

## Article

# Model-Based Control of a 4-DOF Rehabilitation Parallel Robot with Online Identification of the Gravitational Term

Rafael J. Escarabajal <sup>1,\*</sup> , José L. Pulloquina <sup>1</sup> , Vicente Mata <sup>2</sup> , Ángel Valera <sup>1</sup>  and Miguel Díaz-Rodríguez <sup>3</sup> 

<sup>1</sup> Departamento de Ingeniería de Sistemas y Automática, Instituto de Automática e Informática Industrial, Camino de Vera s/n, 46022 Valencia, Spain

<sup>2</sup> Centro de Investigación en Ingeniería Mecánica, Departamento de Ingeniería Mecánica y de Materiales, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain

<sup>3</sup> Departamento de Tecnología y Diseño, Facultad de Ingeniería, Núcleo la Hechicera, Universidad de los Andes, Merida 5101, Venezuela

\* Correspondence: raessan2@doctor.upv.es

**Abstract:** Parallel robots are being increasingly used as a fundamental component of lower-limb rehabilitation systems. During rehabilitation therapies, the parallel robot must interact with the patient, which raises several challenges to the control system: (1) The weight supported by the robot can vary from patient to patient, and even for the same patient, making standard model-based controllers unsuitable for those tasks since they rely on constant dynamic models and parameters. (2) The identification techniques usually consider the estimation of all dynamic parameters, bringing about challenges concerning robustness and complexity. This paper proposes the design and experimental validation of a model-based controller comprising a proportional-derivative controller with gravity compensation applied to a 4-DOF parallel robot for knee rehabilitation, where the gravitational forces are expressed in terms of relevant dynamic parameters. The identification of such parameters is possible by means of least squares methods. The proposed controller has been experimentally validated, holding the error stable following significant payload changes in terms of the weight of the patient's leg. This novel controller allows us to perform both identification and control simultaneously and is easy to tune. Moreover, its parameters have an intuitive interpretation, contrary to a conventional adaptive controller. The performance of a conventional adaptive controller and the proposed one are compared experimentally.

**Keywords:** adaptive control; dynamic parameter identification; relevant parameters; parallel robot



**Citation:** Escarabajal, R.J.; Pulloquina, J.L.; Mata, V.; Valera, Á.; Díaz-Rodríguez, M. Model-Based Control of a 4-DOF Rehabilitation Parallel Robot with Online Identification of the Gravitational Term. *Sensors* **2023**, *23*, 2790. <https://doi.org/10.3390/s23052790>

Academic Editor: Brahim Brahmi

Received: 23 January 2023

Revised: 21 February 2023

Accepted: 28 February 2023

Published: 3 March 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Parallel robots (PRs) consist of closed kinematic chains involving a fixed and a mobile platform, where an end-effector is defined and controlled. They have extensive applications [1] as they present several advantages over their serial counterparts regarding accuracy and dynamic behavior with medical applications being one of the most notable [2,3]. Conventional lower limb rehabilitation is complicated and requires intensive work; thus, robotic assistance is increasingly becoming a solution to complement conventional protocols [4]. Regarding lower limb rehabilitation tasks, devices such as gait trainers [5] and ankle rehabilitation robots [6] have been widely studied. In this research, we consider a PR with four degrees of freedom (DOF) developed for knee rehabilitation and diagnosis purposes.

A mechanical design of a human knee reeducation mechanism was presented in [7]. In [8], a control system implemented on a PR was designed for lower limb rehabilitation of bedridden stroke survivors. Parallel robotic manipulators can be kinematically optimized by design to avoid singularities [9], to achieve accuracy even with potential changes in the geometric parameters during performance [10], or with task-specific criteria using a performance index [11].

Robots can be controlled in various ways. In particular, PID-based controllers predominate in industrial environments [12,13]. These controllers do not generally use a dynamic model. In manipulations involving interactions with a human similar to those considered in this study, a model-based control technique is better suited [14]. For example, Computed-Torque Control [15,16] is an approach that uses the inverse dynamic model to compensate for all the dynamic effects in real time. However, this kind of scheme requires precise model identification [17]. An important limitation of standard model-based control is that it cannot deal with parameter changes, in that if any of the inertial (including the payload) or friction parameters change online, the model remains the same, becoming inaccurate. The variation in parameters may be due to a natural degradation of the mechanism, unmodeled or non-accurate operational regime, and manual change of the payload. Rehabilitation therapies with human–robot interaction are affected by this issue since the force exerted on the robot can vary among patients and even for the same patient, so standard model-based controllers become inaccurate.

It is well-known that the rigid body dynamic model is linear in the inertial parameters [18], and the problem of time-dependent dynamic parameters has been tackled using robust control [19] and adaptive control [20], which are approaches commonly used where the controller tries to respond to uncertainties. However, it may be that not all the dynamic parameters estimated by the adaptive controller are useful as only a subset of those parameters contributes to the behavior of the robot. The subset of reduced dynamic parameters is known as base parameters. Gautier [21] proposed Singular Value Decomposition (SVD) to identify the base parameters of a PR. However, a PR struggles to excite useful trajectories properly due to the constrained range of movements that characterize these robots [22,23]. Díaz-Rodríguez et al. [24] reduced the set of base parameters to a new subset of relevant parameters considering physical feasibility. The relevant parameters lead to simpler, more robust models and fewer computational requirements for an adaptive controller, while keeping the linear relationship of the model with respect to the relevant parameters.

Recent research has been conducted regarding the identification of relevant parameters [24] and adaptive control with real-time updates [25] applied to a 3-DOF parallel manipulator for rehabilitation purposes with a variable payload. Duan et al. [26] proposed a method to identify a payload on a serial robot based on trajectory excitation to compensate for gravity and inertial force. However, they relied on the use of an external force sensor to perform the identification. Kim et al. [27] implemented a system identification method for a delta robot to estimate a set of uncertain parameters to be included in the dynamic model, although it did not experience significant payload changes.

Motivated by the aforementioned studies, the contribution of this paper is the development of a model-based controller comprising a proportional-derivative (PD) controller and an online relevant parameter identifier for the gravitational model applied to a 4-DOF knee rehabilitation PR. These relevant parameters are extracted to improve the condition of the linear model such that simple linear regression techniques can be robustly applied online to get an accurate measure of the gravitational term. Firstly, an offline identification of the relevant parameters is carried out by applying the SVD process to a set of knee rehabilitation trajectories. The second step involves the selection of the relevant parameters considering physical feasibility. The subsequent online estimation of these relevant parameters is performed by using the reduced model obtained from the SVD process and a least squares algorithm that exploits the well-conditioned problem. In particular, the moving-window least squares and recursive least squares methods are implemented to create two model-based controllers that are successfully applied to both simulations and the actual PR.

The experiments performed are designed specifically for knee rehabilitation purposes. In control tasks, this identification allows to keep the error stable after a payload change, while at the same time estimate such payload without requiring a force sensor since it is shown to be included within the relevant parameters, which would turn very convenient for estimating the patient leg's weight in the rehabilitation context. Moreover, the parameters

of the regression are few, as well as intuitive and easy to tune, in contrast to the more involved tuning required for a conventional adaptive controller. The results of the proposed controllers are evaluated along with an adaptive controller designed for the same PR to show the main advantages of the new proposed algorithms.

This paper is organized as follows: Section 2 introduces the 4-DOF PR, its dynamic model, the base parameters identification process based on the SVD procedure, and the relevant parameter selection. Section 3 summarizes the adaptive controller for the PR under study and presents the two variants of the proposed controller. Section 4 describes the experimental procedure and results. Finally, the main conclusions are summarized in Section 5.

## 2. Parallel Robot for Knee Rehabilitation and Diagnosis

### 2.1. Knee Rehabilitation and Diagnosis

The purpose of the PR is the rehabilitation of knee injuries but also the diagnosis of Anterior Cruciate Ligament (ACL)-deficient knee. Additionally, it could be applied to the rehabilitation of ankle injuries.

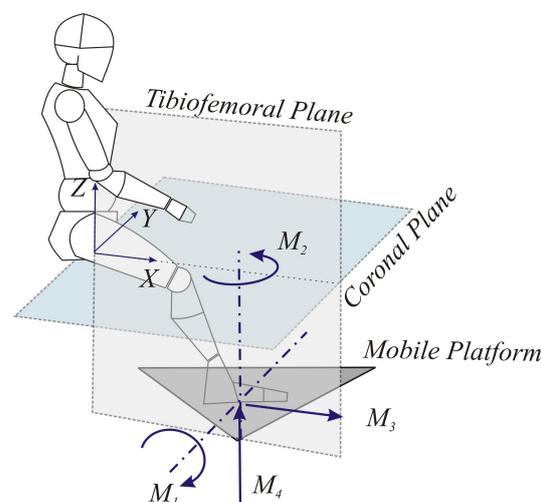
In clinical rehabilitation, knee rehabilitation tasks involve flexion–extension movements of the leg in the tibiofemoral plane. In some cases, the flexion–extension of the knee is combined with hip flexion [28]. During the knee rehabilitation procedure, the ankle is adapted to the best physiological pose of the patient, so its movement must be controlled.

The Anterior Cruciate Ligament (ACL) limits the translational and rotational movements of the tibia with respect to the femur. Consequently, knee instability is an accepted indicator of ACL damage. Specific tests have been developed to detect it. The two most recognized are:

- The Lachman test: Relative displacements of the tibia/femur in the tibiofemoral plane [29].
- The Pivot Shift Test: Reproduce translational and rotational instability in the knee in the coronal plane [30].

Based on the fundamental movements for clinical rehabilitation and diagnosis of the knee, the movements required for robotic devices involved in this task are (Figure 1):

- M1 Flexion of the leg in the direction perpendicular to the tibiofemoral plane.
- M2 Rotation of the leg along an axis perpendicular to the PR mobile platform.
- M3 Translation in the direction of the X-axis, contained in the tibiofemoral plane.
- M4 Translation in the direction of the Z-axis, contained in the tibiofemoral plane.



**Figure 1.** Movements required for knee rehabilitation.

The movements M3 and M4, which must be disengaged, allow the development of knee rehabilitation tasks. For this purpose, M1 is essential to achieve a proper position of

the leg during rehabilitation/diagnosis tasks. In addition, this movement could perform flexion and/or extension of the ankle in the tibiofemoral plane. The Lachman test can be done primarily by the M3 movement. The Pivot Shift Test is specifically performed by the M2 movement. The previous movements can be achieved with the 4-DOF PR designed and built at the Universitat Politècnica de València [31].

2.2. Architecture

The 4-DOF PR to perform rehabilitation and diagnosis is shown in Figure 2. It has a 3UPS+RPU architecture, meaning that the mobile platform is connected to the fixed platform using three external limbs in UPS configuration and a central limb in RPU configuration. The letters R, P, U and S stand for the revolute, prismatic, universal, and spherical joints, respectively. In addition, the actuated joints are underlined.

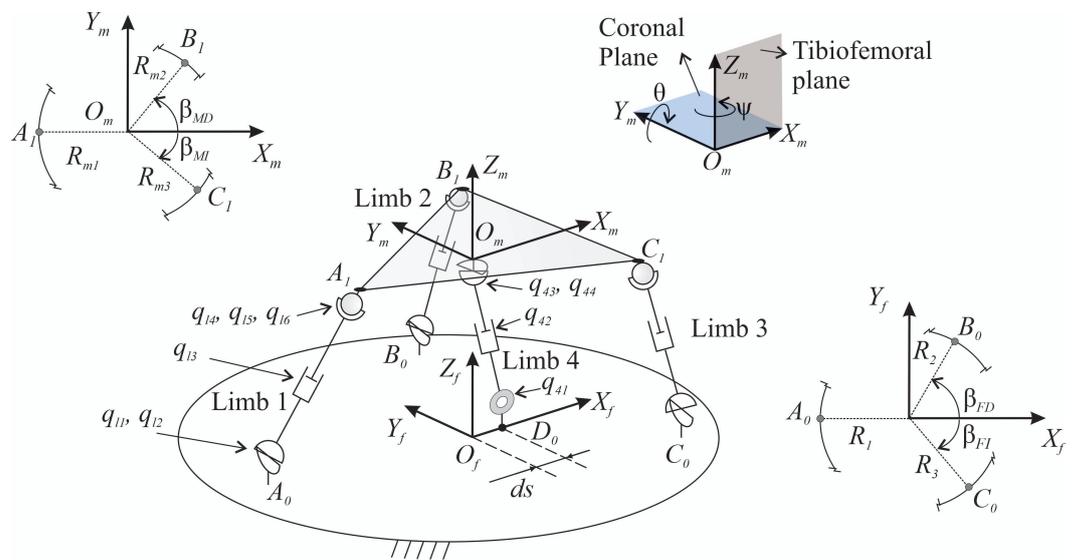


Figure 2. Architecture of the 4-DOF PR.

