End-to-End Assistive Torque Control for Lower-Limb Exoskeleton via Sim-to-Real Reinforcement Learning and Curriculum Learning

***Abstract*—Most existing exoskeleton controllers are designed based on a hierarchical architecture with strong prior assumptions. Although these methods have achieved promising results at a certain level of control, an end-to-end exoskeleton control policy that outputs the optimal assistive torque at each actuation step remains underdeveloped. To address this gap, we propose a reinforcement learning approach for an end-to-end exoskeleton controller that directly maps onboard sensor features to motor commands, without any prior assumptions about the control structure. The proposed approach first learns the optimal assistive policy from the interaction with a musculoskeletal human model, aiming to effectively minimize human effort at each actuation step. The learned device-agnostic exoskeleton control policy is then applied to physical exoskeletons through a zero-shot sim-to-real transfer strategy. This approach alleviates potential discontinuities that commonly arise from reliance on ambulation mode recognition and gait phase estimation. The effectiveness of the proposed framework has been validated on a hip exoskeleton with experiments involving three healthy subjects. [bolei: Here it will be good to highlight some quantitative results, for example: The experiments show that the proposed approach outperforms prior xyz methods with % improvement over energy consumption and balbal.]**

***Index Terms*—End-to-end exoskeleton control, sim-to-real deep reinforcement learning**

I. Introduction

Exoskeletons hold great promise for assisting the elderly and individuals with limited mobility, as well as serving as practical rehabilitation tools for patients with gait impairments [1]. However, their widespread adoption beyond laboratory environments—particularly in community and real-world settings—remains limited due to challenges in both hardware and control algorithms [2], [3]. While significant efforts have been made to design and optimize exoskeleton mechanisms and actuation systems to enhance portability, reduce weight, and increase power efficiency [4], [5]. However, the development of an autonomous control strategy to robustly deliver the assistive torque is missing in the literature. To date, most exoskeleton controllers were extensively developed following a hierarchical architecture [6], wherein a high-level controller is responsible for recognizing the user's intent or environment context, followed by a mid-level controller that generates an appropriate reference for low-level controllers, which is related to the detected ambulation mode from users or environmental states. Given the derived reference, the low-level controllers, direct torque control, impedance control, and position tracking control, compute motor commands according to system feedback. However, the reliance on accurate gait cycle detection and handcrafted parameter tuning at each control level has significantly constrained the adaptability and scalability of such hierarchical controllers.

Most gait-cycle based controllers were developed to derive the assistive torque aligning with the detected gait phase variable (see Fig. 1). Over the past years, most researchers have focused on the high-level controller by recognizing the ambulation modes from users' states [2], [7], e.g., walking speed or environment states [8], [9], e.g., ramp slope. Notably, the advancements in model-free methods based on machine learning methods, including deep neural networks (DNNs), convolutional neural networks (CNNs) [10], long short-term memory networks (LSTMs) [11], [12], and temporal convolutional networks (TCNs) [13], have provided a promising solution for inferring locomotion modes from the collected onboard sensing data, which then were integrated into the mid-level controller for reference design [8]. For instance, model-dependent off-the-shelf spline controllers [14] worked on time-based gait phase estimation (TBE) method to generate the assistive torque with fixed formulation. Afterwards, human-in-the-loop optimization utilizes the probabilistic learning method to optimize a prescribed reference joint torque [15], [16], defined by timing and magnitude values, based on human biomechanical or kinematic feedback. Additionally, human preference-based learning algorithms have been proposed for personalizing exoskeleton assistance based on high-level human feedback [17], [18]. However, these methods did not consider the modeling and dynamics of human-robot interaction. Therefore, they need to re-optimize the parameters of mid-level controller gains for each unseen environment state [2]. A policy iteration method, using least squares regression, was integrated with a finite state machine impedance controller (FSM-IC) as a high-level controller to derive a personalized assistive strategy [19]. However, the biggest limitation of existing hierarchical controllers is that they cannot address the locomotion transition, which is crucial for community activities.

Some non-gait-cycle based controllers were recently developed to mitigate the dependence on gait phase detection and environment state recognition (see Fig. 1). In [20], [21], a time-delay output feedback controller was developed to deliver smooth assistance by responding to changes in leg motion instead of detecting the exact gait phase. However, the effectiveness of this method remains sensitive to the careful tuning of parameters such as the smoothing factor, time delay, and feedback gains. Most recently, data-driven learning-based approaches have emerged to address this gap by predicting unified joint moment rather than classifying the locomotion modes [3], [22]. In these methods, the assistive torque is manually designed from biological joint moment, enabling

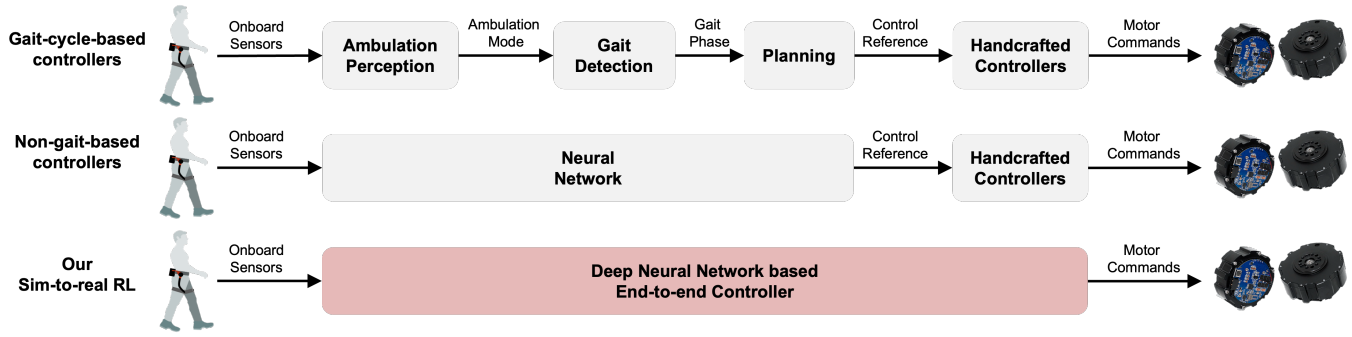


Fig. 1. Most existing state-of-the-art controllers rely on gait phase detection to generate the control reference for implementing handcrafted low-level controllers. Although some supervised learning approaches have utilized neural networks to bypass ambulation mode recognition and gait phase detection, these methods still require considerable effort in parameter tuning for handcrafted controllers. Instead, this work proposes a DNN-based end-to-end exoskeleton controller that eliminates the need for ambulation mode recognition, gait phase detection, and control parameter tuning.

TABLE I
COMPARISON WITH STATE-OF-THE-ART EXOSKELETON CONTROL METHODS

Methods	Publications	Gait Phase Detection	Handcrafted Controllers	Human Online Optimization
Gait-Cycle-based	[2]	CNN (mechanical sensor data)	Pseudo-impedance-based controller	Not Required
	[8]	CNN + Adaptive Oscillators (AOs)	Direct Torque Control	Not Required
	[9]	CNN + Adaptive Oscillators (AOs)	PID Controller	Not Required
	[19]	Two Force Plates	FSM-IC	Required, 5-25 minutes
	[17], [18]	Two Force Plates	Preference based Torque Profile	Required, 22-77 minutes
	[15], [16]	Two Force Plates	Metabolic-cost based Torque Profile	Required, 18-24 minutes
Non-Gait-Cycle-based	[20], [21]	Not Required	Output Feedback Controller	Not Required
	[22], [3]	Not Required	Scale-Delay-Filter	Not Required
	[1]	Not Required	PD Controller	Not Required
End-to-end Controller	**Ours**	**Not Required**	**Not Required**	**Not Required**

it to seamlessly adapt the assistance across different users, ambulation modes, and ambulation intensities. However, such adaptability still depends heavily on high-quality and large datasets, the manual derivation of reference assistive torque depends on substantial domain knowledge or prior experience. In contrast, deep reinforcement learning (DRL) has gained attention as the most promising way to achieve model-free exoskeleton control without handcrafted rules or parameter tuning in simulation [23], [24]. DRL-based methods aim to learn the assistive torque directly from interactions with the human body. However, the online implementation of the DRL method relying on extensive training has limited its applications [25], [26]. To address this challenge, recent advances in sim-to-real RL have successfully demonstrated a new way to mitigate the expensive human-involved training by learning in simulation using a musculoskeletal model [1]. Despite these developments, achieving zero-shot transfer of the end-to-end exoskeleton control policy to physical hardware—without additional manual tuning—remains an open and underexplored problem.

This work develops an end-to-end assistive torque controller using sim-to-real RL and curriculum learning, which directly maps the system feedback from onboard sensors to motor commands without any prior assumptions about the ambulation mode, the gait phase, and the handcrafted controllers (see Fig. 1). Compared to state-of-the-art methods summarized in Table I, our method enables the exploration of optimal assistive torques at each actuation step, effectively minimizing human effort. We propose learning a human control policy that enables walking at any continuous walking speed. On the other hand, a device-agnostic, zero-shot sim-to-real transfer strategy was employed to deploy the learned exoskeleton control policy from a musculoskeletal human model to physical robots. This approach alleviates potential discontinuities that commonly arise from reliance on high-level ambulation mode recognition and gait phase estimation. Our end-to-end control approach doesn't rely on gait phase estimation or manually tuning control parameters.

II. PROBLEM FORMULATION

A. Problem Statement

Providing smooth assistive torque during human-level ground walking with actively changing speeds is essential for enabling community mobility, yet it remains challenging due to limited adaptability. Our objective is to develop an end-to-end exoskeleton control learning approach based on a well-designed human musculoskeletal system (see Section II-B) and limited human reference motion datasets (see Section II-C). We focus on learning an exoskeleton control policy via sim-to-real RL to provide the desired assistance for human-level ground walking at varying speeds. The details of the implementation are illustrated in Fig. 2, which consists of two main phases: policy training in simulation and policy deployment on physical exoskeletons. In the *simulation phase*, learning an exoskeleton control policy relies on accurate modeling and control of the human musculoskeletal system at varying walking speed [4] via two training stages. At the first training stage, a human control policy was trained by integrating a goal-conditioned DRL [27] and curriculum learning scheme to reproduce the desired locomotion behaviors with continuously varying walking speed (see Section III-A). At the second training stage,

