Robotica (2022), **40**, pp. 4512–4534
doi:10.1017/S0263574722001084

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

RBF neural network-based admittance PD control for knee rehabilitation robot

Karam Almaghout[1,2], Bahram Tarvirdizadeh[1,*], Khalil Alipour[1] and Alireza Hadi[1]

[1]Advanced Service Robots (ASR) Lab., Department of Mechatronics Engineering, Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran and [2]Institute of Robotics and Computer Vision, Innopolis University, Innopolis, Russia
*Corresponding author. E-mail: bahram@ut.ac.ir

Received: 12 April 2022; **Revised:** 14 June 2022; **Accepted:** 4 July 2022; **First published online:** 3 August 2022

Keywords: rehabilitation robotics system, sliding mode control, radial basis function, interaction torques, motion recovery

Abstract

Early-stage rehabilitation therapy for post-stroke patients consists of intensive and accurate training sessions. During these sessions, the therapist moves the patient's joint within its range of motion repetitively. Patients, at this stage, often cannot control their muscles, and neurological disorders may occur and lead to undesirable movements. Thus, the therapist should train the joint gently to handle any sudden involuntary movements. Otherwise, the joint may undergo excessive torques, which may injure it. In this paper, we address this case and develop a clinical rehabilitation robotic system for training the knee joint taking into account the occurrence of these undesirable movements. The developed system has an innovative mechanism to measure interaction torques exerted by involuntary movements. Then, we introduce a new control approach consisting of an admittance controller and a proportional-derivative controller augmented by a radial basis function (PD-RBF) neural network. The PD-RBF guides the robot joint along a predefined trajectory, while the admittance part tracks any sudden interaction torques and updates the predefined trajectory accordingly. Thus, the robot trains the knee joint and once an undesirable movement occurs the robot gets along with this movement smoothly, then it gets back to the predefined trajectory. To validate the performance of the proposed admittance PD-RBF controller, we consider two controllers, an admittance adaptive sliding mode control and an admittance conventional PD one. Then, a compatarive study is conducted on these controllers via real-world experiments. The obtained results verify the efficiency of the admittance PD-RBF and prove its superiority over the other aforementioned controllers.

1. Introduction

Rehabilitation therapy aims to restore and regain motor function of the limb after stroke or some other cases like accidents. The hope is that through physical exercises, the patient can recover the lost or weakened function. However, these types of exercises are long-term processes, and conventional rehabilitation is not only time consuming, tedious, and exhausting for the physiotherapist but also imposes a great economic onus on the patient. Rehabilitation robots are presented to overcome these drawbacks of traditional rehabilitation and perform the therapy training more intelligently, accurately, and comfortably. These robots allow intensifying the therapy process by increasing the number of repetitions and the exercise duration without any further physical effort by the expert nor losing the accuracy of the training. Moreover, these robotic systems can acquire and save data during the exercise, like the position, velocity, and interaction torques, for online monitoring and further analysis. Such facilities may not be available in conventional rehabilitation, easily.

Over the last two decades, many lower-limb rehabilitation robotic systems have been developed to aid stroke survivors in performing various styles of rehabilitation exercises, to recover mobility of the limb. These systems can be divided, according to the exercise posture, into two main categories [1]. The first category involves the robots for training in sitting/lying positions. These robots are used in the early stage of therapy where patients cannot stand or walk safely due to high muscle weakness [2, 3]. Robots in this

© The Author(s), 2022. Published by Cambridge University Press.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

category assist the patient in strengthening muscles, developing endurance, and increasing joint mobility and movement coordination. The second category includes the robots for training in walking/standing position, either overground or treadmill [4]. The main aim of these robots is to drive the patient's lower limb, who can stand and has adequate endurance, on a gait trajectory.

The main task of the rehabilitation robotic systems is to move the joints of the injured limb along specific trajectories. To perform this task precisely, various controller approaches have been developed. Proportional integral derivative (PID) controller and its types, PD and PI, are also used in rehabilitation robotic systems. In ref. [5], Naghavi et al. implement a PID control loop to perform trajectory tracking control for a knee rehabilitation robot. A neuromuscular electrical stimulation device is controlled by an adaptive PID controller with an online tuning parameter to perform exercises for the upper limb [6, 7]. In ref. [8], researchers propose an intelligent Proportional-Derivative (PD) controller with a sigmoid function-based tracking differentiator to achieve motion control of a 12-Degree-Of-Freedom (12-DOF) lower-limb exoskeleton robot. Yousefi et al. develop a sliding-backstepping control for a knee rehabilitation robot [9]. Riani et al. develop a robust adaptive integral terminal sliding mode control approach, designed to control an upper-limb rehabilitation exoskeleton [10]. Oliviera et al. introduce a robust sliding mode control and implemented it to a neuromuscular electrical stimulation device to achieve a predetermined angular trajectory for the upper limb [11]. Neural networks (NNs) models have powerful approximation performance and self-learning ability, which can compensate for the uncertainty of the mathematical model of the robotic systems and handle any external disturbances. Radial basis function (RBF), compared to other types of NNs, for example, Markovian NNs [12], has the best approximation, simple training, fast learning convergence speed, and avoids falling into a local minimum. Moreover, it can improve the real-time tracking performance of the controller [13, 14]. Thus, it has been widely used in rehabilitation robotic systems controllers. In ref. [15], researchers propose an adaptive sliding mode control (ASMC) based on RBF NNs for a lower-limb rehabilitation robot for active therapy sessions. The RBF block is introduced to compensate for external disturbances and uncertainties. Shi et al. [16] develop an RBF-based PID control to achieve the trajectory tracking task of a lower-limb rehabilitation robot.

The safety of the patient during the therapy is a crucial criterion in designing rehabilitation robotic systems. Many studies cover hazards caused by issues such as misalignment when an exoskeleton robot does not align with the human limb properly and exceeds the normal range of motion [17–19]. On the contrary, investigation about the safety of rehabilitation robots to handle involuntary movements is limited. Involuntary movements, such as myoclonus[1], are common after stroke, due to muscle disorders [20, 21]. During the training exercise, these movements may occur and exert interaction torques that lead to substantial injuries or damage to the patient's limb. Thus, a safe interaction between the patient and the robot is vital [22]. This criterion requires having a compliant behavior of the robot to avoid harming the human (patient). Admittance control is widely implemented in the literature to ensure safe interactions between humans and robots in many applications (medical, industrial, etc.) [23, 24]. It maps interaction forces/torques into position/velocities, and thus it makes the robot complies with any external forces applied by a human and changes its behavior. In rehabilitation robots, admittance control has been implemented to enable the participation of the patient during the exercise. This participation is mainly required for therapy stages, where the patient can control their limb, but they are still weak to move independently [25, 26].

This research presents a stationary knee rehabilitation robotic system (KRRS), to be used in clinical therapy. KRRS has a new innovative mechanism to measure the interaction torques by load cells. The robot is controlled by the therapist using a user interface (UI), where he/she can monitor the session in real time and save the results for further analysis. A novel admittance proportional-derivative controller augmented by a radial basis function (PD-RBF) control approach is developed for KRRS. This proposed controller consists of two loops, an outer loop and an inner loop. The former has an admittance control algorithm to handle the undesirable interaction torques between the patient's limb and the robot. The

[1]Myoclonus is a brief and involuntary twitching of muscles.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

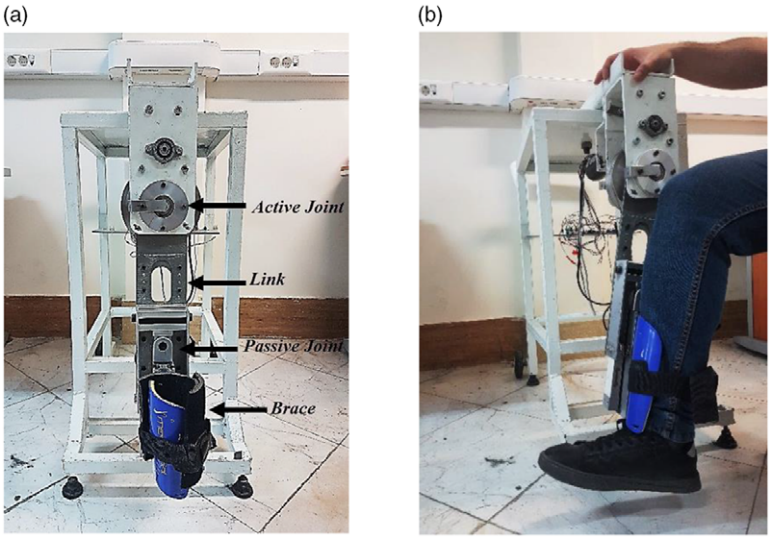


Figure 1. *(a) Prototype of the KRRS; (b) the shank connected to the KRRS.*

latter achieves the trajectory tracking task. We suggest an RBF NN to augment the conventional PD controller as a feedforward loop. This RBF compensates for the nonlinear characteristics of the system and improves the performance of the PD controller with no need for prior knowledge of the dynamic model of the system. To validate the efficiency of this PD-RBF controller, we consider two controllers, a nonlinear model-based controller, ASMC, and a linear model-free one, PD. Then, we evaluate the proposed PD-RBF by a comparative study among these controllers. The integral absolute error (IAE) and the control effort (CE) are defined as performance indices and analyzed for the mentioned controllers. The main contributions of this paper are:

1. Design a new mechanism to measure the interaction torques exerted by the patient's involuntary movements during the therapy session.
2. Propose a new model-free controller to handle these measured torques smoothly and safely.

The remainder of this paper is structured in the following manner. In the next section, the parts of the robot prototype (mechanical components, driver, sensors, etc.) are described. In Section 3, the dynamic model of the robot is extracted. Section 4 includes the design of various control strategies. The experiment results are presented in Section 5. Section 6 includes the results discussion and the comparative study. Finally, the conclusion of this research is abstracted in Section 7.