The position of each joint is defined by 11 generalized coordinates ( $q_{11}, q_{12}, q_{13}, \dots, q_{41}, q_{42}$ ), where the first subindex corresponds to the limb and the second refers to the joint. The 4 DOF of the mobile platform are expressed by four generalized coordinates ( $x_m, z_m, \theta, \psi$ ), see Figure 2. The entire set of generalized coordinates are classified as independent, i.e., actuated. In particular, they correspond to the prismatic joints ( $\vec{q}_{ind}$ ). The secondary coordinates are the passive joints and the mobile platform coordinates ( $\vec{q}_s$ ). The set of coordinates is defined as follows:

$$\vec{q}_s = [q_{11} \ q_{12} \ q_{21} \ q_{22} \ q_{31} \ q_{32} \ q_{41} \ x_m \ z_m \ \theta \ \psi]^T \tag{1}$$

$$\vec{q}_{ind} = [q_{13} \ q_{23} \ q_{33} \ q_{42}]^T$$

In this study, the 15 generalized coordinates ( $N = 15$ ) are organized as follows:

$$\vec{q} = [\vec{q}_s \ \vec{q}_{ind}]^T \tag{2}$$

Both the inverse and forward kinematic analysis of the considered robot have been addressed in [32].

The fundamental movements designed for rehabilitation based on the 3UPS+RPU PR are depicted in Table 1.

**Table 1.** Description of test trajectories for 3UPS+RPU PR.

Movements of Lower Limb	Trajectory of 4-DOF PR
Flexion–extension of knee combined with hip flexion	Elliptical motion on $Z_f$ axis as a function of the displacement on the $X_f$ axis
Flexion–extension of knee combined with hip flexion, ankle, and knee rotations	Elliptical motion on $Z_f$ axis as a function of displacement on the $X_f$ axis, simultaneous to rotations $\theta$ and $\psi$
Partial internal–external knee rotation	Rotation $\psi$ with a constant $X_f$
Ankle rotation	Rotation $\theta$

### 2.3. Dynamic Model

The dynamic model of the 3UPS+RPU PR is determined by the Principle of Virtual Power and applying D'Alembert's Principle. Considering that the PR is modelled by a set of generalized coordinates, the equation of motion of the 3UPS+RPU PR is given by:

$$M_{N \times N} \cdot \ddot{\vec{q}}_{N \times 1} + Q_{cyc_{N \times N}} \cdot \dot{\vec{q}}_{N \times 1} + \vec{Q}_{grav_{N \times 1}} - \vec{Q}_{act_{N \times 1}} - J_{q_{11 \times N}}^T \cdot \vec{\lambda}_{11} = \vec{0} \quad (3)$$

where:

- $\vec{q}$ : Vector of generalized velocities.
- $\ddot{\vec{q}}$ : Vector of generalized accelerations.
- $M$ : PR mass matrix.
- $Q_{cyc}$ : Matrix grouping the generalized inertial forces related to Coriolis and Centrifugal acceleration terms.
- $\vec{Q}_{grav}$ : Vector of generalized gravitational forces.
- $\vec{Q}_{act}$ : Vector of generalized active forces.
- $J_q$ : Constraint Jacobian matrix.
- $\vec{\lambda}$ : Vector of Lagrange multipliers.

In order to avoid  $\vec{\lambda}$ , which are unknown variables, an orthogonal complement  $R^*$  is considered [33], verifying that:

$$R^{*T} \cdot J_q^T = \vec{0} \quad (4)$$

The Jacobian matrix  $J_q$  expresses the derivatives of the geometrical constraints on both the independent and secondary variables. The first columns refer to the  $M = 11$  secondary generalized coordinates, and the last columns correspond to the  $F = 4$  independent variables. Equation (4) is fulfilled by defining  $R^{*T}$  as:

$$R^{*T} = \begin{bmatrix} -[J_q^s]_{M \times M}^{-1} \cdot [J_q^{ind}]_{M \times F} \\ 1_{F \times F} \end{bmatrix}_{N \times F} \quad (5)$$

Multiplying both sides of the expression (3) by  $R^{*T}$ , the dynamic model could be defined in terms of the four independent generalized coordinates ( $F = 4$ ), leading to:

$$(R^*)_{F \times N}^T \cdot (M_{N \times N} \cdot \ddot{\vec{q}}_{N \times 1} + Q_{cyc_{N \times N}} \cdot \dot{\vec{q}}_{N \times 1} + \vec{Q}_{grav_{N \times 1}}) = \vec{\tau}_{F \times 1} \quad (6)$$

where  $\vec{\tau}$  is the vector of forces applied by the actuated joints. Due to the architecture of the 3UPS+RPU PR, the most important friction forces are located at the screw-ball prismatic actuators ( $\vec{F}_f$ ) and can be added to the reduced dynamic model as follows:

$$(R^*)_{F \times N}^T \cdot (M_{N \times N} \cdot \ddot{\vec{q}}_{N \times 1} + Q_{cyc_{N \times N}} \cdot \dot{\vec{q}}_{N \times 1} + \vec{Q}_{grav_{N \times 1}}) + \vec{F}_{f_{F \times 1}} = \vec{\tau}_{F \times 1} \quad (7)$$

Finally, grouping each term using  $(R^*)_{F \times N}^T$  in the reduced dynamic model, Equation (7) is rewritten as follows:

$$M(\vec{q}) \cdot \ddot{\vec{q}} + C(\vec{q}, \dot{\vec{q}}) \cdot \dot{\vec{q}} + \vec{G}(\vec{q}) + \vec{F}_f(\vec{q}) = \vec{\tau} \quad (8)$$

According to [34], the dynamic model of the 3UPS+RPU PR considering a linear friction model can be written linearly with respect to a set of parameters, that is:

$$M(\vec{q}) \cdot \ddot{\vec{q}} + C(\vec{q}, \dot{\vec{q}}) \cdot \dot{\vec{q}} + \vec{G}(\vec{q}) = K_{rb}(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) \cdot \vec{\Phi}_{rb} \quad (9)$$

$$\vec{F}_f(\vec{q}) = K_f(\vec{q}) \cdot \vec{\Phi}_f \quad (10)$$

Applying Equations (9) and (10) to (8), it becomes:

$$\vec{\tau} = \begin{bmatrix} K_{rb}(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) & K_f(\vec{q}) \end{bmatrix} \cdot \begin{bmatrix} \vec{\Phi}_{rb} \\ \vec{\Phi}_f \end{bmatrix} \quad (11)$$

where  $\vec{\Phi}_{rb}$  and  $\vec{\Phi}_f$  are the rigid body dynamics and friction parameters, respectively.  $K_{rb}$  and  $K_f$  are a matrix related to the dynamic model and linear friction model, respectively. Regarding the friction model, the Coulomb and viscous linear model [35] is considered. The friction model was selected based on the requirements of velocity for knee rehabilitation and diagnosis tasks.

#### 2.4. Gravitational Component

Knee rehabilitation and diagnosis is performed at a low velocity, e.g., choosing  $20 \text{ mm} \cdot \text{s}^{-1}$  for translational movements and  $2^\circ \cdot \text{s}^{-1}$  for rotational motion. Figure 3 shows the control actions  $\vec{\tau}$  for the PR's limb 2 in a knee flexion–extension combined with hip flexion and knee rotation [36]. Figure 3a verifies that the dynamic behavior of the PR is mainly affected by the gravitational component  $\vec{G}$  and the friction effect on the actuated joints  $\vec{F}_f$ . In addition, Figure 3b shows a low contribution of the inertial terms ( $\vec{F}_{in} = M(\vec{q}) \cdot \ddot{\vec{q}}$ ) and the Coriolis and centrifugal terms ( $\vec{F}_{cyc} = C(\vec{q}, \dot{\vec{q}}) \cdot \dot{\vec{q}}$ ).

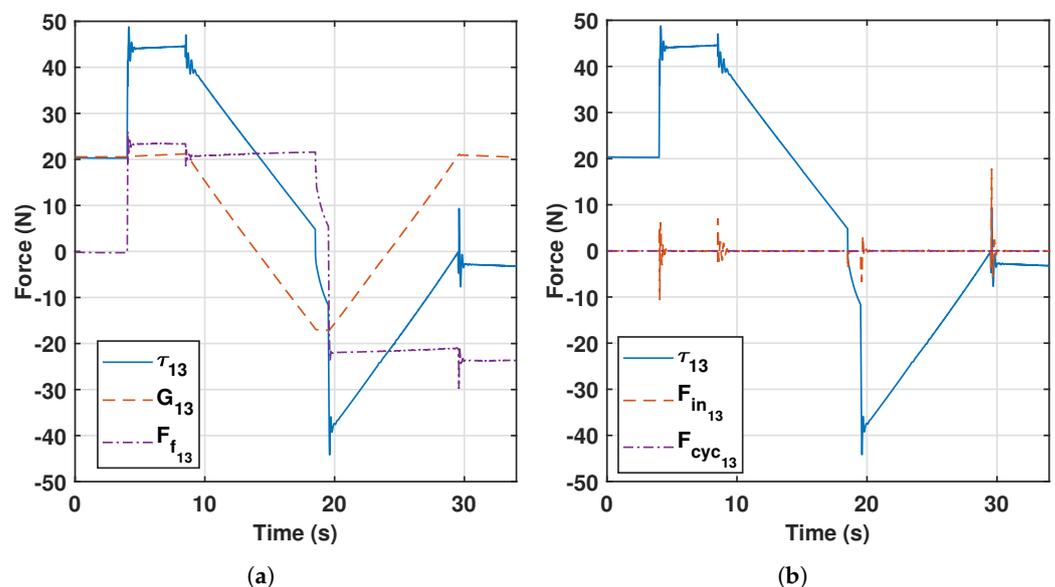


Figure 3. Decomposition of general forces in a rehabilitation trajectory (a)  $\vec{G}$ ,  $\vec{F}_f$ ; (b)  $\vec{F}_{in}$ ,  $\vec{F}_{cyc}$ .

For knee rehabilitation tasks, the term  $\vec{G}$  changes according to the weight of the patient's leg; thus, this research focuses on the identification of gravitational parameters. That is:

$$\vec{G}(\vec{q})_{F \times 1} = K_G(\vec{q})_{F \times 36} \cdot \vec{\Phi}_{G_{36 \times 1}} \quad (12)$$

$$\vec{\Phi}_G = [m_{11} \ m_{11}x_{G_{11}} \ m_{11}y_{G_{11}} \ m_{11}z_{G_{11}} \ m_{12} \ m_{12}x_{G_{12}} \ m_{12}y_{G_{12}} \ m_{12}z_{G_{12}} \ \cdots \ m_m \ m_mx_{G_m} \ m_my_{G_m} \ m_mz_{G_m}]^T \quad (13)$$

where  $m_{ij}$ ,  $m_{ij}x_{G_{ij}}$ ,  $m_{ij}y_{G_{ij}}$ ,  $m_{ij}z_{G_{ij}}$  are the masses and the first moments of mass in components  $x$ ,  $y$ , and  $z$ , respectively. The subindex  $i$  indicates the limb of the PR, and the subindex  $j$  represents the cylinder ( $j = 1$ ) or rod ( $j = 2$ ) of the limb. The subindex  $m$  is related to the mobile platform properties. The first moments of mass are measured with regard to a local reference system so that they do not match the centroid of each link.

Finally, considering the terms  $\vec{G}$  and  $\vec{F}_f$  as the main dynamic components, Equation (11) becomes:

$$\vec{\tau} = [K_G(\vec{q}) \ K_f(\vec{q})] \cdot \begin{bmatrix} \vec{\Phi}_G \\ \vec{\Phi}_f \end{bmatrix} \quad (14)$$

### 2.5. Base Parameter Identification

Dynamic parameter identification may be performed by two methods [37]:

- Direct: Performing a single experiment to identify all the parameters of the dynamic model simultaneously. For the PR under study, this method involves identifying  $\vec{\Phi}_G$  and  $\vec{\Phi}_f$  simultaneously.
- Indirect: Developing different experiments to sequentially identify the parameters, e.g., first the  $\vec{\Phi}_f$  is identified and then the  $\vec{\Phi}_G$  is identified.

The friction parameter identification depends on factors such as the joint surface condition, joint temperature, and lubricant distribution. According to [34], in PRs with a prismatic actuator, the friction effect on actuators makes it difficult to identify the rigid body parameters. Based on this assumption, in this study, we selected the indirect identification of  $\vec{\Phi}_G$ , where a linear friction model is previously identified.

