

Article

Not peer-reviewed version

Disturbance-Resilient Flatness-Based Control for End-Effector Rehabilitation Robotics

Soraya Bououden , [Brahim Brahmi](#) ^{*} , [Naveed Iqbal](#) , [Raouf Fareh](#) , [Mohammad Habibur Rahman](#)

Posted Date: 27 May 2025

doi: 10.20944/preprints202505.2123.v1

Keywords: Rehabilitation robots; Flatness-based control; Kalman filter; Disturbance compensation



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

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.

Article

Disturbance-Resilient Flatness-Based Control for End-Effector Rehabilitation Robotics

Soraya Bououden ^{1,†,‡}, Brahim Brahmi ^{2,†,‡} , Naveed Iqbal ^{3,†,‡}, Raouf Fareh ^{4,†,‡} and Mohammad Habibur Rahman ^{5,*}

- ¹ Electrical Engineering Department: Ferhat Abas Setif 1 University Setif 19000, Algeria; sorayabououden@gmail.com
- ² Department of Electrical Engineering Center for Interdisciplinary Research Center for Intelligent Manufacturing & Robotics (IRC-IMR) King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia
- ³ Department of Electrical Engineering King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia; naveediqbal@kfupm.edu.sa
- ⁴ Department of Electrical Engineering University of Sharjah, Sharjah, The United Arab Emirates; rfareh@sharjah.ac.ae
- ⁵ Department of Mechanical Engineering University of Wisconsin-Milwaukee, WI, USA; rahmanmh@uwm.edu
- * Correspondence: brahim.brahmi@kfupm.edu.sa
- † Current address: Affiliation.
- ‡ These authors contributed equally to this work.

Abstract: Robotic-assisted therapy is an increasingly vital approach for upper-limb rehabilitation, offering consistent, high-intensity training critical to neuroplastic recovery. However, current control strategies often lack robustness against uncertainties and external disturbances, limiting their efficacy in dynamic, real-world settings. Addressing this gap, this study proposes a novel control framework for the iTbot—a 2-DoF end-effector rehabilitation robot—by integrating differential flatness theory with a derivative-free Kalman filter (DFK). The objective is to achieve accurate and adaptive trajectory tracking in the presence of unmeasured dynamics and human-robot interaction forces. The control design reformulates the nonlinear joint-space dynamics into a 0-flat canonical form, enabling real-time computation of feedforward control laws based solely on flat outputs and their derivatives. Simultaneously, the DFK-based observer estimates external perturbations and unmeasured states without requiring derivative calculations, allowing for online disturbance compensation. Extensive simulations across nominal and disturbed conditions demonstrate that the proposed controller significantly outperforms conventional flatness-based control in tracking accuracy and robustness, as measured by reduced mean absolute error and standard deviation. Experimental validation under both simple and repetitive physiotherapy tasks confirms the system’s ability to maintain sub-millimeter Cartesian accuracy and sub-degree joint errors even amid dynamic perturbations. These results underscore the controller’s effectiveness in enabling compliant, safe, and disturbance-resilient rehabilitation, paving the way for broader deployment of robotic therapy in clinical and home-based environments.

Keywords: Rehabilitation robots; Flatness-based control; Kalman filter; Disturbance compensation

1. Introduction

Robotic-assisted therapy has gained significant attention as an effective solution for restoring motor function in individuals with neuromuscular impairments, particularly those recovering from stroke and related conditions [1–3]. These systems enable high-intensity, repetitive, and goal-directed training, which is essential for promoting neuroplasticity and functional recovery. Unlike traditional physical therapy, where therapists manually guide patients through rehabilitation exercises, robotic systems provide structured assistance in passive, active, and assist-as-needed modes. Moreover, they facilitate precise monitoring of patient progress and allow for objective assessment of motor recovery [4,5]. Despite these advantages, significant challenges persist—especially in personalizing therapy for individual patients and scaling robotic systems for broader clinical and at-home applications [6].

Rehabilitation robots are typically categorized into two main types: exoskeleton and end-effector systems [6]. Exoskeleton robots are wearable devices designed to mimic the anatomical structure

of the human arm, enabling multi-joint movement support for comprehensive rehabilitation [7–10]. While these robots can perform complex therapy routines, they tend to be bulky, expensive, and challenging to operate—factors that limit their suitability for home-based use [11]. In contrast, end-effector robots interact with users through a single physical interface, such as the wrist or forearm. Their mechanical simplicity, lower cost, and ease of deployment make them well-suited for non-clinical settings, including home rehabilitation [12]. Nonetheless, access to portable and consistent therapy remains limited, particularly for individuals facing logistical barriers to attending healthcare facilities.

To bridge this gap, we introduce *iTbot*, a two-degrees-of-freedom (2-DoF) end-effector-type therapeutic robot specifically designed for upper-limb rehabilitation. The iTbot is lightweight, compact, and capable of delivering therapy in both clinical and home environments. Its design emphasizes usability, portability, and adaptability, thereby expanding access to high-quality rehabilitation beyond traditional care settings.

A key challenge in robotic rehabilitation is the development of control strategies that facilitate accurate, safe, and robust interactions with users. For end-effector systems, the primary control goal is to achieve precise trajectory tracking during repetitive tasks such as elbow flexion and extension. However, this objective is complicated by actuator dynamics, system nonlinearities, external disturbances, and the inherently variable nature of human-robot interaction [13]. Several control methodologies have been proposed to address these issues. For example, backstepping control employs Lyapunov-based recursive design techniques to ensure stability, but it remains highly sensitive to modeling errors [14]. Other approaches, including fuzzy logic, neural networks, hybrid adaptive control, and machine learning, are capable of managing uncertainties [15–18]; however, they often entail high computational overhead, which complicates real-time implementation. Alternative strategies such as adaptive control [19], EMG-driven methods [20], admittance-based control [21], and sliding mode control [22] are also used but suffer from drawbacks like model dependency and noise sensitivity. Time Delay Estimation (TDE) provides a non-model-based compensation method for unmodeled dynamics [23], though it is often affected by noise and latency. Similarly, Extended Kalman Filters (EKF) have been employed for state estimation [24,25], but the requirement for Jacobian-based linearization can compromise stability and increase implementation complexity.

A promising alternative lies in the flatness-based control paradigm introduced by Fliess and collaborators [26,30]. Differential flatness allows system states and control inputs of nonlinear systems to be expressed in terms of flat outputs and their derivatives, thereby significantly simplifying the control design process. Building upon our previous work [45], this paper presents a novel flatness-based control framework for the iTbot system. The proposed approach transforms the system dynamics into a 0-flat canonical form and integrates a derivative-free Kalman filter (DFK) to estimate unmeasured states and external disturbances in real-time. This combined method enhances tracking accuracy and robustness while maintaining computational efficiency, making it suitable for real-time application.

The main contributions can be summarized as follows:

- Development of a flatness-based control strategy specifically tailored to the iTbot platform, enabling precise and robust trajectory tracking.
- Derivation of a 0-flat canonical form of the system dynamics using differential geometric tools.
- Integration of a derivative-free Kalman filter for real-time estimation and rejection of external disturbances and unmeasured dynamics, without the need for derivative computations.
- Demonstration—through both simulations and real-time experiments—that the proposed method offers improved tracking performance and robustness under a variety of disturbance conditions, all with reduced computational load.

The remainder of this paper is structured as follows. Section 2 details the design and mechanical specifications of the iTbot system. Section 3 presents both the kinematic and dynamic modeling. In Section 4, the flatness-based control methodology is introduced alongside the disturbance estimation scheme employing DFK. Section 5 discusses the simulation-based comparative study, while Section

6 describes the experimental validation. Finally, Section 7 concludes the paper with a summary of findings and outlines directions for future research.

2. Mechanical Design of iTbot

The development of the iTbot robot is centered on a workspace that is accessible to humans, enabling therapy throughout its entire range of motion. Its structural design is intentionally simplified while ensuring that it fully supports robot-assisted rehabilitation, as illustrated in Figure 1. The robot’s framework consists of two linkages: Link-1 and Link-2, both mounted onto a sturdy base. To provide stability during experimental operations, the base of the iTbot is made from a heavy aluminum block. This block is crafted from two aluminum components: one forms the bottom of the robot, while the other acts as a mounting platform for the Motor-1 (Joint-1) assembly, as depicted in Figure 1. Joint-1 (Motor-1) incorporates a harmonic drive gear reducer, which is mounted directly onto the top base, with the motor situated behind it using a custom-designed motor adapter. Link-1, constructed from an aluminum segment featuring an integrated gear reducer, connects directly to the output of the harmonic drive gear unit. In the Link-2 assembly, the fabricated aluminum part also contains the second half of the Motor-2 wire spool holder. Joint-2 (Motor-2) is made up of two interconnected links: Link-1 and Link-2.

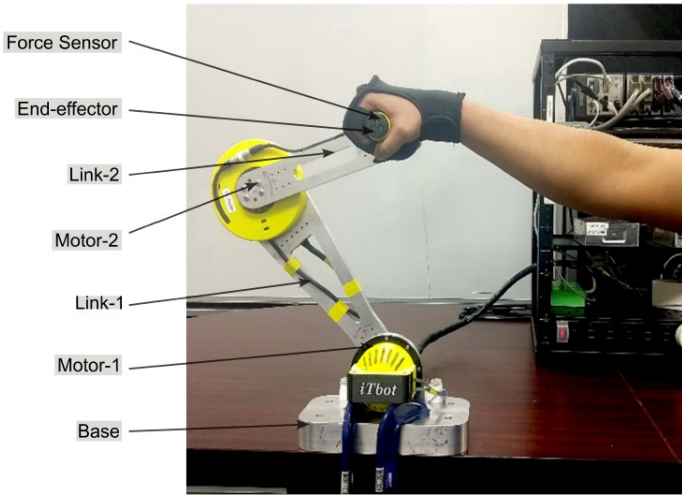


Figure 1. iTbot: Mechanical Architecture [44]

At the end of the connection, an end effector holds both the force sensor and the handle. The handle is specifically tailored to fit the user’s hand profile; in our prototype, its design was based on the average hand size of two adult male laboratory members who participated in the development of the iTbot. This handle includes a base component that has been 3D-printed from polycarbonate plastic and is attached to the force sensor. An inner tube, equipped with mounting points for two bearings at either end, is then screwed into the base component to finalize the assembly.

2.1. Specification of the iTbot

The completed CAD model outlined in Section 2 was created by applying material properties to each component with SolidWorks software. Table 1 provides a summary of the mechanical parameters derived for the iTbot.

Figure 2 demonstrates that the iTbot offers a range of motion of 1.1 meters along the X-axis. This robot was specifically designed to assist the motion of the human upper limb [32] within a workspace that accommodates individuals ranging in height from 1.21 to 1.82 meters, allowing for both horizontal and vertical orientations. In Figure 2, the shaded area indicates the workspace coverage for a human arm, extending from 0.5 meters to 0.66 meters. Moreover, the iTbot’s base design facilitates installation

in either a horizontal or vertical position, provided it is adequately anchored. Its symmetric design, paired with asymmetric joint ranges of motion, makes the robot suitable for use by both left- and right-handed individuals.