2. System description

Although determinants of successful lower-limb rehabilitation training robots are unknown, the following properties seem to be important: safety, comfort, accuracy, and ability to perform various types of exercises with desired intensity and number of repetitions. Taking into account the aforementioned properties along with the ease of use by the therapist and low cost, a prototype of KRRS was designed to assist the patient in exercising the knee joint in the sagittal plane. KRRS is designed for the early stages of rehabilitation therapy. It trains the patient's limb in a sitting position, which is suitable for long-period training sessions, especially for patients not able to stand without assistance/support. Figure 1(a) represents the fabricated prototype of the mentioned robotic system in the ASR Laboratory at the University of Tehran and (b) illustrates how the patient's leg is attached to the robot.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

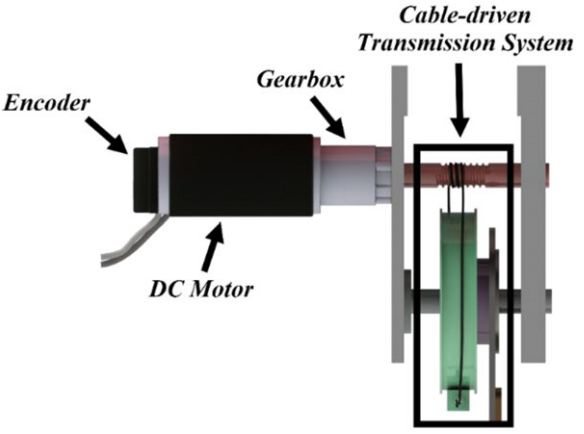


Figure 2. *DC motor with transmission system.*

2.1. System design

The KRRS mainly consists of a link with a pivot joint that simulates the knee movement. The joint is actuated by a geared DC motor through a cable-driven transmission system. An optical encoder is mounted on the motor shaft to measure the joint position. The motor, transmission system, and encoder are shown in Fig. 2, and their specifications are illustrated in Table 1.

The patient's shank is attached to the link in parallel by a brace, as shown in Fig. 1(b). The length of the link can be adjusted in a range between 45 and 63 cm, with an adjustment step of 1.5 cm. This feature permits the KRRS to fit the various lengths of the patient's shank.

2.1.1. Involuntary movements measurements mechanism

To measure potential involuntary movements of the knee, we developed a new mechanism based on the bar load cells. We used two straight bar load cells, mounted on the link sides. The brace is connected to the KRRS link via a passive joint. Once the knee moves involuntarily, this passive joint allows the brace to move and hit one of the load cells. The motion range of the passive joint is limited, and it is restricted by two high-stiffness springs at each side. Figure 3 represents (a) the schematic and (b) the CAD model of the load cells positions along with the brace, springs, and passive joint.

2.2. Control unit

Since the proposed KRRS works in direct contact with a human, any delay in the control loop may endanger the patient's safety. For instance, the delay in acquiring the measured interaction force will lead to mishandle this excessive force and the robot may exceed the predefined limits if there is a delay in reading the actual angular position. Such cases can cause serious harm to the patient's leg. Thus, it is essential that the control unit should work in real time and avoid any delay in both receiving commands from the physiotherapist and acquiring data from the sensors.

Even PCs provide the ability to design control software easily using high-level programming language environments such as MATLAB and LabVIEW. However, they are not designed to achieve uninterrupted real-time control [27]. They share their resources between the control software and other system activities. Some of these activities are under user control, while some others are not. This affects the speed and repeatability of the operation of the robot and is out of user's control. Moreover, when a PC freezes or undergoes hard drive failure, due to some of these activities, the I/O unit controlling the robot will not be often neither controllable nor monitored.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Table I. *Specification of actuating system.*

Brand	**KORMAS**
Model	671-145-66-E
Supply voltage	24 V
Nominal torques	4 N.m
Nominal current	6.5 A
Gearbox rRatio	1:9
Output shaft nominal speed	250 rpm
Transmission system ratio	1:10
Encoder resolution	100 pulses

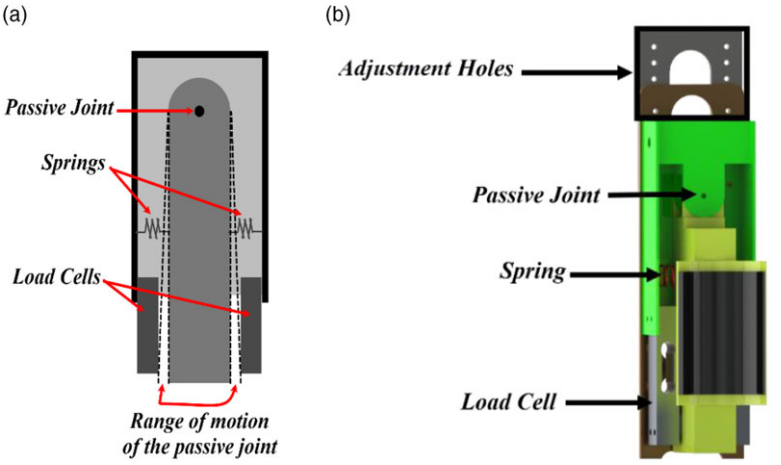


Figure 3. *(a) The schematic and (b) CAD model of the passive joint, brace, springs, and load cells.*

To provide a real-time system, an ARM Cortex-M4-based STM32 microcontroller (model number: STM32F407ZGT6) is utilized as the control unit of KRRS. This microcontroller is programmed by C programming language, and it works on frequency 168 MHz. All control procedures of the robot are implemented by this microcontroller. On the other hand, a host PC is connected to the control unit. Using a UI on this host PC, the therapist can observe the therapy session, as well as command the control unit. Another AVR Atmega328-based board is used for safety, which controls a relay that shuts down the whole system immediately when the emergency pushbutton is pressed.

2.3. User interface

Based on C# programming language, an appropriate and user-friendly UI has been developed. This UI is available as an executable file (∗.exe). Thus, it can be installed on any Windows-based PC. By this UI, a physiotherapist can perform the following functions:

- Send the main commands to the control unit: start, stop, pause, and resume.
- Select the suitable range of motion for the patient within the range of motion of the knee. Figure 4 shows the range of motion of the tibia with respect to the vertical axis according to ref. [28].
- Set the exercise mode: passive or active. In both modes, the robot drives the patient's limb on a reference trajectory. Patients' interaction forces are not considered in the passive mode. While in the active mode, these interaction forces are measured and considered during the session.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

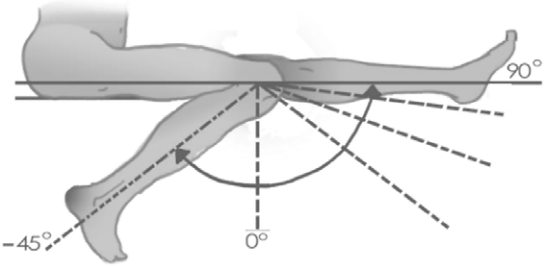


Figure 4. *Range of motion of the knee joint.*

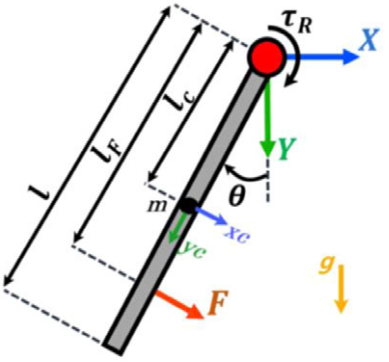


Figure 5. *The schematic of the robot.*

- Monitor data of the therapy session in real time: these data are divided into two sets:

The first set of data is for evaluating the therapy progress of the patient: position and interaction torques. The other set is for evaluating the performance of the robot: tracking error, control output, and CE. Moreover, these data can be saved to an Excel file for further analysis.

3. Dynamic model

To extract the dynamic model of the system, the schematic of the robot is illustrated in Fig. 5. In this schematic, we did not consider the passive joint as part of the dynamic model due to its significantly low range of motion and low motion frequency; hence, it does not have dynamic behavior. The dynamic model of the robot is obtained using the Euler–Lagrange method as follows:

The Lagrangian of the KRRS is given by [29]

$$\mathcal{L} = \mathcal{K} - \mathcal{P} \tag{1}$$

where $\mathcal{K}$ and $\mathcal{P}$ are the kinetic and potential energies, respectively, given as follows:

$$\mathcal{K} = \frac{1}{2}\left(m\dot{x}_c^2 + m\dot{y}_c^2 + I_1\dot{\theta}^2\right) \tag{2}$$

$$\mathcal{P} = -mgy_c \tag{3}$$

where m is the mass of the robot arm and g is the gravitational acceleration.

x_c and y_c represent the position of the centroid of the link with respect to the origin of the coordinates system at its joint, which can be obtained as follows:

$$x_c = -l_c sin\theta \tag{4}$$

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

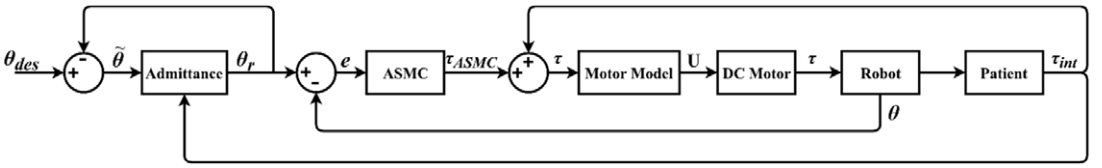


Figure 6. *Block diagram of the ASMC.*

$$y_c = l_c cos\theta \tag{5}$$

where l_c is the distance between the center of mass of the robot arm and the joint and θ and $\dot{\theta}$ are the angular position and velocity of the robot joint, respectively.

$\dot{x}_c$ and $\dot{y}_c$ are velocity of the center of mass of the arm and given as

$$\dot{x}_c = -l_c\dot{\theta}cos\theta \tag{6}$$

$$\dot{y}_c = -l_c\dot{\theta}sin\theta \tag{7}$$

I_1 is the moment of inertia of the robot arm at the joint and given by the parallel axis theorem:

$$I_1 = I_c + ml_c^2 \tag{8}$$

where I_c is the moment of inertia of the robot at the center of mass.

