



**HAL**  
open science

## From Non-model-Based to Model-Based Control of PKMs: A Comparative Study

Hussein Saied, Ahmed Chemori, Maher El Rafei, Clovis Francis, François Pierrot

► **To cite this version:**

Hussein Saied, Ahmed Chemori, Maher El Rafei, Clovis Francis, François Pierrot. From Non-model-Based to Model-Based Control of PKMs: A Comparative Study. Mechanism, Machine, Robotics and Mechatronics Sciences, , pp.153-169, 2019, 978-3-319-89910-7. 10.1007/978-3-319-89911-4\_12 . lirmm-02411641

**HAL Id: lirmm-02411641**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-02411641>**

Submitted on 15 Dec 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# From Non-model-Based to Model-Based Control of PKMs: A Comparative Study



H. Saied, A. Chemori, M. El Rafei, C. Francis and F. Pierrot

**Abstract** This paper deals with control of parallel robots, where different controllers are proposed and compared. It demonstrates the strength of model-based controllers over the non-model-based ones when dealing with parallel kinematic manipulators known with their high nonlinearity, time-varying parameters and uncertainties. More precisely, adaptive model-based algorithms are the preferred control solutions for such kind of manipulators, thanks to their adjustable-parameters feature which is more adequate to the varying and non-accurate nature of parallel kinematic manipulators. These facts are fulfilled here by numerical simulations and real-time experiments on a four-degree-of-freedom parallel robot named VELOCE.

**Keywords** PKM · Dynamic model · Model-based · Non-model-based Adaptive control · Feedforward · PID · Nonlinear PD · Augmented PD

---

H. Saied (✉) · A. Chemori · F. Pierrot  
Laboratoire d'Informatique, de Robotique, et de Microélectronique de Montpellier,  
University of Montpellier, Montpellier, France  
e-mail: hussein.saied@lirmm.fr

A. Chemori  
e-mail: ahmed.chemori@lirmm.fr

F. Pierrot  
e-mail: francois.pierrot@lirmm.fr

H. Saied · M. El Rafei · C. Francis  
Scientific Research Center in Engineering, Lebanese University, Beirut, Lebanon  
e-mail: maher.elrafei@ul.edu.lb

C. Francis  
e-mail: cfrancis@ul.edu.lb

## 1 Introduction

Parallel kinematic manipulators (PKMs) are defined in [1] as follows: “A generalized parallel manipulator is a closed-loop kinematic chain mechanism whose end-effector is linked to the base by several independent kinematic chains”.

PKMs were extensively used in robotized industries in the last few decades since it surpasses their counterpart’s serial structures, particularly, in terms of high rigidity, better tracking performance, good precision, high payload-to-weight ratio and great dynamic [2, 3]. A very wide range of applications take benefit of PKMs. *Stewart* proposed in 1965 a platform that is used as a flight simulator [4]. Delta robot prototype of 3 DOFs proposed initially in 1985 [5] is the leader in pick-and-place operations [1], used in packaging industry, laser cutting [6], medical applications [7] and haptic devices in which they allow the human-computer interaction [8]. Another parallel structures are used in machining tasks [9].

However, some problems associated with such kind of structures still open and are not solved satisfactory. The drawbacks of PKMs are listed as limited range of motion especially the rotational motion [10], small work space, low dexterity, complex forward kinematic solutions [11]. Singularities’ behavior is more complicated than of serial [12], it can occur both inside and on the border of the work space [11].

In the literature, a wide range of control schemes have been proposed aiming to drive PKMs in accurate mode and high precision. The proposed control strategies can be classified in two classes, Model-Based and Non-Model-Based. The non-model-based strategies do not need a priori knowledge about the dynamics of the manipulator except the states (position and velocity). The Proportional-Integral-Derivative (PID) controller [13] is the most used in industrial applications mainly due to its simplicity and easy implementation as well as its acceptable control performance. However, PKMs known by their nonlinear dynamics, and highly increasing non-linearity at high speeds which may even lead to instability. The need for nonlinear controllers arises, knowing that PID lacks to robustness. Nonlinear PD (NPD) controller [14] is more adequate to the nature of PKMs, which can insure stability and disturbances rejection and performs with better robustness towards error variation. Successful application of non-model-based fuzzy controller applied on Stewart platform in [15], shows that this controller can drive the six-degree motion platform accurately, smoothly and in a stable way. On the other hand, researchers developed several model-based controllers depending on the fact that the closed-loop algorithms, rich enough with knowledge about the system dynamics, can compensate their nonlinearities. PD with gravity compensation or with desired gravity compensation were applied intending to achieve better performance than simple PD since it surpasses the effect of gravity [16, 17]. Computed torque (CT) control exploits the full knowledge about the nonlinear system dynamics, leading to a linear closed-loop system in terms of tracking error [18]. Also the Augmented PD (APD) is a model-based strategy, where the dynamic part of the controller is computed from both the desired and measured states improving the global performance of the control mission [19]. Nevertheless, PKMs are featured with time-varying parameters (e.g. payload mass), uncertainties and difficulty to get

accurate model values, then the design of adaptive controllers is very significant. Adaptive model-based controllers recompense the possible variation of parameters and react against the disturbances by dynamical calibration in an online algorithm, such as the adaptive feedforward PD controller (AFFPD) [20].

The control performances of some classical non-model-based controllers, as PD, PID, NPD, and model-based controllers, as APD, AFFPD, are studied and compared in this paper. The main objective is to show that a controller fed with a good dynamic knowledge about the robot will be very powerful and more precise. Moreover, the time-varying PKM environment requires adaptive dynamic knowledge to manage robustness and accuracy, as it was proved with real-time experimental tests.

