AIMS Mathematics, 10(10): 23738–23772.
DOI: 10.3934/math.20251055
Received: 16 July 2025
Revised: 05 October 2025
Accepted: 11 October 2025
Published: 20 October 2025

https://www.aimspress.com/journal/Math

Research article

Ultra-local model-based predefined-time assist-as-needed control for upper limb patient-exoskeleton system under input and performance constraints

Honglin Xie and Yangchun Wei*

School of Data and Computer Science, Shandong Women's University, 2399 Daxuelu, Jinan, 250300, Shandong, China

* **Correspondence:** Email: weiyangchun@sdwu.edu.cn.

Abstract: In this work, an ultra-local model-based predefined-time assist-as-needed controller (UPTAC) is presented for a upper limb patient-exoskeleton system (ULPES) under input and performance constraints. The designed UPTAC has a dual-loop control architecture. The outer impedance sub-control loop includes task performance function and impedance controller, which dynamically adjust the speed of rehabilitation exercises and obtain the desired assistive torque. Next, under the ultra-local model-based control framework, an inner torque sub-control loop was designed by combining barrier Lyapunov function and performance constraints, thereby achieving predefined time convergence and reducing the complexity of controller design. Finally, the effectiveness of UPTAC was verified through simulation.

Keywords: ultra-local model-based control; predefined-time control; assist-as-needed control; barrier Lyapunov function
Mathematics Subject Classification: 93B52, 93C40, 93C85, 93D05

1. Introduction

The global demographic shift toward aging populations has elevated stroke to a predominant neurological disorder, frequently resulting in persistent motor dysfunction [1, 2]. Contemporary rehabilitation paradigms increasingly incorporate robotic exoskeletons, yet critical challenges persist in the control of upper limb patient-exoskeleton systems (ULPES), particularly regarding (i) precision implementation of assist-as-needed (AAN) control, and (ii) resolving time-varying parameter identification.

Early rehabilitation exoskeletons mainly utilize position control strategies, but this approach often overlooks the patient's involvement during the training process [3]. To address this limitation, AAN control methods have been developed, which operate by adjusting torque, being fundamentally

different from traditional position control [4]. The AAN approach dynamically adjusts assistive torque based on real-time assessment of the patient's movement capability, thereby maintaining therapeutic efficacy while actively promoting neuromuscular recovery through preserved volitional movement participation [5]. In [6], an AAN control via a novel stiffness-based mapping framework was implement, which can dynamically adapt both the rehabilitation trajectory and assistive force of the robotic system based on real-time assessment of the affected limb's movement capability. In [7], a trained machine learning model was employed to characterize patient muscle strength, enabling real-time adaptation of exoskeleton drive velocity based on continuous muscular capability assessment. These advancements collectively demonstrate the transition from rigid position control to adaptive, patient-centered AAN strategies in rehabilitation exoskeletons.

The exoskeleton system must maintain high control precision while achieving AAN to deliver accurate assistive torque for optimal rehabilitation training. Barrier Lyapunov function (BLF) can constrain variables in the system, such as tracking errors [8], outputs [9], and states [10], within certain boundaries. Therefore, BLF-based control strategies can limit tracking errors to small intervals, achieving high-precision control. In order to address constraint problems in different situations, BLF has evolved from the initial symmetric log-type BLF [11] to tan-type BLF [12], asymmetric BLF [13], time-varying BLF [14], and integral BLF [15], among others. In [16], a novel asymmetric BLF is designed based on translation transformation, which is smooth and has a high-order derivative with fixed-time convergence. In [17], a recursively constructed adaptive control scheme was presented by combining BLF and funnel control techniques, with the BLF component designed to mitigate effects of unmodeled dynamics and improve transient performance. Due to these advantageous properties, BLF has been widely adopted in robotic systems [18], multi-agent systems [19], and marine surface vessels [20] to ensure both operational safety and control precision. However, the existence of random initial values may cause the constrained variables to exceed the preset boundaries, which can have adverse effects on control performance. To solve the problem of random initial values, shifting function is used to convert tracking errors into shifting errors [21], thereby reducing the impact of random initial values. In [22], the influence of initial conditions on predefined-time convergence was eliminated by introducing auxiliary functions and K^1-sliding surfaces. In addition, infinite initial value performance functions have been proposed in [23], which have infinite initial values, allowing them to remain within the performance function even if the initial values are random. Most existing BLF-based control methods rely on model-dependent designs. For human-exoskeleton rehabilitation systems involving active patient participation, accurate parameter identification becomes particularly challenging. This necessitates the development of BLF-based controllers that do not rely on precise models.

In addition to BLF-based control methods, both predefined-time control [24] and prescribed performance control [25] can improve control accuracy. For predefined-time control, this method can ensure that the system convergence time has a fixed upper limit of a constant, which can be explicitly determined by the user [24]. Predefined-time control generally forces the system to converge within a predefined time by constructing Lyapunov functions on specific forms [26] or sliding mode control laws [27]. Steady-state error is also an important indicator, and prescribed performance control can achieve quantifiable transient and steady-state performance [25]. Its essence is to transform error constraints into core indicators of controller design, requiring errors to satisfy prescribed dynamic boundaries. In the controller design process, constrained errors are generally converted into unconstrained variables through an error conversion function, and the controller design

ensures that the conversion error is bounded, thereby achieving prescribed performance control [28]. In addition, prescribed performance control can also be achieved through BLF [29]. However, most predefined-time control and prescribed performance control are model-based control methods, and when an accurate model cannot be obtained or model parameter identification is inaccurate, the control effect will be affected.

Ultra-local model-based model-free control effectively overcomes the limitations of conventional control methods that require precise mathematical models. This approach simplifies the original system into a second-order ultra-local model (ULM) [30]. All subsequent controller design and stability analysis are then performed based on this simplified ULM representation. Owing to its parsimonious mathematical structure, the ULM demonstrates remarkable compatibility with various advanced control strategies, including PID control [31], sliding mode control [32], prescribed performance control [33], and prescribed time control [34]. This versatility has led to successful deployments across multiple engineering systems, particularly in exoskeletons [35], manipulators [34], car active suspension systems [36], UAV [37], etc.

Building upon these research challenges, an ULM-based predefined-time assist-as-needed controller (UPTAC) is proposed for ULPES, explicitly addressing both input and performance constraints. The key advancements beyond existing approaches include:

(i) Compared with our previous work [38], this study incorporates a task performance function that dynamically adjusts rehabilitation training intensity based on the patient's recovery status. For patients with limited movement capability, the duration of each training cycle is extended to reduce exercise speed; as the patient's movement capability improves, the circulation time is shortened to gradually increase the speed. This ensures the safety of patients and can stimulate their enthusiasm for rehabilitation training.

(ii) BLF is introduced, and special performance constraints are designed to indirectly achieve predefined-time convergence, allowing the tracking error to converge to a small interval within a predefined time. The constraints employ an exponential term that simultaneously enables predefined-time convergence and achieves a smooth transition of performance constraints.

(iii) Unlike conventional BLF-based controllers, the proposed UPTAC method is designed and analyzed using the ULM framework, which significantly reduces both dependence on precise mathematical modeling and controller design complexity.

This paper is structured into five main sections. Section 2 formulates the dynamic model of the ULPES and gives the control objectives. The detailed development of the UPTAC controller is presented in Section 3. Sections 4 and 5 discuss the simulation results and conclusions, respectively.

2. Dynamic modeling of ULPES and preliminaries

To achieve effective assist-as-needed control while satisfying both input and performance constraints, establishing a dynamic model of the ULPES becomes essential. Figure 1 illustrates the 2-DOF mechanical configuration of this rehabilitation system.

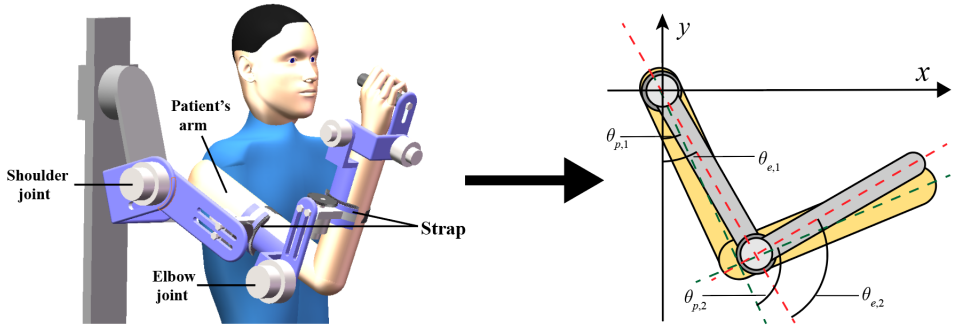


Figure 1. Schematic diagram of ULPES.

As shown in Figure 1, the 2-DOF exoskeleton comprises a shoulder joint and an elbow joint, with their angles denoted as $\theta_{e,1}$ and $\theta_{e,2}$, respectively. The patient's shoulder and elbow joint angles are represented by $\theta_{p,1}$ and $\theta_{p,2}$. The patient's upper limb is connected to the exoskeleton via elastic straps, whose tension provides assistive force to the user.

Applying Hamilton's principle [39] through systematic energy analysis yields the dynamic model for the 2-DOF ULPES:

$$\mathbf{M}_p(\theta_p)\ddot{\theta}_p + \mathbf{C}_p(\theta_p, \dot{\theta}_p)\dot{\theta}_p + \mathbf{G}_p(\theta_p) = \tau_s + \tau_p, \tag{2.1}$$

$$\mathbf{M}_e(\theta_e)\ddot{\theta}_e + \mathbf{C}_e(\theta_e, \dot{\theta}_e)\dot{\theta}_e + \mathbf{G}_e(\theta_e) + \tau_s = \tau_m, \tag{2.2}$$

$$\tau_s = \mathbf{K}_s(\theta_e - \theta_p) + \mathbf{B}_s(\dot{\theta}_e - \dot{\theta}_p), \tag{2.3}$$

where $\tau_s \in \mathbb{R}^2$ is the torque generated by straps, $\tau_p \in \mathbb{R}^2$ denotes the torque generated by patient, $\theta_e = [\theta_{e,1}; \theta_{e,2}] \in \mathbb{R}^2$ is the angle of the exoskeleton, $\theta_p = [\theta_{p,1}; \theta_{p,2}] \in \mathbb{R}^2$ is the angle of the patient's joint, $\tau_m = [\tau_{m,1}; \tau_{m,2}] \in \mathbb{R}^2$ denotes the torque of the exoskeleton, $\mathbf{K}_s = \text{diag}\{k_{s,1}, k_{s,2}\} \in \mathbb{R}^{2\times 2}$ and $\mathbf{B}_s = \text{diag}\{b_{s,1}, b_{s,2}\} \in \mathbb{R}^{2\times 2}$ demonstrate the stiffness and damping of the straps, and the matrix elements are given in our previous work [38].

Addressing both patient safety concerns and the physical torque limits of the motors, input constraints are considered as follows:

$$\tau_{m,i} = \begin{cases} -\mu_{\tau,i} & \tau_{c,i} < -\mu_{\tau,i}, \\ \tau_{c,i} & -\mu_{\tau,i} \le \tau_{c,i} \le \mu_{\tau,i}, \\ \mu_{\tau,i} & \mu_{\tau,i} < \tau_{c,i}, \end{cases} \tag{2.4}$$

where $\tau_{c,i}$ denotes the original control signal obtained by the controller, without accounting for input constraints, and $\mu_{\tau,i} > 0$ is a positive constant representing the boundary of input constraints.

Remark 2.1. *In control systems, input constraints are a critical consideration, especially in rehabilitation exoskeleton systems. This necessity stems from two basic requirements: (i) physical torque limitation of the driving motor, and (ii) the need for patient safety. The selection of input constraints cannot be too small, and the input torque of the driving motor needs to meet the actual rehabilitation task requirements. If the input constraints are selected too small, it is likely that the required rehabilitation task cannot be completed, and the stability of the system will also be affected when disturbances occur. Therefore, the setting of input constraints cannot be too small and needs to be adjusted according to the actual needs of the task.*

This study focuses on the ULPES under input constraints, proposing a ULM-based predefined-time assist-as-needed controller (UPTAC) to achieve the following control objectives: (i) calculate the required assistive torque based on the patient's joint angles and angular velocities, and dynamically adjust the cycle duration of rehabilitation movements according to the assistive torque; (ii) ensure that the patient's upper limb $\mathbf{P}_p = (\mathbf{J}^T(\theta_p))^{-1}\theta_p \in \mathbb{R}^2$ tracks the desired rehabilitation trajectory $\mathbf{P}_d \in \mathbb{R}^2$ within a predefined time interval, where $\mathbf{J}^T(\theta_p) \in \mathbb{R}^{2\times 2}$ is the Jacobian matrix.

3. UPTAC design and stability analysis

To address the control challenges of the ULPES under input and performance constraints, an UPTAC method is proposed with dual-loop architecture (Figure 2). Section 3.1 presents the outer impedance loop design. The inner torque loop synthesis combining admittance control and ultra-local modeling is provided in Section 3.2. The complete stability proof is established in Section 3.3. The design process of the proposed UPTAC method is shown in Figure 2.

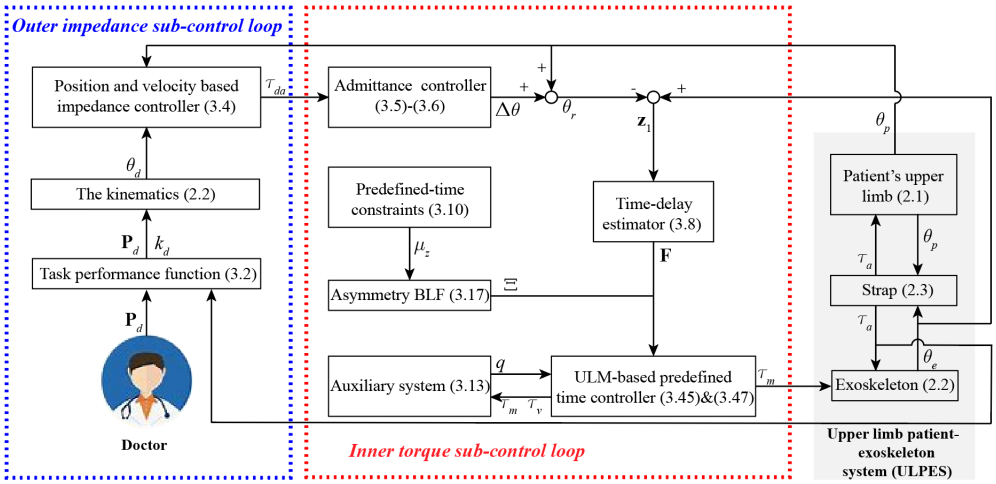


Figure 2. The control structure of the UPTAC.

3.1. Outer impedance sub-control loop design

To determine the desired assistive torque to be provided by the exoskeleton and to enable its accurate delivery, the outer-loop force control framework is designed as follows: First, the rehabilitation trajectory of the patient's upper limb is defined. Subsequently, to actively engage the patient during training, a task performance function is established to dynamically adjust the rehabilitation speed according to the patient's recovery progress. Then, an impedance controller is designed to compute the desired assistive torque for the patient's upper limb.

3.1.1. Desired trajectory and task performance function design

For the desired trajectory $\mathbf{P}_d = [P_{d,X}; P_{d,Y}] \in \mathbb{R}^2$, this study adopts a circular rehabilitation path, formulated as follows:

$$\begin{cases} P_{d,X} = p_{ox} + r_d \cos(2\pi t/k_d), \\ P_{d,Y} = p_{oy} + r_d \sin(2\pi t/k_d), \end{cases} \tag{3.1}$$

where (p_{ox}, p_{oy}) denotes the center position of the circular trajectory in Cartesian coordinate system, r_d represents the trajectory radius, and parameter k_d is a time-scaling factor for cycle duration.

