mathematics

Article

# A Model-Free Control Scheme for Rehabilitation Robots: Integrating Real-Time Observations with a Deep Neural Network for Enhanced Control and Reliability

**Hajid Alsubaie [1,2] and Ahmed Alotaibi [1,2,*]**

[1] Department of Mechanical Engineering, College of Engineering, Taif University, Taif 21944, Saudi Arabia; h.alsubaie@tu.edu.sa

[2] King Salman Center for Disability Research, Riyadh 11614, Saudi Arabia

[*] Correspondence: am.alotaibi@tu.edu.sa

**Abstract:** Effective control of rehabilitation robots is of paramount importance and requires increased attention to achieve a fully reliable, automated system for practical applications. As the domain of robotic rehabilitation progresses rapidly, the imperative for precise and dependable control mechanisms grows. In this study, we present an innovative control scheme integrating state-of-the-art machine learning algorithms with traditional control techniques. Our approach offers enhanced adaptability to patient-specific needs while ensuring safety and effectiveness. We introduce a model-free feedback linearization control method underpinned by deep neural networks and online observation. While our controller is model-free, and system dynamics are learned during training phases, we employ an online observer to robustly estimate uncertainties that the systems may face in real-time, beyond their training. The proposed technique was tested through different simulations with varying initial conditions and step references, demonstrating the controller's robustness and adaptability. These simulations, combined with Lyapunov's stability verification, validate the efficacy of our proposed scheme in effectively controlling the system under diverse conditions.

**Keywords:** rehabilitation robots; machine learning integration; model-free control; patient-specific adaptability; robotic rehabilitation; online uncertainty estimation

**MSC:** 93C85; 68T05; 68T45; 93B60; 92C20; 68U10

check for updates

**Citation:** Alsubaie, H.; Alotaibi, A. A Model-Free Control Scheme for Rehabilitation Robots: Integrating Real-Time Observations with a Deep Neural Network for Enhanced Control and Reliability. *Mathematics* **2023**, *11*, 4791. https://doi.org/10.3390/math11234791

Academic Editors: Fangzheng Gao and Zhongcai Zhang

Received: 22 October 2023
Revised: 18 November 2023
Accepted: 23 November 2023
Published: 27 November 2023

## 1. Introduction

Rehabilitation robots have established a distinct position in the vast landscape of medical technology. As technological advancements continue to revolutionize healthcare, these robots emerge as a beacon of hope for patients striving to regain functions lost due to traumatic injuries or debilitating conditions [1,2]. Beyond the obvious benefits of delivering consistent, precise therapeutic motions, these robots redefine rehabilitative care by transcending traditional boundaries. Their versatility lies not just in the diverse therapeutic regimens they can deliver but also in their adaptability to function across different settings—from sophisticated medical facilities to the comfort of patients' homes [3,4]. This ability to decentralize rehabilitative care ensures that more individuals can access quality therapy, regardless of geographical or logistical barriers. But the potential of these robots is not just in their mechanics or software; it is profoundly intertwined with their control mechanisms [5,6]. The need for exact, reliable control becomes even more paramount when we consider the intimate interactions these robots have with patients. Every movement, no matter how minuscule, must be executed with impeccable precision, underscoring the essence of reliability in their control systems.

In the realm of nonlinear systems control, traditional methodologies span from conventional proportional–integral–derivative (PID) schemes to advanced controllers such as fuzzy logic [7–13] and optimal control [14–19] strategies. Despite their proven utility, these controllers fundamentally rely on precise system models and often falter in the face of intricate, unpredictable disturbances and systemic uncertainties. Acknowledging these limitations, recent scholarly efforts have pivoted towards the conceptualization of state observers. Among these, finite-time observers have emerged prominently, distinguishing themselves by guaranteeing state convergence within a specified finite time as opposed to the asymptotic approach characterized by exponential convergence. This paradigm shift towards finite-time convergence is of considerable interest within control engineering, providing a robust framework that copes more adeptly with the exigencies of real-time system dynamics and the attendant uncertainties [20]. This approach not only offers a more deterministic outlook on state estimation timelines but also enhances the resilience of control systems against unforeseen disturbances, marking a significant advance over traditional control mechanisms. The promise of finite-time controllers extends to their potential integration with modern modeling techniques, paving the way for more intelligent, adaptive control structures that can outperform classical methods, ensuring more reliable and effective system regulation.

Robotics control, particularly in dynamic and unpredictable environments like rehabilitation, continually faces a plethora of challenges [3,21]. While traditional control mechanisms, honed by decades of research and practical use, struggle with the intricacies of modern robotic systems [22], feedback control has been a pivotal element in systems engineering, undergoing substantial evolution. Originally designed to ensure system stability and performance amid disturbances, feedback control has progressively incorporated advanced computational methods. This transition, from fundamental controllers to more adaptive and robust control strategies, addresses the complexities of contemporary systems. In recent developments, feedback control now includes sophisticated methodologies such as Output Feedback Self-Tuning Controllers [23]. These cutting-edge controllers adapt their parameters dynamically, learning from the system's output to incessantly refine control laws. Such self-tuning abilities represent the significant advancement in feedback control technologies, allowing systems to perform efficiently in uncertain scenarios and mirroring the current trend towards more automated and intelligent systems. Nonetheless, these methods still have potential for further enhancement, particularly with the advent of machine learning approaches that have revolutionized nearly all fields, including dynamics modeling and control [24,25].

Control engineering is increasingly recognizing deep learning as a significant area of potential. Originating from advanced algorithms adept at extracting insights from extensive and complex datasets, deep learning introduces an unprecedented level of adaptability [26,27]. Envision a control system that not only evolves, learns, and adapts over prolonged design cycles but also operates in real time, promptly responding to immediate environmental stimuli [28,29]. Deep learning, with its neural networks and layered architectures, can potentially dissect and predict system behaviors with a granularity that could revolutionize adaptive control strategies [30,31]. This potential convergence of deep learning with control mechanisms could very well herald the next big leap in controlling complex, uncertain systems [32]. However, deep learning, like many technological innovations, is not without its shortcomings [33,34]. One notable constraint is its limited prowess in extrapolation, particularly in unfamiliar domains [35]. In the field of rehabilitation robotics, where safety is of utmost importance, unpredictability in system behavior is unacceptable. Solely depending on deep learning without robust control mechanisms may result in ineffective or, in extreme situations, hazardous outcomes. This necessitates the adoption of integrated or hybrid control systems, especially in applications where the stakes are high and the implications of failure are significant. Such strategies are essential to ensure the reliability and safety of robotic systems interacting closely with humans.

In the domain of rehabilitation robotics, the intricate interplay of human biology, robotics, and control systems creates a uniquely complex landscape. These challenges underscore the need for control systems that are not only adaptable to the dynamic nature of human physiology but also exhibit precision in execution to ensure safety and efficacy [1,36]. Additionally, as technology marches forward, we envision a future where these robots, already complex, take on even more multifaceted tasks. Such advancements amplify the need for flawless operation, ensuring patient safety and therapeutic efficacy [37,38]. Furthermore, as rehabilitation robots move towards becoming an integral part of mainstream healthcare, their broad acceptance is contingent upon delivering consistent and reliable performance. These factors highlight the urgent need for dedicated research. A sole reliance on deep learning or traditional controls is insufficient. A blended approach, combining the strengths of both, is essential.