The paper organization is as follow: Sect. 2 describes the structure of VELOCE parallel robot, as well as its kinematic and dynamic modeling. Section 3 is dedicated to the synthesis of the proposed control solutions. Simulation and experimental results are presented and discussed in Sect. 4. Section 5 concludes the paper and states the future work.

## 2 Description and Modeling of VELOCE PKM

In this section, a full description of the whole mechanical structure of VELOCE PKM is presented, then a brief explanation of its kinematic and dynamic models is introduced.

### 2.1 Structure of VELOCE PKM

VELOCE robot (see Fig. 1) is a 4-DOF parallel manipulator designed and fabricated in LIRMM (Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier). It is mainly designed for pick-and-place applications. It consists of four kinematic chains and four degrees-of-freedom, three independent translational degrees in the three dimensions and one rotational degree around the vertical axis. It is note that VELOCE is a fully parallel manipulator [1]. Each kinematic chain is composed, in a serial manner, of an actuator, a rear arm fixed to the actuator's rotor, a forearm including two links forming a parallelogram and connected through ball joints to the rear arm and to the traveling plate (see Fig. 2). The traveling plate is made of two essential parts, upper and lower. Both parts are mounted on a single screw, and the movement of one part with respect to the other generates the rotational action.

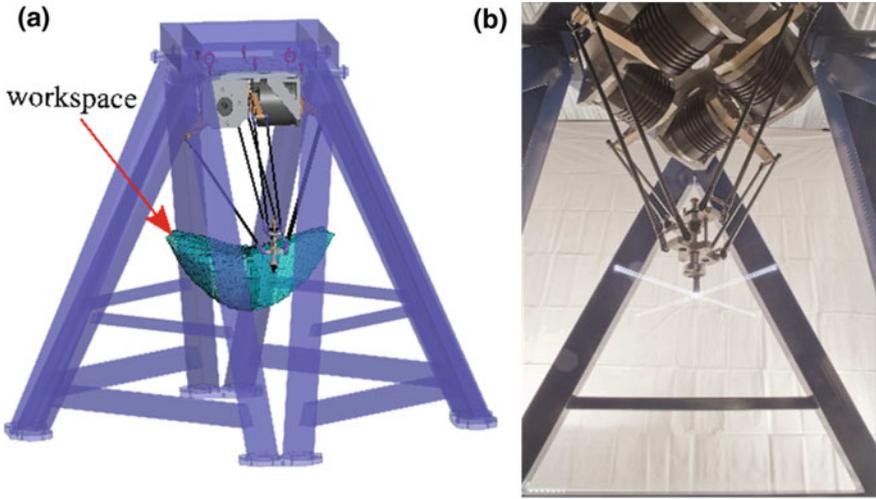


Fig. 1 VELOCE PKM. a: CAD view, b: The manufactured robot

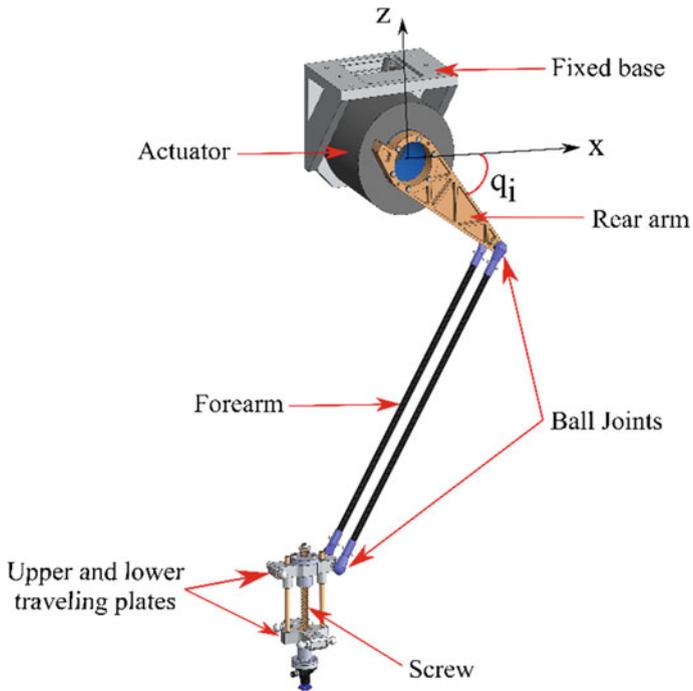


Fig. 2 An  $i$ th kinematic chain of VELOCE PKM

## 2.2 Kinematic Modeling of VELOCE PKM

The Cartesian coordinates of the traveling plate can be presented with respect to the fixed-base frame in four-dimensional space vector  $\mathbf{x} = [x, y, z, \alpha]^T$  such that  $x, y, z$  are the translational coordinates and  $\alpha$  is the rotational angle around z-axis. The orientation and position of the traveling plate are specified by the angular positions of the four actuators, since VELOCE is a fully PKM, represented in another four-dimensional space vector  $\mathbf{q} = [q_1, q_2, q_3, q_4]^T$ . The relation between  $\mathbf{q}$  and  $\mathbf{x}$  is obtained by a geometrical study for the constraints of the closed-loop formed of kinematic chains and traveling plate. The study leads to the following kinematic models, “Forward kinematic (*Fk*)” and “Inverse kinematic (*Ik*)” respectively:  $\mathbf{x} = Fk(\mathbf{q})$ ;  $\mathbf{q} = Ik(\mathbf{x})$ . Applying the *equiprojectivity principle* explained in [3], the inverse Jacobian matrix can be computed and thus a relation between the joints’ velocities and Cartesian velocity of traveling plate is formulated as follows:

$$\dot{\mathbf{q}} = \mathbf{J}_m \dot{\mathbf{x}} \quad (1)$$

where  $\mathbf{J}_m \in \mathbb{R}^{4 \times 4}$  is the inverse Jacobian Matrix. Note that  $\mathbf{J}_m$  is square and invertible for the fully PKMs (as VELOCE), and if the chosen trajectory is away from singularities. By differentiating Eq. (1) with respect to time, we obtain the relation of accelerations between Cartesian space and joint space as follows:

$$\ddot{\mathbf{q}} = \mathbf{J}_m \ddot{\mathbf{x}} + \dot{\mathbf{J}}_m \dot{\mathbf{x}} \quad (2)$$