In rehabilitation training, increasing movement speed can enhance patient motivation. However, speed adjustment must be carefully regulated based on the patient's recovery progress. Excessive speed may cause injury if the patient's movement capability is insufficient. Therefore, this study adopts the following adaptive speed control strategy: For patients with limited movement capability, the duration of each training cycle should be extended to reduce movement speed; as the patient's movement capability improves, the cycle duration should be shortened to progressively increase speed.

In this study, one complete revolution of the circular trajectory constitutes a rehabilitation cycle for the patient. Thus, the movement speed can be regulated by adjusting parameter k_d, with the following adaptive strategy:

$$k_d = k_{max} - (k_{max} - k_{min})(1 - \tanh(\varsigma_1(\bar{\tau}_a - \varsigma_2))), \tag{3.2}$$

where k_{min} and k_{max} denote the minimum and maximum bounds of k_d, respectively, $\bar{\tau}_a$ represents the average assistive torque from the previous cycle, and ς_1 and ς_2 are positive constant parameters.

3.1.2. Desired assistive torque and impedance controller design

Based on established physiological models [40], the torque of patient's upper limb is given as follows

$$\tau_p = k_m(\mathbf{K}_p(\theta_d - \theta_p) + \mathbf{B}_p(\dot{\theta}_d - \dot{\theta}_p)), \tag{3.3}$$

where $\mathbf{B}_p = \text{diag}\{b_{p,1}, b_{p,2}\} \in \mathbb{R}^{2\times2}$ and $\mathbf{K}_p = \text{diag}\{k_{p,1}, k_{p,2}\} \in \mathbb{R}^{2\times2}$ are the damping and stiffness parameters of patient, parameter $k_m \in (0, 1]$ quantifies the patient's movement capability, where decreasing values indicate progressively weaker muscular capability, $k_m = 1$ corresponds to unimpaired physiological movement capability, $\theta_d = [\theta_{d,1}; \theta_{d,2}] = \mathbf{J}^T(\theta_p)\mathbf{P}_d \in \mathbb{R}^2$ is the desired trajectory of patient, and $\mathbf{J}^T(\theta_p) \in \mathbb{R}^{2\times2}$ is the Jacobian matrix.

Patients' diminished movement capacity necessitates exoskeleton-delivered assistive torque. Following the impedance control methodology in [41], the desired assistive torque $\tau_{da} = [\tau_{da,1}; \tau_{da,2}] \in \mathbb{R}^2$ can be derived as:

$$\tau_{da} = K_a(\theta_d - \theta_p) + B_a(\dot{\theta}_d - \dot{\theta}_p), \tag{3.4}$$

where B_a and K_a are the damping and stiffness parameters.

Thus, based on the desired trajectory θ_d and the patient's actual trajectory θ_p, the desired assistive torque τ_{da} is derived through the impedance control algorithm (3.4). Furthermore, an adaptation strategy (3.2) is incorporated to adjust the rehabilitation movement speed according to the patient's recovery progress, thereby enhancing patient motivation. The implementation of τ_{da} for precise tracking of $\mathbf{P}_d$ will be detailed in the next subsection.

3.2. Inner torque sub-control loop design

To ensure the exoskeleton delivers the necessary assistive torque to the patient, the control framework proceeds as follows: First, a reference trajectory $\theta_r = [\theta_{r,1}; \theta_{r,2}] \in \mathbb{R}^2$ for the exoskeleton is generated through an admittance control approach, incorporating both the patient's trajectory θ_p and the desired assistive torque τ_{da}. Subsequently, an ULM-based predefined-time controller is developed

based on ULM, integrating BLF with backstepping techniques to ensure that the patient's upper-limb motion can precisely track the desired rehabilitation trajectory $\mathbf{P}_d$.

3.2.1. Admittance controller design

The admittance control algorithm determines the exoskeleton's reference trajectory. Subsequent assistive torque application occurs via strap mechanisms, where the resulting assistive torque profile follows the dynamics described in Eq (2.3).

The exoskeleton's reference trajectory θ_r is generated through admittance control principles according to the following formulation:

$$\theta_r = \theta_p + \Delta\theta, \tag{3.5}$$

$$\begin{cases} \Delta\theta_1(s) = \frac{1}{B_{s,1}s+K_{s,1}}\tau_{da,1}(s), \\ \Delta\theta_2(s) = \frac{1}{B_{s,2}s+K_{s,2}}\tau_{da,2}(s), \end{cases} \tag{3.6}$$

with $\tau_{da} = [\tau_{da,1}; \tau_{da,2}] \in \mathbb{R}^2$ and $\Delta\theta = [\Delta\theta_1; \Delta\theta_2] \in \mathbb{R}^2$.

3.2.2. Ultra-local model (ULM) design

While backstepping-based controllers traditionally depend on precise system parameters, practical identification inaccuracies frequently compromise their performance. To overcome this limitation, ULM is developed to approximate the system (2.3) [30, 42]:

$$\ddot{\theta}_e = \mathbf{F} + \alpha\tau_m, \tag{3.7}$$

where $\mathbf{F} = [F_1; F_2] \in \mathbb{R}^2$ indicates an unknown lumped term that encompasses all the dynamics characteristics, and $\alpha \in \mathbb{R}$ expresses a constant input gain.

Remark 3.1. *Compared to first-order ULM, second-order ULM can fully capture the dynamic characteristics (including position and velocity) of the upper limb exoskeleton system. Second, compared to higher-order ULM, second-order models achieve a good balance between computational complexity and model accuracy. Moreover, second-order ULM has been proven to be sufficient to describe the exoskeleton patient interaction dynamics in experimental applications [1, 35].*

There is an assumption to be made for the ULM (3.7).

Assumption 3.1. *Physical limitations of the exoskeleton system impose strict bounds on joint kinematics (angular displacement, velocity, acceleration) and actuator torque outputs. Consequently, the dynamic term* $\boldsymbol{F}$ *that encapsulates all unmodeled system properties must be bounded.*

Remark 3.2. *In this work, the input constraints ensure that the input torque of the exoskeleton is limited. Then, in the upper limb patient-exoskeleton system (ULPES), the joint torque of the patient is also limited, so the displacement, velocity, and acceleration values of the exoskeleton and patient joints are bounded. The unknown variable* $\mathbf{F}$ *can be expressed as*

$$\mathbf{F} = \mathbf{M}_e^{-1}(\tau_m - \mathbf{C}_e\dot{\theta}_e - \mathbf{K}_s(\theta_e - \theta_p) + \mathbf{B}_s(\dot{\theta}_e - \dot{\theta}_p)) - \alpha\tau_m.$$

It can be seen that the value of $\mathbf{F}$ *is also bounded, so Assumption 3.1 is reasonable.*

Based on the time-delay estimator (TDE) [43], $\mathbf{F}$ can be approximated by:

$$\hat{\mathbf{F}}(t) = \mathbf{F}(t-\Delta) = \ddot{\theta}_e(t-\Delta) - \alpha\tau_m(t-\Delta), \tag{3.8}$$

where Δ is a small time delay, and $\hat{\mathbf{F}}$ is the estimation of $\mathbf{F}$.

Assumption 3.2. *Since F_i is bounded, as $\hat{F}_i(t) = F_i(t-\Delta)$, the estimation error $\tilde{F}_i = F_i - \hat{F}_i$ satisfies that $|F_i| \leq \zeta_i$, ζ_i is a positive constant, $i = 1, 2$.*

3.2.3. ULM-based predefined-time controller design

The tracking error of the exoskeleton $\mathbf{e} = [e_1; e_2] \in \mathbb{R}^2$ can be rewritten as follows:

$$\mathbf{e} = \theta_e - \theta_r. \tag{3.9}$$

To achieve predefined-time convergence and better control accuracy, and inspired by [29], the prescribed constraints for tracking errors are given as follows:

$$-\mu_{el,i}(t) < e_i < \mu_{eu,i}(t), \tag{3.10}$$

$$\mu_{el,i}(t) = \begin{cases} (\mu_{0l,i} - \frac{t}{T_d})\mathrm{e}^{1-\frac{T_d}{T_d-t}} + \mu_{\infty l,i} & 0 \leq t \leq T_d, \\ \mu_{\infty l,i} & t > T_d, \end{cases} \tag{3.11}$$

$$\mu_{eu,i}(t) = \begin{cases} (\mu_{0u,i} - \frac{t}{T_d})\mathrm{e}^{1-\frac{T_d}{T_d-t}} + \mu_{\infty u,i} & 0 \leq t \leq T_d, \\ \mu_{\infty u,i} & t > T_d, \end{cases} \tag{3.12}$$

where $\mu_{0l,i} > \mu_{\infty l,i} > 0$, $\mu_{0u,i} > \mu_{\infty u,i} > 0$, T_d is the predefined time, e is the natural constant, $i = 1, 2$.

Assumption 3.3. *The prescribed constraints $\mu_{el,i}(t)$ and $\mu_{eu,i}(t)$ have lower bounds and are continuously derivable, and their absolute values of the derivatives have upper bounds, that is, $\mu_{el,i} \geq \mu_{\infty l,i}$, $\mu_{eu,i} \geq \mu_{\infty u,i}$, $\left|\dot{\mu}_{el,i}\right| \leq \mu_{Ml,i}$ and $\left|\dot{\mu}_{eu,i}\right| \leq \mu_{Mu,i}$, where $\mu_{Ml,i}$ and $\mu_{Mu,i}$ are positive constants, $i = 1, 2$.*

Remark 3.3. *The prescribed constraints on exoskeleton tracking errors, as defined in Eqs (14) and (15), guarantee predefined-time convergence. This property arises because the constraint boundaries $\mu_{eu,i}$ and $\mu_{el,i}$ are designed to contract to a very small range $[-\mu_{\infty l,i}, \mu_{\infty u,i}]$ precisely at the predefined time T_d, thereby enforcing error convergence within the specified time frame.*

Remark 3.4. *Due to the consideration of input constraints in this work, the predefined convergence time T_d cannot be set too small in practical applications. Moreover, an excessively small T_d value would cause large initial displacements of the exoskeleton, potentially creating harmful impacts on patients. Therefore, for safety considerations, T_d must be carefully selected within appropriate bounds.*

To deal with the input constraints, the auxiliary system [44] is introduced as follows:

$$\dot{q}_i = \Delta\tau_i - \phi_i q_i, \tag{3.13}$$

where $\Delta\tau_i = \tau_{m,i} - \tau_{v,i}$, $\phi_i > 1$, $i = 1, 2$.

Since the patient's upper limb may not initially lie on the desired rehabilitation trajectory, the exoskeleton's initial position typically deviates from the reference path, potentially causing the initial

tracking error **e** to exceed the prescribed constraint boundaries. However, the subsequent controller design employs BLF techniques, which would diverge if the initial error exceeds the constraints [45]. To resolve this issue, a shifting function [21] is introduced, which transforms the tracking error into a shifting error. The shifting function is defined as follows:

$$g_s(t) = \begin{cases} 1 - \frac{T_s - t}{T_s} + c & 0 \le t < T_s, \\ 1 & t \ge T_s, \end{cases} \tag{3.14}$$

where c is a small constant, and $0 < T_s \le T_d$ is the shifting time.

Then, the shifting error can be defined as:

$$\begin{cases} z_{1,i} = g_s e_i, \\ z_{2,i} = g_s \dot{e}_i, \end{cases} \tag{3.15}$$

with $i = 1, 2$.

The ULM (3.7) can be rewritten as follows:

$$\begin{cases} \dot{z}_{1,i} = z_{2,i} + \dot{g}_s e_{1,i} + \alpha g_s q_i, \\ \dot{z}_{2,i} = g_s(F_i - \ddot{\theta}_{r,i} + \alpha\tau_{v,i} + \alpha\phi_i q_i) + \dot{g}_s e_{2,i}, \end{cases} \tag{3.16}$$

where $e_{1,i} = e_i$, $e_{2,i} = \dot{e}_i$, $i = 1, 2$.

The following mathematical lemmas provide the foundation for the subsequent controller design and stability proof.

Lemma 3.1. [46] *For any $x_1, x_2 \in \mathbb{R}$, if there are two constants $y_1 > 0$ and $0 < y_2 \le 1$, then*

$$|\lfloor x_1 \rfloor^{y_1 y_2} - \lfloor x_2 \rfloor^{y_1 y_2}| \le 2^{1-y_2} |\lfloor x_1 \rfloor^{y_1} - \lfloor x_2 \rfloor^{y_1}|^{y_2},$$

with $\lfloor x_1 \rfloor^{y_1} = |x_1|^{y_1} sgn(x_1)$.

Lemma 3.2. [47] *For any $x_1, x_2 \in \mathbb{R}$, if there is a function $F(x_1, x_2) > 0$, then*

$$|x_1|^{y_1} |x_2|^{y_2} \le \frac{y_1}{y_1 + y_2} L|x_1|^{y_1 + y_2} + \frac{y_2}{y_1 + y_2} L^{-\frac{y_1}{y_2}} |x_2|^{y_1 + y_2}.$$

Lemma 3.3. [47] *For any $x_i \in \mathbb{R}$, if there is a constant $0 < y < 1$, then*

$$(|x_1| + |x_2| + ... + |x_k|)^y \le |x_1|^y + |x_2|^y + ... + |x_k|^y,$$

with $i = 1, 2, ..., k$.

Lemma 3.4. [48] *For any $x \in [0, 1)$,*

$$\tan\left(\frac{\pi x}{2}\right) \le \frac{\pi x}{2} \sec^2\left(\frac{\pi x}{2}\right).$$

The design of the ULM-based predefined-time controller will now be presented. Utilizing the backstepping technique, the controller development proceeds in two stages, corresponding to the second-order ULM approximation of the original system.

Step 1: A BLF is employed to satisfy performance constraints on the exoskeleton's angular tracking

error, thereby indirectly achieving predefined-time convergence;
Step 2: A Lyapunov function is introduced to derive the final control law $\tau_{v,i}$.

For simplicity, the analysis focuses on the i-th joint, $i = 1, 2$.

Step 1: To satisfy performance constraints on the exoskeleton's angular tracking error, a BLF is considered on the region $\Omega_1 = \{z_{1,i} : -\mu_{el,i}(t) < z_{1,i} < \mu_{eu,i}(t)\}$:

$$V_{BLF,i}(z_{1,i}, \mu_{el,i}, \mu_{eh,i}) = a\frac{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}{\frac{2p+\varepsilon}{r_1}\pi}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}\right) + (1-a)\frac{2\mu_{eh,i}^{\frac{2p+\varepsilon}{r_1}}}{\frac{2p+\varepsilon}{r_1}\pi}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{eh,i}^{\frac{2p+\varepsilon}{r_1}}}\right), \tag{3.17}$$

where $p \geq r_1 > r_2 > r_3 > \varepsilon > 0$, $r_2 = r_1 - \varepsilon$, $r_3 = r_2 - \varepsilon$, $a = \begin{cases} 1 & z_{1,i} \leq 0 \\ 0 & z_{1,i} > 0 \end{cases}$.