Addressing the industry's challenges and requirements, our research proposes an innovative control methodology. We envision a system in which advanced machine learning algorithms are harmoniously integrated with time-tested control methodologies. The cornerstone of our approach is a model-free feedback linearization technique, which is significantly enhanced by the application of deep neural networks. This integration not only fortifies the control system's adaptability but also ensures a high level of precision and reliability in its operation. Acknowledging the constraints of deep learning, our proposed technology ventures further. In real-time extrapolative contexts, we integrate a robust online observer. This component continuously estimates and adapts to uncertainties, even those beyond the training purview, ensuring our controller remains steadfast and adaptable. Such a design provides a delicate balance—leveraging the predictive might of machine learning while ensuring real-time reliability and adaptability, crucial for the dynamic world of rehabilitation robots.

## 2. Dynamic of the System

Consider a two-degree-of-freedom (2-DOF) knee rehabilitation robotic system as illustrated in Figure 1. This design captures the essential elements for simulating the biomechanical dynamics inherent to knee rehabilitation exercises. The system comprises two primary links, each characterized by its unique degree of freedom, denoted as $\psi_1$ for the upper link and $\psi_2$ for the lower link. We denote $\psi$ as the vector [$\psi_1$, $\psi_2$], representing the system's generalized coordinates. The respective masses and lengths associated with these links are labeled as $m_1$, $l_1$, and $m_2, l_2$. Springs positioned at the joints suggest an element of elasticity, perhaps designed to emulate the natural compliance found within a human knee. The coordinate points $x_1, y_1$ and $x_2, y_2$ are indicative of specific locations or reference points on the link.

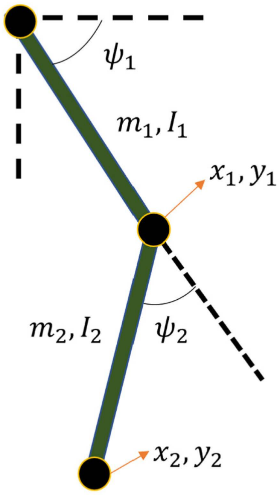

**Figure 1.** Schematic representation of a 2-DOF knee rehabilitation robot, showcasing the primary links, degrees of freedom, and associated parameters. Springs at joints indicate compliance, while labeled points suggest reference coordinates for kinematic analysis.

It is important to note that the 2-DOF knee rehabilitation robotic system is meticulously designed for lower limb rehabilitation, specifically targeting the knee joint. The upper link of the robot, referred to as $\psi_1$, is tailored to mimic the human femur, while the lower link, $\psi_2$, aligns with the tibia. These robotic links are purposefully actuated to replicate the natural movements of the human leg during rehabilitation exercises.

A distinctive feature of our system is the incorporation of springs at the joints. These springs introduce controlled compliance, closely resembling the physiological properties of human muscles and ligaments. This design choice ensures a safe and realistic interaction between the robot and the user, facilitating a responsive rehabilitation experience capable of adapting to the varying levels of stiffness and strength exhibited by recovering patients. Table 1 enumerates and provides descriptions of the parameters and variables of the system.

**Table 1.** Parameters and variables of the systems.

| Parameter | Description | Value | Unit |
|---|---|---|---|
| $\psi_1$ | Degree of freedom for upper link | - | rad |
| $\psi_2$ | Degree of freedom for lower link | - | rad |
| $m_1$ | Mass of the upper link | 2 | kg |
| $I_1$ | Moment of the upper link | 0.0612 | Meters (m) |
| $l_1$ | Length of the upper link | 0.35 | kg·m$^2$ |
| $m_2$ | Mass of the lower link | 0.85 | kg |
| $l_2$ | Length of the lower link | 0.31 | Meters (m) |
| $I_2$ | Moment of inertia of lower link | 0.0204 | kg·m$^2$ |
| $x_1$ | $x$-coordinate of upper link | - | Meters (m) |
| $y_1$ | $y$-coordinate of upper link | - | Meters (m) |
| $x_2$ | $x$-coordinate of lower link | - | Meters (m) |
| $y_2$ | $y$-coordinate of lower link | - | Meters (m) |

The practical applications of this design extend beyond basic rehabilitation exercises to encompass post-operative therapy, injury recovery, and strength conditioning. Our system is designed to provide a versatile solution that not only aids in restoring functional mobility but also quantifies patient progress through the collection of accurate kinematic data. As part of our ongoing work, we plan to conduct clinical trials to validate the system's effectiveness in real-world rehabilitation settings. This step will further bridge the gap between theoretical research and tangible medical benefits, addressing the concerns raised about the interaction of our system with the human body.

### *2.1. Model of the System*

The kinetic energy of the 2-DOF knee rehabilitation robot can be formulated mathematically as follows:

$$T = \frac{1}{2}m_1 l_{c1}^2 \dot{\psi}_1^2 + \frac{1}{2}I_1 \dot{\psi}_1^2 + \frac{1}{2}m_2 l_1^2 \dot{\psi}_1^2 + m_2 l_1 l_{c2} \dot{\psi}_1 \dot{\psi}_c \cos(\psi_2 \Big) + \frac{1}{2}m_2 l_{c2}^2 \dot{\psi}_c{}^2 + \frac{1}{2}I_2 \dot{\psi}_c^2 \tag{1}$$

where $\psi_1$ and $\psi_2$ stand for the generalized coordinates for the system depicted in Figure 1. Also, in our formulations, $\psi_c = \psi_1 + \psi_2$. The distance from joint $i - 1$ to the center of mass of link $i$ is represented by $lc_i$, where $i$ can assume values of 1 or 2. The masses of links 1 and 2 are denoted as $m_1$ and $m_2$, respectively. Furthermore, $I_i$ signifies the moment of inertia of link $i$ about an axis orthogonal to the plane of illustration and situated at its center of mass. The system's potential energy can be articulated mathematically as follows:

$$V_p = m_1 g l_{c2} \sin(\psi_1) + m_2 g[l_1 \sin(\psi_1) + l_{c2} \sin(\psi_c)]. \tag{2}$$

Utilizing Lagrange's equation, the dynamic equations governing the system can be derived as:

$$I\ddot{\psi} + C\dot{\psi} + K + J^T f = q(t) \tag{3}$$

where the inertia matrix, $I(\psi)$, stands as a symmetric and positive definite matrix of size $n \times n$. Furthermore, $C\left(\psi, \dot{\psi}\right)$ encompasses the collective influence of Coriolis and centripetal forces. The gravitational forces acting on the system are depicted by $K(\psi)$. The Jacobian matrix is denoted by $J^T(\psi)$ and is presumed to be nonsingular. The force vector, governed by constraints, is symbolized by $f(t)$ and the input torque, represented by $q(t)$. Utilizing Lagrange's equation, the components constituting this motion equation (delineated in Equation (3)) are detailed as:

$$\begin{gathered}
I = \begin{bmatrix} m_1 l_{c1}^2 + m_2\left(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos\psi_2\right) + I_1 + I_2 & m_2\left(l_{c2}^2 + 2l_1 l_{c2}\cos\psi_2\right) + I_2 \\ m_2\left(l_{c2}^2 + 2l_1 l_{c2}\cos\psi_2\right) + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix} \\
C = \begin{bmatrix} -m_2 l_1 l_{c2}\dot{\psi}_2 + \sin(\psi_2) & -m_2 l_1 l_{c2}\dot{\psi}_c \sin\psi_2 \\ m_2 l_1 l_2 \dot{\psi}_1 \sin\psi_2 & 0 \end{bmatrix} \\
K = \begin{bmatrix} (m_1 l_{c2} + m_2 l_1) g\cos\psi_1 + m_2 l_{c2} g\cos(\psi_c) \\ m_2\left(l_{c2}^2 + l_1 l_{c2}\cos\psi_2\right) + I_2 \end{bmatrix}
\end{gathered} \tag{4}$$

