

## Parallel robot for knee rehabilitation: Reduced order dynamic linear model, mechanical assembly and control system architecture

Armas Elizabeth<sup>1</sup>, Cruz Patricio<sup>2</sup>, Soto Ricardo<sup>1</sup> Lema Henry<sup>1</sup>, Granja Mario<sup>1</sup>, Rosales Andrés<sup>2</sup>,  
Zambrano Iván<sup>1</sup>

<sup>1</sup> Mechanics Department, Escuela Politécnica Nacional

<sup>2</sup> Automation and Industrial Control Department, Escuela Politécnica Nacional

---

### ABSTRACT

---

In this work we present the development of a dynamic linear model of a 3UPS+1RPU parallel robot for knee rehabilitation, which allows the reduction of the error with respect to simulation carried out based on its non-linear model. Furthermore, the design and implementation of a control algorithm in a real robot is detailed, for which a dynamic linear model has been developed based on inertial parameters including a friction model in Coulomb and viscosity parameters. Subsequently, the linear model has reduced applying the numerical method of decomposition into singular values, resulting in a model expressed as function of base parameters. This method uses a base parameter identification path obtained by finite Fourier series. This path is optimized through minimization algorithms restricted by distance, velocity and acceleration of the linear actuators of the robot, as well as the working space of its spherical joints. Then, the compatibility level of the reduced dynamic model is quantified by estimating mean square error determined between the generalized forces of the independent joints obtained from the model and compared with those resulting from simulations performed in Adams/View software for a trajectory obtained by finite Fourier series. Afterwards, mechanical components involved in the implementation of the prototype are selected and the control system of its actuators is designed. Finally, tests are performed in a laboratory through photogrammetry equipment, in order to validate joints mobility in the robot and study its performance, for this task defined trajectories based on criteria of a physiotherapist are used.

---

**Keywords:** Linear dynamic model, base parameters, parallel robot, compatibility analysis, singular values decomposition

---

### *Corresponding Author:*

Iván Zambrano  
Mechanics Department  
Escuela Politécnica Nacional  
Ladrón de Guevara E11-2530  
E-mail: ivan.zambrano@epn.edu.ec

---

### 1. Introduction

Muscle weakness, loss of motion range, and decreased strength affect patients who require physical rehabilitation in their lower limbs, particularly their knees. These conditions reduce their chances of maintaining an independent life. Portable mechatronics systems that can be used by these patients in their homes are of particular interest in the scenario described above. These robotic support platforms present the potential of becoming tools that ease recovery and rehabilitation processes by reducing treatments costs and providing some level of independence for the patients [1], [2]. For example, support platforms have been proposed for active and passive exercises, systems capable of learning and reproducing specific movements, [3], and even robots capable of adapting to the physiological conditions of patients.

Currently, in the clinical rehabilitation area, a particular type of platforms being used are parallel robots [4], since they allow the development of active and passive exercises under minimal supervision by the physiotherapist [4]. These robots are particularly useful for performing fast, accurate and high load capacity movements [5], characteristics that are required in knee mobility recovery activities. However, the complexity of the dynamic model of parallel robots in contrast with serial ones makes it difficult to design and implement advanced controllers that can exploit their advantage of fast and precise movements. For example, in the case of cable-suspended parallel robots, their inverse dynamic analysis is performed using the Lagrange formulation combined with the elastic model of the cables [6], [7].

Dealing with non-linear complex models demands high computational resources [8]. To overcome this limitation, one option is to use reduced order models or simplified linear dynamic models. For instance, a reduced model is defined in [9] which is based on the search of the coefficients of an approximate polynomial equation that represents the robot dynamics and as it is applied to a Cartesian translational manipulator. Reference [10] reduces the model based on relevant parameters obtained in a process that contemplates the statistical importance and physical viability of these parameters. Moreover, the identification of these basic parameters, allows to obtain reduced dynamic linear model with a behavior of high similarity in relation to the real mechanism [11]. This identification is of great relevance in simulating the robot and designing control algorithms. These methodologies for obtaining simplified models have been successfully tested on parallel robots. However, the case study platforms for these works have not been developed for assisting knee rehabilitation.

In our previous work, we presented the design and modelling of a parallel robot for knee rehabilitation [12]. The proposed robotic system consists of four prismatic links, three of them are linked by universal and spherical joints, and the remaining link is coupled by revolution and universal joints. Therefore, its configuration is a 3UPS (Universal, Parallel, Spherical joints) + 1RPU (Revolute, Parallel, Universal joints). A similar platform is proposed in [13], but we extended its kinematic analysis to the derivation of a non-linear dynamic model by applying the Gibbs Appell methodology. In [12], we established the level of compatibility of the proposed dynamic model with respect to a virtual robot simulated in Adams/View software. Thus, we compared the forces computed using the model with the ones obtained from the virtual robot. For the tests, we considered that the parallel robot must perform movements used in knee rehabilitation such as: hip flexion, knee flexion-extension, knee internal-external rotation and ankle flexion-extension. This validation revealed that the proposed non-linear dynamic model presents an error of about 20% with respect to the simulated dynamic behavior of the parallel robot. This error is mainly due to the fact that dynamic, inertial and friction parameters existing that exist in the robot mechanisms were not considered in the modeling process.

To reduce the error reported in [12] for the non-linear dynamic model of the 3UPS+1RPU parallel robot for knee rehabilitation model, we introduce here a reduced order dynamic linear model for this platform. Firstly, we derive a dynamic linear model based on inertial parameters of the bars that conform the system such as inertia, mass, and center of gravity. These parameters are included in order to reduce the estimation error in relation to the simulation of the non-linear model of the parallel robot. Also, the model includes friction which is a natural characteristic in the joints of the robot and it varies in every mechanical system depending on its operating time. Although considering inertial and friction parameters improves the dynamic performance, the model complexity is increased demanding a high computational simulation cost. To overcome this limitation, we simplify the model by applying the approach proposed in [8]. Thus, we identify from the whole set of dynamic parameters those that have a higher physical influence in the dynamic performance of the robot. From the simulation tests, the reduced order dynamic linear model shows a behavior of high similarity in relation to the virtual mechanism in Adams/View software.

In addition to the contributions about modeling the parallel robot, we present its mechanical assembly. It consists of a mobile platform that actuates on the patient and a fixed platform that supports the whole structure. These platforms are connected with a linear actuator distributed asymmetrically to reduce the number of singularities that can be transited during the execution of a trajectory [13]. Despite this, there may be an error in the position reached by the mobile platform, which is mainly due to the looseness presented in the joints [14]. For this reason, different combinations of actuators have been analyzed in order to select the components (platforms, linear actuators and the joints: spherical, universal and revolution) to be used in the assembly and construction of the prototype. Once the platform has been assembled, we designed a basic real time controller to command the position and speed of the linear actuators of the robot. The control scheme is implemented in the middleware OROCOS which runs under ROS [4]. A vision system (Kinescan) [15] and prescribed movements for knee rehabilitation are tested to analyze and validate the platform motion control.

The rest of this paper is organized as follows. The modelling process of the robot, construction of the prototype and control architecture are detailed in Section 2, then simulation and real test results are described in Section 3. The conclusions from the experiments are presented in Section 4.

## 2. Method

### 2.1. Parallel robot modelling

A parallel robot, Fig. 1(a), is a set of closed kinematic chains in which its end effector (moving platform) is connected to the base (fixed platform) through several open kinematic chains. It is well known that in parallel robots, unlike serial robots, solving the inverse kinematic problem is simpler than solving the direct kinematic problem [16]. Moreover, determining the value of the generalized variables from the position and orientation of the final effector in parallel robots is a simple problem since it can be divided by analyzing the inverse kinematic of each chain that links the base and the mobile platforms.

The 3UPS+RPU parallel robot has four degrees of freedom (DOF): two for translation and two for rotation. Fig. 1(a) shows that the mobile platform can be moved along the  $X_m$  and  $Z_m$  axis, and there are two rotation movements, one around the  $Z_m$  and the other around the  $Y_m$  axis. The translation constraint on the  $Y_m$  axis and the rotation around  $X_m$  axis are defined by the central leg of the robot RPU (Revolute Parallel Universal). The robot sketch presented in the Fig. 1(b) shows how the platform is going to be used.

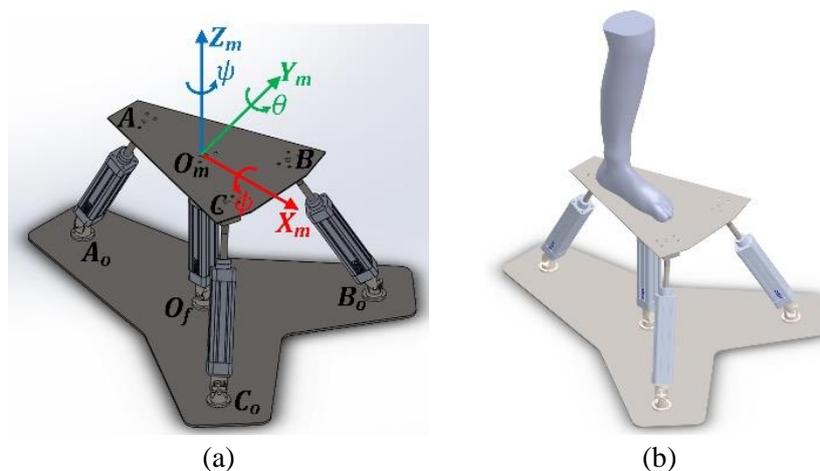


Figure 1. 3UPS + 1RPU Configuration (a). Robot sketch (b)

### 2.1.1. Basic parameters

In parallel robots there is no general equation to establish their dynamics, usually closed kinematic chains are "cut" in certain joints to study them as an equivalent set of open chains, where the internal forces that are had in the cuts in the joints are introduced in the dynamic modeling through the Jacobian constraint matrix. The open chain system model can be applied in a parallel robot, using Lagrange-Euler, Kane or Gibbs-Appell. The general dynamic model of an ideal manipulator robot [10] consisting of  $n$  bodies and expressed in independent coordinates, is defined by:

$$\vec{\tau} = D(\vec{q})\ddot{\vec{q}} + \vec{C}(\vec{q}, \dot{\vec{q}}) + \vec{G}(\vec{q}, \dot{\vec{q}}) + J^T(\vec{q})\vec{F} \quad (1)$$

Where:  $\vec{\tau}$  is the forces/torques vector applied by the actuators,  $\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}$  is the position, velocity and acceleration vector of the joints,  $D(\vec{q})$  is the inertial matrix of the robot,  $\vec{C}(\vec{q}, \dot{\vec{q}})$  is the centrifugal forces and Coriolis vector,  $\vec{G}(\vec{q}, \dot{\vec{q}})$  is the gravitational forces vector,  $\vec{F}$  is the vector of external forces applied to the final effector and  $J(\vec{q})$  is the Jacobian matrix of constraints.

