J = \sqrt{\sum_{i=1}^6 \frac{1}{N} \sum_{j=1}^N e_{q_i}(j)^2} \quad (26)$$

$$\text{RMS}_T = \sqrt{\frac{1}{N} \sum_{j=1}^N e_x(j)^2 + \frac{1}{N} \sum_{j=1}^N e_y(j)^2 + \frac{1}{N} \sum_{j=1}^N e_z(j)^2} \quad (27)$$

$$\text{RMS}_R = \sqrt{\frac{1}{N} \sum_{j=1}^N e_\theta(j)^2} \quad (28)$$

where e_{q_i} is the tracking error of the i^{th} joint, e_x , e_y , and e_z are the Cartesian tracking errors of the x , y , and z -axis, respectively, and e_θ is the tracking error of the rotation of the moving platform. The obtained results are summarized in **Table 3**, where the improvements brought by the proposed controller are clearly highlighted. Indeed, the joint tracking errors were reduced by 79.4% and the Cartesian ones by up to 78.6%. The improvements of the tracking errors are achieved thanks to the better compensation of the nonlinearities of the model. This means that the estimated model parameters do converge closer to their real values, otherwise the tracking performance would have been worst. Indeed, as it can be seen in **Figure 6**, the estimated parameters, initialized to 0, converge to their steady-state reported in **Table 1**. It is noticed that



the rotational inertia of the platform is not adjusted immediately which is due to the pure translational motion at the beginning (i.e., $t \in [0, 6]$ s).

Furthermore, since the model-based non-linearities are appropriately compensated, the estimated parameters forming the adaptive term $\Gamma_{AD}(t)$ would be smaller in the case of the proposed controller. Indeed, these interesting results can clearly be observed in **Figures 7 and 8**, where we can see smaller amplitudes for the estimated parameters in the case of the proposed controller compared to the original one. This means that the structured uncertainties were better compensated by the proposed additional term and, hence, the \mathcal{L}_1 adaptive term should account for reduced non-linearities. Finally, the control input forces generated by both controllers are shown in **Figure 9**. Results can be viewed in the video available at the following URL: <https://youtu.be/rCAocIWMfPs>

6. CONCLUSION AND FUTURE WORK

In this work, a new adaptive control scheme for mechanical manipulators based on \mathcal{L}_1 adaptive control theory is proposed. The original \mathcal{L}_1 adaptive controller is known for being model-free and for its robustness and fast adaptation qualities. Since in our case, a dynamic model of our redundantly actuated parallel manipulator is available, we have proposed to extend the original \mathcal{L}_1 adaptive controller by including dynamics knowledge

in the control design. The motivation behind this proposition is to improve the tracking capabilities of the manipulator. Since the dynamics of the manipulator may be unknown or likely to vary during work conditions, we propose to make the additional model-based term adaptively adjust its parameters. Moreover, to deal with actuation redundancy, a projection of the control inputs was proposed in order to reduce the internal efforts that may have a negative effect on the control performance. Real-time experiments, conducted on a 4-DOF redundantly actuated parallel manipulator, demonstrate our claims. Indeed, a significant enhancement of the tracking performance was observed in both Cartesian and joint spaces. This work can be further extended by investigating other adaptive strategies for the model-based term and considering a real application where the robot is performing an industrial task. Another interesting scenario would be to consider an abrupt change in the platform mass due to payload handling for instance. Moreover, the tuning of the filter of the \mathcal{L}_1 adaptive controller remains an open problem that should be addressed in the future.

ACKNOWLEDGMENTS

This research was supported by the French National Research Agency (ANR) in the framework of the ARROW project (ANR-2011-BS3-006-01-ARROW).

