sensors

MDPI

Article

Passive Exercise Adaptation for Ankle Rehabilitation Based on Learning Control Framework

Fares J. Abu-Dakka [1,*], Angel Valera [2], Juan A. Escalera [3], Mohamed Abderrahim [4], Alvaro Page [5] and Vicente Mata [6]

[1] Intelligent Robotics Group, Department of Electrical Engineering and Automation (EEA), Aalto University, 02150 Espoo, Finland

[2] Instituto Universitario de Automática e Informática Industrial (ai2), Universitat Politècnica de València, 46022 Valencia, Spain; giuprog@isa.upv.es

[3] Instituto Nacional de Técnica Aeroespacial (INTA), 28330 San Martín de la Vega, Spain; escalerapja@inta.es

[4] Department of Systems Engineering and Automation, Carlos III University of Madrid, 28911 Leganés, Spain; mohamed@ing.uc3m.es

[5] Instituto Universitario de Ingeniería Mecánica y Biomecánica, Universitat Politècnica de València, 46022 Valencia, Spain; afpage@ibv.upv.es

[6] Departamento de Ingeniería Mecánica y de Materiales, Universitat Politècnica de València, 46022 Valencia, Spain; vmata@mcm.upv.es

* Correspondence: fares.abu-dakka@aalto.fi

Received: 17 September 2020; Accepted: 26 October 2020; Published: 31 October 2020

check for updates

Abstract: Ankle injuries are among the most common injuries in sport and daily life. However, for their recovery, it is important for patients to perform rehabilitation exercises. These exercises are usually done with a therapist's guidance to help strengthen the patient's ankle joint and restore its range of motion. However, in order to share the load with therapists so that they can offer assistance to more patients, and to provide an efficient and safe way for patients to perform ankle rehabilitation exercises, we propose a framework that integrates learning techniques with a 3-PRS parallel robot, acting together as an ankle rehabilitation device. In this paper, we propose to use passive rehabilitation exercises for dorsiflexion/plantar flexion and inversion/eversion ankle movements. The therapist is needed in the first stage to design the exercise with the patient by teaching the robot intuitively through learning from demonstration. We then propose a learning control scheme based on dynamic movement primitives and iterative learning control, which takes the designed exercise trajectory as a demonstration (an input) together with the recorded forces in order to reproduce the exercise with the patient for a number of repetitions defined by the therapist. During the execution, our approach monitors the sensed forces and adapts the trajectory by adding the necessary offsets to the original trajectory to reduce its range without modifying the original trajectory and subsequently reducing the measured forces. After a predefined number of repetitions, the algorithm restores the range gradually, until the patient is able to perform the originally designed exercise. We validate the proposed framework with both real experiments and simulation using a Simulink model of the rehabilitation parallel robot that has been developed in our lab.

Keywords: rehabilitation robots; parallel robots; dynamic movement primitives; iterative learning control; force control; motion control

1. Introduction

Nowadays, robots are present in many different areas. For instance, rehabilitation devices can be used as therapy aids, e.g., for the development of adjustable devices for assisting different sensorimotor functions [1,2], in schemes for improving therapeutic training [3], and for the assessment of patients' sensorimotor performance [4]. Assistive devices have also been developed [5].

Rehabilitation robotics brings scientists from human–robot interaction and biomedical engineering together with clinicians and therapists in order to develop the necessary technologies to improve patients' quality of life. The main goals here are [6]: (i) to develop implementable technologies that can be easily used by patients, therapists, and clinicians, (ii) to enhance the efficacy of clinicians' therapies, and (iii) to facilitate patients' daily activities. Moreover, using robotic systems, a very precise quantification of motion parameters can be provided by observing position, velocity, forces, etc. [7]. In order to achieve these goals, rehabilitation devices should meet some functional requirements, including: safety, stability, adaptability to the patient's needs, accommodating a wide range of patients, providing a complete Range of Motion (ROM), being equipped with the necessary sensors for haptic and visual feedback, etc. [8,9].

In order to examine the patients' level of adaptability while using rehabilitation devices, techniques such as passive exercise, active assisted exercise, active resistive exercise, active constrained exercise, and adaptive exercise [10] can be used, among others. Passive exercise needs no intervention by the patient and the motion is completely driven by the rehabilitation device. However, in active exercises, the patient actively interacts with the device and vice versa. Adaptive exercise refers to an excessive workout that the robot has never done, from which it tries to adapt to a new unknown pathway. During a rehabilitation treatment, cooperation between therapists and patients is required over many rehabilitation sessions in a clinic. Moreover, patients are required to continue the prescribed exercises at home. It has been documented that recovery, when using conventional treatment, is slow and sometimes takes more than a year [11]. Such a variety of rehabilitation techniques can achieve a certain level of improvement in the mobility of joints and limbs of the human body, such as the ankle joint. Some interesting examples of rehabilitation robots are MIT-MANUS for upper limb rehabilitation [11], LOKOMAT for gait training [12], and Parallel Robots (PRs) for ankle joint rehabilitation [13].

As mentioned above, the purpose of rehabilitation robotics is to design a device that mimics the work done by patient and physiotherapist during a rehabilitation session. In this paper, we propose to use a PR as a rehabilitation device. PRs have a high load capacity, stiffness, precision, compact structure, excellent energy/weight ratio, and provide better feedback control [14]. Interested readers can refer to [15–18] for more details about rehabilitation devices, modified isokinetic tables [19], gait training [20], upper limb rehabilitation [21,22], and ankle rehabilitation [23–25].

Regarding ankle rehabilitation systems, there are devices that generate ankle movements for neurological rehabilitation [2,26,27] or ankle sprains [13]. These devices require very precise control in order to reproduce precise movements. This control system should manage positions and forces during different exercises [28]. The Rutgers Ankle [29] was the first device used for ankle rehabilitation that provided a six-degrees-of-freedom (df) PR movement to the ankle joint. The robot applies assistive or resistive moments depending on whether the exercise is passive or active, respectively. For entertainment during exercises, the platform can be interfaced with game-like virtual environments [30]. The Rutgers Ankle is also being used to perform clinical trials for post-stroke rehabilitation [31]. Despite its use in research and experimentation, the device suffers from redundant actuations. For redundancy reduction, the authors of [13] proposed three-df and four-df PRs with a configurable central strut for sprained ankle treatments [13,32]. Different configurations of the central strut allowed the authors to analyze three different PRs in the stiffness domain.

The characteristics of the exercises to be performed in each case are very different. For that reason, a reconfigurable device is introduced in order to adapt to each patient's range of ankle motion [33]. This robot works on the metatarsophalangeal joint and its controller varies the impedance parameters in order to accommodate different exercise modes. A three-RSS (Revolute, Spherical, and Spherical) PR is proposed by [34] and validated in simulation for ankle rehabilitation. Syrseloudis and Emiris [26] introduced a tripod-based PR actuated by electric motors for ankle rehabilitation.

Actuation redundancy is used to deal with PR singularities [35]. Fan and Yin developed a four-df wearable PR [36]. In their design, the moving platform is linked to the patient's foot, while the fixed platform is attached to the lower extremity. Cable-driven systems [37] have also been used in rehabilitation.

For more recent approaches for ankle rehabilitation using PRs, readers may refer to [38–42]. The selection and design of the control algorithms are based on analysis of the rehabilitation protocol taking into account the dynamics of both the system and the human–robot interaction. Dynamic posturography has been studied [43], where multi-axial perturbations are required. However, they did not use force sensor measurement.

The main motivation behind this work is to improve the therapeutic resources that can be applied to people with locomotive disorders and to offer better rehabilitation results by providing different types of exercises. In this context, it is important to develop an appropriate low-cost mechanical solution that is able to adapt different rehabilitation exercises to different patients. Unlike the aforementioned rehabilitation devices, the proposed system not only has a suitable kinematic and dynamic design but also provides a control system equipped with a learning algorithm that monitors movements and forces that arise during the execution of the exercise and that can adapt to patients' needs using a new Learning from Demonstration (LfD) framework [44]. LfD is an end user technique for non-roboticists to teach new behaviors to a robot by extracting task-relevant information from a demonstration (or several demonstrations) and transfer these skills directly to a robot instead of hard-coding. LfD has proven to be an effective way of teaching robots important motion skills that are necessary when assisting people and providing health care services [45]. Specifically, LfD approaches have been used to teach robots a variety of skills, e.g., physical rehabilitation [46], hand rehabilitation [47], motion planning for rehabilitation [48], robotic surgery [49], and feeding [50], among others.

In this paper, we propose to exploit LfD to learn passive rehabilitation exercises and adapt them based on the patient's needs by integrating Dynamic Movement Primitives (DMPs) [51] and Iterative Learning Control (ILC) [52]. DMPs are trajectory generators that can effectively encode and reproduce trajectories. DMPs were first introduced by [51], then updated in 2013 by [53], before being further updated to include unit quaternion trajectories [54]. Most recently, they have been updated to encode symmetric positive definite matrices profiles [55]. On the other hand, ILC makes it possible to reuse the control signal from the previous iteration cycle in the next one [56].

This paper is an extension of a previous conference paper [57], where we extended the theory behind the rehabilitation device and algorithm, the control scheme, stability analysis, and the validation experiments. The main contributions of this paper are:

- Exploitation of force sensing in an LfD framework for ankle rehabilitation using a PR that integrates ILC and DMPs to learn different passive exercises and adapt them autonomously;
- Implementation of soft emergency stopping due to the integration of DMP phase-stopping in the emergency button control loop in order to provide soft and smooth stopping;
- Provision of a stability analysis of our learning control;
- Provision of a brief review of ankle rehabilitation devices, injuries, and exercises;
- Implementation of different experiments in order to validate our control scheme.

2. Overview of the Ankle Joint: Anatomy, Physiology, and Injuries

The human foot and ankle are composed of 28 bones: tibia, fibula, 7 tarsals, 5 metatarsals, and 14 phalanges. The human ankle joint is a very complex bony structure [58] that is composed of three joints: the ankle joint proper or talocrural joint, the subtalar joint, and the inferior tibiofibular joint.

The ankle joint has rotations in the sagittal, frontal, and transverse planes. Figure 1 shows the ankle motion in these three orthogonal planes. These motions are: (i) plantar flexion and dorsiflexion movements in the sagittal plane that occur around the y-axis, (ii) adduction and abduction movements in the transverse plane that occur around the z-axis, and (iii) inversion and eversion movements in the frontal plane that occur around the x-axis. The ranges and moment requirements are summarized in Table 1.

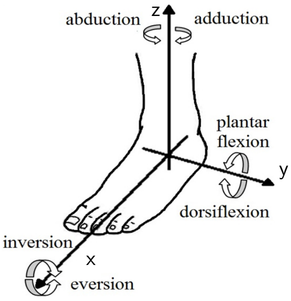

Figure 1. Ankle joint movements in three orthogonal planes [59].

Table 1. ROM and moment requirement for the human ankle.

	Ankle Motion	**Range of Motion ROM [60]**	**Maximum Passive Moment (Nm) [61–63]**
$+\gamma$	Dorsiflexion	20.3° to 29.8°	34.1 ± 14.5
$-\gamma$	Plantarflexion	37.6° to 45.75°	48.1 ± 12.2
$+\beta$	Inversion	14.5° to 22°	33.1 ± 16.5
$-\beta$	Eversion	10,0° to 17°	40.1 ± 9.2
	Adduction	22.0° to 36°	-
	Abduction	15.4° to 25.9°	-

Ankle injuries are among the most common injuries in sport and daily life [58]. Ankle sprains represents 20% to 40% of all sport injuries. These sprains are a stretch or tear of the ligaments due to sudden changes in direction [64,65]. In most cases, ankle sprains can become chronic if the injury is not rehabilitated properly. Approximately 85% of ankle sprains are caused by excessive inversion [66].