Then, the dynamic model is given as follows:

$$\tau = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \left(\frac{\partial L}{\partial \theta}\right)$$

$$= (I_c + 2ml_c^2)\ddot{\theta} + mgl_c sin\theta$$

$$= M_R\ddot{\theta} + G_R(\theta) \tag{9}$$

where $\ddot{\theta}$ is the angular acceleration of the robot joint. M_R and $G_R(\theta)$ are the nominal inertia and gravity terms, respectively, which are expressed as

$$M_R = I_c + ml_c^2 \tag{10}$$

$$G_R(\theta) = mgl_c cos\theta \tag{11}$$

τ is the external torque applied on the robot arm, and it is obtained as follows:

$$\tau = \tau_R - \tau_{int} \tag{12}$$

where τ_R is the actuator torque and τ_{int} is the interaction torque between the robot and the patient's leg which can be derived as follows:

$$\tau_{int} = Fl_F \tag{13}$$

where F denotes the interaction force between the robot and the patient, excluding the weight, inertia, and other elastic/viscoelastic torques in the knee joint, which is the force exerted by the leg and measured by the load cells. l_F is the distance between the load cell and the robot joint, as shown in Fig. 6.

In practice, the nominal values of the dynamic model of the robot do not represent the real dynamic behavior of the system, due to many factors. These factors are either unknown, like system uncertainties and external disturbances, or difficult to compute accurately, like patient's leg. A new term, $D(t)$, is defined that includes all unknown factors, where $D(t)$ is assumed to be unknown, but bounded. Thus, the dynamic model of the KRRS is given as follows:

$$\tau - \tau_{int} = M_R\ddot{\theta} + G_R(\theta) + D(t) \tag{14}$$

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Then the angular acceleration of the robot joint will be as follows:

$$\ddot{\theta} = M_R^{-1} \{\tau - \tau_{int} - G(\theta) - D(t)\} \tag{15}$$

4. Control design

In the KRRS, we develop the control algorithm to fulfill the following objectives: (a) perform an accurate trajectory tracking task and (b) handle the measured interaction torques exerted by the patient safely and appropriately.

To realize these objectives, a control algorithm consists of two loops: the outer control loop and the inner control loop. Both of these loops are implemented in the control unit (STM32 board). The functionality of these loops is explained as follows.

4.1. The outer control loop

During the exercise session by the KRRS, the patient's limb may move either voluntarily by the patient or involuntary because of a neurological disorder. This movement generates an interaction torque. This torque is undesirable and may cause pain or injury for the patient. To reduce the effects of these interaction torques and avoid secondary injury, we implement an admittance (reverse impedance) control strategy. The undesirable interaction torques are measured by load cells. Then, they are applied to the input of admittance control. The admittance control attains the flexibility of the robot by changing its trajectory suitably according to the measured torques.

Using a mass-damper-spring impedance model [30], the interaction torque can be given as follows:

$$M_i\ddot{\tilde{\theta}} + C_i\dot{\tilde{\theta}} + K_i\tilde{\theta} = \tau_{int} \tag{16}$$

$$\tilde{\theta} = \theta_{des} - \theta_r \tag{17}$$

where M_i, C_i, K_i are the inertia, damping, and stiffness impedance terms, respectively. These terms should be chosen in a way that ensures a smooth and safe change in the reference trajectory. θ_{des} is the predefined trajectory and θ_r is the reference trajectory, which is the input of the inner control loop. $\tilde{\theta}$ is the robot trajectory deviation from the predefined trajectory. τ_{int} is the undesirable interaction torque. When $\tau_{int} = 0$, we have $\tilde{\theta} = 0$ and $\theta_r = \theta_{des}$, and when the interaction torque is exerted, the reference trajectory changes and deviates from the predefined trajectory. Substituting Eq. (17) in (16), the reference trajectory, which is proper to the exerted interaction torques, can be obtained as follows:

$$M_i\left(\ddot{\theta}_{des} - \ddot{\theta}_r\right) + C_i\left(\dot{\theta}_{des} - \dot{\theta}_r\right) + K_i\left(\theta_{des} - \theta_r\right) = \tau_{int} \tag{18}$$

$$\ddot{\theta}_r = \ddot{\theta}_{des} - M_i^{-1}\left\{C_i\left(\dot{\theta}_{des} - \dot{\theta}_r\right) + K_i\left(\theta_{des} - \theta_r\right) - \tau_{int}\right\} \tag{19}$$

4.2. The inner control loop

This loop achieves the trajectory tracking task, where the robot guides the leg to move along a reference trajectory with the highest possible accuracy. The tracking error, which is the difference between the reference angular position (θ_r) and the robot actual angular position (θ), and its first derivative are given as follows:

$$e = \theta_r - \theta \tag{20}$$

$$\dot{e} = \dot{\theta}_r - \dot{\theta} \tag{21}$$

A PD-RBF NN is proposed to be applied in this loop. However, before discussing the aforementioned proposed approach, two other controllers are introduced for a comparative study purpose: an ASMC,

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

as a representation of the model-based controller, and a conventional PD controller, as a model-free controller. The results of these controllers are used in a comparative study to prove the efficiency of the PD-RBF controller. The reasons behind choosing these two controllers, PD and ASMC, are

- PD, as a variation of PID, is the most basic and simplest controller for any system and one of the most widely used controllers for robots in trajectory tracking tasks. However, its performance is subject to the nonlinearity level of the system [31]. Therefore, since our system is nonlinear, we start from PD and seek to improve its performance for our system. Then validate the improved PD (PD-RBF) by comparing its performance with the conventional PD.
- Sliding mode control (SMC) has a robust property and a complex structure, which is commonly used to control nonlinear systems with known uncertain bounds [32]. ASMC is an improved version of SMC to control nonlinear systems when the boundaries of the uncertainty are unknown, which is more practical [33]. Thus, since the proposed PD-RBF is an advanced control, it is reasonable to examine its performance with a controller more complex than PD, such as ASMC, to have fair comparison.

4.2.1. Adaptive sliding mode control

In rehabilitation robots, many control methodologies based on SMC have been developed [9, 34–36]. In this research, an ASMC with PID surface has been introduced to perform trajectory tracking control.

The sliding surface can be defined as follows [37]:

$$s = \dot{e} + \beta_1 e + \beta_2 \int e dt \tag{22}$$

where β_1 and β_2 are positive gains.

The aim of the SMC is to force the error to approach the sliding surface and then move the error along the sliding surface toward the zero.

To achieve this aim, the control law of SMC should be designed in a way that the following expression is maintained:

$$\dot{s} = 0 \tag{23}$$

This condition is necessary to ensure that the error stays on the sliding surface.

where $\dot{s}$ is the time derivative of the sliding surface and derived as follows:

$$\begin{aligned} \dot{s} &= \ddot{e} + \beta_1 \dot{e} + \beta_2 e \\ &= \left(\ddot{\theta}_r - \ddot{\theta}\right) + \beta_1 \dot{e} + \beta_2 e \end{aligned} \tag{24}$$

Substituting (15) in (24) yields:

$$\dot{s} = \left(\ddot{\theta}_r - M_R^{-1} \left\{\tau - \tau_{int} - G\left(\theta\right) - D(t)\right\}\right) + \beta_1 \dot{e} + \beta_2 e \tag{25}$$

Equation (25) has known (nominal and measurable) and unknown, but bounded, terms. Herein, we introduce a control law that consists of two terms, an equivalent and auxiliary, the former considers the known term while the latter tries to vanish the impact of the unknown term. The control law is

$$\tau = \tau_{eq} + \tau_{aux} \tag{26}$$

making $\dot{s} = 0$, and neglecting the unknown term $D(t)$, the equivalent control law τ_{eq} is obtained as

$$\tau_{eq} = M_R \left(\ddot{\theta}_r + \beta_1 \dot{e} + \beta_2 e\right) + G_R\left(\theta\right) + \tau_{int} \tag{27}$$

The term of system uncertainties and disturbance ($D(t)$) leads the tracking error to drift away from the sliding surface. In order to overcome this issue, an auxiliary control law is defined. This auxiliary control law has a switching discontinuous function which forces the tracking error to lie on the sliding

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

surface. The auxiliary control law τ_{aux} is given as:

$$\tau_{aux} = M_R\,(K_1 s + K_2 sign\,(s)) \tag{28}$$

where K_1 and K_2 are positive gains. The sign function, *sign*(■), is the ideal discontinuous function, which guarantees an infinitely fast switching. However, in practice, it is impossible to achieve infinitely fast switching control because of the finite time delays for the control computation and limitations of the physical actuators. This imperfect switching in practice raises the oscillation problem, known as chattering. Chattering is a harmful phenomenon because it leads to low control accuracy, may damage the physical parts of the system, and requires lots of energy for good efficiency. A common method to reduce chattering is to define a boundary layer near the sliding surface and then use a saturation function instead of the sign function. When the system is within the boundary layer, a continuous control will be achieved and the chattering will be alleviated. The saturation function *sat*(■) is defined as follows:

$$sat\,(s) = \begin{cases} \varphi & s \geq \varphi \\ -\varphi & s \leq \varphi \\ s & -\varphi < s < \varphi \end{cases} \tag{29}$$

where φ is the boundary layer thickness.

Thus, τ_{aux} will be

$$\tau_{aux} = M_R\,(K_1 s + K_2 sat\,(s)) \tag{30}$$

Further, in order to improve the robustness of the SMC law, an adaptive term has been introduced.

Assumption 1:

The term D(t) is unknown but bounded, that is, $\Lambda > |D(t)|$, *where* Λ *is unknown positive number.* Thus, $\hat{\Lambda}$ is the estimated value of Λ and obtained from the adaptation law:

$$\dot{\hat{\Lambda}} = \kappa\,|s| \tag{31}$$

where κ is a positive number. $\hat{\Lambda}$ is added to τ_{aux} as follows:

$$\tau_{aux} = K_1 s + K_2 sat\,(s) + \hat{\Lambda} sat\,(s) \tag{32}$$

Finally, by substituting (27) and (32) in (26), the ASMC law, τ_{ASMC}, is given as

$$\tau_{ASMC} = M_R\left(\ddot{\theta}_r + \beta_1 \dot{e} + \beta_2 e\right) + G_R\,(\theta) + \tau_{int} + K_1 s + K_2 \text{sat}\,(s) + \hat{\Lambda} sat\,(s) \tag{33}$$

Figure 6 presents the block diagram of the admittance ASMC approach.