## 2.3 Dynamic Modeling of VELOCE PKM

According to [21], the dynamic model can be obtained by analyzing the dynamics in the joint space and in the traveling-plate space separately, then summing up the two equations of motion. Nonetheless, some assumptions are taken to simplify the complexity of the rigid body of such robots. Standing on the light weight of the forearm, its rotational inertia is neglected and its mass is split-up into two parts, one part conjoined to the rear arm and one part to the traveling plate mass. Also the dry and viscous friction in the passive and active joints are ignored, and the effect of gravity can be omitted at high speeds. Regarding the traveling plate, there are three kind of forces acting on it: the gravity forces, the inertial forces and the forces of the load. These forces are transformed into contributions in actuators’ torques using the Jacobian matrices. From the joints side, the gravity of the rear arms with the half-masses of forearms and the arms inertia are also expressed in the actuators’ torques. Then, the total actuators’ torques vector is computed by summing up the contributions of all forces. One can formulate the total inverse dynamic equation to be in the standard joint space form, so we get:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{\Gamma}_{F_{load}} = \mathbf{\Gamma} \quad (3)$$

with  $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{4 \times 4}$  being the inertia matrix,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{4 \times 4}$  is the Coriolis and centrifugal forces matrix,  $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^4$  be the gravitational forces vector and  $\mathbf{\Gamma}_{F_{load}} \in \mathbb{R}^4$  be the payload forces vector. A fundamental property of PKMs is very essential for model-based adaptive controllers consists of linearity of the dynamics with respect to the parameters, such as inertia and masses [22]. So the reformulation of the dynamics in the linear form is expressed as following:

$$\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\mathbf{\Phi} = \mathbf{\Gamma} \quad (4)$$

where  $\mathbf{Y}(\cdot) \in \mathbb{R}^{4 \times n}$  is the regression matrix which is nonlinear function in terms of  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$  and  $\ddot{\mathbf{q}}$ , and  $\mathbf{\Phi} \in \mathbb{R}^n$  is the robot parameters vector to be estimated.

### 3 Proposed Control Solutions

In this section, a design for the proposed control solutions is clarified. The controllers designed are non-model-based controllers: PD, PID and NPD, and model-based controllers: APD and AFFC. Note that the available measurements are directly the joint angles of the actuators, so all controllers are developed in joint space.

#### 3.1 Proportional-Derivative Controller

PD control scheme is composed of two parts, proportional and derivative parts. The general expression of the control input is:

$$\mathbf{\Gamma} = \mathbf{K}_p \mathbf{e} + \mathbf{K}_d \dot{\mathbf{e}} \quad (5)$$

where  $\mathbf{e} = \mathbf{q}_d - \mathbf{q}$  is the joint position error between the desired angles and the actual measured ones.  $\mathbf{K}_p, \mathbf{K}_d \in \mathbb{R}^{4 \times 4}$  are diagonal positive definite matrices which means that no coupling between the joints is considered, and the controller is called a linear single-axis controller [2], knowing that the same gain is used for all joints. The PD control law is asymptotically stable as was addressed in [22]. It is the simplest control law but it has several drawbacks briefed as weak disturbance rejection, no compensation for the nonlinearity and variation nature, and even may leads to instability at high accelerations.

### 3.2 Proportional-Integral-Derivative Controller

It is the same demonstration of the aforementioned PD controller just adding the integral term which is the multiplication of the integral of position error with a positive constant feedback. The control law equation is then:

$$\Gamma = K_p e + K_i \int e dt + K_d \dot{e} \quad (6)$$

where  $K_i \in \mathbb{R}^{4 \times 4}$  is a diagonal positive definite matrix. Same specifications and drawbacks of the PD control, but better global performance related to the tracking error thanks to the contribution of integral term in eliminating the residual errors in the steady state response produced by the proportional term.

### 3.3 Nonlinear Proportional-Derivative Controller

This controller have the same structure of classical PD controller with the time-varying feedback gains instead of being constant. The feedback gains are nonlinear functions in terms of the system states, inputs, and other variables. As For the classical PD, the control law equation can be written as following with the nonlinear gains functions [3]:

$$\Gamma = K_p f(e, \alpha_1, \delta_1) e + K_d f(\dot{e}, \alpha_2, \delta_2) \dot{e} \quad (7)$$

Where

$$f(x, \alpha, \delta) = \begin{cases} |x|^{\alpha-1}, & |x| > \delta \\ \delta^{\alpha-1}, & |x| \leq \delta \end{cases} \quad (8)$$

with  $\alpha_1$  and  $\alpha_2$  can be chosen within the interval  $[0.5, 1]$  and  $[1, 1.5]$  respectively.  $\delta_1$  and  $\delta_2$  are positive constant numbers. From the above nonlinear structure, the feedback gains are adjusted online depending on the value of the error. For small position error, a large gain is produced, and for large position error, a small gain is obtained. On the other hand, large gains for large error rate and small gains for small error rate. This behavior results with rapid transition of the system and favorable damping. NPD is a robust controller against the nonlinearities of PKMs, parametric uncertainties and time delays.