### 2.1.2. Kinematic position constraints

On a kinematic perspective, a parallel robot is equivalent to a set of serial robots that are restricted to follow certain paths. These restrictions are defined by a set of holonomic (not time-dependent) constraints that depend on the geometrical configuration of the robot. In general, kinematic position constraints are set like this:

$$f_i(q_1, q_2, \dots, q_n) = 0 \quad i = 1, 2, \dots, m \quad (2)$$

Where:  $q_1, q_2, \dots, q_n$  are the  $n$  generalized coordinates and  $m$  is the number of constraint equations. With the exception of over-restricted mechanisms, the number of degrees of freedom of a robot is defined by:

$$DOF = n - m \quad (3)$$

According to [12], the position restriction equations of the parallel robot 3UPS+1RPU are established vectorially by four closed kinematic chains formed by each of the legs of the parallel robot.

*Closed chains 1, 2 and 3.*

Mathematically the restriction equations for the first three kinematic chains, see Fig. 2, established with respect to the 0 coordinate system [12], because of their mechanical similarity, are defined as:

$$\begin{aligned} {}^0\vec{r}_{O_0A_0} + {}^0\vec{r}_{A_0A} &= {}^0\vec{r}_{O_0O_m} + {}^0R_m {}^m\vec{r}_{O_mA} \\ {}^0\vec{r}_{O_0B_0} + {}^0\vec{r}_{B_0B} &= {}^0\vec{r}_{O_0O_m} + {}^0R_m {}^m\vec{r}_{O_mB} \\ {}^0\vec{r}_{O_0C_0} + {}^0\vec{r}_{C_0C} &= {}^0\vec{r}_{O_0O_m} + {}^0R_m {}^m\vec{r}_{O_mC} \end{aligned} \quad (4)$$

Where:

$${}^0R_m = \begin{bmatrix} \cos(\psi) \cos(\theta) & -\sin(\psi) \cos(\theta) & \sin(\theta) \\ \sin(\psi) & \cos(\psi) & 0 \\ -\cos(\psi) \sin(\theta) & \sin(\psi) \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (5)$$

Vector  ${}^0\vec{r}_{O_0O_m}$  corresponds to the coordinates of the final effectors, like this:

$${}^0\vec{r}_{O_0O_m} = [X_m \quad Y_m \quad Z_m]^T \quad (6)$$

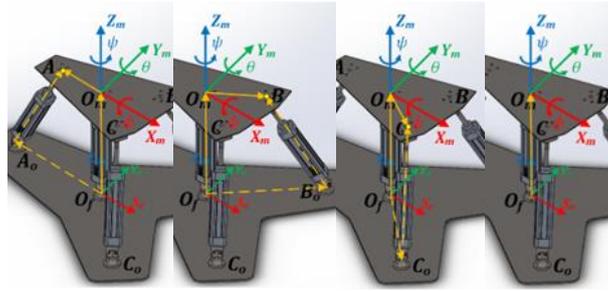


Figure 2. Closed chains 1, 2, 3 and 4

Where  $X_m$ ,  $Y_m$  and  $Z_m$  represent the position of the center of the platform on the X, Y, Z axis respectively. On the other hand,  ${}^0\vec{r}_{O_oA_o}$ ,  ${}^0\vec{r}_{O_oB_o}$ ,  ${}^0\vec{r}_{O_oC_o}$ ,  ${}^m\vec{r}_{O_mA}$ ,  ${}^m\vec{r}_{O_mB}$  and  ${}^m\vec{r}_{O_mC}$  vectors have values defined by the robot geometry and are defined in [12].

Vectors  ${}^0\vec{r}_{A_oA}$ ,  ${}^0\vec{r}_{B_oB}$  &  ${}^0\vec{r}_{C_oC}$  are determined by establishing three coordinate systems in the joints present in the first three kinematic chains. When applying the methodology of [17] the above-mentioned vectors are defined as:

$${}^0\vec{r}_{A_oA} = \begin{bmatrix} q_{13} \cos(q_{11}) \cos(q_{12}) \\ q_{13} \sin(q_{12}) \\ q_{13} \sin(q_{11}) \cos(q_{12}) \end{bmatrix} \quad (7)$$

$${}^0\vec{r}_{B_oB} = \begin{bmatrix} q_{23} \cos(q_{21}) \cos(q_{22}) \\ q_{23} \sin(q_{22}) \\ q_{23} \sin(q_{21}) \cos(q_{22}) \end{bmatrix} \quad (8)$$

$${}^0\vec{r}_{C_oC} = \begin{bmatrix} q_{33} \cos(q_{31}) \cos(q_{32}) \\ q_{33} \sin(q_{32}) \\ q_{33} \sin(q_{31}) \cos(q_{32}) \end{bmatrix} \quad (9)$$

#### Closed chains 4.

Based on [12], the fourth kinematic chain, see Fig. 2, is established by equalizing the position of the central leg (leg 4) according to the location of the joints that make it up with the coordinates of the center of the mobile platform relative to the reference system 0, i.e.:

$${}^0\vec{r}_{O_oO_m} = {}^0R_{P4_2} \cdot {}^2\vec{r}_{O_oO_m} = [X_m \quad 0 \quad Z_m]^T \quad (10)$$

The rotation matrix  ${}^0R_{P4_2}$  in modified D-H notation with the assigned reference systems is:

$${}^0R_{P4_2} = \begin{bmatrix} \sin(q_{41}) & 0 & \cos(q_{41}) \\ 0 & 1 & 0 \\ -\cos(q_{41}) & 0 & \sin(q_{41}) \end{bmatrix} \quad (11)$$

Since expressions (4) and (10) are vector type, each one has three coordinate components, then twelve restriction equations are obtained. However, in (10) Y component is zero so eleven restriction equations are obtained, i.e.  $m = 11$ . The generalized variables,  $n$  are 15 in total, which are:  $q_{11}$ ,  $q_{12}$ ,  $q_{13}$ ,  $q_{21}$ ,  $q_{22}$ ,  $q_{23}$ ,  $q_{31}$ ,  $q_{32}$ ,  $q_{33}$ ,  $q_{41}$ ,  $q_{42}$ ,  $X_m$ ,  $Z_m$ ,  $\theta$ ,  $\psi$ .

#### Kinematic velocity restrictions

The velocity restriction equations of a parallel robot are obtained using the velocity compatibility principle [11]:

$$A(\vec{q}) \cdot \dot{\vec{q}}(t) = \vec{0} \quad (12)$$

Here:  $A(\vec{q})$  is the Jacobian matrix ( $m \times n$ ) of the parallel robot and  $\dot{\vec{q}}(t)$  is the vector ( $n \times 1$ ) of generalized velocity. For the robot in analysis, the speeds at the external vertices of each kinematic chain are defined as:

$$A_{11 \times 15} = \begin{bmatrix} \psi^1 & 0 & 0 & \psi^5 \\ 0 & \psi^2 & 0 & \psi^6 \\ 0 & 0 & \psi^3 & \psi^7 \\ 0 & 0 & 0 & \psi^8 \end{bmatrix} \quad (13)$$

Where:

$${}^p\psi_i^k = \begin{bmatrix} \begin{bmatrix} {}^p\vec{z}_i^k \times {}^p\vec{r}_{O_i O_p}^k \\ {}^p\vec{z}_i^k \end{bmatrix} & \text{if the node (i) of revolution} \\ \begin{bmatrix} {}^p\vec{z}_i^k \\ \vec{0} \end{bmatrix} & \text{if the node (i) of prismatic} \end{bmatrix}$$

and

$$\dot{\vec{q}}(t) = \begin{bmatrix} \dot{q}_{13} & \dot{q}_{12} & \dot{q}_{11} & \dot{q}_{23} & \dot{q}_{22} & \dot{q}_{21} & \dot{q}_{33} & \dots \\ \dot{q}_{32} & \dot{q}_{31} & \dot{q}_{42} & \dot{q}_{41} & \dot{\psi} & \dot{\theta} & \dot{Z}_m & \dot{X}_m \end{bmatrix}^T \quad (14)$$

Deriving (12) the acceleration restriction equations are:

$$\frac{d^2 f_m(q_1, q_2, \dots, q_n)}{dx^2} = A(\vec{q}) \cdot \ddot{\vec{q}}(t) - \vec{b} = \vec{0} \quad (15)$$

Where:

$$\vec{b} = A(\vec{q}) \cdot \ddot{\vec{q}}(t) - \frac{d^2 f_m(q_1, q_2, \dots, q_n)}{dx^2} \quad (16)$$

In addition, the Jacobian matrix ( $A$ ) corresponds to (13) and the vector of generalized accelerations has the structure of (14).

