Vision-Based Hybrid Controller to Release a 4-DOF Parallel Robot from a Type II Singularity

José L. Pulloquinga [1],*, Rafael J. Escarabajal [1], Jesús Ferrándiz [1], Marina Vallés [1], Vicente Mata [2] and Mónica Urízar [3]

[1] Departamento de Ingeniería de Sistemas y Automática, Instituto de Automática e Informática Industrial, Universitat Politècnica de València, 46022 Valencia, Spain; raessan2@etsii.upv.es (R.J.E.); jeferal@etsid.upv.es (J.F.); mvalles@isa.upv.es (M.V.)

[2] Departamento de Ingeniería Mecánica y de Materiales, Instituto Universitario de Investigación Concertado de Ingeniería Mecánica y Biomecánica, Universitat Politècnica de València, 46022 Valencia, Spain; vmata@mcm.upv.es

[3] Department of Mechanical Engineering, Faculty of Engineering in Bilbao, University of the Basque Country, 48013 Bilbao, Spain; monica.urizar@ehu.es

* Correspondence: jopulza@doctor.upv.es; Tel.: +34-96-387-7000 (ext. 75783)

Abstract: The high accuracy and dynamic performance of parallel robots (PRs) make them suitable to ensure safe operation in human–robot interaction. However, these advantages come at the expense of a reduced workspace and the possible appearance of type II singularities. The latter is due to the loss of control of the PR and requires further analysis to keep the stiffness of the PR even after a singular configuration is reached. All or a subset of the limbs could be responsible for a type II singularity, and they can be detected by using the angle between two output twist screws (OTSs). However, this angle has not been applied in control because it requires an accurate measure of the pose of the PR. This paper proposes a new hybrid controller to release a 4-DOF PR from a type II singularity based on a real time vision system. The vision system data are used to automatically readapt the configuration of the PR by moving the limbs identified by the angle between two OTSs. This controller is intended for a knee rehabilitation PR, and the results show how this release is accomplished with smooth controlled movements where the patient's safety is not compromised.

Keywords: singular configuration; parallel robot; motion control; 3D tracking; screw theory

1. Introduction

Parallel robots (PRs) are composed of two or more closed kinematic chains connecting a fixed and a mobile platform that defines the end-effector to be controlled [1]. As opposed to their serial counterpart, they benefit from greater accuracy, stiffness, and load capacity, making them suitable for a great variety of applications [2,3]. Human–robot interaction is one of the major applications, for instance, in the context of medical rehabilitation [4]. Within this field, lower limb rehabilitation [5–9] is an active research area. However, PRs also present several drawbacks regarding the size of their workspace and the presence of singularities within the workspace. The former can be addressed by means of a proper mechanical design of the PR to cover the workspace as required, while the latter requires further analysis.

Singularities in a PR were first analysed by Gosselin and Angeles [10], who established a classification of singular configurations according to the characteristics of the Jacobian matrices calculated from constraint equations. They defined a type I (or inverse kinematic) singularity to refer to the loss of at least one degree of freedom (DOF) due to a degeneracy of the inverse Jacobian matrix ($\|J_I\| = 0$) and a type II (or forward kinematic) singularity to indicate the gain of at least one DOF caused by the degeneracy of the for-

ward Jacobian matrix ($\|J_D\| = 0$). Some other related classifications of singular configurations can be found in [11,12]. type I singularities typically occur as the manipulator approaches the boundary of the workspace and are easy to detect and avoid, but type II singularities can arise within the workspace and are more difficult to treat [13].

type II singularities prevent the mobile platform from bearing external forces despite having all the actuators locked, leading to an uncontrolled motion of the end-effector. The main goal of lower-limb rehabilitation is to perform specific movements that stimulate the motor plasticity of the patient to improve the motor recovery [5]. In conventional rehabilitation, the movements of the patient are controlled and monitored by a physiotherapist, while in robotic rehabilitation, the control task is performed by a PR. For this task, a PR must ensure stiff behaviour despite the presence of type II singularities to maintain control during the rehabilitation process.

Extensive research has been conducted to tackle type II singularities. The determinant of the forward Jacobian J_D gives no further information than the proximity to a singularity, as it lacks a physical meaning [14]. Based on screw theory [15], a transmission index (TI) was designed by Yuan et al. [16] to express the quality of force and motion transmission by using the transmission wrench screw (TWS) and output twist screw (OTS). Takeda and Funabashi [17] designed a TI that expresses how each actuator individually contributes to the motion of the mobile platform by leaving just one actuator active and the rest locked. Subsequently, Wang et al. [18] using the TI proposed by Takeda and Funabashi established that for a type II singularity, at least two OTSs are linearly dependent. Pulloquinga et al. [19] proposed the angle between two instantaneous screw axes from the OTSs (Ω) as a measure for the proximity detection of type II singularities, providing physical meaning and the capability to determine the chains producing the singular configuration.

The extensive analysis of type II singularities presented has been incorporated in motion/force performance evaluation [18,20], path planning, and the design of reconfigurable PRs [21,22]. These analyses have been developed offline, and very little has been found about including this information in the control unit of the PR [23,24]. Abgarwal et al. [24] designed a control scheme to avoid type II singularities of a planar PR by using artificial potential functions. The potential functions are activated near the singularity to alter the trajectory by means of repulsion forces. This setting prevents the PR from entering into a singular configuration by avoiding it. However, an evader controller cannot deal with the situation in which the robot is initially in a type II singularity. Such a task would require extra instrumentation, since solving the forward kinematic problem based on the joint variable measures does not have a single solution. The various possible positions of the mobile platform are due to the degeneracy of the forward Jacobian matrix.

One unambiguous solution to estimate the actual pose of the PR is by using a vision system [25]. Huynh et al. [26] implemented a vision/position hybrid control for a Hexa PR by defining a two-level closed-loop controller. Amarasinghe et al. [27] designed a vision-based hybrid control on a mobile robot. It could autonomously reach a docking station by using a finite-state machine and proportional control combined with image processing.

However, to the best of the authors' knowledge, no research has been published focusing on PR singularity releasing, i.e., letting the robot autonomously get out of a type II singularity. In this paper, a novel algorithm based on online readings from a vision system is proposed to release the PR from a type II singularity. The proposed algorithm is the first to use the angle Ω as an online detector for the proximity to singular configurations. This algorithm is integrated into a two-level closed-loop hybrid controller that results in more compliant manipulation when performing knee rehabilitation tasks. In the inner loop, there is an algebraic closed-loop controller. The outer loop implements a vision-based controller whose algorithm determines the two limbs that most affect the type II singularity by means of the angle Ω. Then, only the references of those two limbs are modified online to feed the inner loop of the controller.

Section 2 describes the 4-DOF PR for knee rehabilitation used to perform the simulations and experiments. Next, the mathematical foundations of type II singularities and the angle Ω are explained. Then, the 3D vision system that has been used to keep track of the pose of the PR is presented, together with a detailed description of the proposed vision-based hybrid controller. Section 3 begins with a description of the requirements for simulation and experimentation as well as the singular trajectories that were designed for this research. The main results are also presented in this section. Finally, the results are discussed in Section 4.

2. Materials and Methods

This section presents the mathematical foundation used in the development of the angle Ω that detects the proximity to a type II singularity in a knee rehabilitation PR. Subsequently, the 3D tracking system (3DTS) used to measure the actual pose of the mobile platform is described. Then the 3DTS and Ω are combined to develop a novel vision-based hybrid controller to release the actual PR under study from a type II singularity. This section also includes a detailed explanation of the algorithm corresponding to this hybrid controller, which detects and moves the actuators according to the angle Ω.

2.1. 3U$\underline{P}$S+R$\underline{P}$U Parallel Robot

After knee surgery, the diagnosis and rehabilitation tasks require two translational movements (x_m, z_m) in the tibiofemoral plane, one rotation (ψ) around the coronal plane and one rotation (θ) around the tibiofemoral plane [28]. These four DOFs are shown in Figure 1. In order to accomplish these requirements, a PR with 4-DOF has been designed, built [29], and optimized [30] at the Universitat Politècnica de València. The PR under study is named 3U$\underline{P}$S+R$\underline{P}$U due to its four-limb architecture. The external limbs or open kinematic chains have a U$\underline{P}$S configuration, while the central one has an R$\underline{P}$U configuration (see Figure 1). The letters R, U, S and P stand for revolute, universal, spherical and prismatic joints, respectively, and the actuated joints are indicated by the underlined format.
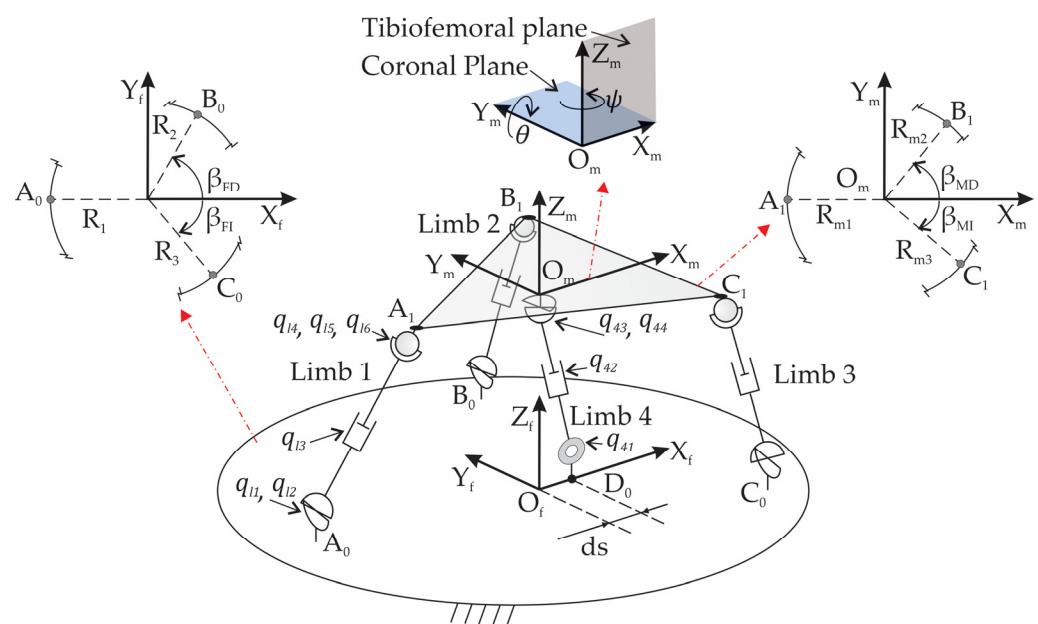
Figure 1. Mechanical configuration of the 3U$\underline{P}$S+R$\underline{P}$U PR.