The force vector and Jacobian matrix are described as follows:

$$\begin{gathered}
f = \begin{bmatrix} l_1\cos\psi_1 + l_2\cos(\psi_c) \\ l_1\sin\psi_1 + l_2\sin(\psi_c) \end{bmatrix} \\
J = \begin{bmatrix} -l_1\sin\psi_1 + l_2\sin(\psi_c) & -l_2\sin(\psi_c) \\ -l_1\cos\psi_1 + l_2\cos(\psi_c) & l_2\cos(\psi_c) \end{bmatrix}
\end{gathered} \tag{5}$$

To recast the motion equation for the 2DOF multi-input multi-output rehabilitation robot into a state-space format, we set $z_1 = [\psi_1, \psi_2]^T$ and $z_2 = \left[\dot{\psi}_1, \dot{\psi}_2\right]^T$. Accordingly, the dynamic behavior of the robot is articulated as:

$$\begin{gathered}
\dot{z}_1 = z_2, \\
\dot{z}_2 = p + I^{-1} q(t)
\end{gathered} \tag{6}$$

From which, drawing upon Equations (3)–(5), the following can be derived:

$$p = I^{-1}\left(-C z_2 - K - J^T f\right) \tag{7}$$

The presented equation delineates the state space of the system, wherein each component may be subject to uncertainty. This is particularly true given that the robot interacts with patients, and the forces and uncertainties can vary considerably between individuals. Relying solely on this dynamic motion for controller design in rehabilitation robots may not be practical for real-world applications. At the very least, the results might be suboptimal due to the inherent uncertainties. In the subsequent section, we introduce a deep neural network to model the system, rendering our controller design independent of the system's model. The deep neural network's capacity for function approximation allows the controller to adeptly learn the system's nonlinear dynamics.

### 2.2. Proposed Neural State-Space Model

Neural state-space models fall under a class of models that utilize neural networks to capture the functions characterizing a system's nonlinear state-space description [24,39–45]. In classical control theory, these models serve to elucidate the behavior of dynamic systems, highlighting the interplay between the system's inputs, outputs, and intrinsic states. Consider a typical state-space format characterized by the subsequent mathematical expression. The system's mathematical configuration is delineated as:

$$\begin{gathered}\dot{\hat{z}}_1(t) = \hat{z}_2 \\ \dot{\hat{z}}_1(t) = p_{dnn} + h_{dnn}q(t)\end{gathered} \tag{8}$$

where $p_{dnn}$ and $h_{dnn}$ are static non-linear mappings, which are dynamically modeled using a deep neural network as opposed to the model derived in the prior section. In this context, our deep neural networks provide a robust mechanism to depict and comprehend nonlinear associations, rendering them particularly apt for shaping state-space models of analogous systems. We introduce a neural state-space model where the state equation is underscored by neural networks. This network encapsulates the function delineating the system's state behaviors. While we employ deep forward neural networks in this instance, one could also adopt recurrent neural structures, such as LSTM or GRU, or alternative architectural designs, contingent on the unique attributes of the system under consideration.

In the neural state-space model, the deep neural network is calibrated using system data. This calibration process refines the network parameters to reduce the variance between the model's forecasts and the system's actual behavior. Several methods, including gradient descent and backpropagation, are applicable to this optimization. After the training phase, the neural state-space model serves to emulate the system's behavior, deduce its inherent states from provided inputs and outputs, and anticipate forthcoming system reactions.

## 3. Proposed Control Technique

In the control of nonlinear systems, there is a shift from classic controllers like PID towards advanced observers for better handling of disturbances and uncertainties. Finite-time observers, in particular, are gaining prominence for their ability to guarantee convergence within a fixed period, offering a more reliable approach for real-time applications. This innovative direction suggests the possibility of integrating finite-time control strategies with modern modeling techniques, potentially leading to smarter and more resilient control solutions in engineering.

Here we take advantage of both the finite-time observer and deep forward neural networks. The controller we propose capitalizes on a pre-trained deep neural network as the system's state space. However, no deep neural network is flawless, and inevitably there will be uncertainties and unanticipated scenarios that surpass the bounds of the training process. To address this, a finite-time estimator is integrated to discern unknown disturbances and uncertainties. Given the limitations of the deep neural network in extrapolation, it might fall short in estimating these within the designed finite-time control framework. Regardless of these intricate uncertainties and disturbances, the intended finite-time control should guide the states of the rehabilitation robot towards the desired position. Consequently, we consider the dynamic of the systems in a realistic form as follows:

$$\begin{gathered}\dot{z}_1(t) = z_2 \\ \dot{z}_1(t) = p_{dnn} + h_{dnn}q(t) + D\end{gathered} \tag{9}$$

in which $D(t, z_1, z_2)$ denotes all uncertain terms that the deep neural network is not able to accurately model. Now, we propose the robust and finite-time approach as follows:

$$q(t) = \frac{1}{h_{dnn}}(p_{dnn} - g(y) + k_1 e_1 + k_2 e_2 + u_{d1}) \tag{10}$$

where $y$ and $g(y)$ respectively denote the desired path and its time derivative. $e_1$ and $e_2$ denote the error vector and its derivate, respectively, and they are given by $e_1 = z_1 - y$ and $e_2 = z_2 - g(y)$. Our proposed technique combines a neural state-space model determined by a deep neural network, represented by $p_{dnn}$ and $h_{dnn}$. This is complemented by a feedback linearization strategy, characterized by the terms $k_1e_1 + k_2e_2$, and a finite-time accurate observer denoted by term $u_{a_1}$. We propose the disturbance compensator $u_{a_1}$ as follows:

$$\dot{u}_{d0} = j_0 + p_{dnn} + h_{dnn}q(t) - g(y) \tag{11}$$

$$j_0 = -\alpha_1 L^{\frac{1}{3}} |u_{d0} - e|^{\frac{2}{3}} \mathrm{sign}(u_{d0} - e) + u_{d1} \tag{12}$$

$$\dot{u}_{d1} = j_1 \tag{13}$$

$$j_1 = -\alpha_2 L^{\frac{1}{2}} |u_{d1} - j_0|^{\frac{1}{2}} \mathrm{sign}(u_{d1} - j_0) + u_{d2} \tag{14}$$

$$\dot{u}_{d2} \ = -\alpha_3 L |u_{d2} - j_1| \mathrm{sign}(u_{d2} - j_1) \tag{15}$$

in which $\alpha_1, \alpha_2\alpha_3$ are positive design parameters and $j_0$, $j_2$, and $j_3$ are auxiliary variables used for designing the proposed observer.