Differentiating the BLF (3.17), one obtains:

$$\dot{V}_{BLF,i} = \frac{\partial V_{BLF,i}}{\partial z_{1,i}}\dot{z}_{1,i} + \frac{\partial V_{BLF,i}}{\partial \mu_{el,i}}\dot{\mu}_{el,i} + \frac{\partial V_{BLF,i}}{\partial \mu_{eh,i}}\dot{\mu}_{eh,i}. \tag{3.18}$$

Considering the first term of (3.18), one has:

$$\begin{aligned}\frac{\partial V_{BLF,i}}{\partial z_{1,i}}\dot{z}_{1,i} =& a\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{zl,i}(t)^{\frac{2p+\varepsilon}{r_1}}}\right)\left\lfloor z_{1,i}\right\rceil^{\frac{2p+\varepsilon-r_1}{r_1}}(z_{2,i} + \dot{g}_r e_{1,i} + \alpha g_r q_i) \\ &+ (1-a)\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{zh,i}(t)^{\frac{2p+\varepsilon}{r_1}}}\right)\left\lfloor z_{1,i}\right\rceil^{\frac{2p+\varepsilon-r_1}{r_1}}(z_{2,i} + \dot{g}_r e_{1,i} + \alpha g_r q_i) \\ =& \Xi(z_{1,i})\left(\left\lfloor s_{1,i}\right\rceil^{\frac{2p+\varepsilon-r_1}{\sigma}} z_{2,i} + \left\lfloor s_{1,i}\right\rceil^{\frac{2p+\varepsilon-r_1}{\sigma}}(\dot{g}_r e_{1,i} + \alpha g_r q_i)\right),\end{aligned} \tag{3.19}$$

where $\Xi(z_{1,i}) = a\Psi_1(z_{1,i}) + (1-a)\Psi_2(z_{1,i})$, $\Psi_1(z_{1,i}) = \sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}\right)$, $\Psi_2(z_{1,i}) = \sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{eh,i}^{\frac{2p+\varepsilon}{r_1}}}\right)$, $s_{1,i} = \left\lfloor z_{1,i}\right\rceil^{\frac{\sigma}{r_1}}$, $p \geq \sigma \geq r_1$, $\lfloor x\rceil^a = |x|^a \operatorname{sgn}(x)$.

Then, consider the second term of Eq (3.18), and one has:

$$\frac{\partial V_{BLF,i}}{\partial \mu_{el,i}}\dot{\mu}_{el,i} = \frac{2a}{\pi}\mu_{el,i}^{\frac{2p+\varepsilon-r_1}{r_1}}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}\right)\dot{\mu}_{el,i} - \frac{a}{\mu_{el,i}}\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}\right)\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}\dot{\mu}_{el,i}. \tag{3.20}$$

For the first term of (3.20), using Lemma 3.4 and Assumption 3.3, one has:

$$\begin{aligned}\frac{2a\mu_{el,i}^{\frac{2p+\varepsilon}{\gamma_1}-1}}{\pi}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{\gamma_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{\gamma_1}}}\right)\dot{\mu}_{el,i} &\leq \frac{2a\mu_{el,i}^{\frac{2p+\varepsilon}{\gamma_1}-1}}{\pi}\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{\gamma_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{\gamma_1}}}\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{\gamma_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{\gamma_1}}}\right)\dot{\mu}_{el,i} \\ &\leq \frac{a}{\mu_{ml,i}}\Psi_1(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Ml,i}.\end{aligned} \tag{3.21}$$

And for the second term of (3.20), considering the Assumption 3.3, and one has:

$$-\frac{a}{\mu_{el,i}}\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{zl,i}^{\frac{2p+\varepsilon}{r_1}}}\right)\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}\dot{\mu}_{el,i} \leq \frac{a}{\mu_{ml,i}}\Psi_1(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Ml,i}. \tag{3.22}$$

Substitute Eqs (3.21) and (3.22) into (3.20), and one has:

$$\frac{\partial V_{BLF,i}}{\partial \mu_{el,i}}\dot{\mu}_{el,i} \leq \frac{2a}{\mu_{ml,i}}\Psi_1(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Ml,i}. \tag{3.23}$$

Similarly, the third term of Eq (3.18) can be rewritten as follows:

$$\frac{\partial V_{BLF,i}}{\partial \mu_{eh,i}}\dot{\mu}_{eh,i} \leq \frac{2(1-a)}{\mu_{mh,i}}\Psi_2(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Mh,i}. \tag{3.24}$$

Substitute Eqs (3.19), (3.23), and (3.24) into (3.18), and one has:

$$\begin{aligned}\dot{V}_{BLF,i} \leq & \Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p+\varepsilon-r_1}{\sigma}}(z_{2,i}-z_{2,i}^*) + \Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p+\varepsilon-r_1}{\sigma}}(\dot{g}_s e_{1,i}+\alpha g_s q_i)\\ & + \Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p+\varepsilon-r_1}{\sigma}}z_{2,i}^* + \frac{2a}{\mu_{ml,i}}\Psi_l(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Ml,i} + \frac{2(1-a)}{\mu_{mh,i}}\Psi_h(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\mu_{Mh,i},\end{aligned} \tag{3.25}$$

where $z_{2,i}^* = -\lambda_1(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{r_2}{\sigma}}$, $\lambda_1(z_{1,i}) = \left(\frac{2\mu_{Ml,i}a}{\mu_{ml,i}}\left|s_{1,i}\right|^{\frac{\varepsilon}{\sigma}} + \frac{2\mu_{Mh,i}(1-a)}{\mu_{mh,i}}\left|s_{1,i}\right|^{\frac{\varepsilon}{\sigma}} + \lambda_0\right)$, $\lambda_0 > 0$ is a constant.

Taking into account the definition of $z_{2,i}^*$, Eq (3.25) can be rewritten as follows:

$$\dot{V}_{BLF,i} \leq \Xi(z_{1,i})\left(\lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(z_{2,i}-z_{2,i}^*) + \lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(\dot{g}_s e_{1,i}+\alpha g_s q_i) - \lambda_0\left|s_{1,i}\right|^{\frac{2p}{\sigma}}\right). \tag{3.26}$$

Step 2: A Lyapunov function that includes the BLF, a power integrator, and the estimation of ζ_i is considered.

Consider the candidate Lyapunov function defined as:

$$V_i(z_{1,i}, z_{2,i}) = V_{BLF,i}(z_{1,i}, \mu_{el,i}, \mu_{eh,i}) + V_{f,i}(z_{1,i}, z_{2,i}) + \frac{\eta}{2}\tilde{\zeta}_i^{\mathrm{T}}\tilde{\zeta}_i, \tag{3.27}$$

$$V_{f,i}(z_{1,i}, z_{2,i}) = \int_{z_{2,i}^*}^{z_{2,i}} \left\lfloor \lfloor \upsilon \rceil^{\frac{\sigma}{r_2}} - \left\lfloor z_{2,i}^* \right\rceil^{\frac{\sigma}{r_2}} \right\rceil^{\frac{2p-r_3}{\sigma}} d\upsilon, \tag{3.28}$$

where ζ_i is added to compensate the TDE estimation error, $\tilde{\zeta}_i = \hat{\zeta}_i - \zeta_i$ is the estimation error of ζ_i, $\hat{\zeta}_i$ is the estimation of ζ_i and $\eta > 0$ is a constant.

By differentiating the Lyapunov function (3.27), one obtains:

$$\begin{aligned}\dot{V}_i =& \dot{V}_{BLF,i} + \frac{\partial V_{f,i}}{\partial z_{1,i}}\dot{z}_{1,i} + \frac{\partial V_{f,i}}{\partial z_{2,i}}\dot{z}_{2,i} + \eta\tilde{\zeta}_i\dot{\tilde{\zeta}}_i\\ \leq & \Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(z_{2,i}-z_{2,i}^*) + \Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(\dot{g}_s e_{1,i}+\alpha g_s q_i) - \lambda_0\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p}{\sigma}}\\ & + \frac{\partial V_{f,i}}{\partial z_{1,i}}\dot{z}_{1,i} + \frac{\partial V_{f,i}}{\partial z_{2,i}}\dot{z}_{2,i} - \eta\tilde{\zeta}_i^{\mathrm{T}}\dot{\tilde{\zeta}}_i.\end{aligned} \tag{3.29}$$

Define that $s_{2,i} = \lfloor z_{2,i}\rceil^{\frac{\sigma}{r_2}} - \left\lfloor z_{2,i}^*\right\rceil^{\frac{\sigma}{r_2}}$, and using Lemma 3.1 and the first term of Eq (3.29), one obtains:

$$\Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(z_{2,i}-z_{2,i}^*) \leq 2^{1-\frac{r_2}{\sigma}}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p-r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}}. \tag{3.30}$$

Using Lemma 3.2, one has:

$$\left|s_{1,i}\right|^{\frac{2p-r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}} \leq \frac{2p-r_2}{2p}\varphi\left|s_{1,i}\right|^{\frac{2p}{\sigma}} + \frac{r_2}{2p}\varphi^{-\frac{2p-r_2}{r_2}}\left|s_{2,i}\right|^{\frac{2p}{\sigma}}, \tag{3.31}$$

where $\varphi = \left(\frac{(2p-r_2)2^{2-\frac{r_2}{\sigma}}}{p\lambda_0}\right)^{-1}$.

Consider Eq (3.31) and (3.30) can be rewritten as follows:

$$\Xi(z_{1,i})\lfloor s_{1,i}\rceil^{\frac{2p-r_2}{\sigma}}(z_{2,i}-z^*_{2,i}) \leq \frac{\lambda_0}{4}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p}{\sigma}} + c_1(\lambda_0)\Xi(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}}, \tag{3.32}$$

where $c_1(\lambda_0) = 2^{-\frac{r_2}{\sigma}}\frac{r_2}{p}\left(\frac{(2p-r_2)2^{2-\frac{r_2}{\sigma}}}{p\lambda_0}\right)^{\frac{2p-r_2}{r_2}}$.

Proceeding to analyze the fourth component of (3.29) and through Lemma 3.1, one obtains:

$$\begin{aligned}\left|\frac{\partial V_{f,i}}{\partial z_{1,i}}\dot{z}_{1,i}\right| &\leq \frac{2p-r_3}{\sigma}\left|z_{2,i}-z^*_{2,i}\right|\left|s_{2,i}\right|^{\frac{2p-r_3}{\sigma}-1}\left|\frac{\partial\left\lfloor z^*_{2,i}\right\rceil^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \\ &\leq \frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}+\frac{2p-r_3}{\sigma}-1}\left|\frac{\partial\left\lfloor z^*_{2,i}\right\rceil^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\dot{z}_{1,i}\right|.\end{aligned} \tag{3.33}$$

For the last term of Eq (3.33), and considering the definition of $z^*_{2,i}$, it follows that:

$$\left|\frac{\partial\left\lfloor z^*_{2,i}\right\rceil^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\right| \leq \left|\frac{\partial\lambda_1^{\frac{\sigma}{r_2}}(z_{1,i})}{\partial z_{1,i}}s_{1,i}\right| \leq c_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}, \tag{3.34}$$

where $c_2(z_{1,i}) = \left|\frac{\partial\lambda_1^{\frac{\sigma}{r_2}}(z_{1,i})}{\partial z_{1,i}}z_{1,i}\right|$.

Consider the definitions of $s_{2,i}$ and $z^*_{2,i}$, and by applying Lemma 3.3, one obtains:

$$\left|z_{2,i}\right| \leq \left|s_{2,i}\right|^{\frac{r_2}{\sigma}} + \lambda_1(z_{1,i})\left|s_{1,i}\right|^{\frac{r_2}{\sigma}}. \tag{3.35}$$

Based on Eqs (3.16), (3.34), and (3.35), it yields that

$$\begin{aligned}\left|\frac{\partial\left\lfloor z^*_{2,i}\right\rceil^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \leq &\left(c_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}} + c_2(z_{1,i})\lambda_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}}\right) \\ &+ c_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|\dot{g}_s e_{1,i} + \alpha g_s q_i\right|.\end{aligned} \tag{3.36}$$

Define that $b_1(z_{1,i}) = c_2(z_{1,i})$ and $b_2(z_{1,i}) = c_2(z_{1,i})\lambda_1(z_{1,i})$, and Eq (3.36) becomes:

$$\left|\frac{\partial\left\lfloor z^*_{2,i}\right\rceil^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \leq b_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}} + b_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}} + b_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|\dot{g}_s e_{1,i} + \alpha g_s q_i\right|. \tag{3.37}$$

Bearing in mind the first term of Eq (3.37), and using Lemma 3.2, it yields:

$$b_1(z_{1,i})\left|s_{1,i}\right|^{\frac{\sigma-r_1}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}} \leq \left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}} + c_3(z_{1,i})\left|s_{2,i}\right|^{1-\frac{\varepsilon}{\sigma}}, \tag{3.38}$$

with $c_3(z_{1,i}) = b_1(z_{1,i})^{\frac{r_2}{\sigma-\varepsilon}}\left(b_1(z_{1,i})^{\frac{\sigma-r_1}{\sigma-\varepsilon}}\right)^{\frac{\sigma-r_2}{r_2}}$.

Substituting Eq (3.38) into (3.37) yields:

$$\left|\frac{\partial\left\lfloor z_{2,i}^*\right\rfloor^{\frac{\sigma}{r_2}}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \leq c_4(z_{1,i})\left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}} + c_3(z_{1,i})\left|s_{2,i}\right|^{1-\frac{\varepsilon}{\sigma}} + b_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|\dot{g}_s e_{1,i} + \alpha g_s q_i\right|, \tag{3.39}$$

with $c_4(z_{1,i}) = b_2(z_{1,i}) + 1$.

By substituting Eq (3.39) into (3.33), it can be deduced that

$$\begin{aligned}\left|\frac{\partial V_{f,i}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \leq & \frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}+\frac{2p-r_3}{\sigma}-1}\left(c_4(z_{1,i})\left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}} + c_3(z_{1,i})\left|s_{2,i}\right|^{1-\frac{\varepsilon}{\sigma}}\right) \\ & + \frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}+\frac{2p-r_3}{\sigma}-1}b_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|\dot{g}_s e_{1,i} + \alpha g_s q_i\right|.\end{aligned} \tag{3.40}$$

For the first term of Eq (3.40), Lemma 3.2 yields:

$$\frac{2p-r_3}{\sigma}c_4(z_{1,i})2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}+\frac{2p-r_3}{\sigma}-1}\left|s_{1,i}\right|^{1-\frac{\varepsilon}{\sigma}} \leq \frac{\lambda_0}{4}\left|s_{1,i}\right|^{\frac{2p}{\sigma}} + c_5(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}}, \tag{3.41}$$

where $c_5(z_{1,i}) = 2^{1-\frac{r_2}{\sigma}}\frac{2p-r_3}{\sigma}c_4(z_{1,i})^{\frac{2p+\varepsilon-\sigma}{2p}}\left(\frac{2^{2-\frac{r_2}{\sigma}}}{\lambda_0}\frac{(2p-r_3)(\sigma-\varepsilon)}{\sigma p}c_4(z_{1,i})\right)^{\frac{\sigma-\varepsilon}{2p+\varepsilon-\sigma}}$.

Substituting Eq (3.41) into (3.40), one gets:

$$\begin{aligned}\left|\frac{\partial V_{f,i}}{\partial z_{1,i}}\dot{z}_{1,i}\right| \leq & \frac{\lambda_0}{4}\left|s_{1,i}\right|^{\frac{2p}{\sigma}} + c_6(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}} \\ & + \frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{2p+\varepsilon-\sigma}{\sigma}}b_1(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|\dot{g}_s e_{1,i} + \alpha g_s q_i\right|,\end{aligned} \tag{3.42}$$

where $c_6(z_{1,i}) = c_5(z_{1,i}) + \frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}c_3(z_{1,i})$.