The kinematic model of the 3U$\underline{P}$S+R$\underline{P}$U PR is established by 15 generalized coordinates as follows:

- The position (x_m, z_m) and the orientation (θ, ψ) of the mobile platform.

- The orientation of the four universal joints: q_{l1}, q_{l2} for limbs $l = 1 \ldots 3$ and q_{43}, q_{44} for limb 4.
- The length of the four linear actuators given by q_{l3} for limbs $l = 1 \ldots 3$ and q_{42} for limb 4.
- The orientation of the three spherical joints represented by q_{l4}, q_{l5}, q_{l6} for external limbs $l = 1 \ldots 3$.
- The orientation of the revolute joint q_{41}.

The variables q are measured with respect to a local reference system attached to each joint. The coordinates x_m, z_m, θ, and ψ are measured with respect to the reference system $\{O_f - X_F Y_F Z_F\}$ attached to the centre of the fixed platform to reduce the complexity of the model.

The locations of A_0, B_0, C_0, and D_0 that link the four limbs to the fixed platform are defined by the geometric variables R_1, R_2, R_3, β_{FD}, β_{FI}, and d_s. The locations of A_1, B_1, C_1, and O_m that link the limbs to the mobile platform are defined by the geometric variables R_{m1}, R_{m2}, R_{m3}, β_{MD}, and β_{MI}. Figure 1 shows the location of A_0, B_0, and C_0 and A_1, B_1, and C_1 on the fixed and mobile platform. Table 1 shows the values of R_1, R_2, R_3, β_{FD}, β_{FI}, and d_s measured with respect to $\{O_f - X_F Y_F Z_F\}$ and R_{m1}, R_{m2}, R_{m3}, β_{MD}, and β_{MI} with respect to $\{O_m - X_M Y_M Z_M\}$. The mobile reference system $\{O_m - X_M Y_M Z_M\}$ is attached to the centre of the mobile platform.

Table 1. Geometric parameters for the 3U$\underline{P}$S+R$\underline{P}$U PR.

R_1 (m)	R_2 (m)	R_3 (m)	β_{FD} (°)	β_{FI} (°)	d_s (m)
0.4	0.4	0.4	90	45	0.15
R_{m1} (m)	R_{m2} (m)	R_{m3} (m)	β_{MD} (°)	β_{MI} (°)	
0.3	0.3	0.3	50	90	

2.2. Type II Singularities

The velocity equations of a PR [10] are defined by the time derivatives of the geometrical constraint equations ($\vec{\Phi}$) as follows:

$$J_D \dot{\vec{X}} + J_I \dot{\vec{q}}_{ind} = \vec{0} \tag{1}$$

$\vec{X}$ is the set of outputs that represents the DOFs of the mobile platform; $\vec{q}_{ind}$ is the set of inputs that corresponds to the active joint length; and J_D and J_I are the forward and inverse Jacobian matrices, respectively. J_D and J_I are FxF square matrices for a non-redundant PR, where F is the number of DOF.

A type II singularity takes place when J_D is rank deficient; i.e., its determinant is zero ($\|J_D\| = 0$). In this configuration, if an external force is applied to the PR, the mobile platform may move ($\dot{\vec{X}} \neq 0$) despite the actuators being locked ($\dot{\vec{q}}_{ind} = 0$). For this reason, in a type II singularity, the control over the PR is lost, becoming potentially dangerous for the user or the robot itself. The PR under study must ensure a safe interaction with the knee of the patient, and, therefore, the treatment of type II singularities is an important problem to solve. A general method to detect the proximity to a type II singularity is by calculating the $\|J_D\|$.

In the 3U$\underline{P}$S+R$\underline{P}$U PR, J_D is defined as a 4×4 matrix,

$$J_D = \begin{bmatrix} \dfrac{\partial \vec{\Phi}}{\partial x_m} & \dfrac{\partial \vec{\Phi}}{\partial z_m} & \dfrac{\partial \vec{\Phi}}{\partial \theta} & \dfrac{\partial \vec{\Phi}}{\partial \psi} \end{bmatrix} \tag{2}$$

with $\vec{X} = [x_m \quad z_m \quad \theta \quad \psi]^T$ and $\vec{q}_{ind} = [q_{13} \quad q_{23} \quad q_{33} \quad q_{42}]^T$.

The online calculation of J_D requires an accurate measure of $\vec{X}$. In a model-based controller, $\vec{X}$ is estimated by solving the forward kinematic problem using the sensors installed in the actuated joints. In a type II singularity, the forward kinematic problem presents several feasible solutions, and an unambiguous estimation of the actual $\vec{X}$ is not

possible. The accurate measure of the actual $\vec{X}$ of the PR requires direct sensing of the mobile platform by means of extra instrumentation, such as a 3DTS.

2.3. Angle between Two Output Twist Screws

The motion of the mobile platform of a PR is produced by the combined action of the active joints, making it difficult to identify the contribution of each actuator. Takeda and Funabashi [17] divided the movement of the mobile platform (\$) into F OTSs as follows:

$$\$ = \rho_1 \hat{\$}_{O_1} + \rho_2 \hat{\$}_{O_2} + \cdots + \rho_F \hat{\$}_{O_F} \tag{3}$$

with

$$\hat{\$}_O = \left(\overrightarrow{\mu_{\omega_O}} ; \overrightarrow{\mu_{v_O}^*} \right) \tag{4}$$

ρ_i is the amplitude for each OTS, and $\overrightarrow{\mu_{\omega_O}}$ and $\overrightarrow{\mu_{v_O}^*}$ are the instantaneous screw axis and the linear component of the normalized OTS ($\hat{\$}_O$), respectively. Each $\hat{\$}_O$ is determined by solving Equation (5) considering that $F - 1$ actuators are locked.

$$\hat{\$}_{O_i} \circ \hat{\$}_{T_j} = 0 \ \ (i, j = 1, 2, \dots, F, i \neq j) \tag{5}$$

where $\circ$ stands for the reciprocal product, and $\hat{\$}_T$ is the unitary TWS.

In [18], Wang et al. proved that for a singular configuration of a PR, at least two $\hat{\$}_O$s are linearly dependent. This means that in a type II singularity, both $\overrightarrow{\mu_{\omega_O}}$ and $\overrightarrow{\mu_{v_O}^*}$ are equal or parallel. Based on this feature, a novel type II singularity proximity index is defined by measuring the angle $\Omega_{i,j}$ between two $\overrightarrow{\mu_{\omega_O}}$s and verifying the equality of their respective $\overrightarrow{\mu_{v_O}^*}$. Grouping in pairs the F $\overrightarrow{\mu_{\omega_O}}$, there are $\binom{F}{2}$ angles Ω, which are defined as:

$$\Omega_{i,j} = acos \left(\overrightarrow{\mu_{\omega_{O_i}}} \cdot \overrightarrow{\mu_{\omega_{O_j}}} \right) \ \ (i, j = 1, 2, \dots, F, i \neq j) \tag{6}$$

where i and j represent the selected $\overrightarrow{\mu_{\omega_O}}$.

In contrast with the $\|J_D\|$, the index $\Omega_{i,j}$ provides a physical scale for the measure of the proximity to a type II singularity. When $\Omega_{i,j} = 0$ and $\overrightarrow{\mu_{v_{O_i}}^*} = \overrightarrow{\mu_{v_{O_j}}^*}$, $\hat{\$}_{O_i}$ and $\hat{\$}_{O_j}$ are linearly dependent in Equation (3), identifying the open chains (i, j) involved in the type II singularity. Considering the centre of the mobile platform of the 3U$\underline{P}$S+R$\underline{P}$U PR, six possible $\Omega_{i,j}$s are considered:

$$\Omega_{1,2} = arccos \left(\overrightarrow{\mu_{\omega_{O_1}}} \cdot \overrightarrow{\mu_{\omega_{O_2}}} \right) \qquad \Omega_{2,3} = arccos \left(\overrightarrow{\mu_{\omega_{O_2}}} \cdot \overrightarrow{\mu_{\omega_{O_3}}} \right)$$

$$\Omega_{1,3} = arccos \left(\overrightarrow{\mu_{\omega_{O_1}}} \cdot \overrightarrow{\mu_{\omega_{O_3}}} \right) \qquad \Omega_{2,4} = arccos \left(\overrightarrow{\mu_{\omega_{O_2}}} \cdot \overrightarrow{\mu_{\omega_{O_4}}} \right) \tag{7}$$

$$\Omega_{1,4} = arccos \left(\overrightarrow{\mu_{\omega_{O_1}}} \cdot \overrightarrow{\mu_{\omega_{O_4}}} \right) \qquad \Omega_{3,4} = arccos \left(\overrightarrow{\mu_{\omega_{O_3}}} \cdot \overrightarrow{\mu_{\omega_{O_4}}} \right)$$

The variables $\overrightarrow{\mu_{\omega_{O_1}}} \dots \overrightarrow{\mu_{\omega_{O_4}}}$ are calculated by solving Equation (5) with the four $\hat{\$}_T$s of the linear actuators defined as

$$\hat{\$}_{T_1} = \begin{bmatrix} \vec{z}_{12} \\ \vec{r}_{O_m A_1} \times \vec{z}_{12} \end{bmatrix}, \hat{\$}_{T_2} = \begin{bmatrix} \vec{z}_{22} \\ \vec{r}_{O_m B_1} \times \vec{z}_{22} \end{bmatrix}, \hat{\$}_{T_3} = \begin{bmatrix} \vec{z}_{32} \\ \vec{r}_{O_m C_1} \times \vec{z}_{32} \end{bmatrix}, \hat{\$}_{T_4} = \begin{bmatrix} \vec{z}_{41} \\ \vec{0} \end{bmatrix} \tag{8}$$

where $\vec{z}$ is the unit vector of the forces applied by the actuators, and $\vec{r}$ is the position vector for the connection point of the limbs with the mobile platform measured from O_m; see Figure 2.