The first ankle treatment after injury includes Rest, Ice, Compression, and Elevation (RICE) of the affected foot [13], which should be followed by stretching and therapy exercise along with partial weight bearing with crutches to maintain mobility in the ankle. In order to avoid muscular atrophy, which may lead to a reduction of the ROM, and stimulate healing of the injured ligaments, patient should start motion therapy within 72 h after the injury [64]. Once the ROM is achieved, strengthening of weakened muscles is essential for rapid recovery and is a preventive measure against further injury. Once patients achieve full weight-bearing capability without pain, proprioceptive exercises are initiated. These exercises aim to recover both balance and postural control using wobble boards. Finally, advanced exercises using an uneven surface wobble board should be performed to regain normal activity functions.

At present, different techniques are used for ankle rehabilitation; however, not all of them have the same effectiveness. Some techniques require the patient to be an active agent in the rehabilitation process [67]. In these cases, the patient performs active work through a series of exercises which are gradually intensified to help the ankle regain its mobility. On the other hand, the patient may also perform passive work [67,68], which usually occurs in the early stages of rehabilitation. In this type of work, an external agent, either a qualified person or a device, moves the patient's ankle without her/his voluntary movement.

In this article, a number of references have been generated to rehabilitate an injured ankle with passive exercises. These passive exercises are used to train dorsiflexion/plantar flexion and inversion/eversion ankle movements. The rehabilitation exercises are performed using a three-PRS (Prismatic, Revolute, and Spherical) PR, which is described in the following section.

3. Parallel Robot: Kinematics And Dynamics

A three-df PR is used as a mechanical device for ankle rehabilitation. Its kinematic and dynamic models are explained in the following subsections.

3.1. Three-PRS Kinematics

The PR has been modeled by means of a set of nine dependent coordinates, as can be seen in Figure 2. These coordinates are: (i) actuated prismatic joints ($\underline{\mathrm{P}}$) represented by q_1, q_6, and q_8, (ii) passive revolute joints (R) in q_2, q_7, and q_9, and (iii) the coordinates q_3, q_4, and q_5 only correspond to one spherical joint (S), which is located at position P_1.

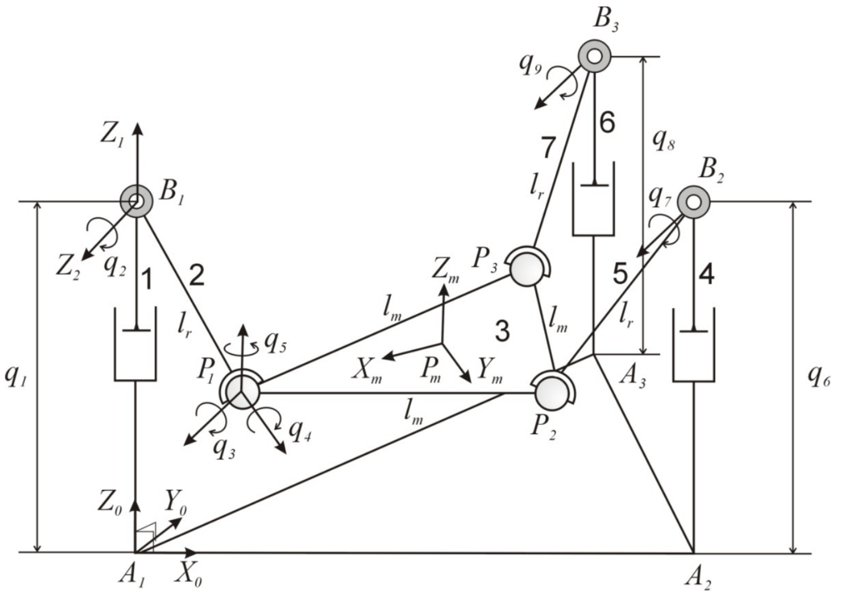

Figure 2. Kinematic diagram of the three-PRS, type of joints, and generalized coordinates.

Explicit expressions can be obtained for the inverse kinematic problem. Given the location of the mobile platform, so that the position and orientation of the reference system $\{\boldsymbol{P}_m - \boldsymbol{X}_m\boldsymbol{Y}_m\boldsymbol{Z}_m\}$ with respect to the fixed one $\{\boldsymbol{A}_1 - \boldsymbol{X}_0\boldsymbol{Y}_0\boldsymbol{Z}_0\}$ attached to the base of the robot, it is possible to obtain the coordinates of points P_1, P_2, and P_3, corresponding to the spherical joints. These points are then used to determine the active coordinates q_1, q_6, and q_8.

The forward kinematics is solved using a geometric approach, taking into account that the length between any consecutive P_d points is constant and equal to l_m, where $d = 1,2,3$. Thus, the following three nonlinear geometrical constraints can be obtained:

$$\begin{aligned}
\Gamma_1(q_{1,2,6,7}) &= \|(\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{B}_1} + \boldsymbol{r}_{\boldsymbol{B}_1\boldsymbol{P}_1}) - (\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{A}_2} + \boldsymbol{r}_{\boldsymbol{A}_2\boldsymbol{B}_2} + \boldsymbol{r}_{\boldsymbol{B}_2\boldsymbol{P}_2})\| - l_m = 0, \\
\Gamma_2(q_{1,2,8,9}) &= \|(\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{B}_1} + \boldsymbol{r}_{\boldsymbol{B}_1\boldsymbol{P}_1}) - (\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{A}_3} + \boldsymbol{r}_{\boldsymbol{A}_3\boldsymbol{B}_3} + \boldsymbol{r}_{\boldsymbol{B}_3\boldsymbol{P}_3})\| - l_m = 0, \\
\Gamma_3(q_{6,7,8,9}) &= \|(\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{A}_3} + \boldsymbol{r}_{\boldsymbol{A}_3\boldsymbol{B}_3} + \boldsymbol{r}_{\boldsymbol{B}_3\boldsymbol{P}_3}) - (\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{A}_2} + \boldsymbol{r}_{\boldsymbol{A}_2\boldsymbol{B}_2} + \boldsymbol{r}_{\boldsymbol{B}_2\boldsymbol{P}_2})\| - l_m = 0,,
\end{aligned} \tag{1}$$

If we know the active generalized coordinates q_1, q_6, and q_8, it is possible to obtain the passive coordinates q_2, q_7, and q_9 by solving the equation (Equation (1)). Afterwards, the locations of points P_1, P_2, and P_3 can be easily obtained. From the coordinates of those three points, the roll γ, the pitch β angles, and the heave z of the mobile platform can be obtained.

The velocity problems, both inverse and forward, are based on the following:

$$\boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{P}_d} + \boldsymbol{r}_{\boldsymbol{P}_d\boldsymbol{P}_n} = \boldsymbol{r}_{\boldsymbol{A}_1\boldsymbol{A}_n} + q_j\boldsymbol{u}_{\boldsymbol{A}_1,\boldsymbol{B}_n} + l_r\boldsymbol{u}_{\boldsymbol{B}_n,\boldsymbol{P}_n}, \tag{2}$$

where $j = 1, 6$, and $n = 1,\ 2,\ 3$. $\boldsymbol{A}_1$ is the fixed frame system. $\boldsymbol{P}_d$ is the origin of the reference system attached to the mobile platform (end effector), $\boldsymbol{u}_{\boldsymbol{A}_n\boldsymbol{B}_n}$ is a unitary vector from joints $\boldsymbol{A}_n$ to $\boldsymbol{B}_n$, $\boldsymbol{u}_{\boldsymbol{B}_n\boldsymbol{P}_n}$ is a unitary vector from points $\boldsymbol{B}_n$ to $\boldsymbol{P}_n$, and l_r is the constant length of links 2, 5, and 7. Figure 3 shows the closed loop for the velocity and acceleration problems for link 1; the same can be applied for links 2 and 3.

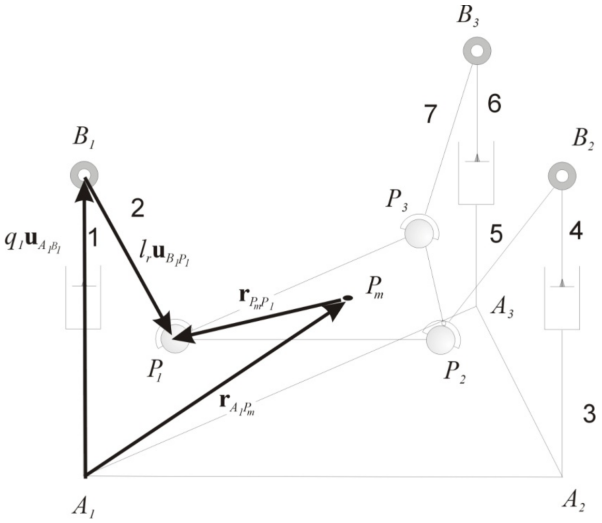

Figure 3. Closed loop for the velocity and acceleration problems.

By multiplying (dot product) both sides of Equation (2) by $\boldsymbol{u}_{\boldsymbol{B}_n\boldsymbol{P}_n}$ and taking time derivatives, the following matrix expression can be obtained. This expression relates the linear and angular velocities of the mobile platform to the time derivative of the active generalized coordinates:

$$J_{\mathsf{x}} \left[\dot{\mathsf{x}} \quad \dot{\mathsf{y}} \quad \dot{\mathsf{z}} \quad \omega_{\mathsf{x}} \quad \omega_{\mathsf{y}} \quad \omega_{\mathsf{z}}\right]^\top = J_q \cdot \left[\dot{q}_1 \quad \dot{q}_6 \quad \dot{q}_8\right]^\top \tag{3}$$

where $\boldsymbol{V}_{P_m} = [\dot{x} \quad \dot{y} \quad \dot{z}]^\top$ is the velocity of the origin of the mobile reference frame (end effector). $\boldsymbol{\omega}_{\boldsymbol{P}_m} = [\omega_x \quad \omega_y \quad \omega_z]^\top$ is the angular velocity of the mobile platform, $\boldsymbol{J}_{\mathsf{x}}$ is the Jacobian matrix in Cartesian space, and $\boldsymbol{J}_q$ is the Jacobian matrix in generalized coordinate space.

Finally, taking into account that the parallel robot has three degrees of freedom, it is possible to obtain a relationship between the velocities of the mobile platform and the time derivatives of the roll, pitch, and heave of the reference system attached to the mobile platform and any choice of three of them forming the matrix $\boldsymbol{J}_m$, such as:

$$\left[\dot{\mathsf{x}} \quad \dot{\mathsf{y}} \quad \dot{\mathsf{z}} \quad \dot{\phi} \quad \dot{\beta} \quad \dot{\gamma}\right]^\top = J_m \cdot \left[\dot{\mathbf{z}} \quad \dot{\boldsymbol{\beta}} \quad \dot{\gamma}\right]^\top \tag{4}$$

so that:

$$J_x J_m \left[\dot{\mathsf{z}} \quad \dot{\beta} \quad \dot{\gamma}\right]^\top = J_q \cdot \left[\dot{q}_1 \quad \dot{q}_6 \quad \dot{q}_8\right]^\top \tag{5}$$

This equation allows us to solve both the inverse and the forward velocity problems. In a similar way, the expressions for the acceleration can be obtained.

3.2. Three-PRS Dynamics

As mentioned before, the parallel robot has been modeled through a set of dependent generalized coordinates, considering which the equation of motion will be as follows:

$$\boldsymbol{M}(\boldsymbol{q}, \boldsymbol{\Theta})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{\Theta})\dot{\boldsymbol{q}} + \boldsymbol{\mathcal{G}}(\boldsymbol{q}, \boldsymbol{\Theta}) = \boldsymbol{\tau} - \boldsymbol{J}^\top \boldsymbol{\lambda} \tag{6}$$

where $\boldsymbol{\Theta}$ is a vector grouping the dynamic parameters (masses, first inertia moments, moments and products of inertia of the links and friction coefficients). $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and $\ddot{\boldsymbol{q}}$ are the generalized coordinates, velocities, and accelerations. $\boldsymbol{M}$ stands for the mass, $\boldsymbol{C}$ for the centrifugal and Coriolis terms, while $\boldsymbol{\mathcal{G}}$ denotes the gravitational vector. $\boldsymbol{\tau}$ is the generalized torque vector. By deriving the constraint equations with respect to all generalized coordinates, we can obtain the Jacobian matrix $\boldsymbol{J}$. $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers. The detailed dynamic model of this PR is described in [69].