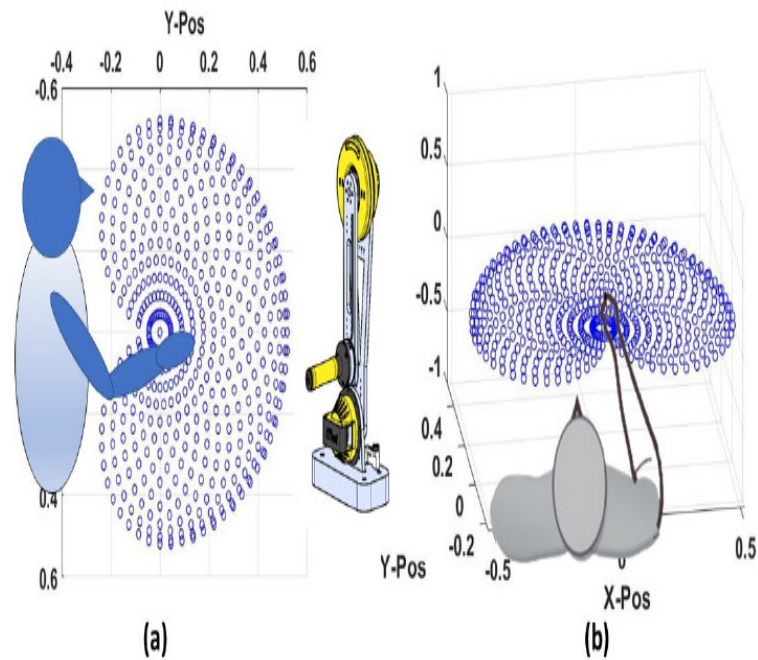


Figure 2. Reachable workspace of the upper limb under (a) vertical and (b) horizontal arrangements [44]

Table 1. Mechanical parameters of iTbot estimated from CAD model [44]

Joint Parameters		
Item	Joint-1	Joint-2
Joint Range of Motion (Degrees)	$\hat{A}\pm85\hat{A}^{\circ}$	$\hat{A}\pm180\hat{A}^{\circ}$
Link Parameters		
Mass (Kg)	1.79	0.65
Location of the center of gravity in link frame (m)	Center of gravity of link 1 in frame {1}, see Figure 3	Center of gravity of link 2 in frame {2}, see Figure 3
	$X_1=0.26,$ $Y_1=0.00,$ $Z_1=0.00$	$X_2=0.15,$ $Y_2=0.00,$ $Z_2=0.02$
Robot Properties		
Mass (Kg)	6.67 (3.2 without base)	
Maximum Horizontal reach (m)	$\hat{A}\pm0.55$	
Maximum Vertical reach (m)	+0.1 to +0.55	

3. Kinematics and Mathematical Model of iTbot

The kinematic model of the iTbot employs the modified Denavit–Hartenberg (DH) notation [44]. As depicted in Figure 3, only two joint angles are actively controlled during rehabilitation training. Each link in the robot’s structure has an associated coordinate frame (link frame) that specifies its position concerning adjacent links. The DH parameters are determined by assigning coordinate frames that link

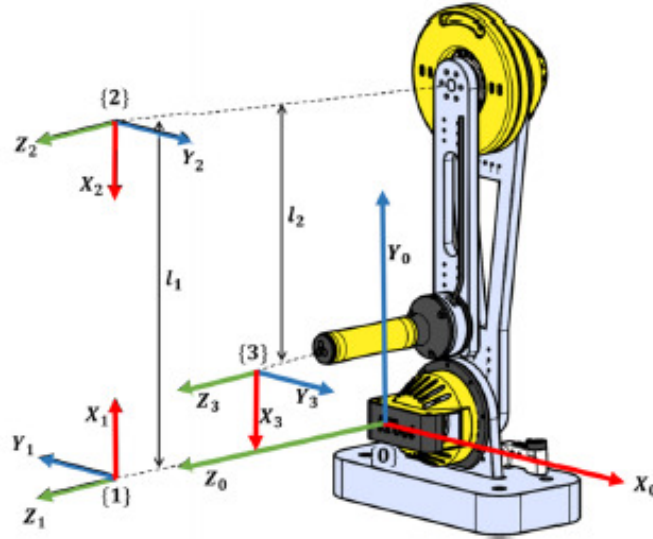


Figure 3. Coordinate frame assignment for the 2-DoF iTbot [44].

one axis of rotation to the next. Frame 1 is associated with Joint 1, Frame 2 corresponds to Joint 2, and Frame 3 is designated for the iTbot's end-effector. Frame 0 acts as the fixed reference frame, aligned with the world (base) frame. The modified DH parameters, which describe the spatial relationships of the link frames (shown in Figure 3), are summarized in Table 2. Additionally, integrating the DH parameters, we obtain the homogeneous transformation matrix, which defines the positions and orientations of the reference frames relative to the fixed base frame.

Table 2. the modified Denavit-Hartenberg (DH)

joint (i)	α_{i-1}	d_i	a_{i-1}	θ_i
ine 1	0	0	0	$\theta_1 + \frac{\pi}{2}$
2	0	0	l_1	$\theta_2 + \pi$
3	0	0	l_2	0
ine				

In Table 2, α_{i-1} denotes the link twist, a_{i-1} is the link length, d_i represents the link offset, and θ_i is the joint angle.

3.1. Dynamics of the iTbot

The dynamic behavior of the iTbot is examined using the iterative Newton–Euler formulation, as shown in Figure 4 [34]. This approach allows for precise simulation of joint movements in experiments that utilize nonlinear control strategies.

The dynamic model of the iTbot, expressed in joint space, is illustrated below:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G_r(\theta) + \zeta_{dis} = \tau \quad (1)$$

where $\theta \in \mathbb{R}^2$ denotes the vector of generalized coordinates. $M(\theta) \in \mathbb{R}^{2 \times 2}$ represents the symmetric positive-definite inertia matrix, $C(\theta, \dot{\theta})\dot{\theta} \in \mathbb{R}^2$ accounts for Coriolis and centrifugal forces, and $G_r(\theta) \in \mathbb{R}^2$ denotes the gravity vector. $\tau \in \mathbb{R}^2$ represents the torque input vector, while $\zeta_{dis} \in \mathbb{R}^2$ incorporates external disturbances. Furthermore, it is important to note that $\zeta_{dis} = J^T F_e$, where $J \in \mathbb{R}^{6 \times 2}$ is the Jacobian matrix and $F_e \in \mathbb{R}^{6 \times 1}$ denotes the contact force between the human and the robot.

The dynamic model presented in Equation (1) can be transformed into a state-space representation by incorporating the variables $x = \theta$ and $\dot{x} = \dot{\theta}$. This results in the following system:

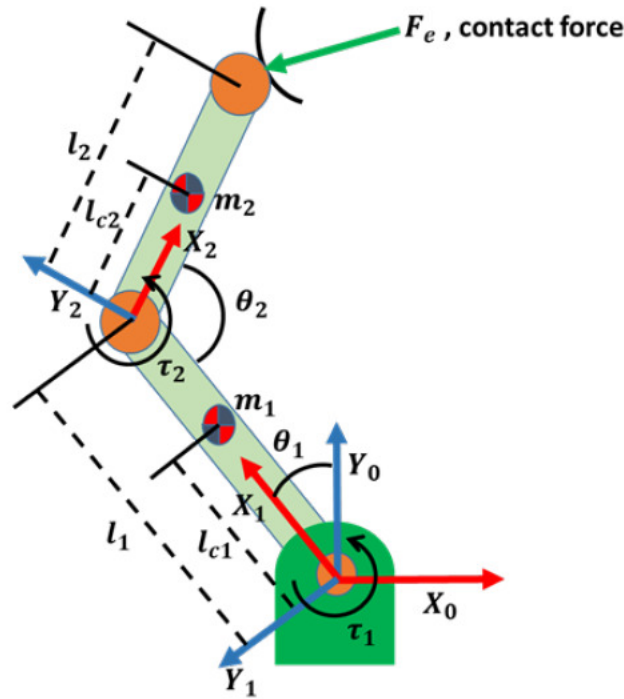


Figure 4. iTBot nomenclature for dynamic modeling, including contact force at the end-effector.

$$\dot{x} = F(x, \dot{x}) + G(x)u \quad (2)$$

where:

- $u = \tau$ is the control input (torque),
- $G(x) = G = M^{-1}(\theta)$ is the inverse of the inertia matrix,
- $F(x, \dot{x}) = M^{-1}(\theta)[-C(\theta, \dot{\theta})\dot{\theta} - G_r(\theta) - \tilde{\zeta}_{dis}(\theta, \dot{\theta})]$ represents the system dynamics including nonlinearities and disturbances.

4. Control Design

It has been shown that the robot model, as described in Eq. 2, is affected by uncertainties and external disturbances. The main objective is to develop a robust controller capable of supporting physical therapy movements. This will be accomplished through the use of a flat controller along with estimation techniques. The approach includes two essential strategies: the first takes advantage of the flatness property applicable to a certain class of dynamic systems, and the second employs the derivative-free Kalman (DFK) method.

4.1. Notation

Firstly, the following are presented [35] :

- **Definition 1 :** The Lie derivative is defined as the derivative of γ , with respect to the vector field F given by the following: $L_F \gamma = \frac{\partial \gamma}{\partial x} F(x)$. $L_F^0 \gamma = \gamma$ for $i = 0$, and for $i = 1, 2, \dots$ we have $L_F^i \gamma = L_F L_F^{i-1} \gamma = \frac{\partial L_F^{i-1} \gamma}{\partial x} F(x)$.
- **Definition 2 :** The Lie Bracket of the two vector fields F and G , denoted as $ad_F G$ or $[F, G](x)$, is defined as $[F, G](x) = \frac{\partial G}{\partial x} F - \frac{\partial F}{\partial x} G$. Additionally, we have $ad_F^0 G = G(x)$ and $ad_F^i G = [F, ad_F^{i-1} G]$.

4.2. The Differential Geometric Approach

Let us start by defining the flatness property for a dynamic system that includes a control input described as follows:

$$\dot{x} = F(x) + \sum_{i=1}^m G_i(x)u_i \quad (3)$$

where x represents the state vector defined on $\mathcal{X} \subseteq \mathbb{R}^n$, u is the input vector defined on $U \subseteq \mathbb{R}^m$ while F, G are the vector fields defined on $\mathcal{X} \times U$.

• **Definition 3** A system described by Equation (3) is considered differentially flat if there are γ variables $\gamma \in \mathbb{R}^m$, known as flat outputs $\gamma = (\gamma_1, \dots, \gamma_m)$, that satisfy the following condition:

1. γ is written as function of the state vector x , the control input vector u and its time derivatives u^i : $\gamma = h(x, u, \dot{u}, \dots, u^{(r_1)})$.
2. The components of the state vector x can be expressed from the flat output γ and its time derivatives : $x = \varphi(\gamma, \dot{\gamma}, \dots, \gamma^{(r_2)})$.
3. The components of the input vector u can be expressed from the flat output and its time derivatives : $u = y(\gamma, \dot{\gamma}, \dots, \gamma^{(r_2+1)})$, where h, φ , and y are smooth functions.