Consider the fifth term of Eq (3.29), it yields:

$$\frac{\partial V_{f,i}}{\partial z_{2,i}}\dot{z}_{2,i} = \lfloor s_{2,i}\rceil^{\frac{2p-r_3}{\sigma}}(g_s(F_i - \ddot{\theta}_{r,i} + \alpha\tau_{v,i} + \alpha\phi_i q_i) + \dot{g}_s e_{2,i}). \tag{3.43}$$

Substituting Eqs (3.32), (3.42), and (3.43) into (3.29), and considering $\Xi(z_{1,i}) > 1$, it yields:

$$\begin{aligned}\dot{V}_i \leq & -\frac{\lambda_{0,i}}{2}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p}{\sigma}} + (c_1(\lambda_{0,i}) + c_6(z_{1,i}))\Xi(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}} \\ & + \Xi(z_{1,i})\left(\frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}c_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|s_{2,i}\right|^{\frac{2p+\varepsilon-\sigma}{\sigma}} + \left|s_{1,i}\right|^{\frac{2p-r_2}{\sigma}}\right)\left|\dot{g}_s e_{1,i} + g_s q_i\right| \\ & + \Xi(z_{1,i})\lfloor s_{2,i}\rceil^{\frac{2p-r_3}{\sigma}}(g_s(F_i - \ddot{\theta}_{r,i} + \alpha\tau_{v,i} + \alpha\phi_i q_i) + \dot{g}_s e_{2,i}) - \eta\tilde{\zeta}_i^{\mathrm{T}}\dot{\hat{\zeta}}_i.\end{aligned} \tag{3.44}$$

With the backstepping method, the control law $\tau_{v,i}$ can be given as follows:

$$\begin{aligned}\tau_{v,i} = &-\frac{1}{\alpha}\left(\hat{F}_i - \ddot{\theta}_{r,i} + \alpha\phi_i q_i + \hat{\zeta}_i + \frac{1}{g_s}\left(\lambda_2(z_{1,i})\lfloor s_{2,i}\rceil^{\frac{\gamma_3}{\sigma}} + \dot{g}_s z_{2,i}\right)\right.\\ &\left.+\frac{1}{g_s}\left(\left(\frac{2p-r_3}{\sigma}2^{1-\frac{r_2}{\sigma}}c_2(z_{1,i})\left|s_{1,i}\right|^{1-\frac{r_1}{\sigma}}\left|s_{2,i}\right|^{\frac{r_2}{\sigma}-1} + \left|s_{1,i}\right|^{\frac{2p-r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{r_3-2p}{\sigma}}\right)\left|\dot{g}_s z_{1,i} + g_s q_i\right|\right)\right),\end{aligned} \tag{3.45}$$

where $\lambda_2(\varsigma_{1,i}) = c_1(\lambda_0) + c_6(\varsigma_{1,i}) + \frac{1}{2}\lambda_0$.

Substituting the control law $\tau_{v,i}$ into Eq (3.44), it holds that:

$$\dot{V}_i \leq -\frac{\lambda_{0,i}}{2}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p}{\sigma}} - \frac{\lambda_{0,i}}{2}\Xi(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}} + \Xi(z_{1,i})\lfloor s_{2,i}\rceil^{\frac{2p-r_3}{\sigma}} g_s\tilde{\zeta}_i - \eta\tilde{\zeta}_i\dot{\hat{\zeta}}_i. \tag{3.46}$$

With the backstepping method, the adaptive estimation law of ζ_i can be given as follows:

$$\dot{\hat{\zeta}}_i = \frac{1}{\eta}g_s\Xi(z_{1,i})\lfloor s_{2,i}\rceil^{\frac{2p-r_3}{\sigma}}. \tag{3.47}$$

In summary, the proposed UPTAC can be composed based on impedance controller (3.4), adaptive strategy (3.2), admittance controller (3.5)–(3.6), ULM-based predefined-time controller (3.45), and adaptive estimation law (3.47).

Remark 3.5. *As illustrated in Figure 3, the outer impedance sub-control loop governs the rehabilitation strategy. Through the impedance controller, circular rehabilitation path, and task performance function, the desired assistive torque and trajectory is generated according to the patient's movement capability. The inner torque sub-control loop achieves precise regulation of the assistive torque. The admittance controller transforms the desired assistive torque and trajectory provided by the outer loop into the exoskeleton's reference trajectory. Subsequently, a predefined-time controller is designed under the ultra-local model framework to enable the exoskeleton to accurately track this reference trajectory, ultimately realizing assist-as-needed control.*

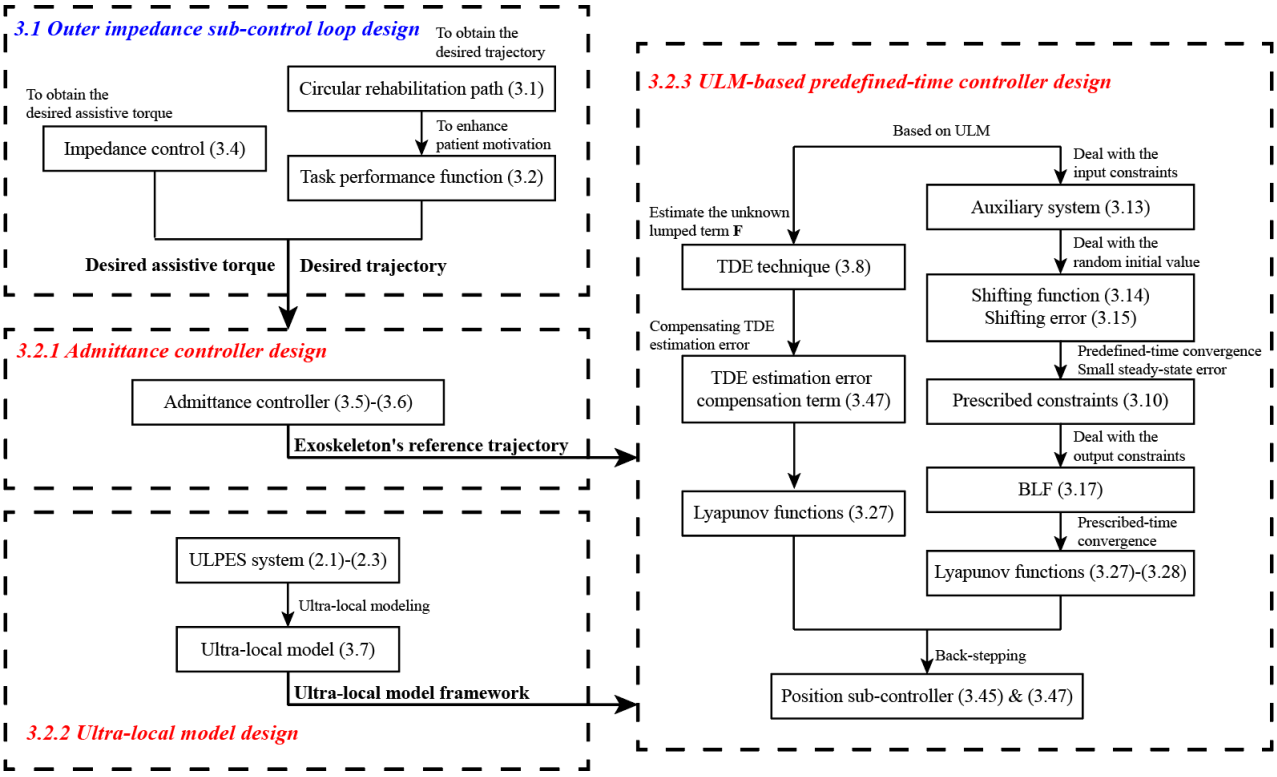


Figure 3. The design process of the controller.

Furthermore, as also illustrated in Figure 2, the patient's upper limb position and the exoskeleton's assistive torque information—obtained through inner loop—also influence the desired assistive torque and trajectory generated by the outer loop.

3.3. Stability analysis

Lemma 3.5. [49] *Consider the system* $\dot{y} = f(y, u)$. *Suppose that there is a continuous function* $V(x)$ *and scalars* $a > 0$, $0 < b < 1$, *and* $0 < c < \infty$ *such that*

$$\dot{V} \leq -aV^b + c.$$

Then, the system is practical finite-time stable.

The stability theorem of the closed-loop 2-DOF ULPES (2.1)–(2.3) under input constraints using the proposed UPTAC is stated and analyzed as follows:

Theorem 3.1. *Considering the 2-DOF ULPES (2.1)–(2.3) under input constraints, applying the proposed UPTHC, there are parameters* $p \geq \sigma \geq r_1$, $r_2 = r_1 - \varepsilon$, $r_3 = r_2 - \varepsilon$, $\eta > 0$ *such that the following properties can be guaranteed:*
(i) The system can be stable;
(ii) The tracking error e_i *satisfies the performance constraints;*
(iii) The tracking error e_i *can converge to a small bounded interval before the predefined time* T_d*;*
(iv) The patient's upper limb can track the desired rehabilitation trajectory $\boldsymbol{P}_d$.

Proof. (i) Substituting the estimation of ζ_i into Eq (3.46), it holds that:

$$\dot{V}_i \leq -\frac{\lambda_{0,i}}{2}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p}{\sigma}} - \frac{\lambda_{0,i}}{2}\Xi(z_{1,i})\left|s_{2,i}\right|^{\frac{2p}{\sigma}} \leq 0. \tag{3.48}$$

From the definition of the BLF $V_{BLF,i}$ and using Lemma 3.4, it can be deduced that

$$\begin{aligned} V_{BLF,i} &\leq a\frac{2r_1}{2p+\varepsilon}\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}\right)\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}} + (1-a)\frac{2r_1}{2p+\varepsilon}\sec^2\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{zh,i}^{\frac{2p+\varepsilon}{r_1}}}\right)\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}} \\ &\leq \frac{2r_1}{2p+\varepsilon}\Xi(z_{1,i})\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}. \end{aligned} \tag{3.49}$$

Then, in view of the definition of $V_{f,i}$ and using Lemma 3.1, one has:

$$\int_{z_{2,i}^*}^{z_{2,i}}\left|\lfloor\upsilon\rceil^{\frac{\sigma}{r_2}} - \left\lfloor z_{2,i}^*\right\rceil^{\frac{\sigma}{r_2}}\right|^{\frac{2p-r_3}{\sigma}} d\upsilon \leq \left|z_{2,i} - z_{2,i}^*\right|\left|s_{2,i}\right|^{\frac{2p-r_3}{\sigma}} \leq 2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{2p+\varepsilon}{\sigma}}. \tag{3.50}$$

Substituting Eqs (3.49) and (3.50) into V_i, it yields:

$$V_i \leq \frac{2r_1}{2p+\varepsilon}\Xi(z_{1,i})\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}} + 2^{1-\frac{r_2}{\sigma}}\left|s_{2,i}\right|^{\frac{2p+\varepsilon}{\sigma}} + \frac{\eta}{2}\tilde{\zeta}_i^2. \tag{3.51}$$

Define that $\chi = \max\left\{\frac{2r_1}{2p+\varepsilon}\Xi(z_{1,i}), 2^{1-\frac{r_2}{\sigma}}\right\}$, one obtains:

$$V_i \leq \chi\left(\left|s_{1,i}\right|^{\frac{2p+\varepsilon}{\sigma}} + \left|s_{2,i}\right|^{\frac{2p+\varepsilon}{\sigma}}\right) + \frac{\eta}{2}\tilde{\zeta}_i^2. \tag{3.52}$$

Consider Eq (3.48), it yields:

$$\dot{V}_i \leq -cV_i^{\frac{2p}{2p+\varepsilon}} + c\left(\frac{\eta}{2}\tilde{\zeta}_i^2\right)^{\frac{2p}{2p+\varepsilon}}, \tag{3.53}$$

with $c = \frac{\beta_0\Xi(z_{1,i})}{2\chi^{\frac{2p}{2p+\varepsilon}}}$.

Noting that $c > 0$, $0 < \frac{2p}{2p+\varepsilon} < 1$, and $0 < c\left(\frac{\eta}{2}\tilde{\zeta}_i^2\right)^{\frac{2p}{2p+\varepsilon}} < \infty$, system stability can be achieved through Lemma 3.5.

(ii) Since the BLF $V_{BLF,i}$ and Lyapunov function V_i are positive defined, and considering $\dot{V}_i < 0$, it yields:

$$V_{BLF,i} \leq V_i \leq V_i(z_{1,i}(0), z_{2,i}(0)). \tag{3.54}$$

In view of the definition of $V_{BLF,i}$, one has:

$$a\frac{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}{\frac{2p+\varepsilon}{r_1}\pi}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{zl,i}^{\frac{2p+\varepsilon}{r_1}}}\right) + (1-a)\frac{2\mu_{eh,i}^{\frac{2p+\varepsilon}{r_1}}}{\frac{2p+\varepsilon}{r_1}\pi}\tan\left(\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{eh,i}^{\frac{2p+\varepsilon}{r_1}}}\right) \leq V_i(z_{1,i}(0), z_{2,i}(0)). \tag{3.55}$$

When $z_{1,i} \leq 0$, Eq (3.55) becomes:

$$\frac{\pi\left|z_{1,i}\right|^{\frac{2p+\varepsilon}{r_1}}}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}} \leq \arctan\left(\frac{\frac{2p+\varepsilon}{r_1}\pi}{2\mu_{el,i}^{\frac{2p+\varepsilon}{r_1}}}V_i(z_{1,i}(0), z_{2,i}(0))\right) < \frac{\pi}{2}. \tag{3.56}$$

Then, from Eq (3.56), one has

$$-\mu_{el,i} < z_{1,i} \leq 0. \tag{3.57}$$

Similarly, it can be proven that $0 < z_{1,i} < \mu_{eh,i}$.

Accordingly, one can get that the tracking error e_i satisfies the performance constraints.

(iii) Since the tracking error e is guaranteed to remain within the prescribed performance constraints, that is $-\mu_{el,i} < z_{1,i} < \mu_{eh,i}$, and considering the asymptotic properties of the constraint boundaries ($\lim_{t\to T_s} \mu_{el,i} = \mu_{\infty l,i}$, $\lim_{t\to T_s} \mu_{eh,i} = \mu_{\infty h,i}$), one can conclude that e_i converges to a small interval $[\mu_{\infty l,i}, \mu_{\infty h,i}]$ prior to the predefined time T_d.

(iv) The second part of the proof demonstrates that the exoskeleton's angular tracking error $\mathbf{e}$ satisfies the performance constraints. Consequently, the generated assistive torque τ_a accurately tracks the desired assistive torque τ_{da}. The impedance controller (3.4), which exhibits a PD-type structure (with convergence guarantees established in [50]), ensures asymptotic convergence of the patient's upper limb to the target rehabilitation trajectory $\mathbf{P}_d$. □

4. UPTAC application and simulation results

The ULPES model, developed in SolidWorks, is imported into MATLAB/SimMechanics for co-simulation to validate the proposed UPTAC. The parameters of the ULPES model are given in Tables 1 and 2.

Table 1. Parameters of ULPES model.

Part name	Mass (kg)	Center of mass position (m)
Exoskeleton upper arm	2.357	(-0.439, -0.403, 0.604)
Exoskeleton forearm	1.177	(-0.689, -0.434, 0.665)
Patient upper arm	1.530	(-0.199, 0.096, 0.974)
Patient forearm	1.084	(-0.951, -0.591, 0.646)

Table 2. Parameters of ULPES model.

Part name	Moment of inertia (kg·mm^2)	Products of inertia (kg·mm^2)
Exoskeleton upper arm	(2054.30, 17954.81, 17625.12)	(0.000077, -2762.28, 0.077)
Exoskeleton forearm	(2686.60, 14117.72, 14474.28)	(1000.85, 3015.29, -3037.84)
Patient upper arm	(11897.13, 1423.72, 12059.21)	(-186.88, -14.95, -1013.00)
Patient forearm	(583.10, 10046.14, 10145.05)	(10.64, -352.71, 12.47)

4.1. Case 1: Comparison of results between TDE-SMC and BLF-SMC

TDE-sliding mode control (TDE-SMC) [51] and BLF-based sliding mode control (BLF-SMC) [48] are applied to illustrate the performance of UPTAC.