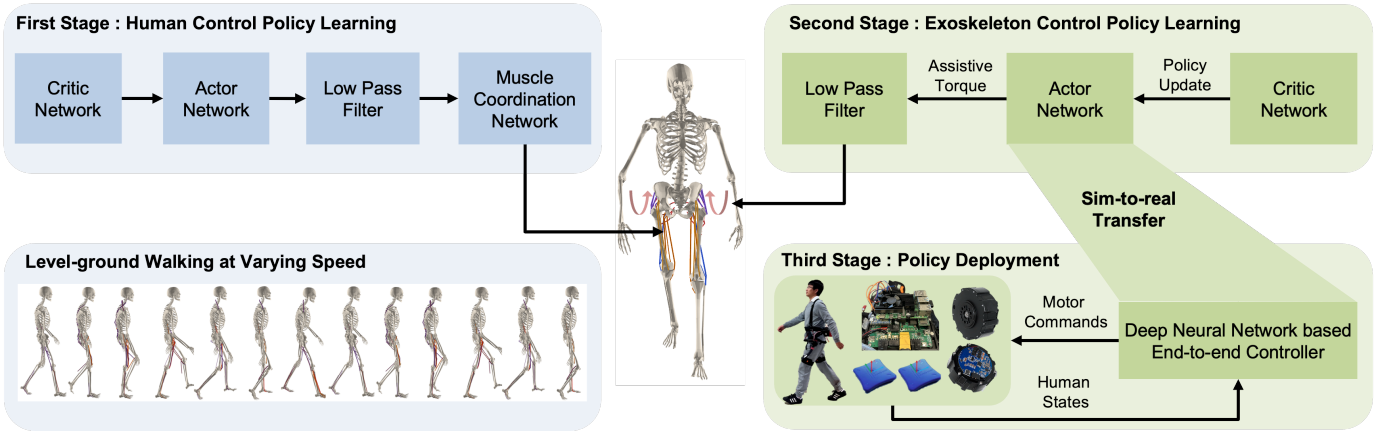


Fig. 2. Overview of the proposed method, which consists of two training stages in simulation and one policy deployment stage. Our end-to-end exoskeleton controller was developed using a human musculoskeletal model through sim-to-real reinforcement learning, employing a two-stage training approach. In the first training stage, a human control policy and a muscle coordination network are trained using PPO on a full-body human musculoskeletal model. This training is guided by a reference motion dataset and subjected to randomized external disturbances to improve robustness. In the second training stage, an exoskeleton control policy is learned based on biomechanical feedback to generate assistive torques for the target human joints, such as hip joints. Once the end-to-end exoskeleton control policy converges, the trained policy network is zero-shot deployed on the physical robot's electronic system as a neural network-based end-to-end controller. During execution, onboard sensors measure the human states, and our end-to-end controller generates motor commands, which are filtered by a low-pass filter before being sent to the motors for actuation.

an end-to-end exoskeleton control policy was trained to provide assistive force on target joints of the human musculoskeletal system by designing the reward to reduce muscle efforts (see Section III-B). Furthermore, the same curriculum learning scheme was designed to train the end-to-end exoskeleton control policy, which can facilitate smooth transitions between two gaits of varying walking speeds. In the *policy deployment phase*, the end-to-end exoskeleton control policy with learned parameters is zero-shot transferred to the customized electronic system of physical exoskeletons, providing the desired assistive torque (see Section III-C).

B. Human Musculoskeletal System Modeling

The dynamics of the human musculoskeletal system and exoskeleton are built upon DART [28], which is used for physics-based dynamic simulation. The full-body human musculoskeletal system consists of a rigid humanoid skeleton model coupled with computational muscle-tendon models.

1) Skeleton Modeling: The dynamics of the human musculoskeletal model with n_h joints are formulated as follows:

$$\boldsymbol{\mathcal{M}}^h(\boldsymbol{q}^h)\ddot{\boldsymbol{q}^h} + \boldsymbol{\mathcal{C}}^h(\boldsymbol{q}^h, \dot{\boldsymbol{q}^h})\dot{\boldsymbol{q}^h} = \boldsymbol{J}_m^T \boldsymbol{F}_m(\boldsymbol{a}) + \boldsymbol{J}_c^T \boldsymbol{F}_c + \boldsymbol{\tau}_{ext} \tag{1}$$

where $\boldsymbol{q}^h \in \mathbb{R}^{n_h}$ and $\dot{\boldsymbol{q}^h} \in \mathbb{R}^{n_h}$ are human joint angles and velocities, respectively. $\boldsymbol{\mathcal{M}}^h(\cdot)$ is the mass matrix, $\boldsymbol{\mathcal{C}}^h(\cdot,\cdot)$ is the Coriolis and gravitational forces. $\boldsymbol{F}_m \in \mathbb{R}^{n_h}$ and $\boldsymbol{F}_c \in \mathbb{R}^{n_h}$ are the muscle and constraint forces. $\boldsymbol{J}_m$ and $\boldsymbol{J}_c$ are Jacobian matrices, which map muscle forces and constraint forces to the human joint space. $\boldsymbol{a} \in \mathbb{R}^{n_m} \in [0,1]$ is muscle activation. n_m is the number of muscle units. $\boldsymbol{\tau}_{ext} \in \mathbb{R}^{n_h}$ is the external torque, including the force applied by both the exoskeleton and the environment.

2) Muscle Modeling: According to the Hill-type model [28], the $i-$th muscle-tendon unit is formulated to derive the muscle force $F_m(a_i; l_i, v_i)$ from activation as $F_{max} \cdot [a_i \cdot f_l(l_i) \cdot f_v(v_i) + F_p(l_i)]$. l_i and v_i represent the normalized muscle length and the rate of muscle changes, respectively. $F_p(l_i)$ is the passive force developed by the muscle. F_{max} is maximum isometric muscle force. Therefore, the muscle force can be rewritten as follows:

$$\boldsymbol{F}_m(\boldsymbol{a}) = \boldsymbol{F}_m^{max} \cdot [\boldsymbol{a} \cdot \boldsymbol{F}_l + \boldsymbol{F}_m^p] = \boldsymbol{A} \cdot \boldsymbol{a} + \boldsymbol{b} \tag{2}$$

where $\boldsymbol{F}_l$ is the function affected by muscle length and the rates of muscle changes.

C. Reference Human Dataset

The objective of human control policy is to allow the human musculoskeletal model to reproduce the desired motion, such as the level ground walking at the desired speed sampled from $[\boldsymbol{v}_{min}, \boldsymbol{v}_{max}]$ (see Fig. 2). The reference motion consists of the human joint angle, velocities, and human skeleton end-to-end position after re-targeting the collected MoCap data [29]. In contrast to existing deep learning-based methods that depend on large human datasets collected from human subjects, the reference human dataset for DRL-based human control policy training could be open-sourced. For instance, $\mathcal{D}_{\mathbf{ref}} = \{\{(\overline{\boldsymbol{q}}_t^h, \overline{\boldsymbol{q}}_t^h)\}_{t=0}^{H-1}; \boldsymbol{v}\}$ consists only of the reference motion at several discrete walking speeds $\{\boldsymbol{v}_0 = 0.75m/s, \boldsymbol{v}_1 = 1.25m/s, \boldsymbol{v}_2 = 1.75m/s\}$.

III. Device-Agnostic End-to-End Exoskeleton Control Policy Learning

Given the human musculoskeletal model and reference motion dataset, the sim-to-real device-agnostic exoskeleton control policy is learned through two stages (see Fig. 2). The human musculoskeletal system is actuated by a human control policy to replicate the given reference motion of level ground walking (Section III-A). Afterwards, the end-to-end exoskeleton control policy is learned by interacting with the human musculoskeletal model, and a curriculum learning method is proposed to learn a general policy by giving the target walking speed to the human control policy (Section III-B). During the simulation phase, the human control policy and

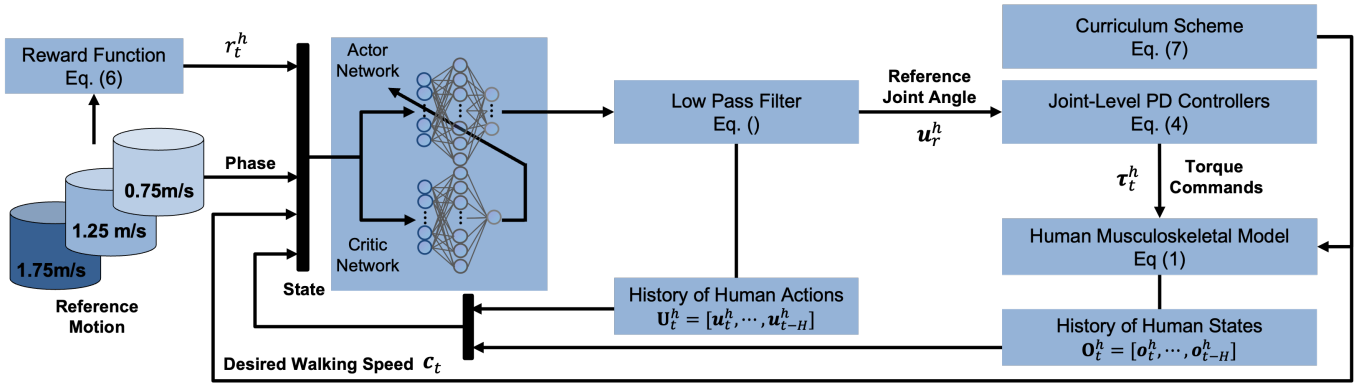


Fig. 3. Diagram of learning the human control policy that can enable the human musculoskeletal model level-ground walking at varying target speeds to train the exoskeleton control policy. The dataset consists of reference motions collected from human walking at three different speeds, which are used to design the adversarial imitation reward in (7). A curriculum learning scheme, as outlined in (8), is employed to generate the desired walking speed as the goal for training the human control policy, which outputs reference joint angles for each human joint. Muscle activations are then produced by combining the joint-level PD controller with the muscle coordination network. [bolei: To make the figure more visible, consider removing the background color. You can use a block outline for each part.]

Algorithm 1 Human Motion Imitation Policy Training

Input: I, N, L, T, M, γ_c
Initialize: $\boldsymbol{\phi}_{PD}$, $\boldsymbol{\phi}_h^m$, $\boldsymbol{\phi}_r^0$
for Each Iteration $i \in 1, 2, \cdots, I$ **do**
  $\mathcal{D}_i \leftarrow \emptyset$
  Sample desired walking speed $\boldsymbol{c}_i$ according to (8)
  for Each Actor $j \in 1, 2, \cdots N$ **do**
    Collect trajectories $\{\mathcal{T}_j^h\}$ from human control policy $\boldsymbol{\pi}_h(\cdot|\boldsymbol{c}_i; \boldsymbol{\phi}_h^m, \boldsymbol{\phi}_r^i, \boldsymbol{\phi}_{PD})$ for T timesteps
    Compute estimated advantages $\boldsymbol{\Omega}_j = \{\widehat{A}_t\}_{t=0}^{T-1}$
    $\mathcal{D}_i \leftarrow \mathcal{D}_i + \{\mathcal{T}_j^h, \boldsymbol{\Omega}_j\}$
  end for
  Update parameters $\boldsymbol{\phi}_r^i$ for L epochs using minibatch data with size M sampled from $\mathcal{D}_i$
  $\boldsymbol{\phi}_r^{i+1} \leftarrow \boldsymbol{\phi}_r^i$
end for
Output optimized parameters: $\boldsymbol{\phi}_r^\star \leftarrow \boldsymbol{\phi}_r^I$

the exoskeleton control policy need to be trained using continuous control DRL methods. By contrast, the action space of human control policy is high-dimensional, which can be learned using proximal policy optimization (PPO) for its stable property [30]. The exoskeleton control policy only outputs lower-dimension assistive torque for target lower-limb joints, which can be learned using the soft actor-critic (SAC) with better exploration [31]. Finally, the learned optimal end-to-end exoskeleton control policy will be zero-shot deployed on a physical exoskeleton (Section III-C).