The experiment used to define the friction model was carried out independently for each actuator and is based on the fact that the friction term can be approximated by subtracting the gravitational force to the total force using (8) and ignoring the inertial and centrifugal and Coriolis terms, as justified in Figure 3. This strategy was followed because measuring directly the friction force is far more complex than sensing the gravitational component, and also the total force is directly provided by the control computer.

To this end, a force sensor was attached on top of the actuator (replacing the spherical joint), corresponding to a 0–100 kg load cell of model JLBm, capable of measuring the force on its axis, which was vertically aligned together with the actuator. This setup allows to perform experiments with different payloads attached to the sensor. The gravitational term comes from the readings of the sensor, which accounts for the external payload applied to the actuator and the acceleration due to the trajectory. However, to this term must be added the force of the rest of elements displaced by the actuators, corresponding to the weights of the sensor itself and the rod. The experiments were performed in two ways (Figure 4):

- Compression experiment, where the mass is directly attached to the axis of the sensor (Figure 4a).
- Traction experiment, where a system of pulleys is used to provoke traction force with the payload attached (Figure 4b).

This experiment allows to determine the effects of different masses on the friction response as depicted in Figure 5, where several masses between 1 and 4 kg were used for the external actuators, and from 1 to 10 kg were attached to the central actuator, both for compression and traction. This justifies that the friction force depends on the velocity, but not on the magnitude of the mass placed. For the external limbs, the results are similar since the actuator is the same for all of them, while the central limb shares the shape of the curve but with different magnitude. The data have been fitted with a Coulomb and viscous linear model since it is the most widely used model due to its simplicity and it captures the behavior of the signals (see Figure 5).

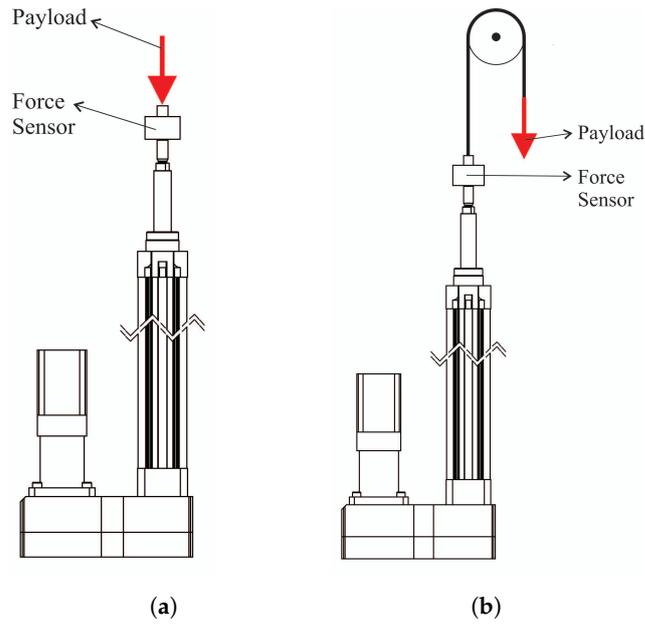


Figure 4. Friction experiments for (a) compression and (b) traction.

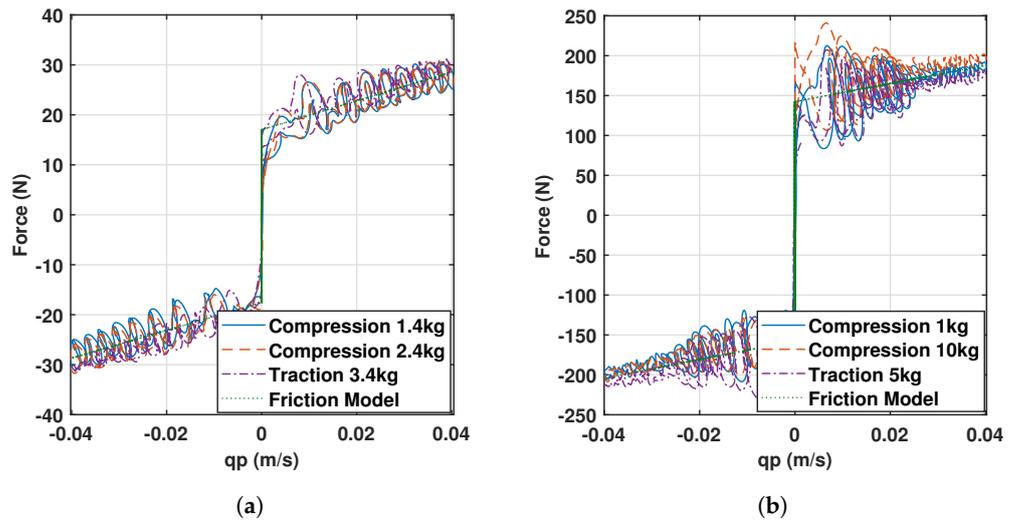


Figure 5. Friction responses and models for (a) the external limbs and (b) the central limb of the 3UPS+RPU PR.

Usually, the number of parameters to identify is greater than the number of available dynamic equations, making (14) an undetermined system. By applying (14) to  $N_{pts}$  different configurations of the 3UPS+RPU PR, the following overdetermined system is obtained:

$$\vec{T} = W_G(\vec{q}) \cdot \vec{\Phi}_G \tag{15}$$

with:

$$\vec{T} = \begin{bmatrix} \vec{\tau}_1 - \vec{F}_{f_1} \\ \vec{\tau}_2 - \vec{F}_{f_2} \\ \vdots \\ \vec{\tau}_{N_{pts}} - \vec{F}_{f_{N_{pts}}} \end{bmatrix} \tag{16}$$

$$W_G(\vec{q}) = \begin{bmatrix} K_{G_1} \\ K_{G_2} \\ \vdots \\ K_{G_{N_{pts}}} \end{bmatrix} \quad (17)$$

where  $\vec{T}$  and  $W_G$  are observations from linear model (14) in different configurations.

After applying the SVD procedure to (15), the parameters  $\vec{\Phi}_G$  are linearly combined to obtain a set of base parameters  $\vec{\Phi}_{base}$  that simplifies the overdetermined system as follows:

$$\vec{T} = W_{base_{(F \cdot N_{pts}) \times r}}(\vec{q}) \cdot \vec{\Phi}_{base_{r \times 1}} \quad (18)$$

where  $r$  is the rank of the observation matrix  $W_G$ . For an arbitrary configuration, the identified  $\vec{\Phi}_{base}$  determine  $\vec{\tau}$  as:

$$\vec{\tau} = \vec{F}_f + K_{base}(\vec{q}) \cdot \vec{\Phi}_{base} \quad (19)$$

where  $K_{base}$  is defined by the first  $r$  columns of the product  $K_G \cdot P$ , with  $P$  as the permutation matrix of the SVD procedure [21]. A detailed explanation of this base parameter identification procedure is presented in [34].

## 2.6. Offline Identification Results

The workspace of a PR is smaller than that of a serial one, and a PR usually has Type II singularities inside its workspace, reducing its capacity to develop exciting trajectories. In consequence, the first base parameter identification performed in this study considers nine fundamental knee rehabilitation trajectories [36] for the 3UPS+RPU PR (offline identification). In this offline identification process the vector of forces  $\vec{T}$  is obtained by stacking together several samples coming from the subtraction of the friction force to the total control action in  $N_{pts}$  time steps, according to (16). Note that the friction force is calculated with the Coulomb and viscous model as defined in Section 2.5 using the velocity provided by the control computer. On the other hand, the observation matrix  $W_G$  also collects data from those time steps and depends on the generalized coordinates ( $\vec{q}$ ) as expressed in (17), which are obtained from the active generalized coordinates ( $\vec{q}_{ind}$ ) measured by the encoders.

The trajectories define a non-singular  $W_G$  matrix, which has 6 null columns related to 6 parameters of  $\vec{\Phi}_G$ . These 6 parameters, with no dynamic effect, are not included in the  $W_G$  matrix; thus, the SVD identification procedure starts with  $n = 30$  parameters. After applying the SVD procedure to the 3UPS+RPU PR,  $r = 15$  base parameters are identified, see Table 2.

**Table 2.** Base parameters for the 3UPS+RPU PR.

No.	Base Parameter
1	$m_{12} + m_{22} + m_{32} + m_{42} + m_m$
2	$m_{11}x_{G_{11}} + m_{12}x_{G_{12}}$
3	$m_{21}x_{G_{21}} + m_{22}x_{G_{22}}$
4	$m_{11}z_{G_{11}} + m_{12}z_{G_{12}}$
5	$m_{21}z_{G_{21}} + m_{22}z_{G_{22}}$
6	$m_{31}z_{G_{31}} + m_{32}z_{G_{32}}$
7	$m_{31}x_{G_{31}} + m_{32}x_{G_{32}}$
8	$m_{42}y_{G_{42}} - m_{41}z_{G_{41}}$
9	$0.19m_{22} - 0.30m_{12} + m_m x_{G_m}$
10	$0.23m_{22} - 0.30m_{32} + m_m y_{G_m}$
11	$m_m z_{G_m}$
12	$m_{42}x_{G_{42}} - m_{41}x_{G_{41}}$
13	$m_{11}y_{G_{11}} + m_{12}y_{G_{12}}$
14	$m_{21}y_{G_{21}} + m_{22}y_{G_{22}}$
15	$m_{31}y_{G_{31}} + m_{32}y_{G_{32}}$

In addition, we applied the inertia transfer procedure [38] to the PR under study. In this way, the symbolic equations of each base parameter are obtained. Table 3 shows the results for base parameters 9 and 10. The inertia transfer method verifies the physical sense of the base parameter identified by the SVD procedure.

**Table 3.** Symbolic coefficients for the identified base parameters.

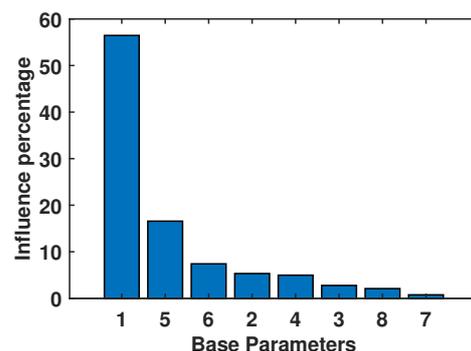
No.	Base Parameter	Inertia Transfer
9	$0.19m_{22} - 0.30m_{12} + m_m x_{G_m}$	$R_{m2} \cos(\beta_{MD}) m_{22} - R_{m1} m_{12} + m_m x_{G_m}$
10	$0.23m_{22} - 0.30m_{32} + m_m y_{G_m}$	$R_{m2} \sin(\beta_{MD}) m_{22} - R_{m3} \sin(\beta_{MI}) m_{32} + m_m y_{G_m}$

Despite the original matrix  $W_G$  being non-singular, it has a condition number of  $8.64 \times 10^{33}$ , which is extremely high for effective identification with experimental data. After estimating the 15 base parameters, the condition number of the matrix  $W_{base}$  is reduced to 647, which is still high. The architecture of the 3UPS+RPU PR has been optimized [39], and even so, its workspace is not free from Type II singularities. Therefore, the optimization of the rehabilitation trajectories to reduce the  $cond[W_{base}]$  is not a practical option. Here, the reduction of the  $cond[W_{base}]$  is achieved by eliminating the base parameters with less effect on the dynamic behavior of the PR under study.

The reduction is performed by estimating the accuracy (in percentage) gained in the estimation of the gravitational term  $\vec{G}$  when using a subset of  $r' < r$  base parameters. The resulting gravitational term is  $\vec{G}' = K'_{G_{F \times r'}} \cdot \vec{\Phi}'_{G_{r' \times 1}}$ , and the influence of the  $r'$  selected parameters is calculated as:

$$\text{influence (\%)} = \frac{\sqrt{\vec{G}'^T \cdot \vec{G}'}}{\sqrt{\vec{G}^T \cdot \vec{G}}} \cdot 100 \quad (20)$$

This expression can be applied by starting from  $r' = 1$  and gradually increasing  $r'$  to see how much influence is gained after the addition of each base parameter. For a value of  $r' = 8$ , the gravitational forces  $\vec{G}$  can be estimated with 95% accuracy, and Figure 6 breaks down the importance of each parameter in the estimation of  $\vec{G}$  in descending order. With this setup, the condition number gets down to 74. This set of 8 base parameters is considered in this study as the relevant parameters ( $\vec{\Phi}_{rel}$ ).



**Figure 6.** Base parameters with 95% dynamic influence.

The identification results for the eight relevant parameters are presented in Table 4. The simulated and actual results are presented in the second and third columns, respectively. All the values in the Table are expressed in SI units.