Let $G(x) = [G_1(x), \dots, G_m(x)]$ denote a set of vector fields. We assume that this set has a rank of m .

The flat system is characterized by flat outputs, where all control inputs and state vectors are defined in relation to these output variables and a limited number of their derivatives. For a specific class of differentially flat systems, in this work we have only $\gamma = h(x)$ considered in (1) of Definition (3), this means that these systems are called 0-flat.

It demonstrates that dynamic affine systems with a codimension of 1, possessing n states and $n - 1$ inputs, are controllable if and only if these systems are 0-flat [36].

One of the primary objectives of this work is to apply the properties of differential flatness. Once the system is demonstrated to be flat, it can undergo a transformation into the proposed 0-flat normal form, where geometric conditions are adequately provided. Similarly, this procedure yields an algorithm for computing the flat outputs. Additionally, a dynamic system (3) which has n states and m inputs, is classified as having codimension 2 when the difference between n and m equals 2.

4.3. 0-Flat Form for Co-Dimension 2 System

The suggested canonical normal form for co-dimension 2 dynamic systems is presented as follows:

Proposition 1: A dynamic system of co-dimension 2 is considered 0-flat if it can be expressed in the following :

$$\begin{cases} \dot{\zeta}_{1,1} &= \zeta_{1,2} + \sum_{i=3}^{m=n-2} \varphi_{1,1}^i(\zeta)u_i \\ \dot{\zeta}_{1,2} &= \beta_{1,2}(\zeta) + \varphi_{1,2}^1(\zeta)u_1 + \varphi_{1,2}^2(\zeta)u_2 + \dots + \varphi_{1,2}^m(\zeta)u_m \\ \dot{\zeta}_{2,1} &= \zeta_{2,2} + \sum_{i=3}^{m=n-2} \varphi_{2,1}^i(\zeta)u_i \\ \dot{\zeta}_{2,2} &= \beta_{2,2}(\zeta) + \varphi_{2,2}^1(\zeta)u_1 + \varphi_{2,2}^2(\zeta)u_2 + \dots + \varphi_{2,2}^m(\zeta)u_m \\ \dots &= \dots \dots \dots \dots \\ \dot{\zeta}_{j,1} &= \beta_{j,1}(\zeta) + \sum_{i=3, i \neq j}^{m=n-2} \varphi_{j,1}^i(\zeta)u_i \end{cases} \quad (4)$$

The flat outputs are represented as : $\gamma_j = (\zeta_{j,1})_{1 \leq j \leq m=n-2}$, and $\zeta = ((\zeta_{j,1})_{1 \leq j \leq m=n-2}, \zeta_{1,2}, \zeta_{2,2})$ is the updated vector of the system.

Proof 1 of proposition 1

Now, we will demonstrate that the previously described system dynamics, represented in the proposed normal form of equations (4) is 0-flat. We will examine the following m equations:

$$T_1 = \zeta_{1,1} - \zeta_{1,2} - \sum_{i=3}^{m=n-2} \varphi_{1,1}^i(\zeta) u_i = 0 \quad (5)$$

$$T_2 = \check{\zeta}_{2,1} - \zeta_{2,2} - \sum_{i=3}^{m=n-2} \varphi_{2,1}^i(\zeta) u_i = 0 \quad (6)$$

$$T_j = \zeta_{j,1} - \beta_{j,1}(\zeta) - \sum_{i=3, i \neq j}^{m=n-2} \varphi_{j,1}^i(\zeta) u_i = 0 \quad (7)$$

where $(3 \leq j \leq m = n - 2)$.

Consider the vector $\eta = (\zeta_{1,2}, \zeta_{2,2}, u_1, u_2, \dots, u_{m=n-2})$, which consists of the unknown system variables. We will use this vector later to compute the partial derivative of $T_1, T_2, \dots, T_{m=n-2}$ as follows:

$$\frac{\partial(T_1, T_2, \dots, T_m)}{\partial \eta} = -I_m + \rho(\eta) \quad (8)$$

where I_m denotes the $(m \times m)$ identity matrix, and $\rho(\eta)$ comprises terms that are of order one with respect to the η variables. Using the equations provided (5) – (7), and recognizing that the partial derivative $\frac{\partial(T_1, \dots, T_m)}{\partial \eta}$ is locally invertible, we can apply the implicit function theorem. This enables us to conclude that there exist two functions $\lambda_k()$ and $\delta_k()$ such that :

$$\zeta_{k,2} = \lambda_k(\gamma_j, \dot{\gamma}_j, j = 1, \dots, n-2, k = 1:2) \quad (9)$$

$$u_k = \delta_k(\gamma_j, \dot{\gamma}_j, j = 1, \dots, n-2, k = 3 : m) \quad (10)$$

Next, by substituting equations (9) and (10) into equation (4), we can derive the unknown system variables, which are the control inputs (u_1 , u_2 , and $u_3, \dots, u_m$). These inputs are expressed in terms of the flat outputs $(\gamma_j, \dot{\gamma}_j)_{(j=1:m=n-2)}$ and their second time derivative $(\gamma_1^2, (\gamma_2^2), \dots, \gamma_m^2$.

Thus, the state vector and control inputs can be represented as functions of flat outputs γ and their successive time derivatives. This indicates that dynamic systems with co-dimension 2 exhibit differential flatness properties. Consequently, Proposition 1 has been proven.

4.4. Geometrical Background

This section focuses on outlining the geometrically sufficient conditions required for our dynamic system to be transformed into the previously proposed flatness normal form, in which the outputs are flat.

Next, we assume that the controllability distribution, represented by Δ , has a dimension of n such that

$$\Delta = span\{G_1, G_2, ..., G_{n-2}, ad_F G_i, ad_F G_j\} \quad (11)$$

where $ad_F G_i, ad_F G_j$ denotes Lie brackets.

It's important to recognize that $G_i, G_j \in \{G_1, G_2, G_3, G_4, \dots, G_{n-2}\}$. For ease of discussion, the vector fields G_i and G_j can be expressed as $G_i = G_1$ and $G_j = G_2$, respectively, without losing generality.

Theorem 1.

The transformation of the system of Equation (3) into the proposed 0-flat form occurs through a change of coordinates if and only if the distribution $\bar{\Delta} = \{G_1, G_2\} \subseteq \Delta$ is involutive.

Proof of Theorem 1.

Δ is considered as nonsingular distribution generated by the set of vector fields. If the involutive condition $\bar{\Delta} = \{G_1, G_2\} \subseteq \Delta$ is verified, then Frobenius theorem [37] states that there exist $m = n - 2$

independent state functions ($\gamma_i(x) = \zeta_{i,1}$) ($1 \leq i \leq m=n-2$) such that:

$$\bar{\Delta} = \bigcap_{i=1}^m \ker d\gamma_i$$

where $\ker d\gamma_i$ denotes the kernel of the differential of the function γ_i . Next, we introduce the following Lie brackets:

$$L_{[G_k, F]} \gamma_k \neq 0, \text{ for } k = 1, 2 \quad (12)$$

$$L_{G_k} \gamma_k = 0, \text{ for } k = 1 : 2 \quad (13)$$

As a result, the new state vector ζ is created based on the following equations: ((12), (13), detailed as follows :

$$(\zeta_{i,1} = \gamma_i(x), (1 \leq i \leq m=n-2)), \text{ and } (\zeta_{1,2} = L_F \zeta_{1,1}, \zeta_{2,2} = L_F \zeta_{2,1}) \quad (14)$$

Now, if we consider $\zeta_{1,1} = \gamma_1(x)$, $\zeta_{1,2} = L_F \gamma_1(x)$, the derivation of $\zeta_{1,1}$ results in

$$\dot{\zeta}_{1,1} = \zeta_{1,2} + \sum_{i=3}^{m=n-2} L_{G_i} \gamma_1(x) u_i \quad (15)$$

Thus, we can put $\varphi_{1,1}^i(\zeta) = L_{G_i} \gamma_1(x)$ and $\varphi_{2,1}^i(\zeta) = L_{G_i} \gamma_2(x)$, while the other variables: $(\varphi_{1,2}^1(\zeta), \varphi_{1,2}^2(\zeta) \dots \varphi_{1,2}^m(\zeta))$ and $(\varphi_{2,2}^1(\zeta), \varphi_{2,2}^2(\zeta) \dots \varphi_{2,2}^m(\zeta))$ are also defined in Equation (4) by the derivation of the variable $\zeta_{1,2}$ and $\zeta_{2,2}$.

Now, we can obtain the proposed flatness normal form (4) by applying the same procedure of successive derivations to all other selected outputs of the system.

4.5. Design of a Flatness Based Controller for iTbot's Robot System

To demonstrate that iTbot's dynamic model (2) is differentially flat, we will utilize the following transformations to achieve a proposed 0-flat form (4). Let us begin by rewriting the robot's dynamic model (2) in this format:

$$\dot{x} = F(x) + \sum_{i=1}^{m=n-2} G_i(x) u_i \quad (16)$$

where $x^T = [\theta_1 \ \theta_2 \ \dot{\theta}_1 \ \dot{\theta}_2] = [x_1 \ x_2 \ x_3 \ x_4]$ represents the state space vector of iTbot's dynamic model, u_i signifies the inputs, and $F(x)$ and $G_i(x)$ are smooth vector fields.

$$F(x) = \begin{bmatrix} x_3 \\ x_4 \\ F_1(x) \\ F_2(x) \end{bmatrix} \quad (17)$$

with $F'(x) = [F_1(x) \ F_2(x)]^T$ where $F'(x) = M^{-1}(x_1)[\tau - \mathcal{C}(x_1, x_2)x_2 - G_r(x_1) - \zeta_{dis}]$, and

$$G_i(x) = \begin{bmatrix} 0 \\ 0 \\ G_{i1}(x) \\ G_{i2}(x) \end{bmatrix} \quad (18)$$

where $i = n - 2$.

Next, we will examine the controllability distribution defined by the following vector fields:

$\Delta = \text{span}\{G_1, G_2, \text{ad}_F G_1, \text{ad}_F G_2\}$. In contrast, the dimension is consistently set to $n=4$. Meanwhile,

the distribution of dimensions $\bar{\Delta} = \text{span}\{G_1, G_2\}$ is 2..

Thus, when verifying the following Lie brackets, one finds that

$$[G_i, G_j] \in \Delta \quad \forall \quad 1 \leq i, j \leq n - 2 = m$$

The distribution $\bar{\Delta} \in \Delta$ is involutive, that the earlier condition stated in Theorem 1 is satisfied.