The capability to detect the proximity to a type II singularity given by the six $\Omega_{i,j}$ indices defined in Equation (7) has been verified from an analytical and experimental perspective [19]. However, the capability to identify the pair of limbs responsible for the type II singularity has not been exploited. Therefore, this study proposes a novel hybrid controller that takes advantage of the index $\Omega_{i,j}$ to release the 3U$\underline{P}$S+R$\underline{P}$U PR from a type II

singular configuration. The index $\Omega_{i,j}$ is defined by means of the position and orientation of the mobile platform. For this reason, an accurate measure of the actual $\vec{X}$ is essential for developing the hybrid controller proposed.
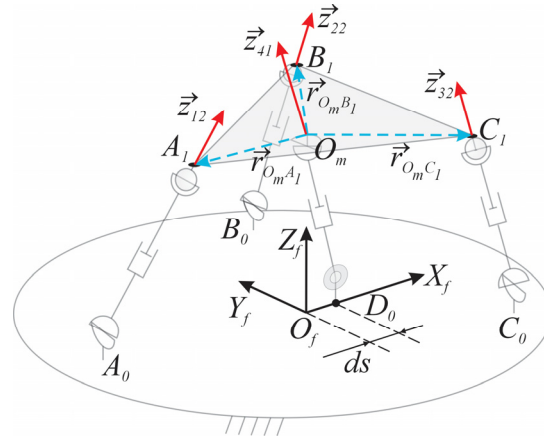

Figure 2. TWSs in the 3U$\underline{P}$S+R$\underline{P}$U PR.

2.4. 3D Tracking System

To be able to capture the movements of the mobile platform of the PR, a 3D tracking system (3DTS) based on artificial vision was used. The system consists of 10 Flex13 cameras from the manufacturer OptiTrack (Corvallis, OR, USA). These cameras use the infrared emission principle to be able to capture and detect the reflection that it creates on markers made of reflective 3M material.

Figure 3 shows the Robotics Laboratory and some cameras of the 3DTS used in this work. The cameras have a 1.3 Megapixel resolution and a capture velocity of 120 Hz. They have a latency or frame delay of 8.3 ms. The set of 10 cameras and the use of high-quality 14 mm markers make it possible to obtain an accuracy of more than 0.1 mm.
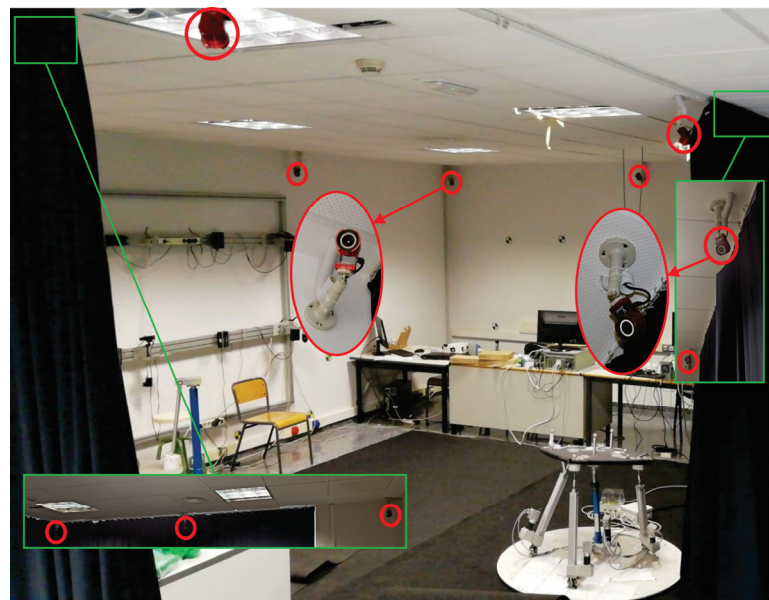

Figure 3. Robotics Laboratory equipped with the OptiTrack 3DTS.

The cameras are connected to two OptiHub2 devices. The OptiHub2 allows higher and more consistent power delivery to cameras for enhanced tracking range, simpler camera setup and cabling, and support for camera synchronization. The OptiHub2 devices are connected to high-speed USB ports in the camera control computer, and this computer

communicates with the robot control computer using an Ethernet connection. The Figure 4 below shows the architecture of the OptiTrack 3DTS of the laboratory.
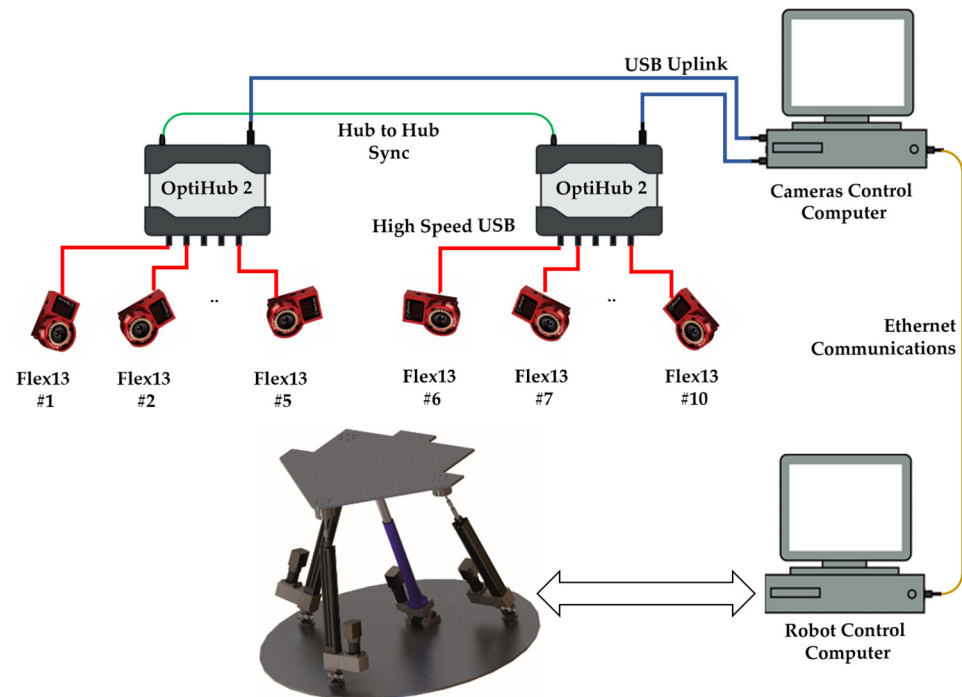

Figure 4. Laboratory OptiTrack 3DTS architecture.

The Motive Tracker software (Motive) from the same manufacturer, OptiTrack, is used on the camera control computer. This software is used to perform vision system calibration and obtain 6-DOF positioning results of objects within the tracking area. Motive uses high-level tracking filters and constraints to fine tune the performance of the high-speed object tracking. Motive associates a custom set of markers to a virtual element called rigid body and offers data access at any stage in the pipeline, i.e., 2D camera images, marker centroid data, labelled markers, and rigid bodies. In addition, it is possible to completely replace the Motive user interface and directly control the system operation in a new application with the NatNet SDK.

NatNet's client/server architecture allows client applications to run on the same system as the tracking software (Motive), on separate system(s), or both. The SDK integrates seamlessly with standard APIs (C/C++/.NET), tools (Microsoft Visual Studio), and protocols (UDP/Unicast/Multicast). Using the NatNet SDK, developers can quickly integrate OptiTrack motion tracking data into new and existing applications, including custom plugins for third-party applications and engines for real-time streaming. In addition, this SDK provides a .NET interface and sample programs that work directly with MATLAB core, requiring no additional MATLAB modules. Figure 5 summarizes the software architecture of the 3DTS used in this study.

Regarding the experimental setup, each camera individually builds a 2D image based on the markers' location, so a calibration process is required prior to the experiments in order to ensure that the system correctly reconstructs the 3D position of every marker.

The first step involved in this process is the correct orientation of the cameras to aim them at the workspace and, specifically, at the tracking volume, which, in this case, is the 4-DOF PR. Since the robot always operates in the same location and its workspace is limited, no changes in the camera location or orientation are required, and, therefore, they remain in the same position from the moment they are installed.
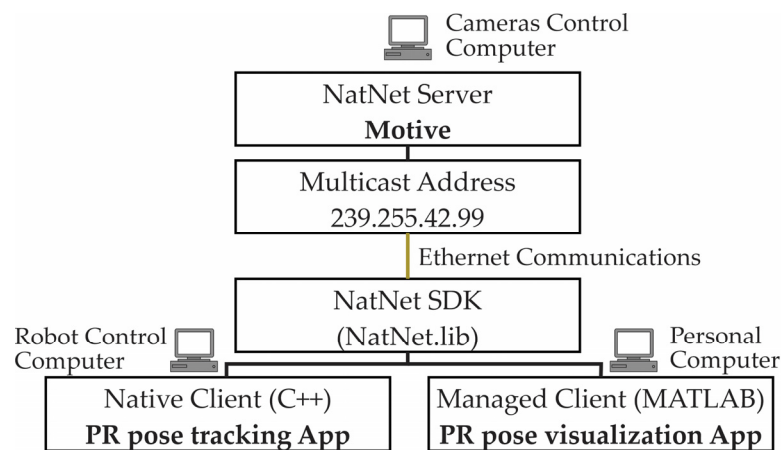
Figure 5. Software architecture of the OptiTrack 3DTS.