### 3.4 Augmented Proportional Derivative Controller

APD, known also as PD+, is one of the conventional model-based controllers composed of two main parts, feedback part and dynamic model part. The feedback part is a simple PD controller that guarantees the stability and the dynamical part represents the nonlinear dynamics of the system that compensates its effects and enhances the control performance. The control law form of APD looks as follows [19]:

$$\begin{aligned} \Gamma &= M(q)\ddot{q}_d + C(q, \dot{q})\dot{q}_d + G(q) \\ &+ \Gamma_{F_{load}} + K_p e + K_d \dot{e} \end{aligned} \quad (9)$$

As shown in Eq. (9), the dynamical term is computed from the desired and actual trajectories. However, such kind of controllers relying mostly on the dynamics of the robot needs to have an accurate model information, and it cannot compensate the effect of time-varying parameters and uncertainties of PKMs.

### 3.5 Adaptive Feedforward with PD Controller

The AFFPD controller is quietly similar to the APD in the general form, meaning that it is divided into two parts, one part a simple PD feedback to conserve the stability and the other part is the adaptive feedforward dynamics of the PKM to reduce the influence of variation in parameters and uncertainties. Thanks to the property of linearizing the dynamic model, the adaptive term is the multiplication of the regression matrix with the estimated vector of parameters [20]. The control law equation is as follows:

$$\Gamma = Y(q_d, \dot{q}_d, \ddot{q}_d) \hat{\Phi} + K_p e + K_d \dot{e} \quad (10)$$

All the parameters ( $\hat{\Phi} \in \mathbb{R}^6$ ) need to be estimated and adapted depending on the error. The controller relies on the desired trajectories instead of the measured ones which can improve the efficiency. The estimating algorithm is in function of the measured error as follows:

$$\dot{\hat{\Phi}} = KY(\cdot)^T \tau_{FB} \quad (11)$$

where  $K \in \mathbb{R}^{6 \times 6}$  is a positive definite matrix that need to be chosen for a good estimation and tracking error stability.  $\tau_{FB}$  is the torque computed from the feedback part. After linearizing the inverse dynamic Eq. (3) considering all the PKM parameters need to be estimated, the following vector of parameters is obtained:  $\Phi = [MTP \ MUTP \ Ms \ I_a \ m_a r_a \ m_{load}]^T$  such that  $MTP$  is the total mass of the traveling plate including the contribution of the forearms,  $MUTP$  is the mass of the

upper traveling plate including the contribution of the forearms,  $M_s$  is mass of upper traveling plate with the equivalent mass to rotate the screw. According to [23], the used adaptive control scheme achieve a global asymptotic stability respecting the necessary and sufficient conditions for adaptive control [24], in which the reference trajectory should be chosen rich enough with frequencies to converge the parameters estimation's error to zero, with a suitable initial values of the parameters.

## 4 Numerical Simulations and Experiments

VELOCE has four direct-drive motors TMB0140-100-3RBS ETEL, they can provide maximum torque of 127 Nm and reach up a speed of 550 rpm. All actuators are supplied with non-contact incremental optical encoders of 5000 pulses per revolution. The global structure can hold as maximum payload of 10 kg, achieve a peak velocity of 10 m/s and peak acceleration of 200 m/s<sup>2</sup>.

### 4.1 Simulation Results

Simulations were done in Matlab/Simulink environment implementing the controllers in discrete-time schemes similar to real robots control. A fixed-step solver was chosen of sample-time equal to 0.1 ms. The chosen desired trajectory is a sequence of point-to-point motions with a duration of each motion  $T=0.5$  s. A nominal scenario of motion is used to compare the performances of the controllers such that no payload is considered in the simulations. The evaluation criteria proposed to monitor the performances is the computation of the Root Mean Square Error over the Translational (RMSET) and Rotational (RMSE<sub>R</sub>) degrees-of-freedom as follow:

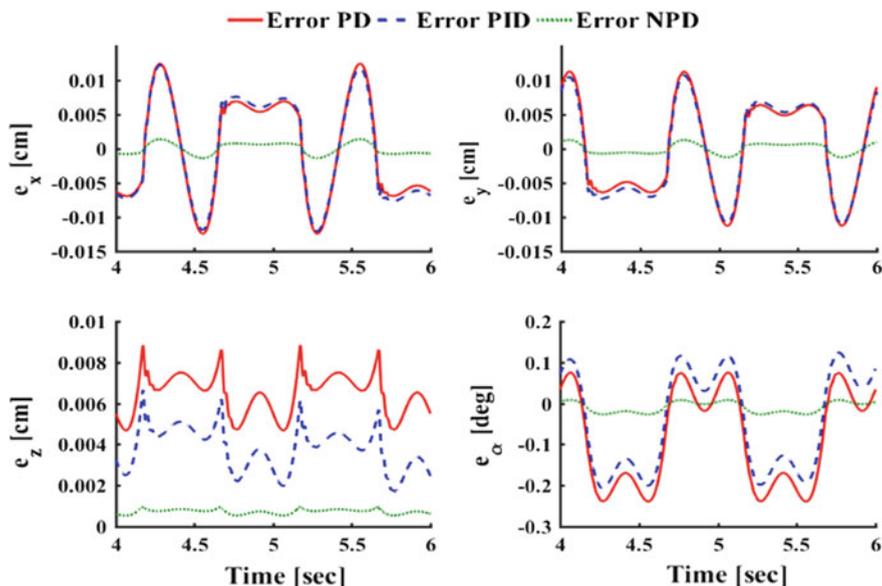
$$RMSET = \left( \frac{1}{N} \sum_{i=1}^N (e_x^2(i) + e_y^2(i) + e_z^2(i)) \right)^{\frac{1}{2}} \quad (12)$$

$$RMSE_R = \left( \frac{1}{N} \sum_{i=1}^N (e_\alpha^2(i)) \right)^{\frac{1}{2}} \quad (13)$$

where  $N$  is the number of the time-samples,  $e_x$ ,  $e_y$ ,  $e_z$  represent the tracking errors along the axes x, y and z,  $e_\alpha$  denotes the tracking error along denotes the tracking error along the rotational angle. The gains for each controller in these simulations are specified by the trial and error technique and shown in Table 1. The comparison between the three non-model-based controllers, in Fig. 3, shows that a NPD performs better than the linear controllers (PD, PID). Thanks to its adjustable gains with the error state, as discussed before, that grants it more robustness and rejection for nonlinearity. For clarity, a zoom in from 4 to 6 s is done in the plot of the Cartesian error in Fig. 3, and the control input signals for the three controllers are depicted