A. Human Control Policy Training via Goal-conditioned Adversarial Imitation and Curriculum Learning

At the first training stage, a parameterized human control policy $\boldsymbol{\pi}_h(\boldsymbol{a}|\boldsymbol{s}^h, \boldsymbol{c}; \boldsymbol{\phi}_h)$ is learned to output muscle activation for replicating the reference motion given the task objective $\boldsymbol{c} \in \boldsymbol{\mathcal{C}}$ (see Fig. 2). $\boldsymbol{\mathcal{C}}$ is a predefined task space to represent walking speeds. Inspired by the nature muscle synergies controlled by central nervous system [28], [32]–[34], we decoupled the human control policy as two sub policies: human motion imitation policy $\boldsymbol{\pi}_h(\boldsymbol{u}^h|\boldsymbol{s}^h, \boldsymbol{c}; \boldsymbol{\phi}_h^s)$ and human muscle coordination policy $\boldsymbol{\pi}_m(\boldsymbol{a}|\boldsymbol{u}^h, \boldsymbol{s}_m^h; \boldsymbol{\phi}_h^m)$. The human muscle coordination policy is task-independent, allowing it to be learned via a regression-based method [1], [28]. Here, $\boldsymbol{u}^h$ is the actuation torque for desired motion imitation. Furthermore, in order to improve the sample efficiency for learning [27], [28], [35], the human motion imitation policy is decoupled into a reference motion prediction policy $\boldsymbol{\pi}_r(\boldsymbol{u}_r^h|\boldsymbol{s}^h; \boldsymbol{\phi}_r, \boldsymbol{c})$ and a joint-level proportional-derivative (PD) controller $\boldsymbol{\pi}_{PD}(\boldsymbol{u}^h|\boldsymbol{u}_r^h, \boldsymbol{s}^h; \boldsymbol{\phi}_{PD})$. Therefore, as detailed in see Fig. 3, learning a human motion imitation policy is degraded to learn a human reference motion prediction policy that outputs the reference joint angles $\boldsymbol{u}_r^h$ for the joint-level PD controller, which will generate the torque commands for the human musculoskeletal model.

We formulate the human motion imitation policy learning problem as a Markov Decision Process (MDP) $< \mathcal{S}_h, \mathcal{U}_h, \mathcal{R}_h, \mathcal{T}_h^h, \mathcal{P}_h^0, \gamma_h >$. $\boldsymbol{s}^h \in \mathcal{S}_h$ is the human state of the musculoskeletal model. $\boldsymbol{u}^h \in \mathcal{U}_h$ is the action taken by the human motion imitation policy. $\mathcal{R}_h : \mathcal{S}_h \times \mathcal{U}_h \rightarrow r^h(\boldsymbol{s}^h, \boldsymbol{u}^h) \in \mathbb{R}$ is the human reward function. $\mathcal{P}_h : \mathcal{S}_h \times \mathcal{U}_h \rightarrow \mathcal{S}_h$ is the state transition function. p_h^0 is an initial distribution of human state and $\gamma_h \in (0, 1)$ is the discount factor. The parameter of the human motion imitation policy will be updated by maximizing the expected cumulative reward, as follows:

$$\boldsymbol{\mathcal{J}}(\boldsymbol{\phi}_r) = \mathbb{E}_{p(\boldsymbol{c})} \mathbb{E}_{\mathcal{T}^h \sim p(\mathcal{T}^h|\boldsymbol{\phi}_r, \boldsymbol{c})} \left[\sum_{t=0}^{T-1} \gamma_h^t r_t(\boldsymbol{s}_t^h, \boldsymbol{u}_t^h, \boldsymbol{c}) \right] \tag{3}$$

where $\mathcal{T}^h = \{(\boldsymbol{s}_t^h, \boldsymbol{u}_t^h, r_t)_{t=0}^{T-1}, \boldsymbol{s}_T\}$ is the sampled human motion trajectory during one episode with the time horizon T. $p(\mathcal{T}^h|\boldsymbol{\phi}_r, \boldsymbol{c})$ is the distribution over all possible trajectories. For each interaction episode, $\boldsymbol{s}_0^h \sim \mathcal{P}_h^0$ is sampled and action $\boldsymbol{u}_t^h \sim \boldsymbol{\pi}_r(\boldsymbol{u}_t^h|\boldsymbol{s}_t^h; \boldsymbol{\phi}_r)$ is sampled at each control timestep $\boldsymbol{t}$. The joint-level PD controller $\boldsymbol{\pi}_{PD}(\boldsymbol{u}^h|\boldsymbol{u}_r^h, \boldsymbol{s}^h; \boldsymbol{\phi}_{PD})$ is defined to generate actuation torque for each joint as:

$$\boldsymbol{\tau}^h = \boldsymbol{K}_p^h \cdot (\boldsymbol{u}_r^h - \boldsymbol{q}_t^h) + \boldsymbol{K}_d^h \cdot \dot{\boldsymbol{q}}_t^h \tag{4}$$

where $\phi_{PD} = [\boldsymbol{K}_p^h, \boldsymbol{K}_d^h]$ is the predefined parameters of joint-level PD controller. Afterwards, the muscle activation

Algorithm 2 End-to-end Exoskeleton Control Policy Training

Input: ϕ_h^m, $\phi_r^\star$, ϕ_{PD}, β_c
Initialize: ψ_e^0, $\mathcal{D}_e \leftarrow \varnothing$
for Each Iteration $i \in 0, 1, 2, \cdots, I$ **do**
  Sample desired walking speed $\boldsymbol{c}$
  for Each Actuation Step $\boldsymbol{t}$ **do**
    Sample action from policy: $\boldsymbol{u}_t^e \sim \boldsymbol{\pi}_e(\boldsymbol{u}_t^e|\boldsymbol{o}_t^e; \boldsymbol{\psi}_e^i)$
    Sample transition from $\boldsymbol{o}_{t+1}^e \sim \mathcal{P}_e(\boldsymbol{o}_{t+1}^e|\boldsymbol{o}_t^e, \boldsymbol{u}_t^e)$
    $\mathcal{D}_e \leftarrow \mathcal{D}_e + \{(\boldsymbol{o}_t^e, \boldsymbol{u}_t^e, r_t^e, \boldsymbol{o}_{t+1}^e)\}$
  end for
  for Each Gradient Step **do**
    Update parameter $\boldsymbol{\psi}_e^i$ using samples in $\mathcal{D}_e$
  end for
end for
Output optimized parameters: $\boldsymbol{\psi}_e^\star \leftarrow \boldsymbol{\psi}_e^I$

$\boldsymbol{a}$ is derived from $\boldsymbol{\pi}_m(\boldsymbol{a}|\boldsymbol{\tau}^h, \boldsymbol{s}^h; \boldsymbol{\phi}_m)$ to drive the human musculoskeletal model. The next state is sampled by $\boldsymbol{s}_{t+1}^h \sim \mathcal{P}_h(\boldsymbol{u}_t^h|\boldsymbol{s}_t^h; \boldsymbol{\phi}_h)$ and the reward r_h^t can be accordingly calculated. Additionally, a low pass filter was employed to smooth the reference joint angle, as follows:

$$H(\boldsymbol{q}_t^h) = \frac{1}{\sqrt{1 + (\boldsymbol{q}_t^h/\boldsymbol{\omega}_c)^{2n_c}}} \tag{5}$$

where $\boldsymbol{\omega}_c$ is the cutoff frequency and n_c is the order of the filter. Afterwards, the reference joint angle is clipped to meet the requirements of the physical robot.

Similar to the formulation of learning a goal-conditioned RL policies [30], [36]–[38], the state space for human motion prediction policy learning is defined as follows

$$\boldsymbol{s}_t^h = [\mathbf{O}_t^h, \mathbf{U}_t^h, \varsigma, \boldsymbol{c}_t] \tag{6}$$

where $\mathbf{O}_t^h = [\boldsymbol{o}_t^h, \cdots, \boldsymbol{o}_{t-T}^h]$ represents the history of the past human states. $\mathbf{U}_t^h = [\boldsymbol{u}_t^h, \cdots, \boldsymbol{u}_{t-T}^h]$ represents the history of the reference motions. A phase variable $\varsigma \in [0, 1]$ is employed to align with the reference motion. $\boldsymbol{c}_t$ is the task objective, the desired walking speed.

The goal-conditioned reward r_t^h at each actuation step $\boldsymbol{t}$ consists of four terms to encourage the musculoskeletal model to reproduce human natural motion and fulfill the desired task requirements:

$$\begin{aligned} r_t^h &= w^I r_t^I + w^G r_t^G + w^C r_t^C \\ r_t^I &= w^p r_t^p + w^v r_t^v + w^e r_t^e \\ r_t^G &= \exp(-\sigma_G \frac{||\dot{\mathbf{x}}_t^{com} - \mathbf{v}_t^e||^2}{|\dot{\mathbf{x}}_t^{com}|}) \\ r_t^C &= w^m r_t^m + w^s r_t^s \end{aligned} \tag{7}$$

where r_t^I is used to encourage the policy to produce the reference motion in datasets. r_t^G is used to encourage one to fulfill the task-specific objective, here it is to track the desired walking speed. $\dot{\mathbf{x}}_t^{com}$ is the actual center-of-mass velocity of human musculoskeletal model. r_t^C is used to fulfill external requirements for the policy, such as energy reduction r_t^m and smooth motion r_t^s. $\{w^I, w^G, w^C, w^p, w^v, w^m, w^s, w^e\}$ are the weights of each sub-reward, which should be pretested for each task.

Curriculum learning is employed to facilitate the training of the human control policy across different walking speeds [27], [39]. For level-ground walking in particular, we adopt the following update rule to monotonically increase the walking speed:

$$\boldsymbol{c}_k = \boldsymbol{v}_{min} + (\boldsymbol{v}_{max} - \boldsymbol{v}_{min}) \cdot (1 - \beta_c^i) \tag{8}$$

where i is the iteration index. $0 < \beta_c < 1$ is the curriculum factor.

Given the pretrained muscle coordination network with ϕ_h^m and joint-level PD controller $\boldsymbol{\phi}_{PD}$, the human motion imitation policy $\boldsymbol{\pi}(\boldsymbol{u}_r^h|\boldsymbol{s}^h; \boldsymbol{\psi}_r)$ is trained using PPO [40] chosen for its efficiency and suitability for high-dimension control problem. As illustrated in Fig. 3, multilayer perception (MLP)-based encoder is employed for both actor and critic networks. The implementation details of training a human motion imitation policy using N parallel actors are concluded in Algorithm 1.