Next, we can also ensure the existence of $n - 2$ smooth functions γ_1, γ_2 that satisfy the following partial differential equations, as dictated by the Frobenius theorem:

$$d\gamma_1(x) \cdot \bar{\Delta} = 0 \quad (19)$$

$$d\gamma_2(x) \cdot \bar{\Delta} = 0 \quad (20)$$

Equations (19) and (20) written as :

$$d\gamma_1(x)[G_1(x), G_2(x)] = 0 \quad (21)$$

$$d\gamma_2(x)[G_1(x), G_2(x)] = 0 \quad (22)$$

From (19) and (20), we can conclude that for all (x_1, x_2) , it holds that:

$$\frac{\partial \gamma_i}{\partial x_j} = 0 \quad (i = 1 : 2 \text{ and } j = 3 : 4) \quad (23)$$

As a result of the analysis presented above, we can describe $\gamma_1(x)$ and $\gamma_2(x)$ in the following manner:

$$\gamma_1(x) = x_1 = \zeta_{1,1} \quad (24)$$

$$\gamma_2(x) = x_2 = \zeta_{2,1} \quad (25)$$

where $\zeta_{1,1}$, and $\zeta_{2,1}$ represent the flat outputs generated by iTbot's dynamic model.

The new state variables ζ of the system are created from the collection of flat outputs $\zeta_{1,1}$, $\zeta_{2,1}$, and 2, along with the additional variables: $\zeta = (\zeta_{i,1} = \gamma_i(x), 1 \leq i \leq 2)$ and $\zeta_{1,2} = L_F \zeta_{1,1}, \zeta_{2,2} = L_F \zeta_{2,1}$.

The updated vector of the system is expressed as $\zeta = [\zeta_{1,1}, \zeta_{1,2}, \zeta_{2,1}, \zeta_{2,2}]$,

The first equation in normal form (4) is derived from the derivative of the initial flat output $\dot{\zeta}_{1,1}$. Consequently, the derivatives of other variables, we can outline the triangular 0-flat form for a dynamic system of co-dimension 2 as follows:

$$\begin{aligned} \dot{\zeta}_{1,1} &= \zeta_{1,2} + \sum_{i=3}^{m=n-2} \varphi_{1,1}^i(\zeta) u_i \\ \dot{\zeta}_{1,2} &= \beta_{1,2}(\zeta) + \varphi_{1,2}^1(\zeta) u_1 + \varphi_{1,2}^2(\zeta) u_2 + \tilde{d}_1 \\ \dot{\zeta}_{2,1} &= \zeta_{2,2} + \sum_{i=3}^{m=n-2} \varphi_{2,1}^i(\zeta) u_i \\ \dot{\zeta}_{2,2} &= \beta_{2,2}(\zeta) + \varphi_{2,2}^1(\zeta) u_1 + \varphi_{2,2}^2(\zeta) u_2 + \tilde{d}_2 \end{aligned} \quad (26)$$

In the normal flat form, the variables $\tilde{d}_1$ and $\tilde{d}_2$ represent external disturbances, appearing in the second and fourth rows, respectively.

Furthermore, all state variables and the control input of the iTbot model are defined as functions of the flat output and its derivatives. As a result, the studied system is considered to be differentially 0-flat. Now, we can establish the expressions for the appropriate control inputs u_1 and u_2 based on the second and fourth equations from the triangular form (26). Consequently, ϑ_i ($i = 1 : 2$) defines two new control inputs that are created using the specified flat outputs $\zeta_{1,1}$ and $\zeta_{2,1}$, along with their

derivatives, aligned with the desired values of the designated reference trajectories. $\zeta^{ref}, \dot{\zeta}^{ref}, \ddot{\zeta}^{ref}$ such that:

$$\vartheta_i = \ddot{\zeta}_{i,1}^{ref} + k_{d_{i,1}}(\dot{\zeta}_{i,1}^{ref} - \dot{\zeta}_{i,1}) + k_{p_{i,1}}(\zeta_{i,1}^{ref} - \zeta_{i,1}), \quad i=1:2 \quad (27)$$

where $k_{d_{i,1}}$ and $k_{p_{i,1}}$ are selected control gains that ensure the asymptotic elimination of the tracking errors, $\lim_{t \rightarrow \infty} e_i(t) = 0$.

In the following section, we will estimate the external disturbances $\tilde{d}_i$ that influence the robotic model, as well as the disturbance terms related to parametric uncertainty. This will be accomplished simultaneously using the derivative-free Kalman Filtering (DFK) method.

4.6. Filtering Kalman for Dynamical Systems

It is assumed that the dynamical system can be represented as a discrete-time state model ([38], [39], [40]) in the following manner:

$$\begin{aligned} x(k+1) &= A(k)x(k) + J(k)u(k) + \omega(k) \\ z(k) &= Cx(k) + v(k) \end{aligned} \quad (28)$$

where $x \in \mathbb{R}^{m \times 1}$ represents the system's state vector, $z \in \mathbb{R}^{p \times 1}$ is the system's output. $\omega(k)$ and $v(k)$ signify the process noise, which is uncorrelated with the respective covariance matrices $\check{Q}$ and $\check{R}$. A is a $(m \times m)$ real matrix, while C is a $p \times m$ matrix consisting of real numbers. The structure of the discrete-time Kalman filter equations is organized as follows:

$$\begin{aligned} \hat{x}(k) &= \hat{z}^-(k) + \check{K}_F(k)[x(k) - C\hat{z}^-(k)] \\ \check{P}(k) &= \check{P}^-(k) - \check{K}_F(k)C\check{P}^-(k) \end{aligned} \quad (29)$$

The initial stage is referred to as the time update or prediction stage, which provides estimates $x(k)$ prior to the availability of the output measurement $z(k)$ (resulting in an a priori estimate).

$$\begin{aligned} \check{K}_F(k) &= \check{P}^{-1}(k)C^T[C\check{P}^{-1}(k)C^T + \check{R}]^{-1} \\ \check{P}^-(k+1) &= A(k)\check{P}(k)A^T(k) + \check{Q}(k) \\ \hat{x}^-(k+1) &= A(k)\hat{x}(k) + J(k)u(k) \end{aligned} \quad (30)$$

$x(k)$ is estimated during the second part, known as the measurement update or correction stage, after $z(k)$ has been made available (a posteriori estimate).

4.7. Disturbances Compensation Using (DFK) Derivative-Free Kalman Filtering

The controller described here is capable of providing highly accurate tracking of the desired trajectory while also exhibiting robust performance in the face of disturbances. These disturbances may arise from parametric uncertainties, variations, or external factors. To effectively counter these disturbances and uncertainties, we can incorporate Derivative-Free Kalman Filtering (DFK) into the flat-based feedback controller loop, as illustrated in Figure 5.

In Section 4.5, we showed that the nonlinear model of iTbot's robot, as outlined in equation (16), is differentially 0-flat. This was established by converting it into a triangular 0-flat normal form, accomplished through equation (26).

In the dynamic model of the robot expressed in a triangular 0-flat normal form, the stabilizing feedback controller is represented as follows:

$$\begin{aligned} \vartheta_1 &= \ddot{\zeta}_{1,1}^{ref} + k_{d_{1,1}}(\dot{\zeta}_{1,1}^{ref} - \dot{\zeta}_{1,1}) + k_{p_{1,1}}(\zeta_{1,1}^{ref} - \zeta_{1,1}) \\ \vartheta_2 &= \ddot{\zeta}_{2,1}^{ref} + k_{d_{2,1}}(\dot{\zeta}_{2,1}^{ref} - \dot{\zeta}_{2,1}) + k_{p_{2,1}}(\zeta_{2,1}^{ref} - \zeta_{2,1}) \end{aligned}$$

The dynamics of the tracking error are obtained as follows:

$$\begin{aligned}(\ddot{\zeta}_{1,1}^{ref} - \ddot{\zeta}_{1,1}) + k_{d1,1}(\dot{\zeta}_{1,1}^{ref} - \dot{\zeta}_{1,1}) + k_{p1,1}(\zeta_{1,1}^{ref} - \zeta_{1,1}) &= 0 \\(\ddot{\zeta}_{2,1}^{ref} - \ddot{\zeta}_{2,1}) + k_{d2,1}(\dot{\zeta}_{2,1}^{ref} - \dot{\zeta}_{2,1}) + k_{p2,1}(\zeta_{2,1}^{ref} - \zeta_{2,1}) &= 0\end{aligned}$$

Assuming that the effects of both modeling uncertainty and external perturbations on the iTbot's robot are represented as additive disturbance inputs ($\tilde{d}_1, \tilde{d}_2$), these disturbances are incorporated into the initial nonlinear model of the iTbot as follows:

$$\begin{aligned}\ddot{x}_1 &= \vartheta_1 + \tilde{d}_1 \\ \ddot{x}_2 &= \vartheta_2 + \tilde{d}_2\end{aligned}\tag{31}$$

Or in the second and fourth rows of the normal flat form after transformation (26) such that:

$$\begin{aligned}\ddot{\zeta}_{1,1} &= \beta_{1,2}(\zeta) + \varphi_{1,2}^1(\zeta)u_1 + \varphi_{1,2}^2(\zeta)u_2 + \tilde{d}_1 \\ \ddot{\zeta}_{2,1} &= \beta_{2,2}(\zeta) + \varphi_{2,2}^1(\zeta)u_1 + \varphi_{2,2}^2(\zeta)u_2 + \tilde{d}_2\end{aligned}\tag{32}$$

and by establishing the new control inputs

$$\begin{aligned}\vartheta_1 &= \beta_{1,2}(\zeta) + \varphi_{1,2}^1(\zeta)u_1 + \varphi_{1,2}^2(\zeta)u_2 \\ \vartheta_2 &= \beta_{2,2}(\zeta) + \varphi_{2,2}^1(\zeta)u_1 + \varphi_{2,2}^2(\zeta)u_2\end{aligned}\tag{33}$$

The dynamics can be expressed as follows:

$$\begin{aligned}\ddot{\zeta}_{1,1} &= \vartheta_1 + \tilde{d}_1 \\ \ddot{\zeta}_{2,1} &= \vartheta_2 + \tilde{d}_2\end{aligned}$$

The behavior of the disturbance terms is characterized by the time derivatives of the disturbance variables, extending up to order n. [41], [42]. These terms are formatted as follows for n=2:

$$\begin{aligned}\ddot{\tilde{d}}_1 &= f_{\tilde{d}_1} \\ \ddot{\tilde{d}}_2 &= f_{\tilde{d}_2}\end{aligned}\tag{34}$$

To account for the additional state variables, disturbance inputs, and their time derivatives, we expand the system's state vector. The new state variables of the system are then determined as follows:

$$\zeta_e = [\zeta_{1,1} \ \zeta_{1,2} \ \zeta_{2,1} \ \zeta_{2,2} \ \zeta_{\tilde{d}_{1,1}} \ \zeta_{\tilde{d}_{1,2}} \ \zeta_{\tilde{d}_{2,2}} \ \zeta_{\tilde{d}_{2,2}}]^T,$$