Stability analysis. To ensure the stability of the designed controller, a candidate Lyapunov positive definite function V is defined as follows:

$$V = \frac{1}{2}\kappa M_R s^2 + \tilde{\Lambda}^2 \tag{34}$$

where

$$\tilde{\Lambda} = \Lambda - \hat{\Lambda} \tag{35}$$

Assumption 2:

Assigning the initial value of $\hat{\Lambda}$ as $\hat{\Lambda}_0 \ll \Lambda$, where $\hat{\Lambda}_0$ is the value of $\hat{\Lambda}$ at $t = 0$, and for a relatively small value of κ in Eq. (31), $\hat{\Lambda}$ will increase slowly. Moreover, as much as the sliding surface s converges to

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

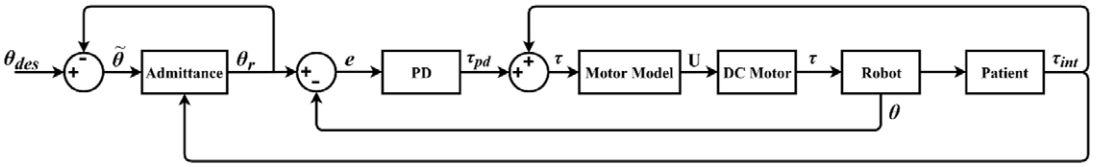


Figure 7. *Block diagram of the PD controller.*

zero, $\dot{\hat{\Lambda}}$ converges to zero. This will ensure that $\hat{\Lambda}$ will flatten quickly and will not exceed Λ. Thus, the value of $\tilde{\Lambda}$ will be ≥ 0. Generally, it is guaranteed that $\tilde{\Lambda}$ is positive, always.

The derivative of Lyapunov function is given as follows:

$$\dot{V} = \kappa M_R s\dot{s} - 2(\Lambda - \hat{\Lambda})\dot{\hat{\Lambda}} \tag{36}$$

Substituting (33), in (25), and, substituting the obtained formula and (31) in (36) lead to:

$$\begin{aligned}\dot{V} &= \kappa s\left(-K_1 s - K_2 sat\,(s) - \hat{\Lambda} sat\,(s) + (D(t)\right) - 2(\Lambda - \hat{\Lambda})\kappa\,|s| \\ &= -\kappa K_1 s^2 - \kappa s K_2 sat\,(s) - \kappa s\hat{\Lambda} sat\,(s) + \kappa s(D(t) - 2(\hat{\Lambda} - \Lambda)\kappa\,|s| \\ &\leq -\hat{\Lambda}\kappa\,|s| + |(D(t)|\,\kappa\,|s| - 2\left(\Lambda - \hat{\Lambda}\right)\kappa\,|s| + \Lambda\,|s| - \Lambda\,|s|\end{aligned} \tag{37}$$

Thus, the first derivative of Lyapunov function $\dot{V}$ can be written as follows:

$$\dot{V} \leq -\left(\Lambda - |(D(t)|\right)\kappa\,|s| - \left(\Lambda - \hat{\Lambda}\right)\kappa\,|s| \tag{38}$$

Since $|D(t)| < \Lambda$, according to the assumption 1, and by choosing $\hat{\Lambda}_0$ and κ properly (based on assumption 2), that provide the condition $\tilde{\Lambda} \geq 0$, $\dot{V}$ is negative definite, and according to the global stability theorem of Lyapunov, the stability of the system is guaranteed.

4.2.2. PD controller:

PID controller and its other types, PD and PI, are model-free controller where they do not require any knowledge about the dynamic system. These controllers are widely used for controlling both linear and nonlinear systems due to their simple structure, robustness, ease of implementation, low cost, and high performance [31, 38, 39]. The PD equation used in this research is described as follows:

$$\tau_{pd} = K_p e + K_d \dot{e} \tag{39}$$

The total torque to be applied on the robot is

$$\tau = \tau_{pd} + \tau_{int} \tag{40}$$

where K_p and K_d are the positive proportional and derivative gains, respectively. τ_{pd} is the PD controller output. Figure 7 shows the block diagram of PD controller.

4.2.3. PD augmented by RBF

Since the patient's limb is connected directly to the robot in parallel and the physical characteristics of the leg differ from one patient to another, the nonlinear characteristics of the robot vary dramatically for each patient. This change in the characteristics makes obtaining the accurate dynamic model difficult and affects the performance of the tracking control due to the high nonlinearity. To overcome this problem and provide the KRRS with highly precise tracking performance, we propose a PD controller augmented by an RBF NN block as feedforward loop. The aim of this feedforward loop is to compensate for the nonlinearity terms in the system and improve the tracking accuracy of the model-free controller (PD). RBF is widely implemented to approximate the nonlinearity due to its simplicity and fast convergence

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

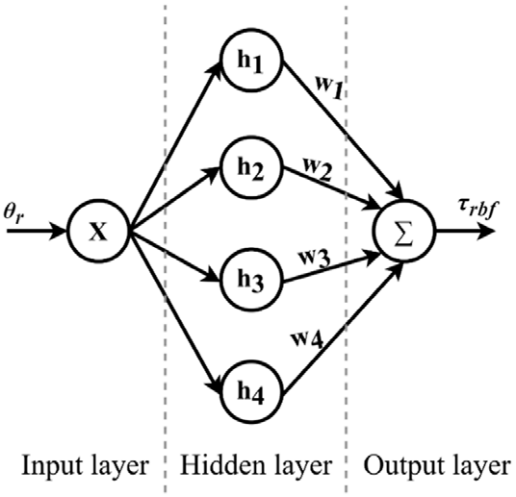


Figure 8. *The architecture of the RBF neural network.*

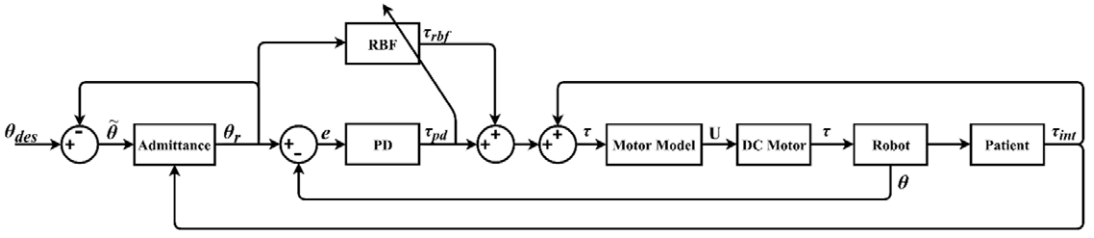


Figure 9. *Block diagram of the PD-RBF controller.*

[40, 41]. Figure 8 shows the architecture of the proposed RBF NN, which consists of an input layer, hidden layer, and output layer. Gaussian functions are used as an RBF in the hidden layer. Then, the output of RBF, τ_{rbf}, is obtained as follows:

$$h_i = \exp\left(-\frac{\|X - c_i\|^2}{2b_i^2}\right) \tag{41}$$

where $i = 1, \ldots, 4$ is the number of the neuron and h_i, c_i, b_i are the Gaussian function, center value, and the width values of the neuron i, respectively. X is the reference trajectory θ_r.

And the output of the RBF NN is given as follows:

$$\tau_{rbf} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{hsat}\left(\tau_{pd}\right) \tag{42}$$

where $\boldsymbol{w} \in \mathcal{R}^{4\times 1}$ is the weights vector.

Defining the weights adaptive law as follows:

$$\dot{\boldsymbol{w}} = \frac{1}{2}\mathbf{P}\,\boldsymbol{h}\left|\tau_{pd}\right| \tag{43}$$

where $\mathbf{P} \in \mathcal{R}^{4\times 4}$ is a positive diagonal matrix.

Thus, the proposed control output, shown in Fig. 9, is given as follows:

$$\tau = \tau_{rbf} + \tau_{pd} + \tau_{int} \tag{44}$$

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Stability analysis

To analyze the stability of the proposed control strategy, the following assumptions are considered, without any lake of generality:

Assumptions 3:

Define a new term includes the nonlinear and unknown terms of Eq. (15), this new term is unknown but bounded.

$$\Pi(\theta, t) = G(\theta) + D(t) \tag{45}$$

The ideal approximation of the NN output $\boldsymbol{w}^{*\mathrm{T}}\boldsymbol{h}$, that is, $\boldsymbol{w}^{*\mathrm{T}}\boldsymbol{h} > |\Pi(\theta, t)|$ where $\boldsymbol{w}^{*}$ denotes the best approximation of the NN weights.

Assumptions 4:

K_p and K_d are chosen to ensure that the following inequalities are true:

$$2K_pK_d > M_RK_d^{-1}K_p^2 \tag{46}$$

$$K_d^2 > M_RK_p \tag{47}$$

Rewriting the dynamic Eq. (15) as function of error and substituting Eqs. (45), (44), (42), and (39) in it as follows:

$$M_R\ddot{e} + K_d\dot{e} + K_p\mathrm{e} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{h}\mathrm{sat}\left(\tau_{pd}\right) - \Pi(t) = 0 \tag{48}$$

Defining the Lyapunov candidate as follows:

$$V = \frac{1}{2}M_RK_d^{-1}\left(K_pe + K_d\dot{e}\right)^2 + \frac{1}{2}\left(2K_pK_d - M_RK_d^{-1}K_p^2\right)e^2 + 2\tilde{\boldsymbol{w}}^{\mathrm{T}}\tilde{\boldsymbol{w}} \tag{49}$$

where

$$\tilde{\boldsymbol{w}} = \boldsymbol{w}^{*} - \boldsymbol{w} \tag{50}$$

Assumption 5:

Assigning the initial value of $\boldsymbol{w}$ as $\boldsymbol{w_0} \ll \boldsymbol{w}^{*}$, where $\boldsymbol{w_0}$ is the value of $\boldsymbol{w}$ at $t = 0$, and for a relatively small values of $\mathbf{P}$ in Eq. (43), $\boldsymbol{w}$ will increase slowly. Moreover, as much as the error converges to zero, $\dot{\boldsymbol{w}}$ converges to zero. This will ensure that $\boldsymbol{w}$ will be flatten quickly and will not exceed $\boldsymbol{w}^{*}$. Thus, the value of $\tilde{\boldsymbol{w}}$ will be $\geq \mathbf{0}$. Generally, it is guaranteed that $\tilde{\boldsymbol{w}}$ is positive, always.