Explicit equations of Gibbs Appell

Starting from Gibbs Appell expressions and using the recursive expressions established by Benimeli [18] for the partial derivatives  $\partial^i \vec{\omega}_i / \partial \dot{q}_k$  and  $\partial^i \vec{r}_{G_i}^{\ddot{}} / \partial \ddot{q}_k$ , Mata, Benimeli, Farhat & Valera [19], have developed explicit expressions for the generalized pairs  $\vec{\tau}$ , which can be: revolute (R) or prismatic (P) type:

$$\vec{\tau} = K(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) \cdot \vec{\Phi}_i \quad (17)$$

Here  $\vec{\Phi}_i$  groups the inertial parameters, the moments of inertia are considered first  $[I_{xx_j} \ I_{xy_j} \ I_{xz_j} \ I_{yy_j} \ I_{yz_j} \ I_{zz_j}]$ , then the first mass moments  $[mx_j \ my_j \ mz_j]$  and finally the masses of the links ( $m_j$ ) that conform the parallel robot. For the analyzed robot, the matrix  $K(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}})$  is defined by the four kinematic chains; in this study we have 9 links, so we define 90 inertial parameters, then we have:

$$K_{115 \times 90} = \begin{bmatrix} K_{P1} & 0_{3 \times 20} & 0_{3 \times 20} & 0_{3 \times 20} & 0_{3 \times 10} \\ 0_{3 \times 20} & K_{P2} & 0_{3 \times 20} & 0_{3 \times 20} & 0_{3 \times 10} \\ 0_{3 \times 20} & 0_{3 \times 20} & K_{P3} & 0_{3 \times 20} & 0_{3 \times 10} \\ 0_{2 \times 20} & 0_{2 \times 20} & 0_{2 \times 20} & K_{P4} & 0_{2 \times 10} \\ 0_{4 \times 20} & 0_{4 \times 20} & 0_{4 \times 20} & 0_{4 \times 20} & K_{PM} \end{bmatrix} \quad (18)$$

By applying the principle of minimum Gaussian action, the generalized forces of the independent joints, for a parallel robot, are defined as:

$$\begin{aligned}\vec{\tau}_i &= [K_i - X^T \cdot K_d] \cdot \vec{\Phi}_i \\ \vec{\tau}_i &= K \cdot \vec{\Phi}_i\end{aligned}\quad (19)$$

The  $K_i$  matrix is obtained by extracting the rows of the  $K_1$  matrix associated with the independent variables, i.e. the four linear actuators ( $q_{13}, q_{23}, q_{33}, q_{42}$ ). According to the organization of the generalized variables of (14), the expression is:

$$K_{i_{4 \times 90}} = [K_1(1,:) \quad K_1(4,:) \quad K_1(7,:) \quad K_1(10,:)]^T \quad (20)$$

The  $K_d$  matrix is related to the generalized dependent variables:

$$K_{d_{11 \times 90}} = [K_1(2:3,:) \quad K_1(5:6,:) \quad K_1(8:9,:) \quad K_1(11:15,:)]^T \quad (21)$$

### 2.1.3. Friction model

In accordance with Díaz [10], the most used friction model for identifying inertial and friction parameters in parallel 4DOF and 6DOF robots is the Coulomb and viscous model. Its frequent use is due to the fact that the generalized friction forces of independent joints, in serial robots, can be expressed in a linear manner as:

$$\vec{\tau}_{f_i} = K_f(\dot{q}_i) \cdot \vec{\Phi}_f \quad (22)$$

$$\text{With } \vec{\Phi}_f = \left[ \begin{array}{cc} F_{v_j} & F_{c_j} \\ \text{j=DOF,DOF-1,...,1} \end{array} \right]^T$$

Where  $F_{v_j}$  represents the viscosity friction parameters and  $F_{c_j}$  represents Coulomb friction parameters. For the robot in analysis, considering the active and passive joints,  $K_f(\dot{q}_i)$  is:

$$K_{1f_{15 \times 22}} = \begin{bmatrix} K_{f_{P1}} & 0_{3 \times 6} & 0_{3 \times 6} & 0_{3 \times 4} \\ 0_{3 \times 6} & K_{f_{P2}} & 0_{3 \times 6} & 0_{3 \times 4} \\ 0_{3 \times 6} & 0_{3 \times 6} & K_{f_{P3}} & 0_{3 \times 4} \\ 0_{2 \times 6} & 0_{2 \times 6} & 0_{2 \times 6} & K_{f_{P4}} \\ 0_{4 \times 6} & 0_{4 \times 6} & 0_{4 \times 6} & 0_{4 \times 4} \end{bmatrix} \quad (23)$$

Finally, to express the linear model friction of closed chain, based on the open chain model, results in:

$$\begin{aligned}\vec{\tau}_{f_i} &= [K_{f_i} - X^T \cdot K_{f_d}] \cdot \vec{\Phi}_f \\ \vec{\tau}_{f_i} &= K_f \cdot \vec{\Phi}_f\end{aligned}\quad (24)$$

In (24) the  $K_{f_i}$  matrix is obtained by extracting the rows of the  $K_{1f}$  matrix associated with the independent variables, that means the four linear actuators ( $q_{13}, q_{23}, q_{33}, q_{42}$ ):

$$K_{f_{i_{4 \times 22}}} = [K_{1f}(1,:) \quad K_{1f}(4,:) \quad K_{1f}(7,:) \quad K_{1f}(10,:)]^T \quad (25)$$

While  $K_{fd}$  relates to generalized dependent variables such as:

$$K_{fd_{11 \times 22}} = [K_{1f}(2:3, :) \ K_{1f}(5:6, :) \ K_{1f}(8:9, :) \ K_{1f}(11:15, :)]^T \quad (26)$$

#### 2.1.4. Linear dynamic model in inertial and friction parameters

The linear model in inertial and friction parameters for the parallel robot 3UPS + 1RPU is determined by combining (19) and (24), according to [11], mathematically:

$$\vec{\tau} = K^* \cdot \vec{\Phi} \quad (27)$$

Where:

$$K^* = [K_{4 \times 90} \ K_{f_{4 \times 22}}] \quad (28)$$

$$\vec{\Phi} = \begin{bmatrix} \vec{\Phi}_{i_{90 \times 1}} \\ \vec{\Phi}_{f_{22 \times 1}} \end{bmatrix} \quad (29)$$

#### 2.1.5. Identification trajectories

The identification of parameters consists in adjusting the response of the dynamic model to the real behavior of the analyzed robot. This implies that each trajectory of excitation exerted by the robot corresponds to a specific dynamic response and seeks not to excite inertial parameters that do not contribute to the dynamic response Diaz [10]. To achieve this, the excitation trajectory must be optimized, through the target function  $S$  which is the mathematical expression to be minimized in order to satisfy the constants of movements in the actuators represented by  $g$ .

$$\begin{aligned} & \text{Minimize } S(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) \\ & \text{subject to } g(\vec{q}, \dot{\vec{q}}, \ddot{\vec{q}}) \leq 0 \end{aligned} \quad (30)$$

$$k(W) = \frac{s_{max}}{s_{min}}$$

In the present work the objective function is the condition number  $k$  of a matrix, defined as the relationship between the maximum ( $s_{max}$ ) and minimum ( $s_{min}$ ) singular value of the array  $W$ ;  $k$  indicates the degree to which the measurement error is transmitted to the final result [8]. Therefore, a very high  $k$  shows that the error in the readings of position, speed and acceleration produces an even greater error in the estimation of the forces in the actuators. The parameters of the identification paths, which are intended to be optimized, are those of the Fourier series ( $q_{i0}$ ,  $a_{ij}$  &  $b_{ij}$ ) but bounded by linear and non-linear constraints, see Fig. 3. The optimization process with constraints is performed by the MATLAB function "fmincon". Its target function  $f(x)$  is  $k$  bounded by linear constraints (limits) of speed and acceleration in the actuators, as well as, the working limit of the spherical joints. In the present work, two different optimal paths are generated, one for the identification of the base parameters and the other is employed in the compatibility analysis of the forces estimated by the compact model and the simulation in Adams.

*Parameters for the generation of trajectories:*

The general parameters for the generation and optimization of the paths are:

*Sampling time:* Based on the simulation characteristics established [12] and to maintain a suitable precision without increasing the optimization process, a sampling period of  $dt = 0.1s$  is established.

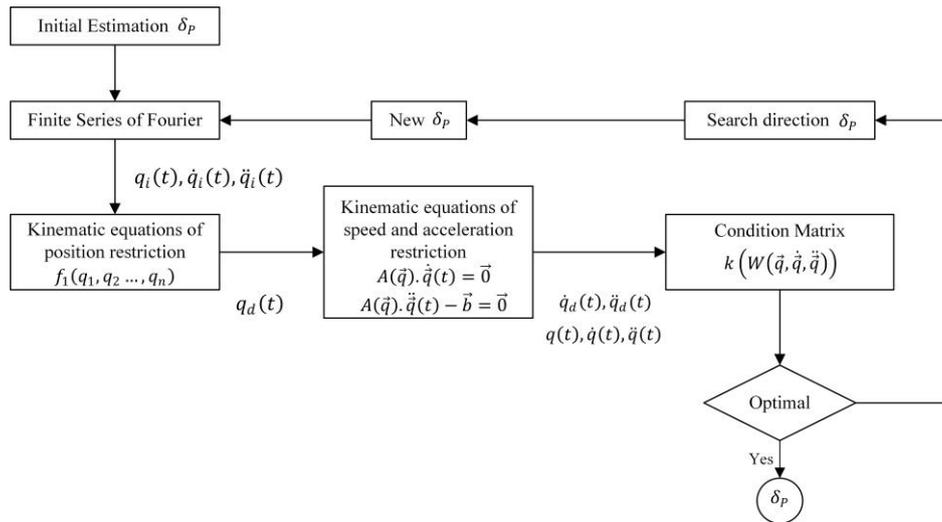


Figure 3. Optimization of trajectories

*Number of harmonics ( $n_H$ ):* 7 harmonics are established for the generation of finite Fourier series. After several tests performed, shown in Fig. 4, it is determined that if they are less than 7 the curves are pure sinusoidal functions and with a number greater than 7 the complexity of the optimization process increases.

*Position limits for each linear actuator:* Based on the three-dimensional model [12] for the parallel robot in study, the minimum position limits are established  $lb_{p_i}$  and maximum  $ub_{p_i}$  within which the optimal trajectories generated for the independent joints ( $q_{13}, q_{23}, q_{33}, q_{42}$ ) must be maintained:

$$lb_{p_i} = [\underbrace{0.4257}_{q_{13}} \quad \underbrace{0.4285}_{q_{23}} \quad \underbrace{0.4285}_{q_{33}} \quad \underbrace{0.3758}_{q_{42}}] \quad (31)$$

$$ub_{p_i} = [0.6090 \quad 0.6110 \quad 0.6110 \quad 0.5753]$$

*Speed limits of each actuator:* According to [12], the liner actuators in the active joints of the parallel robot ( $\dot{q}_{13}, \dot{q}_{23}, \dot{q}_{33}, \dot{q}_{42}$ ), can reach a speed between  $lb_{v_i} = 0.01 [m/s]$  to  $ub_{v_i} = 0.01 [m/s]$ .

*Operating range for spherical joints:* The maximum opening angle of the spherical joints in the geometric analysis of the mechanism in Adams/View is  $\alpha_{max}/2 = 50^\circ$ .