For control purposes, the generalized internal forces term is not convenient, so it could be canceled by multiplying both terms of Equation (6) by an orthogonal complement $\boldsymbol{R}$ [70]. Thus, Equation (6) can be rewritten as follows:

$$\boldsymbol{R}^{\top}\left(\boldsymbol{M}(\boldsymbol{q},\boldsymbol{\Theta})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{\Theta})\dot{\boldsymbol{q}} + \boldsymbol{\mathcal{G}}(\boldsymbol{q},\boldsymbol{\Theta})\right) = \boldsymbol{R}^{\top}\boldsymbol{\tau} \tag{7}$$

By considering the relationship between all the generalized coordinates and the active ones, Equation (7) can be written as follows:

$$\boldsymbol{M}^{*}_{df\times df}(\boldsymbol{q},\boldsymbol{\Theta})\ddot{\boldsymbol{q}}^{*}_{df\times 1} + \boldsymbol{C}^{*}_{df\times df}(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{\Theta})\dot{\boldsymbol{q}}^{*}_{df\times 1} + \boldsymbol{\mathcal{G}}^{*}_{df\times 1}(\boldsymbol{q},\boldsymbol{\Theta}) = \boldsymbol{\tau}^{*}_{F\times 1} \tag{8}$$

where df is the number of degrees of freedom of the parallel robot, and the new vectors $\dot{\boldsymbol{q}}^{*}, \ddot{\boldsymbol{q}}^{*}$ correspond to the active generalized velocities and accelerations.

4. Policy Learning and Adaptation Algorithm

In this section, we introduce our trajectory learning and adaptation algorithm for robots employed in rehabilitation activities. The proposed algorithm is general and can accommodate other robotics applications that involve contact with the environment, such as force-based trajectory tracking.

Figure 4 shows an illustrative diagram of the proposed framework. The position controller is fed by $q_c(\chi)$ (Equation (14)), which in turn results from either the emergency reference trajectory or the adapted one from the policy learning block (in gray). The policy learning block is covered in Sections 4.1–4.4, while feedback error and offset learning is covered in Section 4.5. A more detailed diagram is shown below in Figure 6.

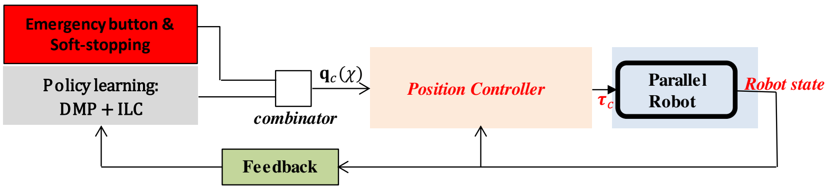

Figure 4. Simple diagram of the control of the proposed force-based trajectory learning and adaptation.

4.1. Learning from Demonstration for Rehabilitation Exercises

In this section, the patient's ankle reference exercise trajectory is designed and the robot learning procedure of the rehabilitation trajectory is described. The trajectory is designed by a medical professional, who guides the mobile platform of the PR while the patient's foot is in the orthopedic boot [38]. This means that the specialist moves the platform in specific directions in dorsiflexion/plantar flexion and eversion/inversion. During this movement, the trajectory of the orthopedic boot (with the patient's foot) is measured by proprioception. The specialist moves the platform, performing an appropriate rehabilitation exercise and setting the maximum positions in each direction. These maximums are determined for each patient according to the pain that the specialist considers the patient can endure in each direction of the movement.

The force sensor is located under the orthopedic boot (Figure 5) and, consequently, the forces exerted by the specialist (human operator) during the demonstration affect the measured forces and torques. Therefore, to obtain the net forces and torques exerted by the patient, the acquired trajectory is replayed by the PR interacting solely with the patient's foot in the boot and without any adaptation.

The resulting force profiles are then recorded. These profiles indicate the actual maximum forces allowed by the patient. This procedure should be repeated for each patient.

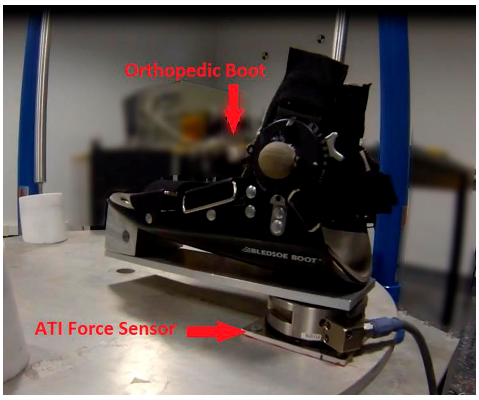

Figure 5. The orthopedic boot and the force sensor attached to the PR.

At this stage, the exercise reference trajectory is determined. Moreover, patients should be able to repeat the exercise without any danger, because they are doing a customized exercise designed for them. In order to make the exercise more comfortable for patients and reduce the maximum forces (pain) applied to them, the specialist marks a threshold a little below those maximum forces. Thus, patients would be able to repeat the exercise assisted by the PR with less pain. After several repetitions, patients may be able to repeat the reference trajectory perfectly without pain. Afterwards, the specialist would determine whether a new exercise reference trajectory should be designed or whether the treatment should end.

4.2. Overview of DMPs

In this paper, robot trajectories are encoded by DMPs. They have the ability to slow execution of the trajectory (exercise) down using a phase-stopping mechanism [51] whenever it is necessary to adapt to the patient's needs. DMPs can be found in many applications, e.g., biped locomotion [71], adaptive frequency modulation [72], reinforcement learning [73], automatic assembly [54,74,75], etc.

For each exercise, as mentioned in Section 4.1, a medical professional sets a personalized exercise reference trajectory for each patient. These trajectories are encoded by DMPs. A DMP for a single arbitrary trajectory y is defined by the following nonlinear differential equations [53]:

$$\tau\dot{z} = \alpha_z(\beta_z(y_d - y) - z) + f(\chi), \tag{9}$$

$$\tau\dot{y} = z, \tag{10}$$

$$\tau\dot{\chi} = -\alpha_\chi\chi, \tag{11}$$

where χ is the phase variable, z is an auxiliary variable, and τ is the time constant. Both parameters α_z and β_z define the behavior of the second-order system described by Equations (9) and (10). The phase evolution is defined by Equation (11). With the choice of the time constant $\tau > 0$, $\alpha_z = 4\beta_z$, and $\alpha_\chi > 0$, the convergence of the underlying dynamic system to a unique attractor point at $y = y_d$ and $z = 0$ is guaranteed [53]. $f(\chi)$ is the linear combination of N nonlinear radial basis functions, which enable the robot to follow any trajectory smoothly from an initial position y_0 to a target position y_d. In the basic DMP Equations (9) and (10), each df is encoded as a separate DMP (one for dorsiflexion/plantar flexion and another for eversion/inversion); however, all the dfs share the same phase variable χ.

4.3. Exercise Generation Using DMPs

In order to encode exercise trajectories we substitute y, y_g, $f(\chi)$ in Equation (9) for $\boldsymbol{q}$, $\boldsymbol{q}_d$, $\boldsymbol{f}_q(\chi)$, and $\dot{y}$ in Equation (10) for $\dot{\boldsymbol{q}}$.

$$\boldsymbol{f}_q(\chi) = \boldsymbol{D}\frac{\sum_{i=1}^{N} w_i \Psi_i(\chi)}{\sum_{i=1}^{N} \Psi_i(\chi)}\chi, \tag{12}$$

where $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{q_d} - \boldsymbol{q}) \in \mathbb{R}^{3\times3}$. The diagonal matrix $\boldsymbol{D}$ is used to scale the movement amplitude if the target configuration changes. $i = 0, 1, \cdots, N$, $\Psi_i(\chi) = \exp(-h_i\,(\chi - c_i)^2)$ are fixed basis functions. c_i are the centers of Gaussian distributed functions throughout the phase of the trajectory and h_i are their widths. w_i are adjustable weights. For each df trajectory, the weights w_i are estimated from nominal trajectories using regression [76]. Thus, the resulting DMPs encode the desired exercise trajectory. To track the desired trajectory,Equations (9) and (10) need to be integrated for all dfs with the common phase in Equation (11).

Since forces $\boldsymbol{F}_d$ and torques $\boldsymbol{M}_d$ (obtained from human demonstration in Section 4.1) are used as desired variables along the trajectory and not as robot control variables, they do not need to be encoded by DMPs. Instead, linear combinations of radial basis functions are used to approximate the desired forces throughout the phase $\chi_i = \chi(t_i)$:

$$\boldsymbol{F}_d(\chi) = \frac{\sum_i \boldsymbol{w}_i^F \Psi_i(\chi)}{\sum_i \Psi_i(\chi)}\chi, \quad \boldsymbol{M}_d(\chi) = \frac{\sum_i \boldsymbol{w}_i^M \Psi_i(\chi)}{\sum_i \Psi_i(\chi)}\chi, \tag{13}$$

Thus, six systems of linear equations need to be solved in order to estimate $\boldsymbol{w}_i^F$ and $\boldsymbol{w}_i^M$ from the measured force/torque data.

4.4. Overview of ILC

ILC [52,56] is a tracking control method for systems that execute the same trajectory in a repetitive mode. ILC assumes that the performance of an agent that repeatedly performs the same task can be improved by learning from past executions. In the conventional ILC formulation, the objective is to reduce the trajectory tacking error while rejecting periodic disturbances. This is obtained by adjusting the pre-defined control input with a corrective term that linearly depends on the tracking error.

Standard ILC assumes: (1) stable system dynamics, (2) fixed common initial conditions for each trial, and (3) the same duration for each trial. In the case of this paper, the third assumption cannot be fulfilled due to the slowing-down/speeding-up of the trajectory. However, in order to overcome this problem, the trajectory is temporarily scaled as a function of the phase variable using Equation (11). In this case, it is possible to sample the same number of times in each trial. In other word, all trials have the same phase even though they have different durations.

4.5. Error Feedback and DMP Phase Stopping

In human–robot interaction, the interaction might change the resulting measured forces/torques when the robot executes the demonstrated trajectory. These forces may be different from the ones (desired forces) recorded by the human demonstration. Consequently, the robot has to adapt the trajectory in order to minimize this difference between the measured and desired forces. As a solution for this problem, either admittance or impedance control can be implemented. In this work, admittance control [77] has been implemented. In human life, it can be observed that individuals are able to acquire skills in many different ways (through work, play, etc.) by repeating the same action over and over again. This means that humans learn skills from repeating actions. In the same way, and for the same task, the robot should learn from previous repetitions to adapt the executed trajectory, especially when the robot is interacting with humans for safety reasons. Hence the importance of tracking and monitoring the error in the previous repetition, which can be used to improve the performance of the next repetition for the same action (trajectory). This principle is used to learn and adapt rehabilitation exercises (trajectories) which is the basic idea of ILC [52,56].