Another aspect to control is the brightness and illumination of the scene, as this allows the markers to be visible for the cameras, and, as such, no other unwanted objects are detected. Since the lighting conditions are the same for all experiments, some configuration parameters of the cameras, such as the exposure time, the gain, and the threshold, are set at constant values for all cameras using the software. If any intrusive markers are detected, they can either be manually covered by a cloth or masked in the software before performing the calibration.

After configuring the cameras, the calibration process starts with an empty scene where no markers should be detected, except for those attached to the calibration wand.

By moving the calibration wand, which is provided by OptiTrack, around the workspace, the cameras provide successive 2D projections of the markers. The 2D projections are used to compute the relative position of the cameras. The software shows the increasing precision of this estimation as the process progresses (Figure 6), and when a high enough quality is achieved, the process is manually stopped.
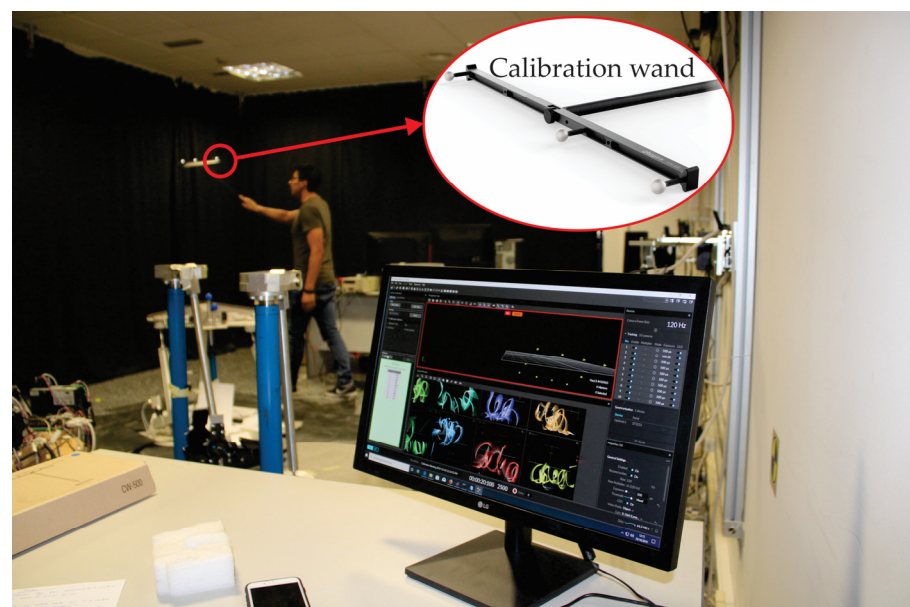

Figure 6. Calibration wand and experiment to determine the location of the markers.

The second tool, which concludes the calibration process, is the calibration square shown in Figure 7. This object includes three markers in right angle that define the origin and axes of the world coordinate system (also called ground plane by Motive). The ground plane is placed on the floor within the workspace area in such a way that its markers can

be visible by as many cameras as possible. This tool incorporates a level to ensure its horizontal position.

Figure 7. Calibration square.

Although all the cameras remain in the same position, minor movements of any of the cameras between experiments (for example, due to vibrations) can lead to poor tracking performance. For that reason, the calibration process must be performed once a day to ensure reliable 3D tracking. The calibration steps take no more than five minutes. After calibrating the cameras, Motive starts streaming data from all rigid bodies within the workspace. Rigid bodies are a set of 3 or more (maximum 20) markers whose relative distances remain constant. In this research, there are 2 rigid bodies represented by the fixed and mobile platform, respectively, and a set of 4 markers was attached to them. Three of the markers describe the cartesian coordinate frame of both platforms, and the fourth is added in a random (but known) position. If one of the markers is missed by the software during an experiment, the other three make it possible to reconstruct its position and keep streaming enough accurate data.

In the PR pose tracking App presented in this paper, the NatNet SDK provides a client class to communicate with the Motive server. A data handler is attached to this client, which works as a call-back that is executed every time there is a new frame of data available from the server. This handler has been used for retrieval of the x, y, z position of three markers placed on the fixed platform and another three placed on the mobile platform of the PR. Given the coordinates of the six markers, the actual position and orientation of the mobile platform ($\vec{X}_c$) are calculated with respect to the $\{O_f - X_F Y_F Z_F\}$ coordinate frame. Finally, $\vec{X}_c$ is sent through ROS2 messages to feed the control system. A MATLAB program has also been designed to provide an online view of $\vec{X}_c$ and calculate the actual actuator's length by solving the inverse kinematics. Figure 8 presents the graphic user interface (GUI) for online measures of $\vec{X}_c$. It is important to note that this program is independent of the control system (and therefore runs in a personal computer) and simply offers a viewing tool.
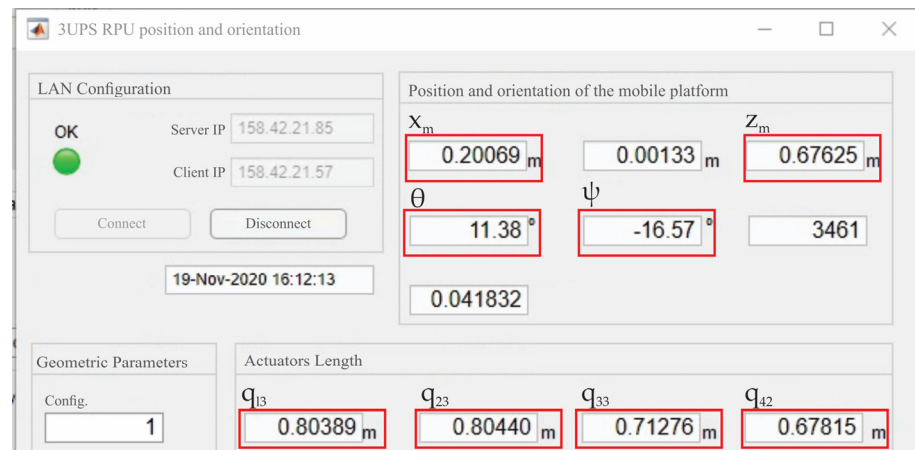
Figure 8. GUI for position and orientation tracking designed in MATLAB.

2.5. Hybrid Controller Description

If a PR reaches a type II singularity, a controller must move the actuators to release the PR from the singularity, maintaining a minimum deviation from the original configuration. Therefore, a method to identify the best set of actuators to be moved is needed. The index $\Omega_{i,j}$, using the position and orientation of the mobile platform, is able to identify the actuators involved in the type II singularity. However, in a type II singularity, the measurement of the actual position and orientation of the PR require an external sensor, such as a 3DTS. For this reason, a novel controller able to release the PR under study from a type II singularity using the index $\Omega_{i,j}$ and a 3DTS is proposed. It is important to note that this is the first time that the index $\Omega_{i,j}$ is employed as an online proximity detector to a type II singularity.

The novel vision-based hybrid controller to release the 3UPS+RPU PR from a type II singularity is shown in Figure 9. The hybrid controller combines two-level closed loops: an algebraic algorithm (inner loop) and a type II singularity releaser (outer loop). The type II singularity releaser calculates the $\Omega_{i,j}$ indices using the position and orientation of the PR provides by the OptiTrack 3DTS.
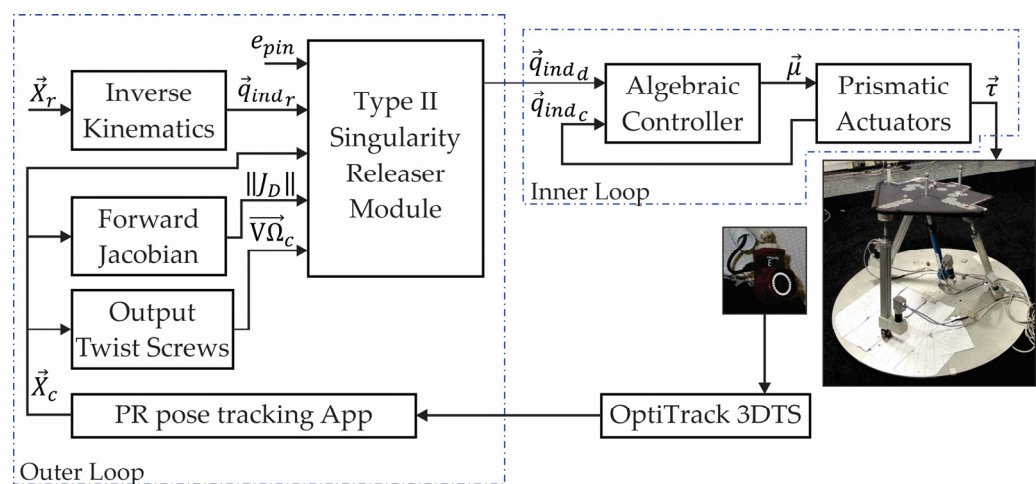

Figure 9. Hybrid controller architecture.

In the inner loop, the control signals ($\vec{\mu}$) to track the desired actuator's location $\vec{q}_{ind_d}$ are calculated by an algebraic algorithm based on the measured location of the actuators $\vec{q}_{ind_c}$. The $\vec{\mu}$ is proportional to the forces ($\vec{\tau}$) applied by the linear actuators to move the mobile platform.

In the outer loop, the reference location of the actuators ($\vec{q}_{ind_r}$) is obtained by solving the inverse kinematics for a knee rehabilitation trajectory ($\vec{X}_r$), and $\vec{X}_r$ is designed for the 4-DOF of the 3U$\underline{P}$S+R$\underline{P}$U PR.

Based on the $\vec{X}_c$ measured by the 3DTS, the proximity to a type II singularity is detected by $\overrightarrow{V\Omega}_c$ and $\|J_D\|_c$ at every time step. $\overrightarrow{V\Omega}_c$ stores the six $\Omega_{i,j}$ indices as $[\Omega_{1,2} \quad \Omega_{1,3} \quad \Omega_{1,4} \quad \Omega_{2,3} \quad \Omega_{2,4} \quad \Omega_{3,4}]^T$. If the 3U$\underline{P}$S+R$\underline{P}$U PR gets close to a type II singularity, $\vec{q}_{ind_r}$ is modified to define $\vec{q}_{ind_d}$. In Figure 9, the type II singularity releaser module (SRM) calculates the desired location of the actuator as follows:

$$\vec{q}_{ind_d} = \vec{q}_{ind_r} + v_d \cdot t_s \cdot \overrightarrow{\Delta\iota} \tag{9}$$

where v_d is the releasing velocity module for each actuator, t_s stands for the controller sample time, and $\overrightarrow{\Delta\iota}$ represents an integer vector that counts the deviation required in the F actuators to release the PR from a type II singularity.