Differentiating (49), we obtain the following relation for $\dot{V}$.

$$\begin{aligned}\dot{V} &= \left(K_pe + K_d\dot{e}\right)\left(M_RK_d^{-1}K_p\dot{e} + M_R\ddot{e}\right) + \left(2K_pK_d - M_RK_d^{-1}K_p^2\right)e\dot{e} - 4\tilde{\boldsymbol{w}}^{\mathrm{T}}\dot{\boldsymbol{w}} \\ &= M_RK_d^{-1}K_p^2e\dot{e} + M_RK_p\dot{e}\ddot{e} + M_RK_p\dot{e}^2 + M_RK_d\dot{e}\ddot{e} + \left(2K_pK_d - M_RK_d^{-1}K_p^2\right)e\dot{e} - 2\tilde{\boldsymbol{w}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right|\end{aligned} \tag{51}$$

Substituting (48) and (43) in (51):

$$\begin{aligned}\dot{V} = {} & M_RK_d^{-1}K_p^2e\dot{e} + \left(K_pe + K_d\dot{e}\right)\left(-K_pe - K_d\dot{e} - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{h}\mathrm{sat}\left(\tau_{pd}\right) + \Pi(t)\right) + M_RK_p\dot{e}^2 \\ & + \left(2K_pK_d - M_RK_d^{-1}K_p^2\right)e\dot{e} - 2\tilde{\boldsymbol{w}}^{\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right|\end{aligned} \tag{52}$$

Then, the derivative of the Lyapunov function can be written as follows:

$$\begin{aligned}\dot{V} &= -K_p^2e^2 - \left(K_d^2 - M_RK_p\right)\dot{e}^2 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{h}\mathrm{sat}\left(\tau_{pd}\right)\tau_{pd} + \Pi(t)\,\tau_{pd} - 2\tilde{\boldsymbol{w}}^{\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right| \\ &= -K_p^2e^2 - \left(K_d^2 - M_RK_p\right)\dot{e}^2 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{h}\left|\tau_{pd}\right| + \Pi(t)\,\tau_{pd} - 2\tilde{\boldsymbol{w}}^{\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right| + \boldsymbol{w}^{*\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right| - \boldsymbol{w}^{*\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right| \\ &= -K_p^2e^2 - \left(K_d^2 - M_RK_p\right)\dot{e}^2 - \left(\boldsymbol{w}^{*\mathrm{T}}\mathbf{P}\boldsymbol{h} - |\Pi(t)|\right)\left|\tau_{pd}\right| - \tilde{\boldsymbol{w}}^{\mathrm{T}}\mathbf{P}\boldsymbol{h}\left|\tau_{pd}\right|\end{aligned} \tag{53}$$

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

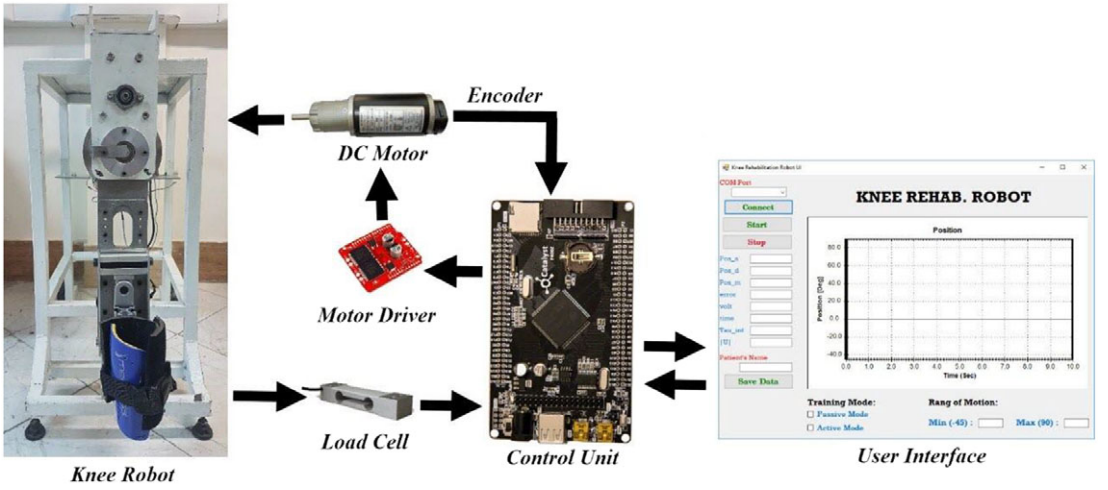


Figure 10. *Diagram of the various parts of the KRRS.*

$$\dot{V} \leq -\left(K_d^2 - M_R K_p\right)\dot{e}^2 - \left(\boldsymbol{w}^{*\mathrm{T}}\mathbf{P}\boldsymbol{h} - |\Pi\,(t)|\right)\left|\tau_{pd}\right| - \tilde{\boldsymbol{w}}^{\mathrm{T}}\mathbf{P}\,\boldsymbol{h}\,\left|\tau_{pd}\right| \tag{54}$$

Considering assumptions 3, 4, and 5 stated earlier, it can be seen that $\dot{V}$ is negative definite, and the system is stable.

5. Experiment results

To evaluate the designed controllers, several experiments and tests were performed in the passive mode, where the robot drives the patient's leg without consideration to any interaction torques, and in the active mode, where the undesirable interaction torques are taken into account. Figure 10 shows a diagram of all parts of the proposed KRRS. The controller program in the experimental environment is written in ARM-based STM32 board in C. The control loop sample time is 20 ms. The 100-pulse encoder is mounted at the motor shaft; the transmission system ratio is 1:90 (1:9 head gearbox, 1:10 cable-driven system); thus, the control resolution is 0.04 degree/pulse ($\frac{360}{\text{encoder pulses (100)}} \times$ transmission system ratio (1 : 90)). Table 2 shows the characteristics of the equipped load cells. Two 50 kg straight bar load cells are utilized in the system. The safety precautions have been implemented in the KRRS on three levels of mechanical, electrical, and software. Mechanical constraints are added to restrict the robot arm exceeding the range of motion of the knee joint. An emergency button immediately shuts the system down, once it is pressed by the observer or the patient. Joint limits conditions are also considered in the software design to keep the robot arm in the appropriate range of motion. Besides, the observer can stop the robot instantly from the UI.

In this section, the results of three experiments, passive with no load, passive loaded, and active loaded, are discussed. The IAE and the CE are defined as the performance indices to analyze the performance of the designed controllers. IAE and CE are given by:

$$IAE = \int |e|\, dt \tag{55}$$

$$CE = \int |\tau|\, dt \tag{56}$$

Table 3 shows the admittance parameters and the gains of the ASMC, PD, and PD-RBF. The KRRS parameters used are illustrated in Table 4.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Table II. *The load cell specifications.*

Specifications	Technique	Specifications	Technique
Capacity	50 kg	Rated output	2.0 ± 0.1mV/V
Excitation	12 V	Nonlinearity	0.015% F.S.[2]
Hysteresis	0.015% F.S.	Repeatability	0.02% F.S.

Table III. *The various controllers parameters.*

Admittance	$M_i = 0.18$, $C_i = 20$, $K_i = 15$
ASMC	$\beta_1 = 20$, $\beta_2 = 2.5$, $K_1 = 1.2$, $K_2 = 3.4$,$\kappa = 0.2$, $\varphi = 0.15$
PD	$K_p = 48$, $K_d = 5$
PD-RBF	$K_p = 48$, $K_d = 5$, $c = [-50 \quad -10 \quad 10 \quad 50]$, $b = 50$, $p_{11}, \ldots, p_{44} = 0.3$

Table IV. *KRRS parameters.*

Parameter	Value
m	3.0 kg
l	0.43 m
l_c	0.215 m
I_c	0.0462 kg.m^2

Remark: The admittance control parameters were experimentally tuned to have a smooth change in the trajectory. The gains of ASMC, PD, and RBF-PD controllers are also experimentally selected considering the IAE and CE indices, where the chosen gains of each controller are the ones that offer a low IAE for an appropriate CE value. It is worth to mention that various tests have conducted to choose the appropriate parameters of the mentioned controllers. For each controller, the best parameters have selected and the related results have reported.

The supply source used for the KRRS is DC voltage source; thus, using the motor dynamic, the obtained torque τ from the controllers is converted to voltage and applied to the DC motor by the control unit to the motor driver as PWM signals.

The motor equations are

$$\tau_m = \frac{\tau}{k_g} \tag{57}$$

$$i = \frac{\tau_m}{k_m} \tag{58}$$

$$\mathrm{U} = Ri + L\frac{di}{dt} + k_v\omega\sim \tag{59}$$

where τ_m is the motor torque, $k_g = 1:90$ is the transmission system ratio, k_m and k_v are the motor torque and velocity constants, respectively. i and U are the motor current and voltage, respectively. R is the motor resistance, L is the inductance, and ω is the motor speed. The physical parameters of the utilized DC motor in the KRRS are obtained using the system identification algorithms. Initially, a persistently exciting pseudo-random binary sequence is applied to the DC motor as input signal, and the shaft speed is measured. These input and output sequences are saved. Then, an appropriate structure of model is considered as the motor model. Finally, the system identification algorithms are utilized to estimate the parameters of the considered model using acquired input/output data [42, 43]. Several estimation processes are carried out to obtain the most accurate values of these parameters. Table 5 illustrates the estimated values of motor parameters.