In passive ankle rehabilitation exercises, the robot is required to follow a specific predetermined trajectory, as described in Section 4.1. However, depending on the state of the patient's ankle, this exercise may cause some pain if the robot executes its preset trajectory. To avoid this, trajectory adaptation is introduced whenever the measured force exceeds a certain safety threshold due to the resistance of the patient's ankle to follow the exercise. In order to adapt the PR to the new situation, the trajectory is modified according to the admittance control law [77]:

$$\boldsymbol{q}_c(\chi) = \boldsymbol{\varphi}_q(\chi) + \boldsymbol{K} \cdot \boldsymbol{e}_q(\chi) + \boldsymbol{q}_{DMP}(\chi), \tag{14}$$

where $\boldsymbol{q}_c(\chi)$ is the new position commanded of the robot controller. $\boldsymbol{q}_{DMP}(\chi)$ is the reference trajectory obtained by DMPs and $\boldsymbol{K}$ is the gain matrix. The force feedback control is provided by the feedback error $\boldsymbol{K} \cdot \boldsymbol{e}_q(\chi)$, where $\boldsymbol{e}_q(\chi) = \boldsymbol{F}_d(\chi) - \boldsymbol{F}$, $\boldsymbol{F}$ and $\boldsymbol{F}_d(\chi)$ are the actual measured force profile and the desired force one as a function of phase χ, respectively. $\boldsymbol{\varphi}_q(\chi)$ is the on-line learned offset to be added to the original trajectory, where its initial value is $[0,0,0]^\top$. This way, the commanded trajectory is modified by adding the offset to the original one, instead of modifying it.

To ensure safety for the patients, our proposed framework adapts the execution trajectory in order to minimize the force error between the desired and measured forces. Thus, low gains are used in order to achieve stable and robust force adaptation. Moreover, for efficient force adaptation the algorithm slow the trajectory execution down. DMP slow-down mechanism is derived from Equation (11) [51]:

$$\tau\dot{\chi} = -\frac{\alpha_\chi \chi}{1 + \alpha_{p\chi}\epsilon}, \tag{15}$$

$$\tau\dot{q} = z + \alpha_{py}(\tilde{\boldsymbol{q}} - \boldsymbol{q}), \tag{16}$$

where $\epsilon = \|\tilde{\boldsymbol{q}} - \boldsymbol{q}\|$, $\boldsymbol{q}$ and $\tilde{\boldsymbol{q}}$ are the DMP output and the the corresponding actual position of the robot, respectively. In this work, $\epsilon = \|\boldsymbol{e}_q^{\mathrm{T}}\|$, and $\alpha_{p\chi}, \alpha_{py}$ are positive constants.

4.6. Offset Learning

The aim of our learning framework is to iteratively modify the reference trajectory so that the patient can safely repeat the rehabilitation exercise. The offset is updated after each trial through

$$\delta^q_{t,l+1} = \boldsymbol{\varphi}_{q,l}(\chi_t) + \boldsymbol{K} \cdot \boldsymbol{e}_q(\chi_t), \tag{17}$$

where l is the iterator. Each offset component φ_k is represented as a linear combination of M radial basis functions as follows

$$\varphi_k(\chi) = \frac{\sum_{i=1}^M w_{i,k}\Psi_i(\chi)}{\sum_{i=1}^M \Psi_i(\chi)}\chi. \tag{18}$$

The new data points $\{\delta^k_{t,l+1}\}$, $t = 0, \ldots, T$, are obtained from the k-th component of the offsets trajectory, where $k = 1,2,3$. This optimization problem aims to find $\{w_{i,k}\}$ that minimize the quadratic objective function:

$$\sum_{j=0}^{T} (\varphi_k(\chi_t) - \delta^k_{j,l+1})^2. \tag{19}$$

4.7. Ankle Rehabilitation Control Scheme

A detailed control scheme of the proposed ankle rehabilitation framework is shown in Figure 6. The reference trajectory $\boldsymbol{Q}_d(t)$ is designed by the specialist. The transformation of data from time domain to phase domain and vice versa are done in the blocks $t \Rightarrow \chi$ and $\chi \Rightarrow t$, respectively. $\boldsymbol{q}_d(\chi)$ is the DMP reproduction of the encoded original trajectory $\boldsymbol{Q}_d(t)$. $\boldsymbol{\mathcal{F}}_d(t)$ describes the reference/desired force profile that produces $\boldsymbol{F}_d(\chi)$ by applying Equation (13). The measured forces $\boldsymbol{F}$ may have high values depending on the patient's response during the exercise. The objective of this work is to adapt the exercise by reducing the difference between $\boldsymbol{F}$ and $\boldsymbol{F}_d(\chi)$. By applying Equation (14), the new offset is estimated and added to the one from the previous repetitions $\boldsymbol{\varphi}(\chi)$.

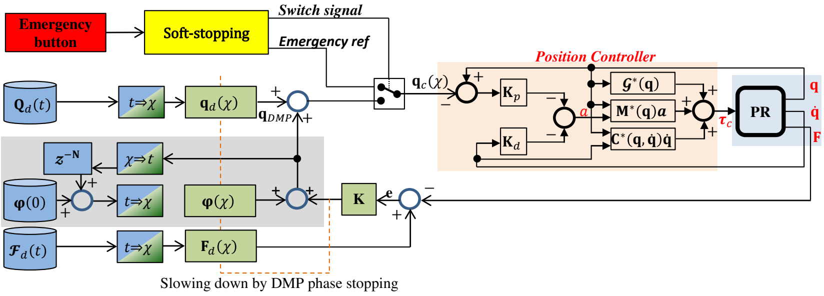

Figure 6. Control scheme.

$\boldsymbol{q}_c(\chi)$ is the commanded trajectory to be executed by the robot and represented as the aggregation of force feedback in Equation (14), the learned offset in several repetitions through Equation (17), and the DMP-generated trajectory. This procedure is repeated until the desired and measured forces match or no further improvement is possible. The gray shaded area in Figure 6 represents the learning procedure, which belongs to the ILC algorithms, where current iteration causal learning is applied, as described in [52,56].

In our framework, we do not modify the reference/original trajectory, instead, we adapt it by adding iteratively an offset learned from the previous iteration. This offset update is represented by the discrete delay $\mathcal{Z}^{-\mathrm{N}}$ block in Figure 6, where N is the number of samples of that repetition.

Under normal conditions, the DMP provides the reference (heave, pitch, and roll angles) for the parallel robot control unit (see Figure 6). The controller compares the reference with the robot position/velocity to calculate the torques as in Equation (8), providing them at a frequency of 100 Hz.

It should be noted that the robot is equipped with an emergency button that can be actuated by the patient or by the specialist. If the button is pressed, the DMP stops the rehabilitation exercise and brings smoothly the platform into a safe configuration. In our case, it has been considered that the platform is at zero value for the heave as well as for the angles of eversion (roll) and dorsiflexion (pitch). When the emergency button is released, the DMP smoothly brings it to the pose where the platform was in before the button was pressed and continues with the rehabilitation exercise.

4.8. Stability Analysis

In order to prove the stability of the learning control, we assumed that the closed control loop of the 3-PRS is stable, without the iteration loop, with proper choice of the admittance feedback gain $\boldsymbol{K}$ [78,79]. However, closed-loop stability does not necessarily imply that the system will remain stable during the repetitive learning. In this regard, the aim of this section is to determine how our system may be affected by the iteration loop.

To clarify the notation, let uppercase letters denote one-sided Z-transform of the corresponding time-discrete signal, which is denoted with lower case letters. Note that the signals in Equations (14), (17) and (18) are phase dependent. However, we can always express the time-dependent counterpart and express the corresponding Z-transform. For the sake of simplicity, explicit dependence on z in transfer functions and Z-transform of the signals is omitted. After that, it is assumed that the nonlinear dynamics of the robot is fully compensated for using feedback control. By assuming a known environment stiffness $\boldsymbol{K}_s$, the force at iteration l ($\boldsymbol{F}_l$) can be predicted:

$$\boldsymbol{F}_l = \boldsymbol{K}_s(\boldsymbol{G}\boldsymbol{P}_l - \boldsymbol{P}_o), \tag{20}$$

where $\boldsymbol{K_s}$ is a diagonal positive definite environment stiffness matrix, $\boldsymbol{G}$ is a diagonal matrix containing the decoupled dynamics of the robot in the form of a second-order system which maps the desired position vector $\boldsymbol{P}_l$ into the actual position, and $\boldsymbol{P}_o$ denotes the environment contact positions. According to Equations (14) and (17), the Z-transform of the error function $\boldsymbol{E}_l$, the position update function $\boldsymbol{P}_l$, and the learned offset function $\boldsymbol{\Phi}$ are:

$$\begin{aligned} \boldsymbol{E}_l &= \boldsymbol{F}_d - \boldsymbol{F}_l, && (21)\\ \boldsymbol{P}_l &= \boldsymbol{P}_d + \boldsymbol{\Phi}_l + \boldsymbol{K}\boldsymbol{E}_l, && (22)\\ \boldsymbol{\Phi}_l &= \mathbb{Q}\big(\boldsymbol{\Phi}_{l-1} + \boldsymbol{K}\boldsymbol{E}_{l-1}\big). && (23) \end{aligned}$$

where $\mathbb{Q}$ presents a transfer function which maps the original sampled function to a function approximated with Gaussian kernel functions. $\boldsymbol{K}$ is the gain matrix from Equation (14). In [80] it was shown that $\mathbb{Q}$ can be approximated with a second-order transfer function. However, with enough M (in Equation (18)), the approximated function, with Gaussian kernel functions, is close enough to the original function, so that $\mathbb{Q}$ can be set to $\boldsymbol{I}$. By defining $\boldsymbol{E}_l$ as the error function and $\boldsymbol{E}_{l-1}$ as the error function in the previous learning cycle [81]:

$$\begin{aligned} \boldsymbol{E}_l &= \boldsymbol{F}_d - \boldsymbol{F}_l \\ &= \boldsymbol{F}_d - \boldsymbol{K_s}\big(\boldsymbol{G}\boldsymbol{P}_l - \boldsymbol{P}_o\big) \\ &= \boldsymbol{F}_d - \boldsymbol{K_s}\big(\boldsymbol{G}\big(\boldsymbol{P}_d + \boldsymbol{\Phi}_l + \boldsymbol{K}\boldsymbol{E}_l\big) - \boldsymbol{P}_o\big) \\ &= \boldsymbol{F}_d - \boldsymbol{K_s}\big(\boldsymbol{G}\big(\boldsymbol{P}_d + \boldsymbol{\Phi}_{l-1} + \boldsymbol{K}\boldsymbol{E}_{l-1}\big) - \boldsymbol{P}_o\big) \qquad (24) \\ &\quad -\boldsymbol{K_s}\boldsymbol{G}\boldsymbol{K}\boldsymbol{E}_l \\ &= \boldsymbol{F}_d - \boldsymbol{F}_{l-1} - \boldsymbol{K_s}\boldsymbol{G}\boldsymbol{K}\boldsymbol{E}_l \\ &= \boldsymbol{E}_{l-1} - \boldsymbol{K_s}\boldsymbol{G}\boldsymbol{K}\boldsymbol{E}_l \end{aligned}$$

Dividing Equation (24) by $\boldsymbol{E}_l$ and re-arranging leads to:

$$\frac{\boldsymbol{E}_l}{\boldsymbol{E}_{l-1}} = \frac{\boldsymbol{I}}{\boldsymbol{I} + \boldsymbol{K_s}\boldsymbol{G}\boldsymbol{K}} \qquad (25)$$

Asymptotic stability is assured if $\frac{\boldsymbol{E}_l}{\boldsymbol{E}_{l-1}} < 1, \forall\ l$. Inserting the z dependence into transfer functions and signals and substituting $z = e^{j\omega}$ in Equation (25), the condition for asymptotic stability becomes [52]:

$$\frac{\boldsymbol{I}}{\boldsymbol{I} + \boldsymbol{K_s}\boldsymbol{G}\big(e^{j\omega}\big)\boldsymbol{K}\big(e^{j\omega}\big)} < 1; \quad \forall\ \omega \qquad (26)$$

With a proper selection of $\boldsymbol{K_s}$ and $\boldsymbol{K}$, the above equation is fulfilled and the learning stability is guaranteed.