The SRM calculates $\vec{q}_{ind_d}$ at every time step, although $\overrightarrow{\Delta\iota}$ is modified only if an enable pin (e_{pin}) is activated. Two versions of the algorithms have been proposed to contrast the results when (i) moving the actuators that cause the singularity and (ii) moving the actuators that do not cause the singularity according to $\overrightarrow{V\Omega}_c$.

The first version (SRM-V1) releases a PR from a type II singularity by moving the limbs identified by $\min\Omega_c$, which represents the minimum value of $\overrightarrow{V\Omega}_c$. If $\min\Omega_c$ or $\|J_D\|_c$ is lower than a certain limit, Ω_{lim} and $\|J_D\|_{lim}$, respectively, the two rows of $\overrightarrow{\Delta\iota}$ that have to change are identified by $\vec{\iota}_{ch}$. The possible change combinations for the two rows of $\overrightarrow{\Delta\iota}$ are defined by the columns of M_{inc} as follows:

$$\mathrm{M}_{inc} = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{10}$$

where 0, 1, and −1 correspond to the stop, unit forward motion, and unit backward motion commands for an actuator, respectively.

For each column of M_{inc}, an auxiliary variable $\vec{q}_{ch}$ is initialized as $\vec{q}_{ind_d}$, and then its elements indexed by $\vec{\iota}_{ch}$ are modified using the current M_{inc} column. Then, it is checked that this position is confined within the geometrical limits. If $\vec{q}_{ch}$ is inside the actuators' displacement range, the forward kinematic problem is solved ($\vec{X}_{ch}$). Next, the angles reached by the spherical joints ($\vec{\alpha}_{ch}$) are calculated. If $\vec{\alpha}_{ch}$ is within the working range, a new $\Omega_{i,j}$ is calculated for the limbs identified by $\vec{\iota}_{ch}$, and it is added to $\overrightarrow{V\Omega}_{ch}$. Then, $\overrightarrow{\Delta\iota}$ takes the value of the column of M_{inc} that produces the maximum element of $\overrightarrow{V\Omega}_{ch}$ ($\max\Omega_{ch}$), as that combination contributes the most to releasing the singularity without exceeding any range limit. Finally, $\vec{q}_{ind_d}$ is updated using the new $\overrightarrow{\Delta\iota}$.

An alternative algorithm called SRM-V2 has been proposed to test the behaviour when moving the wrong limbs. It modifies the rows of $\overrightarrow{\Delta\iota}$ that are not related to $\min\Omega_c$ ($\vec{\iota}_{nc}$) to release the PR from the type II singularity caused by the actuators $\vec{\iota}_{ch}$. SRM-V2 is designed to verify that moving the actuators identified by $\min\Omega_c$ is the best way to release the 3U$\underline{P}$S+R$\underline{P}$U PR from a type II singularity.

The complete process performed by SRM-V1 is described in the pseudocode shown in Algorithm 1, where SRM-V2 is obtained by adding and replacing, the lines marked with $*$ and $**$, respectively. A description of the variables used in Algorithm 1 is presented in Table 2.

Table 2. Description of parameters, inputs, and outputs of SRM-V1 and SRM-V2.

Parameters		
Variable	Description	Default
v_d	releasing velocity module in m/s	0.01
t_s	controller sample time in s	0.01
$\|J_D\|_{lim}$	experimental limit for $\|J_D\|$	0.015

Ω_{lim}	experimental limit for $\Omega_{i,j}$	1.8°
$\overrightarrow{max q}_{ind}$	maximum feasible values for the actuators' length in m, $4x1$ vector	$[0.93 \quad 0.92 \quad 0.93 \quad 0.82]^T$
$\overrightarrow{min q}_{ind}$	minimum feasible values for the actuators' length in m, $4x1$ vector	$[0.65 \quad 0.64 \quad 0.65 \quad 0.54]^T$
$\vec{\alpha}_{lim}$	experimental limits for the spherical joints, $3x1$ vector	38°
M_{inc}	possible increments/decrements for $\overrightarrow{\Delta i}$	See equation (10)
$\overrightarrow{\Delta i}$	column vector $4x1$, persistent variable	-

Inputs		
Variable	**Description**	**Default**
e_{pin}	enable pin	-
$\|J_D\|_c$	determinant of the forward Jacobian matrix, feedback signal	-
$\overrightarrow{V\Omega}_c$	column vector with the six $\Omega_{i,j}$ indices, feedback signals	-
$\vec{X}_c$	position and orientation of the mobile platform, feedback signal	-
$\vec{q}_{ind_r}$	trajectory for the actuators, reference signal	-

Outputs		
Variable	**Description**	**Default**
$\vec{q}_{ind_d}$	trajectory for the actuators, desired signal	-

---

Algorithm 1. Initialization

$$\overrightarrow{\Delta i} = \vec{0}$$

N_{ch} =number of columns of M_{inc}.
BEGIN
$\vec{q}_{ind_d} = \vec{q}_{ind_r} + v_d \cdot t_s \cdot \overrightarrow{\Delta i}$
IF $e_{pin} == true$
  $\min\Omega_c = $ minimum element in $\overrightarrow{V\Omega}_c$
  IF $\min\Omega_c < \Omega_{lim}$ OR $\|J_D\|_c < \|J_D\|_{lim}$
    IF $\min\Omega_c == \overrightarrow{V\Omega}_c(1)$
      $\vec{i}_{ch} = [1 \quad 2]$
      $* \ \vec{i}_{nc} = [3 \quad 4]$
    ELSEIF $\min\Omega_c == \overrightarrow{V\Omega}_c(2)$
      $\vec{i}_{ch} = [1 \quad 3]$
      $* \ \vec{i}_{nc} = [2 \quad 4]$
    ELSEIF $\min\Omega_c == \overrightarrow{V\Omega}_c(3)$
      $\vec{i}_{ch} = [1 \quad 4]$
      $* \ \vec{i}_{nc} = [2 \quad 3]$
    ELSEIF $\min\Omega_c == \overrightarrow{V\Omega}_c(4)$
      $\vec{i}_{ch} = [2 \quad 3]$
      $* \ \vec{i}_{nc} = [1 \quad 4]$
    ELSEIF $\min\Omega_c == \overrightarrow{V\Omega}_c(5)$
      $\vec{i}_{ch} = [2 \quad 4]$
      $* \ \vec{i}_{nc} = [1 \quad 3]$
    ELSE $\min\Omega_c == \overrightarrow{V\Omega}_c(6)$
      $\vec{i}_{ch} = [3 \quad 4]$
      $* \ \vec{i}_{nc} = [1 \quad 2]$
    ENDIF
    $\overrightarrow{V\Omega}_{ch} = $ column vector of N_{ch} zeros
    FOR $c_1 = 1:N_{ch}$

$$\vec{q}_{ch} = \vec{q}_{ind_d}$$

$$\vec{q}_{ch}(\vec{\iota}_{ch}) = \vec{q}_{ch}(\vec{\iota}_{ch}) + v_d \cdot t_s \cdot \mathrm{M}_{inc}(:, c_1)$$

$$** \ \vec{q}_{ch}(\vec{\iota}_{nc}) = \vec{q}_{ch}(\vec{\iota}_{nc}) + v_d \cdot t_s \cdot \mathrm{M}_{inc}(:, c_1)$$

IF $\overline{min}\vec{q}_{ind} < \vec{q}_{ch} < \overline{max}\vec{q}_{ind}$ (element-wise comparison)

　$\vec{\mathrm{X}}_{ch} =$ Solve the Forward Kinematics for $\vec{q}_{ch}$, using $\vec{X}_c$ as initial condition

　$\vec{\alpha}_{ch} =$ Angle of spherical joints for $\vec{\mathrm{X}}_{ch}$

　IF $\vec{\alpha}_{ch} < \vec{\alpha}_{lim}$ (element-wise comparison)

　　$\overrightarrow{\mathrm{V}\Omega}_{ch}(c_1) =$ Calculate the index $\Omega_{i,j}$ for $\vec{\mathrm{X}}_{ch}$ with $i, j = \vec{\iota}_{ch}$

　ENDIF

ENDIF

ENDFOR

$c_1 = argmax(\overrightarrow{\mathrm{V}\Omega}_{ch})$

$\overrightarrow{\Delta\iota}(\vec{\iota}_{ch}) = \overrightarrow{\Delta\iota}(\vec{\iota}_{ch}) + \mathrm{M}_{inc}(:, c_1)$

$\vec{q}_{ind_d} = \vec{q}_{ind_r} + v_d \cdot t_s \cdot \overrightarrow{\Delta\iota}$

ENDIF

ENDIF

END

Due to the properties of the index $\Omega_{i,j}$, the SRM algorithm has the advantage of moving a pair of F actuators simultaneously in each time step of the controller. For this reason, the SRM reduces the consumption of computing resources and the difference between $\vec{q}_{ind_d}$ and the original $\vec{q}_{ind_r}$.

3. Results

This section begins with a detailed description of the simulation setup, including the singular trajectories to be tested with SRM-V1 and SRM-V2 versions of the hybrid controller. Next, the performance of the hybrid controller in simulation is evaluated, where SRM-V1 appears to be better than SRM-V2. Subsequently, the experimental setup and the features of the actual 3UPS+RPU PR are detailed. Finally, the main experimental results show the effectiveness of the hybrid controller using SRM-V1 to release the PR under study from a type II singularity.