**Theorem 1.** *Using the suggested robust intelligent control approach, system (9) consistently adheres to the desired state, even when confronted with unforeseen uncertainties and perturbations in the training of the neural state space of the system.*

**Remark 1.** *The robust online observer is a critical component in the presented control scheme for rehabilitation robots, as it addresses a significant challenge in the realm of automated systems: the ability to manage uncertainties and variations in real-time. This observer functions as a dynamic element within the control system, continuously analyzing and adjusting to the discrepancies between the predicted and actual performance of the robot. Its integration is particularly vital because, despite the model-free nature of the control method and the initial learning of system dynamics, there remains an inherent unpredictability in real-world scenarios. The use of this observer allows the system to not just rely on pre-learned data but to actively adapt to new situations, ensuring both the safety and effectiveness of the robotic rehabilitation process. By continually estimating and compensating for unforeseen variations and uncertainties, the robust online observer significantly enhances the reliability of the control system, a crucial aspect in medical applications where patient-specific needs and safety are paramount.*

**Proof.** In the initial step, we demonstrate that the estimator precisely approximates the composite nonlinearity, $D$. To initiate, we introduce the auxiliary error variables in the following manner:

$$e_{u_{d0}} = u_{d0} - e_1,\ e_{u_{d1}} = u_{d1} - D,\ e_{u_{d2}} = u_{d2} - \dot{D} \tag{16}$$

Taking into account Equations (9)–(15), one can derive the following:

$$\dot{e}_{u_{d0}} = -\alpha_1 L^{\frac{1}{3}} |u_{d0} - e|^{\frac{2}{3}} \mathrm{sign}(u_{d0} - e) + u_{d1} - D = -\alpha_1 L^{\frac{1}{3}} |u_{d0} - e|^{\frac{2}{3}} \mathrm{sign}(e_{u_{d0}}) + e_{u_{d1}} \tag{17}$$

$$\dot{e}_{u_{d1}} = -\alpha_2 L^{\frac{1}{2}} |u_{d1} - j_0|^{\frac{1}{2}} \mathrm{sign}(u_{d1} - j_0) + u_{d2} - \dot{D} = -\alpha_2 L^{\frac{1}{3}} \left|e_{u_{d1}} - \dot{e}_{u_{d0}}\right|^{\frac{1}{2}} \mathrm{sign}(e_{u_{d1}} - \dot{e}_{u_{d0}}) + e_{u_{d2}} \tag{18}$$

$$\dot{e}_{u_{d2}} = -\alpha_3 L |u_{d2} - \xi 1| \mathrm{sign}(u_{d2} - j_1) - \ddot{D} = -\alpha_3 L \left|e_{u_{d2}} - \dot{e}_{u_{d1}}\right| \mathrm{sign}\left(e_{u_{d2}} - \dot{e}_{u_{d1}}\right) - \ddot{D} \tag{19}$$

where Equation (19) is equivalent to

$$\dot{e}_{u_{d2}} \in= -\alpha_3 L \left|e_{u_{d2}} - \dot{e}_{u_{d1}}\right| \mathrm{sign}\left(e_{u_{d2}} - \dot{e}_{u_{d1}}\right) + [-L_D, L_D] \tag{20}$$

in which $[-L_D, L_D]$ denotes the bound for the unknown term $\ddot{D}$. As established in Lemma 2 of reference [46], the estimation errors $e_{u_{d0}}, e_{u_{d1}}$ and $e_{u_{d2}}$ converge to zero within a finite time. Consequently, after a short time range $T_o$, the following equations are valid:

$$e_{u_{d0}}(t) = e, u_{d1}(t) = D, u_{d2} = \dot{D} \tag{21}$$

Up to this point, we have demonstrated that the estimation is accurate, and the proposed observer compensates for inaccuracies in deep neural network estimation. Now, to show the stability of whole closed-loop systems, we introduce the following Lyapunov function

$$V_1 = \frac{1}{2}k_1e_1^2 + \frac{1}{2}e_2^2 \tag{22}$$

The time derivative of the Lyapunov function as presented in Equation (22) can be expressed as:

$$\dot{V}_1 = k_1e_1e_2 + e_2\dot{e}_2 = k_1e_1e_2 + e_2(p_{dnn} + h_{dnn}q(t) + D - g(y)) \tag{23}$$

Substituting the proposed control input into Equation (23), we obtain:

$$\dot{V}_1 = k_1e_1e_2 + e_2(p_{dnn} - (p_{dnn} - g(y) + k_1e_1 + k_2e_2 + k_3e_d + u_{a_1}) + D - g(y)) \tag{24}$$

In Equation (21), we demonstrated $u_{a_1} = D$; consequently, with the results in Equation (24) it is directly deduced that

$$\dot{V}_1 = -k_2e_2^2 \tag{25}$$

This derivative being a negative definite signifies the stability of the closed-loop system. As a result, the closed-loop system naturally gravitates towards the desired trajectory, ensuring that the error dynamics eventually reduce to zero. Importantly, this stability is maintained regardless of the precision of the neural state-space estimator or the chosen initial conditions. □

Figure 2 depicts the suggested control strategy. This approach integrates robust control as proposed in Equation (10), facilitating the accommodation of disturbances within the model. Such a configuration guarantees that the controller remains adaptive and robust, making it ideal for managing rehabilitation robots.

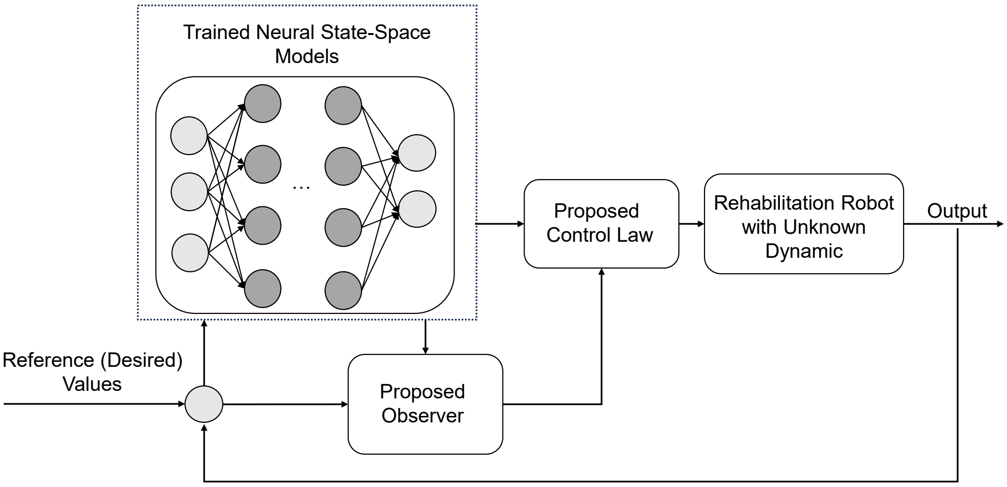
Trained Neural State-Space Models
...
Proposed Control Law
Rehabilitation Robot with Unknown Dynamic
Output
Reference (Desired) Values
Proposed Observer

**Figure 2.** Schematic representation of the proposed model-free neural state-space controller for rehabilitation robots, integrating a trained neural state-space model with a proposed observer and control law to drive the robot with unknown dynamics towards desired reference values.

## 4. Numerical Simulations

In this section, we assess the efficacy of the proposed approach using various examples. Specifically, in Examples 1 and 2, we showcase the capability of the proposed technique in managing the unknown intricate dynamics of the system under two distinct initial conditions. The values of the system's parameters are defined according to the values listed in Table 1. In Example 3, we leverage data derived from the algorithm to achieve control objectives that change over time.

### *4.1. Training Neural State-Space Model*

For data generation, we used step inputs of varying amplitudes (with the amplitude altered randomly) to excite the system, subsequently measuring and logging the position and velocity of both links in the 2-DOF robot. These data were sourced by simulating the system model as described in Equation (6).