**Table 4.** Relevant parameter identification results for the simulated and actual 3UPS+RPU PR.

No.	Relevant Parameter	Simulation	Actual
1	$m_{12} + m_{22} + m_{32} + m_{42} + m_m$	14.12	15.60
2	$m_{11}x_{G_{11}} + m_{12}x_{G_{12}}$	-0.35	1.45
3	$m_{21}x_{G_{21}} + m_{22}x_{G_{22}}$	0.16	0.82
4	$m_{11}z_{G_{11}} + m_{12}z_{G_{12}}$	1.58	0.82
5	$m_{21}z_{G_{21}} + m_{22}z_{G_{22}}$	1.41	-0.89
6	$m_{31}z_{G_{31}} + m_{32}z_{G_{32}}$	1.40	-1.58
7	$m_{31}x_{G_{31}} + m_{32}x_{G_{32}}$	0.15	-0.37
8	$m_{42}y_{G_{42}} + m_{41}z_{G_{41}}$	-0.94	-0.76

For an arbitrary configuration of the PR, the term  $\vec{G}$  is estimated using  $\vec{\Phi}_{rel}$  as follows:

$$\vec{G}(\vec{q})_{F \times 1} \approx K_{rel}(\vec{q})_{F \times 8} \cdot \vec{\Phi}_{rel_{8 \times 1}} \quad (21)$$

where  $K_{rel}$  is a matrix comprising all the rows and columns of  $K_{base}(\vec{q})$  associated with  $\vec{\Phi}_{rel}$ .

### 3. Model-Based Control

In this Section, we describe the control of the 3UPS+RPU PR with online identification of the relevant parameters. These parameters have already been appropriately expressed so that they can be estimated accurately by using (21). However, when a human limb is placed on the PR, only the first relevant parameter changes because of the modification of the mobile platform's mass (row 1 of Table 4). Therefore, the matrix  $K_{rel_{F \times 8}}$  should be partitioned into two submatrices  $K_{E_{F \times EP}}$  and  $K_{NE_{F \times NEP}}$  (which will be denoted simply as  $K_E$  and  $K_{NE}$  unless size matters), where  $EP$  denotes the number of estimated parameters, and  $NEP = 8 - EP$  is the number of non-identified parameters, as they are supposed to be known and invariant. Similarly, the vector of parameters  $\vec{\Phi}_{rel_{8 \times 1}}$  is divided into  $\vec{\theta}_{EP \times 1}$  and  $\vec{\theta}_{NE_{NEP \times 1}}$ . These are the true values, unknown for the case of  $\vec{\theta}_E$ ; thus, the controller estimation of this vector is referred as  $\vec{\hat{\theta}}$ . Note that as the number of identified parameters increases, some estimation precision is lost if these are invariant because unnecessary uncertainty is being added. Nevertheless, the experiments will show how the controller reacts when more than one parameter varies. The experiments below show the robustness of the methods presented in this paper.

Section 3.1 briefly presents one of the current control systems used to estimate parameters online, which is applied to the 3UPS+RPU PR with a reduced set of relevant parameters. Based on this idea, Sections 3.2 and 3.3 propose two new models of control schemes where relevant parameter identification is performed online directly by regression and least squares estimation techniques, where the estimator parameters are straightforward to tune.

#### 3.1. Adaptive Controller

As a preliminary step, let us consider a well-known adaptive controller proposed by Bayard & Wen [40], which has been successfully applied to PRs, e.g., in [25]. The adaptation law is applied along with an underlying PD controller and can be written as follows:

$$\vec{\tau}_c = Y(\vec{q}) \cdot \vec{\hat{\theta}}(t) + \vec{G}_{NE}(\vec{q}) - K_d \vec{q}_{ind} - K_p \vec{e} \quad (22)$$

$$\frac{d\vec{\hat{\theta}}(t)}{dt} = -\Gamma_0 \cdot Y^T(\vec{q}) \cdot \vec{s}_1 \quad (23)$$

$$\vec{s}_1 = \vec{e} + \lambda_1 \cdot I \cdot \vec{e} \quad (24)$$

with

- $Y$ : Regressor matrix that participates in the estimation of  $\vec{\theta}_E$ . This matrix is implicitly multiplied by  $R^{*T}$ , and is calculated as  $Y = K_E$ .
- $\vec{G}_{NE}$ : Non-adaptive gravitational term, also comprising the effect of  $R^{*T}$ , and it compensates for the unidentified, known relevant parameters:  $\vec{G}_{NE} = K_{NE} \cdot \vec{\theta}_{NE}$ .
- $K_p, K_d$ : Proportional and derivative gains of the PD controller.
- $\Gamma_0$  and  $\lambda_1$ : Matrix and a scalar, which define the dynamics of the estimation process, acting like observer parameters.

The initial estimation of  $\vec{\theta}_E$  is set to the values obtained by the first EP rows of Table 2. The next rows, up to the eighth, are used to define the constant vector  $\vec{\theta}_{NE}$ . The last seven rows are not used because they only minimally affect the dynamic behavior of the PR under study. It is worth mentioning that the parameters  $\Gamma_0$  and  $\lambda_1$  are not intuitive to adjust and it is based on trial and error.

### 3.2. Online Identifier with Window-Based Least Squares (WLS)

This method performs a least squares estimation of the relevant parameters when certain conditions are satisfied. First of all, the control law of Equation (22) remains the same, i.e., there is a PD controller where gravity is compensated via two terms: an adaptive one given by the regressor matrix  $Y$  and a fixed one expressed by the vector  $\vec{G}_{NE}$ , which are also calculated similarly. However, the estimation of  $\vec{\theta}_E$  relies on an operation of the form:

$$\vec{\theta}_{EP \times 1} = W_{E(S \cdot F) \times EP}^+ \cdot \vec{T}_{E(S \cdot F) \times 1} \quad (25)$$

where the symbol (+) refers to the Moore-Penrose pseudoinverse.  $S$  is the first predefined parameter indicating the size of the window or the number of samples to collect before taking the pseudoinverse (which will be stacked, providing that they improve the information given by the followed trajectory, as will be discussed shortly), and  $F = 4$  is the number of actuators, as mentioned in Section 2.3.  $W_E$  is a matrix generated by concatenating  $S$  matrices  $K_E$  (each of size  $F \times EP$ ) as new valid data from the trajectory is fed to the system:

$$W_E = \begin{bmatrix} K_{E1} \\ K_{E2} \\ \vdots \\ K_{ES} \end{bmatrix} \quad (26)$$

where the subscripts  $1, 2, \dots, S$  denote different time steps, which need not be temporally equidistant. Similarly,  $\vec{T}_E$  is a vector comprising the exerted forces discounting the fixed gravity contribution and the friction effect, whose samples temporally match the ones chosen for  $W_E$ :

$$\vec{T}_{E(S \cdot F) \times 1} = \begin{bmatrix} \vec{\tau}_{E1} \\ \vec{\tau}_{E2} \\ \vdots \\ \vec{\tau}_{ES} \end{bmatrix} = \begin{bmatrix} \vec{\tau}_{c1} - \vec{G}_{NE1} - \vec{F}_{f1} \\ \vec{\tau}_{c2} - \vec{G}_{NE2} - \vec{F}_{f2} \\ \vdots \\ \vec{\tau}_{cS} - \vec{G}_{NEs} - \vec{F}_{fS} \end{bmatrix} = \begin{bmatrix} \vec{\tau}_{PD1} + KE \cdot \vec{\theta}_1 - \vec{F}_{f1} \\ \vec{\tau}_{PD2} + KE \cdot \vec{\theta}_2 - \vec{F}_{f2} \\ \vdots \\ \vec{\tau}_{PDS} + KE \cdot \vec{\theta}_S - \vec{F}_{fS} \end{bmatrix} \quad (27)$$

This method does not perform continuous estimation of the parameters  $\vec{\theta}_E$ . To justify this statement, let us suppose the matrix  $W_E$  is always full, that is, currently estimating the right parameters, but a sudden change of any of them is met. At this instant, new incoming data, of a different nature, will be feeding the matrix  $W_E$ , which will have mixed information from both situations, leading to poor performance that will be challenging to overcome because the upcoming data will also be affected by this drift. Moreover, considering reasons of computational efficiency, the online identifier will only be active

when an error threshold is surpassed, in terms of joint positions and references (i.e., the same error used by the PD controller). This error threshold is the second tuned parameter.

In addition to the error-based criterion that initiates the identification process, a stop condition must be designed so that the algorithm knows when the new steady state (as far as the estimation is concerned) is reached. This method proposes a convergence-based stop condition, i.e., it compares the current and previous values of  $\vec{\theta}$  and, if they differ by less than a convergence threshold, it finishes the process of identification. This threshold is the third parameter involved in the design of the controller. Both error-based and convergence-based conditions can be combined in just one condition statement that starts the algorithm.

Another aspect to consider is the way  $W_E$  and  $\vec{T}_E$  are filled and cleared. The online identification method uses a strategy based on the condition number: append new data to  $W_E$  and  $\vec{T}_E$  as long as  $W_E$  remains well-conditioned after the addition. A fourth parameter indicating the threshold for the condition number of  $W_E$  is required for such purpose.

However, in some situations the previous constraint may not be enough. For example, when estimating just one relevant parameter (which is generally the case)  $W_E$  is a vector and, thus, the condition number is always one (because it has just one singular value). To ensure that the matrix  $W_E$  is filled with samples as independently as possible even in this case, the samples are only collected if a velocity threshold (which is the fifth and last parameter) is exceeded. In other words, a trajectory is being described and the robot is not stuck in a fixed position. In addition, the matrices are cleared gradually (unless already empty), discarding the oldest sample when the error and convergence conditions are not fulfilled.

The algorithm performing the process of identification described above for every time step is given by Algorithm 1.

PARAMETERS		
Variable	Description	Default
$S$	number of samples	400
$et$	error threshold	0.001
$ct$	convergence threshold	0.001
$nt$	condition number threshold	400
$qpt$	joint velocity threshold	0.001
INPUTS		
Variable	Description	
$K_E$	matrix associated with the calculation of estimated relevant parameters	
$\vec{e}$	vector of error between position and reference	
$\vec{\tau}_E$	exerted force discounting the non-estimated gravity and friction term	
OUTPUTS		
Variable	Description	Initialization
$\vec{\theta}_{k+1}$	vector of new estimated parameters	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
PERSISTENT VARIABLES		
Variable	Description	Initialization
$\vec{\theta}_{k+1}$	vector of new estimated parameters (also output)	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
$\vec{\theta}_k$	vector of previous estimated parameters	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
$W_E$	input matrix to perform regression concatenating $K_E$ matrices	empty matrix
$\vec{T}_E$	output vector to perform regression concatenating $\vec{\tau}_E$ vectors	empty vector

**Algorithm 1:** Window-based least squares estimator

---

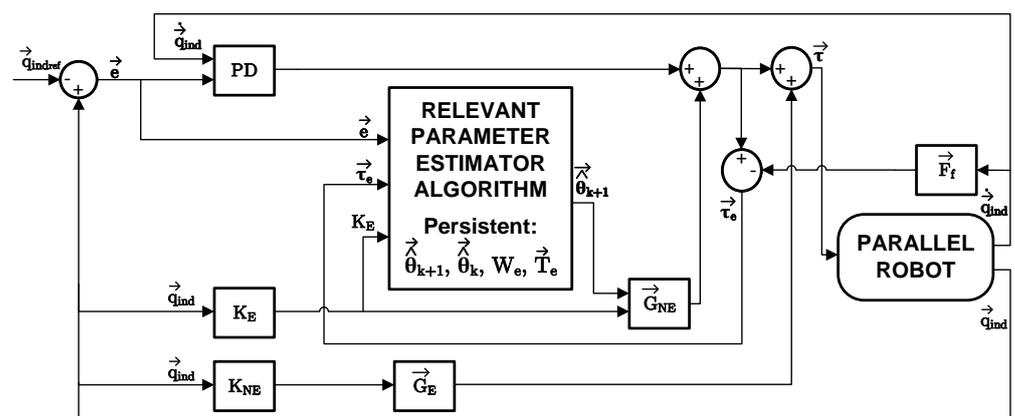
```

1 Calculate the mean absolute error  $e$  between current position and reference, from
  vector  $\vec{e}$ 
2 Calculate the convergence index  $\Delta\theta$  as the mean absolute difference of  $\vec{\theta}_{k+1}$  and  $\vec{\theta}_k$ 
3  $\vec{\theta}_{k+1} = \vec{\theta}_k$ 
4 if  $e > et$  or  $\Delta\theta > ct$  then
5   Calculate  $\vec{\tau}_E = \vec{\tau}_{PD} + \vec{G}_E - \vec{F}_f$  where  $\vec{G}_E = K_E \cdot \vec{\theta}_k$ 
6   if  $W_E$  is not full (number of rows less than  $S \cdot F$ ) then
7     Define  $W_{aux}$  by concatenating  $W_E$  and  $K_E$ 
8     Define  $\vec{T}_{aux}$  by concatenating  $\vec{T}_E$  and  $\vec{\tau}_E$ 
9   else
10    Define  $W_{aux}$  by discarding the oldest sample of  $W_E$  and appending  $K_E$ 
11    Define  $\vec{T}_{aux}$  by discarding the oldest sample of  $\vec{T}_E$  and appending  $\vec{\tau}_E$ 
12   end
13   if the condition number of  $W_{aux}$  is less than  $nt$  and the joint velocities are greater than
      $qpt$  then
14      $W_E = W_{aux}$ 
15      $\vec{T}_E = \vec{T}_{aux}$ 
16     if  $W_E$  is full then
17        $\vec{\theta}_{k+1} = W_E^+ \cdot \vec{T}_E$ 
18     end
19   end
20 else
21   Remove oldest sample from  $W_E$  (unless already empty)
22   Remove oldest sample from  $\vec{T}_E$  (unless already empty)
23 end

```

---

The representation of the controller using this estimator is described in Figure 7, where the vector  $\vec{G}_E = K_E \cdot \vec{\theta}_k$  stands for the final gravitational effect due to the estimated relevant parameters in Equation (22).