5. Results

In this section, we develop different simulation examples as will as real experiments in order to validate the performance of our LfD framework. The trajectories used in these examples were obtained from the real robot by a medical professional guiding the moving platform of the robot, while the patient's foot is installed in the boot, according to the procedure detailed in Section 4.1. These trajectories represent only passive exercises, which require the robot to follow them accurately.

To simulate the robot, we used a MATLAB Simulink® model. The model accurately imitates the real robot [69]. The description of the robot's hardware is detailed in the next section, where the kinematic and dynamic models used in the simulation are based on that setup.

5.1. Hardware Description

The rehabilitation robot (see Figure 7) consists of three PRS kinematic chains (1, 4, 6 in Figure 2) connected at one end to a coupling bar (2, 5, 7 in Figure 3), and at the other end they are perpendicularly attached to the platform's base, as shown in Figure 2. Each leg is driven by a direct drive ball screw actuator (actuated prismatic joints) ($\underline{\mathrm{P}}$). The coupling bar is connected on one side to the leg with a revolute (R) joint and with a spherical joint (S) to the moving platform on the other side. The legs are distributed forming an equilateral triangular configuration at the base. The choice of this configuration is based on the need to develop a low-cost robot with two rotational dfs that are required to perform the main rehabilitation exercises; dorsiflexion/plantar flexion and eversion/inversion. In addition, a translational df is used to adapt the platform with respect to the patient's height while sitting on a chair. The configuration and dimensions of the robot fulfill the requirements to perform the lower limb rehabilitation exercises proposed in this paper.

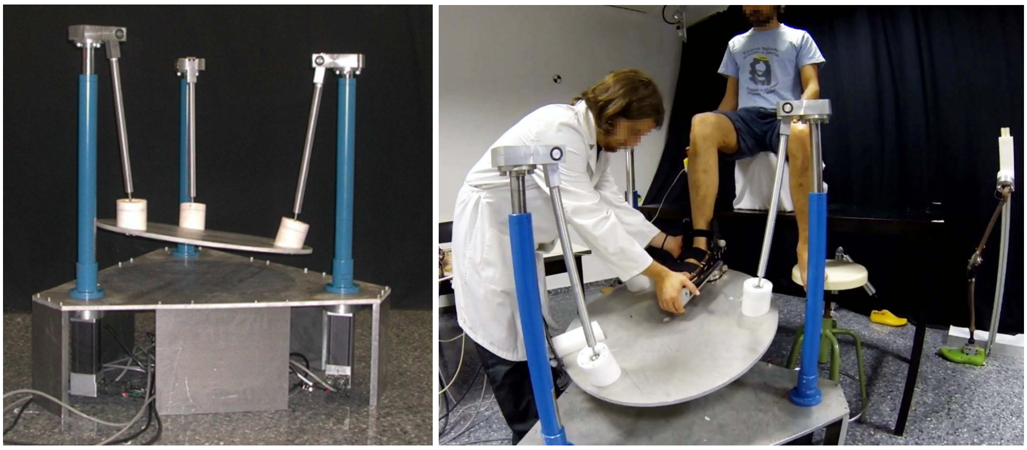

Figure 7. *Left*: a snapshot showing our 3-PRS robot. *Right*: a physiotherapist is fitting the patient's foot in the orthopedic boot.

The motor for each actuated prismatic joint is a brushless DC servomotor equipped with a power amplifier, with the following specifications: continuous stall torque of 2.86 Nm and continuous peak torque of 11.43 Nm. The lead of the ball screw is 20 mm and the actuators are Aerotech BMS465 AH brushless servomotors.

The control unit of the robot is based on an industrial PC equipped with two AdvantechTM cards: a PCI-1720 to supply the control actions by means of digital-to-analog outputs, and a PCI-1784, to read the actuated prismatic joint positions. The PC is equipped with a Linux Ubuntu operating system (patched with the real-time kernel Xenomai) and the open software middlewares Open RObot COntrol Software (Orocos) and Robot Operating System (ROS).

We have used open-source middlewares installed on an industrial PC for the robot control architecture. The main advantages of that are: (i) open-source with high-level tasks programming capabilities (e.g., control based on external sensing using a force sensor, automatic trajectory generation, and artificial vision, etc) and (ii) it is low-cost where the total cost of the hardware is around $2000, in addition to free operating system and programming tools.

The parallel robot is equipped with an orthopedic boot and Delta SI-330-30 ATI force sensor (Figure 5). This sensor is six-df and is capable of measuring forces and torques in 3D using a amonolithic instrumented transducer. The force/torque sensor is integrated into the system to measure the effort exerted by the patient. This configuration provides the possibility of implementing different types of rehabilitation exercises (active and passive), although this paper focuses on passive exercises. A complete description of the mechatronics of this robot can be found in [79].

5.2. Experiments in Simulation

5.2.1. Execution of Different Exercises

In the first simulation test, we have applied our proposed framework to learn and adapt an exercise trajectory that moves the robot platform in z while maintaining zero values for the pitch and roll angles, Figure 8. The robot moves along the nominal trajectory as long as the vertical force exerted by the patient keeps within the admissible values. Whenever the sensed forces exceeds a threshold designed by the therapist, the control system will trigger the phase-stopping mechanism in order to slow the trajectory evolution down, Equation (15). During the slow-down, a new z-offset will be estimated to be added to the trajectory in the next cycle which subsequently will reduce the sensed forces. Consequently, across cycles, the phase stopping becomes less frequent and the execution time decreases in each learning iteration.

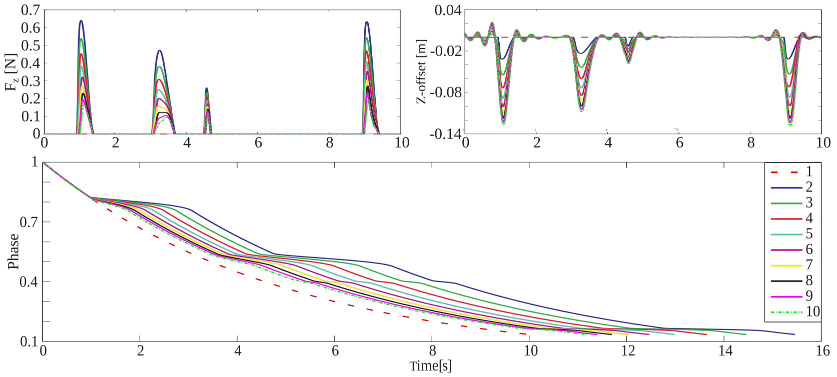

Figure 8. Force adaptation of the proposed framework after nine trials. *Top-right*: shows how the forces adapt, decreasing from *Line*-2 to *Line*-10. *Top-left*: represents the learned offsets, which increase throughout the trials. *Bottom*: shows the phase evolution after nine trials.

If we observe Figure 8, the measured forces decrease from one trial to the next (from 2 to 10). This is because the PR adapts its trajectory whenever the measured forces exceed the desired forces. Moreover, the learned offset increases with each iteration (from 2 to 10) to adapt the trajectory to the new situation. *Line*-1 in Figure 8 corresponds to the reference force, offset, and phase. At this stage of this work, the desired force profile is equal to zero to test the functionality of the proposed algorithm on our robot.

Figure 9 shows a simulation for an exercise that has been repeated 14 times. The purpose of this simulation is to show the adaptability of the system to new situations. For instance, when the measured forces are high, trajectory amplitude is reduced. The reference trajectory peaks are at 0.5 m and the corresponding forces are at 17.5 N. Now, if we set the force threshold to 13 N; which corresponds to position 0.35 m, it can be observed from the figure that the algorithm modifies the reference trajectory amplitude by adding a position offset in each repetition. This offset is calculated from the feedback forces error. After 14 repetitions, the algorithm is able to replay the trajectory into the safe region.

The next experiment, shown in Figure 10, demonstrates a three-df exercise. In this experiment the algorithm runs γ, β, and z trajectories.

Figure 10-*bottom* shows the phase evolution of the whole exercise, the learned offset for each df is shown in Figure 10-*left*, while the force adaptation for each df is illustrated in Figure 10-*middle*. Figure 10-*right* shows the original exercise trajectory for each df as a dashed red line, the first, fifth, and fifteenth repetitions (cycles) of the same exercise. It is clear from the figure that the algorithm does not change the shape of the original trajectory. In fact, it slows the exercise down and tries to reduce the resulting forces.

It is noteworthy that when the system stabilizes and there is no further adaptation, after a period of time determined by the specialist, the algorithm starts to go backward in the direction of the original trajectory by gradually removing the added offsets. At each offset removal, the system repeats the exercise for another period of time, and so on.

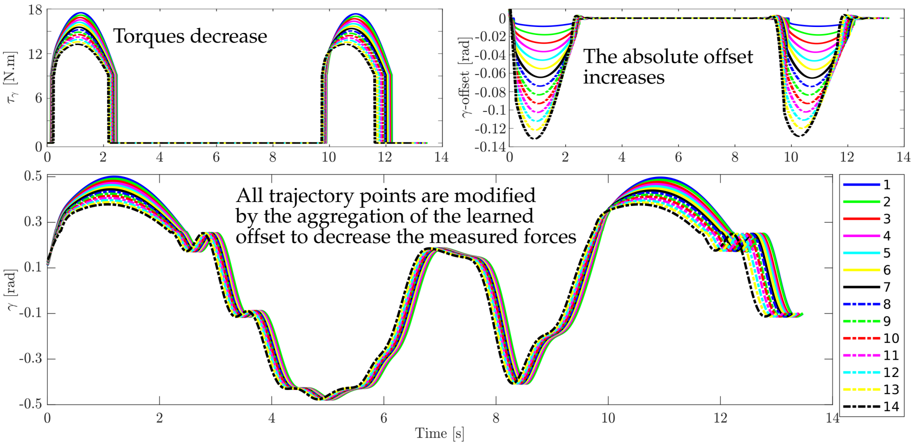

Figure 9. Adaptation of torques of the proposed framework after 14 trials. *Top-right*: shows torques decreasing from *Line*-1 to *Line*-14. *Top-left*: represents the learned offsets and increases throughout the trials. *Bottom*: shows adaptation of the exercise trajectory after 14 trials.

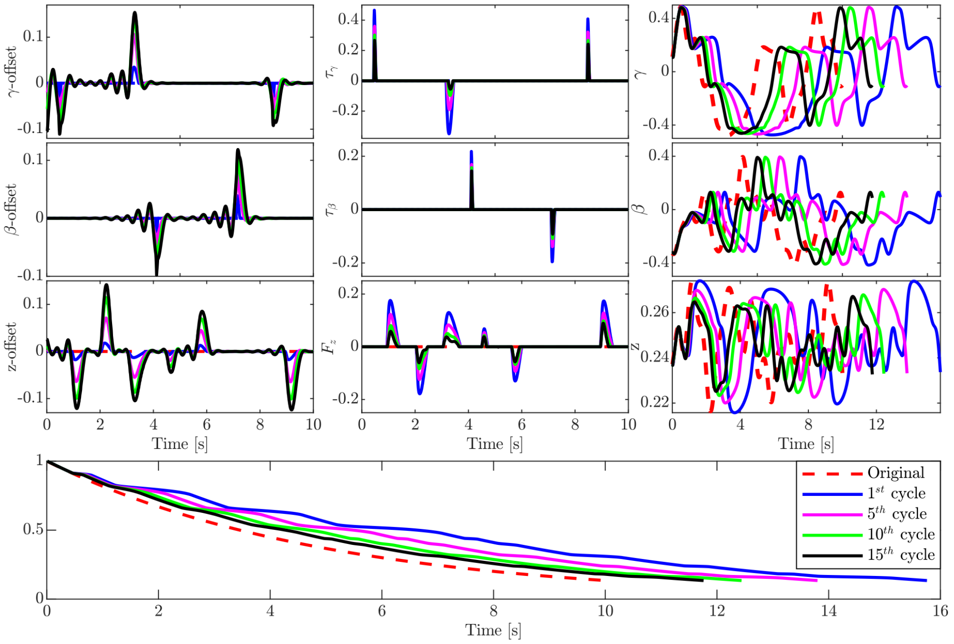

Figure 10. Adaptation for a three-df exercise. *Left-column*: increase of learned offset over 15 cycles. *Middle-column*: force adaptation. *Right-column*: exercise trajectory adaptation. *Bottom*: phase evolution.