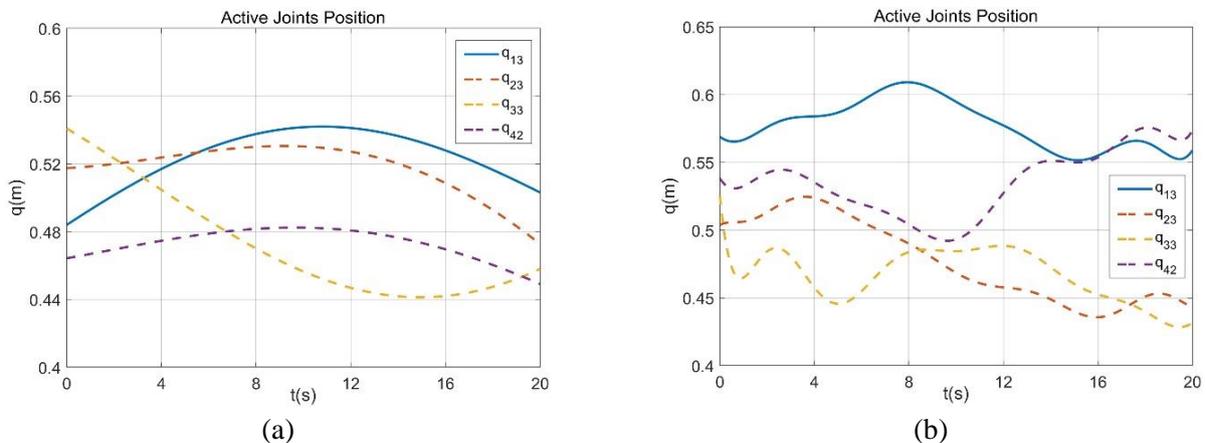


Figure 4. Optimization of trajectories: (a)  $n_H = 2$ , (b)  $n_H = 14$

*Trajectories for the identification of base parameters:*

Based on the amount of time required for the movements in the knee rehabilitation process developed in [12], an average of 20 seconds is taken as the time of simulation (TS) for the optimal curve to be used for the identification of base parameters, for that reason, in this work the trajectories for identification were established as shown in Fig. 5(a), with  $k$  less than 300, which is acceptable for parallel robots according to [18].

*Trajectories for verification of base parameters:*

In order to analyze the compatibility of the identified base parameters, the inverse dynamic problem is solved for that model using optimal trajectories different from those used in the identification process. The optimal verification trajectories are established with a  $TS$  of 10s to differentiate them from those mentioned above. Fig. 5(b) shows the trajectories used in this work to verify the compatibility of the identified base parameters. In these generated curves, the  $q_{23}$  and  $q_{33}$  joints present an identical form in order to prevent the parallel robot from reaching singularity points.

**2.1.6. Base parameters identification**

The base parameter identification process, shown in Fig. 6, begins with the identification trajectories generated for the variables  $q_i$  (actuators). They are introduced into the linear dynamic model with friction model to obtain the path observation matrix ( $W^*$ ). The  $W^*$  matrix together with the parameters  $\vec{\Phi}$ , allow the determination of the base parameters by the methodology of singular value decomposition (SVD). Finally, to reduce the estimation error of the model, the calculated generalized forces ( $\vec{\tau}_m$ ) are compared with the simulated generalized forces ( $\vec{\tau}_s$ ), in a real robot or in its absence in a simulation equivalent to the real one. After this comparison, the base parameters are adjusted to decrease the error between  $\vec{\tau}_m$  and  $\vec{\tau}_s$  through a physical feasibility analysis of the base parameters, i.e, the adjusted base parameters must be physically possible.

**2.2. Real platform assembly and control**

The first model presented in Fig. 1 was analyzed in order to establish an appropriate distribution of the actuators. This revealed that in the configuration  $X = 0, \theta = 0^\circ, \psi = 0^\circ$ , the robot fell into a singularity in the Jacobian when solving the problem of direct kinematic of position [20]. This implies that in that position the platform loses stability and cannot be controlled. By solving the inverse kinematics problem for the configuration, the following point was found:

$$\begin{aligned} q_{13} &= 0.2236067977, & q_{23} &= 0.2276468090 \\ q_{33} &= 0.2276468089, & q_{44} &= 0.2000000000 \end{aligned}$$

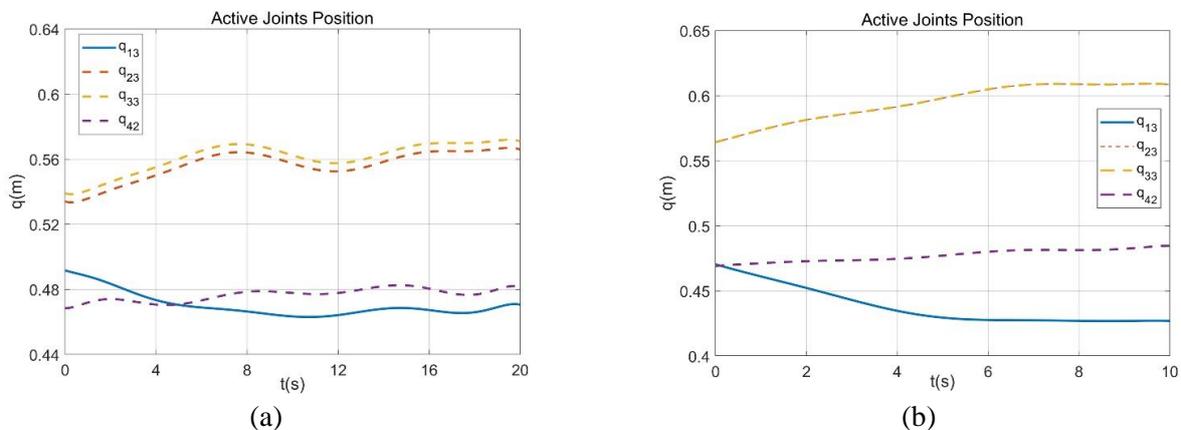


Figure 5. Optimum position trajectories in the four linear actuators to: (a) identify basic parameters, (b) verify the identified base parameters

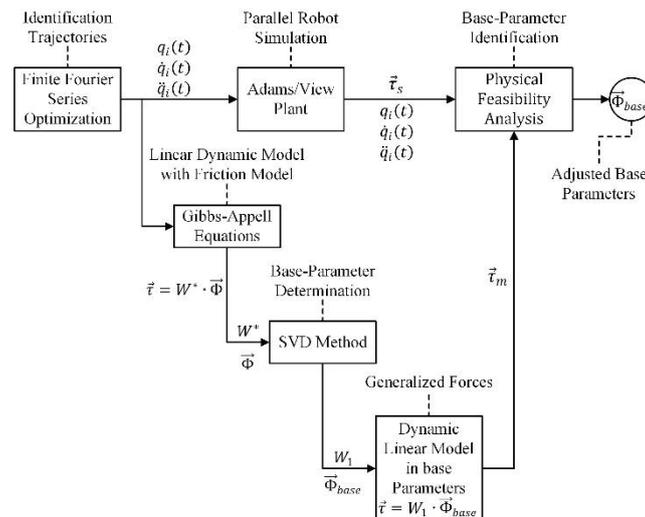


Figure 6. Process of identifying base parameters

Solving the direct problem with these values, the Jacobian is now full range, which required changing the geometrical configuration of the linear actuators. The geometrical configuration is illustrated on Fig. 7(a). The first configuration corresponds to the measurements presented in Table 1. As it was stated before, in order to avoid problems with singularities, the geometry of the robot was modified. The symmetry in the arrangement of the end joints on both the base and the mobile platform was modified. Consequently, a new final configuration was proposed and is presented in Table 1. With this new geometry, we proceed with the design of the structure and the assembly of the robot.

The design is based on the fulfillment of the parameters of physical mobility of the knee established with the help of a physiotherapist. As part of these movement, we have:

*Hip flexion:* It is performed by a trajectory in Z with a displacement of 0.2m.

*Knee flexion-extension:* It is obtained with a trajectory from  $X = 0$  to  $X = 0.2\text{ m}$  and then it is moved to  $X = -0.11\text{m}$ .

*Internal and external rotation of the knee:* it is performed around the Z axis ( $\psi = 15^\circ$ ) and  $\psi = -15^\circ$ .

*Ankle flexion-extension:* it is achieved through the movement around the Y axis ( $\theta = 30^\circ$  y  $\theta = -30^\circ$ ) [12]

### 2.2.1. Mechanical components and assembly

These elements have been selected according to their specifications, geometries and applications. It is also important to consider the ease of adaptation with other elements. Fig. 7(b) shows the main parts of the designed parallel robot, while Fig. 7(c) shows the assembly of the different components that are part of the parallel robot 3RPU+1UPS.

Table 1. Initial geometrical configuration

| Starting Configuration  |                         | Final Configuration     |                         |
|-------------------------|-------------------------|-------------------------|-------------------------|
| Base Platform           | Mobile Platform         | Base Platform           | Mobile Platform         |
| $R_1 = 0.4\text{ m}$    | $R_{m1} = 0.2\text{ m}$ | $R_1 = 0.4\text{ m}$    | $R_{m1} = 0.3\text{ m}$ |
| $R_2 = 0.4\text{ m}$    | $R_{m2} = 0.2\text{ m}$ | $R_2 = 0.4\text{ m}$    | $R_{m2} = 0.3\text{ m}$ |
| $R_3 = 0.4\text{ m}$    | $R_{m3} = 0.2\text{ m}$ | $R_3 = 0.4\text{ m}$    | $R_{m3} = 0.3\text{ m}$ |
| $ds = 0\text{ m}$       |                         | $ds = 0.15\text{ m}$    |                         |
| $\beta_{FD} = 50^\circ$ | $\beta_{MD} = 40^\circ$ | $\beta_{FD} = 90^\circ$ | $\beta_{MD} = 50^\circ$ |
| $\beta_{FI} = 40^\circ$ | $\beta_{MI} = 30^\circ$ | $\beta_{FI} = 45^\circ$ | $\beta_{MI} = 90^\circ$ |

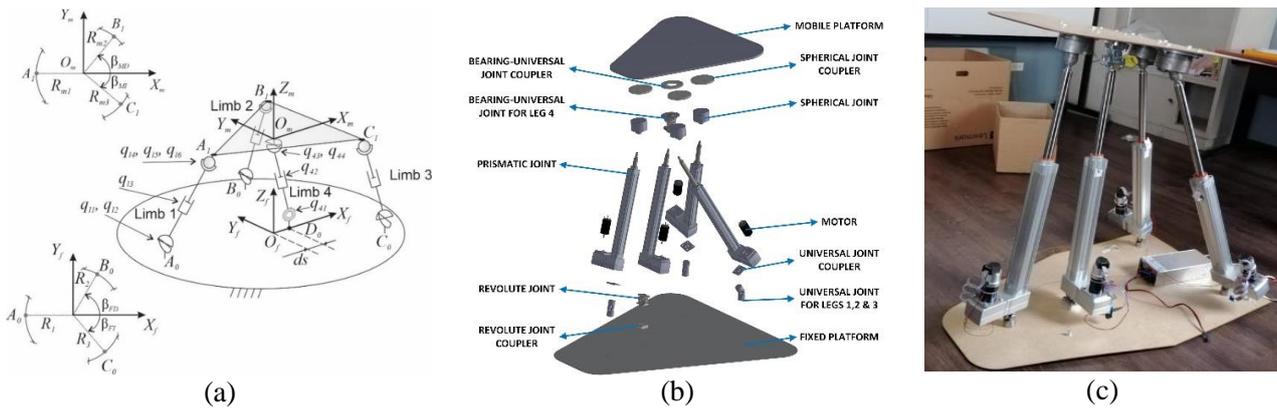


Figure 7. Final geometrical configuration of the parallel robot (a). Parts of the designed parallel robot (b). 3UPS+1RPU parallel robot assembly (c)

**Prismatic joints:** For the prismatic seals, Fig. 8(a), Festo electromechanical cylinders ESBF\_BS-32-100-10P are used on all legs. This element was selected because of its precision in linear movement of 300 mm.