**Figure 7.** Window-based control scheme with online relevant parameter identifier.

The parameter that most affects the proposed control scheme is the number of samples (or window size), as the incoming data  $\vec{\tau}_E$  is usually a noisy signal and robustness is obtained using a large amount of variable data. Thus, there is a tradeoff between the accuracy of the estimation and the time required to update the estimated parameter, since

both of them increase with the window size. The thresholds are by default set to 0.001 (in their corresponding SI units) except for the condition number threshold, which is set to 400.

### 3.3. Online Identification Using Recursive Least Squares (RLS)

The recursive least squares method [41,42] computes the new estimate set of the relevant parameters in every iteration using the previous estimate and assuming that they are drawn from a Gaussian distribution. Moreover, the model relating the relevant parameters with the forces  $\vec{\tau}_E$  is linear with white noise  $\vec{v}$ , whose covariance is a known matrix  $V$ :

$$\vec{\theta}(t) \sim N(\vec{\theta}, \Sigma) \quad (28)$$

$$\vec{v} \sim N(\vec{0}, V) \quad (29)$$

$$\vec{\tau}_E = K_E \cdot \vec{\theta}_E + \vec{v} \quad (30)$$

By applying Bayesian inference on the distribution of  $\vec{\theta}_E$  assuming constant noise [43], the resulting posterior distribution of  $\vec{\theta}_E$  is also a Gaussian whose estimation of the mean and covariance is given by:

$$\Lambda = \lambda_f^{-1} \Sigma \quad (31)$$

$$\vec{\theta}_{k+1} = \vec{\theta}_k + \Lambda \cdot K_E^T \cdot (K_E \cdot \Lambda \cdot K_E^T + V)^{-1} \cdot (\vec{\tau}_E - K_E \cdot \vec{\theta}_k) \quad (32)$$

$$\Sigma = \Lambda - \Lambda \cdot K_E^T \cdot (K_E \cdot \Lambda \cdot K_E^T + V)^{-1} \cdot K_E \cdot \Lambda \quad (33)$$

In these expressions, instead of using the value of  $\Sigma$  to estimate  $\vec{\theta}_{k+1}$ , a forgetting factor  $\lambda_f \in (0, 1)$  is introduced to account for the extra uncertainty in the data (because the model is time-variant) and defines a new higher a priori covariance  $\Lambda$  [44]. The initial parameters of the prior distribution are set to  $\vec{\theta}_0 = \vec{\Phi}_{rel}(1 : EP)$  or  $\vec{\theta}_0 = \vec{0}$  and  $\Sigma_0 = 10^{-5} \cdot I_{EP \times EP}$ , and the noise is considered low non-zero value to avoid matrix singularities:  $V = 0.01 \cdot I_{F \times F}$ . The forgetting factor  $\lambda_f$  usually takes a value between 0.9 and 1 according to the uncertainty and dynamic behavior of the estimation. As  $\lambda_f$  approaches 1, the estimation gets smoother but slower, as it uses more information from previous data. At the limit of  $\lambda_f = 1$ , the estimation becomes inaccurate because no data is forgotten, meaning that it equates a normal regression for which all the past data (before and after the change) is considered. Values close to 1 offer good dynamic behavior of the estimation.

The forgetting factor is the only tunable parameter, which is restricted to a narrow range. Thus, this model is more robust in terms of result variability. No error or convergence checking is required because the update is performed in every iteration. Moreover, an increasing size matrix such as the one used in the least squares window-based method is no longer required.

To account for the uncertainty in the estimation (analogously to the role of the condition number in the window-based method), a variance threshold can be established as a limit for the updated a priori covariance  $\Lambda$ . In particular, the greatest eigenvalue of the matrix shows the highest variance under certain direction, after applying the forgetting factor. If that variance exceeds the threshold, the forgetting factor can be set to 1 to prevent previous useful data from being forgotten.

The pseudocode described in Algorithm 2 is run in every time step, and it implements the recursive least squares identification as follows:

PARAMETERS		
Variable	Description	Default
$\lambda_f$	forgetting factor	0.999
$vt$	variance threshold	0.001
$V$	covariance matrix of the noise	$0.01 \cdot I_{F \times F}$
INPUTS		
Variable	Description	
$K_E$	matrix associated with the calculation of estimated relevant parameters	
$\vec{\tau}_E$	exerted force discounting the non-estimated gravity and friction term	
OUTPUTS		
Variable	Description	Initialization
$\vec{\theta}_{k+1}$	vector of new estimated parameters	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
PERSISTENT VARIABLES		
Variable	Description	Initialization
$\vec{\theta}_{k+1}$	vector of new estimated parameters (also output)	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
$\vec{\theta}_k$	vector of previous estimated parameters	$\vec{\Phi}_{rel}(1 : EP)$ or $\vec{0}$
$\Sigma$	covariance matrix of the estimated values	$10^{-5} \cdot I_{EP \times EP}$

---

**Algorithm 2:** Recursive least squares estimator

1 Calculate the input force for the estimation  $\vec{\tau}_E = \vec{\tau}_{PD} + \vec{G}_E - \vec{F}_f$  where

$$\vec{G}_E = K_E \cdot \vec{\theta}_k$$

2 Calculate  $\Lambda_{aux} = \lambda_f^{-1} \Sigma$

3 **if** maximum eigenvector of  $\Lambda_{aux}$  is lower than  $vt$  **then**

4 |  $\Lambda = \Lambda_{aux}$

5 **else**

6 |  $\Lambda = \Sigma$

7 **end**

8 Calculate the updated parameters

$$\vec{\theta}_{k+1} = \vec{\theta}_k + \Lambda \cdot K_E^T \cdot (K_E \cdot \Lambda \cdot K_E^T + V)^{-1} \cdot (\vec{\tau}_E - K_E \cdot \vec{\theta}_k)$$

9 Calculate the updated covariance matrix

$$\Sigma = \Lambda - \Lambda \cdot K_E^T \cdot (K_E \cdot \Lambda \cdot K_E^T + V)^{-1} \cdot K_E \cdot \Lambda$$


---

The control scheme is the same as in Figure 7 except for the relevant parameter estimator block.

#### 4. Experimental Results

In this Section, we present the results of the relevant parameter identification and control. The fundamental objective is the simultaneous estimation of the gravitational term (i.e., the associated relevant parameters) while keeping the tracking error low thanks to the online adaptation. For that reason, both performance criteria are evaluated and compared. The reliability is also evaluated by measuring the rate of variation of the control action and the rate of variation of the estimation of the relevant parameter. A raw PD+G controller is the baseline for comparing all the methods, which leads to an important consideration: great tracking accuracy can be obtained by just increasing the proportional and derivative gains of the PD. However, this can lead to severe oscillations of the control action, which tries to adapt blindly to any condition, leading to saturations of the actuators and a great sensitivity to noise, stimulating high frequency vibration nodes. This situation becomes

inadmissible when it concerns human–robot interaction. This is a justification for why an adaptive control scheme with relatively low gains turns out to be interesting compared with a PD+G. The comparison will be established using the same gains for both controllers: the proportional gains  $K_p$  are 5800 for the external limbs and 65,000 for the central limb, and the derivative gains  $K_d$  are 400 for the external limbs and 875 for the central limb. These gains were chosen in a previous experiment with no disturbances in combination with a stationary gravity compensator. Afterwards, the gravity compensator was replaced by the estimator in the experiments with the changing environment.

#### 4.1. Experimental Procedure

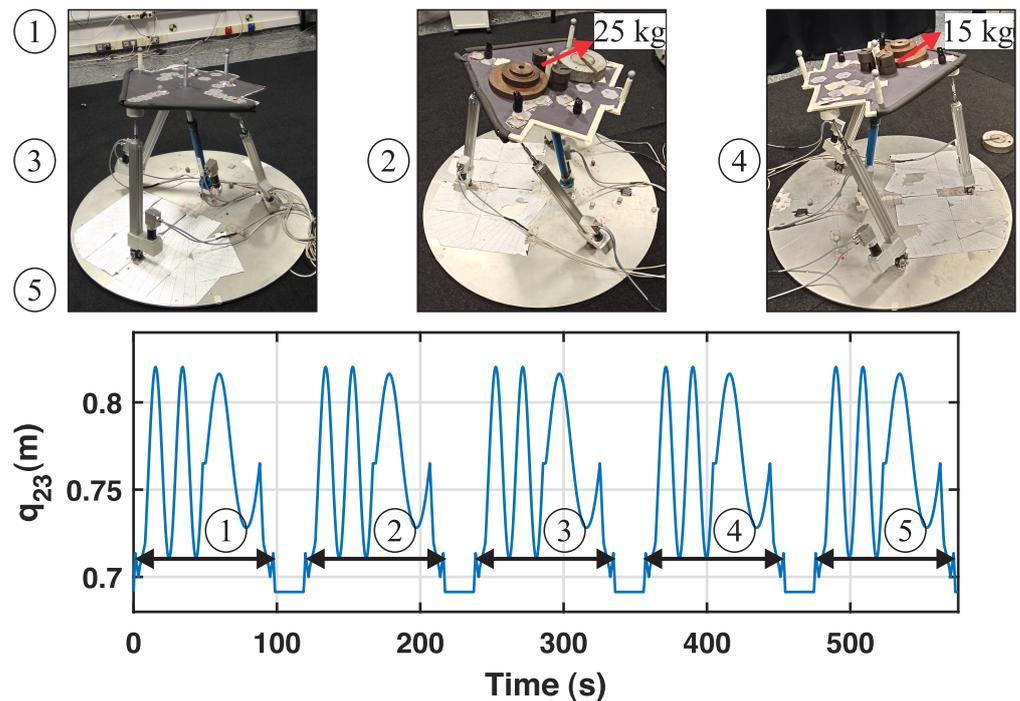
Each leg of the actual robot is driven by a prismatic actuator, Festo DNCE 32-BS10 for the external limbs ( $q_{13}$ ,  $q_{23}$ ,  $q_{33}$ ) and NIASA M100-F16 for the central limb ( $q_{42}$ ), all attached to Maxon 148867 DC motors, which are commanded by ESCON 50/5 servocontrollers. These controllers offer an accurate current control of the motors, and the current-torque relationship is defined according to their datasheets. DC motors are equipped with incremental encoders with a resolution of 500 counts per turn.

The control unit is connected to an industrial computer through acquisition cards. The reading of the position using the encoder is accomplished through a PCI 1784 Advantech card, with four 32-bit quadruple AB phase encoder counters, while control actions are provided with a 12-bit, 4 channel PCI 1720 Advantech card. The program receives the set of references in the format of generalized active coordinates  $\vec{q}_{ind_{ref}}$ , coming from the solution of the inverse kinematics given the cartesian coordinates of the end-effector. This reference is sampled at a rate of 100 Hz and the positions, velocities, and control actions are stored for posterior analysis.

The controllers described in this paper are implemented in a modular way, using real-time middleware OROCOS (Open Robot Control Software) [45] combined with Robot Operating System (ROS) [46], and the C++ programming language.

For these experiments, the trajectories employed involve sinusoidal movements around the generalized coordinates  $(x_m, z_m, \theta, \psi)$ , which entail two hip flexions combined with a flexion–extension of knee [36]. Note that different trajectories will stimulate different inertial parameters; thus, mild changes are expected when trying to generalize. Although the trajectories are defined in cartesian coordinates, the sensors and actuators are directly related to joint coordinates, so the first step is to apply Inverse Kinematics to transform the trajectory from cartesian space to joint space. Likewise, the controller works in joint space, so all the performance criteria are also evaluated in joint coordinates. Since the controllers work without a force sensor to estimate the gravitational term, the experiments are performed using payloads of known weight instead of the human limb. This is very convenient to evaluate our controllers because the sensor is not needed and also because it offers much more stable measurements that allow a clearer assessment and comparison of the controllers. However, the same test can be performed with the human limb to perform exercises as explained in Section 2.1. In fact, the trajectory employed is suitable for rehabilitation purposes.