B. Device-agnostic End-to-end Exoskeleton Control Policy Learning via Maximum Entropy RL

On the second training stage, given the learned human control policy at the desired walking speed $\boldsymbol{c}$ sampled from the same curriculum scheme, the objective of learning the end-to-end exoskeleton control policy by interacting with the human musculoskeletal model (see Fig. 2). A parameterized end-to-end exoskeleton control policy $\boldsymbol{\pi}_e(\boldsymbol{u}^e|\boldsymbol{s}^e, \boldsymbol{c}; \boldsymbol{\psi}_e)$ is formulated and learned from interaction data by ideally applying assistive force on the desired human joints (see Fig. 4). $\boldsymbol{u}^e \in \mathcal{U}_e$ is the motor commands for the exoskeleton, which is ideally transformed into the actuation torque applied to the desired human joints. $\boldsymbol{s}^e \in \mathcal{S}_e$ is the full state of the exoskeleton agent, which should include both human states and robot states. However, the end-to-end exoskeleton control policy needs to be deployed on physical exoskeletons. Therefore, at each actuation step t, only partial observation of the human state $\boldsymbol{o}_t^e \in \mathcal{O}_e$ that is accessible and measured from the onboard sensors can be used for policy $\boldsymbol{\pi}_e(\boldsymbol{u}_t^e|\boldsymbol{o}_t^e; \boldsymbol{\psi}_e)$ learning. The exoskeleton control policy learning problem can be formulated as a POMDP $< \mathcal{S}_e, \mathcal{U}_e, \mathcal{P}_e, \mathcal{R}_e, \mathcal{O}_e, \mathcal{P}_e^{\mathcal{O}}, \gamma_e >$. The object is to maximize the average expected cumulative reward given as follows:

$$\boldsymbol{\mathcal{J}}(\boldsymbol{\psi}_e) = \mathbb{E}_{\mathcal{T}^e \sim p_{\boldsymbol{\psi}_e}(\mathcal{T}^e)} \left[\sum_{t=0}^{\infty} \gamma_e^t r_t^e(\boldsymbol{o}_t^e, \boldsymbol{u}_t^e) \right] \tag{9}$$

where $r_t^e(\boldsymbol{o}_t^e, \boldsymbol{u}_t^e)$ represents the received reward at the actuation step $\boldsymbol{t}$. $\mathcal{T}^e = \{(\boldsymbol{o}_0^e, \boldsymbol{u}_0^e, r_0^e), \cdots, (\boldsymbol{o}_T^e, \boldsymbol{u}_T^e, r_T^e)\}$ is the sampled one episode trajectory including all assistive torque, observations, and reward feedback.

The observation for training the end-to-end exoskeleton control is defined as follows:

$$\boldsymbol{o}_t^e = [\boldsymbol{q}_t^e, \dot{\boldsymbol{q}}_t^e] \tag{10}$$

where $\boldsymbol{q}_t^e$ and $\dot{\boldsymbol{q}}_t^e$ are the measured angle and velocity of both hip joints from onboard sensors at each actuation step $\boldsymbol{t}$. Therefore, in order to deal with the partial observable problem [41], the LSTM-based encoder was employed for both critic and actor networks by preserving the historic information (see Fig. 4).

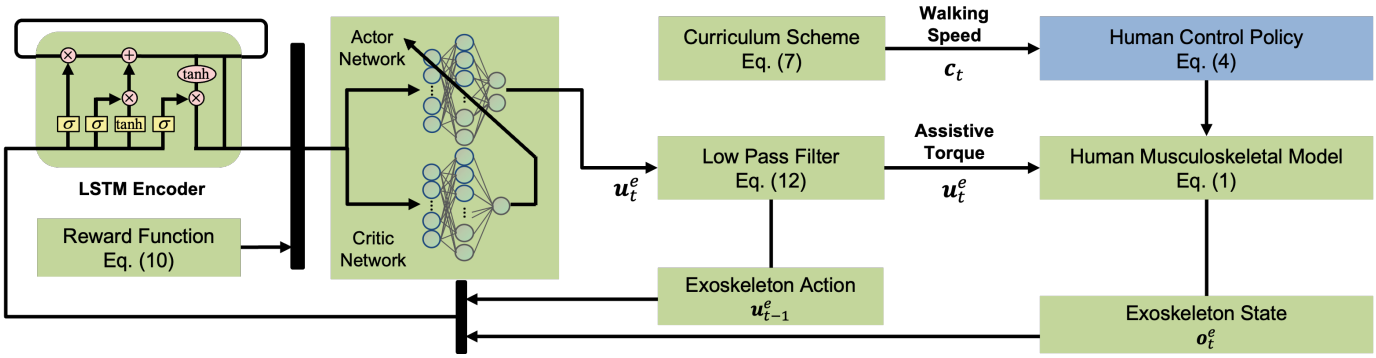


Fig. 4. Diagram illustrating the learning of an end-to-end exoskeleton control policy using the same curriculum scheme as in (8) and the learned human control policy. The reward function defined in (11) is designed to train the end-to-end exoskeleton control policy that can reduce human muscle effort. The human musculoskeletal model is driven by the learned human control policy to achieve level-ground walking at varying speeds. The assistive torque generated by the LSTM-based actor network is subsequently filtered and scaled before being directly applied to the target human joints. [bolei: To make the figure more visible, consider removing the background color. You can use a block outline for each part.]

The reward for training an end-to-end exoskeleton control policy consists of several terms as follows:

$$r_t^e(\boldsymbol{o}_t^e, \boldsymbol{u}_t^e) = w^h r_t^h + w^m r_t^m + w^c r_t^c \tag{11}$$

where r_t^h is the first sub-reward that is used to encourage the human performance walking under the given desired speed. r_t^m is the second sub-reward to encourage the reduction of muscle work. r_t^c is the third sub-reward to encourage the smoothness of the generated assistive torque. $\{w^h, w^m, w^c\}$ are the weights of each sub-reward, which should be pretested. Furthermore, only the muscle activation $\boldsymbol{m}_l \in \mathbb{R}^{30}$ of potentially assisted muscles was employed to evaluate the end-to-end exoskeleton control policy, as follows:

$$r_t^m = \exp\left(-\sigma_m ||\boldsymbol{m}_l||^2\right) \tag{12}$$

where σ_m is a sensitive factor.

The implementation of training end-to-end exoskeleton control policy based on SAC [31] is concluded in Algorithm 2. Furthermore, the termination criteria of the training reference prediction policy depend on those of the training human control policy. Additionally, a low-pass filter was employed to smooth assistive torque, as follows:

$$H(\boldsymbol{\tau}_c) = \frac{1}{\sqrt{1 + (\boldsymbol{\tau}_c/\boldsymbol{\omega}_c)^{2n_c}}} \tag{13}$$

where $\boldsymbol{\omega}_c$ is the cutoff frequency and n_c is the order of the filter. Afterwards, the actuation torque is clipped to meet the requirements of a physical robot.

C. Zero-shot Development of Learned End-to-end Exoskeleton Control Policy

When the end-to-end exoskeleton control policy has achieved the desired termination criteria, the device-agonistic policy $\boldsymbol{\pi}(\boldsymbol{u}^e|\boldsymbol{o}^e; \boldsymbol{\psi}_e)$ can be adapted to the physical exoskeleton for the same target joints (see Fig. 2). The LSTM-based actor network with optimized parameters $\boldsymbol{\psi}_e^*$ can be zero-shot transferred and run on the customized electronic system (see Fig. 2). Without loss of generality, the hip exoskeleton driven by QDDs is taken as an example. The input to the LSTM-based actor network is the observation of human state $\boldsymbol{o}_t^e$, including the human hip joint angle $\boldsymbol{q}_t^e$ and velocity $\dot{\boldsymbol{q}}_t^e$, that are measured by IMUs.

TABLE II
HYPER-PARAMETERS OF PPO

Parameter	Value
Number of Actors	$N = 16$
Number of Iterations	$I = 5000$
Number of Epochs	$L = 10$
Horizon of Each Episode	$T = 2048$
Minibatch Size	$B = 128$
Discount	$\gamma = 0.99$
Clip rate	$\epsilon = 0.2$
Learning Rate	$\alpha_a = 0.0001, \alpha_c = 0.001$
Simulation frequency	600Hz
Control frequency	30Hz
Reward weights	$w^I = 0.5$, $w^G = 0.5$, $w^C = 0.05$

TABLE III
HYPER-PARAMETERS FOR OPTIMIZING SAC

Parameter	Value
Number of Episodes	$N = 200000$
Learning Rate	$\alpha_a = 0.0001, \alpha_c = 0.001$
Minibatch Size	$B = 128$
Discount factor	$\gamma_e = 0.99$
Relay buffer size	10^6
Control frequency	30Hz
Reward weights	w^h =0.1, w^m=0.5, w^c=0.8

The sensing frequency f_s and control frequency f_p of the customized electronic system are set to the same values as in the simulation phase.

IV. Simulation Validation

The simulation was conducted to learn and evaluate the human control policy and exoskeleton control policy. First, the learned human control policy was validated by walking on the level ground with actively changing walking speed. Second, the learned end-to-end exoskeleton control was evaluated by comparing it with three state-of-the-art assistive controllers.

A. Protocol of Policies Learning in Simulation

The simulation of a human musculoskeletal model and a hip exoskeleton were built based on DART at the frequency of 600Hz. The human control policy and exoskeleton control policy were trained at a control frequency of 30Hz on a PC with one GPU (GEFORCE RTX 4080).

1) Protocol of Training Human Control Policy in Simulation: PPO was used to train the human control policy using the public reference motion dataset and the parameters summarized in Table II. Furthermore, two ablation studies were conducted to compare the training performance. First, the proposed curriculum scheme in (8) enables the human control policy to track a changing desired walking speed for each episode. Without using the curriculum scheme, the human control policy only needs to follow a fixed walking speed (1.2 m/s), named as Ours-No-CL. Second, using the curriculum scheme but instead of using the adversarial intimation reward, the imitation reward depending on exponential form [1] was utilized to train the human control policy, named as Ours-Exponential Reward.

2) Protocol of Training End-to-end Exoskeleton Control Policy in Simulation: The optimal parameters for the human control policy trained using adversarial imitation reward and the muscle coordination network were loaded to drive the human musculoskeletal model under the predefined curriculum scheme. Afterwards, the end-to-end exoskeleton control was trained using SAC following the implementation in Table 2. The details of LSTM encoder for actor and critic networks of SAC are summarized in Table III. Additionally, for the ablation study, the exoskeleton control policy was trained by interacting with two human control policies learned in Section IV-A1, separately.

B. Results of Policies Evaluation in Simulation

The learned human control policy was evaluated across a range of walking speeds from $0.6m/s$ to $1.8m/s$. Biological torque under three walking speeds (0.6m/s, 1.2m/s, and 1.8m/s) was collected from the open-source dataset [22]. First, kinematic feedback, kinetic feedback, and muscle activation data were collected and compared against the reference dataset. Secondly, the assistive torque and positive power for each gait cycle were calculated to assess the timing and effectiveness of assistance, and the results were compared. Thirdly, the timing of assistance and the proportion of positive power [20] are calculated as two metrics for comparing to three state-of-the-art methods as follows

$$\begin{aligned}\textbf{Timing of Assistance} &= \sum \frac{P_{assistive} - P_{velocity}}{P_{velocity}} \\ \textbf{Proportion of Positive Power} &= \frac{\sum_{i=1}^{n_{gait}} \mathcal{I}(\boldsymbol{\tau}_i \cdot \dot{\boldsymbol{q}}_i^h)}{n_{gait}}\end{aligned} \tag{14}$$

where $P_{assistive}$ and $P_{velocity}$ are the point index of the assistive torque and hip joint velocity during one gait cycle. n_{gait} is the number of samples during one gait cycle. $\mathcal{I}(\boldsymbol{\tau}_i \cdot \dot{\boldsymbol{q}}_i^h)$ equals to 1 if $\boldsymbol{\tau}_i \cdot \dot{\boldsymbol{q}}_i^h > 0$, else equals to 0. An AOs-based controller [8] was implemented to generate an assistive torque corresponding to the detected gait phase. Samsung controller [20] was implemented using manually tuned parameter $k_{sam} = 10$ and $\Delta t = 0.03$s. TCN-based controller [22] was implemented using the pre-trained parameters.