Now, we can express the dynamics of the extended state space model of the robot as follows:

$$\zeta_{1,1} = x_1, \ \zeta_{1,2} = \dot{x}_1, \ \zeta_{2,1} = x_2, \ \zeta_{2,2} = \dot{x}_2, \ \zeta_{\tilde{d}_{1,1}} = \tilde{d}_1, \ \zeta_{\tilde{d}_{1,2}} = \dot{\tilde{d}}_1, \ \zeta_{\tilde{d}_{2,1}} = \tilde{d}_2, \ \zeta_{\tilde{d}_{2,2}} = \dot{\tilde{d}}_2,$$

Therefore, the structure of the extended state space of iTbot's robot represented by it's normal flat form (26) of codimension 2 can be written as follows :

$$\begin{aligned}\dot{\zeta}_e &= (A_e)\zeta_e + (B_e)\vartheta \\ \zeta_e^{me} &= (C_e)\zeta_e\end{aligned}\tag{35}$$

A state estimator has been created for the extended state-space representation of the system:

$$\begin{aligned}\dot{\hat{\zeta}}_e &= (A_e)\hat{\zeta}_e + (B_e)\vartheta + K_f(\zeta_e^{me} - C_e\hat{\zeta}_e) \\ \zeta_e^{me} &= C_e\hat{\zeta}_e\end{aligned}\tag{36}$$

Where

$$A_e = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (37)$$

and

$$B_e = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, C_e = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (38)$$

The estimation of the unmeasurable elements of the state vector, as well as the disturbance inputs and their derivatives, is performed using the Kalman Filter recursion, which is outlined in the following step:

- First step (measurement update):

$$\begin{aligned} \check{P}_F(k) &= \check{P}^-(k) C_{di}^T [C_{di} \check{P}^-(k) C_{di}^T + \check{R}]^{-1} \\ \hat{\zeta}_e(k) &= \hat{\zeta}_e^-(k) + \check{K}_F(k) [z_e^m(k) - C_{di} \hat{\zeta}_e^-(k)] \\ \check{P}(k) &= \check{P}^-(k) - K_F(k) C_{di} \check{P}^-(k) \end{aligned} \quad (39)$$

- Second step (time update):

$$\begin{aligned} \check{P}^-(k+1) &= A_{di}(k) \check{P}(k) A_{di}^T(k) + \check{Q}(k) \\ \hat{\zeta}_e^-(k+1) &= A_{di}(k) \hat{\zeta}_e(k) + B_{di}(k) \vartheta(k) \end{aligned} \quad (40)$$

A_{di} , B_{di} and C_{di} are defined as the discrete-time equivalents of the previous matrices A_e , B_e and C_e .

As a result, the new control input for the iTbot's robot has been adjusted to account for the effects of disturbances, expressed as follows:

$$\begin{aligned} \vartheta_1 &= \ddot{\zeta}_{1,1}^{ref} + k_{d1,1} (\dot{\zeta}_{1,1}^{ref} - \dot{\zeta}_{1,1}) + k_{p1,1} (\zeta_{1,1}^{ref} - \zeta_{1,1}) - \hat{\zeta}_{\ddot{d}_{1,1}} \\ \vartheta_2 &= \ddot{\zeta}_{2,1}^{ref} + k_{d2,1} (\dot{\zeta}_{2,1} - \dot{\zeta}_{2,1}) + k_{p2,1} (\zeta_{2,1}^{ref} - \zeta_{2,1}) - \hat{\zeta}_{\ddot{d}_{2,1}} \end{aligned} \quad (41)$$

The dynamics of tracking error for the closed-loop system can be described by the following equation:

$$\ddot{e}_i + k_{d_i} \dot{e}_i + k_{p_i} e_i = 0 \quad (42)$$

4.8. Stability Studies

In the following section, the Lyapunov function to demonstrate the asymptotic stability of the closed-loop system, which can be :

$$\ddot{e}_i + k_{d_i} \dot{e}_i + k_{p_i} e_i = 0 \quad (43)$$

Proposition 2

The dynamic model of the iTbot described earlier has been shown to be 0-flat, based on the principles of differential flatness theory (26). According to equation (42), when the system operates in a closed-loop configuration using the two control inputs u_1 and u_2 from (26), it achieves global exponential stability. This stability is attained by selecting positive definite diagonal gain matrices (k_p and k_d). Moreover, the errors $e_i, \dot{e}_i$, along with the additive disturbance inputs $\tilde{d}_1$ and $\tilde{d}_2$, are kept within bounded limits.

Proof of proposition 2

We introduce the following Lyapunov function:

$$V_l(e, \dot{e}) = \sum_{i=1}^2 \frac{1}{2} k_{p_i} e_i^2 + \sum_{i=1}^2 \frac{1}{2} \dot{e}_i^2 \quad (44)$$

The derivative of V_l can be expressed as:

$$\dot{V}_l(e, \dot{e}) = \sum_{i=1}^2 k_{p_i} e_i \dot{e}_i + \sum_{i=1}^2 \dot{e}_i \ddot{e}_i \quad (45)$$

Substituting Equation (42) into Equation (45) yields:

$$\dot{V}_l(e, \dot{e}) = \sum_{i=1}^2 k_{p_i} e_i \dot{e}_i + \sum_{i=1}^2 \dot{e}_i (-k_{d_i} \dot{e}_i - k_{p_i} e_i) \quad (46)$$

Simplifying results in:

$$\dot{V}_l(e, \dot{e}) = \sum_{i=1}^2 k_{p_i} e_i \dot{e}_i - \sum_{i=1}^2 k_{d_i} \dot{e}_i^2 - \sum_{i=1}^2 k_{p_i} e_i \dot{e}_i \quad (47)$$

Therefore, we ultimately arrive at:

$$\dot{V}_l(e, \dot{e}) = - \sum_{i=1}^2 k_{d_i} \dot{e}_i^2 \leq 0 \quad (48)$$

Then, by applying Barbalat's lemma [43], we can confirm the global stability of the system. After differentiating equation (48) obtain:

$$\ddot{V}_l(e, \dot{e}) = -2 \sum_{i=1}^2 k_{d_i} \dot{e}_i \ddot{e}_i \quad (49)$$

Since all gains k_p and k_d are positive, the state vector ζ, ζ^{ref} , and the additive disturbance $\tilde{d}_1, \tilde{d}_2$ are limited in scope. Similarly, the errors $e_i, \dot{e}_i$, and $\ddot{e}_i$ are also confined within certain bounds. Therefore, it can be concluded that $\dot{V}_l(e, \dot{e})$ is likewise bounded.

The derived $\dot{V}(e, \dot{e})_l$ exhibits uniform continuity. Given that the target trajectory $\zeta^{ref}, \dot{\zeta}^{ref}$ and $\ddot{\zeta}^{ref}$ are also uniformly continuous, it follows that the errors $e_i, \dot{e}_i$, and $\ddot{e}_i$ are uniformly continuous as well. According to Barbalat's lemma, $\lim_{t \rightarrow \infty} \dot{V}(e, \dot{e})_l = 0$, $\lim_{t \rightarrow \infty} e_i = 0$, and $\lim_{t \rightarrow \infty} \dot{e}_i = 0$. Under these conditions, the iTbot system achieves global asymptotic stability.

Figure 5 illustrates the global schematic of the proposed control system.

Figure 5 illustrates the global schematic diagram of the proposed control strategy.

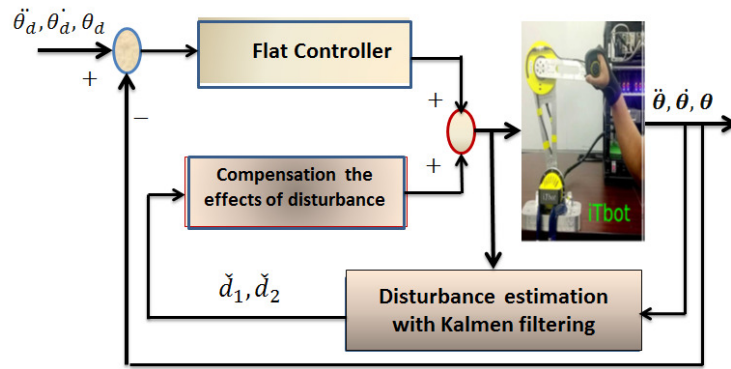


Figure 5. Global schematic of proposed control for iTbot robot

5. Simulation Setup and Performance Evaluation

In this section, we assess the effectiveness of the proposed control strategy in comparison with the conventional flatness-based control approach presented in [45]. The evaluation is conducted under three distinct simulation scenarios using the CAD model of the iTbot robot within the MATLAB/Simulink 2024a environment. The robot's dynamic model is assumed to be fully known, and its parameters are detailed in [44].

5.1. Scenario 1: Nominal Conditions

In the first scenario, the conventional flatness-based control (FBC) method [45] under nominal conditions, indicating that no external disturbances are present. The disturbance vector is set to zero: $\zeta_{\text{dis}} = \mathbf{0}$.

5.2. Scenario 2: External Disturbance

In the second scenario, we apply the same control method, but this time the system experiences external disturbances that vary with time, represented by:

$$\zeta_{\text{dis}} = \begin{bmatrix} 4 \sin(t) + 0.5 \sin(200\pi t) \\ \cos(3t) + 0.5 \sin(200\pi t) \end{bmatrix}. \quad (50)$$

5.3. Scenario 3: Proposed Control Strategy

In the third scenario, the disturbance profile established in Scenario 2 while implementing the proposed control strategy (refer to Equation (41)). This arrangement allows us to directly compare the performance of the proposed method with that of conventional approaches in the same conditions.

5.4. Trajectory and Control Parameters

The desired joint trajectory used in all simulations is defined as:

$$\theta_d(t) = \begin{bmatrix} 0.2 + 2 \sin(t) \\ 0.1 + 2 \cos(t) \end{bmatrix}.$$

The control gains are established as follows for all scenarios: $k_p = 10I_{2 \times 2}$, $k_d = 50I_{2 \times 2}$, where $I_{2 \times 2}$ represents the 2×2 identity matrix. The parameters for the Kalman filter applied in the observer design are summarized below.

$$P = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix}; \quad Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix};$$

$$R = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix};$$

5.5. Performance Metrics

To thoroughly assess and compare the effectiveness of the control strategies, we utilize a range of quantitative metrics: the Mean Absolute Error (MAE), the Standard Deviation of the Error (SDE), and the Standard Deviation of the Control (SDC). The metrics related to tracking errors are defined as follows:

$$\begin{cases} \text{MAE} = \frac{1}{N} \sum_{i=1}^N \|e(i)\|, \\ \text{SDE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\|e(i)\| - \text{MAE})^2}, \end{cases}$$

where $e(i)$ represents the tracking error at the i^{th} time step, and N is the total number of samples gathered.

5.6. Simulation Results

Figures 6–8 illustrate the trajectory tracking performance of different control strategies applied to the joint motion of the robotic system. Figure 6 presents the response obtained using a Flatness-Based Control (FBC) strategy under ideal conditions, where no external disturbances are introduced. The controller demonstrates satisfactory performance in tracking the desired joint trajectory, maintaining relatively smooth and accurate convergence. However, minor deviations observed at high-slope segments of the trajectory indicate a certain sensitivity of the control law to small model mismatches or measurement imperfections.