During the experiment, four pauses are introduced where the robot maintains the mobile platform in a fixed horizontal position, waiting for a payload to be placed (or removed) for 20 s before the real rehabilitation exercise is performed. Starting with no payload (stage 1), a 25 kg payload is placed at instant  $t = 100$  s (stage 2) and removed at  $t = 220$  s (stage 3). Afterwards, a 15 kg payload is placed at  $t = 340$  s (stage 4) and removed at  $t = 460$  s (stage 5). Figure 8 shows the experiment of mass addition in the real PR.



**Figure 8.** Actual robot and illustration of the stages of the experiment.

#### 4.2. Model-Based Controllers with Online Identification of the Gravitational Term

In this Section, we compare the results from estimators and error tracking for the two designed controllers and also the baseline, where the gravitational part is calculated by taking the entire set of relevant parameters as fixed.

Regarding the proposed controllers, the rest of the parameters (thresholds) are set to the default values provided in the description of the algorithm. Special attention is required on tuning the window size (set to 400, i.e., the estimation lasts approximately four seconds) and the forgetting factor (0.999) of both controllers, which offer a good tradeoff between the accuracy and timing of the estimations. Most of these constants are initially tuned by a Genetic Algorithm [47] in simulation to establish a fair comparison, and slightly modified in the actual robot.

The trajectory of the performed experiment is shown in Figure 9a for the  $q_{23}$  generalized coordinate, during the second stage, in which the 25 kg payload is on the platform. All controllers offer good accuracy, but according to Figure 9b–d, which represent the absolute tracking error of the second generalized coordinate, the adaptive and the proposed controllers outperform the raw baseline controller when the payload is placed on the mobile platform.

From these results, the adaptive and proposed controllers maintain the error below the baseline even when the payload is on the mobile platform. The baseline is more affected because the PD+G considers a constant set of relevant parameters. The results show some peaks of 4.5 mm after a mass addition. The WLS controller also presents some error peaks at the beginning of each stage (see Figure 9b), as it uses this error history to eventually perform a new estimation and improve the tracking a few seconds later.

Table 5 depicts the mean absolute error (MAE) at each stage of the experiment for the second generalized coordinate, and the last row represents the rate of variation of the control action, calculated as follows:

$$RV = \frac{1}{n} \sum_{i=2}^n |\tau_k - \tau_{k-1}| \quad (34)$$

where  $n$  is the number of samples of the trajectory, and  $\tau_k$  is the control action at instant  $k$ , applied in this analysis by the second actuator. High values of this parameter imply sudden changes of the control action.

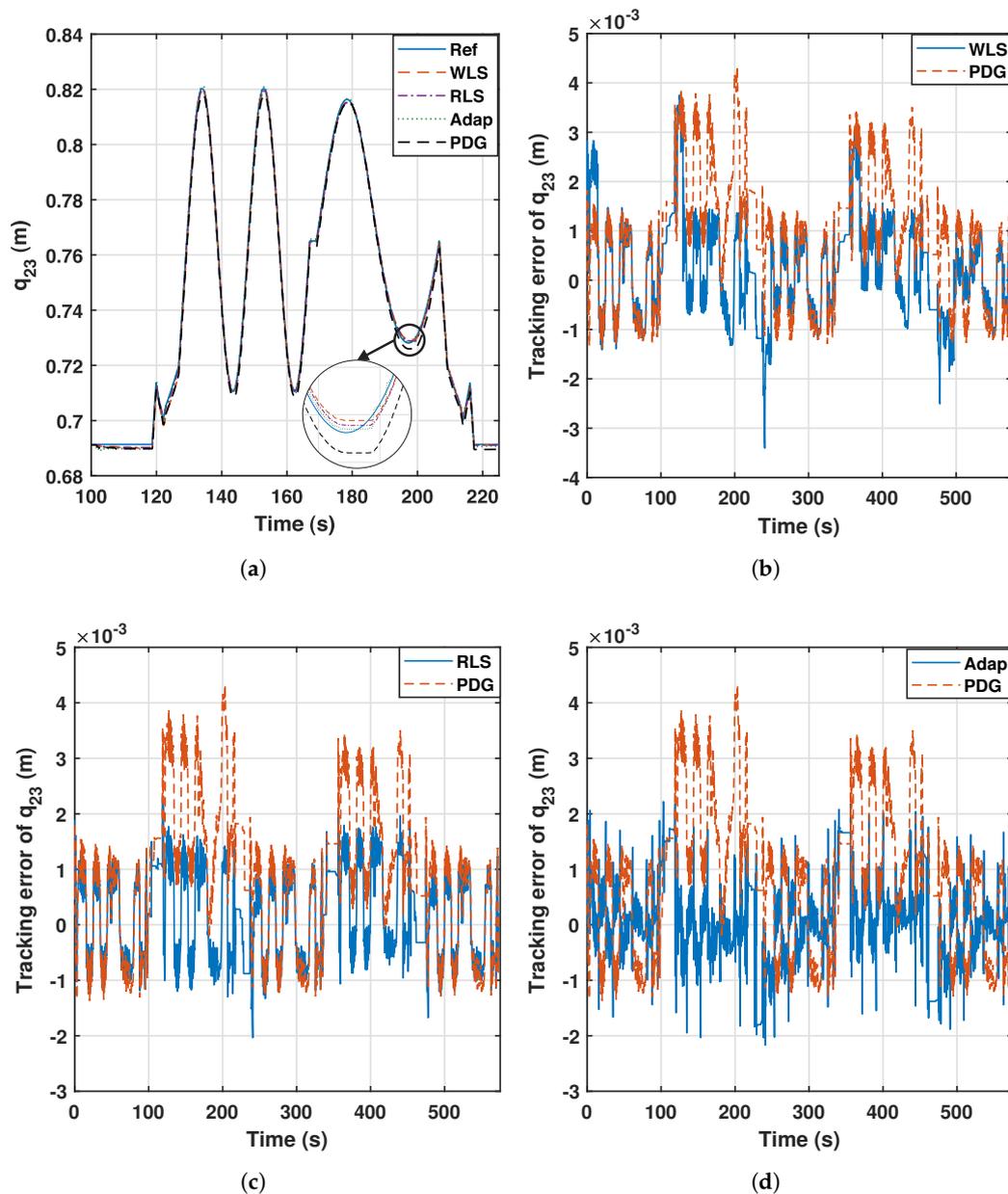


Figure 9. Trajectory tracking of  $q_{23}$  and comparison with the PD+G controller.

All controllers keep the error below 1 mm at stages 1, 3 and 5, in which no payload is used. However, the raw PD+G suffers the effect of the added mass at stages 2 and 4. This effect is compensated for by the rest of the controllers, which keep their errors low. Moreover, the WLS and RLS controllers maintain the nature of the raw PD+G in terms of the smoothness of the control action. The adaptive controller manages to lower the tracking error at the expense of a more fluctuating action according to the last row of Table 5, and this effect is illustrated in Figure 10.

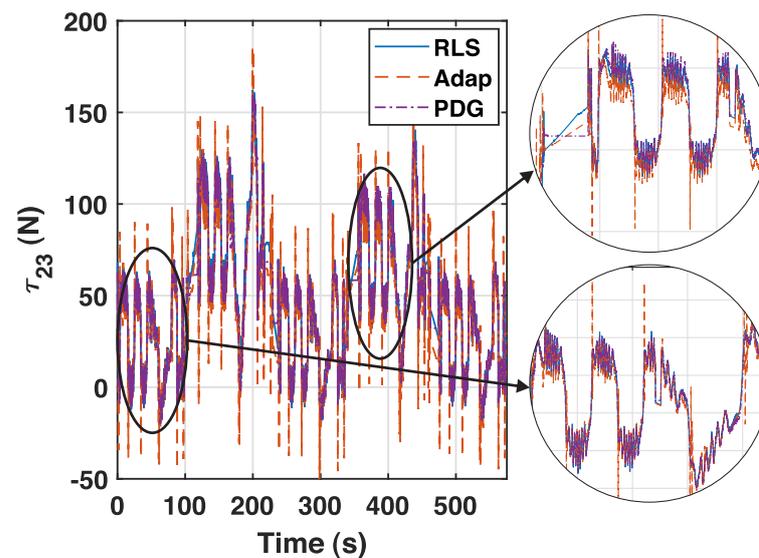


Figure 10. Control actions of the raw PDG, RLS, and adaptive controllers for the second actuator.

Table 5. Performance comparison of the tracking of the second generalized coordinate.

Parameter	PDG	WLS	RLS	Adapt.
MAE of tracking of $q_{23}$ in stage 1 (mm)	0.85	0.98	0.81	0.32
MAE of tracking of $q_{23}$ in stage 2 (mm)	2.10	1.19	0.94	0.34
MAE of tracking of $q_{23}$ in stage 3 (mm)	0.85	0.85	0.78	0.32
MAE of tracking of $q_{23}$ in stage 4 (mm)	1.75	1.10	0.90	0.36
MAE of tracking of $q_{23}$ in stage 5 (mm)	0.83	0.86	0.77	0.30
Overall MAE of tracking of $q_{23}$ (mm)	1.27	1.00	0.84	0.33
Overall std error of tracking of $q_{23}$ (mm)	1.32	1.18	0.91	0.45
RV of control action (N)	0.37	0.41	0.39	0.58

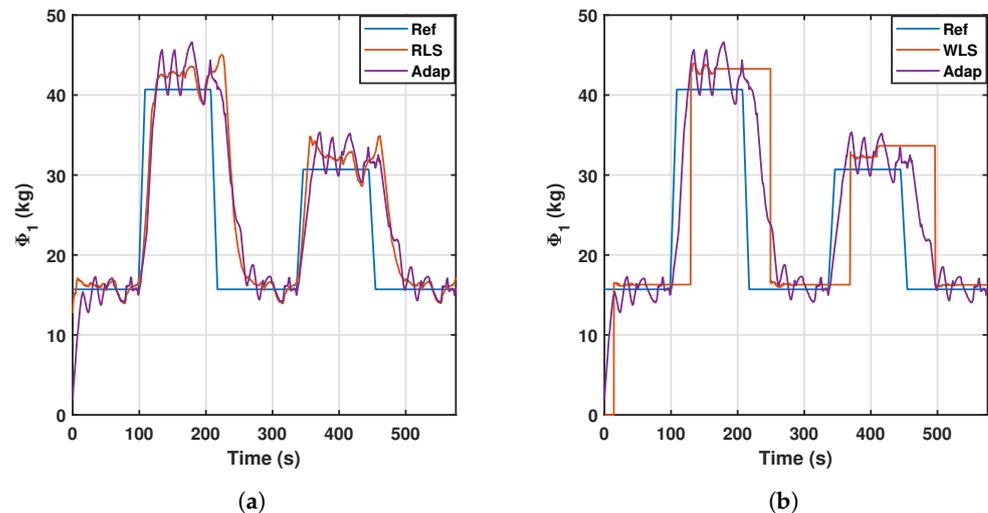
The adaptive controller presents several peaks along the trajectory, while the new controllers remain approximately as smooth as the baseline. This is essentially because both proposed controllers share the fundamental structure of the raw PD+G, while the regression of the adaptive controller is performed via sliding modes. Moreover, the new controllers with online estimator are easier to tune than the adaptive controller, as only one parameter with intuitive meaning (the window size in the case of the WLS controller, or the forgetting factor in the RLS controller) is required in the controller design.

Another important variable to analyze is the estimation of the first relevant parameter, which is illustrated in Figure 11 for the adaptive, WLS, and RLS controllers. The figure shows the effect of both controllers quite well. Starting from a zero-payload reference point of 15.6 kg (according to the first row of Table 4), the payload is changed at 20-second intervals, and then remains constant for 100 s. The window-based controller performs very few estimations; thus, it is suitable for situations where the estimation is intended to be constant until a noticeable change occurs during operation.

However, for dynamic monitoring of the relevant parameter, it would be more interesting to use adaptive or recursive least squares identification, which are capable of updating the relevant parameter immediately, at the expense of a less stable signal.

