2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Deep Reinforcement Learning for Hip Exoskeleton Control via Predictive Simulation of Reflex-Based Human Gait

Hossein Barati[1], Sangdo Kim[2,3], Nguyen Thanh Xuan[1], Jongwon Lee[2,†], Young Jin Park[1,†]

Abstract— **Lower-limb exoskeletons have the potential to enhance mobility and reduce the metabolic cost of walking, while conventional control strategies often lack adaptability and require labor-intensive tuning. Recent advances in reinforcement learning (RL) provide new opportunities for generating efficient and personalized assistance. In this study, we propose a predictive simulation framework that integrates a reflex-based musculoskeletal walking model with a hip exoskeleton controller trained using Proximal Policy Optimization (PPO) with a Long Short-Term Memory (LSTM) actor network. The reflex-based model reproduces realistic gait kinematics without relying on experimental motion data, while the LSTM-PPO controller learns to map kinematic states directly to assistive torques. Domain randomization was applied during training to enhance robustness and facilitate sim-to-real transfer. The learned controller was deployed onto a physical hip exoskeleton and evaluated in human subject experiments. Results showed that the LSTM-PPO controller reduced the metabolic cost of walking by an average of 9.1%. These findings highlight the potential of predictive simulation and deep RL for developing intelligent, experiment-free exoskeleton controllers that improve walking efficiency and robustness in real-world conditions.**

I. INTRODUCTION

Assistive robots have been developed to support various types of human body movement [1]. In particular, lower-limb exoskeletons have demonstrated the ability to provide support and improve human performance during gait cycles, both for able-bodied individuals [2] and for people with gait impairments as part of rehabilitation [3]. Along with the rapid growth of exoskeleton technology, several control algorithms have also been developed. Broadly, these algorithms can be categorized into conventional control techniques and AI-based approaches such as reinforcement learning (RL) [4].

Studies show that RL algorithms outperform conventional methods in cost efficiency, reliability, and adaptability to different environments and situations [5]. Traditional control algorithms face several limitations: they are not adaptive to individual users or environments; they require time- and energy-consuming tuning of the assistance torque profile for each user; and they struggle with transitions across different locomotion modes because they rely on detecting and classifying gait phases [6].

In recent years, various deep reinforcement learning algorithms have been introduced to overcome these challenges. Given that motor torque control in robotic systems inherently involves continuous action spaces, these algorithms have demonstrated strong robustness in handling them [7] while also generating optimal control policies [8]. However, safety and reliability remain crucial considerations in RL-based control of assistive and rehabilitation robots [4]. Musculoskeletal simulations provide an opportunity to develop intelligent exoskeleton controllers and safely evaluate the controllers' performance and biomechanical effects on human gait prior to real-world deployment [9]. Several studies have successfully trained and developed RL-based exoskeleton controllers in simulation, and more recently, some have demonstrated deployment of learned controllers on real exoskeletons [6], [9]. Despite these advances, important challenges remain:

- Accurately simulating a musculoskeletal model that reproduces practical human gait data is difficult, requiring models with a high number of degrees of freedom (DOFs) and muscles. This complexity makes controller learning more challenging.
- Coupling the exoskeleton model with the human model and accurately defining interaction forces further increases system complexity and complicates sim-to-real transfer.
- Most existing deep RL approaches map human walking kinematics to assistive torque through complex reward functions that track experimental reference data. This approach may reduce the generalizability and adaptability of the learned policy to new users, while also limiting the exploration of the action space, which is critical in RL algorithms.

To address the aforementioned challenges, we propose a framework that utilizes a simplified musculoskeletal model, as modeling complex biomechanical structures can significantly affect the accuracy of human gait reproduction [10], [11]. Moreover, controlling high-dimensional models is difficult and may result in reduced accuracy [12]. We then integrated our model with a virtual hip exoskeleton that applies

*This work was supported in part by the KETEP grant funded by the Ministry of Trade, Industry & Energy (MOTIE), Republic of Korea (No. 20241K00000010; Robot Utilization Production Support Center for SMR), in part by the Glocal University 30 Project Fund of Gyeongsang National University in 2026.

[1]Hossein Barati, Nguyen Thanh Xuan, and Young Jin Park are with the School of Mechanical Engineering, College of Engineering, Gyeongsang National University, Jinju-si, Gyeongsangnam-do, 52828, Republic of Korea hosseinbarati39@gmail.com, xuan.hn12@gnu.ac.kr, youngjin.park@gnu.ac.kr

[2]Sangdo Kim, Jongwon Lee are with Artificial Intelligence and Robotics Institute, Korea Institute of Science and Technology, Seoul, South Korea sangdo322@kist.re.kr, jwlee@kist.re.kr

[3]Sangdo Kim is also affiliated with School of Electrical Engineering, Korea University, Seoul, South Korea ysmc1440@korea.ac.kr

[†]These authors contributed equally to this work and serve as co-corresponding authors jwlee@kist.re.kr; youngjin.park@gnu.ac.kr

idealized assistive torques at the hip joints. This approach isolates the control policy from device-level complexities, such as actuator dynamics and attachment compliance, and enables the RL agent to discover torque patterns that enhance human walking [13]. The musculoskeletal model is actuated via reflex-based muscle controllers to replicate human walking as closely as possible to experimental data in terms of gait kinematics, ground reaction forces, and muscle activations. Simultaneously, the deep RL algorithm processes human kinematics data such as hip angles and angular velocities to generate assistive torque profile.