5.2.2. Position Tracking Error

Different model-based control strategies have been implemented for the PR, such as passivity-based control, inverse dynamic control, and adaptive control. More detailed information about the design and implementation of these controllers can be found in [79]. In addition, the authors in [78] demonstrated that the closed-loop system (robot/adaptive controller) is convergent, so the tracking error asymptotically converges to zero and all internal signals remain bounded under suitable conditions of the controller gains.

Figure 11 illustrates the robot response for the third active generalized coordinate q_8. Figure 11-*left* shows the reference and the robot response, while Figure 11-*right* represents the joint error. As can readily be appreciated in these figures, the robot response obtained is accurate because the robot joint follows the reference with a very small error. The other generalized coordinates, q_1 and q_6, have a very similar behavior. For verification, Table 2 shows the mean error, the root–mean–square error (RMSE) and the variance between the references and the parallel robot active joint positions.

Table 2. Robot position errors (mean and RMSE) and variance.

Joint	Mean Errors	RMSE	Variance
q_1	0.00343	0.00405	1.643×10^{-3}
q_6	0.00320	0.00396	1.567×10^{-3}
q_8	0.00265	0.00341	1.156×10^{-3}

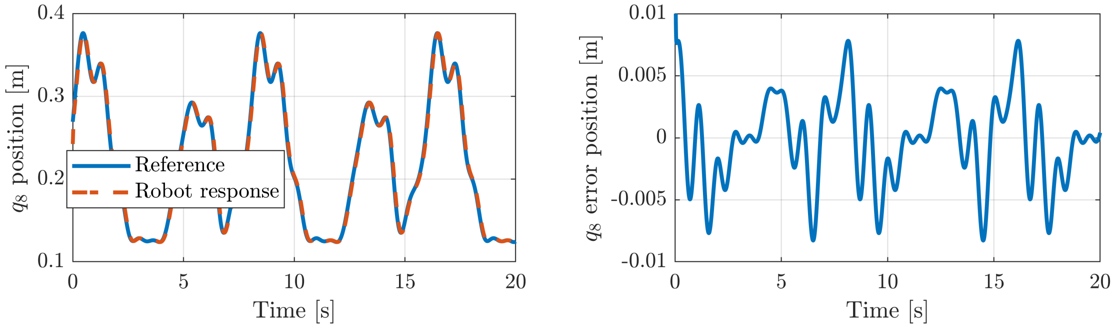

Figure 11. The response of the generalized coordinate q_8 in Simulink.

5.2.3. Emergency Button Testing

In order to validate soft-emergency-stopping, in this simulation we have used an isokinetic rehabilitation exercise based on gait trajectory training [33]. Such exercises are used to restore the original mobility and ROM and to strengthen the affected limbs or ankle joint. The ankle and foot motions are generated based on a gait of normal walking on level ground.

This exercise is used to restore the range of flexion/dorsiflexion motion. In order to establish the exercise, we need to elevate the platform to a certain heave. Figure 12-*left* shows the reference trajectory of the heave and the corresponding robot response, while Figure 12-*right* represents the error between both signals. In the same way, Figure 13-*left* illustrates the plantar flexion/dorsiflexion trajectory of the gait exercise along with the robot response. The error between both signals is shown in Figure 13-*left*. In all the cases, the robot controller provides very good performance, with a small error value.

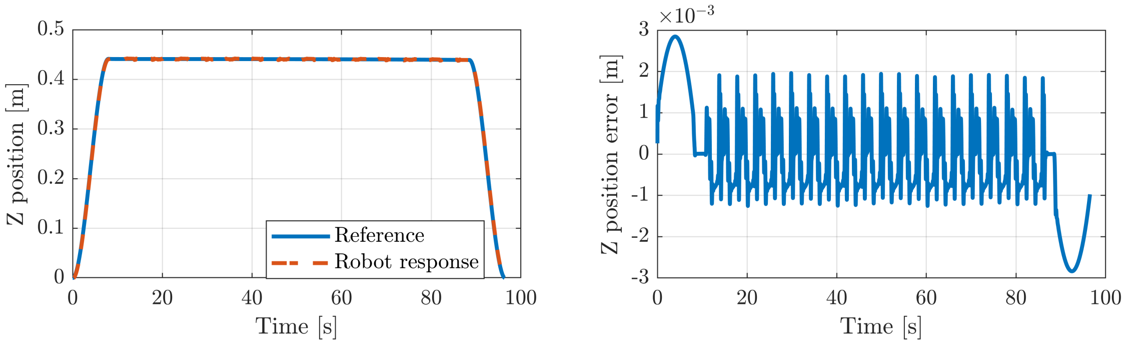

Figure 12. Mobile platform heave of the rehabilitation robot.

The following figures show another application of the DMPs with the rehabilitation robot. In this case, the patient and the medical doctor have an alarm button (red block in Figure 6). By pressing it, the control unit stops the normal execution of the exercise and moves the platform to a rest position using linear DMPs in Equation (9) with $f_q(\chi) = 0$ (yellow block in Figure 6). As soon as the alarm button is released, the robot returns to the initial position before the alarm activation and restarts the rehabilitation exercise. The resting configuration that has been considered for the moving platform is a heave position of 0.3 m with an orientation for the roll and pitch of 0 rad. In this trial, the alarm button was pressed at time t = [19, 24] s. and at time t = [44, 53] s. Figures 14 and 15 show the heave and the pitch evolution for the mobile platform in this experiment.

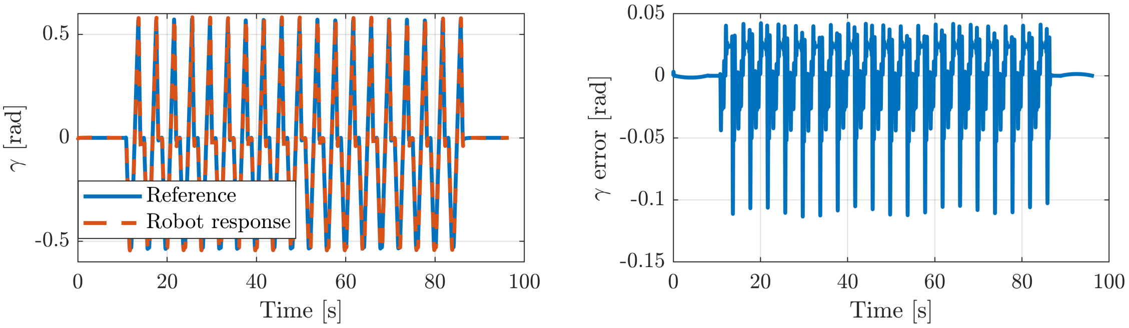

Figure 13. Mobile platform pitch of the Simulink model of the rehabilitation robot.

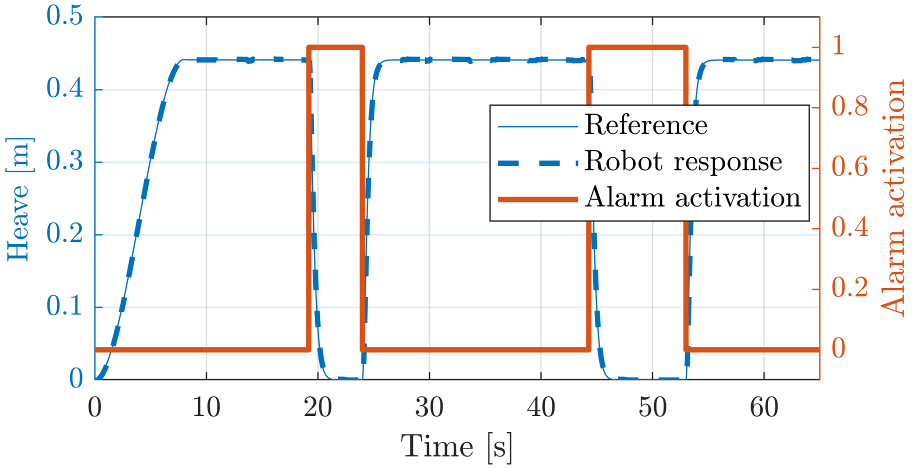

Figure 14. Demonstrates the effect of soft-stopping (yellow block in Figure 6) on heave evolution.

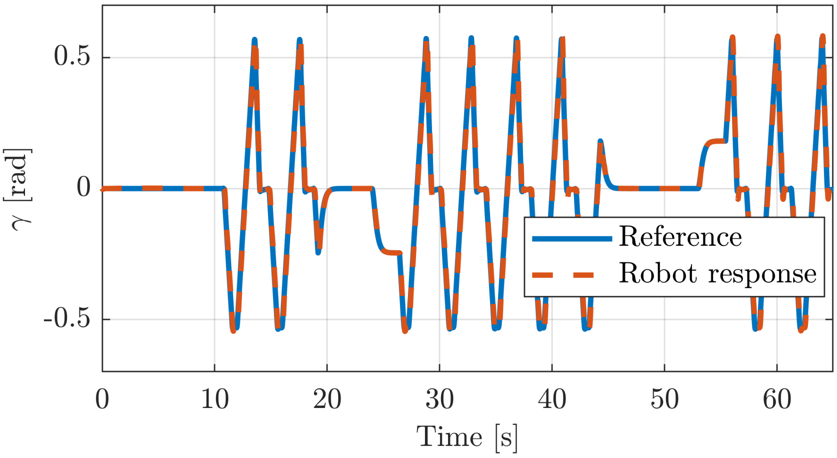

Figure 15. Pitch evolution with alarm activation in the Simulink model of the PR.

5.3. Experiments in Real Robot

Similarly, the previous simulated experiments (Section 5.2.3) have been tested here in real setup using the PR shown in Figure 7. Figure 16 illustrates how accurately the robot tracks a reference trajectory. Figure 16a shows the reference and the response of the active joint q_8, while (b) shows its response when the emergency button is pressed and released in two different locations during the execution.

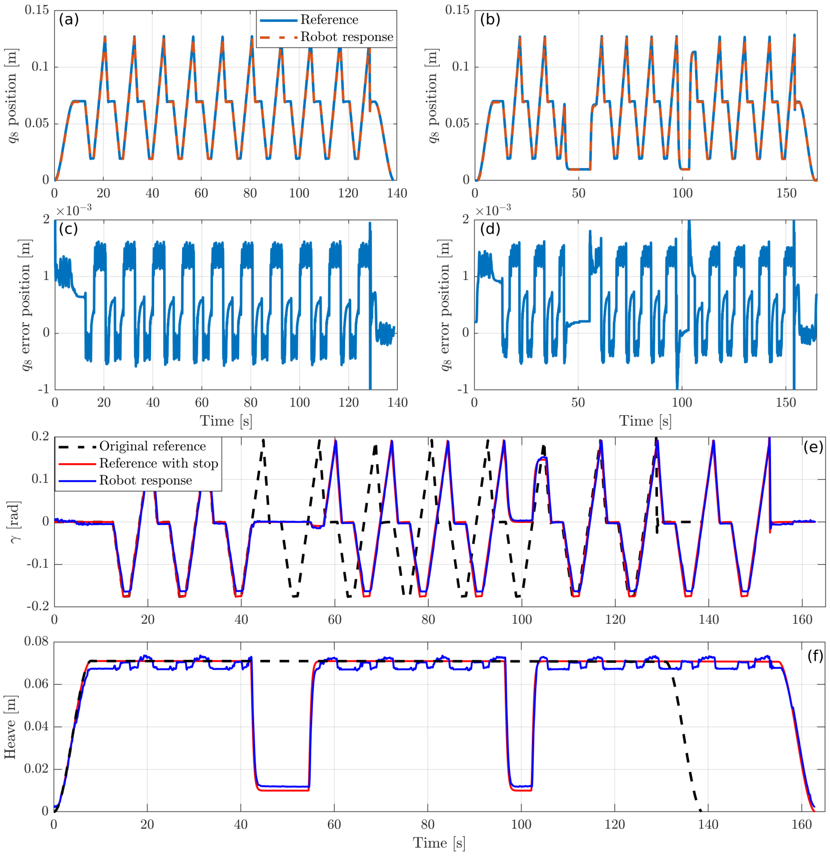

Figure 16. The response of the generalized coordinate q_8 of the real robot is shown in (**a**) while the effect of soft-stopping is shown in (**b**). (**c**,**d**) show the tracking error in both cases. (**e**,**f**) demonstrate the effect of soft-stopping on pitch and heave evolution.