In contrast, Figure 7 evaluates the same flatness-based controller (FBC) under the influence of external disturbances (torque variations). As evident in the Figure 7, the control system exhibits increased tracking error, especially during dynamic transitions and at trajectory peaks. The presence of disturbances significantly deteriorates the controller's performance, leading to overshoots, oscillations, and noticeable lag between the desired and actual trajectories. This result highlights a key limitation of standard FBC: its lack of inherent robustness against unmodeled dynamics.

To address these shortcomings, Figure 8 introduces the proposed improved control strategy, which integrates a Kalman filter into the flatness-based framework. The Kalman filter is employed to estimate the true system states and attenuate the effect of external disturbances. The resulting trajectory tracking performance is markedly improved compared to the previous cases. The proposed FBC-Kalman controller effectively suppresses noise-induced fluctuations and ensures smoother, more accurate tracking with minimal error. The system responds reliably even during rapid trajectory transitions, showcasing both stability and robustness.

The comparative analysis highlights the benefits of integrating the Kalman filter technique with nonlinear control method. While the basic FBC performs well in ideal settings, it lacks robustness in disturbed environments. By incorporating a Kalman filter, the system's resilience and tracking accuracy are greatly improved, making this integrated approach more applicable to real-world situa-

tions, especially in rehabilitation contexts where effectively managing disturbances is essential. This conclusion is further supported by the performance metrics detailed in Table 3.

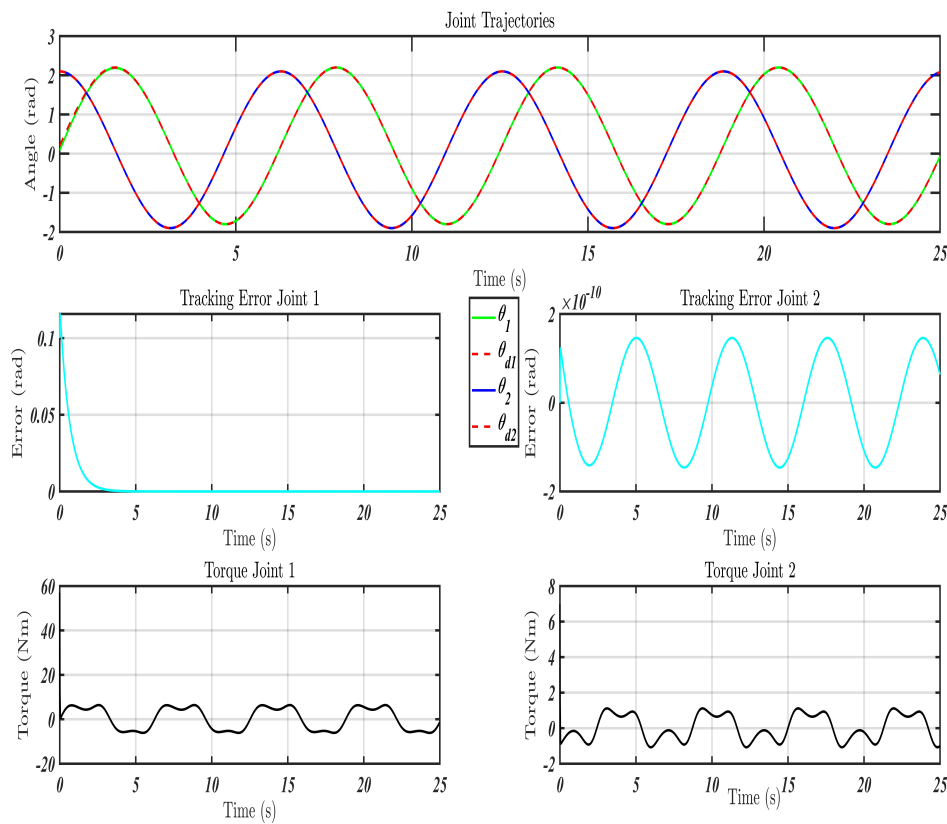


Figure 6. Joint Trajectory Tracking Using Flatness-Based Control [44]

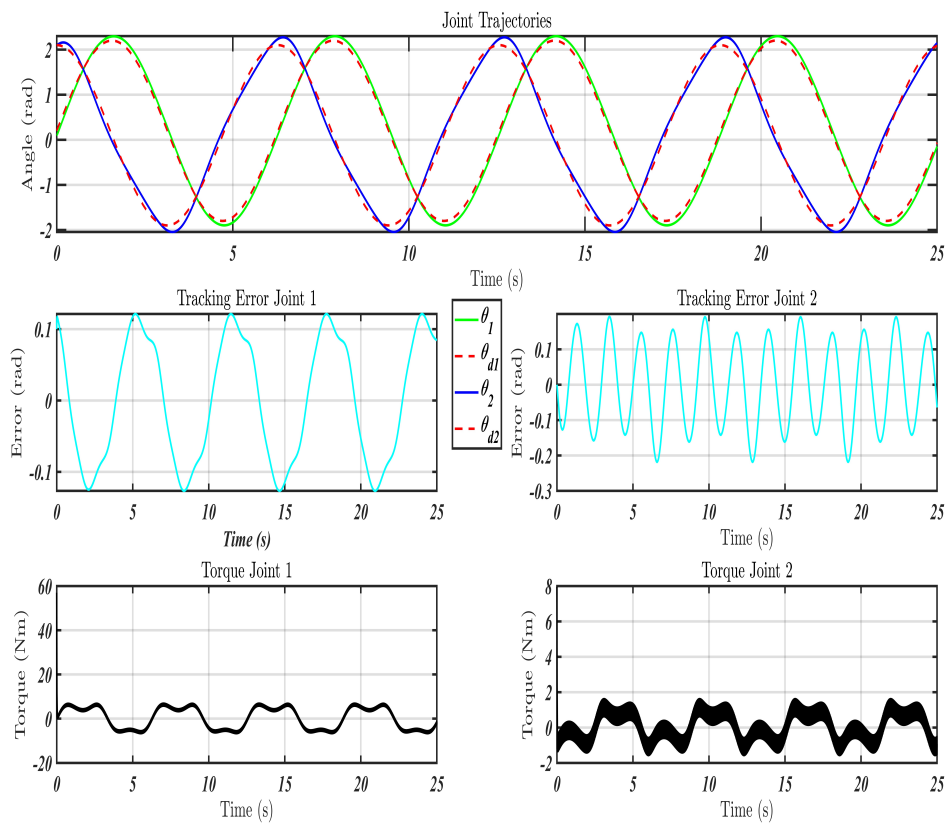


Figure 7. Joint Trajectory Tracking Using Flatness-Based Control under external disturbances

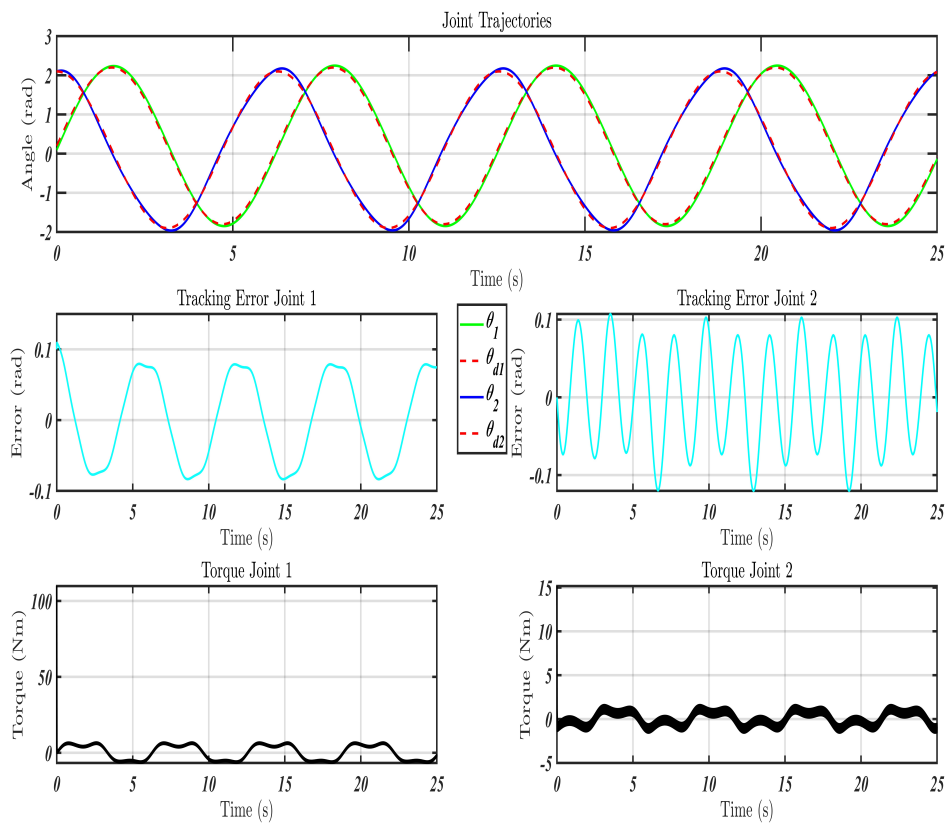


Figure 8. Joint Trajectory Tracking Using the proposed controller (41)

Table 3. Performance comparison across different scenarios

Performance	Scenario 1		Scenario 2		Scenario 3	
	Joint 1	Joint 2	Joint 1	Joint 2	Joint 1	Joint 2
MAE (rad)	0.0032	0.0010	0.0789	0.1055	0.0561	0.0559
SDE (rad)	0.0136	0.0015	0.0345	0.0545	0.0254	0.0298

6. Experimental Results

The iTbot rehabilitation robot system is built on a three-tier computing architecture. At the top level, a user interface (UI) on a personal computer enables control commands and real-time visualization of joint positions and force sensor data. The second level uses a National Instruments CompactRIO as a Real-Time (RT) controller, where LabVIEW RT applications handle trajectory planning, control algorithms, feedback processing, and data logging. At the lowest level, the system’s FPGA embedded within the CompactRIO executes fast-loop tasks every 50μs, including PI motor current control and position measurement from motor hall-sensor signals. The robot’s joints are actuated by Maxon EC-45 Flat BLDC motors (30 W and 70 W models) paired with Harmonic Drive gearboxes (100:1 ratio), and powered by ZB12A servo amplifiers. The built-in hall sensors serve both for motor commutation and precise joint feedback.

6.1. Experimental Evaluation

The proposed control approach (41) was experimentally tested under two scenarios to assess its performance in practical conditions. In the first scenario, a simple physiotherapy task was conducted. The subject’s hand was attached to the end-effector of the exoskeleton robot and guided through

a predefined trajectory. In this case, the external disturbance originated from the subject's natural interaction with the robot during motion execution. The second scenario involved a repetitive task, designed to evaluate the controller's robustness under continuous human-robot contact and dynamic parameter variations introduced by repeated movements. This setup mimics realistic rehabilitation activities where disturbances and nonlinearities may arise over time. The controller gains used in both experiments were: $k_p = 18I_{2 \times 2}$, $k_d = 75I_{2 \times 2}$. The Kalman filter parameters remained the same as specified in the previous section. Experimental results are presented in both joint space and Cartesian space, utilizing the Jacobian matrix to perform coordinate transformations between these domains.