**Revolute joint:** The revolute joint, Fig. 8(b), which is used in the design is a Festo brand. Two elements are used: swivel flanges SNCL and SNCB obtaining a working angle of  $\pm 47^\circ$ .

**Spherical joint:** This spherical joint, Fig. 8(c), is made by Cicrosa model 123R which has a working angle of  $\pm 30^\circ$ . This joint must be machined in order to fit the system.

**Universal joint for legs 1, 2 and 3:** The universal joints, Fig. 8(d), are Belden model UJ-NB1250x00. This element has a working angle of  $\pm 45^\circ$  and must also be machined at both ends to fit the robot.

**Parallel set and motor:** The connection between the electromechanical cylinder, the motor and the seals on the fixed platform is made using a Festo kit type EAMM-U-50-D32. A Maxon 148877 motor is used, which provides a nominal torque of 0.187 Nm. The connection between the three elements is shown in Fig. 8(e).

**Leg 4 universal joint assembly:** The universal joint of leg 4, see Fig. 9(a), consists of a rolling element that allows movement around the Z axis and a revolution joint connected to a coupling by a threaded hole in the stem of the prismatic joint.

**Spherical joints assembly:** The base of the spherical joint is connected to a coupling, which is assembled to the mobile platform through countersunk Allen screws. The bottom of the joint has a threaded hole that allows connection to the prismatic joint stem as shown in Fig. 9(b).



Figure 8. Prismatic Joint (a). Revolute Joint (b). Spherical Joint (c). Universal Joint for legs 1, 2 and 3 (d). Parallel set, Maxon motor and electromechanical cylinder Festo (e)

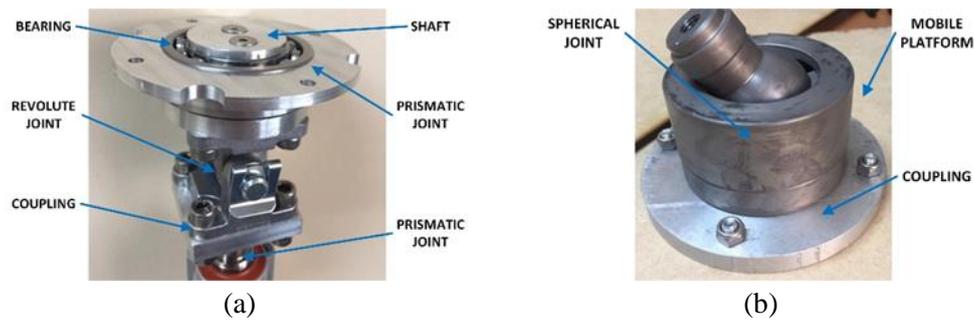


Figure 9. 2 DOF universal joint for leg 4 (a). Spherical joint assembly (b)

### 2.2.2. Control system

After assembling the physical structure and establishing the matrix model of the robot, the programming stage of the controllers is carried out in order to command the positions of the platform [21] and verify its behavior. The components that are part of the control scheme are classified into two groups: hardware and software.

#### *Hardware architecture*

The parallel robot is controlled by an industrial PC that communicates with the PCI-1720 and PCI-1784 cards. The first one is used to send analogical activation signals to the motor drivers by means of the four channels it has available, the second one has four channel counters to be able to read the signals coming from the motor encoders and thus calculate the position of the linear actuators. The cards are shown in Fig. 10 (a). The hardware installed directly in the robot structure corresponds to the motors, motor drivers and their respective quadrature encoders. The motor drivers are the ESCON 50/5. They have analog inputs which are used to enter the required values to control the motors. The motors are Maxon DC model RE40 GB with a maximum power of 150W and they are in charge of moving the axis of the linear drives. Finally, the ENC model HEDL 9140 500IMP encoders are factory installed on the motors and they are able to send 500 pulses for each revolution. The architecture described above is shown in Fig. 10(a) which illustrated the components setup within the platform.

#### *Software architecture*

The software architecture components are shown in Fig. 10(b). The industrial computer runs Ubuntu 14.04 operating system since this system is compatible with ROS Indigo and OROCOS middleware [22]. ROS framework is used to create routines and subroutines that contain the different programming codes and its environment allow an easy manipulation and compilation. OROCOS middleware is composed of a set of libraries designed for the implementation of applications that require to be executed in real time. By using the two middlewares it is possible to manipulate different programming modules and at the same time modify them to integrate components that can be executed in real time. This configuration makes it possible to create codes that contain control routines.

The libraries of both OROCOS and PCI cards are written in C++, for this reason the components generated to execute the control of the structure [23] are written in the same programming language. This allows the executed programs to work efficiently in real time and at the same time access the resources of the PCI cards to communicate with the robot hardware. The work of the mobile platform consists of moving along a trajectory. To achieve this goal, two components are generated. The first called Sync with which the reference points are organized and sent to the controllers and the second called Drivers with which the position of a linear actuator is controlled. This last component can be executed several times in parallel due to the characteristics of OROCOS, so the controllers for all the motors can be started in this fashion.

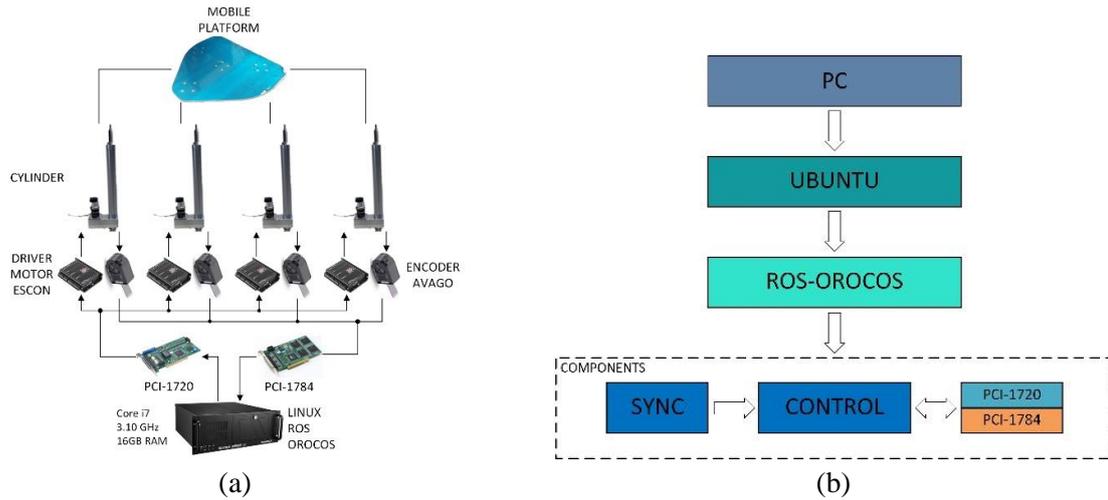


Figure 10. Hardware architecture (a). Software architecture (b)

### 3. Results and discussion

#### 3.1. Simulation results

##### 3.1.1. Compatibility of the dynamic model in base parameters

Based on [12], the compatibility analysis of the identified base parameters starts by finding the generalized forces of the four active joints  $F_{m_1}, F_{m_2}, F_{m_3}, F_{m_4}$ . They are found based on the optimal position, speed and acceleration validation paths determined for the four active joints ( $q_{13}, q_{23}, q_{33}, q_{42}$ ) of the robot, see Fig. 11(a). On the other hand, the paths for compatibility analysis are sent to the simulation plant in Adams/View and the results of the generalized forces of the four prismatic joints ( $F_{s_1}, F_{s_2}, F_{s_3}, F_{s_4}$ ) are recorded in MATLAB. Finally, the residues between the forces calculated by the dynamic model in base parameters ( $F_{m_i}$ ) and the simulated forces ( $F_{s_i}$ ) are analyzed. This error is quantified by calculating the root mean square error.

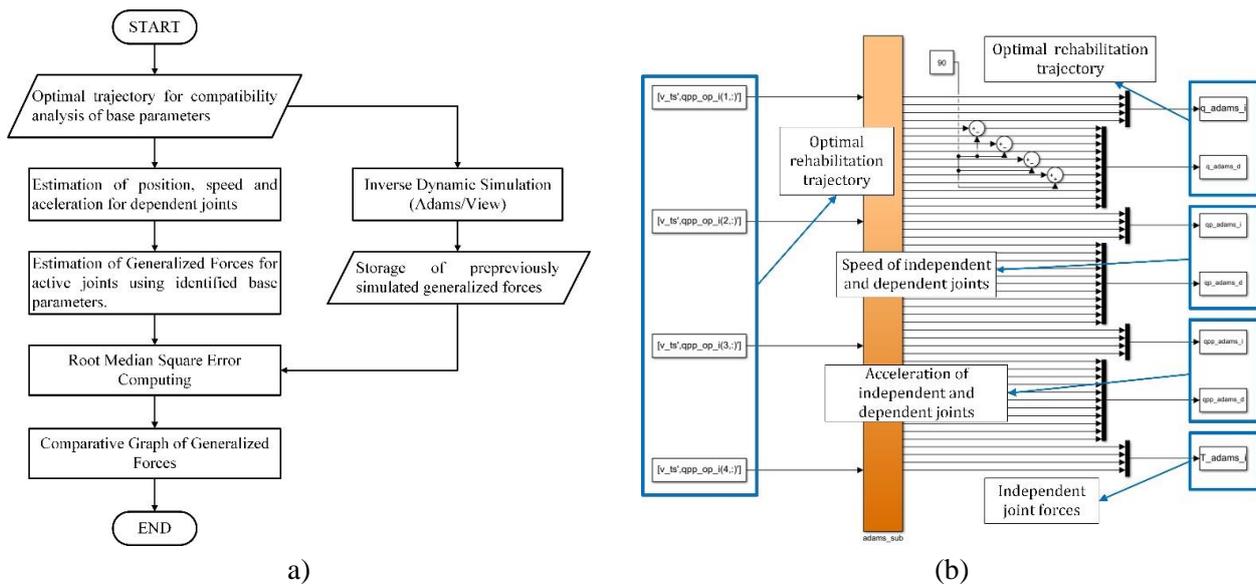


Figure 11. Flowchart of the compatibility analysis of the base parameters identified for the parallel robot 3UPS + 1RPU (a). Diagram for starting the dynamic simulation of the parallel robot (b)