**Table 1** Control design gains in numerical simulation

PD/APD	PID	NPD	AFFPD
$k_p = 4000$ $k_d = 6$	$k_p = 4000$ $k_d = 6$ $k_i = 500$	$k_p = 2800$ $\alpha_1 = 0.5$ $\delta_1 = 0.0062$ $k_d = 10$ $\alpha_2 = 1.5$ $\delta_2 = 2.4131$	$k_p = 8000$ $k_d = 100$ $K = \text{diag} ([100 \ 100 \ 5 \times 10^4 \ 0.5 \ 0.51])$



**Fig. 3** Evolution of the Cartesian tracking error in numerical simulation

in Fig. 4. It is obvious that the control input signals are within the allowable range that can be handled by the real actuators. Similarly, the comparison of the moving platform’s tracking error for the two model-based controllers is presented in Fig. 5. Apparently, the benefit of parameters’ adaptation in the closed-loop of a controller (AFFPD) is very significant in improving the precision and accuracy, unlike the non-adaptive model-based controller (APD) which is limited in rejecting the uncertainties and parameters variation. Both control signals are still under saturation and proper with the real actuators limits (see Fig. 6). A good parameters’ estimation convergence of the AFFPD controller is shown in Fig. 7 reducing more the moving platform’s tracking error, knowing that we initialize the parameters with much closed values to the optimal numbers. The quantifications of the errors all over the trajectory are shown in Table 2 with the improvements of each controller. It is notable to say that the simulated model-based controllers are more accurate than the non-model-based, as the calculation of the percentages shows high improvements.

**Table 2** Performance evaluation of the controllers in numerical simulation

	PD	PID	Improvement on PD (%)	NPD	Improvement on PID (%)	APD	Improvement on NPD (%)	AFFPD	Improvement on APD (%)
RMSET [cm]	0.0094	0.0078	17.02	0.0011	85.9	$2.1018 \times 10^{-5}$	98.09	$1.0862 \times 10^{-5}$	48.32
RMSE [deg]	0.1309	0.0983	24.90	0.0147	85.04	$1.6907 \times 10^{-4}$	98.85	$5.309 \times 10^{-5}$	68.6

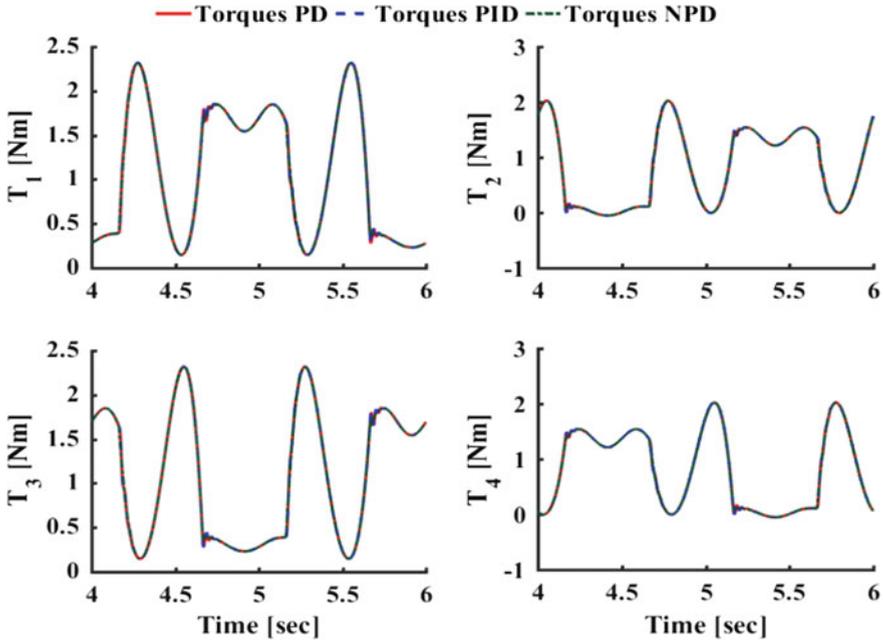


Fig. 4 Evolution of the control input signals in numerical simulation

## 4.2 Experimental Results

Due to its interesting specifications, AFFPD controller is applied in real-time experiments on VELOCE robot, and compared to the PD controller. The control architecture of the VELOCE robot is implemented using Simulink from Mathworks Inc. and compiled using XPC Target (an industrial computer of frequency 10 kHz i.e. the sample time is 0.1 ms) and the Real-Time toolboxes. Same evaluation criteria used in simulations is considered in the experiments. Retuning the gains of control design is needed for experiments, and the obtained gains for AFFPD are:  $k_p = 4000$ ,  $k_d = 6$ , and  $K = \text{diag}([2.5 * 10^{-3} \ 0.125 \ 0.1 \ 10^{-5} \ 10^{-5} \ 10^{-3}])$ . The plot of the tracking error in Cartesian space for both controllers is represented in Fig. 8, showing the better and improved global performance of AFFPD controller with respect to PD controller. The evaluations and improvements in the tracking error are computed and shown in Table 3 validating our pretend that model-based controllers are more powerful than non-model-based controllers in real-time experiments. More precisely, control schemes that include adaptive dynamics provide robustness against parameters variation and uncertainties. The control input signals of both controllers are under saturation and in the safe range (see Fig. 9). Figure 10 visualize a good convergence for the estimated parameters in the AFFPD controller, which contributes in minimizing the tracking error as possible. One can notice the