Rehabilitation exercises can be classified in (i) passive exercises, in which the patient does not exert any effort upon the rehabilitation agent, in this case the mobile platform of the PR. Here, the gravitational term stays more stable and includes the weight of the patient's leg since no extra force is being applied. In contrast, for (ii) active exercises, the patient must exert extra forces on the mobile platform and the gravitational term is expected to dynamically change during the exercise. The choice of algorithm could be based for instance on this classification. In this regard, a WLS controller would be more suitable for

passive exercises, to make very few estimations and only if the gravitational force severely changes, while a RLS controller can perform better in active exercises to catch the dynamic behavior of the changing forces.



**Figure 11.** WLS (a) and RLS (b) estimation of the first relevant parameter. Comparison with the adaptive controller.

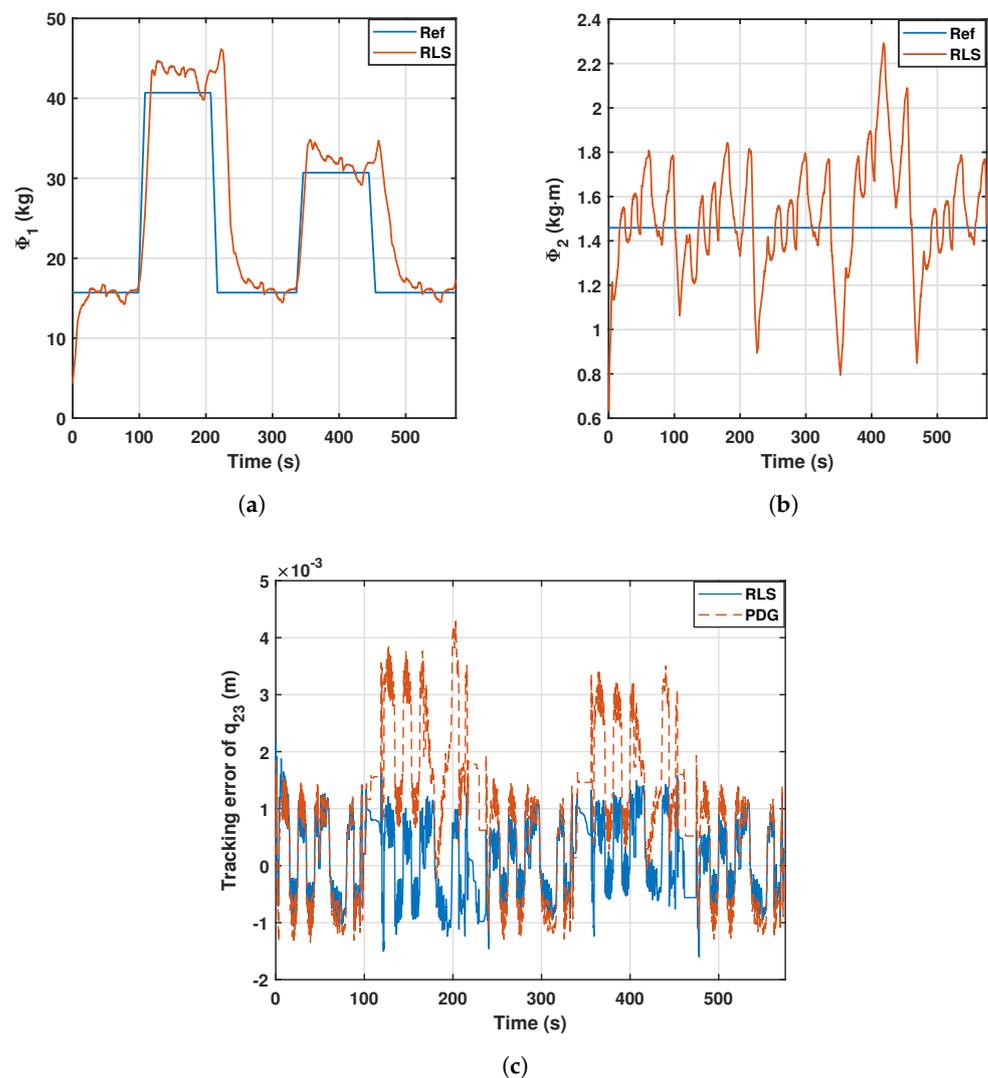
Table 6 shows the errors of estimation of the first relevant parameter and the rate of variation with the three controllers (WLS, RLS, and adaptive), and reflects the previous statement: WLS presents a greater error but lower rate of variation, while RLS has a lower error at the expense of an increment in the rate of variation. Overall, RLS outperforms the adaptive controller in the estimation task.

**Table 6.** Performance comparison in the estimation of the first relevant parameter.

Parameter	WLS	RLS	Adapt.
MAE of estimation of $\Phi_1$ in stage 1 (kg)	0.61	0.65	0.98
MAE of estimation of $\Phi_1$ in stage 2 (kg)	6.07	1.96	2.90
MAE of estimation of $\Phi_1$ in stage 3 (kg)	0.57	0.83	1.03
MAE of estimation of $\Phi_1$ in stage 4 (kg)	4.66	1.71	2.18
MAE of estimation of $\Phi_1$ in stage 5 (kg)	0.97	0.78	0.90
Overall MAE of estimation of $\Phi_1$ (kg)	3.27	1.59	2.02
RV of estimation of $\Phi_1$ ( $\text{kg} \cdot 10^{-2}$ )	0.31	0.38	0.56

It is interesting to note that all of the controllers approximately converge at the true value when no payload is present. However, the error increases for all of them when the payload is placed on the platform. There are two reasons for this mismatch. Firstly, the standard relevant parameters of Table 4 are estimated using data from a set of various trajectories distinct from the one tested in this experiment; thus, inertial parameters are excited in different ways. Another fact arises when the payload is placed on the platform, as not only is the total mass (represented by the first parameter) increased, but also other inertial properties (centers of mass, moments, etc.) that present a lower contribution to the dynamics are affected, altering, to a certain extent, the value of the other relevant parameters, which are forced to remain constant. The estimation of the first relevant parameter absorbs this effect, which is indeed a good way to avoid reflecting these discrepancies in the tracking error. A recording of the experiment using the RLS controller to estimate the first parameter is available in [https://imbio3r.ai2.upv.es/nuevo\\_video/hybrid-control-identification-one-parameter-high-speed-high-resolution-video-summary](https://imbio3r.ai2.upv.es/nuevo_video/hybrid-control-identification-one-parameter-high-speed-high-resolution-video-summary) (accessed on 22 January 2023).

The last aspect to discuss is the performance of the controllers when trying to estimate a second relevant parameter. In this regard, an additional experiment is conducted using the recursive least squares controller, which is shown in Figure 12. In this experiment, the payload positioning follows the sequence of the previous experiments. However, a second relevant parameter is estimated, which remains constant. The results show that the tracking of the first relevant parameter remains accurate, and the second relevant parameter rapidly agrees with the true value from the beginning of the experiment. The tracking error does not experience any substantial change, and its overall mean absolute value for the second generalized coordinate is 0.80, which is similar to those presented in Table 5. This experiment shows the generalization capabilities of the proposed controllers and can be viewed in [https://imbio3r.ai2.upv.es/nuevo\\_video/hybrid-control-identification-two-parameters-high-speed-high-resolution-video-summary](https://imbio3r.ai2.upv.es/nuevo_video/hybrid-control-identification-two-parameters-high-speed-high-resolution-video-summary) (accessed on 22 January 2023).



**Figure 12.** Estimation of the first (a) and second (b) relevant parameters, and error (c) of the second joint with the RLS controller.

## 5. Discussion

This study has proposed two new model-based controllers that estimate certain relevant parameters online involving the PR dynamics. The main gravitational and friction forces have been described, in contrast to the negligible inertial and centrifugal forces for knee rehabilitation and diagnostic tasks. After applying an offline SVD procedure, a subset of 15 gravitational base parameters with a physical sense were identified. These are reduced to a subset of eight relevant parameters by selecting the 95% most dynamic

influential parameters. The first relevant parameter includes the mass of the PR's mobile platform, i.e., the patient's leg mass can be identified without a force sensor.

The resulting estimation problem becomes straightforward and more robust. Moreover, instructions were embedded into the controller to perform both identification and control tasks simultaneously in rehabilitation tasks. In the literature, some adaptive controllers based on sliding modes have been proposed. However, the proposed approach involves a linear regression, exploiting the linearity property that ties the base parameters and the gravitational forces. The proposed controllers use window-based least squares and recursive least squares, which constitute a new way of performing online identification of a reduced model governed by certain relevant parameters of a PR. To the best of the authors' knowledge, no previous research has been carried out exploiting the combination of extracting the relevant parameters of a PR to obtain a well-conditioned system and performing a regression of them online in a real experiment in order to (i) keep the tracking error low even during positional changes and (ii) be able to estimate the gravitational force coming from the environment (e.g., the patient's leg).

This estimation task is ensured to be well conditioned due to the model reduction, and the results demonstrate this by providing outcomes comparable to an adaptive controller with the advantage of easier and more intuitive tuning of its parameters. Moreover, their inherent PD+G structure makes them smoother than the adaptive controller in terms of the control action. The window-based identifier provides a more stable signal at the expense of a slower and intermittent estimation, while the recursive least squares identifier responds dynamically. Moreover, the generalization capabilities were verified when trying to estimate two relevant parameters simultaneously. A ranking of the controllers for each performance criterion evaluated previously is depicted in Table 7, from 1 being the best to 3 being the worst.

**Table 7.** Ranking of performance for each criterion using the analyzed controllers.

Criterion	WLS	RLS	Adapt.
Position tracking	3	2	1
RV of the control action	2	1	3
Estimation of $\Phi$	3	1	2
RV of the estimation of $\Phi$	1	2	3

In this study, we consider that gravitational terms were expected to change over time; however, they are not the only parameters expected to change over time. Indeed, actuators are prone to deterioration, and friction forces at the joints can change. In future contributions, the proposed identification method will be extended to consider friction parameters, actuator dynamics, or both sets of parameters simultaneously. Since a formal evaluation of the controllers has been provided, the system can also be applied in experiments using a human limb with little or no modification of the algorithms.

**Author Contributions:** Conceptualization, Á.V. and V.M.; methodology, R.J.E. and J.L.P.; software, R.J.E. and J.L.P.; validation, V.M., Á.V. and M.D.-R.; writing, R.J.E. and J.L.P.; visualization, V.M., Á.V. and M.D.-R.; supervision, V.M. and Á.V.; project administration, Á.V.; funding acquisition, V.M. and Á.V. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was partially funded by Fondo Europeo de Desarrollo Regional (PID2021-125694OB-I00), cofounded by Vicerrectorado de Investigación de la Universitat Politècnica de València (PAID-11-21) and by Ministerio de Universidades, Gobierno de España (FPU18/05105).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data sharing not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

### List of Symbols

$\vec{q}_{ind}$	Independent coordinates
$\vec{q}_s$	Secondary coordinates
$\vec{q}$	Generalized coordinates
$M$	Mass matrix
$Q_{cyc}$	Matrix of inertial forces related to Coriolis and Centrifugal terms
$\vec{Q}_{grav}$	Vector of generalized gravitational forces
$\vec{Q}_{act}$	Vector of generalized active forces
$J_q$	Constraint Jacobian matrix
$\vec{\lambda}$	Vector of Lagrange multipliers
$R^*$	Orthogonal complement
$\vec{\tau}$	Forces applied by the actuated joints
$\vec{F}_f$	Friction forces
$\vec{G}$	Gravitational component in the reduced dynamic model
$\vec{\Phi}_{rb}$	Rigid body parameters
$\vec{\Phi}_f$	Friction parameters
$\vec{\Phi}_G$	Gravitational parameters
$K_{rb}$	Matrix of linear relationship between the rigid body parameters and $\vec{\tau}$
$K_f$	Matrix of linear relationship between the friction parameters and $\vec{F}_f$
$K_G$	Matrix of linear relationship between the gravitational parameters and $\vec{G}$
$W_G$	Stacked observations of $K_G$ for different configurations
$\vec{T}$	Stacked observations of $\vec{\tau} - \vec{F}_f$ for different configurations
$\vec{\Phi}_{base}$	Base parameters obtained from linear combinations of $\vec{\Phi}_G$
$K_{base}$	Matrix of linear relationships between the base parameters and $\vec{\tau}$
$W_{base}$	Stacked observations of $K_{base}$ for different configurations
$\vec{\Phi}_{rel}$	Subset of base parameters selected because of their dynamic influence
$K_{rel}$	Matrix of linear relationships between the relevant parameters and $\vec{G}$
$K_E$	Subset of $K_{rel}$ containing the identified parameters
$K_{NE}$	Subset of $K_{rel}$ containing the non-identified parameters
$\vec{\theta}_E$	Subset of $\vec{\Phi}_{rel}$ containing the identified parameters
$\vec{\theta}_{NE}$	Subset of $\vec{\Phi}_{rel}$ containing the non-identified parameters
$\vec{\hat{\theta}}$	Estimation of $\vec{\theta}_E$ performed by the controller
$\vec{G}_{NE}$	Non-identified (fixed) part of the gravitational term
$\vec{G}_E$	Final gravitational effect due to the estimated relevant parameters
$W_E$	Stacked observations of $K_E$ for different configurations
$\vec{T}_E$	Stacked observations of exerted forces discounting the fixed non-identified gravity contribution and friction
$Y$	Regressor matrix in the adaptive controller
$K_p, K_d$	Proportional and derivative gains of the PD controller
$\Gamma_0, \lambda_1$	Matrix and scalar defining the dynamics of the adaptive controller
$\vec{e}$	Tracking error
$S$	Number of samples for the WLS method
$et$	Error threshold for the WLS method
$ct$	Convergence threshold for the WLS method
$nt$	Condition number threshold for the WLS method
$qpt$	Joint velocity threshold for the WLS method
$\Sigma$	Covariance matrix for the RLS method
$\Lambda$	A priori covariance matrix for the RLS method
$\lambda_f$	Forgetting factor for the RLS method
$vt$	Variance threshold for the RLS method
$V$	Covariance matrix of the noise for the RLS method
$RV$	Rate of variation, applied to control action or parameter estimation