In our research, we employed a two-layer multi-layer perceptron (MLP) neural network, each layer containing 128 nodes. We selected the hyperbolic tangent (tanh) activation function for its effectiveness in normalizing outputs. The network's initial weights were set using the Glorot initializer, and biases were initialized to zeros, providing a stable starting point for training. The model was trained over 2000 epochs using the Adam optimizer, a choice that ensures efficient and adaptive learning. This setup strikes a balance between learning capacity and computational efficiency, tailored to the demands of our state estimation task.

The accumulated training data were then utilized to train the neural network in an offline mode. For this training, 100 time histories of states of the system and their corresponding derivatives were utilized. A representative sample from these 100 samples, as employed in the training phase, is illustrated in Figure 3. Although the dataset may appear modest in size, it is sufficient for several reasons:

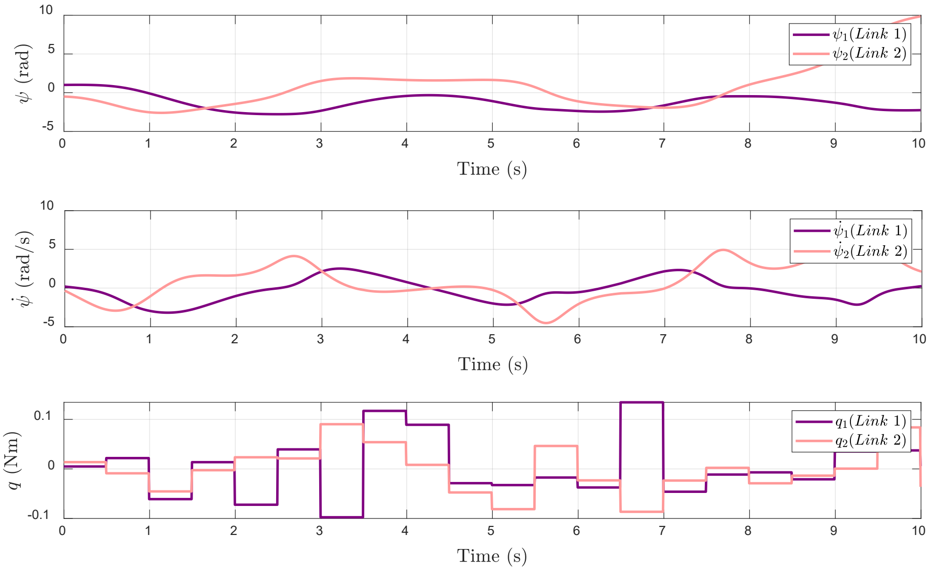

**Figure 3.** Temporal evolution of the training data for the state-space neural network, illustrating the behavior of link 1 and link 2.

- The diversity of the data: The time histories were generated using varying amplitudes of step inputs, which provided a rich set of dynamic responses from the system, encompassing a wide range of possible states.
- Suitability for a shallow network: The neural network employed is relatively shallow; such networks can effectively model complex relationships without the need for large volumes of data that might be required for deeper networks.

- Convergence of the loss function: The loss function, as shown in Figure 4 of our paper, exhibits a strong convergence pattern. After a significant initial decrease, it stabilizes with minor fluctuations, indicating that the network has successfully captured the underlying system dynamics.

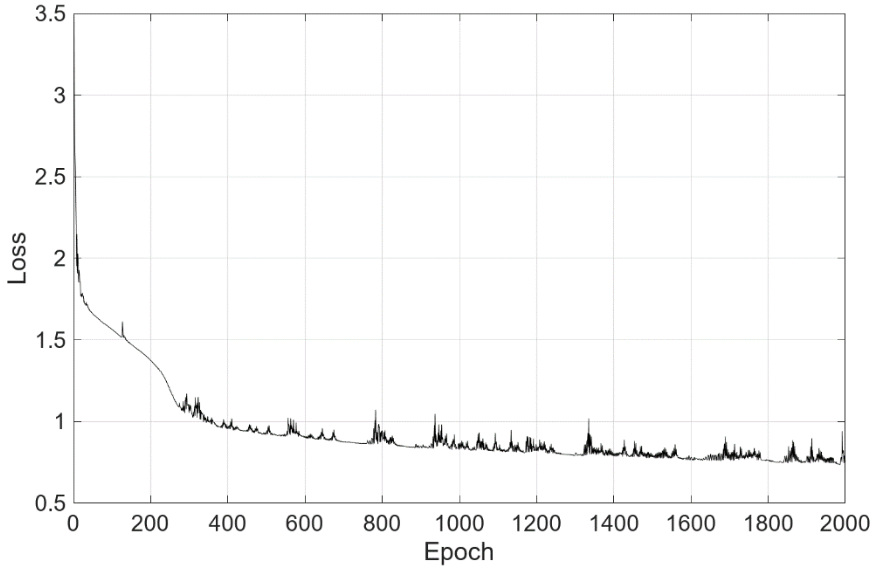

**Figure 4.** Convergence behavior of the loss function for the neural state-space model over 2000 epochs, highlighting the rapid initial decrease followed by stabilization with minor fluctuations.

Figure 4 depicts the loss function evolution during the neural network training for representing the system's neural state-space model. This loss function offers a perspective into the neural network's optimization history. Observing this function allows us to gauge the evolution and convergence of the training, confirming that the neural network effectively represents the core dynamics of the system.

### *4.2. Example 1*

Figure 5 showcases the control outcomes for link 1 and link 2 of the rehabilitation robot, utilizing the proposed control methodology with initial system states defined as $\left[\psi_1(0), \psi_2(0), \dot{\psi}_1(0), \dot{\psi}_2(0)\right] = [1, 0.5, 0, 0]$. The visuals effectively highlight the notable proficiency of the recommended robust adaptive controller. Also, this figure illustrates the controller's adeptness, integrating a neural state-space model combined with a finite-time observer and feedback linearization term, in tackling the control challenges of a completely unknown system. It is noteworthy that no information from the system's model is employed, underscoring the method's relevance for real-world applications. This intrinsic capacity to navigate uncertainties is paramount when controlling rehabilitation robots.

### *4.3. Example 2*

To further gauge the efficacy of our proposed controller, we conducted an auxiliary test by altering the initial state values of the system. We present the system's time history and control input when the ignition velocities of the robot links deviate from zero, differing from the previous example. The chosen initial states for this test are $\left[\psi_1(0), \psi_2(0), \dot{\psi}_1(0), \dot{\psi}_2(0)\right] = [1, -0.5, 0.2, -0.3]$.

Figure 6 provides insight into the controller's capability to ensure system stability under these varied conditions. Moreover, the controller consistently operates within a permissible spectrum, guaranteeing viable control signal magnitudes. Remarkably, even when faced with wholly unknown system dynamics, the proposed controller showcases the excellent performance, accomplishing state stabilization in under 0.2 time units.