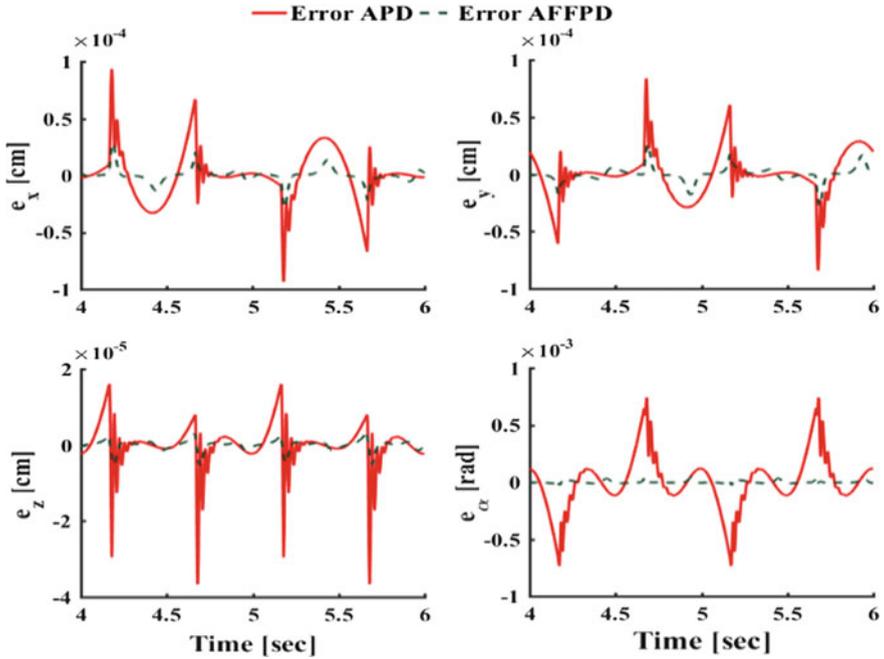


Fig. 5 Evolution of the Cartesian tracking error in numerical simulation

Table 3 Performance evaluation of PD and AFFPD controllers in real-time experiments

	PD	AFFPD	Improvements (%)
RMSET [cm]	0.0156	0.0092	41.03
RMSER [deg]	1.077	0.7596	29.47

degradation of such root mean squares of Cartesian tracking error of the two controllers from numerical simulations to real-time experiments, and that is normal because of the inaccurate model of PKMs exist in the literature, in which they simplify friction, actuators’ dynamics, transmission system,...etc.