REFERENCES

- Andreas, M., and Hufnagel, T. (2011). "A projection method for the elimination of contradicting control forces in redundantly actuated PKM," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'11)* (Shanghai: IEEE), 3218–3223.
- Bennehar, M., Chemori, A., and Pierrot, F. (2014a). "A new extension of direct compensation adaptive control and its real-time application to redundantly actuated PKMs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)* (Chicago, IL), 1670–1675.
- Bennehar, M., Chemori, A., and Pierrot, F. (2014b). "A novel rise-based adaptive feedforward controller for redundantly actuated parallel manipulators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'14)* (Chicago, IL), 2389–2394.
- Bennehar, M., Chemori, A., and Pierrot, F. (2015). "L1 adaptive control of parallel kinematic manipulators: design and real-time experiments," in *IEEE International Conference on Robotics and Automation (ICRA'15)* (Seattle, WA), 1587–1592.
- Cao, C., and Hovakimyan, N. (2006a). "Design and analysis of a novel \mathcal{L}_1 adaptive controller, part i: control signal and asymptotic stability," in *American Control Conference (ACC'06)* (Minneapolis, MN), 3397–3402.
- Cao, C., and Hovakimyan, N. (2006b). "Design and analysis of a novel \mathcal{L}_1 adaptive controller, part ii: guaranteed transient performance," in *American Control Conference (ACC'06)* (Minneapolis, MN), 3403–3408.
- Corbel, D., Gouttefarde, M., and Company, O. (2010). "Towards 100g with PKM is actuation redundancy a good solution for pick and place?," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'10)* (Anchorage, AK), 4675–4682.
- Craig, J. J., Hsu, P., and Sastry, S. S. (1987). Adaptive control of mechanical manipulators. *Int. J. Robot. Res.* 6, 16–28. doi:10.1177/027836498700600202
- Dubowsky, S., and DesForges, D. T. (1979). The application of model-referenced adaptive control to robotic manipulators. *J. Dyn. Syst. Meas. Control* 101, 193–200. doi:10.1115/1.3426424
- Horowitz, R., and Tomizuka, M. (1986). An adaptive control scheme for mechanical manipulators – compensation of nonlinearity and decoupling control. *J. Dyn. Syst. Meas. Control* 108, 127–135. doi:10.1115/1.3143754
- Hovakimyan, N., and Cao, C. (2010). *L1 Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Philadelphia, PA: SIAM.
- Khalil, H. (2002). *Nonlinear Systems*, Vol. 3. Upper Saddle River, NJ: Prentice Hall.
- Markiewicz, B. (1973). *Analysis of the Computed Torque Drive Method and Comparison with Conventional Position Servo for a Computer-Controlled Manipulator*. Pasadena, CA: Jet Propulsion Laboratory, California Institute of Technology.
- Merlet, J. P. (2006). *Parallel Robots, Second Edition*. Dordrecht: Springer.
- Nguyen, K. D., Dankowicz, H., and Hovakimyan, N. (2013). "Marginal stability in \mathcal{L}_1 adaptive control of manipulators," in *Proceedings of the 9th ASME International Conference on Multibody Systems, Nonlinear Dynamics, and Control*, Portland, OR.
- Ortega, R., and Spong, M. W. (1989). Adaptive motion control of rigid robots: a tutorial. *Automatica* 25, 877–888. doi:10.1016/0005-1098(89)90054-X
- Sadegh, N., and Horowitz, R. (1990). Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *Int. J. Robot. Res.* 9, 74–92. doi:10.1177/027836499000900305
- Sartori Natal, G., Chemori, A., and Pierrot, F. (2015). Dual-space control of extremely fast parallel manipulators: payload changes and the 100g experiment. *IEEE Trans. Control Syst. Technol.* 23, 1520–1535. doi:10.1109/TCST.2014.2377951
- Shang, W., and Cong, S. (2010). Nonlinear adaptive task space control for a 2-dof redundantly actuated parallel manipulator. *Nonlinear Dyn.* 59, 61–72. doi:10.1007/s11071-009-9520-1
- Shayya, S. (2015). *Towards Rapid and Precise Parallel Kinematic Machines*. Ph.D. thesis, University of Montpellier, Montpellier.
- Shayya, S., Krut, S., Company, O., Baradat, C., and Pierrot, F. (2014). "A novel (3t-2r) parallel mechanism with large operational workspace and rotational capability," in *IEEE International Conference on Robotics and Automation (ICRA'14)* (Hong Kong: IEEE), 5712–5719.
- Slotine, J., and Weiping, L. (1987). On the adaptive control of robot manipulators. *Int. J. Robot. Res.* 6, 49–59. doi:10.1177/027836498700600303

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The Review Editor Francesco Pierri and the Associate Editor Filippo Arrichiello declared a current collaboration and the Associate Editor states that the process nevertheless met the standards of a fair and objective review.

Copyright © 2015 Bennehar, Chemori, Pierrot and Creuze. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.