3.1. Simulation of the Vision-Based Hybrid Controller

Prior to implementing the algorithm on the actual PR, some simulations are performed on a kinematic and dynamic model of the 3UPS+RPU PR designed in MATLAB/Simulink. In both simulation and experimentation, the PR is moved from the initial position to a singular test configuration without activating the releaser. Then, it remains in the singular configuration for 15 s, after which the loop of the SRM is activated via e_{pin}. In that moment, one of the SRMs in Section 2.5. is launched based on the assumption that it will help release the robot from the type II singularity. The SRM launched has a lapse of 15s, allowing it to move the PR under study to a non-singular configuration.

Due to the lack of a simulated model of the 3DTS (see Figure 9) for MATLAB/Simulink, $\vec{X}_c$ is calculated directly by solving the forward kinematic problem. The main objective of the simulation is to test that the novel hybrid controller increases the values of $\|J_D\|$ and $\Omega_{i,j}$ in the vicinity of a type II singular configuration; i.e., it is able to release the PR under study from the type II singularity.

Since the 3UPS+RPU PR was built to interact with human knees, it is used to execute three rehabilitation movements: flexion of the hip, flexion–extension of the knee, and internal–external rotation of the knee [19]. This study, combining these three fundamental movements for simulation and experimentation, performs five knee rehabilitation trajectories ending in a type II singular configuration (see Table 3). The singular configurations of these five trajectories have $\|J_D\|$ and $\Omega_{i,j}$ close to zero but not exactly zero, avoiding

several forward kinematic solutions in the simulation. All five knee trajectories are designed with constant velocity; in this case, the translational DOFs move at 0.02 m/s and the rotational ones at 0.03 rad/s.

Table 3. Description of the trajectories with a type II singularity at the end.

Trajectory	Description	Type II Singularity			
		x_m (m)	z_m (m)	θ (rad)	ψ (rad)
1	Hip flexion	0.01	0.70	0.15	0.31
2	Partial internal–external knee rotation	0.01	0.70	−0.02	0.14
3	Flexion–extension of the knee combined with ankle and knee rotations	0.05	0.72	−0.01	0.15
4	Flexion–extension of the knee combined with hip flexion	0.12	0.77	−0.06	0.11
5	Complete internal–external knee rotation	−0.05	0.73	0.10	0.33

The simulation of the five knee rehabilitation trajectories verifies that SRM-V1 and SRM-V2 release the 3U$\underline{P}$S+R$\underline{P}$U PR from a singular configuration. Figure 10 shows how the type II singularity indices $\|J_D\|_c$ and $min\Omega_c$ increase when e_{pin} is activated for trajectory 1. These results verify from an analytical perspective that the hybrid controller proposed releases the 3U$\underline{P}$S+R$\underline{P}$U PR from a type II singularity.
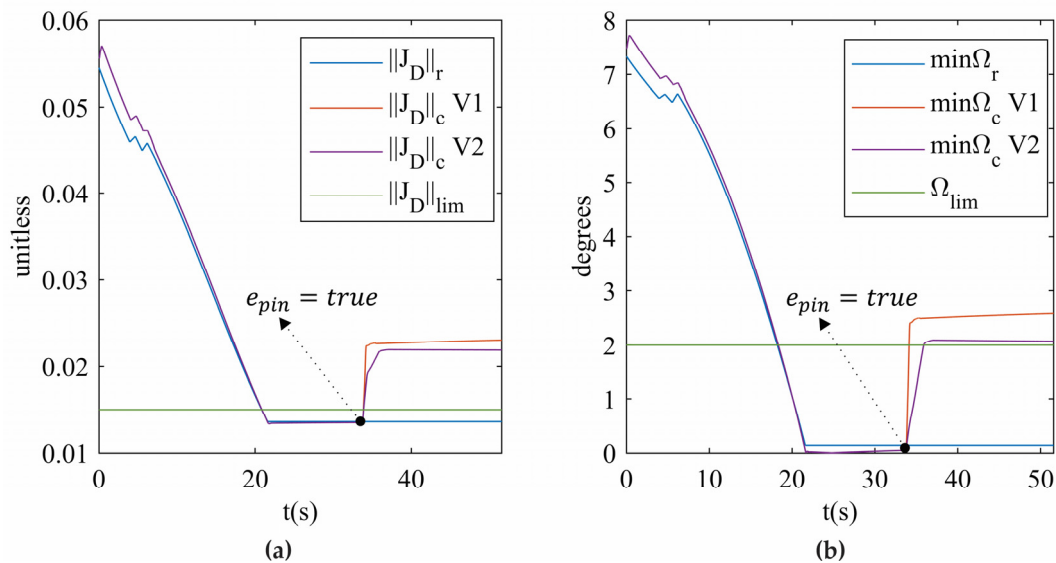

Figure 10. (**a**) $\|J_D\|$ (**b**) $min\Omega$ for trajectory 1 in the simulation.

The performance of the proposed hybrid controller in tracking $\vec{q}_{ind_r}$ is evaluated by three overall measures:

- The mean absolute error (MAE)

$$MAE = \frac{1}{F}\sum_{i=1}^{F}\left(\frac{1}{n}\sum_{j=1}^{n}|q_{ind_r}(i,j) - q_{ind_c}(i,j)|\right) \qquad (11)$$

- The mean absolute percentage error (MAPE)

$$\text{MAPE} = \frac{100}{F} \sum_{i=1}^{F} \left(\frac{1}{n} \sum_{j=1}^{n} \left| \frac{q_{ind_r}(i,j) - q_{ind_c}(i,j)}{q_{ind_r}(i,j)} \right| \right) \tag{12}$$

- The mean distance travelled for type II singularity release (MDSR)

$$\text{MDSR} = \frac{1}{F} \sum_{i=1}^{F} \left(\sum_{j=1}^{n} \left| q_{ind_r}(i,k) - q_{ind_c}(i,j) \right| \right) \tag{13}$$

where n is the number of samples taken after the activation of e_{pin} at instant k, and i and j are the actuator and the time instant, respectively.

Table 4 shows the MAE, MAPE, and MDSR results for the simulation of the hybrid controller with SRM-V1 and SRM-V2. In this table, the MAE and the MAPE show that SRM-V1 has less error in position tracking than that of SRM-V2 during release from the type II singularity. In addition, the MDSR shows that SRM-V1 needs fewer movements of the actuators than SRM-V2 to release the PR from a singular configuration. These results show that moving the pair of actuators identified by the index $\Omega_{i,j}$ (SRM-V1) is the best option to release a PR from a type II singularity.

Table 4. Performance of the hybrid controller using SRM-V1 and SRM-V2 in the simulation.

Trajectory	MAE (mm)		MAPE (%)		MDSR (mm)	
	SRM-V1	**SRM-V2**	**SRM-V1**	**SRM-V2**	**SRM-V1**	**SRM-V2**
1	3.87	10.74	0.53	1.40	7.01	18.18
2	1.09	2.04	0.14	0.28	5.05	2.92
3	1.77	6.15	0.24	0.82	4.78	6.74
4	3.00	10.24	0.38	1.25	7.48	10.81
5	10.74	10.44	1.43	1.37	15.47	35.23
MEAN	**4.09**	**7.92**	**0.54**	**1.02**	**7.95**	**14.77**

3.2. Experimentation of the Vision-Based Hybrid Controller

After testing the novel vision-based controller in simulation, the next step is implementing the hybrid controller on the real robot according to the diagram shown in Figure 9. Although both the simulation and experimentation have the same procedure, the experimentation presents two notable differences:

- $\vec{X}_c$ is provided by processing the data stream from the 3DTS in real time.
- During the 15 s before the SRM is activated, an external perturbation is applied to the PR. Since in a type II Singularity the PR can vary its position and orientation without moving any actuators, the researcher can apply some forces to the PR by hand to check whether the mobile platform experiences uncontrolled motion.

In the experimental context, the type II singularity release can be tested by trying to move the PR by hand before (when the PR is expected to move) and after the SRM is activated. After the activation of the SRM, the 3U$\underline{P}$S+R$\underline{P}$U PR will regain the stiffness required to ensure safe interaction with a patient.

Regarding the actual robot, the external limbs are driven by Festo DNCE 32-BS10 prismatic actuators, and the central limb is driven by a NIASA M100-F16 prismatic actuator. All the actuators are attached to Maxon 148867 150 W DC motors commanded by ESCON 50/5 servo controllers, which control the current by means of pulse width modulation (PWM). The current is proportional to the applied voltage (which comes from the control actions), and the torque is in turn proportional to the current. The DC motors are equipped with incremental encoders with a resolution of 500 counts per turn.

The control unit is connected to an industrial computer using acquisition cards. A PCI 1784 Advantech card is used to read the position from the encoders, having four 32-

bit quadruple AB phase encoder counters. On the other hand, a 12-bit, 4-channel PCI 1720 Advantech card is used to send the control actions $\vec{\mu}$.

The proposed vision-based hybrid controller runs on the Robot Operating System 2 (ROS2) [31,32]. The two levels of the hybrid controller and the processing of the data stream from the 3DTS are implemented in a modular way using the C++ and Python programming languages. The controller receives the set of references $\vec{q}_{ind_r}$ from the solution of the inverse kinematics given the Cartesian references for the end-effector. The $\vec{q}_{ind_r}$ is sampled at a rate of 100 Hz, and the desired releasing velocity v_d is set to 0.01 m/s. These parameters are suitable for knee rehabilitation requirements.

For the actual PR, a fourth performance index is added to evaluate the smoothness of the movements performed by the controller, which is measured with the absolute variation rate (AVR) of the control actions as follows:

$$\text{AVR} = \frac{1}{F} \sum_{i=1}^{F} \left(\sum_{j=2}^{n} |\tau(i,j) - \tau(i,j-1)| \right) \tag{14}$$

During the first run of trajectory 1 using the hybrid controller with SRM-V2, the actual 3U$\underline{P}$S+R$\underline{P}$U PR reaches an AVR of 8N, which is too high for knee rehabilitation. For this reason, the experiment on the actual PR under study only focuses on the hybrid controller with SRM-V1. This decision is also supported by the better performance shown in the simulation (see Table 4).