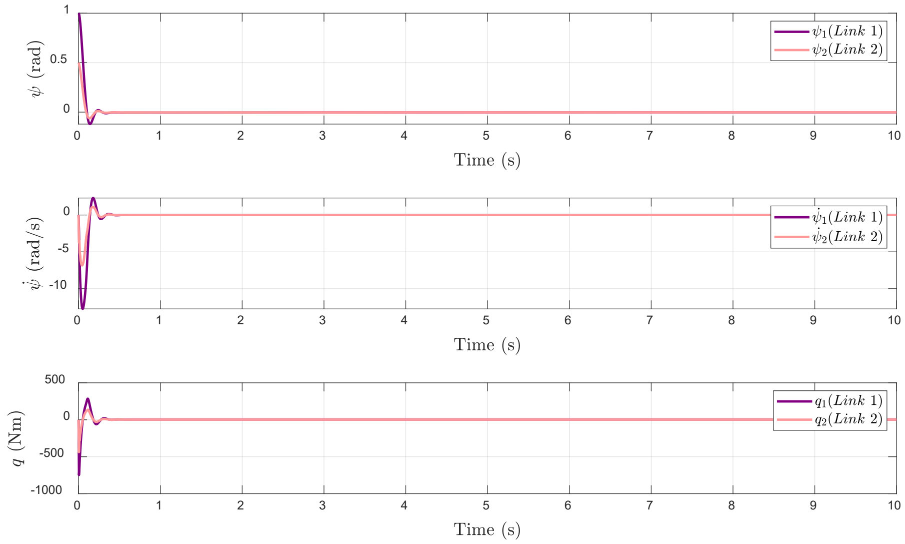

**Figure 5.** Temporal response of the rehabilitation robot's states $\psi$, $\dot{\psi}$, and $q$ and control input for both link 1 and link 2. The depicted behavior corresponds to a scenario where both links have a reference point at the origin and an initial condition given by $\left[\psi_1(0), \psi_2(0), \dot{\psi}_1(0), \dot{\psi}_2(0)\right] = [1, 0.5, 0, 0]$.

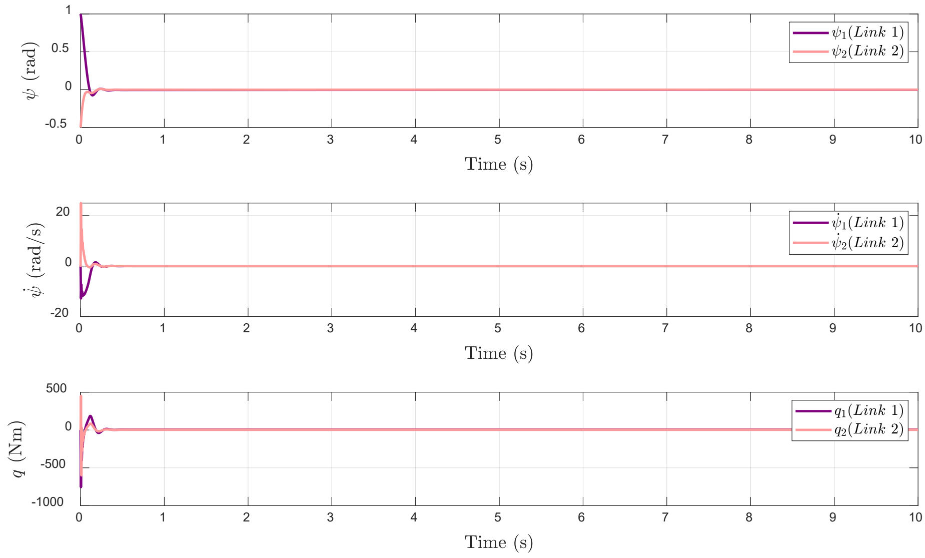

**Figure 6.** Temporal response of the rehabilitation robot's states $\psi$, $\dot{\psi}$, and $q$ and control input for both link 1 and link 2. The depicted behavior corresponds to a scenario where both links have a reference point at the origin and an initial condition given by $\left[\psi_1(0), \psi_2(0), \dot{\psi}_1(0), \dot{\psi}_2(0)\right] = [1, -0.5, 0.2, -0.3]$.

### 4.4. Example 3

We now examine a scenario where the reference values for links 1 and 2 are represented by a step function and differ from one another. The initial conditions remain consistent with the previously discussed example (Example 2). Figure 7 presents the outcomes for this scenario. As depicted in the figure, both links adhere to their respective reference points and achieve these points within a short timeframe. The controller operates based on the error dynamics and the knowledge derived from the pre-trained neural state space.

In summary, our simulations reveal that the introduced control methodology excels in tracking control for the rehabilitation robot, even when faced with entirely unknown dynamics. It ensures the system's stability and robustness. These results underscore the effectiveness of the proposed control strategy and highlight its suitability for real-world scenarios characterized by unexpected dynamics and prevalent uncertainties.

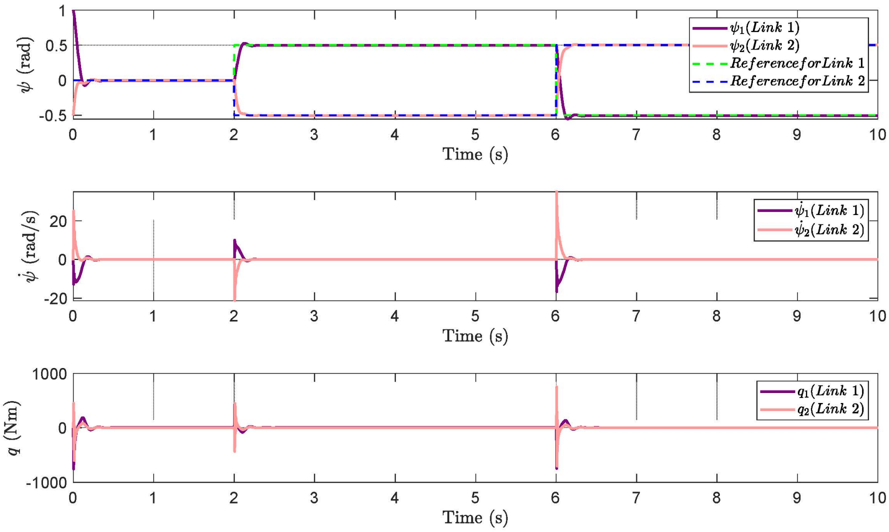

**Figure 7.** Time-series representation of the rehabilitation robot's states $\psi$, $\dot{\psi}$, and $q$ for both link 1 and link 2, tracking their respective step references. The data highlight the robot's behavior when transitioning toward different reference values, starting from an initial condition defined by $\left[\psi_1(0), \psi_2(0), \dot{\psi}_1(0), \dot{\psi}_2(0)\right] = [1, -0.5, 0.2, -0.3]$. The dashed lines indicate the desired reference trajectories for each link.

## 5. Conclusions

In this study, we introduced an innovative control scheme tailored for rehabilitation robots, emphasizing the integration of machine learning algorithms with classical control techniques. As the field of robotic rehabilitation continues its swift evolution, the need for a reliable and precise control mechanism becomes ever more pressing. Our methodology responds to this imperative by seamlessly integrating adaptability with precision. It is designed to accommodate the unique rehabilitation needs of each patient, ensuring personalized therapy without compromising on safety and effectiveness. Central to our approach is the model-free feedback linearization control mechanism, enriched by the capabilities of deep neural networks. Even though the controller operates without a predefined model, learning system dynamics primarily during the training phase, we incorporated an online observer. This observer ensures real-time, robust estimation of uncertainties, facilitating the system's adaptability to challenges that might arise beyond its initial training. Our reliance on the Lyapunov stability theorem offered the requisite verification of the controller's stability. For numerical simulations, at first, we generated data by applying step inputs with randomly varied amplitudes to a simulated 2-DOF robotic system, capturing the resulting positions and velocities. This simulation provided 100 time histories of the system's states and their derivatives, which served as the training dataset for our neural network. Post-training, we implemented the model in three distinct numerical simulations with varying initial conditions and step references for each robotic link. These simulations demonstrated the efficacy of our controller, showcasing its robust performance and adaptability to unknown and dynamic uncertainties. While our presented approach lays a robust foundation for controlling rehabilitation robots, there remain avenues for further enhancements. One such recommendation would be to explore the integration of finite-time control policies as an alternative to feedback linearization. Such an integration could potentially elevate the agility and overall performance of the control system, ensuring even more efficient and responsive rehabilitation robot operations in diverse scenarios.