6.2. Scenario 1: Simple Motion

In Scenario 1, which involves a guided physiotherapy task with moderate external interference from human interaction, the proposed control scheme demonstrates strong trajectory tracking capabilities in Cartesian space (Figure 9). The desired and actual end-effector paths show a close match, with minimal deviation along all three axes. The position error plots confirm this observation, maintaining sub-millimeter deviations in most instances and not exceeding ± 0.2 cm in any axis. This high level of precision highlights the controller's ability to reject mild disturbances and ensure smooth motion execution.

Joint space analysis (Figure 10) reinforces this performance. The actual joint angles accurately follow the reference trajectories for both joints, with maximum joint errors staying below ± 0.4 degrees. The smoothness of the joint torque signals, particularly in the first joint, suggests effective damping behavior and a lack of abrupt corrective actions. This behavior indicates good compliance with minimal overshooting, which is essential for safe human-robot interaction during rehabilitation exercises.

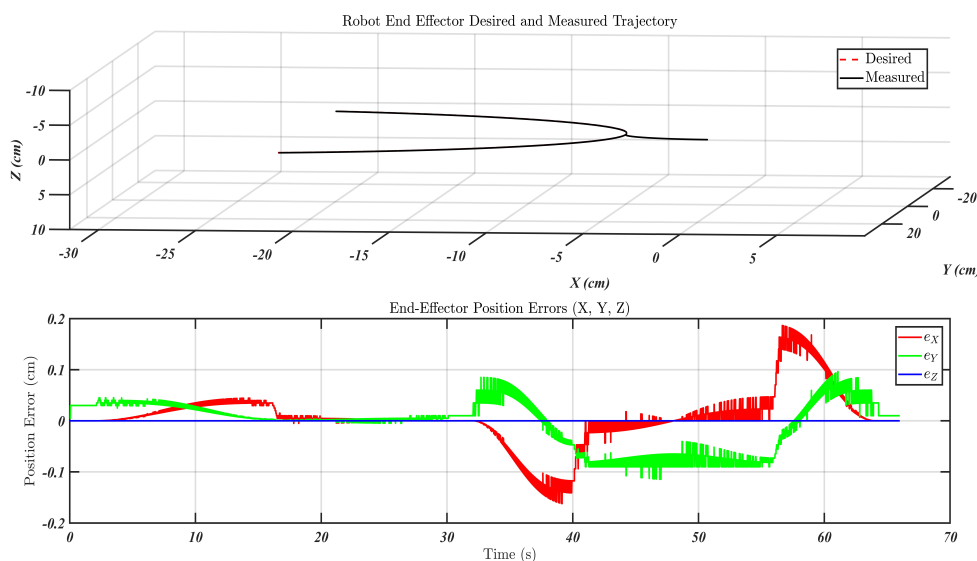


Figure 9. Cartesian Trajectory Tracking and Position Error in a simple Physiotherapy Task (Scenario 1)

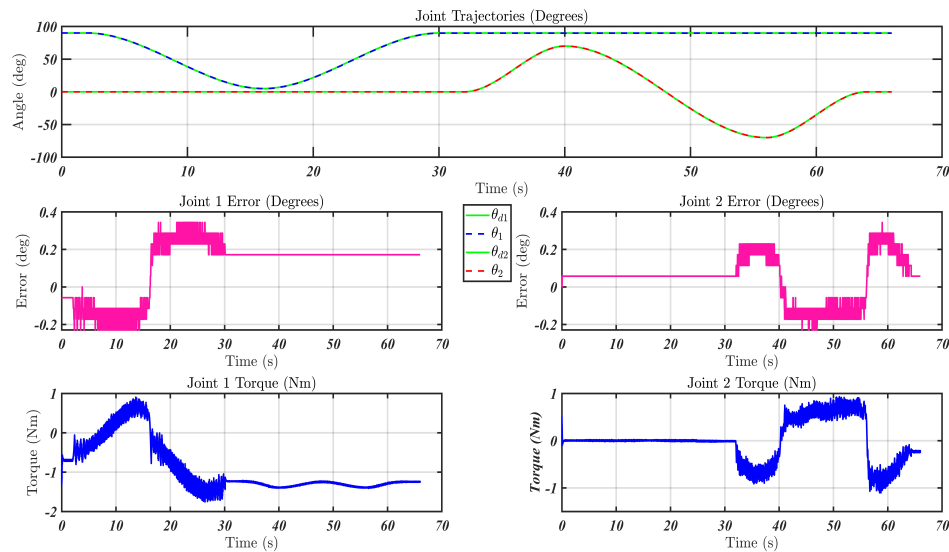


Figure 10. Joint Space Tracking, Error, and Torque Response in a simple Physiotherapy Task (Scenario 1)

6.3. Scenario 2: Repetitive Motion

In Scenario 2, involving repetitive tasks and dynamic variations due to continued human contact, the control system maintains stable performance albeit under more challenging conditions (Figures 11 and 12). Although Cartesian tracking remains precise, noticeable deviations, particularly along the X and Y axes, where peak position errors can increase by up to ± 0.8 cm. These errors reflect the increased difficulty of maintaining precision under time-varying disturbances and unmodeled dynamics. Nonetheless, the system recovers promptly from disturbances, suggesting robust adaptive behavior.

Joint space metrics in Figure 12 reveal higher joint angle errors compared to Scenario 1, with values reaching up to ± 2 degrees in Joint 1. The torque profiles exhibit more aggressive corrective actions, indicative of the controller's responsiveness to persistent disturbances. Despite these challenges, trajectory following remains stable and smooth, with no signs of instability or oscillations, which emphasizes the effectiveness of the proposed control law even under non-ideal conditions.

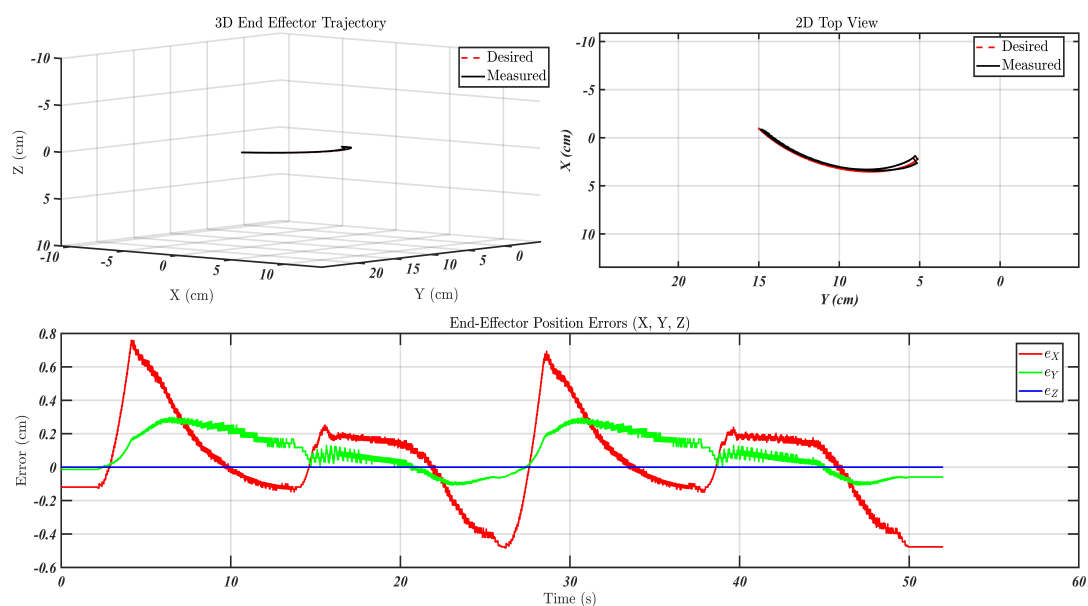


Figure 11. Cartesian Trajectory Tracking and Position Error in a Repetitive Task (Scenario 2)

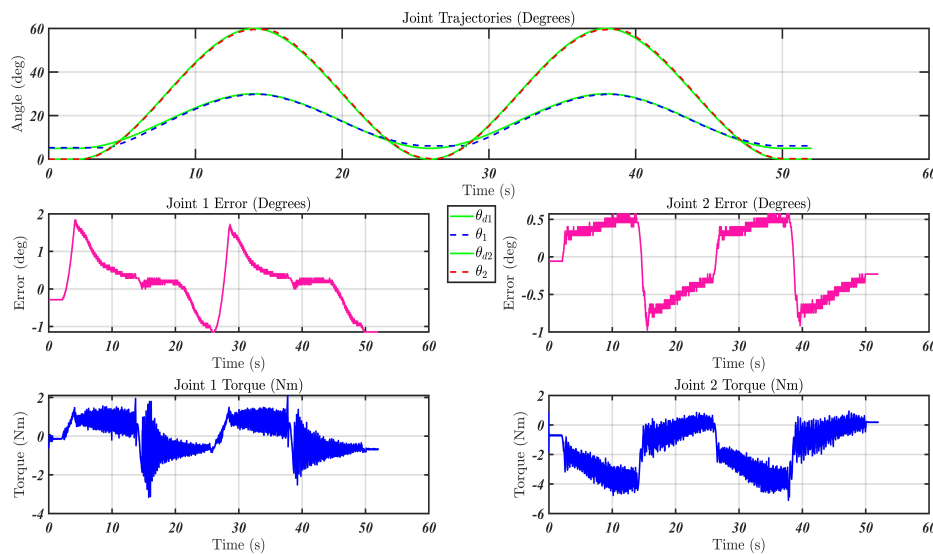


Figure 12. Joint Space Tracking, Error, and Torque Response in a Repetitive Task (Scenario 2)

7. Conclusions

This study tackles the vital challenge of achieving robust, adaptive, and safe control in end-effector-based rehabilitation robots, with a specific emphasis on the iTbot system designed for upper-limb therapy. Conventional control methods often struggle with dynamic uncertainties and the forces generated during human-robot interactions, limits their practical effectiveness. To address these challenges, the research presents an innovative flatness-based control framework, enhanced by a derivative-free Kalman filter (DFK) that facilitates real-time state estimation and disturbance compensation. This control scheme converts the nonlinear dynamics into a 0-flat canonical form, which simplifies trajectory tracking while maintaining computational efficiency. Both simulation and experimental results consistently illustrate improved tracking accuracy, compliance, and robustness compared to traditional approaches, even in the presence of significant external disturbances. These findings are significant for expanding robotic rehabilitation into both clinical and home settings, providing a promising avenue for more accessible and effective therapy options. However, the study is limited to a 2-DoF system and predefined disturbances, which may not fully capture the complexities encountered in real patient scenarios. Future research should aim to adapt the method for multi-DoF systems, incorporate human intent recognition, and validate its performance through clinical trials.

Author Contributions: All authors contributed equally to this work, including system design, implementation, validation, and manuscript development.