[2]F.S. stands for full scale – the maximum rated output voltage.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Table V. *DC motor parameters.*

Parameter	**Value**
Resistance (R)	0.378 Ω
Inductance (L)	0.0025 H
Back-emf constant (K_v)	$0.055\ V \cdot s \cdot rad^{-1}$
Torque constant (K_m)	$0.07\ N \cdot m \cdot A^{-1}$

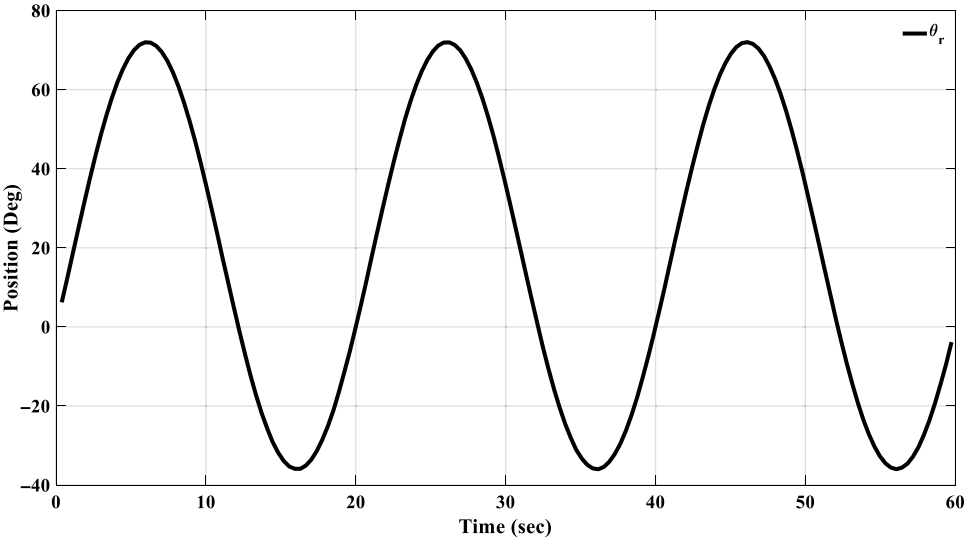


Figure 11. *The predefined trajectory.*

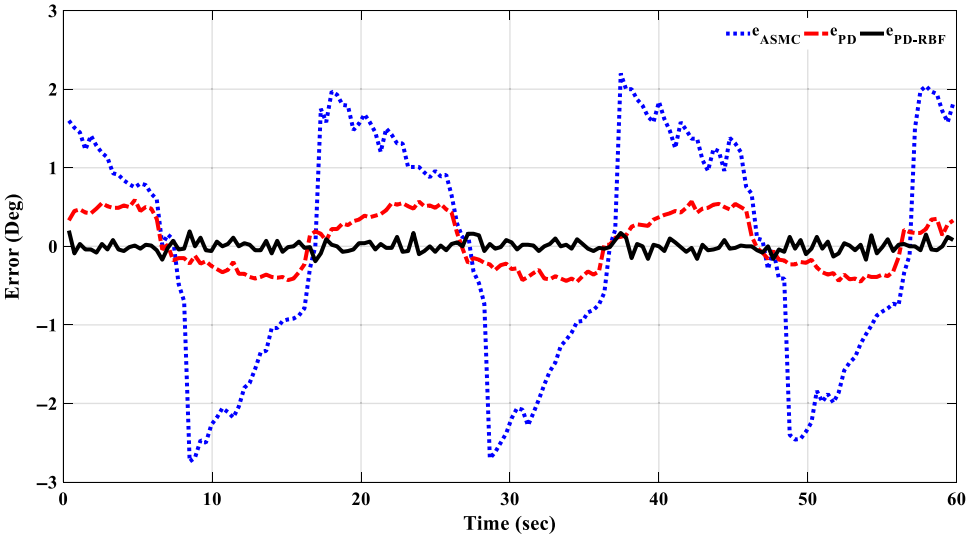


Figure 12. *Tracking error in the passive no-load mode.*

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press



Figure 13. *The robot with a connected leg.*

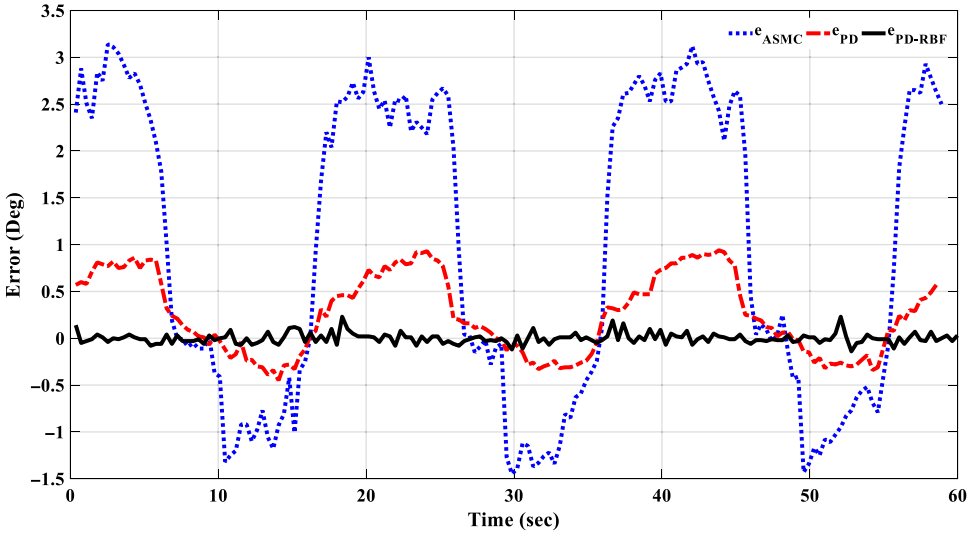


Figure 14. *Tracking error in the passive with-load mode.*

5.1. Passive mode

In this mode, the admittance control loop is disabled and the interaction torques between the KRRS and the leg are not considered. The robot follows the predefined trajectory without any changes. Figure 11 represents the predefined trajectory. First, the robot is tested, for each controller, without any load (the leg is not connected to the robot). Figure 12 shows the tracking error for the controllers in the passive no-load mode.

Then, the patient's leg is connected, and the KRRS moves it along the predefined trajectory. Figure 13 shows a leg connected to the robot. Figure 14 represents the tracking error for the controllers in the passive with-load mode.

Performance indices, IAE and CE, results of each controller in this mode are reported in Table 6.

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Table VI. *IAE and CE of the controllers in the passive mode.*

	Passive with no load		**Passive with load**	
	IAE	**CE**	**IAE**	**CE**
ASMC	79.72	143.2	94.5	180.2
PD	20.0	136.9	24.34	173.3
PD-RBF	3.43	134.5	2.60	171.7

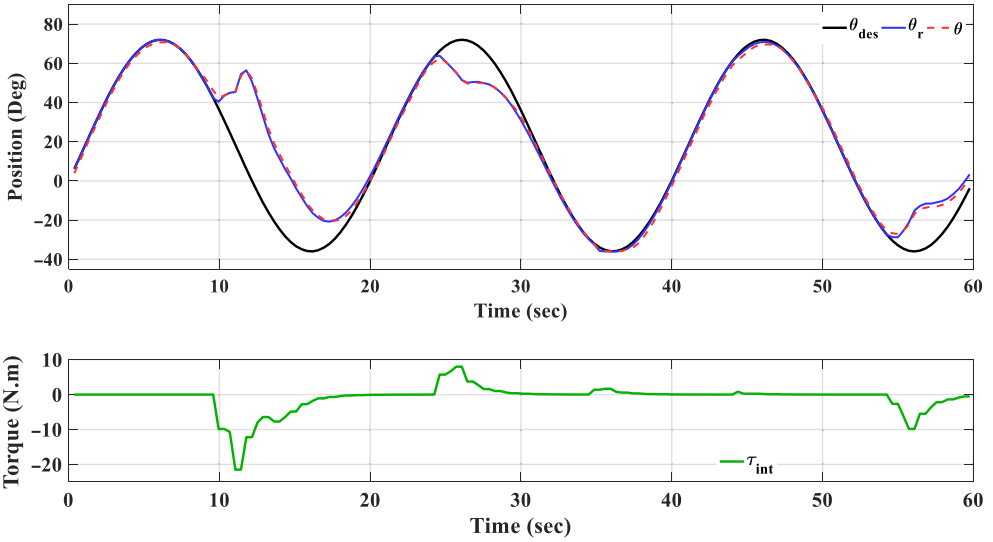


Figure 15. *Trajectory changes in the presence of the interaction torque τ_{int} and tracking performance of ASMC controller.*

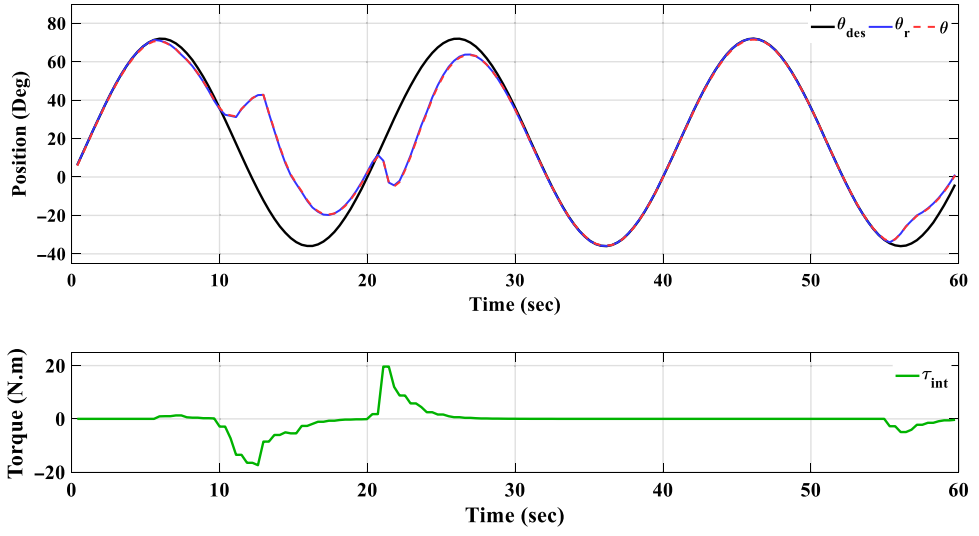


Figure 16. *Trajectory changes in the presence of the interaction torque τ_{int} and tracking performance of PD controller.*