*Virtual plant in Adams/View:* The plant in Adams/View used for the simulation of the dynamic behavior of the 3UPS+1RPU robot takes as it inputs the independent accelerations  $\ddot{q}_{13}$ ,  $\ddot{q}_{23}$ ,  $\ddot{q}_{33}$  &  $\ddot{q}_{42}$  and as the simulation proceeds the forces required to be exerted on the active joints are recorded ( $F_{s1}$ ,  $F_{s2}$ ,  $F_{s3}$ ,  $F_{s4}$ ). Fig. 11(b) shows the block diagram in MATLAB/Simulink to start the simulation in Adams/View. The block diagram in Fig. 11(b), is based on the diagram developed in [12]. In order to develop the identification of the base parameters, the inputs related to external forces are removed. Also, the optimal trajectories are introduced through the accelerations in each independent joint and making Adams/View integrates and determines the corresponding velocity and position curves.

*Base parameters:* After performing the dynamic modelling process described in section 2.1, the model of the parallel 3UPS+1RPU robot is determined based on 90 inertial and 22 friction parameters. By analyzing the null columns of the observation matrix  $K_1$  given in (18), 16 inertial parameters are discarded, see Table 2 . Therefore, 74 inertial and 22 friction parameters are entered to the SVD process for identification of base parameters. As initial values, we use the inertial parameters of the virtual mechanism developed by [12] and the friction parameters determined in the work of Díaz [24]. After running the SVD method, a total of 56 base parameters are determined such that 34 are linear combinations of the inertial parameters and 22 are friction parameters. It can be noted that the inertial parameters were reduced from 90 to 34 parameters. This reduction highlights the advantages of simplification applied to the mathematical model.

*Generalized forces for the optimal identification trajectory*

From the difference between the generalized forces of the actuators applying the reduced linear model considering the friction effects ( $F_m$ ) and the corresponding forces obtained from the simulation ( $F_s$ ), the residue graphs are obtained in Fig. 12.

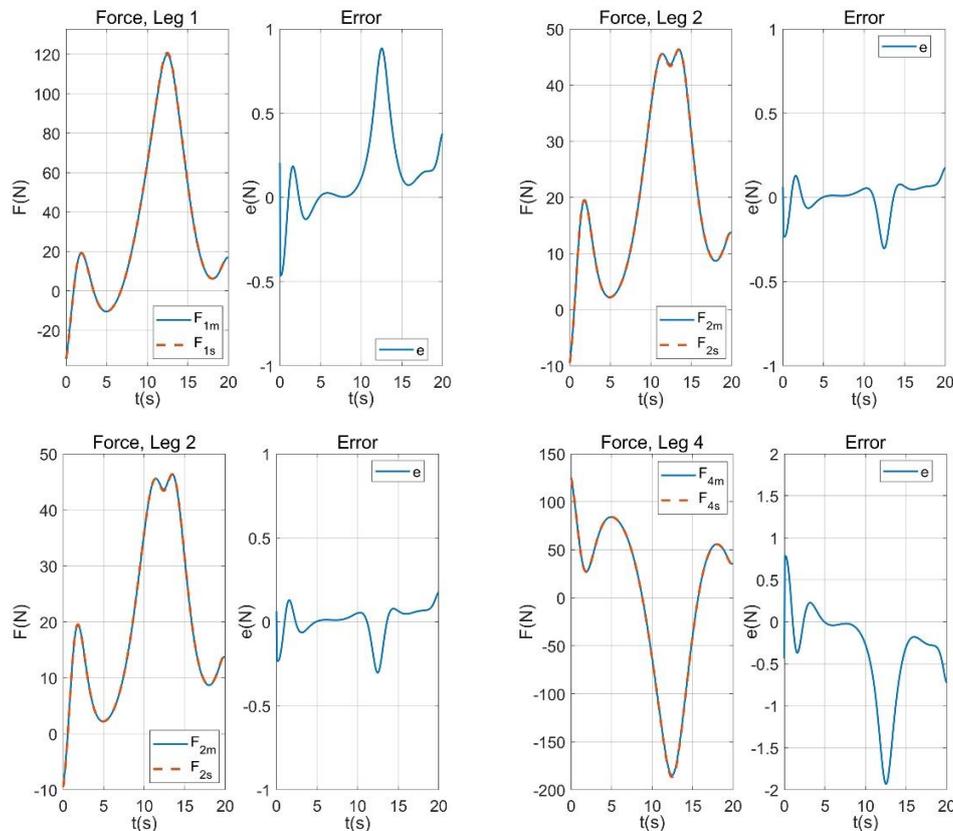


Figure 12. Force in legs during identification trajectory

Table 2. Inertial parameters that have no effect on the dynamic model

| Inertial Parameters Discarded |            |            |            |
|-------------------------------|------------|------------|------------|
| $m_{11}$                      | $IXX_{41}$ | $IYZ_{41}$ | $IYZ_{42}$ |
| $m_{21}$                      | $IXY_{41}$ | $IXX_{42}$ | $IZZ_{42}$ |
| $m_{31}$                      | $IXZ_{41}$ | $IXY_{42}$ | $my_{42}$  |
| $m_{41}$                      | $IYY_{41}$ | $IXZ_{42}$ | $mz_{41}$  |

Since there is no standard for robot modeling applied to medicine or clinical rehabilitation, Zeigler's [25] criterion is used. It establishes a maximum error of 5% as acceptable for system modeling with medical applications. In Table 3 the maximum error shown is 1.86% (0.29 N), so the dynamic model in base parameters is compatible with the dynamics of the 3UPS+1RPU robot.

*Generalized forces for the optimal verification trajectory*

For the optimal verification paths, the difference between  $F_m$  and  $F_s$  is calculated. From this analysis, the residue graphs are obtained and they are shown in Fig. 13.

Table 4 shows a maximum error of 15.80% (1.48 N). An error less than 16% determines that the linear dynamic model in base parameters is compatible for the dynamic behavior of the parallel 3UPS+1RPU robot for knee rehabilitation applications [25].

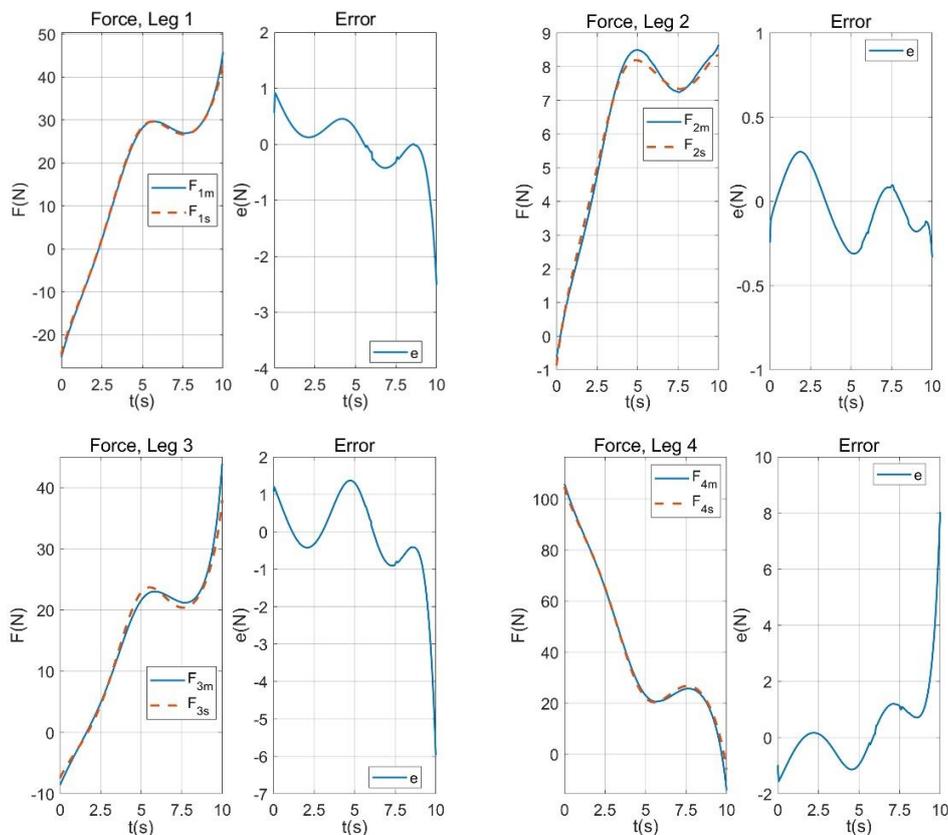


Figure 13. Force on legs during verification trajectory

Table 3. ECM of generalized active robot forces, identification

|              | Unit | Leg 1 | Leg 2 | Leg 3 | Leg 4 |
|--------------|------|-------|-------|-------|-------|
| Error Forces | N    | 0.291 | 0.095 | 0.419 | 0.625 |
|              |      | 8     | 7     | 6     | 9     |
|              | %    | 1.869 | 0.696 | 1.457 | 0.847 |

### 3.2. Real robot performance test

#### *System Setup*

Once the control of the actuators has been established it is possible to send reference positions to the controllers to operate the mobile platform. These points are calculated based on trajectories defined in a script developed in MATLAB in order to generate a coordinated movement to maintain the physical integrity of the platform. To verify the trajectory described by the platform, a system of cameras that are part of the Kinescan vision system [15] is used. These cameras can obtain positions in the space of a given area having coordinates in time. This information can be imported into MATLAB and thus contrast the calculated trajectories with the real ones.