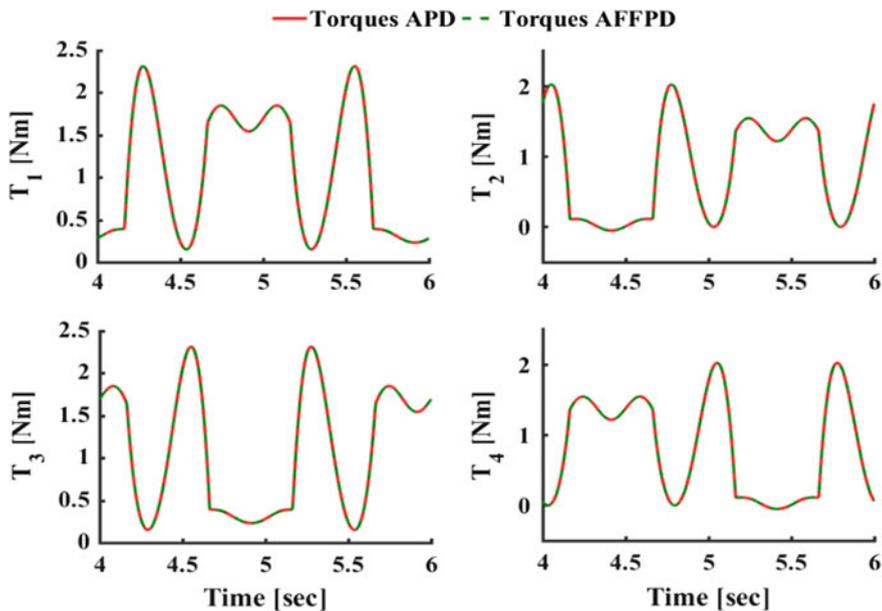


Fig. 6 Evolution of the control input signals in numerical simulation

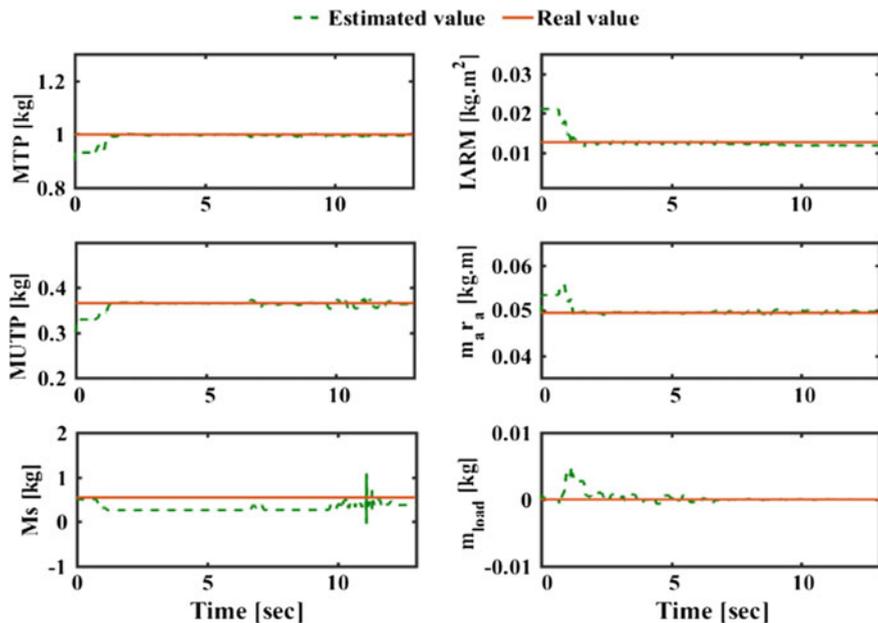


Fig. 7 Parameters estimation in numerical simulation of AFFPD

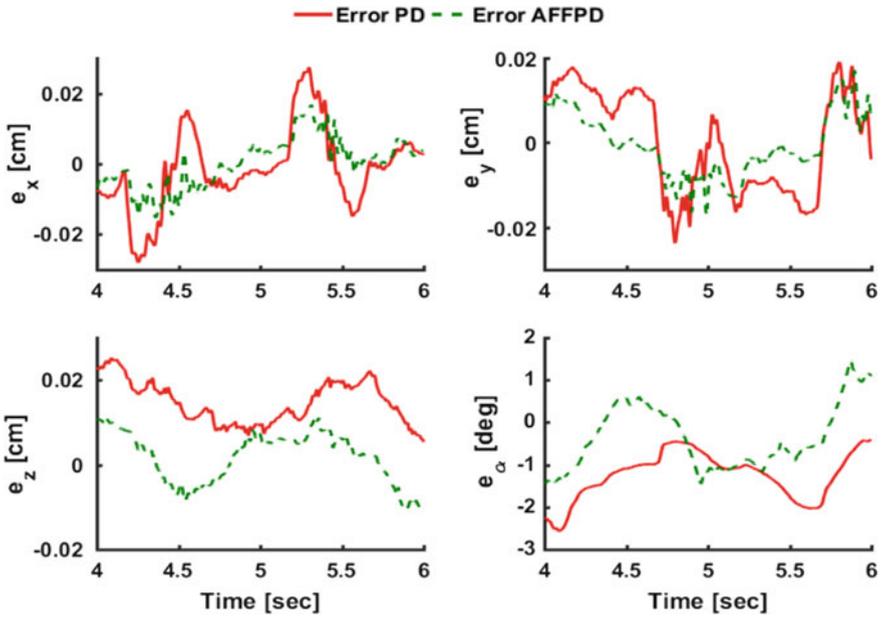


Fig. 8 Evolution of the Cartesian tracking error in real-time experiments

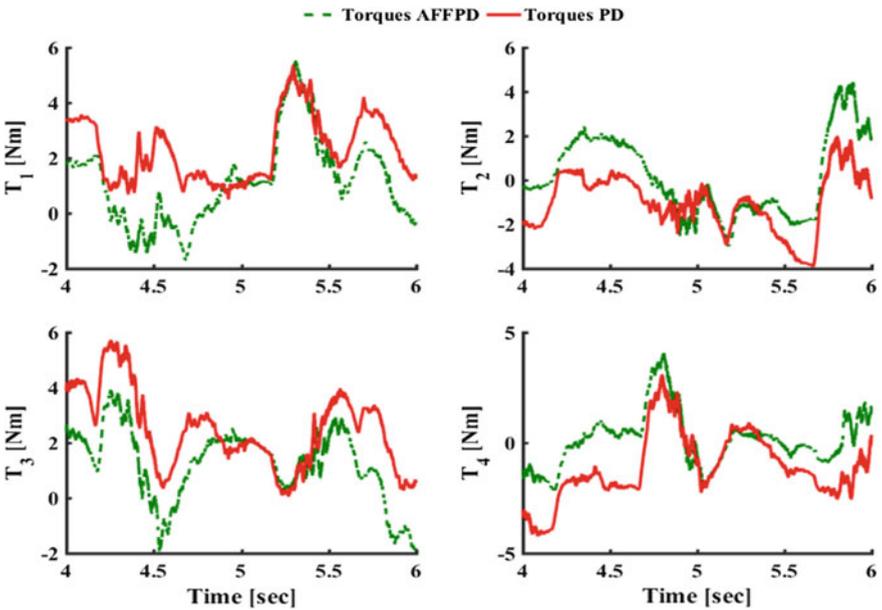


Fig. 9 Evolution of the control input signals in real-time experiments

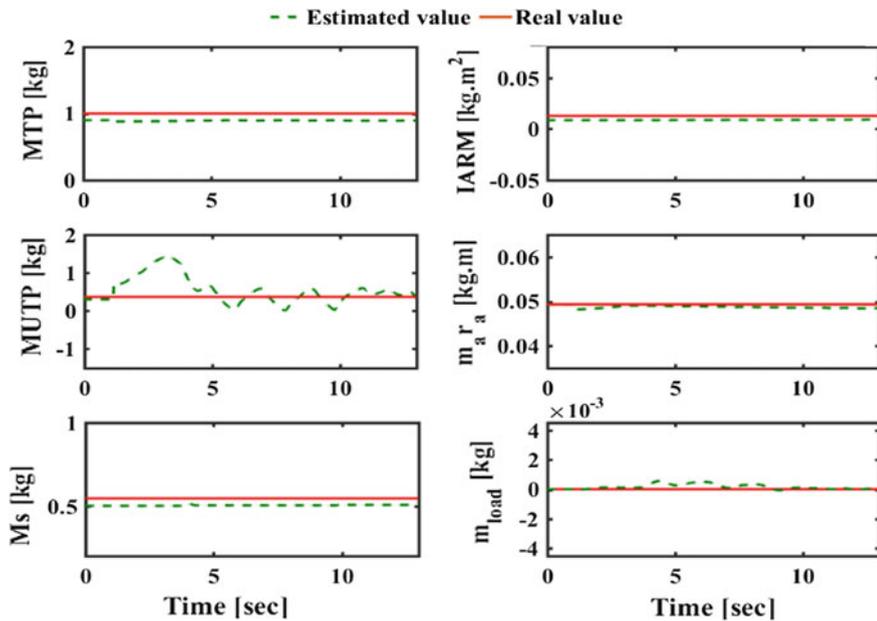


Fig. 10 Parameters estimation in real-time experiments of AFFPD

## 5 Conclusion and Future Work

In this paper, a comparison between the performances of model-based (Augmented PD, Adaptive Feedforward with PD) and non-model-based (PD, PID, Nonlinear PD) controllers was done by numerical simulation sketching and interpreting. We show the importance of including the dynamic model of the PKMs in the closed-loop control, and its main role in enhancing the performance of the controller, especially when adapting the dynamical parameters of the PKMs. Real-time experiments of PD and AFFPD controllers were conducted on a 4-DOF parallel robot to verify the validation of simulation results in the real applications of parallel robots.

As a future perspective, one can look for more accurate models of PKM involve the full dynamics such as articulations' friction, actuators' dynamics, motor drivers, and transmission system. Corporate these models in adaptive closed-loop algorithms to improve the performance of parallel robots, in terms of precision, motion speed and robustness.

**Acknowledgements** This paper has been supported by the Erasmus+ mobility project, ARPE ARROW project and the Lebanese University.

## References

1. Merlet J-P (2006) *Parallel robots*. Springer, The Netherlands
2. Bennehar M (2015) Some contributions to nonlinear adaptive control of PKMs : from design to real-time experiments. PhD thesis, University of Montpellier
3. Natal GS (2016) Control of parallel robots: towards very high accelerations. PhD thesis, University of Montpellier
4. Stewart D (1965) A platform with six degrees of freedom. *Arch Proc Inst Mech Eng* 1847–1982(180):371–386
5. Clavel R (1990) Device for the movement and positioning of an element in space, Switzerland. US Patent 4976582A. <https://patents.google.com/patent/US4976582A>
6. Bruzzone L, Molino R, Razzoli R (2002) Modelling and design of a parallel robot for laser-cutting applications. In: Proc. IASTED international conference on modeling, identification and control, Innsbruck, Austria