The referred TDE-SMC [51] is given as follows:

$$\begin{cases} \mathbf{s} = \dot{\mathbf{e}} + r_1\mathbf{e} + r_2\rho(\mathbf{e}), \\ \mathbf{u} = \hat{\mathbf{m}}(\ddot{\mathbf{y}}_d + \boldsymbol{\Omega} + \mu_1\mathbf{s} + \mu_2\mathrm{sgn}(\mathbf{s})^{\beta}) + \hat{\mathbf{m}}\mathbf{k}(\sigma)|\boldsymbol{\Omega}|\,\mathrm{sgn}(\mathbf{s}) + \mathbf{u}_{t-\Delta t} - \hat{\mathbf{m}}\ddot{\mathbf{y}}_{t-\Delta t}. \end{cases} \tag{4.1}$$

Then, the BLF-SMC [48] is given as follows:

$$\begin{aligned} u_i = &-\beta_2(s_{1,i})\Phi(s_{1,i})\left\lfloor \lfloor s_{2,i}\rfloor^{\frac{a}{r_2}} + \beta_1^{\frac{a}{r_2}}(s_{1,i})\lfloor s_{1,i}\rfloor^{\frac{a}{r_1}}\right\rfloor^{\frac{r_3}{a}} \\ &- \frac{G(x)}{H}\cdot \operatorname{sgn}\left(\lfloor s_{2,i}\rfloor^{\frac{a}{r_2}} + \beta_1^{\frac{a}{r_2}}(s_{1,i})\lfloor s_{1,i}\rfloor^{\frac{a}{r_1}}\right). \end{aligned} \tag{4.2}$$

For rehabilitation task specification, the end-effector trajectory $\mathbf{P}_d$ in Cartesian space follows a circular path centered at [0.35 m, 0] with 0.15 m radius. The parameters of the patient's upper limb are selected as $k_{p,1} = k_{p,2} = 85$, $b_{p,1} = b_{p,2} = 15.5$. Assuming that the patient's movement capability will gradually recover through rehabilitation training, the parameter k_m is selected as follows:

$$k_m = \begin{cases} 0.1 & 0 \le t \le 6, \\ 0.1 + 0.05(t-6) & 6 < t \le 24, \\ 1 & t > 24. \end{cases}$$

The input constraints of each joint are set as $\mu_{\tau,1} = 20$, $\mu_{\tau,2} = 10$.

The predefined time is set as $T_d = 0.5$s. Then, to guarantee predefined-time convergence while maintaining good tracking precision after the predefined time T_d, the prescribed constraints for tracking errors are set as follows:

$$\mu_{el,1}(t) = \begin{cases} (2\times 10^{-3} - \frac{t}{0.5})\mathrm{e}^{1-\frac{0.5}{0.5-t}} + 7.5\times 10^{-4} & 0 \le t \le 0.5, \\ 7.5\times 10^{-4} & t > 0.5, \end{cases}$$

$$\mu_{el,2}(t) = \begin{cases} (3\times 10^{-3} - \frac{t}{0.5})\mathrm{e}^{1-\frac{0.5}{0.5-t}} + 5\times 10^{-4} & 0 \le t \le 0.5, \\ 5\times 10^{-4} & t > 0.5, \end{cases}$$

$$\mu_{eu,1}(t) = \begin{cases} (3\times 10^{-3} - \frac{t}{0.5})\mathrm{e}^{1-\frac{0.5}{0.5-t}} + 5\times 10^{-4} & 0 \le t \le 0.5, \\ 5\times 10^{-4} & t > 0.5, \end{cases}$$

$$\mu_{eu,2}(t) = \begin{cases} (2\times 10^{-3} - \frac{t}{0.5})\mathrm{e}^{1-\frac{0.5}{0.5-t}} + 5\times 10^{-4} & 0 \le t \le 0.5, \\ 5\times 10^{-4} & t > 0.5. \end{cases}$$

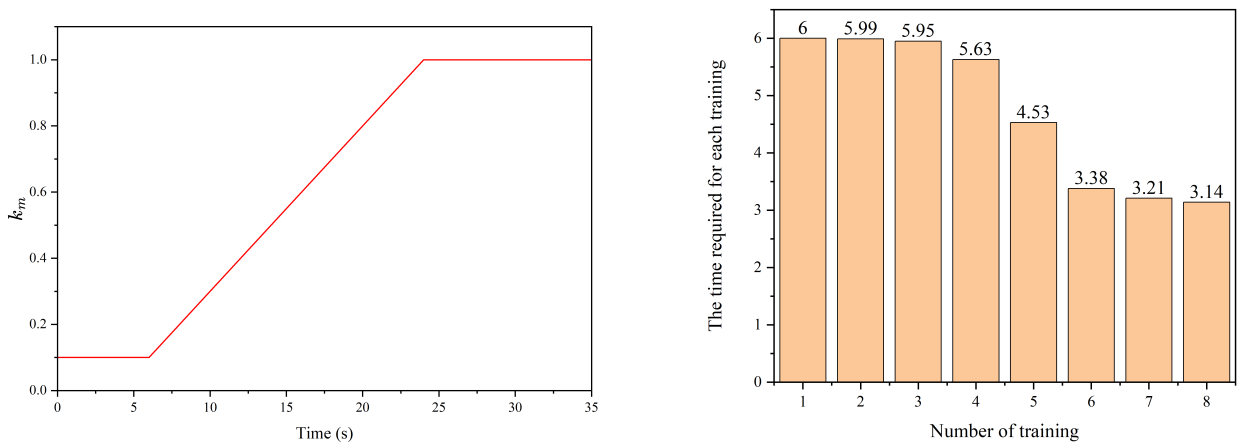
Other parameters of UPTAC are selected as $k_{max} = 6$, $k_{min} = 3$, $\varsigma_1 = 1/13$, $\varsigma_2 = 55$, $T_s = 0.5$s, $\alpha = 1$, $\Delta = 0.001$, $\phi_i = 100$, $p = 1$, $\sigma = 1$, $r_1 = 1$, $\epsilon = 1/9$, $\lambda_0 = 10$, $\eta = 1$. The parameters of TDE-iPD are set as $\alpha = 1$, $k_d = 10$, $k_p = 100$. The parameters of TDE-SMC are chosen as $r_1 = 50$, $r_2 = 1$, $\sigma = 0.25$, $\mu_1 = 100$, $\mu_2 = 10$, $\beta = 0.8$. And the parameters of BLF-SMC are chosen as $r_1 = 1$, $r_2 = 8/9$, $r_3 = 7/9$, $a = 1$, $H = 1$, $G(x) = 10$.

Remark 4.1. *The parameter selection for the proposed UPTAC control method is given as follows. For the task performance function, k_{min} and k_{max}, representing the minimum and maximum duration of each rehabilitation movement, are set to 3 and 6, respectively. Then, ς_1 is the slope coefficient used to control the rate of change of k_d between k_{max} and k_{min}. The smaller ς_1, the smoother it is. To increase patient comfort, $\varsigma_1 = 1/13$ is chosen. And ς_2 is a threshold parameter that determines the critical torque value for triggering speed adjustment. Increasing ς_2 will delay the triggering timing for speed increase, and $\varsigma_2 = 55$ is selected through trial-and-error method. For the parameter selection of ULM-based predefined-time controller, α will affect the convergence speed and overshoot. Therefore,*

in this simulation, $\alpha = 1$ is selected to ensure tracking performance. Parameters p, σ, r_1, and η are commonly assigned a value of 1. Parameter ϵ is selected as 1/9 through trial-and-error. Each element in parameter λ_0 is generally selected as an integer between intervals $[1, 10]$. To improve convergence speed, but also to avoid excessive overshoot, the predefined time $T_d = 0.5$ is selected. The shifting time T_s needs to satisfy $0 < T_s \leq T_d$, so $T_s = 0.5$ is selected. For the parameter ϕ_i in the auxiliary system, $\phi_i = 100$ is selected through trial-and-error. The time delay $\Delta = 0.001$ s is implemented via MATLAB/SimMechanics co-simulation with 1 kHz sampling frequency.

Remark 4.2. *The selection of TDE-SMC and BLF-SMC as comparative methods is reasonable. Regarding TDE-SMC, this method represents a classical model-free control method, similar to the proposed UPTAC, and it is also an ultra-local model-based controller that utilizes time-delay estimation techniques. Moreover, this method has been successfully applied in robotic systems [51]. The reason for choosing BLF-SMC as the comparison method is that it also uses BLF, which can form a comparison in the performance constraints section. Additionally, since this method does not account for the impact of random initial errors, it serves as an effective contrast to UPTAC, which incorporates a shifting function to address this issue.*

Under the specified simulation conditions and controller configurations, the obtained results are illustrated in Figures 4–9 and Table 3. Figure 4 illustrates the progression of movement capability parameter k_m and the adaptive training time per cycle. As evidenced in subfigure 4a, the patient's movement capability (quantified by k_m) exhibits improvement. Correspondingly, subfigure 4b demonstrates that the proposed algorithm dynamically adjusts the cycle duration based on assistive force feedback from prior cycles—reducing the time per training session from 6.0 to 3.14 s as motor function enhances. This progressive adaptation not only optimizes rehabilitation efficiency but also sustains patient engagement.



(a) **(b)**

Figure 4. (a): The value of parameter k_m; (b): The time required for each training in case 1.

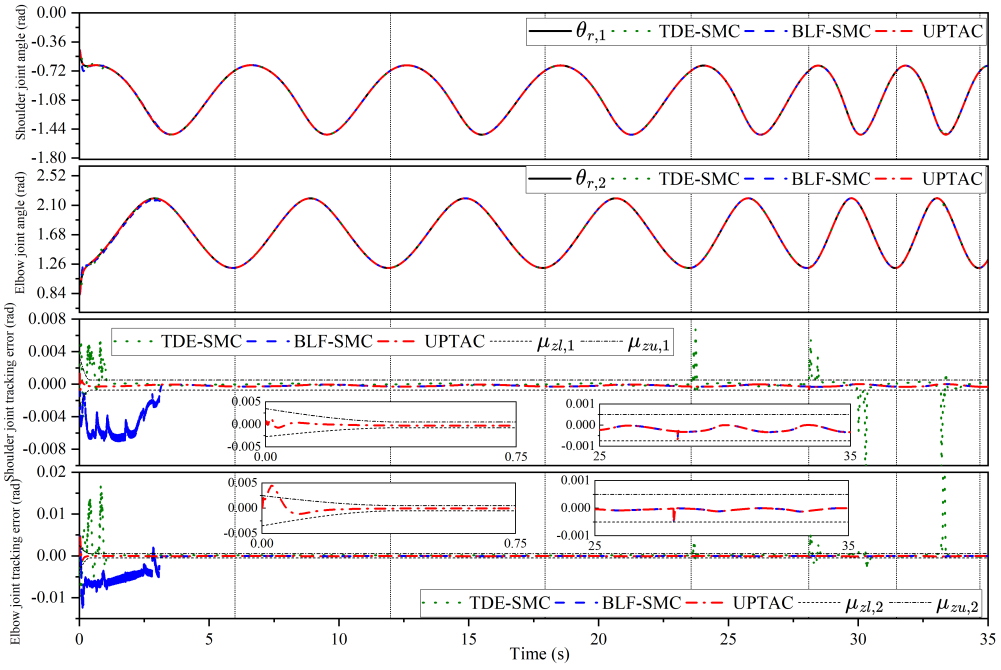


Figure 5. The tracking curves of θ_r of each method in case 1.

As evidenced in Figure 5, by applying the UPTAC, exoskeleton joints (red dashed lines) can accurately track the reference trajectory θ_r while satisfying performance constraints on tracking error. Steady-state errors converge to $(-7.5 \times 10^{-4}, 5 \times 10^{-4})$ (shoulder joint) and $(-5 \times 10^{-4}, 5 \times 10^{-4})$ (elbow joint). And, despite initial error violations of the performance constraints, the incorporation of a shifting function enables the proposed UPTAC to achieve rapid convergence. Furthermore, the proposed UPTAC guarantees predefined-time convergence since the tracking error remains within the performance constraints, and the definition of performance constraints indicates that the tracking error converges to a minimal interval within 0.5 seconds.

The comparative TDE-SMC method lacks error constraint mechanisms, resulting in significant steady-state error fluctuations, indirectly leading to fluctuations in assistive torque and causing impact on patients, resulting in decreased safety. Moreover, this method can only achieve finite time convergence, and in the case where the initial position of the patient's upper limb is random, its convergence time is relatively longer than the proposed method.

For the compared BLF-SMC, the poor initial performance can be attributed to the fact that BLF-based controllers require the initial error to be maintained within the prescribed performance constraints. However, the initial position of the patient's upper limb is often randomized and cannot guarantee such a condition. Furthermore, BLF-SMC lacks effective mechanisms—such as the shifting function—to compensate for these random initial errors, resulting in its poor control performance initially. Then, as shown in Table 3, the average error of the proposed UPTAC is only 70% of that of TDE-SMC and 30.4% of that of BLF-SMC.

Figure 6 compares the assistive torque tracking performance of three methods across shoulder and elbow joints. By comparing with Figure 4a, it can be seen that as the patient's movement

capability recovers, the required assistive torque gradually decreases. The proposed UPTAC method can accurately provide the required assistive torque, with fast convergence speed and small overshoot. However, the TDE-SMC may have some impact and pose a risk to patient safety. Although the BLF-SMC can stably provide the required auxiliary torque after 3 s, its convergence speed is slow. In addition, as shown in Table 3, the average tracking error of the proposed UPTAC on the expected assist torque is only 16.4% of TDE-SMC and 24.1% of BLF-SMC.

The simulation results of the patient's upper limb endpoint trajectory are shown in Figures 7 and 8. It is evident that the proposed UPTAC outperforms other methods in both overshoot and convergence speed. Moreover, as shown in Table 3, UPTAC achieves only 17.8% of TDE-SMC and 11.8% of BLF-SMC average tracking error in the x-axis trajectory, while its y-axis average tracking error is merely 62.9% of TDE-SMC and 25.2% of BLF-SMC.

Figure 9 gives the control input signals of the three methods, confirming that all methods are limited within the input constraints. Notably, the proposed UPTAC method has the smoothest control signals.

Table 3. The average absolute value of tracking error for each method in case 1.

	TDE-SMC	BLF-SMC	UPTAC
Reference trajectory $\theta_{r,1}$ tracking error	2.60E-4	5.98E-4	1.82E-4
Reference trajectory $\theta_{r,2}$ tracking error	4.38E-4	4.99E-4	5.11E-5
Assistive torque error in shoulder joint	1.25E-1	8.52E-2	2.05E-2
Assistive torque error in elbow joint	1.78E-1	1.02E-1	1.03E-2
Position error in x axis	8.77E-4	1.32E-3	1.56E-4
Position error in y axis	1.28E-3	3.19E-3	8.05E-4

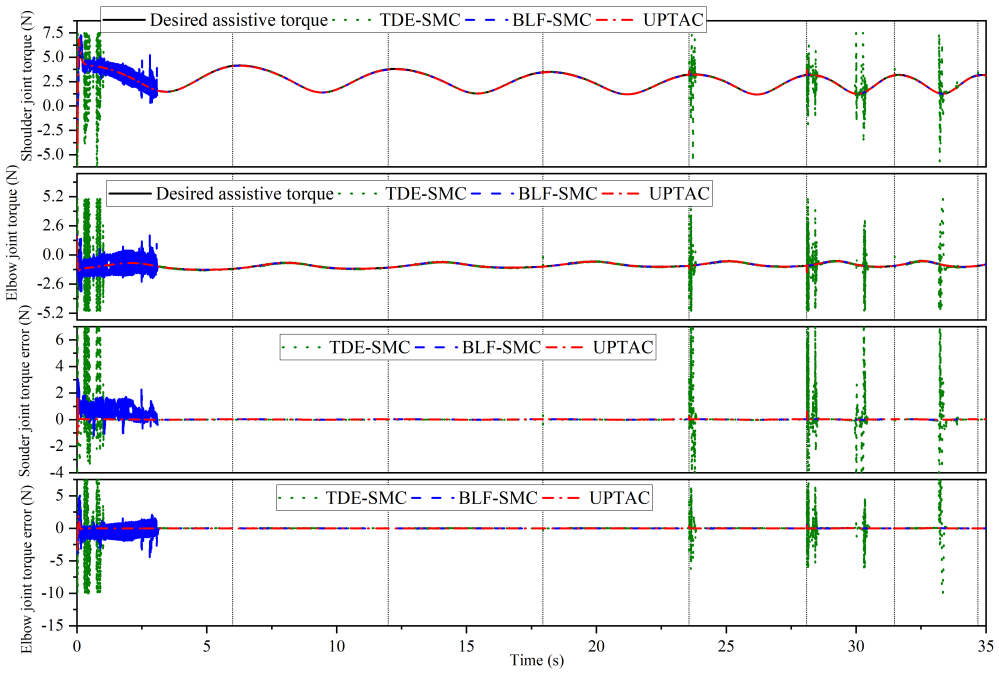


Figure 6. The assistive torque tracking curves of each method in case 1.