1) Results of Human Control Policy Evaluation: The episode rewards of training a human control policy with/without using curriculum learning were shown in Fig. 14(a). The episode rewards of training a human control policy using or without using adversarial imitation reward were plotted in Fig. 14(b). The trained human control policy is evaluated on treadmill walking given three commonly used walking speeds ($0.6m/s$, $1.2m/s$, and $1.8m/s$). The angles, velocities, and biological torque of two hip joints at three walking speeds are plotted in Fig. 6. We also compared the calculated mean values with the reference values collected from a public dataset [42]. The kinematics and kinetic results in Fig. 6 indicate that the human musculoskeletal model driven by the learned human control policy can walk at three walking speeds.

TABLE IV
BENCHMARK RESULTS

Methods	Metrics	
	TA	PP
Adaptive Oscillators-based Controller [8]	×	
Samsung Controller [20]	0.35±0.03	
TCN-based Controller [22]	×	
Ours	0.64±0.10	

2) Results of End-to-end Exoskeleton Control Policy Evaluation: The episode rewards of training end-to-end exoskeleton control policy with/without using curriculum learning was shown in Fig. 5(c). The episode rewards of training end-to-end exoskeleton control policy using or without using adversarial imitation reward were plotted in Fig. 5(d). The results without wearing hip exoskeleton can be considered as the baseline, named as *No-Exo* mode. The muscle activation of five muscle groups over the recent five gait cycles can be collected. The human musculoskeletal model wearing a hip exoskeleton by providing zero assistive force is named as *Zero-Torque* mode. The results of using learned end-to-end exoskeleton control policy is named as *Assistive mode*. The comparison of muscle activation between *No-Exo* mode and *Assistive* mode of learned end-to-end exoskeleton controller across three walking speeds are plotted in Fig. 7. The colormap to visualize the muscle activation of the human imitation agent assisted by the exoskeleton control policy is plotted. All four assistive torque profiles and biological torque under three walking speeds (0.6m/s, 1.2m/s, and 1.8m/s) of a gait cycle are plotted in Fig. 8. Additionally, four assistive torque profiles of the varying walking speed are plotted in Fig. 9. We can see that the Samsung controller with the same parameters cannot adapt to a changed walking speed. Afterwards, the assistive power is calculated for each method, and two metrics are accordingly calculated. The comparison of the calculated metrics was given in the benchmark Table IV. The highest proportion of positive power indicates that our learned exoskeleton control policy can provide more effective assistance. Furthermore, our method enables the higher timing of assistance.

V. EXPERIMENTAL VALIDATION

The experiments were conducted to evaluate the effectiveness of zero-shot deployment of the learned end-to-end exoskeleton policy on a hip exoskeleton with a customized electronic system during both treadmill walking and outdoor walking. The performance was further validated by comparing it against three state-of-the-art assistive controllers using experiments involving ten subjects (five young adults and five elderly participants).

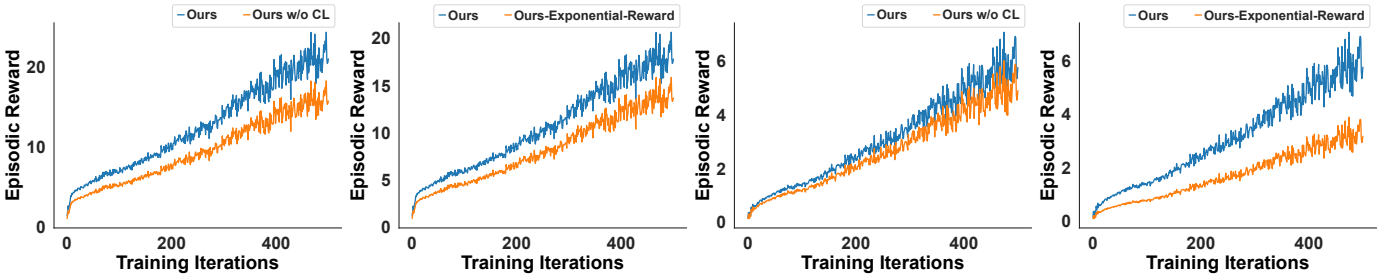


Fig. 5. Episodic reward in simulation indicate that both the human control policy and exoskeleton control policy can be learned after 5000 iterations learning. (a) Episode rewards of the human control policy with and without the curriculum scheme. (b) Episode rewards of the human control policy with and without adversarial imitation. (c) Episode rewards of the exoskeleton control policy with and without the curriculum scheme. (d) Episode rewards of the exoskeleton control policy with and without using adversarial imitation reward.

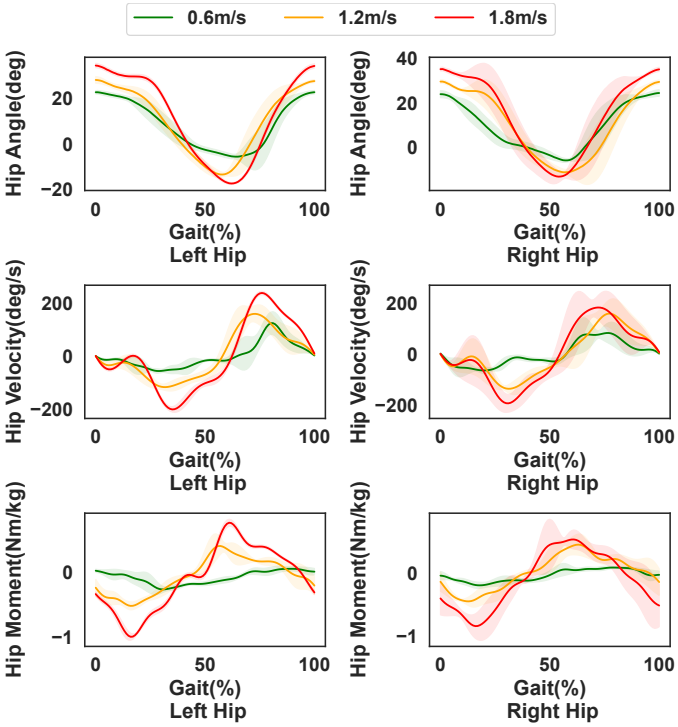


Fig. 6. Simulation results showing the mean and variance of hip joint angle, velocity, and biological torque over five gait cycles during the evaluation of the learned human control policy. The results indicate that the learned human control policy successfully replicates both kinematic and kinetic patterns across all three walking speeds.

A. Setups of Customized Electronic System for Exoskeleton Control Policy Deployment

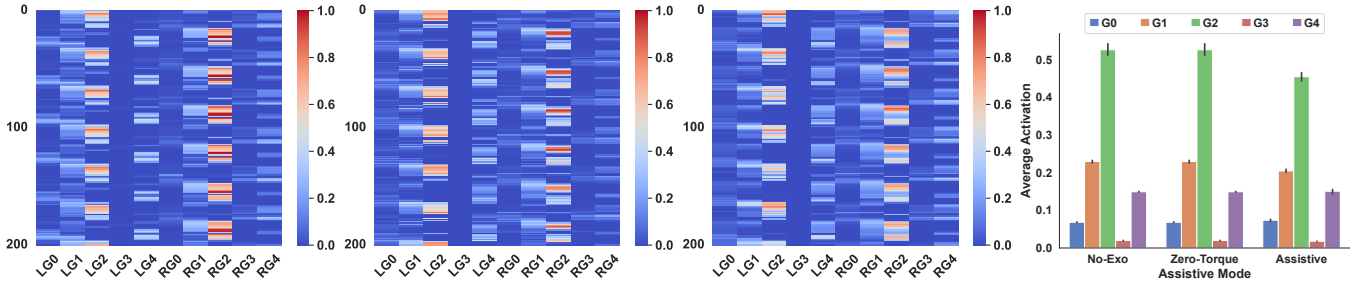
The developed hip exoskeleton and its customized electronic system for treadmill walking and outdoors walking are illustrated in Fig. 10(a). The protocol of deploying the trained end-to-end exoskeleton control policy is depicted in Fig. 10(b). Human hip joint angles and angular velocities are measured using two IMUs mounted on the thighs (LPMS-B2, ALUBI, China). Additionally, two other IMUs are attached to the ankle to detect the gait cycle for evaluation purposes. Human observation data are collected using the serial port of a microcontroller, Teensy 4.1, at a frequency of 300Hz. The data are processed via a Butterworth digital filter with a cutoff frequency of 5Hz, which is sent to the Raspberry Pi 4B at 100Hz as the inputs to the policy network. After loading the learned parameters in simulation, the LSTM-based actor network was developed on Raspberry Pi 4B to derive the motor commands, which are sent to QDD actuators at 100Hz. A user guidance interface was developed using PyQt5 to visualize the actual interaction data via Bluetooth at 30Hz.

B. Experimental Protocol for Exoskeleton Control Policy Development

Five able-bodied young adults and five elderly adults will wear the hip exoskeleton introduced in Section V-A to validate the effectiveness of the proposed controller using the following three evaluation metrics. First, the assistive force profiles and assistive power were calculated to assess the performance for both treadmill walking and outdoor walking. Second, the timing of assistance and proportion of the positive power are calculated across multiple gait cycles. Thirdly, the metabolic cost reduction was measured to assess the energy saving performance following the detailed experimental protocol described in Appendix VIII-C1. Finally, after completing both treadmill walking and outdoor walking, four questionnaires are administered to collect subjective user evaluations: SUS Usability [43], MDMT [44], NASA Task Load Index [45], and customized experience questionnaires (described in Appendix VIII-C2).

1) Experimental Protocol of Deploying Exoskeleton Control Policy during Treadmill Walking: Fig. 10(a) illustrates a representative subject wearing the hip exoskeleton for treadmill walking. All subjects are allowed to complete level ground treadmill walking at four different speeds: $0.6m/s$, $1.2m/s$, $1.8m/s$, and varying walking speed patterns. Each participant walks 8 minutes for each speed, four controllers in Table IV are implemented in a random order. Each controller will provide the assistive force for a period of two minutes. After completing each walking speed, each participant will have a 5-minute break.

2) Experimental Protocol of Deploying Exoskeleton Control Policy during Outdoor Walking: All participants are instructed to walk outdoors following the predefined route at actively changing walking speed using all four methods concluded in Table IV. For the elderly adults, the completion time will be recorded to evaluate the overall assistance performance of each controller.



(a) (b) (c) (d)

Fig. 7. Simulation results showing muscle activation performance over five gait cycles using the learned human control policy. LG0, LG1, LG2, LG3, and LG4 represent the five muscle groups of the left limb, while RG0, RG1, RG2, RG3, and RG4 represent the five muscle groups of the right limb. (a) Heatmap of the muscle activation under *No-Exo* mode. (b) Heatmap of the muscle activation under *Zero-Torque* mode. (c) Heatmap of the muscle activation under *Assistive* mode. (d) Comparison of average muscle activation over five gait cycles under three assistive modes.