The system described is represented in the diagram of Fig. 14(a) The ROS/OROCOS computer refers to the industrial PC that controls all the hardware of the robot that executes the trajectories requested from the controller. The Kinescan computer is responsible for the execution of the platform motion analysis system through of a set of six cameras. After the system is configured, it proceeds to move to the platform. Trajectories are generated containing thousands of reference points arranged in a 10 ms interval [4] these are stored within a plain text file. The Sync component is then used to access this file. This component reads and sends the references to each of the running controllers in an orderly manner.

Both control and synchronization components are configured to create communication ports for sending position data. This allows the Sync component to read the positions from the text file, sort and send them through the ports of each controller. The process described is composed of three types of components, the file containing the path, the component to synchronize, and the controllers for each motor. The process described previously allows the execution of the controllers for the four motors without the need to depend on an external process.

In order to put the platform into operation, a trajectory was created using a set of scripts developed in MATLAB. The process consisted of generating a simple trajectory that moves the platform in such a way that it focuses on a single degree of freedom during each movement. It will move first on the X axis, then on the Z axis and rotate on  $\psi$ . Afterwards, it will move backwards, so the platform returns to its original position. This movement was designed to avoid positions that could damage the platform. The trajectory assumed a linear speed of 20 mm/s and a rotational speed of 2°/s. These values were established experimentally. At higher speeds, the response of the hardware components is compromised and the movement described by the platform presents interferences.

Table 4. ECM of generalized active robot forces, verification

|              | Unit | Leg 1  | Leg 2  | Leg 3  | Leg 4  |
|--------------|------|--------|--------|--------|--------|
| Error Forces | N    | 0.5050 | 0.1805 | 1.1703 | 1.4822 |
|              | %    | 3.458  | 4.492  | 14.537 | 15.806 |

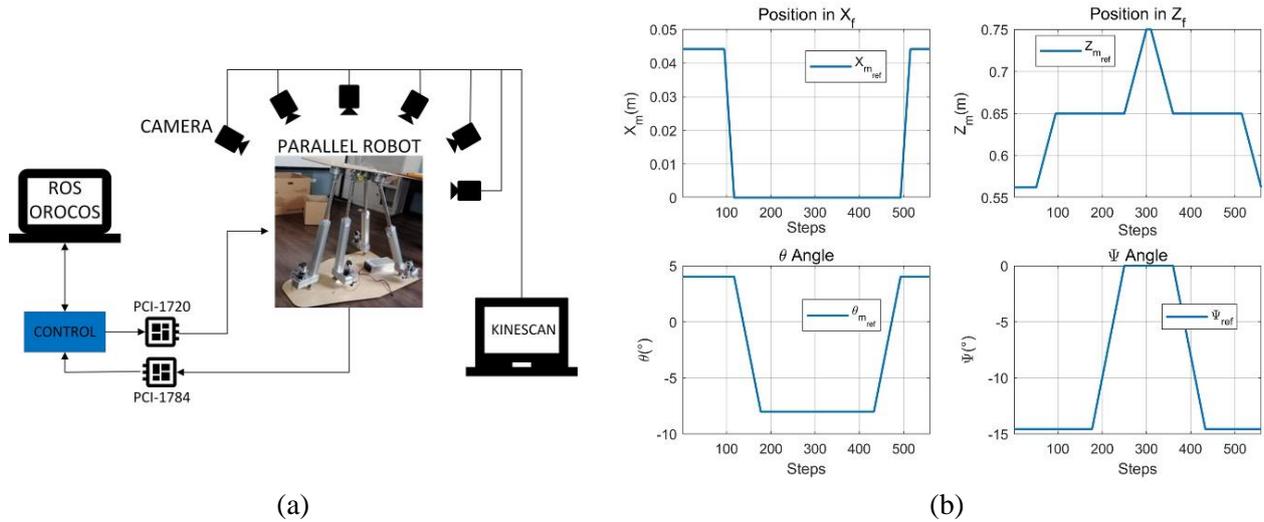
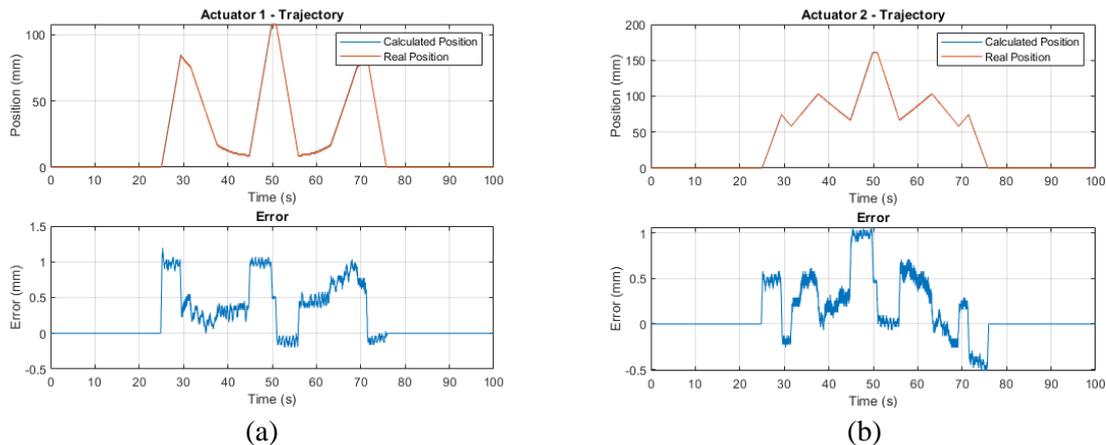


Figure 14. Complete system schematic (a). Trajectory movements on the four axes of freedom (b)

The mobile platform was first aligned with the base to facilitate data manipulation during the processing of the results and to avoid deviations during the execution of the movement. The next step was to raise the platform a height of approximately 200 mm in the Z-axis, and once in this position a 15° rotation around the Z-axis was generated. When the described point was reached, the platform was moved to its starting position by executing the movements backwards. While executing the movements, the industrial computer stored the data of the actuators. Thus, it is possible to contrast the reference values with the ones measured and plot the position errors of each linear actuator. In addition, the motion of the mobile platform was captured by the motion analysis system described at the beginning of this section. This system allows to record the position of the platform versus time. Once the movement is finished, this data helps to contrast the path described by the robot and the one calculated to find the errors in the position. Since the motion analysis system saves the data once the movement is finished, the error can only be performed after the trajectory is completed. Furthermore, this implies that no online feedback can be sent to the system for the control of the platform position. The described trajectory in this paragraph is presented in Fig. 14(b), which shows the movements of the four degrees of freedom of the platform.

The prescribed trajectory was executed after the behavior of each actuator was plotted individually in order to contrast the reference position with the real one. Fig. 15 shows the graphs of the behavior of the four actuators. In Fig. 15 (a), (b) and (c) the position error does not exceed 1 mm, however, in Fig. 15(d) the position error exceeds 2 mm, i.e. the position error of actuator 4 is greater than all others. This is because the weight of the mobile platform lies directly on this actuator, while the others help it to balance the load.



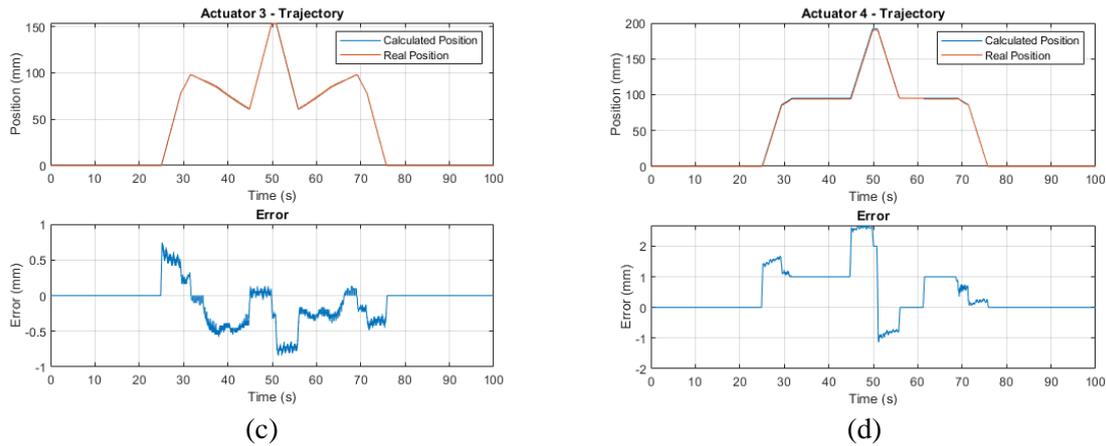
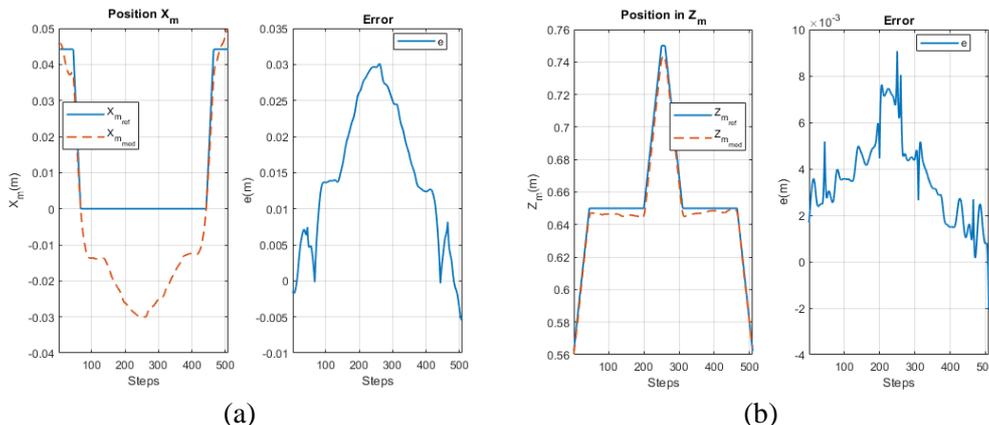


Figure 15. Performance of the motors when executing the trajectory

The analysis of the platform position is presented by comparing the position of the mobile platform with the reference trajectory. Their results are shown in Fig. 16. The position in the X axis is presented in Fig. 16 (a), where it can be seen that the error has a maximum peak of 30 mm. As for the movement in the Z axis presented in Fig. 16 (b), a maximum error of 10mm can be observed. Fig. 16 (c) shows the rotation around the Y axis, where the most significant error is equal to  $-2.5^\circ$ . Fig. 16 (d) shows the rotation around the Z axis, where the maximum error is  $7^\circ$ . The maximum error obtained in the simulation process was 15.80%, which confirms that the approximation of the mathematical model to the real robot is improved by integrating the friction effect to the dynamic model developed under the Gibbs-Appell equations. In the validation of the implicit dynamic model of the parallel robot 3UPS + 1RPU developed by [12], a maximum error of 19.84% was obtained. Therefore, it can be observed that the dynamic model in base parameters reduces by 4.04% the error of estimation of the generalized forces in the independent joints. In addition, the reduced model will allow to design and implement controllers based on passivity that will be developed in the next phase of this project.