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

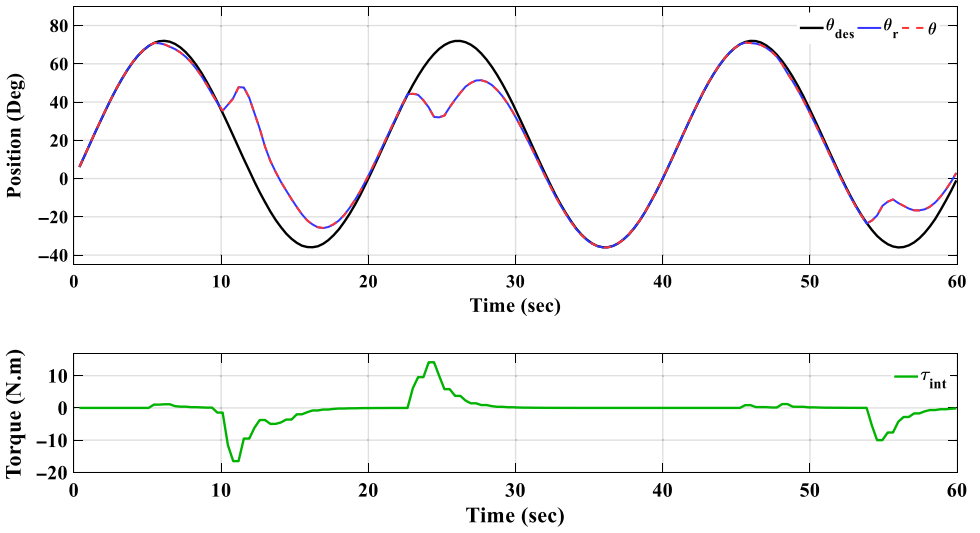


Figure 17. *Trajectory changes in the presence of the interaction torque τ_{int} and tracking performance of PD-RBF controller.*

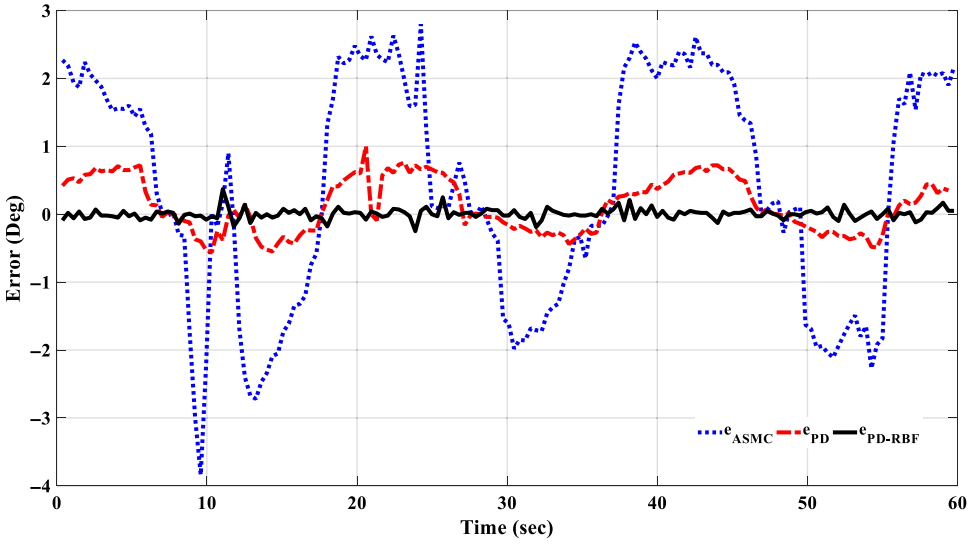


Figure 18. *Tracking error of the ASMC, PD, and PD-RBF controllers.*

5.2. Active mode

The admittance control loop is enabled in this mode, and the exerted interaction forces are measured by load cells and filtered to consider only forces of the sudden undesirable movements. Then these forces are fed back into the admittance equation, where the predefined trajectory is modified accordingly to reduce the effects of these interaction forces and ensure safe and smooth movements. Figures 15, 16, and 17 show the tracking performance of the ASMC, PD, and PD-RBF, respectively. These figures also represent how the trajectory is modified when interaction torque is applied. The tracking error of these controllers is illustrated in Fig. 18.

The CE of the robot in the active mode is subject to the interaction torques. Thus, to have a reasonable comparison between the controllers in their CE, the applied interaction torques on each of them should

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

Table VII. *The IAE and η of the controllers in the active mode.*

	IAE	***η***
ASMC	89.57	1.925
PD	21.35	1.90
PD-RBF	3.40	1.89

be similar. Since such a condition cannot be realized in practical experiment, a new ratio between the control output and the applied interaction torque is defined as follows:

$$\eta = \frac{\int |\tau|\, dt}{\int |\tau_{int}|\, dt} \tag{60}$$

η ratio represents the CE with respect to the interaction torques exerted by the patient.

Table 7 demonstrates the IAE and the ratio η for each controller in the active mode.

6. Results and discussion

In this section, the obtained results from experimental tests are analyzed, discussed, and compared. Graphical results indicate that all three controllers showed robustness in their performance, where there is no significant change in the performance for each controller in the three testing modes (passive no-load, passive with-load, and active with-load). Figures 15, 16, and 17 show that the modification in the trajectory path due to the applied interaction torques is smooth enough, with no sharp or tough changes, to provide safe movement and prohibit any injuries.

Figures 12, 14, and 18 show the tracking error of the three controllers in the passive no-load, passive with-load, and active modes, respectively. The performance of the ASMC is highly dependent on the dynamic model, which is difficult to be computed accurately and dramatically varies according to the patient's limb specifications. ASMC had a steady-state error ranging approximately between $(-3^\circ, 3^\circ)$ in the passive mode and $(-4^\circ, 3^\circ)$ in the active one. The PD performed better than the ASMC since it is a model-free control with a steady-state error less than 1° in the all modes. However, it suffers to compensate the high nonlinearity in the system. In the PD-RBF algorithm, the PD achieves the tracking task of the system, while the RBF is fed forward to compensate for the high nonlinearity terms. Thus, the controller is independent of the dynamic model of the system and able to compensate for the system's high nonlinearity. The RBF-PD has a tracking precision higher than both PD and ASMC with a steady-state error not exceeding a peak of 0.2°. According to Tables 6 and 7, the performance indices indicate that the proposed PD-RBF has the best performance compared to the others. These outcomes prove the superiority of the PD-RBF approach over the conventional PD and ASMC.

In ref. [5], a similar work is introduced. A PID controller is used in controlling 1-DOF mechanism for knee rehabilitation, and the results of that research show an error more than 2° in the passive and the active modes. In comparing the results of ref. [5] with the proposed admittance PD-RBF algorithm, the admittance PD-RBF shows more accurate performance.

7. Conclusion

In this paper, a KRRS is developed. An admittance control is introduced to reduce the effects of the excessive forces between the patient's leg and the robot. For trajectory tracking control, a PD-RBF approach is introduced, and its stability is analyzed. The performance of this developed approach, in the current paper, is compared with two linear and nonlinear control approaches. One of them is ASMC, which represents a nonlinear model-based control method, and the other one is conventional proportional-derivative control, which represents the model-free linear control technique. The ASMC

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

is dependent on the dynamic model of the robot. In practice, the dynamic model is difficult to obtain correctly. Furthermore, the chattering phenomenon is reduced at the expense of performance accuracy. Thus, the ASMC has a high steady-state error compared with the other controllers. PD is a model-free controller, easy to be tuned compared to ASMC, has a simple structure, and does not need complicated calculation that makes it faster to force the error to converge to zero. PD has a small steady-state error due to the complex nonlinearity terms of the system. In the proposed PD-RBF, the RBF acts as a compensator for the complex nonlinearity of the system, which improves the performance of the PD controller significantly. The controller parameters are tuned experimentally for the best performance based on the considered metrics. Future works can include an improvement of the controller for online tuning of the parameters; introduce the admittance with variable parameters to possibly achieve smoother modification of the trajectory; and extend the robotic system to train the ankle joint in addition to the knee.

Funding. The authors have not received support from any organization for the submitted work.

Conflicts of interest. The authors declare that they have no conflict of interest.

Code or data availability. Data and material available on request from the authors.

Authors' contributions. All authors contributed to all parts of the research. Design and analysis of the model were performed mainly by Bahram Tarvidizadeh and Alireza Hadi. Design and analysis of the control system were held mainly by Khalil Alipour and Karam Almaghout. All authors participated in the experiments. The first draft of the manuscript was written by Karam Almaghout, and all authors commented on the previous versions of the manuscript. All authors read and approved the final manuscript.

Ethics approval. This study was held without involving any patient and without using any patient's data. The experiments were held in the lab by the authors themselves to evaluate the efficiency of the design and the control of the system. Thus, ethics approval was not required for this study.

Consent to participate. The authors were the only participants in this study. They were all well aware of the nature of the research, and their participation in the experiments was to benefit the system evaluation.

Consent for publication. The authors affirm that the images in Figures 1(b) and 13 belong to one of the authors who consents to publish to the benefit of the research.