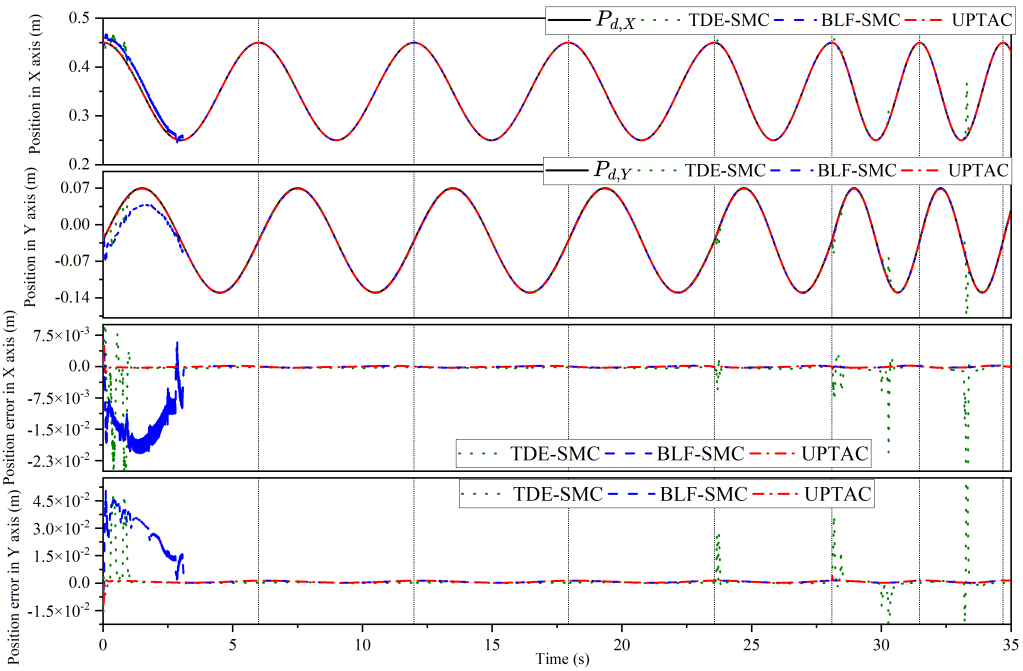


Figure 7. The trajectory tracking curves of the endpoint of patient's upper limb of each method in case 1.

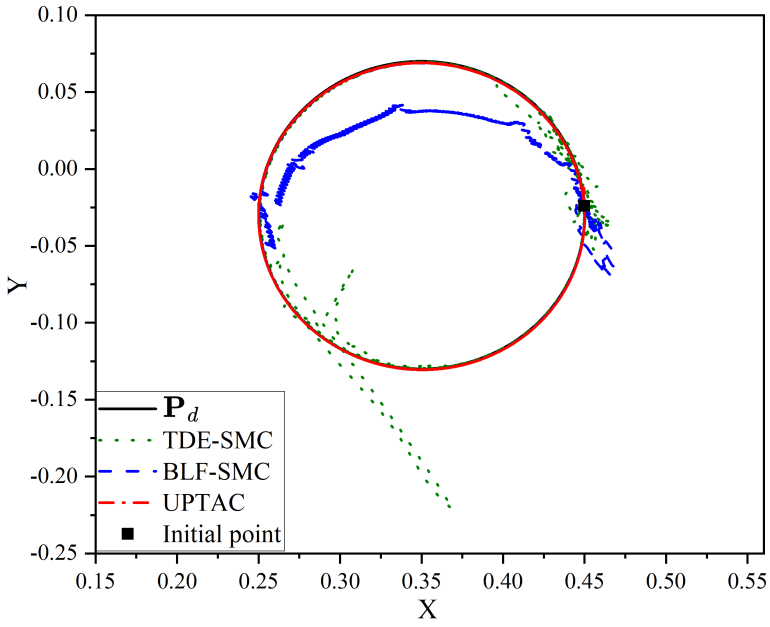


Figure 8. The motion trajectory of the patient's upper limb endpoint of each method in case 1.

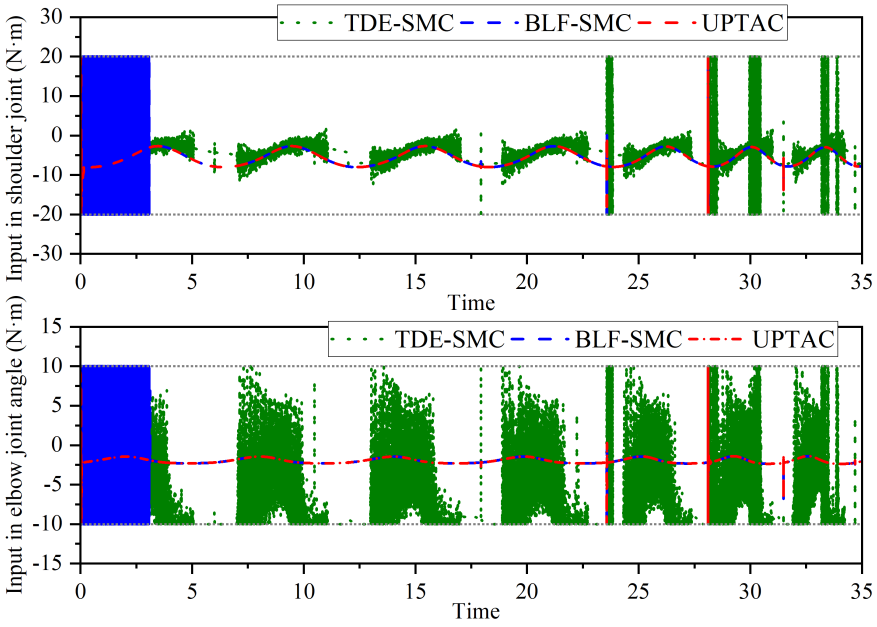


Figure 9. The control input signal of each method in case 1.

4.2. Case 2: Comparison results with iPD

In this case, a comparative analysis between the proposed UPTAC and the iPD method is conducted. The UPTAC controller is designed based on the ultra-local modeling approach, which was originally developed in conjunction with the iPD method [30]. The iPD method has the characteristic of simple structure, significantly reducing complexity in stability analysis and parameter tuning, making it an efficient and reliable control solution.

The referred iPD [52] is given as follows:

$$u = \frac{\hat{F} - \ddot{\theta}_r + K_p e + K_d \dot{e}}{\alpha}. \tag{4.3}$$

(i) The input constraints of each joint are set as: $\mu_{\tau,1} = 20$, $\mu_{\tau,2} = 10$.

The settings of the desired trajectory P_d, parameters of patient's upper limb, input constraints, and parameters of the proposed UPTAC are same as the Case 1. The parameters of iPD are chosen as $\alpha = 1$, $K_p = 4000$, $K_d = 50$.

Under the specified simulation conditions and controller configurations, the results are illustrated in Figures 10 and 11. From Figure 10, the compared iPD can only converge to a smaller interval after the 25th second, and there will be significant oscillations at the beginning of operation. By analyzing the control input signals in Figure 11, it can be concluded that the reason for the above situation is most likely due to the small input constraints. Therefore, in the presence of input constraints, the control effect of the proposed UPTAC is better than that of the iPD controller. Next, to further compare the control effects of the iPD method and the proposed UPTAC, the input constraints will be amplified.

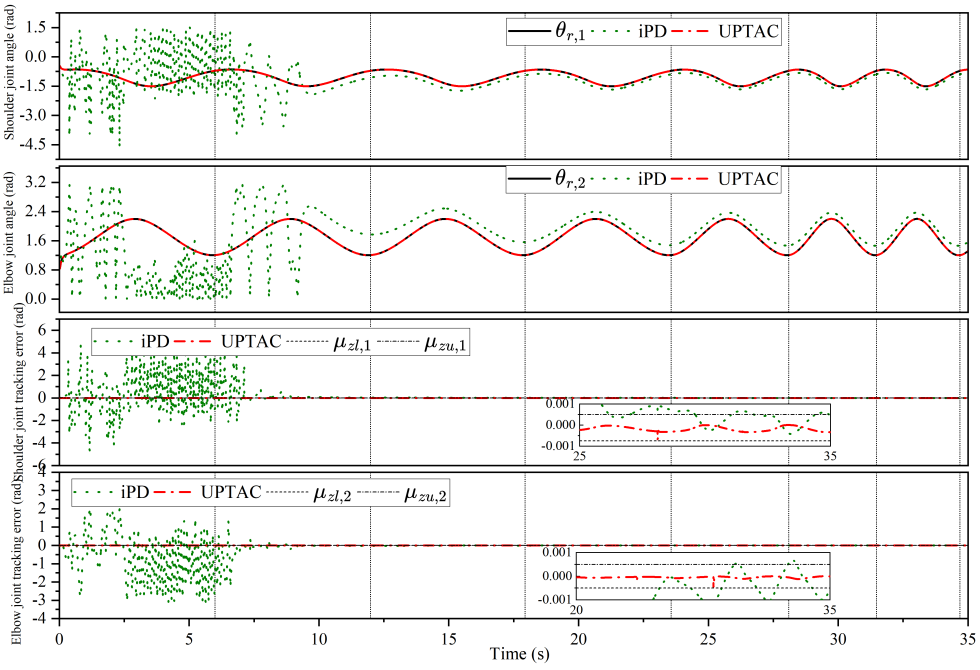


Figure 10. The tracking curves of θ_r of each method in case 2 (i).

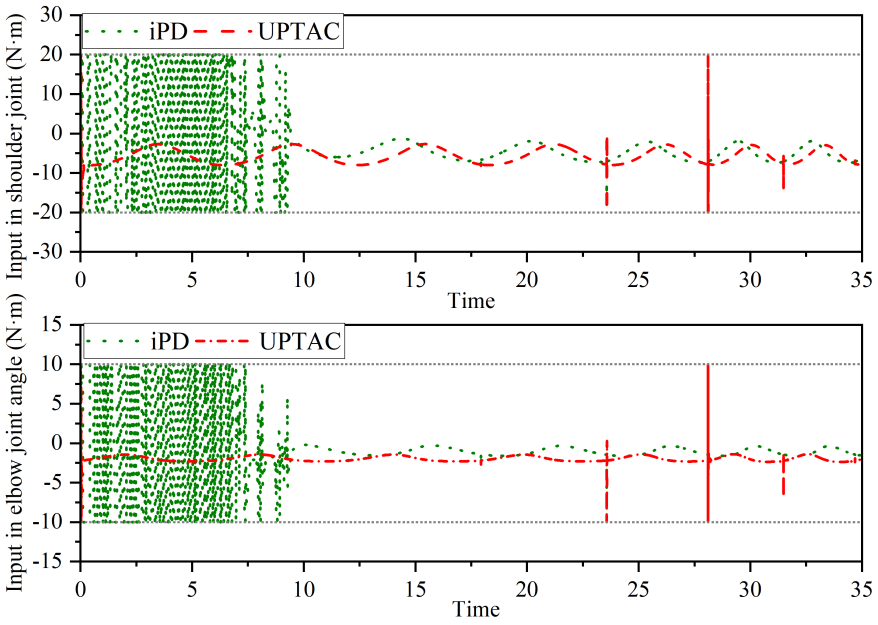


Figure 11. The control input signal of each method in case 2 (i).

(ii) The input constraints of each joint are set as: $\mu_{\tau,1} = 40$, $\mu_{\tau,2} = 20$.

In this case, the parameter selection for iPD and UPTAC is the same as (i).

Under the specified simulation conditions and controller configurations, the results are illustrated in Figures 12 and 13. As shown in Figure 12, after the input constraints are amplified, the convergence speed of the iPD controller is significantly improved, reaching a small error range within two seconds. However, there will be fluctuations between every two rehabilitation cycles, that is, between two circular movements of the patient's upper limb, which may pose potential safety risks. In contrast, although the proposed UPTAC also exhibits fluctuations between cycles, their amplitude is very small and its impact on patients is minimal. Therefore, the proposed UPTAC is more suitable for rehabilitation training compared to the iPD controller.

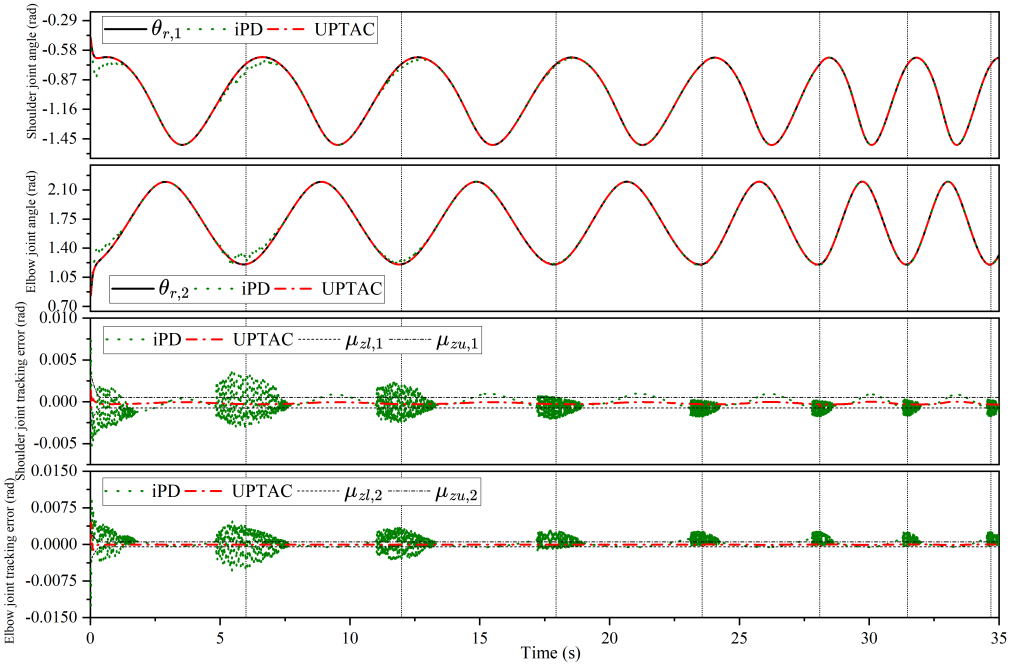


Figure 12. The tracking curves of θ_r of each method in case 2 (ii).

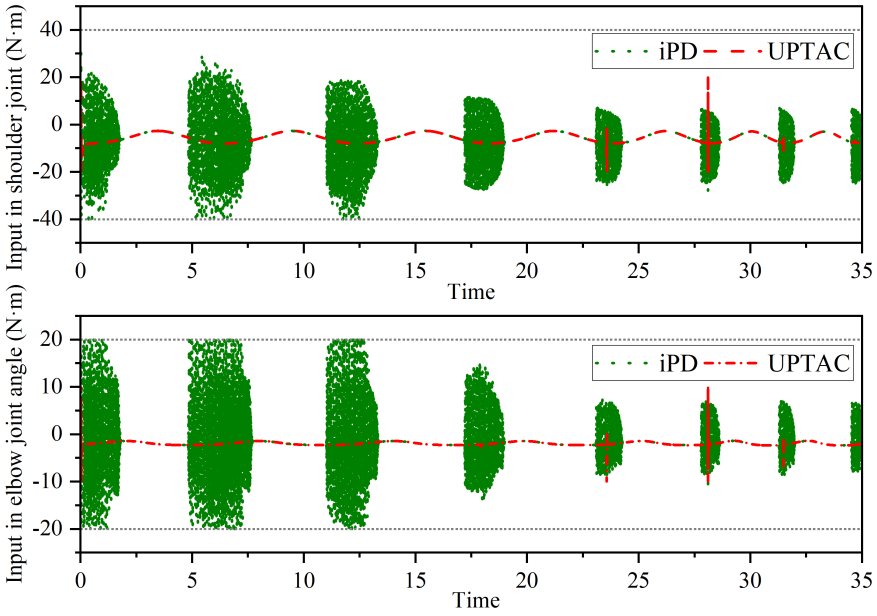


Figure 13. The control input signal of each method in case 2 (ii).