**Author Contributions:** Conceptualization, H.A. and A.A.; methodology, H.A. and A.A.; software, H.A. and A.A.; validation, H.A. and A.A.; formal analysis, H.A. and A.A.; data curation, H.A. and A.A.; writing—original draft preparation, H.A. and A.A.; writing—review and editing, H.A. and A.A.; visualization, H.A. and A.A.; supervision, H.A. and A.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by the King Salman Center for Disability Research through Research Group no KSRG-2023-487.

**Data Availability Statement:** No new data were created or analyzed in this study. Data sharing is not applicable to this article.

**Acknowledgments:** The authors extend their appreciation to the King Salman Center for Disability Research for funding this work through Research Group no KSRG-2023-487.

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

## References

1. Pană, C.F.; Popescu, D.; Rădulescu, V.M. Patent Review of Lower Limb Rehabilitation Robotic Systems by Sensors and Actuation Systems Used. *Sensors* **2023**, *23*, 6237. [CrossRef] [PubMed]
2. Luo, S.; Meng, Q.; Li, S.; Yu, H. Research of Intent Recognition in Rehabilitation Robots: A Systematic Review. *Disabil. Rehabil. Assist. Technol.* **2023**, 1–12. [CrossRef] [PubMed]
3. Han, S.; Wang, H.; Yu, H. Human–Robot Interaction Evaluation-Based AAN Control for Upper Limb Rehabilitation Robots Driven by Series Elastic Actuators. *IEEE Trans. Robot.* **2023**, *39*, 3437–3451. [CrossRef]
4. Dong, M.; Zhou, Y.; Li, J.; Rong, X.; Fan, W.; Zhou, X.; Kong, Y. State of the Art in Parallel Ankle Rehabilitation Robot: A Systematic Review. *J. NeuroEngineering Rehabil.* **2021**, *18*, 52. [CrossRef] [PubMed]
5. Shi, D.; Zhang, W.; Zhang, W.; Ju, L.; Ding, X. Human-Centred Adaptive Control of Lower Limb Rehabilitation Robot Based on Human–Robot Interaction Dynamic Model. *Mech. Mach. Theory* **2021**, *162*, 104340. [CrossRef]
6. Zhang, L.; Guo, S.; Sun, Q. Development and Assist-as-Needed Control of an End-Effector Upper Limb Rehabilitation Robot. *Appl. Sci.* **2020**, *10*, 6684. [CrossRef]
7. Jahanshahi, H.; Yousefpour, A.; Munoz-Pacheco, J.M.; Moroz, I.; Wei, Z.; Castillo, O. A New Multi-Stable Fractional-Order Four-Dimensional System with Self-Excited and Hidden Chaotic Attractors: Dynamic Analysis and Adaptive Synchronization Using a Novel Fuzzy Adaptive Sliding Mode Control Method. *Appl. Soft Comput.* **2020**, *87*, 105943. [CrossRef]
8. Jahanshahi, H.; Yousefpour, A.; Soradi-Zeid, S.; Castillo, O. A Review on Design and Implementation of Type-2 Fuzzy Controllers. *Math. Methods Appl. Sci.* **2022**, *1*, 485–500. [CrossRef]
9. Yousefpour, A.; Jahanshahi, H.; Gan, D. Fuzzy Integral Sliding Mode Technique for Synchronization of Memristive Neural Networks. In *Mem-Elements for Neuromorphic Circuits with Artificial Intelligence Applications*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 485–500.
10. Kosari, A.; Jahanshahi, H.; Razavi, S. An Optimal Fuzzy PID Control Approach for Docking Maneuver of Two Spacecraft: Orientational Motion. *Eng. Sci. Technol. Int. J.* **2017**, *20*, 293–309. [CrossRef]
11. Mahmoodabadi, M.; Jahanshahi, H. Multi-Objective Optimized Fuzzy-PID Controllers for Fourth Order Nonlinear Systems. *Eng. Sci. Technol. Int. J.* **2016**, *19*, 1084–1098. [CrossRef]
12. Najafizadeh Sari, N.; Jahanshahi, H.; Fakoor, M. Adaptive Fuzzy PID Control Strategy for Spacecraft Attitude Control. *Int. J. Fuzzy Syst.* **2019**, *21*, 769–781. [CrossRef]
13. Jahanshahi, H.; Rajagopal, K.; Akgul, A.; Sari, N.N.; Namazi, H.; Jafari, S. Complete Analysis and Engineering Applications of a Megastable Nonlinear Oscillator. *Int. J. Non-Linear Mech.* **2018**, *107*, 126–136. [CrossRef]
14. Alsaade, F.W.; Jahanshahi, H.; Yao, Q.; Al-zahrani, M.S.; Alzahrani, A.S. A New Neural Network-Based Optimal Mixed H2/H∞ Control for a Modified Unmanned Aerial Vehicle Subject to Control Input Constraints. *Adv. Space Res.* **2023**, *71*, 3631–3643. [CrossRef]
15. Bekiros, S.; Jahanshahi, H.; Bezzina, F.; Aly, A.A. A Novel Fuzzy Mixed H2/H∞ Optimal Controller for Hyperchaotic Financial Systems. *Chaos Solitons Fractals* **2021**, *146*, 110878. [CrossRef]
16. Sari, N.N.; Jahanshahi, H.; Fakoor, M.; Volos, C.; Nikpey, P. Optimal Robust Control Approaches for a Geostationary Satellite Attitude Control. *Int. J. Autom. Control* **2020**, *14*, 333–354. [CrossRef]
17. Ding, Q.; Jahanshahi, H.; Wang, Y.; Bekiros, S.; Alassafi, M.O. Optimal Reinforcement Learning-Based Control Algorithm for a Class of Nonlinear Macroeconomic Systems. *Mathematics* **2022**, *10*, 499. [CrossRef]
18. Chen, S.-B.; Soradi-Zeid, S.; Alipour, M.; Chu, Y.-M.; Gomez-Aguilar, J.; Jahanshahi, H. Optimal Control of Nonlinear Time-Delay Fractional Differential Equations with Dickson Polynomials. *Fractals* **2021**, *29*, 2150079. [CrossRef]
19. Wang, H.; Jahanshahi, H.; Wang, M.-K.; Bekiros, S.; Liu, J.; Aly, A.A. A Caputo–Fabrizio Fractional-Order Model of HIV/AIDS with a Treatment Compartment: Sensitivity Analysis and Optimal Control Strategies. *Entropy* **2021**, *23*, 610. [CrossRef]
20. Yousefpour, A.; Jahanshahi, H.; Munoz-Pacheco, J.M.; Bekiros, S.; Wei, Z. A Fractional-Order Hyper-Chaotic Economic System with Transient Chaos. *Chaos Solitons Fractals* **2020**, *130*, 109400. [CrossRef]