References

[1] W. Wang, Z.-G. Hou, L. Tong, F. Zhang, Y. Chen and M. Tan, "A novel leg orthosis for lower limb rehabilitation robots of the sitting/lying type," *Mech. Mach. Theory* **74**(5), 337–353 (2014).
[2] A. B. Farjadian, M. Nabian, C. Mavroidis and M. K. Holden, "Implementation of A Task-Dependent Anisotropic Impedance Controller Into A 2-DOF Platform-Based Ankle Rehabilitation Robot," **In:** *IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2015).
[3] H. Wang, Y. Feng, D. Zhang, F. Chen, Y. Du, H. Yan and L. Vladareanu, "Retracted: Research on a new applicable lower limb rehabilitation robot," *J. Fundamental Appl. Sci*. **10**(4S), 181–186 (2018).
[4] S. Hussain, S. Q. Xie and P. K. Jamwal, "Control of a robotic orthosis for gait rehabilitation," *Robot. Autonom. Syst*. **61**(9), 911–919 (2013).
[5] N. Naghavi and M. J. Mahjoob, "Design and control of an active 1-DoF mechanism for knee rehabilitation," *Disab. Rehabil. Assist. Technol*. **11**(7), 588–594 (2016).
[6] P. Paz, T. Q. Oliveira, A. V. Pino and A. P. Fontana, "Model-Free neuromuscular electrical stimulation by stochastic extremum seeking," *IEEE Trans Contr. Syst. Technol.* **28**, 238–253 (2019).
[7] T. Roux-Oliveira, L. R. Costa, A. V. Pino and P. Paz, "Extremum seeking-based adaptive PID control applied to neuromuscular electrical stimulation," *Anais da Academia Brasileira de Ciências* **91**(suppl. 1), e20180544 (2019).
[8] S. Han, H. Wang and Y. Tian, "A linear discrete-time extended state observer-based intelligent PD controller for a 12 DOFs lower limb exoskeleton LLE-RePA," *Mech. Syst. Signal Process.* **138**(3), 106547 (2020).
[9] F. Yousefi, K. Alipour, B. Tarvirdizadeh and A. Hadi, "Knee Rehabilitation Robot Control By Sliding-Backstepping and Admittance Control," **In:** *Artificial Intelligence and Robotics (IRANOPEN)* (IEEE, 2017).

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

[10] A. Riani, T. Madani, A. Benallegue and K. Djouani, "Adaptive integral terminal sliding mode control for upper-limb rehabilitation exoskeleton," *Control Eng. Pract.* **75**(2), 108–117 (2018).

[11] T. R. Oliveira, L. R. Costa, J. M. Y. Catunda, A. V. Pino, W. Barbosa and M. N. de Souza, "Time-scaling based sliding mode control for neuromuscular electrical stimulation under uncertain relative degrees," *Med. Eng. Phys.* **44**(9), 53–62 (2017).

[12] X. Li, F. Li, X. Zhang, C. Yang and W. Gui, "Exponential stability analysis for delayed semi-Markovian recurrent neural networks: A homogeneous polynomial approach," *IEEE Trans. Neur. Net. Learn.* **29**(12), 6374–6384 (2018).

[13] S. Beyhan and M. Alcı, "Stable modeling based control methods using a new RBF network," *ISA Trans.* **49**(4), 510–518 (2010).

[14] I. Poultangari, R. Shahnazi and M. Sheikhan, "RBF neural network based PI pitch controller for a class of 5-MW wind turbines using particle swarm optimization algorithm," *ISA Trans.* **51**(5), 641–648 (2012).

[15] P. Zhang, J. Zhang and Z. Zhang, "Design of RBFNN-based adaptive sliding mode control strategy for active rehabilitation robot," *IEEE Access* **8**, 155538–155547 (2020).

[16] J. Shi, L. Xu, G. Cheng, J. Xu, S. Chen and X. Liang, "Trajectory Tracking Control Based on RBF Neural Network of The Lower Limb Rehabilitation Robot," **In:** *2020 IEEE International Conference on Mechatronics and Automation (ICMA)* (IEEE, 2020).

[17] Y. Akiyama, Y. Yamada, K. Ito, S. Oda, S. Okamoto and S. Hara, "Test Method for Contact Safety Assessment of A Wearable Robot-Analysis of Load Caused By A Misalignment of the Knee Joint," **In:** *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication* (IEEE, 2012).

[18] Y. He, D. Eguren, T. P. Luu and J. L. Contreras-Vidal, "Risk management and regulations for lower limb medical exoskeletons: A review," *Medical Dev*. **10**, 89 (2017).

[19] R. Mallat, M. Khalil, G. Venture, V. Bonnet and S. Mohammed, "Human-exoskeleton Joint Misalignment: A Systematic Review," **In:** *Fifth International Conference on Advances in Biomedical Engineering (ICABME)* (IEEE, 2019).

[20] S. Bansil, N. Prakash, J. Kaye, S. Wrigley, C. Manata, C. Stevens-Haas and R. Kurlan, "Movement disorders after stroke in adults: A review," *Tremor Other Hyperkinetic Movements* **2**, tre-02-42-195-1 (2012).

[21] F. Alarcón, J. C. M. Zijlmans, G. Dueñas and N. Cevallos, "Post-stroke movement disorders: report of 56 patients," *J. Neurol. Neurosurg. Psych*. **75**(11), 1568–1574 (2004).

[22] J. Bessler, G. B. Prange-Lasonder, L Schaake, J. F. Saenz, C. Bidard, I. Fassi, M. Valori, A. B. Lassen and J. H. Buurke, "Safety assessment of rehabilitation robots: a review identifying safety skills and current knowledge gaps," *Front. Robot. AI* **8**, 33 (2021).

[23] A.-N. Sharkawy, P. N. Koustoumpardis and N. Aspragathos, "A recurrent neural network for variable admittance control in human-robot cooperation: simultaneously and online adjustment of the virtual damping and inertia parameters," *Int. J. Intell. Robot. Appl*. **4**(4), 441–464 (2020).

[24] W. Yu and A. Perrusquía, "Simplified stable admittance control using end-effector orientations," *Int. J. Soc. Robot.* **12**(5), 1061–1073 (2020).

[25] C. Ochoa Luna, M. H. Rahman, M. Saad, P. S. Archambault and S. B. Ferrer, "Admittance-based upper limb robotic active and active-assistive movements," *Int. J. Adv. Robot. Syst.* **12**(9), 117 (2015).

[26] A. Taherifar, G. Vossoughi and A. S. Ghafari, "Variable admittance control of the exoskeleton for gait rehabilitation based on a novel strength metric," *Robotica* **36**(3), 427–447 (2018).

[27] S. Gupta and S. Sharma, "Selection and application of advance control systems: PLC, DCS and PC-based system." *J. Sci. Ind. Res.* **64**, 249–255 (2005).

[28] E. Akdoğan, E. Taçgın and M. A. Adli, "Knee rehabilitation using an intelligent robotic system," *J. Intell. Manuf.* **20**(2), 195–202 (2009).

[29] D. Morin. *The Lagrangian Method, in Introduction to Classical Mechanics: with Problems and Solutions* (Cambridge University Press, 2008).

[30] P. Song, Y. Yu and X. Zhang, "A tutorial survey and comparison of impedance control on robotic manipulation," *Robotica* **37**(5), 801–836 (2019).

[31] H. Feng, C. Yin, W. Weng, W. Ma, J. Zhou, W. Jia and Z. Zhang, "Robotic excavator trajectory control using an improved GA based PID controller," *Mech. Syst. Signal Process.* **105**, 153–168 (2018).

[32] Y. Shtessel, C. Edwards, L. Fridman and A. Levant, *Sliding Mode Control and Observation*, vol. **10**. (Birkhäuser New York, New York, 2014).

[33] S. Roy, S. Baldi and L. M. Fridman, "On adaptive sliding mode control without a priori bounded uncertainty," *Automatica* **111**(12), 108650 (2020).

[34] M. Babaiasl, et al., "Sliding Mode Control of an Exoskeleton Robot for use in Upper-Limb Rehabilitation," **In:** *3rd RSI International Conference on Robotics and Mechatronics (ICROM)* (IEEE, 2015).

[35] S. Hussain, S. Q. Xie and P. K. Jamwal, "Adaptive impedance control of a robotic orthosis for gait rehabilitation," *IEEE Trans. Cybern*. **43**(3), 1025–1034 (2013).

[36] M. Torabi, M. Sharifi and G. Vossoughi, "Robust adaptive sliding mode admittance control of exoskeleton rehabilitation robots," *Sci. Iran* **25**(5), 2628–2642 (2018).

[37] I. Eker, "Sliding mode control with PID sliding surface and experimental application to an electromechanical plant," *ISA Trans.* **45**(1), 109–118 (2006).

[38] T. Kara and A. H. Mary, "Adaptive PD-SMC for nonlinear robotic manipulator tracking control," *Stud. Inform. Control* **26**(1), 49–58 (2017).

[39] A.-N. Sharkawy and P. N. Koustoumpardis, "Dynamics and computed-torque control of a 2-DOF manipulator: Mathematical analysis," *Int. J. Adv. Sci. Technol*. **28**(12), 201–212 (2019).

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press

[40] V.-T. Nguyen, C.-Y. Lin, S.-F. Su, and Q.-V. Tran, "Adaptive chattering free neural network based sliding mode control for trajectory tracking of redundant parallel manipulators," *Asian J. Control* **21**(2), 908–923 (2018).
[41] J. Li and H. Tian, "Position control of SMA actuator based on inverse empirical model and SMC-RBF compensation," *Mech. Syst. Signal Process.* **108**, 203–215 (2018).
[42] L. Ljung, *System Identification—Theory for the User, 2nd edn. (Upper Saddle River, NJ: PTR Prentice Hall)*. Feasible Mind Uploading, vol. **101** (1999).
[43] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models* (Springer Berlin, Heidelberg, 2001).

Cite this article: K. Almaghout, B. Tarvirdizadeh, K. Alipour and A. Hadi (2022). "RBF neural network-based admittance PD control for knee rehabilitation robot", Robotica **40**, 4512–4534. https://doi.org/10.1017/S0263574722001084

https://doi.org/10.1017/S0263574722001084 Published online by Cambridge University Press