Considering that the proposed UPTAC involves many parameters and its stability proof is complex, the iPD controller will be incorporated in subsequent work to simplify the controller structure and reduce the complexity of the proof process.

4.3. Case 3: Simulation results with uncertainties

In actual rehabilitation training, there may be uncertain factors such as model parameter uncertainties, patient's muscle shaking, and exoskeleton measurement noise. The uncertainty of model parameters is set to account for the uncertainty in the mass of the patient's and exoskeleton's upper limbs, as detailed in Table 4. The torque caused by the patient's muscle shaking is set as $0.075\sin(8\pi t)$, and the measurement noise is set as white noise between the intervals $[-1\times10^{-4}, 1\times10^{-4}]$.

Table 4. The model parameter uncertainties of ULPES model.

Part name	Mass (kg)
Exoskeleton upper arm	$2.357\times(1+0.2\times(\text{rand}(1)-0.5))$
Exoskeleton forearm	$1.177\times(1+0.2\times(\text{rand}(1)-0.5))$
Patient upper arm	$1.530\times(1+0.2\times(\text{rand}(1)-0.5))$
Patient forearm	$1.084\times(1+0.2\times(\text{rand}(1)-0.5))$

In this case, more complex upper limb movement trajectories are considered to better simulate real rehabilitation movements. The trajectory is set as follows:

$$\begin{cases} b = 0.15\sin(4\pi t/k_d), \\ P_{d,X} = 0.35 + b\sin(2\pi t/k_d), \\ P_{d,Y} = b\cos(2\pi t/k_d). \end{cases}$$

Due to the complexity of the trajectory, the prescribed constraints for tracking errors are set as follows:

$$\mu_{el,1}(t) = \begin{cases} (3\times10^{-3} - \frac{t}{0.5})e^{1-\frac{0.5}{0.5-t}} + 1\times10^{-3} & 0 \le t \le 0.5, \\ 1\times10^{-3} & t > 0.5, \end{cases}$$

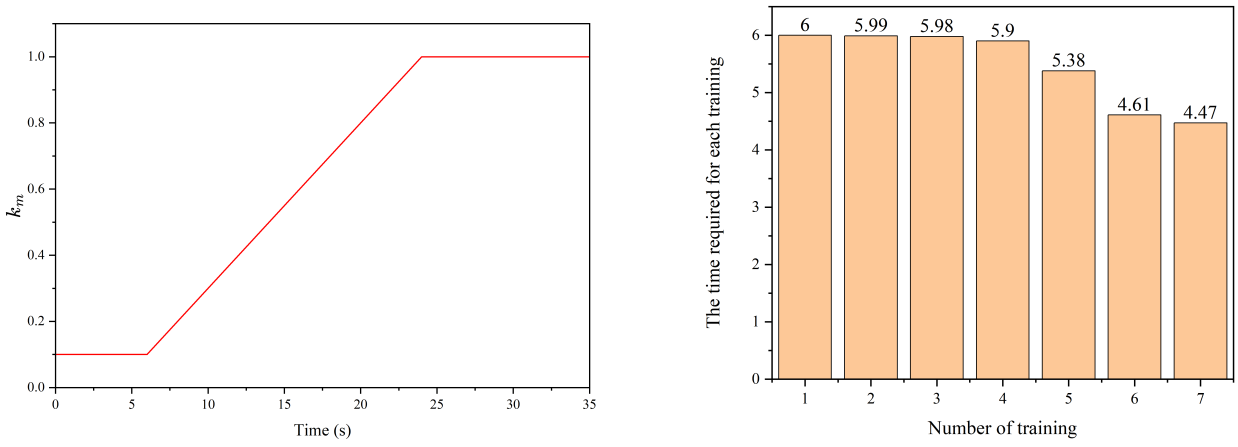
$$\mu_{el,2}(t) = \begin{cases} (3\times10^{-3} - \frac{t}{0.5})e^{1-\frac{0.5}{0.5-t}} + 1\times10^{-3} & 0 \le t \le 0.5, \\ 1\times10^{-3} & t > 0.5, \end{cases}$$

$$\mu_{eu,1}(t) = \begin{cases} (3\times10^{-3} - \frac{t}{0.5})e^{1-\frac{0.5}{0.5-t}} + 5\times10^{-4} & 0 \le t \le 0.5, \\ 5\times10^{-4} & t > 0.5, \end{cases}$$

$$\mu_{eu,2}(t) = \begin{cases} (2\times10^{-3} - \frac{t}{0.5})e^{1-\frac{0.5}{0.5-t}} + 5\times10^{-4} & 0 \le t \le 0.5, \\ 5\times10^{-4} & t > 0.5. \end{cases}$$

The parameters of UPTAC are selected as $\varsigma_2 = 60$. The selection of other parameters is the same as in case 1.

Under the above simulation conditions, the corresponding simulation results are demonstrated in Figures 14–19. As demonstrated in Figure 14, the proposed UPTAC dynamically adjusts the cycle duration based on assistive torque feedback from previous rehabilitation cycles. Notably, the designed task performance function maintains effectiveness in optimizing rehabilitation efficiency despite the presence of uncertain factors.



(a) **(b)**

Figure 14. (a): The value of parameter k_m; (b): The time required for each training.

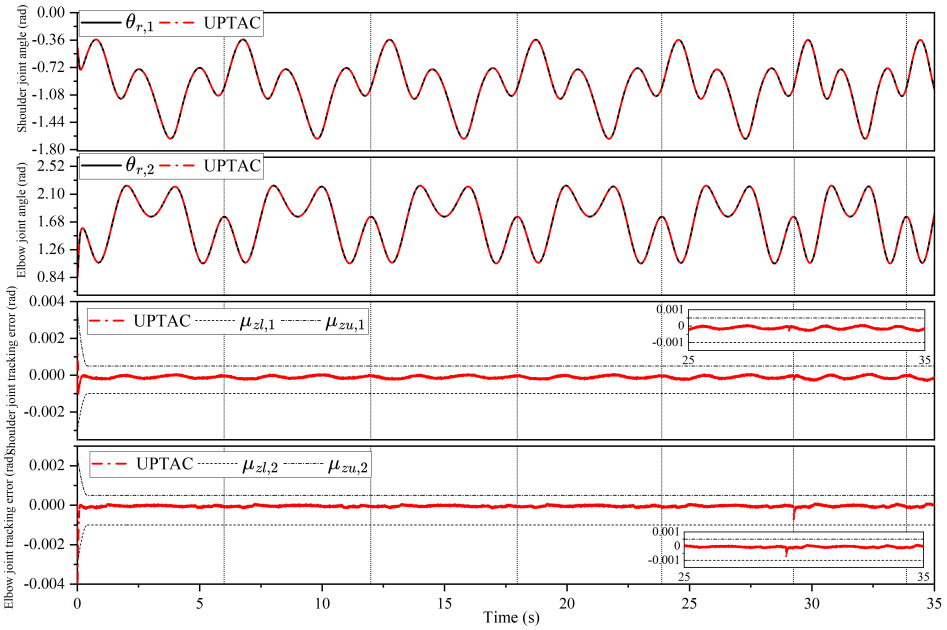


Figure 15. The tracking curves of θ_r by UPTAC in case 3.

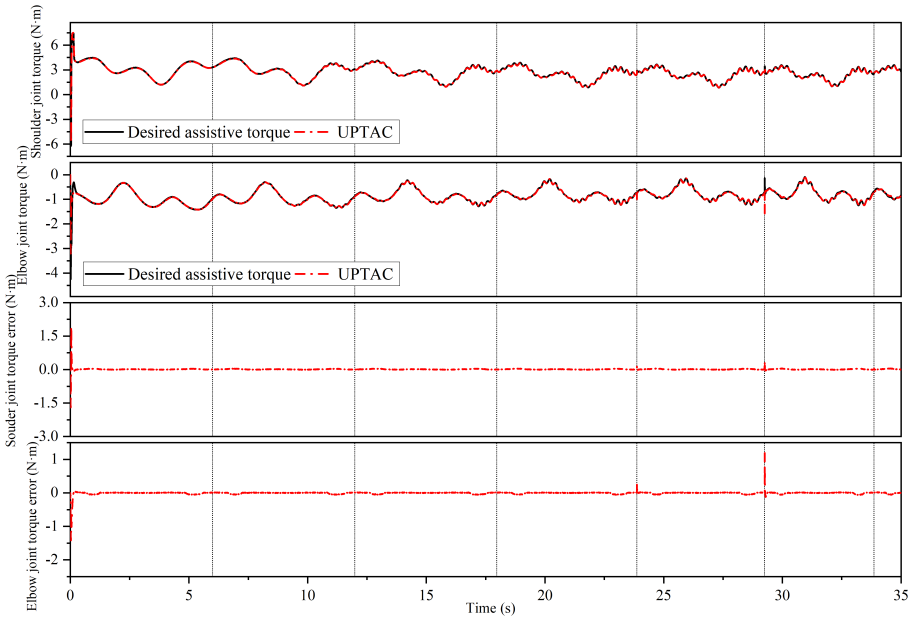


Figure 16. The assistive torque tracking curves by UPTAC in case 3.

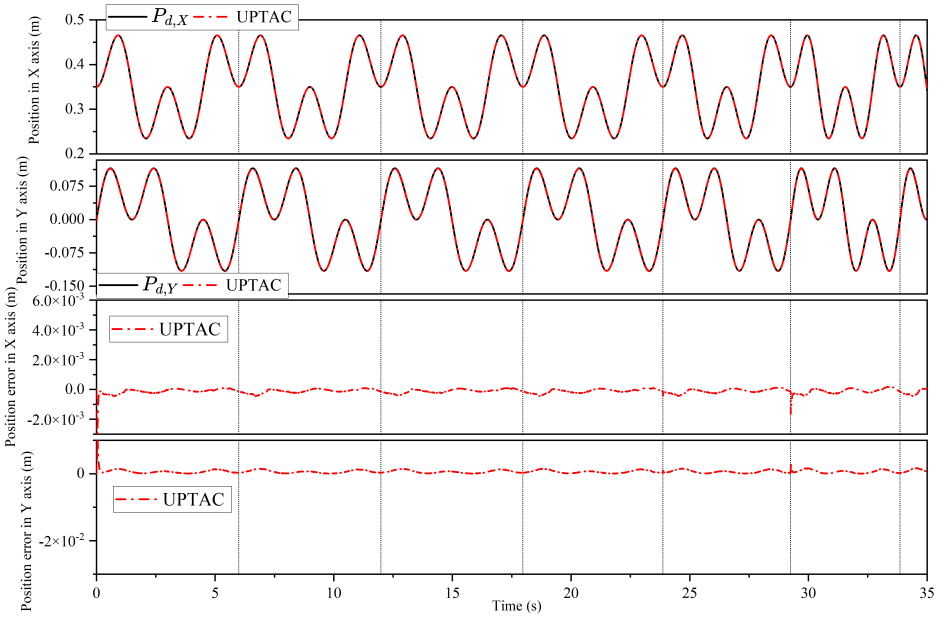


Figure 17. The trajectory tracking curves of the endpoint of patient's upper limb by UPTAC in case 3.

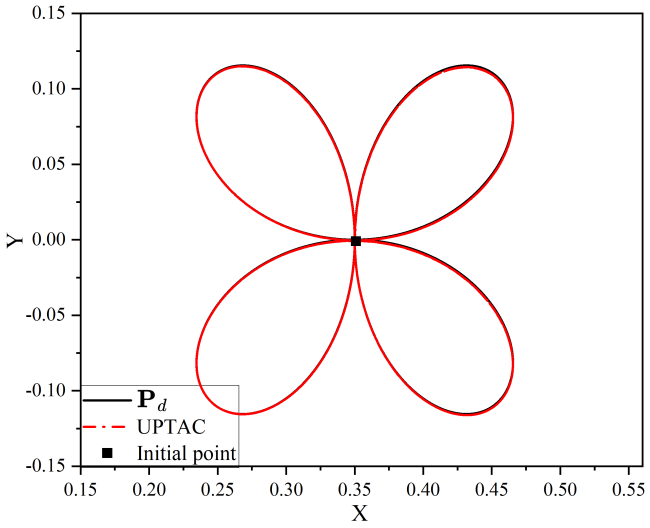


Figure 18. The motion trajectory of the patient's upper limb endpoint by UPTAC in case 3.

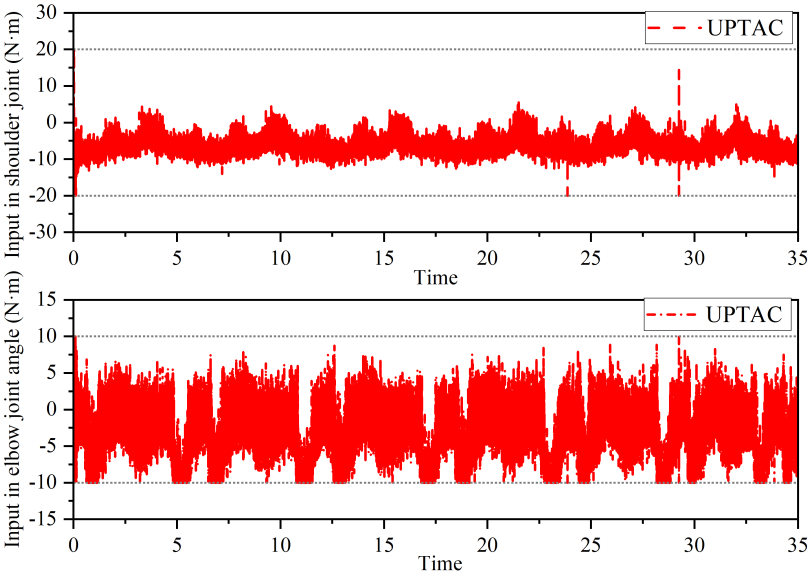


Figure 19. The control input signal by UPTAC in case 3.

As illustrated in Figure 15, the proposed UPTAC enables the exoskeleton joints (red dashed lines) to accurately track the reference trajectory θ_r while satisfying performance constraints on tracking errors, despite the presence of uncertainties. The steady-state errors converge to $(-1 \times 10^{-3}, 5 \times 10^{-4})$ within the predefined time T_d.

A transient fluctuation occurs at $t = 29.25$s when the patient initiates a new rehabilitation cycle, caused by the significant variation in k_d between consecutive training sessions. Since k_d directly affects the desired trajectory and its second derivative is utilized in the controller, this parameter change induces momentary disturbances. Future work will focus on smoothing the transition between two rehabilitation cycles through k_d adaptation.

Figure 16 demonstrates the assistive torque tracking performance for both shoulder and elbow joints. The proposed UPTAC method can maintain precise assistive torque transmission even in the presence of muscle shaking, ensuring the exoskeleton provides the required assistive torque. Furthermore, as evidenced in Figures 17-18, the endpoint trajectory of the patient's upper limb can track the desired rehabilitation path even in the presence of uncertainties, enabled by the accurate assistive torque from the exoskeleton.

5. Conclusions

In this paper, an UPTAC method is proposed for the ULPES under input and performance constraints. The developed UPTAC implements assist-as-needed control via an impedance control approach, incorporating a task performance function that dynamically adjusts rehabilitation movement speed based on the patient's recovery progress. The inner torque sub-control loop of UPTAC is designed based on the ultra-local model, which reduces both dependency on precise system modeling and controller complexity. Furthermore, by employing BLF with specific designed performance constraints, the system guarantees predefined-time convergence. Comparative co-simulations with

TDE-SMC and BLF-SMC demonstrate that UPTAC not only realizes predefined-time convergence but also exhibits minimal overshoot and faster convergence time. Quantitatively, UPTAC achieves average tracking errors of only 16.4% of TDE-SMC and 24.1% of BLF-SMC, confirming its superior performance.