Table 5 shows the results of performance tracking of $\vec{q}_{ind_r}$ of the hybrid controller with SRM-V1 implemented on the 3U$\underline{P}$S+R$\underline{P}$U PR. The MAE and MAPE for experimentation are similar to the simulation results, with a low AVR ensuring smooth movements of the mobile platform. In contrast, the actual MDSR is lower than the values calculated in the simulation. The reduction in MDSR is due to the accurate measure of $\vec{X}_c$ provided by the 3DTS, which is fundamental for a proper measure of the proximity to a type II singularity.

Table 5. Performance of the hybrid controller using SRM-V1 in the experimentation.

Trajectory	MAE (mm)	MAPE (%)	MDSR (mm)	AVR (N)
1	3.26	0.45	3.64	0.22
2	3.02	0.41	7.61	0.52
3	2.05	0.27	1.60	0.17
4	2.14	0.27	1.90	0.46
5	10.66	1.42	11.82	1.44
MEAN	**4.22**	**0.56**	**5.31**	**0.56**

Figure 11 shows the measures of the two indices ($\|J_D\|_c$ and $\min\Omega_c$) when the actual PR is released from a singular configuration, corresponding to trajectory 1 with $\Omega_{3,4}$ as $\min\Omega_c$. The variation of $\|J_D\|_c$ and $\min\Omega_c$ before SRM-V1 is activated is due to the external force applied to the actual PR. It is important to mention that the actual PR recovers its stiffness at the end of all experiments. To the best of the author's knowledge, this is the first time that an actual PR has been driven to a type II singularity and successfully released from it by using the index $\Omega_{i,j}$. The results can be seen in Video 1 and Video 2 provided as Supplementary Material of this research.
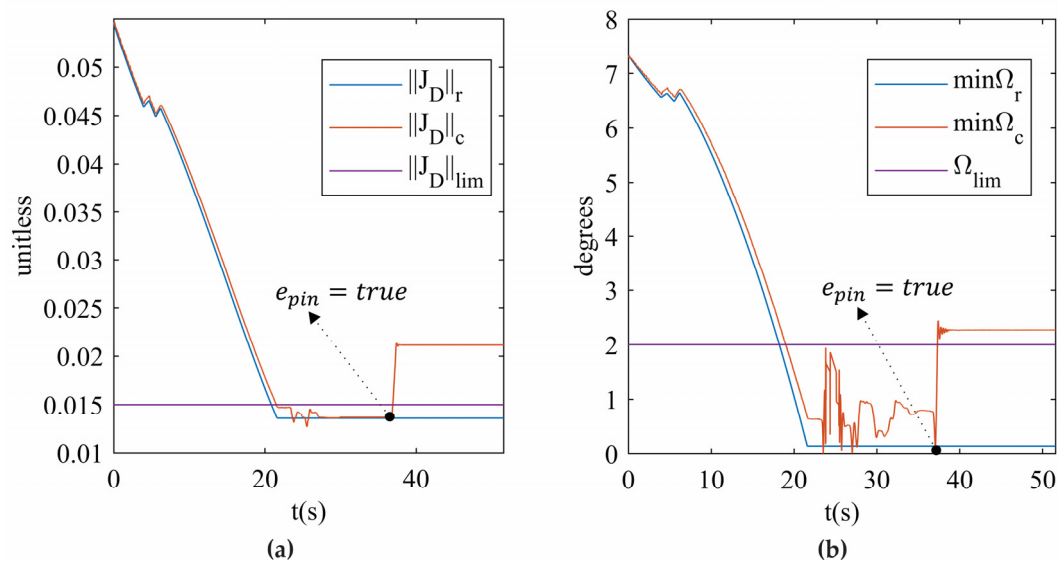
Figure 11. (**a**) $\|J_D\|$ (**b**) $\min\Omega$ for trajectory 1 in the experimentation.

Figure 12 shows the reference (r) trajectory for x_m in contrast to its estimation ($\hat{c}$) by using the forward kinematic model and the experimental measures (c) based on data streaming from the 3DTS. Despite both estimated and experimental measures being calculated online, only the experimental measure detects the movement produced by the external force applied to the PR. This verifies that when the 3UPS+RPU PR is in a type II singularity, the actual x_m cannot be determined by solving the forward kinematic.
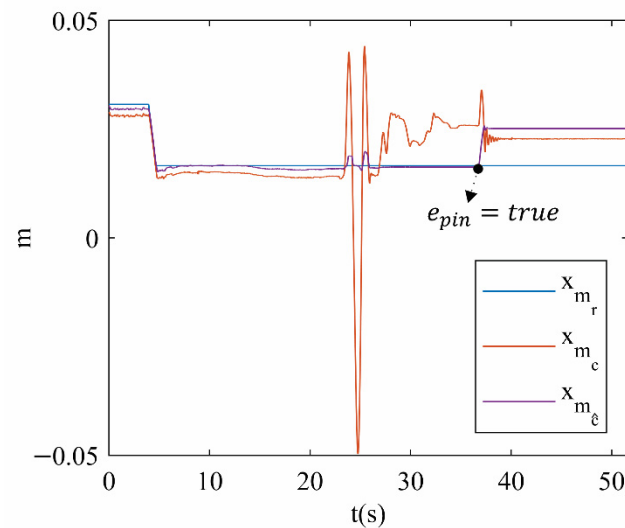

Figure 12. x_m position for trajectory 1.

Figure 13a shows the position for limb 3, which is one of the two limbs involved in the type II singularity in trajectory 1. In this figure, the measured position (c) accurately tracks the desired position (d), which differs from the reference (r) only after SRM-V1 activation. Furthermore, Figure 13a clearly shows that the desired position is modified by a few millimetres from the reference to release the actual PR from the type II singularity. Finally, Figure 13b shows the smooth control actions calculated by the hybrid controller implemented on the actual PR using SMR-V1. Video 1 provides an interactive view of the results presented in Figures 12 and 13 and can be found in the Supplementary Material Section.
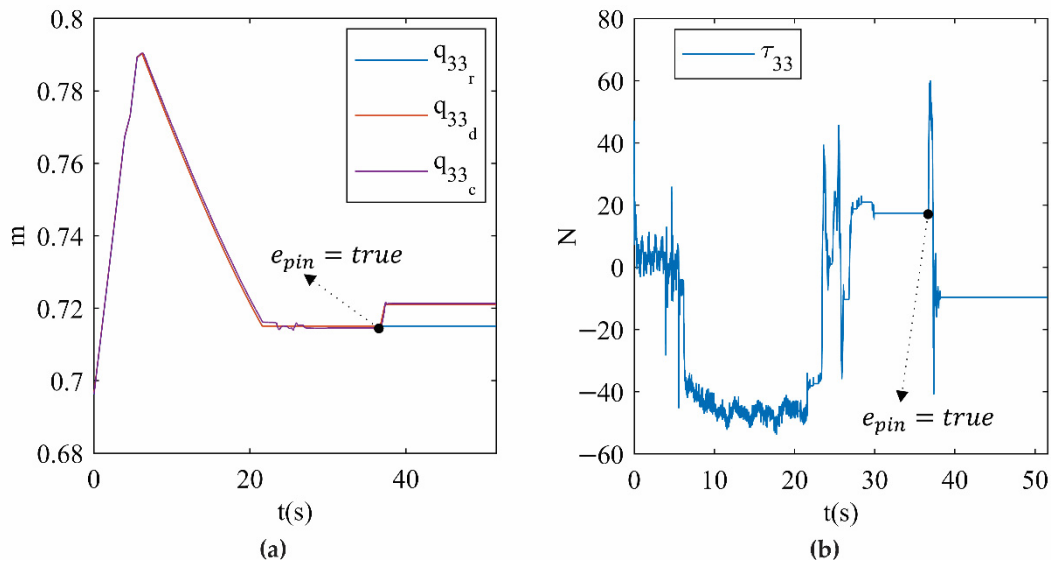
Figure 13. (a) q_{ind} (b) τ on limb 3 for trajectory 1.

The experimental results conclude that the vision-based hybrid controller with SMR-V1 releases an actual PR from a type II singularity with minimum deviation from the original reference. In addition, the OptiTrack 3DTS allows the hybrid controller with SMR-V1 to take advantage of the features of the index $\Omega_{i,j}$.

4. Discussion

This study has addressed the novel task of releasing a 4-DOF PR from type II singular configuration using the index $\Omega_{i,j}$ to identify the limbs involved in the singularity. The hybrid controller proposed combines an algebraic controller with an external computational loop that modifies the joint references only for the limbs that are causing the singularity. This mechanism can be activated whenever the robot enters into a type II singularity by measuring the $\|J_D\|_c$ and $\min\Omega_c$. Both $\|J_D\|_c$ and $\min\Omega_c$ are measured based on the actual position and orientation of the mobile platform that is provided online by a OptiTrack 3DTS. The embedded sensorization includes a set of encoders attached to the motors to ascertain the joint positions.

To show the effectiveness of the designed method, several experiments have been conducted with trajectories that leave the robot in distinct singular configurations, where the releasing algorithm is activated. This scheme has been implemented in both simulation and actual settings to compare the differences in performance when moving the limbs involved (SRM-V1) or not involved (SRM-V2) in the type II singularity. The algorithm for SRM-V1 and SRM-V2 defined the movement of the actuators based on the results of $\min\Omega_c$.

The results of the simulation in Section 3.1 clearly show that SRM-V1 makes the robot behave better in terms of all the performance measures with respect to SRM-V2. According to Table 4, SRM-V1 presents a 0.54% (4.09 mm) mean error in tracking the original reference with a mean distance travelled of 7.95 mm for releasing the PR from a type II singularity. These errors are approximately half of those obtained with SRM-V2, thus verifying that moving the actuators identified by $\min\Omega_c$ is the best option to release the 3U$\underline{P}$S+R$\underline{P}$U PR from a type II singularity. In fact, no trajectories were performed with the actual robot using SRM-V2, as a first experiment using this algorithm showed that the robot was struggling to get out of the singular configuration, with sharper control actions than those obtained in simulation.