7. Yangmin L, Qingsong X (2007) Design and development of a medical parallel robot for cardiopulmonary resuscitation. *IEEE/ASME Trans Mechatron* 12:265–273
8. Grange S, Conti F, Rouiller P, Helmer P, Baur C (2001) Overview of the delta haptic device. *Proc Eurohaptics* 1:5–7
9. Toyama T, Susuki M (1998) Machine tool having parallel structure, Japan. US Patent 5,715,729, 1998. <https://www.google.com/patents/US5715729>
10. Kucuk S (2012) Serial and parallel robot manipulators-kinematics, dynamics, control and optimization. InTech, Croatia
11. Simaan N (1999) Analysis and synthesis of parallel robots for medical applications. Master thesis, Israel Institute of Technology
12. Liu GF, Wu YL, Wu XZ, Kuen YY, Li ZX (2001) Analysis and control of redundant parallel manipulators, South Korea. In: Proceedings of 2001 ICRA, IEEE international conference on robotics and automation (Cat. No. 01CH37164), vol 4, pp 3748–3754
13. Ziegler JG, Nichols NB (1942) Optimum settings for automatic controllers. *Trans Am Soc Mech Eng* 64:759–768
14. Su YX, Duan BY, Zheng CH (2004) Nonlinear PID control of a six-DOF parallel manipulator. *IEE Proc Control Theory Appl* 151:95–102
15. Chung I-F, Chang H-H, Lin C-T (1999) Fuzzy control of a six-degree motion platform with stability analysis. In: IEEE SMC'99 Proceedings of 1999 IEEE international conference on systems, man, cybernetics (Cat. No. 99CH37028), vol 1, pp 325–330
16. Yang C, Huang Q, Jiang H, Ogbobe Peter O, Han J (2010) PD control with gravity compensation for hydraulic 6-DOF parallel manipulator. *Mech Mach Theory* 45:666–677
17. Kelly R (1997) PD control with desired gravity compensation of robotic manipulators: a review. *Int J Rob Res* 16:660–672
18. Codourey A (1998) Dynamic modeling of parallel robots for computed-torque control implementation. *Int J Rob Res* 17:1325–1336
19. Shang WW, Cong S, Li ZX, Jiang SL (2009) Augmented nonlinear PD controller for a redundantly actuated parallel manipulator. *Adv Robot* 23:1725–1742
20. Codourey A, Honegger M, Burdet E (1997) A body-oriented method for dynamic modeling and adaptive control of fully parallel robots. In: Proceedings of 5th Symposium Robot Control, pp 443–450
21. Nabat V (2007) Robots parallèles à nacelle articulée, du concept à la solution industrielle pour le pick-and-place. PhD thesis, University of Montpellier
22. Khalil W, Dombre E (2004) *Modeling, identification and control of robots*. Elsevier Ltd., London
23. Sadegh N, Horowitz R (1990) Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *Int J Rob Res* 9:74–92
24. Boyd S, Sastry SS (1986) Necessary and sufficient conditions for parameter convergence in adaptive control. *Automatica* 22:629–639