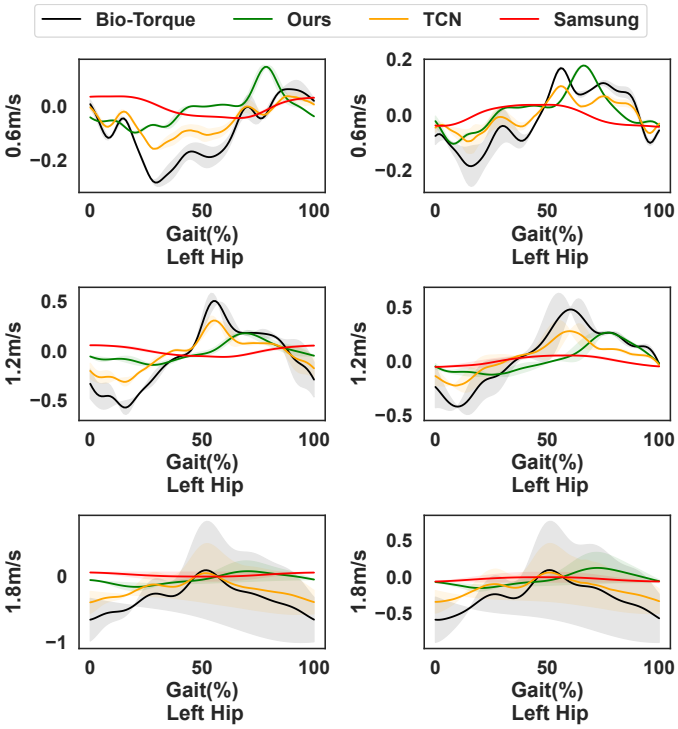


Fig. 8. Simulation results showing four assistive torque profiles and the corresponding biological torque within one gait cycle at three walking speeds: 0.6 m/s, 1.2 m/s, and 1.8 m/s.

C. Results of Developing Exoskeleton Control Policy

1) Results of Biological Torque and Assistive Torque Walking at Four Different Speeds: The actual biological torque and corresponding assistive torque of three fixed walking speeds at $0.6m/s$, $1.2m/s$, and $1.8m/s$ were plotted in Fig. 11. Additionally, the biological torque of the varying walking speed on the treadmill is predicted using the TCN proposed in [22]. The biological torque and assistive torque of all four methods for outdoor walking are plotted in Fig. 12.

2) Results of Positive Power and Timing of Assistance of All Participants: The proportion of assistive power and timing of assistance for able-bodied young subjects and elderly adults are calculated for the recent five gait cycles, separately. The mean and variance of each walking speed across all participants are calculated and plotted in Fig. 13.

3) Results of Metabolic Cost Reduction and Completion Time of All Elderly Participants Walking Outdoors: All five elderly participants are instructed to walk outdoors following a fixed route at an actively selected walking speed. The metabolic cost reduction and completion time of all four methods were calculated and plotted in Fig. 14.

VI. Discussion and Limitation

The success of learning an end-to-end exoskeleton control policy relies on three key components: human control policy learning, exoskeleton control policy learning, and zero-shot policy deployment.

A. Human Control Policy Learning

Different from the existing deep learning based methods [22], our method utilizes fewer open-source human reference datasets to train a human control policy that can reproduce the desired behavior with continuous walking speed. Particularly, the proposed learning method enables the human control policy to drive the human musculoskeletal model, reproducing the behavior with varying walking speeds. Unlike previous works [1], [46], this study demonstrates that the human musculoskeletal model, driven by a human control policy, can adapt to external assistance, which is reused to train and evaluate exoskeleton control policies. Especially, the adversarial imitation reward and curriculum scheme enable the human control policy to reproduce the behavior without reference motion. The kinematic feedback, kinetic feedback, and muscle activation have been collected to evaluate the performance of the learned human control policy by comparing it to the reference dataset.

This study focuses on enhancing generalization across different walking speeds, rather than achieving higher human walking performance or improving sample efficiency. Future work may explore reward weight tuning and improve network architecture to enhance the policy performance [47].

B. Exoskeleton Control Policy Learning

The end-to-end exoskeleton control policy, which directly outputs motor commands, was learned using SAC with maximum entropy for improved exploration [31]. The reward was designed to reduce the muscle efforts given a well-trained human control policy and a curriculum scheme. Both simulation and experiment results indicate that the sim-to-real framework can learn efficient and effective assistive torque in terms of

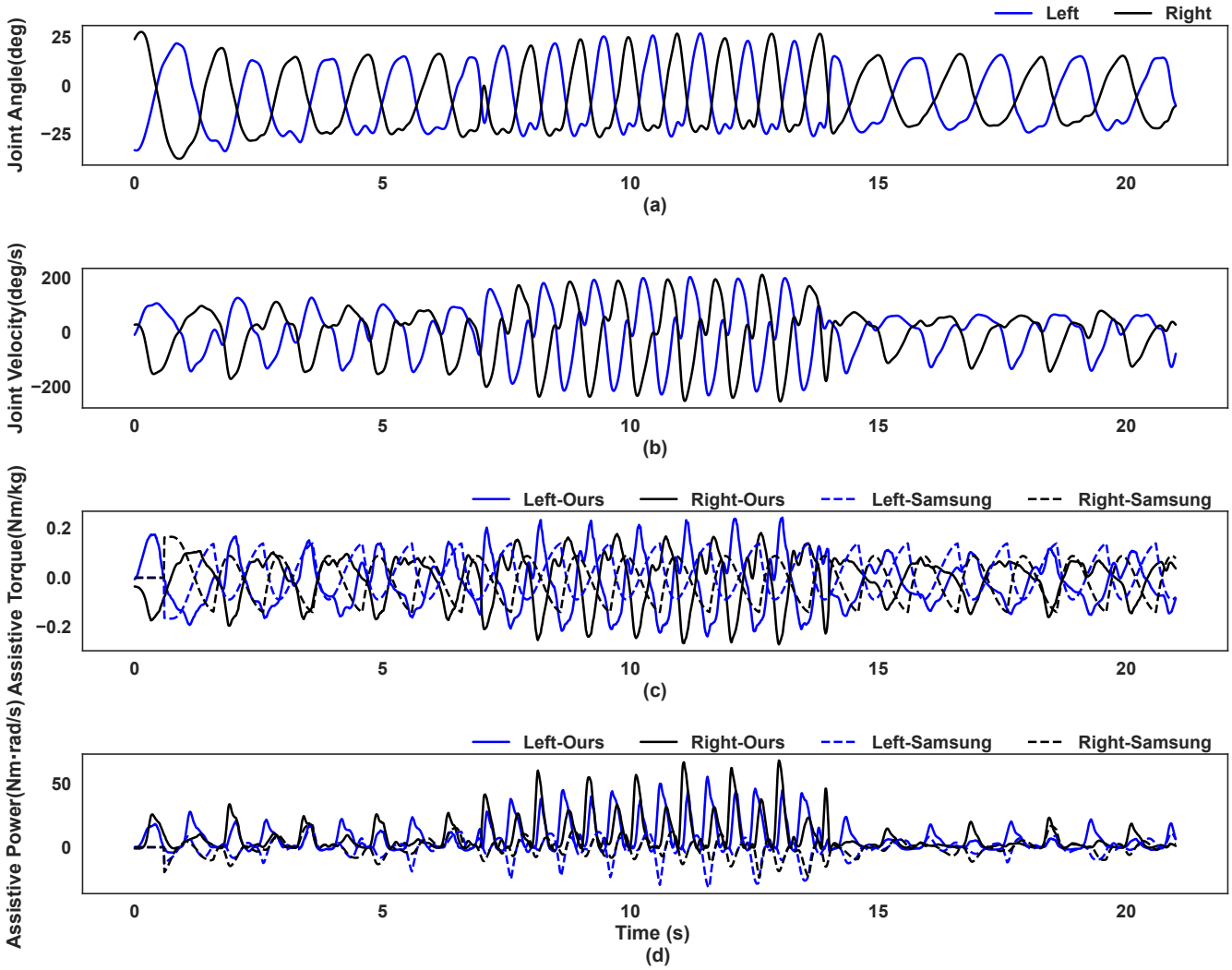
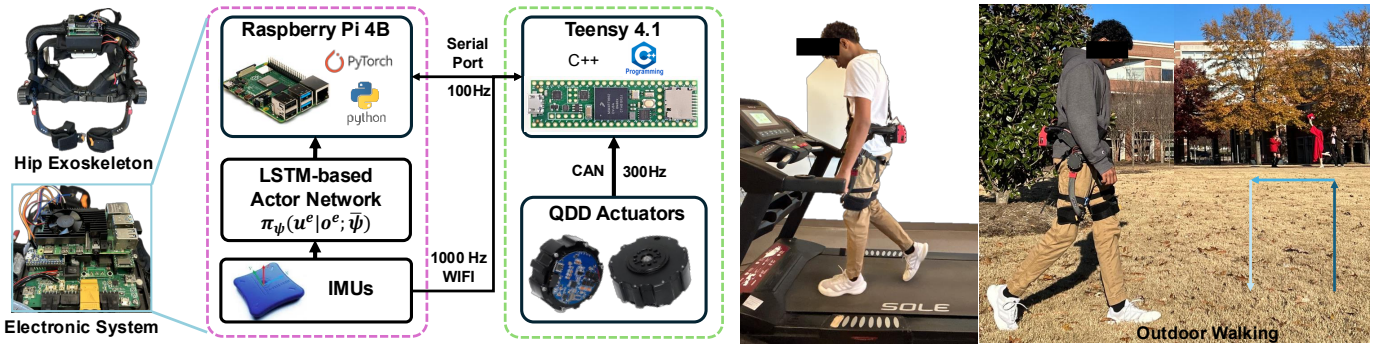


Fig. 9. Simulation results showing biological torque and four assistive torque profiles for treadmill walking with varying walking speeds.



(a) (b)

Fig. 10. Experimental setups for policy deployment. (a) Customized electronic system for policy deployment on quasi-direct driven hip exoskeletons. The forward propagation of trained LSTM-based actor network with optimized parameters is run on Raspberry Pi 4B to generate the motor commands. The lower-level controller is run on Teensy 4.1 to collect the human state from IMUs and send the motor commands to actuators via CAN. (b) Experiments demonstrating policy deployment on hip exoskeletons for treadmill and outdoor walking.

the timing of assistance and positive power. Furthermore, the learned exoskeleton control policy was validated by outdoor walking at varying speeds.

This article aims to validate the feasibility of such a learning framework for varying walking speeds. Therefore, the proposed method was only validated by the level-ground walking at varying changing speed, which can definitely be employed for other locomotion modes. Additionally, the reward is designed for training an end-to-end exoskeleton control policy that can reduce the muscle efforts. The proposed method can also be

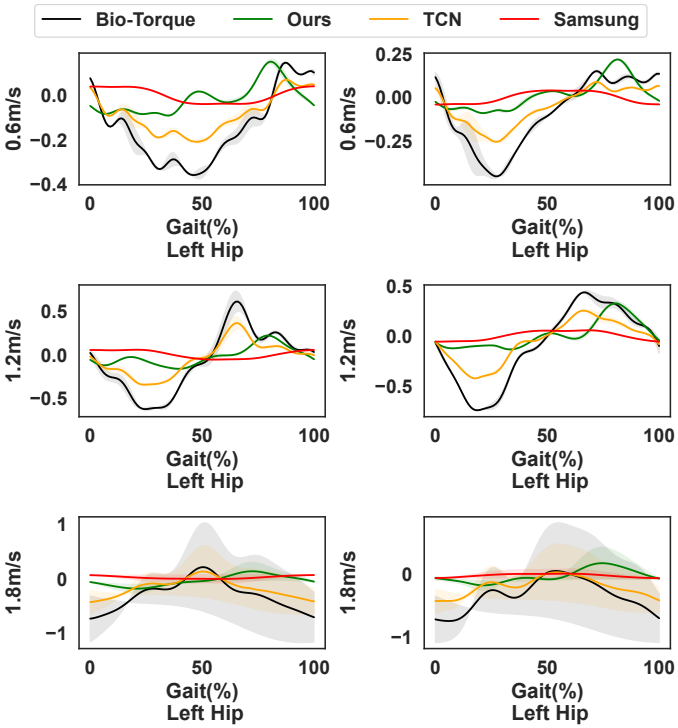


Fig. 11. Experiment results of assistive torque profiles within one gait cycle of all four methods at three different walking speeds.

employed to train an end-to-end exoskeleton control policy for other objectives.