Section 3.2 shows that by using knowledge of the true position and orientation of the mobile platform, the hybrid controller with SRM-V1 can successfully release the actual PR

from a type II singularity. All singular trajectories were overcome, even in the cases where the mobile platform was manipulated to change its position during the standby time. The results show how the simulated and real experiments are alike, as all of the indicators for SMR-V1 are somewhat similar. These errors are proven to be dependent on the starting singular configuration, since trajectory 5 is harder for the PR to overcome.

Based on the results of simulation and experimentation, this is the first use of a vision-based hybrid controller capable of releasing a 4-DOF PR from a singular configuration. It is also notable that the effectiveness of the release from a type II singularity with a minimum deviation of the original reference depends on $\min\Omega_c$. The smoother response of the vision-based hybrid controller is achieved because of the accurate measures of the 3DTS, making it a fundamental element of the hybrid controller. It is important to highlight that before this research, the $\Omega_{i,j}$ had not been used as an online detector of the proximity to type II singularities for controlling purposes.

The proposed vision-base hybrid controller compensates a main drawback of PRs, and it represents a step forward towards compliant manipulation of PRs. This system improves the performance of knee rehabilitation tasks by ensuring the safety of the patient during human–robot interaction, even if the PRs arise a type II singularity.

In future research, SRM-V1 can be extended for its use in the task of type II singularity avoidance, i.e., preventing the PR from entering into a singular configuration. Although little literature exists regarding this field, the SRM-V1 algorithm offers valuable insight into the limbs that are expected to lead the robot to a type II singularity. After adding the possibility of returning to the original reference to SRM-V1, the avoidance of type II singularities could be achieved in a more reliable way than using other methods such as artificial potential fields.

Supplementary Materials: The following videos are available online at https://im-bio3r.ai2.upv.es/videos/TypeII_singularities: Video 1: 4-DOF parallel robot: vision-based hybrid controller to release from a type II singularity. Trajectory 1; Video 2: 4-DOF parallel robot: vision-based hybrid controller to release from a type II singularity. Trajectory 5.

Author Contributions: Conceptualization, V.M., M.V., and M.U.; methodology, V.M., M.V., and J.L.P.; software, R.J.E. and J.F.; validation, V.M. and M.U.; formal analysis, V.M.; investigation, J.L.P.; resources, V.M. and M.V.; data curation, R.J.E. and J.F.; writing—original draft preparation, M.V.; writing—review and editing, J.L.P.; visualization, M.V.; supervision, V.M.; project administration, V.M. and M.V.; funding acquisition, V.M. and M.V. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Briot, S.; Khalil, W. *Dynamics of Parallel Robots—From Rigid Links to Flexible Elements*; Springer: Berlin/Heidelberg, Germany, 2015; ISBN 978-3-319-19787-6.
2. Patel, Y.D.; George, P.M. Parallel Manipulators Applications—A Survey. *Mod. Mech. Eng.* **2012**, *2*, 57–64, doi:10.4236/mme.2012.23008.
3. Aliakbari, M.; Mahboubkhah, M. An adaptive computer-aided path planning to eliminate errors of contact probes on free-form surfaces using a 4-DOF parallel robot CMM and a turn-table. *Meas. J. Int. Meas. Confed.* **2020**, *166*, 108216, doi:10.1016/j.measurement.2020.108216.
4. Xie, S. *Advanced Robotics for Medical Rehabilitation: Current State of the Art and Recent Advances*; Springer: Cham, Switzerland, 2016; ISBN 978-3-319-19895-8 978-3-319-19896-5.

5. Díaz, I.; Gil, J.J.; Sánchez, E. Lower-Limb Robotic Rehabilitation: Literature Review and Challenges. *J. Robot.* **2011**, *2011*, 1–11, doi:10.1155/2011/759764.

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, doi:10.1109/ROBIO.2009.5420487.

7. Ai, Q.; Zhu, C.; Zuo, J.; Meng, W.; Liu, Q.; Xie, S.; Yang, M. Disturbance-Estimated Adaptive Backstepping Sliding Mode Control of a Pneumatic Muscles-Driven Ankle Rehabilitation Robot. *Sensors* **2017**, *18*, 66, doi:10.3390/s18010066.

8. Atashzar, S.F.; Shahbazi, M.; Patel, R.V. Haptics-enabled Interactive NeuroRehabilitation Mechatronics: Classification, Functionality, Challenges and Ongoing Research. *Mechatronics* **2019**, *57*, 1–19.

9. Kataoka, Y.; Takeda, R.; Tadano, S.; Ishida, T.; Saito, Y.; Osuka, S.; Samukawa, M.; Tohyama, H. Analysis of 3-d kinematics using h-gait system during walking on a lower body positive pressure treadmill. *Sensors* **2021**, *21*, 2619, doi:10.3390/s21082619.

10. Gosselin, C.; Angeles, J. Singularity Analysis of Closed-Loop Kinematic Chains. *IEEE Trans. Robot. Autom.* **1990**, *6*, 281–290, doi:10.1109/70.56660.

11. Park, F.C.; Kim, J.W. Singularity Analysis of Closed Kinematic Chains. *J. Mech. Des.* **1999**, *121*, 32–38, doi:10.1115/1.2829426.

12. di Gregorio, R.; Parenti-Castelli, V. Mobility analysis of the 3-UPU parallel mechanism assembled for a pure translational motion. In Proceedings of the 1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No.99TH8399), Atlanta, GA, USA, 19–23 September 1999; IEEE: New York, NY, USA, 1999; pp. 520–525.

13. Slavutin, M.; Shai, O.; Sheffer, A.; Reich, Y. A novel criterion for singularity analysis of parallel mechanisms. *Mech. Mach. Theory* **2019**, *137*, 459–475, doi:10.1016/j.mechmachtheory.2019.03.001.

14. Voglewede, P.A.; Ebert-Uphoff, I. Measuring "closeness" to singularities for parallel manipulators. In Proceedings of the IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April–1 May 2004; IEEE: New York, NY, USA, 2004; Volume 5, pp. 4539–4544.

15. Davidson, J.K.; Hunt, K.H.; Pennock, G.R. Robots and Screw Theory: Applications of Kinematics and Statics to Robotics. *J. Mech. Des.* **2004**, *126*, 763, doi:10.1115/1.1775805.

16. Yuan, M.S.C.; Freudenstein, F.; Woo, L.S. Kinematic Analysis of Spatial Mechanisms by Means of Screw Coordinates. Part 2—Analysis of Spatial Mechanisms. *J. Eng. Ind.* **1971**, *93*, 67, doi:10.1115/1.3427919.

17. Takeda, Y.; Funabashi, H. Motion Transmissibility of In-Parallel Actuated Manipulators. *JSME Int. J. Ser. C Dyn. Control Robot. Des. Manuf.* **1995**, *38*, 749–755, doi:10.1299/jsmec1993.38.749.

18. Wang, J.; Wu, C.; Liu, X.-J. Performance evaluation of parallel manipulators: Motion/force transmissibility and its index. *Mech. Mach. Theory* **2010**, *45*, 1462–1476, doi:10.1016/J.MECHMACHTHEORY.2010.05.001.

19. Pulloquinga, 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, doi:10.1016/j.mechmachtheory.2020.104242.

20. Scalera, L.; Carabin, G.; Vidoni, R.; Wongratanaphisan, T. Energy efficiency in a 4-dof parallel robot featuring compliant elements. *Int. J. Mech. Control* **2019**, *20*, 1–9.

21. Marchi, T.; Mottola, G.; Porta, J.M.; Thomas, F.; Carricato, M. Position and singularity analysis of a class of planar parallel manipulators with a reconfigurable end-effector. *Machines* **2021**, *9*, 7, doi:10.3390/machines9010007.

22. Llopis-Albert, C.; Rubio, F.; Valero, F. Optimization approaches for robot trajectory planning. *Multidiscip. J. Educ. Soc. Technol. Sci.* **2018**, *5*, 1, doi:10.4995/muse.2018.9867.

23. Bordalba, R.; Porta, J.M.; Ros, L. A Singularity-Robust LQR Controller for Parallel Robots. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, Madrid, Spain, 1–5 October 2018; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2018; pp. 5048–5054.

24. Agarwal, A.; Nasa, C.; Bandyopadhyay, S. Dynamic singularity avoidance for parallel manipulators using a task-priority based control scheme. *Mech. Mach. Theory* **2016**, *96*, 107–126, doi:10.1016/j.mechmachtheory.2015.07.013.

25. Lee, L.W.; Chiang, H.H.; Li, I.H. Development and control of a pneumatic-actuator 3-dof translational parallel manipulator with robot vision. *Sensors* **2019**, *19*, 28–30, doi:10.3390/s19061459.

26. Huynh, B.-P.; Su, S.-F.; Kuo, Y.-L. Vision/Position Hybrid Control for a Hexa Robot Using Bacterial Foraging Optimization in Real-time Pose Adjustment. *Symmetry* **2020**, *12*, 564, doi:10.3390/sym12040564.

27. Amarasinghe, D.; Mann GK, I.; Gosine, R.G. *Vision-Based Hybrid Control Strategy for Autonomous Docking of a Mobile Robot*; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2005; pp. 1600–1605.

28. 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, doi:10.1115/1.4037800.

29. 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, doi:10.1080/15397734.2017.1355249.

30. 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, doi:10.1016/j.mechatronics.2020.102382.

31. Maruyama, Y.; Kato, S.; Azumi, T. Exploring the performance of ROS2. In Proceedings of the Proceedings of the 13th International Conference on Embedded Software, EMSOFT 2016, Pittsburgh, PA, USA, 1–7 October 2016; Association for Computing Machinery, Inc.: New York, NY, USA, 2016; pp. 1–10.

32. Jiang, Z.; Gong, Y.; Zhai, J.; Wang, Y.P.; Liu, W.; Wu, H.; Jin, J. Message Passing Optimization in Robot Operating System. *Int. J. Parallel Program.* **2020**, *48*, 119–136, doi:10.1007/s10766-019-00647-w.