As can be observed in Figure 16c,d, the response of the real robot is very precise, having a tracking error around 1 mm in both previous executions.

Figure 16e,f show the reference trajectory and the response of the robot for γ and the heave, respectively. In this execution, the emergency button has been activated twice, as it can be observed in the figure. This execution has been obtained with a system of 10 cameras that detect the position and orientation of the mobile platform. As we can see, the tracking for the angle γ is accurate, while the heave has a tracking error of about 3 mm due to the mechanical clearances of the robot.

6. Conclusions

In this paper, we proposed an LfD framework to learn and adapt passive exercises for ankle rehabilitation using a PR. This framework, exploits DMPs along with ILC in order to iteratively adapt the exercise trajectory by transferring the feedback error into an offset that can be added to the original trajectory.

Moreover, we solved the forward and inverse kinematic models for our device as well as the dynamic model and the Jacobian needed to implement force control. A model-based controller was chosen to carry out the active generalized coordinates position control. The response obtained with this position control showed an accurate response in terms of position error.

In order to validate our system, we conducted several simulation examples in addition to real experiments in order to test the adaptability, robustness, and accuracy of the system. In these tests, we used passive exercises trajectories where different movement references for γ, β, and z have been executed by the robot. Observing the experiments, the algorithm was able to successfully adapt the exercise to the patient's needs by learning the offset that leads to a reduction in the measured forces exerted by the patient.

Finally, our proposed framework successfully adapts trajectories based on sensed forces. However, in the future we still need to extend this approach in two directions: (i) perform a clinical study, and (ii) speed-up the exercise execution based on the therapeutic recommendations; for example, when the patient repeats the original exercise, the algorithm starts to speed up the exercise.

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

Funding: This work has been partially funded by the FEDER-CICYT project with reference DPI2017-84201-R (Integración de modelos biomecánicos en el desarrollo y operación de robots rehabilitadores reconfigurables) financed by Ministerio de Economía, Industria e Innovación (Spain).

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

Abbreviations

The following abbreviations are used in this manuscript:

DMP	Dynamic Movement Primitive
ILC	Iterative Learning Control
PR	Parallel Robot
RICE	Rest, Ice, Compression, and Elevation
ROM	Range of Motion

References

1. Brewer, B.R.; McDowell, S.K.; Worthen-Chaudhari, L.C. Poststroke upper extremity rehabilitation: A review of robotic systems and clinical results. *Top. Stroke Rehabil.* **2007**, *14*, 22–44. [CrossRef]
2. Michmizos, K.P.; Rossi, S.; Castelli, E.; Cappa, P.; Krebs, H.I. Robot-aided neurorehabilitation: A pediatric robot for ankle rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2015**, *23*, 1056–1067. [CrossRef] [PubMed]
3. Marchal-Crespo, L.; Reinkensmeyer, D.J. Review of control strategies for robotic movement training after neurologic injury. *J. Neuroeng. Rehabil.* **2009**, *6*, 20. [CrossRef] [PubMed]
4. Balasubramanian, S.; Colombo, R.; Sterpi, I.; Sanguineti, V.; Burdet, E. Robotic assessment of upper limb motor function after stroke. *Am. J. Phys. Med. Rehabil.* **2012**, *91*, S255–S269. [CrossRef] [PubMed]
5. Rea, P.; Ottaviano, E.; Castelli, G. A procedure for the design of novel assisting devices for the sit-to-stand. *J. Bionic Eng.* **2013**, *10*, 488–496. [CrossRef]
6. Sale, P.; Franceschini, M.; Waldner, A.; Hesse, S. Use of the robot assisted gait therapy in rehabilitation of patients with stroke and spinal cord injury. *Eur. J. Phys. Rehabil. Med.* **2012**, *48*, 111–121.
7. Rohrer, B.; Fasoli, S.; Krebs, H.I.; Hughes, R.; Volpe, B.; Frontera, W.R.; Stein, J.; Hogan, N. Movement smoothness changes during stroke recovery. *J. Neurosci.* **2002**, *22*, 8297–8304. [CrossRef] [PubMed]
8. Martí Carrillo, F.; Butchart, J.; Knight, S.; Scheinberg, A.; Wise, L.; Sterling, L.; McCarthy, C. Adapting a General-Purpose Social Robot for Paediatric Rehabilitation through In Situ Design. *ACM Trans. Hum. Robot. Interact.* **2018**, *7*, 1–30. [CrossRef]
9. Chisholm, K.; Klumper, K.; Mullins, A.; Ahmadi, M. A task oriented haptic gait rehabilitation robot. *Mechatronics* **2014**, *24*, 1083–1091. [CrossRef]

10. Munih, M.; Bajd, T. Rehabilitation robotics. *Technol. Health Care* **2011**, *19*, 483–495. [CrossRef]
11. Krebs, H.I.; Palazzolo, J.J.; Dipietro, L.; Ferraro, M.; Krol, J.; Rannekleiv, K.; Volpe, B.T.; Hogan, N. Rehabilitation robotics: Performance-based progressive robot-assisted therapy. *Auton. Robot.* **2003**, *15*, 7–20. [CrossRef]
12. Hesse, S.; Schmidt, H.; Werner, C.; Bardeleben, A. Upper and lower extremity robotic devices for rehabilitation and for studying motor control. *Curr. Opin. Neurol.* **2003**, *16*, 705–710. [CrossRef]
13. Dai, J.S.; Zhao, T.; Nester, C. Sprained ankle physiotherapy based mechanism synthesis and stiffness analysis of a robotic rehabilitation device. *Auton. Robot.* **2004**, *16*, 207–218. [CrossRef]
14. Zhen, H.; Yongsheng, Z.; Tieshi, Z. *Advanced Spatial Mechanism*; Higher Education Press: Beijing, China, 2006; pp. 195–201.
15. Díaz, I.; Gil, J.; Sánchez, E. Lower-limb robotic rehabilitation: literature review and challenges. *J. Robot.* **2011**, *2011*, 1–11. [CrossRef]
16. del Ama, A.; Koutsou, A.; Moreno, J. Review of hybrid exoskeletons to restore gait following spinal cord injury. *J. Rehabil. Res. Dev.* **2012**, *49*, 497–514. [CrossRef] [PubMed]
17. Zhang, M.; Davies, T.C.; Xie, S. Effectiveness of robot-assisted therapy on ankle rehabilitation—A systematic review. *J. Neuroeng. Rehabil.* **2013**, *10*, 30. [CrossRef] [PubMed]
18. Jamwal, P.K.; Hussain, S.; Xie, S.Q. Review on design and control aspects of ankle rehabilitation robots. *Disabil. Rehabil. Assist. Technol.* **2015**, *10*, 93–101. [CrossRef] [PubMed]
19. Schmitt, C.; Métrailler, P.; Al-Khodairy, A. The Motion Maker^{TM}: A Rehabilitation System Combining an Orthosis with Closed-Loop Electrical Muscle Stimulation. In Proceedings of the 8th Vienna International Workshop on Functional Electrical Stimulation, Vienna, Austria, 10–13 September 2004; pp. 117–120.
20. Peshkin, M.; Brown, D.; Santos-Munné, J. KineAssist: A robotic overground gait and balance training device. In Proceedings of the 9th IEEE International Conference on Rehabilitation Robotics, (ICORR'05), Chicago, IL, USA, 28 June–1 July 2005; pp. 241–246.
21. Abdullah, H.; Tarry, C.; Datta, R.; Mittal, G.; Abderrahim, M. Dynamic biomechanical model for assessing and monitoring robot-assisted upper-limb therapy. *J. Rehabil. Res. Dev.* **2007**, *44*, 43–62. [CrossRef]
22. Van Delden, A.; Peper, C.; Kwakkel, G. A Systematic Review of Bilateral Upper Limb Training Devices for Poststroke Rehabilitation. *Stroke Res. Treat.* **2012**, *2012*, 1–17. [CrossRef]
23. Wang, C.; Wang, L.; Qin, J.; Wu, Z.; Duan, L.; Li, Z.; Cao, M.; Li, W.; Lu, Z.; Li, M.; et al. Development of an ankle rehabilitation robot for ankle training. In Proceedings of the Information and Automation, 2015 IEEE International Conference on IEEE, Lijiang, China, 8–10 August 2015; pp. 94–99.
24. Ayas, M.S.; Altas, I.H.; Sahin, E. Fractional order based trajectory tracking control of an ankle rehabilitation robot. *Trans. Inst. Meas. Control* **2018**, *40*, 550–564. [CrossRef]
25. Ayas, M.S.; Altas, I.H. Fuzzy logic based adaptive admittance control of a redundantly actuated ankle rehabilitation robot. *Control Eng. Pract.* **2017**, *59*, 44–54. [CrossRef]
26. Syrseloudis, C.E.; Emiris, I.Z. A parallel robot for ankle rehabilitation-evaluation and its design specifications. In Proceedings of the 8th IEEE International Conference on BioInformatics and BioEngineering, Athens, Greece, 8–10 October 2008.
27. Saglia, J.; Tsagarakis, N.; Dai, J.; Caldwell, D. Control Strategies for Patient-Assisted Training Using the Ankle Rehabilitation Robot (ARBOT). *IEEE/ASME Trans. Mechatron.* **2012**, *99*, 1–10. [CrossRef]
28. Patel, Y.D.; George, P.M. Parallel Manipulators Applications—A Survey. *Mod. Mech. Eng.* **2012**, *2*, 57–64. [CrossRef]
29. Girone, M.; Burdea, G.; Bouzit, M. The Rutgers Ankle orthopedic rehabilitation interface. In *Dynamic Systems and Control Division*; ASME: New York, NY, USA, 1999; Volume 67, pp. 305–312.
30. Girone, M.; Burdea, G.; Bouzit, M.; Popescu, V.; Deutsch, J. Orthopedic rehabilitation using the "Rutgers ankle" interface. *Stud. Health Technol. Inform.* **2000**, *70*, 89–95. [PubMed]
31. Deutsch, J.E.; Lewis, J.A.; Burdea, G. Technical and patient performance using a virtual reality-integrated telerehabilitation system: Preliminary finding. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2007**, *15*, 30–35. [CrossRef] [PubMed]
32. Saglia, J.A.; Tsagarakis, N.G.; Dai, J.S.; Caldwell, D.G. Control strategies for ankle rehabilitation using a high performance ankle exerciser. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA2010), Anchorage, AK, USA, 3–7 May 2010; pp. 2221–2227.