C. Device-agnostic Zero-shot Policy Transfer for Real-world Deployment

An LSTM-based encoder was employed to solve the partial observation problem for policy deployment. It means that only the onboard sensors are required to collect human kinematic data. The learned exoskeleton control policy can be run on Pi 4B to infer the smooth assistive torque for varying walking speeds without using external sensors for ambulation classification and gait phase detection. Most importantly, in the simulation phase, the output assistive torque was directly applied to human target joints. Therefore, the learned exoskeleton control policy can be directly deployed on any isomorphic hip exoskeletons.

Device-agnostic policy learning has been considered a good way for real-world policy deployment. However, addressing the sim-to-real gap is not the focus of this article, which can be further improved by using domain randomization.

VII. Conclusion

This article demonstrates the effectiveness of learning an end-to-end exoskeleton control policy that can adapt to continuously changing walking speeds and be deployed on any isomorphic exoskeletons. Future work will focus on two main directions to further enhance the effectiveness and generalization ability of the learned policy. First, the proposed method will be extended to develop exoskeleton control policies for devices targeting other joints, such as the knee and ankle. Second, we aim to extend the framework to handle additional ambulation modes. Furthermore, the proposed approach can be integrated with other learning methods to develop a universal control policy that supports multiple ambulation modes seamlessly.

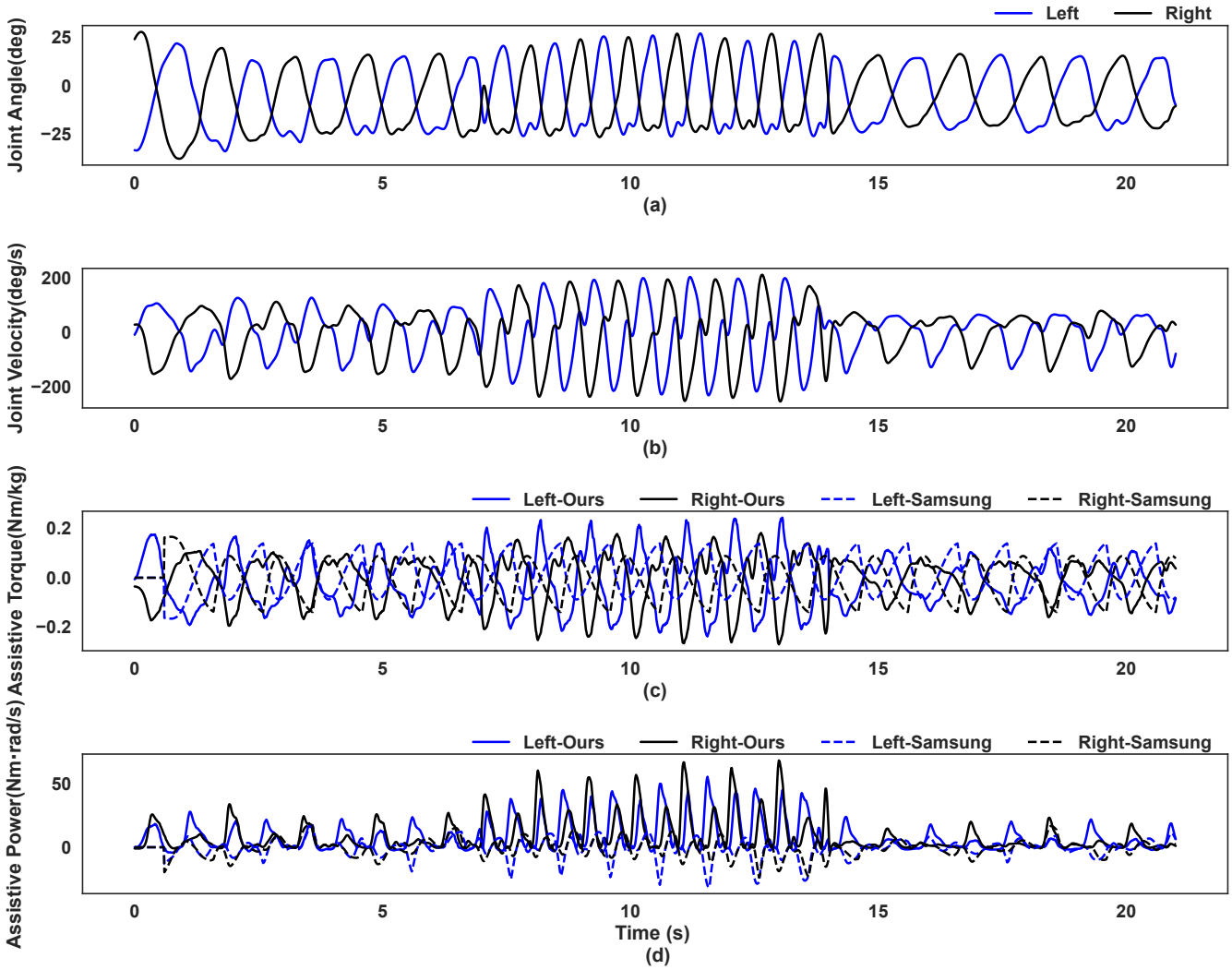


Fig. 12. Experiment results of biological torque and assistive torque profiles using four methods for outdoor walking.

VIII. Appendix

A. Human Control Policy Learning

The reward of tracking human joint angle $\overline{\boldsymbol{q}_j^h}$, joint velocity $\overline{\dot{\boldsymbol{q}}_j^h}$, and end-effector position $\overline{\boldsymbol{x}_j^h}$ are defined as follows:

$$\begin{aligned} r_{\boldsymbol{q}} &= \exp(-\beta_{\boldsymbol{q}} \sum\nolimits_j ||\overline{\boldsymbol{q}_j^h} - \boldsymbol{q}_j^h||^2) \\ r_{\boldsymbol{v}} &= \exp(-\beta_{\boldsymbol{v}} \sum\nolimits_j ||\overline{\dot{\boldsymbol{q}}_j^h} - \dot{\boldsymbol{q}}_j^h||^2) \\ r_{\boldsymbol{ee}} &= \exp(-\beta_{\boldsymbol{ee}} \sum\nolimits_j ||\overline{\boldsymbol{x}_j^h} - \boldsymbol{x}_j^h||^2) \\ r_h^c &= \exp(-\beta_{\boldsymbol{com}} |\Delta \boldsymbol{x}_0^{com}|) \end{aligned} \tag{15}$$

where $\beta_{\boldsymbol{q}} = 2.0$, $\beta_{\boldsymbol{v}} = 0.1$, $\beta_{\boldsymbol{ee}} = 40.0$, and $\beta_{\boldsymbol{com}} = 500000$ are the selected sensitive coefficients. The weighting coefficients are set as $\alpha_h^q = 0.75$, $\alpha_h^v = 0.1$, and $\alpha_h^c = 0.1$.

B. Exoskeleton Control Policy Learning

The human effort reward r_m can be designed to reduce the biological torque. The constraint reward r_e^c is defined to smooth actuation control commands, as follows:

$$r_m = -\boldsymbol{a}_{hip}^3 + \exp(-\beta_{\boldsymbol{\tau}} \sum\nolimits_j ||\boldsymbol{\tau}_j^{hip}||^2) \tag{16}$$

where $\boldsymbol{a}_{hip}$ is the hip related activated muscles, depicted in Fig. 7(a). $\boldsymbol{\tau}_{hip} \in \mathbb{R}^2$ is the biological torque of hip joints. All weighting coefficients are given as $\alpha_e^h = 0.9$, $\alpha_e^m = 0.3$, $\alpha_e^c = 0.1$, and $\beta_{\boldsymbol{\tau}} = 0.008$.

C. Experiment Details

1) Experimental Protocol of Metabolic Cost Measurement: The varying speed following the pattern as follows from 0.6m/s to 1.0m/s to 0.7m/s to 1.2m/s to 0.9m/s to 1.8m/s.

2) Customized Experience Questionnaires:

TABLE V
DETAILS OF CUSTOMIZED EXPERIENCE QUESTIONS

Index	Description	Answers [0-5]
1	Can you clearly perceive the timely assistance provided by the exoskeleton?	4
2	Do you feel that the exoskeleton provides smooth and timely assistance when your walking speed changes?	5
3	How would you describe the difference in assistive force when the walking speed changes?	5
4	Can you perceive the change in the magnitude of assistive force when the walking speed changes?	4
5	The assistance from the exoskeleton made outdoor walking easier compared to no assistance.	5

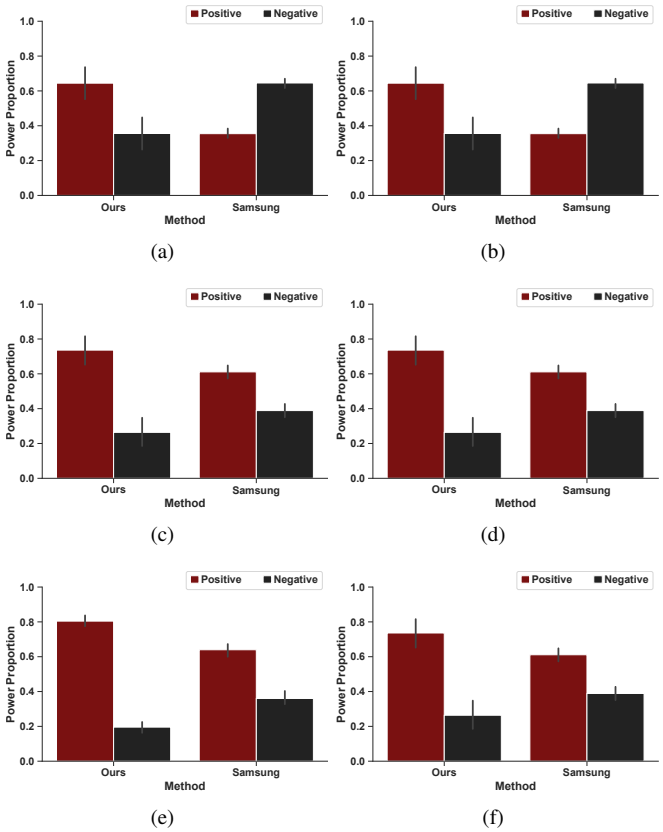


Fig. 13. Experiment results of proportion of positive power and timing of assistance across all participants at three fixed walking speeds. (a) Proportion of positive power at 0.6m/s. (b) Timing of assistance at 0.6m/s. (c) Proportion of positive power at 1.2m/s. (d) Timing of assistance at 1.2m/s. (e) Proportion of positive power at 1.8m/s. (f) Timing of assistance at 1.8m/s.