## References

1. Patel, Y.; George, P. Parallel Manipulators Applications—A Survey. *Mod. Mech. Eng.* **2012**, *2*, 57–64. [[CrossRef](#)]
2. Xie, S. Advanced robotics for medical rehabilitation. *Springer Tracts Adv. Robot.* **2016**, *108*, 357. [[CrossRef](#)]
3. Beasley, R.A. Medical robots: Current systems and research directions. *J. Robot.* **2012**, *2012*, 401613. [[CrossRef](#)]
4. Díaz, I.; Gil, J.J.; Sánchez, E. Lower-limb robotic rehabilitation: Literature review and challenges. *J. Robot.* **2011**, *2011*, 759764. [[CrossRef](#)]
5. Hesse, S.; Uhlenbrock, D. A mechanized gait trainer for restoration of gait. *J. Rehabil. Res. Dev.* **2000**, *37*, 701–708.
6. Sui, P.; Yao, L.; Lin, Z.; Yan, H.; Dai, J.S. Analysis and synthesis of ankle motion and rehabilitation robots. In Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 19–23 December 2009; pp. 2533–2538. [[CrossRef](#)]
7. Brahmia, A.; Kelaiaia, R. Design of a Human Knee Reeducation Mechanism. *Acta Univ. Sapientiae Electr. Mech. Eng.* **2019**, *11*, 42–53. [[CrossRef](#)]
8. Pisla, D.; Nadas, I.; Tucan, P.; Albert, S.; Carbone, G.; Antal, T.; Banica, A.; Gherman, B. Development of a Control System and Functional Validation of a Parallel Robot for Lower Limb Rehabilitation. *Actuators* **2021**, *10*, 277. [[CrossRef](#)]
9. Llopis-Albert, C.; Valero, F.; Mata, V.; Pulloquina, J.; Zamora-Ortiz, P.; Escarabajal, R. Optimal Reconfiguration of a Parallel Robot for Forward Singularities Avoidance in Rehabilitation Therapies. A Comparison via Different Optimization Methods. *Sustainability* **2020**, *12*, 5803. [[CrossRef](#)]
10. Brahmia, A.; Kelaiaia, R.; Chemori, A.; Company, O. On Robust Mechanical Design of a PAR2 Delta-Like Parallel Kinematic Manipulator. *J. Mech. Robot.* **2021**, *14*, 011001. [[CrossRef](#)]
11. Díaz-Rodríguez, M.; Araujo-Gómez, P.; González-Estrada, O.A. Performance Index for Dimensional Synthesis of Robots for Specific Tasks. *Robotics* **2022**, *11*, 51. [[CrossRef](#)]
12. Siciliano, B.; Sciavicco, L.; Villani, L.; Oriolo, G. *Robotics: Modelling, Planning and Control*; Springer Science & Business Media: London, UK, 2010. [[CrossRef](#)]
13. Wu, F.X.; Zhang, W.J.; Li, Q.; Ouyang, P.R. Integrated design and PD control of high-speed closed-loop mechanisms. *J. Dyn. Sys. Meas. Control* **2002**, *124*, 522–528. [[CrossRef](#)]
14. An, C.H.; Atkeson, C.G.; Hollerbach, J.M. Model-based control of a robot manipulator. *Choice Rev. Online* **1988**, *26*, 1542. [[CrossRef](#)]
15. Paccot, F.; Andreff, N.; Martinet, P. A review on the dynamic control of parallel kinematic machines: Theory and experiments. *Int. J. Robot. Res.* **2009**, *28*, 395–416. [[CrossRef](#)]
16. Piltan, F.; Mirzaei, M.; Shahriari, F.; Nazari, I.; Emamzadeh, S. Design baseline computed torque controller. *Int. J. Eng.* **2012**, *6*, 129–141.
17. Sharifzadeh, M.; Tale Masouleh, M.; Kalhor, A.; Shahverdi, P. An experimental dynamic identification & control of an overconstrained 3-DOF parallel mechanism in presence of variable friction and feedback delay. *Robot. Auton. Syst.* **2018**, *102*, 27–43. [[CrossRef](#)]
18. Spong, M.W.; Vidyasagar, M. *Robot Dynamics and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2008.
19. Abdallah, C.; Dorato, P.; Jamshidi, M.; Dawson, D. Survey of Robust Control for Rigid Robots. *IEEE Control Syst.* **1991**, *11*, 24–30. [[CrossRef](#)]
20. Kelly, R.; Davila, V.S.; Perez, J.A.L. *Control of Robot Manipulators in Joint Space*; Springer Science & Business Media: London, UK, 2006. [[CrossRef](#)]
21. Gautier, M. Numerical calculation of the base inertial parameters of robots. *J. Robot. Syst.* **1991**, *8*, 485–506. [[CrossRef](#)]
22. Díaz-Rodríguez, M.; Mata, V.; Farhat, N.; Provenzano, S. Identifiability of the dynamic parameters of a class of parallel robots in the presence of measurement noise and modeling discrepancy. *Mech. Based Des. Struct. Mach.* **2008**, *36*, 478–498. [[CrossRef](#)]
23. Gao, G.; Sun, G.; Na, J.; Guo, Y.; Wu, X. Structural parameter identification for 6 DOF industrial robots. *Mech. Syst. Signal Process.* **2018**, *113*, 145–155. [[CrossRef](#)]
24. Diaz-Rodriguez, M.; Valera, A.; Mata, V.; Valles, M. Model-based control of a 3-DOF parallel robot based on identified relevant parameters. *IEEE/ASME Trans. Mechatron.* **2012**, *18*, 1737–1744. [[CrossRef](#)]
25. Cazalilla, J.; Vallés, M.; Mata, V.; Díaz-Rodríguez, M.; Valera, A. Adaptive control of a 3-DOF parallel manipulator considering payload handling and relevant parameter models. *Robot. Comput.-Integr. Manuf.* **2014**, *30*, 468–477. [[CrossRef](#)]
26. Duan, J.; Liu, Z.; Bin, Y.; Cui, K.; Dai, Z. Payload Identification and Gravity/Inertial Compensation for Six-Dimensional Force/Torque Sensor with a Fast and Robust Trajectory Design Approach. *Sensors* **2022**, *22*, 439. [[CrossRef](#)]
27. Kim, T.H.; Kim, Y.; Kwak, T.; Kanno, M. Metaheuristic Identification for an Analytic Dynamic Model of a Delta Robot with Experimental Verification. *Actuators* **2022**, *11*, 352. [[CrossRef](#)]
28. Kapandji, A.I. *Fisiología Articular T2: Miembro Inferior*; Editorial Médica Panamericana: Madrid, Spain, 2012.
29. Wiertsema, S.; van Hooff, H.; Mighelsen, L.; Steultjens, M. Reliability of the KT1000 arthrometer and the Lachman test in patients with an ACL rupture. *Knee* **2008**, *15*, 107–110. [[CrossRef](#)] [[PubMed](#)]
30. Lopomo, N.; Zaffagnini, S.; Signorelli, C.; Bignozzi, S.; Giordano, G.; Marcheggiani Muccioli, G.M.; Visani, A. An original clinical methodology for non-invasive assessment of pivot-shift test. *Comput. Methods Biomech. Biomed. Eng.* **2012**, *15*, 1323–1328. [[CrossRef](#)]

31. Vallés, M.; Araujo-Gómez, P.; Mata, V.; Valera, A.; Díaz-Rodríguez, M.; Page, Á.; Farhat, N.M. Mechatronic design, experimental setup, and control architecture design of a novel 4 DoF parallel manipulator. *Mech. Based Des. Struct. Mach.* **2018**, *46*, 425–439. [[CrossRef](#)]
32. Araujo-Gómez, P.; Mata, V.; Díaz-Rodríguez, M.; Valera, A.; Page, A. Design and kinematic analysis of a novel 3UPS/RPU parallel kinematic mechanism with 2T2R motion for knee diagnosis and rehabilitation tasks. *J. Mech. Robot.* **2017**, *9*, 061004. [[CrossRef](#)]
33. Wehage, R.A.; Haug, E.J. Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. *J. Mech. Des. Trans. ASME* **1982**, *104*, 247–255. [[CrossRef](#)]
34. Díaz-Rodríguez, M.; Mata, V.; Valera, Á.; Page, Á. A methodology for dynamic parameters identification of 3-DOF parallel robots in terms of relevant parameters. *Mech. Mach. Theory* **2010**, *45*, 1337–1356. [[CrossRef](#)]
35. Liu, Y.F.; Li, J.; Zhang, Z.M.; Hu, X.H.; Zhang, W.J. Experimental comparison of five friction models on the same test-bed of the micro stick-slip motion system. *Mech. Sci.* **2015**, *6*, 15–28. [[CrossRef](#)]
36. Pulloquina, J.L.; Mata, V.; Valera, Á.; Zamora-Ortiz, P.; Díaz-Rodríguez, M.; Zambrano, I. Experimental analysis of Type II singularities and assembly change points in a 3UPS+ RPU parallel robot. *Mech. Mach. Theory* **2021**, *158*, 104242. [[CrossRef](#)]
37. Díaz Rodríguez, M.Á. Identificación de Parámetros Dinámicos de Robots Paralelos Basada en un Conjunto de Parámetros Significativos. Ph.D. Thesis, Universitat Politècnica de València, Valencia, Spain, 2009. [[CrossRef](#)]
38. Ros, J.; Iriarte, X.; Mata, V. 3D inertia transfer concept and symbolic determination of the base inertial parameters. *Mech. Mach. Theory* **2012**, *49*, 284–297. [[CrossRef](#)]
39. Valero, F.; Díaz-Rodríguez, M.; Vallés, M.; Besa, A.; Bernabéu, E.; Valera, Á. Reconfiguration of a parallel kinematic manipulator with 2T2R motions for avoiding singularities through minimizing actuator forces. *Mechatronics* **2020**, *69*, 102382. [[CrossRef](#)]
40. Bayard, D.S.; Went, J.T. New class of control laws for robotic manipulators part 2. Adaptive case. *Int. J. Control* **1988**, *47*, 1387–1406. [[CrossRef](#)]
41. Benveniste, A. Design of adaptive algorithms for the tracking of time-varying systems. *Int. J. Adapt. Control. Signal Process.* **1987**, *1*, 3–29. [[CrossRef](#)]
42. Eleftheriou, E.; Falconer, D. Tracking properties and steady-state performance of RLS adaptive filter algorithms. *IEEE Trans. Acoust. Speech Signal Process.* **1986**, *34*, 1097–1110. [[CrossRef](#)]
43. Peterka, V. *Bayesian Approach To System Identification*; IFAC: New York, NY, USA, 1981; pp. 239–304. [[CrossRef](#)]
44. Guo, L.; Ljung, L.; Priouret, P. Performance analysis of the forgetting factor RLS algorithm. *Int. J. Adapt. Control. Signal Process.* **1993**, *7*, 525–537. [[CrossRef](#)]
45. Bruyninckx, H. Open robot control software: The OROCOS project. In Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, Republic of Korea, 21–26 May 2001; Volume 3, pp. 2523–2528. [[CrossRef](#)]
46. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: an open-source Robot Operating System. In Proceedings of the ICRA Workshop on Open Source Software, Kobe, Japan, 12–17 May 2009; Volume 3, p. 5.
47. Reynoso-Meza, G.; Garcia-Nieto, S.; Sanchis, J.; Blasco, F.X. Controller tuning by means of multi-objective optimization algorithms: A global tuning framework. *IEEE Trans. Control. Syst. Technol.* **2012**, *21*, 445–458. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.