21. Hramov, A.E.; Maksimenko, V.A.; Pisarchik, A.N. Physical Principles of Brain–Computer Interfaces and Their Applications for Rehabilitation, Robotics and Control of Human Brain States. *Phys. Rep.* **2021**, *918*, 1–133. [CrossRef]
22. Xue, X.; Yang, X.; Deng, Z. Efficacy of Rehabilitation Robot-Assisted Gait Training on Lower Extremity Dyskinesia in Patients with Parkinson's Disease: A Systematic Review and Meta-Analysis. *Ageing Res. Rev.* **2023**, *85*, 101837. [CrossRef] [PubMed]
23. Navarrete, M.V.; Velasco, L.R.; Mayorga, C.D.; Carreón, P.A.; Hernández, J.V.; Romero, V.D.; Parra-Vega, V.; Vera, M.M. Output Feedback Self-Tuning Wavenet Control for Underactuated Euler-Lagrange Systems. *IFAC-PapersOnLine* **2018**, *51*, 633–638. [CrossRef]
24. Amiri-Margavi, A.; Babaee, H. On the Connection between Resolvent Analysis and Forced Optimally Time-Dependent Decomposition. In *APS Division of Fluid Dynamics Meeting Abstracts*; NASA: Washington, DC, USA, 2021; p. E31-004.
25. Amiri-Margavi, A.; Babaee, H. On-the-Fly Reduced-Order Modeling of Transient Flow Response Subject to High-Dimensional External Forcing. *Bull. Am. Phys. Soc.* **2022**, *67*, 19–37.
26. Moutinho, D.; Rocha, L.F.; Costa, C.M.; Teixeira, L.F.; Veiga, G. Deep Learning-Based Human Action Recognition to Leverage Context Awareness in Collaborative Assembly. *Robot. Comput.-Integr. Manuf.* **2023**, *80*, 102449. [CrossRef]
27. Ng, K.K.; Chen, C.-H.; Lee, C.K.; Jiao, J.R.; Yang, Z.-X. A Systematic Literature Review on Intelligent Automation: Aligning Concepts from Theory, Practice, and Future Perspectives. *Adv. Eng. Inform.* **2021**, *47*, 101246. [CrossRef]
28. Ai, Q.; Liu, Z.; Meng, W.; Liu, Q.; Xie, S.Q. Machine Learning in Robot Assisted Upper Limb Rehabilitation: A Focused Review. *IEEE Trans. Cogn. Dev. Syst.* **2021**. [CrossRef]
29. Vu, P.P.; Chestek, C.A.; Nason, S.R.; Kung, T.A.; Kemp, S.W.; Cederna, P.S. The Future of Upper Extremity Rehabilitation Robotics: Research and Practice. *Muscle Nerve* **2020**, *61*, 708–718. [CrossRef]
30. Suppiah, R.; Kim, N.; Abidi, K.; Sharma, A. A Comprehensive Review of Motor Movement Challenges and Rehabilitative Robotics. *Smart Health* **2023**, *47*, 100402. [CrossRef]
31. Lee, H.; Choi, Y.; Eizad, A.; Song, W.-K.; Kim, K.-J.; Yoon, J. A Machine Learning-Based Initial Difficulty Level Adjustment Method for Balance Exercise on a Trunk Rehabilitation Robot. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2023**, *31*, 1857–1866. [CrossRef]
32. Zhou, M.; Feng, Y.; Xue, C.; Han, F. Deep Convolutional Neural Network Based Fractional-Order Terminal Sliding-Mode Control for Robotic Manipulators. *Neurocomputing* **2020**, *416*, 143–151. [CrossRef]
33. Kuutti, S.; Bowden, R.; Jin, Y.; Barber, P.; Fallah, S. A Survey of Deep Learning Applications to Autonomous Vehicle Control. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 712–733. [CrossRef]
34. Brunke, L.; Greeff, M.; Hall, A.W.; Yuan, Z.; Zhou, S.; Panerati, J.; Schoellig, A.P. Safe Learning in Robotics: From Learning-Based Control to Safe Reinforcement Learning. *Annu. Rev. Control Robot. Auton. Syst.* **2022**, *5*, 411–444. [CrossRef]
35. Wang, X.; Chen, T.; Chen, S.; Zhu, Y.; Liu, J.; Xu, J.; Soradi-Zeid, S.; Yousefpour, A. Deep Learning-Driven Differentiated Traffic Scheduling In Cloud-Iot Data Center Networks. *Fractals* **2023**, *31*, 1–14. [CrossRef]
36. Hussain, S.; Jamwal, P.K.; Van Vliet, P.; Ghayesh, M.H. State-of-the-Art Robotic Devices for Wrist Rehabilitation: Design and Control Aspects. *IEEE Trans. Hum.-Mach. Syst.* **2020**, *50*, 361–372. [CrossRef]
37. Malik, M.A. Evaluation of Automated Organ Segmentation for Total-Body PET-CT. Available online: https://www.doria.fi/handle/10024/187173 (accessed on 9 October 2023).
38. Guillén, M.F. *2030: How Today's Biggest Trends Will Collide and Reshape the Future of Everything*; St. Martin's Press: New York, NY, USA, 2020; ISBN 1-250-26818-4.
39. Alharbi, N.S.; Jahanshahi, H.; Yao, Q.; Bekiros, S.; Moroz, I. Enhanced Classification of Heartbeat Electrocardiogram Signals Using a Long Short-Term Memory–Convolutional Neural Network Ensemble: Paving the Way for Preventive Healthcare. *Mathematics* **2023**, *11*, 3942. [CrossRef]
40. Alsaadi, F.E.; Jahanshahi, H.; Yao, Q.; Mou, J. Recurrent Neural Network-Based Technique for Synchronization of Fractional-Order Systems Subject to Control Input Limitations and Faults. *Chaos Solitons Fractals* **2023**, *173*, 113717. [CrossRef]
41. Yao, Q.; Jahanshahi, H.; Moroz, I.; Bekiros, S.; Alassafi, M.O. Indirect Neural-Based Finite-Time Integral Sliding Mode Control for Trajectory Tracking Guidance of Mars Entry Vehicle. *Adv. Space Res.* **2023**, *71*, 3723–3733. [CrossRef]
42. Jahanshahi, H.; Yao, Q.; Khan, M.I.; Moroz, I. Unified Neural Output-Constrained Control for Space Manipulator Using Tan-Type Barrier Lyapunov Function. *Adv. Space Res.* **2023**, *71*, 3712–3722. [CrossRef]
43. Alsaade, F.W.; Al-zahrani, M.S.; Yao, Q.; Jahanshahi, H. A Self-Evolving Neural Network-Based Finite-Time Control Technique for Tracking and Vibration Suppression of a Carbon Nanotube. *Mathematics* **2023**, *11*, 1581. [CrossRef]
44. Jahanshahi, H.; Castillo, O.; Yousefpour, A. *Chaotic Variable-Order Fractional Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2022.
45. Jahanshahi, H.; Shahriari-Kahkeshi, M.; Alcaraz, R.; Wang, X.; Singh, V.P.; Pham, V.-T. Entropy Analysis and Neural Network-Based Adaptive Control of a Non-Equilibrium Four-Dimensional Chaotic System with Hidden Attractors. *Entropy* **2019**, *21*, 156. [CrossRef]
46. Shtessel, Y.B.; Shkolnikov, I.A.; Levant, A. Smooth Second-Order Sliding Modes: Missile Guidance Application. *Automatica* **2007**, *43*, 1470–1476. [CrossRef]

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