33. Yoon, J.; Ryu, J.; Lim, K. Reconfigurable ankle rehabilitation robot for various exercises. *J. Robot. Syst.* **2006**, *11*, 15–33. [CrossRef]
34. Liu, G.; Gao, J.; Yue, H.; Zhang, X.; Lu, G. Design and kinematics simulation of parallel robots for ankle rehabilitation. In Proceedings of the International Conference on Mechatronics and Automation, Luoyang, China, 25–28 June 2006; pp. 1109–1113.
35. Wang, C.; Fang, Y.; Guo, S.; Zhou, C. Design and kinematic analysis of redundantly actuated parallel mechanisms for ankle rehabilitation. *Robotica* **2015**, *33*, 366–384. [CrossRef]
36. Fan, Y.; Yin, Y. Mechanism design and motion control of a parallel ankle joint for rehabilitation robotic exoskeleton. In Proceedings of the IEEE Robotics and Biomimetics, Guilin, China, 19–23 December 2009; pp. 2527–2532.
37. Mao, Y.; Agrawal, S.K. Design of a cable-driven arm exoskeleton (CAREX) for neural rehabilitation. *IEEE Trans. Robot.* **2012**, *28*, 922–931. [CrossRef]
38. Vallés, M.; Cazalilla, J.; Valera, Á.; Mata, V.; Page, Á.; Díaz-Rodríguez, M. A 3-PRS parallel manipulator for ankle rehabilitation: towards a low-cost robotic rehabilitation. *Robotica* **2017**, *35*, 1939–1957. [CrossRef]
39. Azcaray, H.; Blanco, A.; García, C.; Adam, M.; Reyes, J.; Guerrero, G.; Guzmán, C. Robust GPI Control of a New Parallel Rehabilitation Robot of Lower Extremities. *Int. J. Control Autom. Syst.* **2018**, *16*, 2384–2392. [CrossRef]
40. Chang, T.C.; Zhang, X.D. Kinematics and reliable analysis of decoupled parallel mechanism for ankle rehabilitation. *Microelectron. Reliab.* **2019**, *99*, 203–212. [CrossRef]
41. Zhang, M.; McDaid, A.; Veale, A.J.; Peng, Y.; Xie, S.Q. Adaptive trajectory tracking control of a parallel ankle rehabilitation robot with joint-space force distribution. *IEEE Access* **2019**, *7*, 85812–85820. [CrossRef]
42. Li, J.; Fan, W.; Dong, M.; Rong, X. Implementation of passive compliance training on a parallel ankle rehabilitation robot to enhance safety. *Ind. Robot. Int. J. Robot. Res. Appl.* **2020**.10.1108/ir-02-2020-0040. [CrossRef]
43. Patanè, F.; Cappa, P. A 3-DOF Parallel Robot With Spherical Motion for the Rehabilitation and Evaluation of Balance Performance. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2011**, *19*, 157–166. [CrossRef]
44. Billard, A.; Calinon, S.; Dillmann, R.; Schaal, S. Robot Programming by Demonstration. In *Springer Handbook of Robotics*; Springer: Berlin/Heidelberg, Germany, 2008; Chapter 59, pp. 1371–1394.
45. Ravichandar, H.; Polydoros, A.S.; Chernova, S.; Billard, A. Recent advances in robot learning from demonstration. *Annu. Rev. Control Robot. Auton. Syst.* **2020**, *3*, 297–330. [CrossRef]
46. Fong, J.; Tavakoli, M. Kinesthetic teaching of a therapist's behavior to a rehabilitation robot. In Proceedings of the 2018 International Symposium on Medical Robotics (ISMR), IEEE, Atlanta, GA, USA, 1–3 March 2018; pp. 1–6.
47. Ma, Z.; Ben-Tzvi, P.; Danoff, J. Hand rehabilitation learning system with an exoskeleton robotic glove. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2015**, *24*, 1323–1332. [CrossRef] [PubMed]
48. Lauretti, C.; Cordella, F.; Guglielmelli, E.; Zollo, L. Learning by demonstration for planning activities of daily living in rehabilitation and assistive robotics. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1375–1382. [CrossRef]
49. Wang, H.; Chen, J.; Lau, H.Y.; Ren, H. Motion planning based on learning from demonstration for multiple-segment flexible soft robots actuated by electroactive polymers. *IEEE Robot. Autom. Lett.* **2016**, *1*, 391–398. [CrossRef]
50. Bhattacharjee, T.; Lee, G.; Song, H.; Srinivasa, S.S. Towards robotic feeding: Role of haptics in fork-based food manipulation. *IEEE Robot. Autom. Lett.* **2019**, *4*, 1485–1492. [CrossRef]
51. Ijspeert, A.J.; Nakanishi, J.; Schaal, S. Nonlinear Dynamical Systems for Imitation with Humanoid Robots. In Proceedings of the IEEE International Conference on Humanoid Robots (Humanoids), Seoul, Korea, 21–26 May 2001; pp. 219–226.
52. Moore, K.; Chen, Y.; Ahn, H.S. Iterative Learning Control: A Tutorial and Big Picture View. In Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 13–15 December 2006; pp. 2352–2357.
53. Ijspeert, A.; Nakanishi, J.; Hoffmann, H.; Pastor, P.; Schaal, S. Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Comput.* **2013**, *25*, 328–373. [CrossRef] [PubMed]
54. Abu-Dakka, F.J.; Nemec, B.; Jørgensen, J.A.; Savarimuthu, T.R.; Krüger, N.; Ude, A. Adaptation of manipulation skills in physical contact with the environment to reference force profiles. *Auton. Robot.* **2015**, *39*, 199–217. [CrossRef]

55. Abu-Dakka, F.J.; Kyrki, V. Geometry-aware dynamic movement primitives. In Proceedings of the IEEE International Conference on Robotics and Automation, Paris, France, 31 May–31 August 2020.
56. Bristow, D.; Tharayil, M.; Alleyne, A. A survey of iterative learning control. *IEEE Control Syst. Mag.* **2006**, *26*, 96–114.
57. Abu-Dakk, F.J.; Valera, A.; Escalera, J.A.; Vallés, M.; Mata, V.; Abderrahim, M. Trajectory adaptation and learning for ankle rehabilitation using a 3-PRS parallel robot. In *Proceedings of the International Conference on Intelligent Robotics and Applications*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 483–494.
58. Dul, J.; Johnson, G. A kinematic model of the human ankle. *J. Biomed. Eng.* **1985**, *7*, 137–143. [CrossRef]
59. Alcocer, W.; Vela, L.; Blanco, A.; Gonzalez, J.; Oliver, M. Major Trends in the Development of Ankle Rehabilitation Devices. *Dyna* **2012**, *79*, 45–55.
60. Siegler, S.; Chen, J.; Schneck, C. The three-dimensional kinematics and flexibility characteristics of the human ankle and subtalar joints–Part I: Kinematics. *J. Biomech. Eng.* **1988**, *110*, 364–373. [CrossRef]
61. Parenteau, C.S.; Viano, D.C.; Petit, P. Biomechanical properties of human cadaveric ankle-subtalar joints in quasi-static loading. *J. Biomech. Eng.* **1998**, *120*, 105–111. [CrossRef] [PubMed]
62. Kearney, R.; Weiss, P.; Morier, R. System identification of human ankle dynamics: Intersubject variability and intrasubject reliability. *Clin. Biomech.* **1990**, *5*, 205–217. [CrossRef]
63. Kleipool, R.P.; Blankevoort, L. The relation between geometry and function of the ankle joint complex: A biomechanical review. *Knee Surg. Sports Traumatol. Arthrosc.* **2010**, *18*, 618–627. [CrossRef]
64. Safran, M.; Benedetti, R.; Bartolozzi, A. Lateral ankle sprains: A comprehensive review: Part 1: Etiology, pathoanatomy, histopathogenesis, and diagnosis. *Med. Sci. Sports Exerc.* **1999**, *31*, 429–437. [CrossRef]
65. DiStefano, L.J.; Padua, D.A.; Brown, C.N.; Guskiewicz, K.M. Lower extremity kinematics and ground reaction forces after prophylactic lace-up ankle bracing. *J. Athl. Train.* **2008**, *43*, 234–241. [CrossRef]
66. Takao, M.; Uchio, Y.; Naito, K.; Fukazawa, I.; Ochi, M. Arthroscopic assessment for intra-articular disorders in residual ankle disability after sprain. *Am. J. Sports Med.* **2005**, *33*, 686–692. [CrossRef] [PubMed]
67. Ren, Y.; Xu, T.; Wang, L.; Yang, C.; Guo, X.; Harvey, R.; Zhang, L.Q. Develop a Wearable Ankle Robot for in-Bed Acute Stroke Rehabilitation. In Proceedings of the 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Boston, MA, USA, 30 August–3 September 2011; pp. 7483–7486.
68. Cioi, D.; Kale, A.; Burdea, G.; Engsberg, J.; Janes, W.; Ross, S. Ankle Control and Strength Training for Children with Cerebral Palsy Using the Rutgers Ankle CP: A Case Study. In Proceedings of the IEEE international conference on rehabilitation robotics, Zurich, Switzerland, 29 June–1 July 2011; pp. 654–659.
69. Díaz-Rodríguez, M.; Mata, V.; Valera, A.; Page, A. A Methodology for Dynamic Parameters Identification of 3-DOF Parallel Robots in Terms of Relevant Parameters. *Mech. Mach. Theory* **2010**, *45*, 1337–1356. [CrossRef]
70. Garcia de Jalon, J.; Bayo, E. *Kinematic and Dynamic Simulation of Multibody Systems: The Real Time Challenge*; Springer: New York, NY, USA, 1994.
71. Nakanishi, J.; Morimoto, J.; Endo, G.; Cheng, G.; Schaal, S.; Kawato, M. Learning from demonstration and adaptation of biped locomotion. *Robot. Auton. Syst.* **2004**, *47*, 79–91. [CrossRef]
72. Gams, A.; Ijspeert, A.J.; Schaal, S.; Lenarčič, J. On-line learning and modulation of periodic movements with nonlinear dynamical systems. *Auton. Robot.* **2009**, *27*, 3–23. [CrossRef]
73. Peters, J.; Schaal, S. Reinforcement learning of motor skills with policy gradients. *Neural Netw.* **2008**, *21*, 682–697. [CrossRef]
74. Nemec, B.; Abu-Dakka, F.J.; Ridge, B.; Ude, A.; Jorgensen, J.; Savarimuthu, T.R.; Jouffroy, J.; Petersen, H.G.; Kruger, N. Transfer of assembly operations to new workpiece poses by adaptation to the desired force profile. In Proceedings of the 16th International Conference on Advanced Robotics (ICAR13), Montevideo, Uruguay, 25–29 November 2013 ; pp. 1–7.
75. Abu-Dakka, F.J.; Nemec, B.; Kramberger, A.; Buch, A.G.; Krüger, N.; Ude, A. Solving peg-in-hole tasks by human demonstration and exception strategies. *Ind. Robot Int. J.* **2014**, *41*, 575–584. [CrossRef]
76. Ude, A.; Gams, A.; Asfour, T.; Morimoto, J. Tasks-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Trans. Robot.* **2010**, *26*, 800–815. [CrossRef]
77. Villani, L.; De Schutter, J. Force Control. In *Springer Handbook of Robotics*; Siciliano, B., Khatib, O., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; Chapter 7, pp. 161–185.

78. Cazalilla, J.; Vallés, M.; Mata, V.; Díaz-Rodríguez, M.; Valera, A. Adaptive control of a 3-DOF parallel manipulator considering payload handling and relevant parameter models. *Robot. Comput. Integr. Manuf.* **2014**, *30*, 468–477. [CrossRef]
79. Vallés, M.; Díaz-Rodríguez, M.; Valera, A.; Mata, V.; Page, A. Mechatronic Development and Dynamic Control of a 3-DOF Parallel Manipulator. *Mech. Des. Struct. Mach.* **2012**, *40*, 434–452. [CrossRef]
80. Nemec, B.; Petric, T.; Ude, A. Force adaptation with recursive regression Iterative Learning Controller. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, Hamburg, Germany, 28 September–2 October 2015; pp. 2835–2841.
81. Likar, N.; Nemec, B.; Zlajpah, L.; Ando, S.; Ude, A. Adaptation of bimanual assembly tasks using iterative learning framework. In Proceedings of the IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), Seoul, Korea, 3–5 November 2015; pp. 771–776.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.