There are still some limitations in this paper. Considering that the proposed UPTAC involves many parameters and its stability proof is complex, subsequent work will incorporate iPD control [52] for simplification. Additionally, the command-filtered backstepping technique [53,54] will be employed to further mitigate complexity in the control design, ensuring robustness while maintaining performance. Then, to address random initial errors from uncertain limb positioning, while shifting function offers a solution, it increases controller complexity. Fixed-time control [55] enables initial-state-independent convergence, which can better handle the above problems, motivating its adoption in future designs.

Author contributions

Honglin Xie: Conceptualization, software, validation, formal analysis, investigation, data curation, writing—original draft preparation, project administration, funding acquisition; Yangchun Wei: Conceptualization, methodology, validation, resources, writing—review and editing, visualization, supervision. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they did not use Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was funded by Shandong Women's University School Project grant number 2022RCYJ12.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. Y. Wang, Y. Tian, Y. Guo, H. P. Wang, Active torque-based gait adjustment multi-level control strategy for lower limb patient–exoskeleton coupling system in rehabilitation training, *Math. Comput. Simul.*, **215** (2024), 357–381. https://doi.org/10.1016/j.matcom.2023.08.020
2. P. Delgado, Y. Yihun, Integration of task-based exoskeleton with an assist-as-needed algorithm for patient-centered elbow rehabilitation, *Sensors*, **23** (2023), 2460. https://doi.org/10.3390/s23052460
3. Y. Wang, H. P. Wang, Y. Tian, Adaptive interaction torque-based AAN control for lower limb rehabilitation exoskeleton, *ISA Trans.*, **128** (2022), 184–197. https://doi.org/10.1016/j.isatra.2021.10.009

4. K. Gui, U. X. Tan, H. Liu, D. Zhang, Electromyography-driven progressive assist-as-needed control for lower limb exoskeleton, *IEEE Trans. Med. Robot. Bionics*, **2** (2020), 50–58. https://doi.org/10.1109/TMRB.2020.2970222

5. Y. Wang, Y. Guo, Y. Tao, Y. Tian, H. P. Wang, Human-centered active torque-based AAN impedance control for lower limb patient-exoskeleton coupling system in the rehabilitation, *Int. J. Robust Nonlinear Control*, **35** (2025), 4096–4131. https://doi.org/10.1002/rnc.6996

6. B. Guo, Z. Li, M. Huang, X. Li, J. Han, Patient's healthy-limb motion characteristic-based assist-as-needed control strategy for upper-limb rehabilitation robots, *Sensors*, **24** (2024), 2082. https://doi.org/10.3390/s24072082

7. J. C. Castiblanco, I. F. Mondragon, C. Alvarado-Rojas, J. D. Colorado, Assist-as-needed exoskeleton for hand joint rehabilitation based on muscle effort detection, *Sensors*, **21** (2021), 4372. https://doi.org/10.3390/s21134372

8. X. Tian, J. Lin, H. Liu, X. Huang, Event-triggered finite-time formation control of underactuated multiple ASVs with prescribed performance and collision avoidance, *Sensors*, **23** (2023), 6756. https://doi.org/10.3390/s23156756

9. L. Fang, L. Ma, S. Ding, Finite-time fuzzy output-feedback control for p-norm stochastic nonlinear systems with output constraints, *AIMS Math.*, **6** (2021), 2244–2267. https://doi.org/10.3934/math.2021136

10. Y. Li, D. Wang, Z. Cai, On asymptotic fixed-time controller design for uncertain nonlinear systems with pure state constraints, *AIMS Math.*, **8** (2023), 27151–27174. https://doi.org/10.3934/math.20231389

11. K. P. Tee, S. S. Ge, E. H. Tay, Barrier Lyapunov functions for the control of output-constrained nonlinear systems, *Automatica*, **45** (2009), 918–927. https://doi.org/10.1016/j.automatica.2008.11.017

12. Y. Wei, H. P. Wang, Y. Tian, Prescribed performance model-free hybrid force/position control for 3-DOF SEA-based manipulator under partial state constraints, *J. Franklin Inst.*, **361** (2024), 106944. https://doi.org/10.1016/j.jfranklin.2024.106944

13. Y. Wei, H. P. Wang, Y. Tian, Asymmetric time-varying BLF-based model-free hybrid force/position control for SEA-based 2-DOF manipulator, *Int. J. Adapt. Control Signal Process.*, **37** (2023), 1716–1737. https://doi.org/10.1002/acs.3599

14. Y. Wei, H. P. Wang, Y. Tian, Fixed-time asymmetric barrier Lyapunov function-based model-free hybrid position/force control for 3-DOF SEA-based manipulator with output constraints, *Int. J. Robust Nonlinear Control*, **34** (2024), 1324–1342. https://doi.org/10.1002/rnc.7032

15. D. Tian, X. Song, Addressing complex state constraints in the integral barrier Lyapunov function-based adaptive tracking control, *Int. J. Control*, **96** (2023), 1202–1209. https://doi.org/10.1080/00207179.2022.2036371

16. H. Xue, Y. Ou, A novel asymmetric barrier Lyapunov function-based fixed-time ship berthing control under multiple state constraints, *Ocean Eng.*, **281** (2023), 114756. https://doi.org/10.1016/j.oceaneng.2023.114756

17. Y. H. Liu, L. L. Chen, Q. Zhou, C. Y. Su, Asymptotic output tracking control with prescribed transient performance of nonlinear systems in the presence of unknown dynamics, *Int. J. Robust Nonlinear Control*, **32** (2022), 9363–9379. https://doi.org/10.1002/rnc.6336

18. T. Zhang, P. Yan, Asymmetric integral barrier function-based tracking control of constrained robots, *AIMS Math.*, **9** (2024), 319–339. https://doi.org/10.3934/math.2024019

19. L. Zhi, J. Wu, Adaptive constraint control for nonlinear multi-agent systems with undirected graphs, *AIMS Math.*, **6** (2021), 12051–12064. https://doi.org/10.3934/math.2021698

20. X. Liang, C. Xu, D. Wang, Adaptive neural network control for marine surface vehicles platoon with input saturation and output constraints, *AIMS Math.*, **5** (2020), 587–602. https://doi.org/10.3934/math.2020039

21. Y. D. Song, S. Zhou, Tracking control of uncertain nonlinear systems with deferred asymmetric time-varying full state constraints, *Automatica*, **98** (2018), 314–322. https://doi.org/10.1016/j.automatica.2018.09.032

22. K. Shao, J. Zheng, Predefined-time sliding mode control with prescribed convergent region, *IEEE/CAA J. Autom. Sinica*, **9** (2022), 934–936. https://doi.org/10.1109/jas.2022.105575

23. J. Shao, W. W. Che, K. Shao, Nonlinear prescribed performance sliding mode control of hypersonic vehicles, *Int. J. Robust Nonlinear Control*, **34** (2024), 9928–9948. https://doi.org/10.1002/rnc.7503

24. S. Xie, Q. Chen, Adaptive nonsingular predefined-time control for attitude stabilization of rigid spacecrafts, *IEEE Trans. Circuits Syst. II*, **69** (2022), 189–193. https://doi.org/10.1109/TCSII.2021.3078708

25. Y. Zhu, J. Qiao, L. Guo, Adaptive sliding mode disturbance observer-based composite control with prescribed performance of space manipulators for target capturing, *IEEE Trans. Ind. Electron.*, **66** (2019), 1973–1983. https://doi.org/10.1109/TIE.2018.2838065

26. D. Liu, Z. Liu, C. L. P. Chen, Y. Zhang, Prescribed-time containment control with prescribed performance for uncertain nonlinear multi-agent systems, *J. Franklin Inst.*, **358** (2021), 1782–1811. https://doi.org/10.1016/j.jfranklin.2020.12.021

27. D. Ye, A. M. Zou, Z. Sun, Predefined-time predefined-bounded attitude tracking control for rigid spacecraft, *IEEE Trans. Aerosp. Electron. Syst.*, **58** (2022), 464–472. https://doi.org/10.1109/TAES.2021.3103258

28. Y. Wang, H. P. Wang, Y. Tian, Nonlinear disturbance observer based flexible-boundary prescribed performance control for a lower limb exoskeleton, *Int. J. Syst. Sci.*, **52** (2021), 3176–3189. https://doi.org/10.1080/00207721.2021.1922952

29. C. Ding, S. Ding, K. Mei, Adaptive prescribed-time SOSM controller design for nonlinear systems with prescribed performance, *IEEE Trans. Circuits Syst. II*, **71** (2024), 1311–1315. https://doi.org/10.1109/TCSII.2023.3323661

30. M. Fliess, C. Join, Model-free control, *Int. J. Control*, **86** (2013), 2228–2252. https://doi.org/10.1080/00207179.2013.810345

31. X. Zhang, H. Wang, Y. Tian, L. Peyrodie, X. Wang, Model-free based neural network control with time-delay estimation for lower extremity exoskeleton, *Neurocomputing*, **272** (2018), 178–188. https://doi.org/10.1016/j.neucom.2017.06.055

32. Y. Wei, H. Wang, Y. Tian, Adaptive sliding mode observer–based integral sliding mode model-free torque control for elastomer series elastic actuator–based manipulator, *Proc. Inst. Mech. Eng., Part I: J. Syst. Control Eng.*, **236** (2022), 1010–1028. https://doi.org/10.1177/09596518211064757

33. D. He, X. Ma, Y. Tian, H. Wang, Model-free fractional-order finite-time control with prescribed performance for mechatronic systems under actuator failure, *Nonlinear Dyn.*, **113** (2025), 2465–2486. https://doi.org/10.1007/s11071-024-10353-z

34. Y. Wei, H. P. Wang, Y. Tian, Ultra-local model-based prescribed-time hybrid force/position control for 3-DOF series elastic actuator-based manipulator under input and output constraints, *Commun. Nonlinear Sci. Numer. Simul.*, **150** (2025), 109013. https://doi.org/10.1016/j.cnsns.2025.109013

35. S. Han, H. Wang, Y. Tian, H. Yu, Enhanced extended state observer-based model-free force control for a series elastic actuator, *Mech. Syst. Signal Process.*, **183** (2023), 109584. https://doi.org/10.1016/j.ymssp.2022.109584

36. G. I. Mustafa, H. Wang, A new adaptive fuzzy logic control for nonlinear car active suspension systems based on the time-delay, *J. Vib. Control.*, 2024. https://doi.org/10.1177/10775463241281395

37. D. He, H. Wang, Y. Tian, K. Zimenko, Event-triggered discrete extended state observer–based model-free controller for quadrotor position and attitude trajectory tracking, *Proc. Inst. Mech. Eng., Part I: J. Syst. Control Eng.*, **236** (2022), 754–771. https://doi.org/10.1177/09596518211055362

38. Y. Wei, H. P. Wang, Y. Tian, Ultra-local model-based prescribed performance assist-as-needed control for series elastic actuator-based upper limb patient-exoskeleton system under complex state constraints, *Nonlinear Dyn.*, **112** (2024), 17183–17204. https://doi.org/10.1007/s11071-024-09928-7

39. Y. Wei, H. P. Wang, Y. Tian, Adaptive time-varying barrier Lyapunov function-based model-free hybrid position/force control for series elastic actuator-based manipulator, *IEEE Trans. Circuits Syst. II*, **71** (2024), 271–275. https://doi.org/10.1109/TCSII.2023.3297600

40. K. P. Tee, E. Burdet, C. M. Chew, T. E. Milner, A model of force and impedance in human arm movements, *Biol. Cybern.*, **90** (2004), 368–375. https://doi.org/10.1007/s00422-004-0484-4

41. Y. Guo, H. Wang, Y. Tian, D. G. Caldwell, Task performance-based adaptive velocity assist-as-needed control for an upper limb exoskeleton, *Biomed. Signal. Proces.*, **73** (2022), 103474. https://doi.org/10.1016/j.bspc.2021.103474

42. H. Wang, X. Ye, Y. Tian, G. Zheng, N. Christov, Model-free–based terminal SMC of quadrotor attitude and position, *IEEE Trans. Aerosp. Electron. Syst.*, **52** (2016), 2519–2528. https://doi.org/10.1109/TAES.2016.150303

43. H. Wang, Y. Tian, S. Ni, N. Christov, Intelligent proportional trajectory tracking controllers: Using ultra-local model and time delay estimation techniques, In: *The 27th Chinese control and decision conference (2015 CCDC)*, 2015, 430–435. https://doi.org/10.1109/CCDC.2015.7161731

44. Z. Chen, J. Wang, K. Ma, X. Huang, T. Wang, Fuzzy adaptive two-bits-triggered control for nonlinear uncertain system with input saturation and output constraint, *Int. J. Adapt. Control Signal Process.*, **34** (2020), 543–559. https://doi.org/10.1002/acs.3098

45. Y. Wei, H. P. Wang, Y. Tian, Shifting asymmetric time-varying BLF-based model-free hybrid force/position control for 3-DOF SEA-based manipulator with random initial error, *Appl. Math. Comput.*, **463** (2024), 128363. https://doi.org/10.1016/j.amc.2023.128363

46. C. C. Chen, Z. Y. Sun, A unified approach to finite-time stabilization of high-order nonlinear systems with an asymmetric output constraint, *Automatica*, **111** (2020), 108581. https://doi.org/10.1016/j.automatica.2019.108581

47. S. Shi, H. Min, Y. Hu, Y. Sun, B. Wang, A novel hybrid scheme for fixed-time SOSM control of nonlinear uncertain systems subject to mismatched terms, *Appl. Math. Comput.*, **386** (2020), 125511. https://doi.org/10.1016/j.amc.2020.125511

48. S. Ding, J. H. Park, C. C. Chen, Second-order sliding mode controller design with output constraint, *Automatica*, **112** (2020), 108704. https://doi.org/10.1016/j.automatica.2019.108704

49. Z. Zhu, Y. Xia, M. Fu, Attitude stabilization of rigid spacecraft with finite-time convergence, *Int. J. Robust Nonlinear Control*, **21** (2011), 686–702. https://doi.org/10.1002/rnc.1624

50. C. Zhao, L. Guo, PID controller design for second order nonlinear uncertain systems, *Sci. China Inf. Sci.*, **60** (2017), 022201. https://doi.org/10.1007/s11432-016-0879-3

51. Y. Wang, S. Li, D. Wang, F. Ju, B. Chen, H. Wu, Adaptive time-delay control for cable-driven manipulators with enhanced nonsingular fast terminal sliding mode, *IEEE Trans. Ind. Electron.*, **68** (2021), 2356–2367. https://doi.org/10.1109/tie.2020.2975473

52. M. Fliess, C. Join, An alternative to proportional-integral and proportional-integral-derivative regulators: Intelligent proportional-derivative regulators, *Int. J. Robust Nonlinear Control*, **32** (2022), 9512–9524. https://doi.org/10.1002/rnc.5657

53. G. Yang, State filtered disturbance rejection control, *Nonlinear Dyn.*, **113** (2025), 6739–6755. https://doi.org/10.1007/s11071-024-10449-6

54. G. Yang, J. Yao, Multilayer neurocontrol of high-order uncertain nonlinear systems with active disturbance rejection, *Int. J. Robust Nonlinear Control*, **34** (2024), 2972–2987. https://doi.org/10.1002/rnc.7118

55. Z. Liu, O. Zhang, Y. Gao, Y. Zhao, Y. Sun, J. Liu, Adaptive neural network-based fixed-time control for trajectory tracking of robotic systems, *IEEE Trans. Circuits Syst. II*, **70** (2023), 241–245. https://doi.org/10.1109/TCSII.2022.3194917

AIMS Press