Funding: This research received no external funding.

Informed Consent Statement: This study exclusively involved healthy adult participants interacting with robotic systems. According to U.S. federal regulations (45 CFR 46) and institutional policies, this type of human-robot interaction study with healthy volunteers does not constitute human subjects research requiring IRB approval. All participants provided verbal acknowledgment of the experimental procedures.

Data Availability Statement: Research materials supporting this work are available upon request to the corresponding author.

Acknowledgments: The authors gratefully acknowledge the technical support and contributions from the Bio-robotics Laboratory team at the University of Wisconsin-Milwaukee and the Intelligent Robotics & Manufacturing team at King Fahd University of Petroleum and Minerals, particularly for their assistance with hardware integration, experimental validation, and code verification.

Conflicts of Interest: The authors declare no conflict of interest regarding the publication of this work.

References

1. Yu, Shuangyue, et al. Quasi-Direct Drive Actuation for a Lightweight Hip Exoskeleton with High Backdrivability and High Bandwidth. *IEEE/ASME Transactions on Mechatronics* (2020).
2. Teng, Long, Muhammad Ahsan Gull, and Shaoping Bai. PD Based Fuzzy Sliding Mode Control of A Wheelchair Exoskeleton Robot. *IEEE/ASME Transactions on Mechatronics* (2020).
3. S. Balasubramanian, R. Wei, M. Perez, B. Shepard, E. Koeneman, J. Koeneman, et al., RUPERT: An exoskeleton robot for assisting rehabilitation of arm functions, in *Virtual Rehabilitation*, 2008, 2008, pp. 163-167.
4. Khan, M.M.R.; Ahmed, T.; Pallares, J.R.H.; Islam, M.R.; Brahmi, B.; Rahman, M.H. Development of A Desktop-mounted Rehabilitation Robot For Upper Extremities. In *Proceedings of the International Conference on Industrial Mechanical Engineering and Operations Management Dhaka, Bangladesh*, 26–27 December 2021.
5. Cao, W.; Chen, C.; Hu, H.; Fang, K.; Wu, X. Effect of hip assistance modes on metabolic cost of walking with a soft exoskeleton. *IEEE Trans. Autom. Sci. Eng.* 2020, 18, 426–436.
6. Loureiro, R.C.; Harwin, W.S.; Nagai, K.; Johnson, M. Advances in upper limb stroke rehabilitation: A technology push. *Med. Biol. Eng. Comput.* 2011, 49, 1103 – 1118.
7. Nef, T.; Guidali, M.; Riener, R. ARMin III–arm therapy exoskeleton with an ergonomic shoulder actuation. *Appl. Bionics Biomech.* 2009, 6, 127–142.
8. Kim, B.; Deshpande, A.D. An upper-body rehabilitation exoskeleton Harmony with an anatomical shoulder mechanism: Design, modeling, control, and performance evaluation. *Int. J. Robot. Res.* 2017, 36, 414–435.
9. Liu, L.; Shi, Y.Y.; Xie, L. A novel multi-dof exoskeleton robot for upper limb rehabilitation. *J. Mech. Med. Biol.* 2016, 16, 1640023.
10. Pignolo, L.; Dolce, G.; Basta, G.; Lucca, L.; Serra, S.; Sannita, W. Upper limb rehabilitation after stroke: ARAMIS a “robomechatronic” innovative approach and prototype. In *Proceedings of the 2012 4th IEEE RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, Rome, Italy, 24–27 June 2012; pp. 1410–1414.
11. 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.
12. Loureiro, R.C.; Harwin, W.S.; Nagai, K.; Johnson, M. Advances in upper limb stroke rehabilitation: A technology push. *Med. Biol. Eng. Comput.* 2011, 49, 1103–1118.
13. Kim, B.; Deshpande, A.D. An upper-body rehabilitation exoskeleton Harmony with an anatomical shoulder mechanism: Design, modeling, control, and performance evaluation. *Int. J. Robot. Res.* 2017, 36, 414–435.
14. B. Brahmi, M. Saad, M. H. Rahman and C. Ochoa-Luna, “Cartesian trajectory tracking of a 7-DOF exoskeleton robot based on human inverse kinematics,” *IEEE Trans. Syst. Man Cybernetics: Syst. PP*(99), 1–12 (2017).
15. W. Chen, S. S. Ge, J. Wu and M. Gong, “Globally stable adaptive backstepping neural network control for uncertain strict-feedback systems with tracking accuracy known a priori,” *IEEE Trans. Neural Networks Learning Syst.* 26(9), 1842–1854 (2015).
16. B. K. Yoo and W. C. Ham, “Adaptive control of robot manipulator using fuzzy compensator,” *IEEE Trans. Fuzzy Syst.* 8(2), 186–199 (2000).
17. J. J. Craig, *Introduction to Robotics: Mechanics and Control* (Pearson Prentice Hall Upper Saddle River, 2005).
18. B. Siciliano, L. Sciacivco, L. Villani and G. Oriolo, *Kinematics* (Springer, 2009).
19. S. Ferrer, C. Ochoa-Luna, M. Rahman, M. Saad and P. Archambault, “HELIOS: The Human Machine Interface forMARSERobot,” *Proceedings of the 6th International Conference on Human System Interaction (HSI)*, (IEEE, 2013) (2013) pp. 117–122.
20. C. O. Luna, M. H. Rahman, M. Saad, P. Archambault and W.-H. Zhu, “Virtual decomposition control of an exoskeleton robot arm,” *Robotica* 34(07), 1587–1609 (2016).
21. M. H. Rahman, T. Kittel-Ouimet, M. Saad, J.-P. Kenn and P. S. Archambault, “Dynamic modeling and evaluation of a robotic exoskeleton for upper-limb rehabilitation,” *Int. J. Inform. Acquisition* 8(01), 83–102 (2011).
22. P. Chen, C.-W. Chen, and W.-L. Chiang, “GA-based modified adaptive fuzzy sliding mode controller for nonlinear systems,” *Expert Systems with Applications*, vol. 36, pp. 5872-5879, 2009.

23. Brahim Brahmi., Maarouf Saad, Cristobal Ochoa Luna, Philippe S. Archambault, and Mohammad H. Rahman, Passive and active rehabilitation control of human upper-limb exoskeleton robot with dynamic uncertainties. *Robotica* (2018) volume 36, pp. 1757–1779. Cambridge University Press 2018.
24. Rigatos, G.G.: Extended Kalman and particle Filtering for sensor fusion in motion control of mobile robots. *Math. Comput. Simul.*, Elsevier 81(3), 590–607 (2010)
25. Rigatos, G., Zhang, Q.: Fuzzy Model Validation using the Local Statistical Approach, Publication Interne IRISA No 1417. Rennes, France (2001)
26. Fliess, M., Levine, J., Martin, P., and Rouchon, P., 1995, "Flatness and Defect of Non-Linear Systems: Introductory Theory and Examples," *Int. J. Control*, 61(6), pp. 1327–1361.
27. M. Fliess, J. Levine, P. Martin, and P. Rouchon. On differentially flat nonlinear systems. *Proc. of the IFAC-Symposium on nonlinear control systems*, pp. 159-163, 1992.
28. M. Fliess, J. Levine, P. Martin, and P. Rouchon. Flatness defect of nonlinear systems introductory theory and examples. *Internat. J. Control*, 61: pp 1327-1361, 1995.
29. Martin, Ph., M. Murray and P. Rouchon. Flat systems In: *Plenary Lectures and Mini- Courses. 4th European Control Conference ECC 97, Brussels Belgium (M. Gevers G. Bastin, Ed.).*1997, pp. 211-264.
30. H. Sira Ramirez, S.K. Agrawal. *Differentially flat systems*. Marcel Dekker, New York, 2004.
31. S. Bououden, D. Boutat, G. Zheng b , J.-P. Barbot, F. Kratz. A triangular canonical form for a class of 0-flat nonlinear systems. *International Journal of Control*, Vol. 84, No. 2, 261269, February 2011.
32. Chen, Y.; Fan, J.; Zhu, Y.; Zhao, J.; Cai, H. A passively safe cable driven upper limb rehabilitation exoskeleton. *Technol. Health Care* 2015, 23, S197–S202.
33. Xiao, F.; Gao, Y.; Wang, Y.; Zhu, Y.; Zhao, J. Design of a wearable cable-driven upper limb exoskeleton based on epicyclic gear trains structure. *Technol. Health Care* 2017, 25, 3–11.
34. Luh, J.Y.; Walker, M.W.; Paul, R.P. On-line computational scheme for mechanical manipulators. *J. Dyn. Syst. Meas. Control* 1980, 102, 69–76.
35. Martin, P., Murray, R., and Rouchon, P. (1997), Flat Systems, in *European Control Conference*, pp. 211 264.
36. P.S. Pereira da Silva, Flatness of nonlinear control systems a Cartan–Kähler approach, in: *Proc. Mathematical Theory of Networks et Systems, MTNS 2000, Perpignan, June 19–23, 2000*, pp. 1–10.
37. A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, 3rd edition, 1995.
38. Rigatos, G.G., Tzafestas, S.G.: Extended Kalman filtering for fuzzy modelling and multisensor fusion. *Math. Comput. Model. Dyn. Syst.*, Taylor-Francis 13, 251–266 (2007)
39. Rigatos, G., Siano, P., Zervos, N.: Derivative-free nonlinear Kalman filtering for PMSG sensorless control. In: Habib, M. (ed.) *Mechatronics Engineering: Research Development and Education*, Wiley (2012).
40. Rigatos, G., Siano, P.: A derivative-free extended information filtering approach for sensorless control of nonlinear systems. In: *MASCOT 2010. IMACS Workshop on Scientific Computation*, Italian Institute for Calculus Applications, Gran Canaria, Spain, Oct 2010.
41. Chen, W.H., Ballance, D.J., Gawthrop, P.J., Reilly, J.O.: A nonlinear disturbance observer for robotic manipulators. *IEEE Trans. Industr. Electron.* 47(4), 932–938 (2000)
42. Cortesao, R.: On Kalman active observers. *J. Intell. Robot. Syst.*, Springer 48(2), 131–155 (2006).
43. Slotine, J.J.E and W.Li, *Applied nonlinear control*, Prentia Hall, 1991.
44. Khan, Md Mahafuzur Rahaman, Asif Al Zubayer Swapnil, Tanvir Ahmed, Md Mahbubur Rahman, Md Rasedul Islam, Brahim Brahmi, Raouf Fareh, and Mohammad Habibur Rahman. "Development of an end-effector type therapeutic robot with sliding mode control for upper-limb rehabilitation." *Robotics* 11, no. 5 (2022): 98.
45. Brahmi, Brahim, Tanvir Ahmed, Ibrahim El Bojairami, Asif Al Zubayer Swapnil, Md Assad-Uz-Zaman, Katie Schultz, Erin McGonigle, and Mohammad Habibur Rahman. "Flatness based control of a novel smart exoskeleton robot." *IEEE/ASME Transactions on Mechatronics* 27, no. 2 (2021): 974-984.

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.