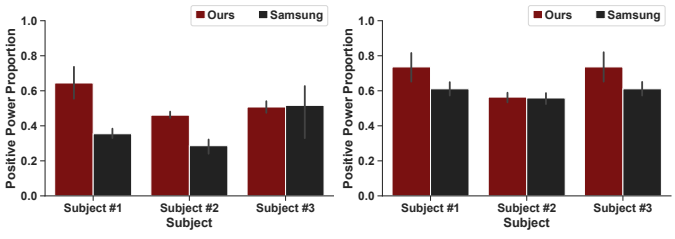


Fig. 14. Experiment results of metabolic cost reduction and completion time across all ten participants for outdoor walking. (a) Comparison of metabolic cost reduction. (b) Comparison of completion time.

REFERENCES

[1] S. Luo, M. Jiang, S. Zhang, J. Zhu, S. Yu, I. Dominguez Silva, T. Wang, E. Rouse, B. Zhou, H. Yuk *et al.*, "Experiment-free exoskeleton assistance via learning in simulation," *Nature*, vol. 630, no. 8016, pp. 353–359, 2024.

[2] D. Lee, S. Lee, and A. J. Young, "Ai-driven universal lower-limb exoskeleton system for community ambulation," *Science Advances*, vol. 10, no. 51, p. eadq0288, 2024.

[3] D. D. Molinaro, I. Kang, and A. J. Young, "Estimating human joint moments unifies exoskeleton control, reducing user effort," *Science Robotics*, vol. 9, no. 88, p. eadi8852, 2024.

[4] M. K. Ishmael, D. Archangeli, and T. Lenzi, "A powered hip exoskeleton with high torque density for walking, running, and stair ascent," *IEEE/ASME transactions on mechatronics*, vol. 27, no. 6, pp. 4561–4572, 2022.

[5] S. Yu, T.-H. Huang, X. Yang, C. Jiao, J. Yang, Y. Chen, J. Yi, and H. Su, "Quasi-direct drive actuation for a lightweight hip exoskeleton with high backdrivability and high bandwidth," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 4, pp. 1794–1802, 2020.

[6] R. Baud, A. R. Manzoori, A. Ijspeert, and M. Bouri, "Review of control strategies for lower-limb exoskeletons to assist gait," *Journal of neuroengineering and rehabilitation*, vol. 18, pp. 1–34, 2021.

[7] T. Ma, Y. Wang, X. Chen, C. Chen, Z. Hou, H. Yu, and C. Fu, "A piecewise monotonic smooth phase variable for speed-adaptation control of powered knee-ankle prostheses," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8526–8533, 2022.

[8] Y. Qian, C. Chen, J. Xiong, Y. Wang, Y. Leng, H. Yu, and C. Fu, "Terrain-adaptive exoskeleton control with predictive gait mode recognition: A pilot study during level walking and stair ascent," *IEEE Transactions on Medical Robotics and Bionics*, vol. 6, no. 1, pp. 281–291, 2024.

[9] E. Tricomi, G. Piccolo, F. Russo, X. Zhang, F. Missiroli, S. Ferrari, L. Gionfrida, F. Ficuciello, M. Xiloyannis, and L. Masia, "Leveraging geometric modeling-based computer vision for context aware control in a hip exosuit," *IEEE Transactions on Robotics*, 2025.

[10] Y. Qian, Y. Wang, C. Chen, J. Xiong, Y. Leng, H. Yu, and C. Fu, "Predictive locomotion mode recognition and accurate gait phase estimation for hip exoskeleton on various terrains," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6439–6446, 2022.

[11] S. Zhang, J. Zhu, T.-H. Huang, S. Yu, J. S. Huang, I. Lopez-Sanchez, T. Devine, M. Abdelhady, M. Zheng, T. C. Bulea *et al.*, "Actuator optimization and deep learning-based control of pediatric knee exoskeleton for community-based mobility assistance," *Mechatronics*, vol. 97, p. 103109, 2024.

[12] J. Lee, W. Hong, and P. Hur, "Continuous gait phase estimation using lstm for robotic transfemoral prosthesis across walking speeds," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1470–1477, 2021.

[13] C. P. O. Nuesslein and A. J. Young, "A deep learning framework for end-to-end control of powered prostheses," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 3988–3994, 2024.

[14] R. L. Medrano, G. C. Thomas, C. G. Keais, E. J. Rouse, and R. D. Gregg, "Real-time gait phase and task estimation for controlling a powered ankle exoskeleton on extremely uneven terrain," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 2170–2182, 2023.

[15] Y. Ding, M. Kim, S. Kuindersma, and C. J. Walsh, "Human-in-the-loop optimization of hip assistance with a soft exosuit during walking," *Science robotics*, vol. 3, no. 15, p. eaar5438, 2018.

[16] P. Slade, M. J. Kochenderfer, S. L. Delp, and S. H. Collins, "Personalizing exoskeleton assistance while walking in the real world," *Nature*, vol. 610, no. 7931, pp. 277–282, 2022.

[17] K. A. Ingraham, C. D. Remy, and E. J. Rouse, "The role of user preference in the customized control of robotic exoskeletons," *Science robotics*, vol. 7, no. 64, p. eabj3487, 2022.

[18] U. H. Lee, V. S. Shetty, P. W. Franks, J. Tan, G. Evangelopoulos, S. Ha, and E. J. Rouse, "User preference optimization for control of ankle exoskeletons using sample efficient active learning," *Science Robotics*, vol. 8, no. 83, p. eadg3705, 2023.

[19] Q. Zhang, J. Si, X. Tu, M. Li, M. D. Lewek, and H. Huang, "Toward task-independent optimal adaptive control of a hip exoskeleton for locomotion assistance in neurorehabilitation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2024.

[20] B. Lim, J. Lee, J. Jang, K. Kim, Y. J. Park, K. Seo, and Y. Shim, "Delayed output feedback control for gait assistance with a robotic hip exoskeleton," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 1055–1062, 2019.

[21] B. Lim, B. Choi, C. Roh, S. Hyung, Y.-J. Kim, and Y. Lee, "Parametric delayed output feedback control for versatile human-exoskeleton interactions during walking and running," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4497–4504, 2023.

[22] D. D. Molinaro, K. L. Scherpereel, E. B. Schonhaut, G. Evangelopoulos, M. K. Shepherd, and A. J. Young, "Task-agnostic exoskeleton control via biological joint moment estimation," *Nature*, vol. 635, no. 8038, pp. 337–344, 2024.

[23] L. Rose, M. C. Bazzocchi, and G. Nejat, "A model-free deep reinforcement learning approach for control of exoskeleton gait patterns," *Robotica*, vol. 40, no. 7, pp. 2189–2214, 2022.

[24] S. Song, Ł. Kidziński, X. B. Peng, C. Ong, J. Hicks, S. Levine, C. G. Atkeson, and S. L. Delp, "Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation," *Journal of neuroengineering and rehabilitation*, vol. 18, pp. 1–17, 2021.

[25] M. Li, Y. Wen, X. Gao, J. Si, and H. Huang, "Toward expedited impedance tuning of a robotic prosthesis for personalized gait assistance by reinforcement learning control," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 407–420, 2021.

[26] Z. Hou, T. Ma, W. Wang, and H. Yu, "Contextual policy search for task-level adaptation in physical human–robot interaction," *IEEE/ASME Transactions on Mechatronics*, 2025.

[27] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal, "Rapid locomotion via reinforcement learning," *The International Journal of Robotics Research*, vol. 43, no. 4, pp. 572–587, 2024.

[28] S. Lee, M. Park, K. Lee, and J. Lee, "Scalable muscle-actuated human simulation and control," *ACM Transactions On Graphics (TOG)*, vol. 38, no. 4, pp. 1–13, 2019.

[29] N. Wagener, A. Kolobov, F. Vieira Frujeri, R. Loynd, C.-A. Cheng, and M. Hausknecht, "Mocapact: A multi-task dataset for simulated humanoid control," *Advances in Neural Information Processing Systems*, vol. 35, pp. 35 418–35 431, 2022.

[30] A. Tang, T. Hiraoka, N. Hiraoka, F. Shi, K. Kawaharazuka, K. Kojima, K. Okada, and M. Inaba, "Humanmimic: Learning natural locomotion and transitions for humanoid robot via wasserstein adversarial imitation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 107–13 114.

[31] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. Pmlr, 2018, pp. 1861–1870.

[32] C. Zuo, K. He, J. Shao, and Y. Sui, "Self model for embodied intelligence: Modeling full-body human musculoskeletal system and locomotion control with hierarchical low-dimensional representation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 062–13 069.

[33] C. Berg, V. Caggiano, and V. Kumar, "Sar: Generalization of physiological agility and dexterity via synergistic action representation," *Autonomous Robots*, vol. 48, no. 8, p. 28, 2024.

[34] Y. Feng, X. Xu, and L. Liu, "Musclevae: Model-based controllers of muscle-actuated characters," in *SIGGRAPH Asia 2023 Conference Papers*, 2023, pp. 1–11.

[35] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: Adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 4, pp. 1–20, 2021.

[36] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for robust parameterized locomotion control of bipedal robots," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2811–2817.

[37] G. Feng, H. Zhang, Z. Li, X. B. Peng, B. Basireddy, L. Yue, Z. Song, L. Yang, Y. Liu, K. Sreenath *et al.*, "Genloco: Generalized locomotion controllers for quadrupedal robots," in *Conference on Robot Learning*. PMLR, 2023, pp. 1893–1903.

[38] D. Kim, G. Berseth, M. Schwartz, and J. Park, "Torque-based deep reinforcement learning for task-and-robot agnostic learning on bipedal robots using sim-to-real transfer," *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6251–6258, 2023.

[39] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.

[40] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[41] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[42] R. V. Schulte, E. C. Prinsen, L. Schaake, R. P. Paassen, M. Zondag, E. S. van Staveren, M. Poel, and J. H. Buurke, "Database of lower limb kinematics and electromyography during gait-related activities in able-bodied subjects," *Scientific Data*, vol. 10, no. 1, p. 461, 2023.

[43] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[44] D. Ullman and B. F. Malle, "Measuring gains and losses in human-robot trust: Evidence for differentiable components of trust," in *2019 14th ACM/IEEE international Conference on human-robot interaction (HRI)*. IEEE, 2019, pp. 618–619.

[45] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.

[46] S. Luo, G. Androwis, S. Adamovich, E. Nunez, H. Su, and X. Zhou, "Robust walking control of a lower limb rehabilitation exoskeleton coupled with a musculoskeletal model via deep reinforcement learning," *Journal of neuroengineering and rehabilitation*, vol. 20, no. 1, p. 34, 2023.

[47] H.-J. Geiβ, F. Al-Hafez, A. Seyfarth, J. Peters, and D. Tateo, "Exciting action: Investigating efficient exploration for learning musculoskeletal humanoid locomotion," in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*. IEEE, 2024, pp. 205–212.