By analyzing the performance of the virtual model and the assembled prototype, it was possible to determine that the error on the X axis is larger than on the Z axis since moving on that axis is more complex due to the intervention of the four linear actuators and any slight deviation from one of them results in an appreciable error in the position. On the other hand, the movement on the Z axis has a lower error because a linear actuator is aligned with this axis while the other actuators help to level the platform. The rotation described by the angle  $\theta$  does not show any major deviation in this case due to the configuration of the trajectory. Meanwhile, the angle described by  $\psi$  showed considerable errors due to the fact that the synchronized intervention of the linear actuators is required to achieve rotation around it. In addition, the looseness of the actual spherical joints has an important role to consider since they limit the free mobility of the platform and then the range of movement.



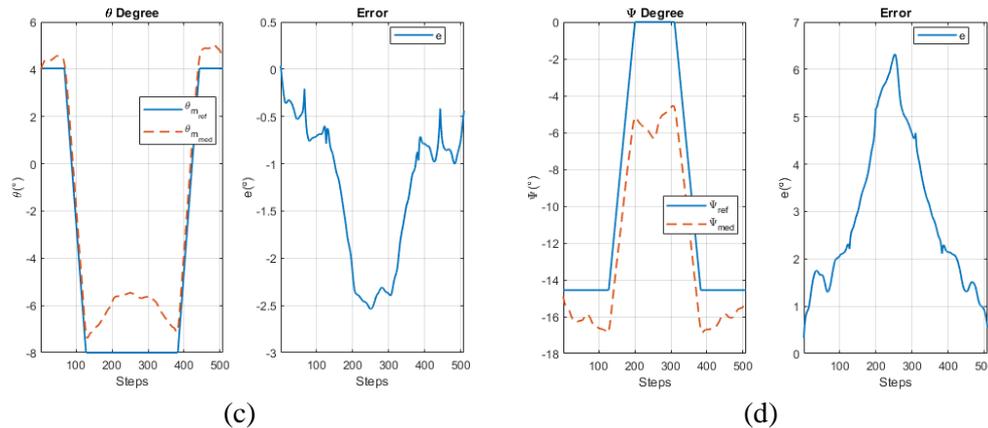


Figure 16. Contrast between calculated and executed trajectory

#### 4. Conclusion

The dynamic linear model is obtained based on inertial parameters of the 3UPS+1RPU parallel robot, including a symmetric linear friction model, then this model is reduced from 112 dynamic parameters to 56, information that highlights the advantages of simplification applied to the model. Statistical comparison between forces estimated by the reduced dynamic linear model and the forces simulated in Adams/View results in a maximum error of 15.80% (1.48N). This value is admissible for knee rehabilitation applications. The error in the evaluation process of the independent generalized forces is 15.80%, which is less than the error of 19.84% obtained in the simulation of the non-linear model, this information demonstrates that integrating friction models into the dynamic model of a robot, the error in the estimation of the forces in the independent joints is reduced.

The accuracy of the robot in the Z axis is higher than in any other axis due to its design, the fourth linear actuator acts directly in the base of the mobile platform, thus allowing the reduction of the position errors caused by the remaining actuators which act on the edges of the mentioned platform. On the other hand, the errors found in the rotational axes show a significant value caused by the looseness in the spherical joints.

The performance of the robotic prototype is consistent with the data obtained from the virtual model; for this reason, the trajectories could be used successfully for both cases. This allows the proposition of new studies that consist of analyzing the performance of the robot under different loads and develop a model of the ankle to study its behavior on the platform. The results obtained by the simulation of the reduced dynamic linear model in Adams/View and validated by photogrammetry are consistent and demonstrate the validity of the presented work with the established criteria.

#### Acknowledgment

This work has been carried out in the Faculty of Mechanical Engineering of the Escuela Politécnica Nacional (EPN), as part of the Inter-Multidisciplinary Project PIMI-1504 "Adaptive control based on artificial intelligence applied to a mechatronic system based on a parallel robot for diagnosis and rehabilitation". The authors acknowledge the financial support provided by the institution for the development of this project.

#### References

- [1] W. Meng, Q. Liu, Z. Zhou, Q. Ai, B. Sheng and S. Xie, "Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation," *Mechatronics*, vol. 31, pp. 132-145, 2015.
- [2] B. Chen, B. Zi, Y. Zeng, L. Qin and W.-H. Liao, "Ankle-foot orthoses for rehabilitation and reducing metabolic cost of walking: Possibilities and challenges," *Mechatronics*, vol. 53, pp. 241-250, 2018.

- [3] E. Akdoğan and M. A. Adli, "The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherobot," *Mechatronics*, vol. 21, no. 3, pp. 509-522, 2011.
- [4] M. Vallés, J. Cazalilla, Á. Valera, V. Mata, Á. Page and M. Díaz-Rodríguez, "A 3-PRS parallel manipulator for ankle rehabilitation: towards a low-cost robotic rehabilitation," *Robotica*, vol. 35, no. 10, pp. 1939-1957, 2017.
- [5] W. Shang, S. Cong and F. Kong, "Identification of dynamic and friction parameters of a parallel manipulator with actuation redundancy," *Mechatronics*, vol. 20, no. 2, pp. 192-200, 2010.
- [6] B. Zi, B. Y. Duan, J. L. Du and H. Bao, "Dynamic modeling and active control of a cable-suspended parallel robot," *Mechatronics*, vol. 18, no. 1, pp. 1-12, 2008.
- [7] T. Dallej, M. Gouttefarde, N. Andreff, P.-E. Hervé and P. Martinet, "Modeling and vision-based control of large-dimension cable-driven parallel robots using a multiple-camera setup," *Mechatronics*, vol. 61, pp. 20-36, 2019.
- [8] M. Díaz-Rodríguez, A. Valera, V. Mata and M. Valles, "Model-Based Control of a 3-DOF Parallel Robot Based on Identified Relevant Parameters," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 6, pp. 1737-1744, 2013.
- [9] L. Carbonari, "Simplified approach for dynamics estimation of a minor mobility parallel robot," *Mechatronics*, vol. 30, pp. 76-84, 2015.
- [10] M. Díaz-Rodríguez, J. A. Carretero and R. Bautista-Quintero, "Solving the dynamic equations of a 3-PRS Parallel Manipulator for efficient model-based designs," *Mechanical Sciences*, vol. 7, no. 1, pp. 9-17, 2016.
- [11] N. Farhat, V. Mata, Á. Page and F. Valero, "Identification of dynamic parameters of a 3-DOF RPS parallel manipulator," *Mechanism and Machine Theory*, vol. 43, pp. 1-17, 2008.
- [12] J. Pulloquina, S. Aquino, E. Pozo, P. Cruz and O. Zambrano, "Dynamic Model of a Parallel Robot Type 3UPS + 1RPU for Knee Rehabilitation," in *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*, 2018.
- [13] P. Araujo-Gómez, V. Mata, M. Díaz-Rodríguez, A. Valera and A. Page, "Design and Kinematic Analysis of a Novel 3UPS/RPU Parallel Kinematic Mechanism with 2T2R Motion for Knee Diagnosis and Rehabilitation Tasks," *Journal of Mechanisms and Robotics*, vol. 9, no. 6, 2017.
- [14] Y. Dai, Y. Fu, B. Li, X. Wang, T. Yu and W. Wang, "Clearance effected accuracy and error sensitivity analysis: A new nonlinear equivalent method for spatial parallel robot," *Journal of Mechanical Science and Technology*, vol. 31, no. 11, pp. 5493-5504, 2017.
- [15] J. V. Hoyos, J. Montero, R. Llobet, J. M. Belda, G. Brizuela and , "Sistema de análisis de movimientos KINESCAN-IBV," *Revista de Biomecánica*, pp. 22-27, 14 January 1997.
- [16] J. P. Merlet, "Parallel Robots," *Solid Mechanics and Its Applications*, vol. 128, 2006.
- [17] M. Díaz-Rodríguez, X. Iriarte, V. Mata and J. Ros, "On the Experiment Design for Direct Dynamic Parameter Identification of Parallel Robots," *Advanced Robotics*, vol. 23, pp. 329-348, 2009.
- [18] F. Benimeli, V. Mata and F. Valero, "A comparison between direct and indirect dynamic parameter identification methods in industrial robots," *Robotica*, vol. 24, no. 5, pp. 579-590, 2006.
- [19] V. Mata, F. Benimeli, N. Farhat and A. Valera, "Dynamic parameter identification in industrial robots considering physical feasibility," *Journal of Advanced Robotics*, vol. 19, no. 1, pp. 101-120, 2005.
- [20] J. Aginaga, X. Iriarte, A. Plaza and V. Mata, "Kinematic Design of a New Four Degree-of-Freedom Parallel Robot for Knee Rehabilitation," *Journal of Mechanical Design*, vol. 140, no. 9, 2018.

- [21] W. Meng, S. Q. Xie, Q. Liu, C. Z. Lu and Q. Ai, "Robust Iterative Feedback Tuning Control of a Compliant Rehabilitation Robot for Repetitive Ankle Training," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 1, pp. 173–184, 2017.
- [22] H. Fischer, P. Vulliez, J.-P. Gazeau and S. Zegloul, "An industrial standard based control architecture for multi-robot real time coordination," in *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, Poitiers, France, 2016.
- [23] N. Hochgeschwender, G. Biggs and H. Voos, "A reference architecture for deploying component-based robot software and comparison with existing tools," in *2018 Second IEEE International Conference on Robotic Computing (IRC)*, Laguna Hills, 2018.
- [24] M. Díaz-Rodríguez, V. Mata, N. Farhat and S. Provenzano, "Identifiability of the Dynamic Parameters of a Class of Parallel Robots in the Presence of Measurement Noise and Modeling Discrepancy," *Mechanics Based Design of Structures and Machines*, vol. 36, no. 4, pp. 478-498, 2008.
- [25] B. Zeigler, "Theory of Modelling and Simulation: Discrete Event & Iterative System Computational Foundations," *Elsevier*, 2018.