We adopt the Proximal Policy Optimization (PPO) algorithm with an Long Short-Term Memory (LSTM)-based actor network, which maps human state directly to assistive torque patterns. To improve sim-to-real transfer, we apply domain randomization before deploying the learned policy on a hip exoskeleton. The controller is evaluated in human subject trials in terms of metabolic cost reduction, speed adaptability, and similarity of torque patterns between simulation and the real device. The overall process of this study is depicted in Fig. 1. The main contributions of this work are as follows:

- Proposing an LSTM-PPO algorithm that learns temporal dependencies within the gait cycle and generates assistive torque patterns for a hip exoskeleton directly from measurable human kinematics data.
- Implementing a fully predictive simulation approach, where the learning process requires no experimental data for either human walking generation or reward function design in the exoskeleton controller.
- Integrating the proposed deep RL algorithm with a reflex-based musculoskeletal model to accelerate training for human gait assistance and reduce system complexity.
- Deploying the learned controller directly onto a hip exoskeleton and validating its performance in human subject experiments.

II. RELATED WORKS

A. Simulation-Only Studies

Several studies have developed controllers for assistive devices such as exoskeletons and prosthetics using reinforcement learning in musculoskeletal simulations [13], [14]. Although these works accurately simulate musculoskeletal models and achieve strong policy performance in simulation, evaluating the practical effectiveness of the learned controllers on real devices remains critical.

In this work, we first trained our controller through predictive simulation and subsequently deployed the trained policy on a real hip exoskeleton. The performance of the learned controller was validated by measuring reductions in the metabolic cost of walking and comparing the generated assistive torque patterns between simulation and real-world trials.

B. Studies Without Simulation

As mentioned earlier, ensuring the safety and beneficial effects of the learned controller on human gait is crucial. This issue can be evaluated through a robust simulation using a sensory-feedback neuromechanical controller [9], which can faithfully reproduce human gait characteristics (including kinematics, muscle activations, and control mechanisms) and generate appropriate responses to robotic assistance. Several prior approaches [15]–[17] placed greater emphasis on hardware implementation and experimental results, while not accurately considering the effects of their controllers on musculoskeletal models.

In this study, we employ a reflex-based simulation to reproduce human gait without relying on experimental data for motion imitation. This approach allows us to optimize controller performance in simulation, thereby increasing confidence in its effect on human gait before practical implementation.

C. Studies Combining Simulation and Real Deployment

Recent work can be considered state-of-the-art in developing controllers in simulation and implementing them on real robots [6]. This study simulated human gait models with high DOF and numerous muscles, coupled them with exoskeleton models, and applied domain randomization to successfully deploy a PPO-based controller on hardware. Another study employed a similar methodology but extended the analysis to compare muscle activations between simulation and experiments [9].

Although these studies demonstrated strong performance, their controller training relied on reward functions designed

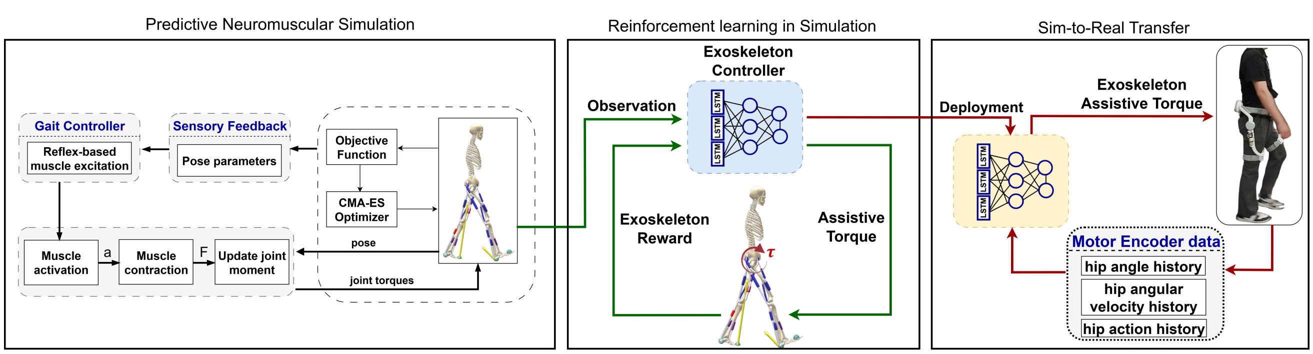


Fig. 1. Overall process of the proposed intelligent exoskeleton controller, from simulation training to real-world deployment.

to track experimental reference patterns, which limited flexibility in generating assistive torque profiles. Furthermore, the exoskeleton control network generates joint positions, which are then converted into assistive torque through PD control, indicating that kinematic gait is not being mapped directly to the assistive torque.

Unlike prior works that rely on experimentally recorded motion data in the reward, our approach uses a reflex-based musculoskeletal model as an experiment-free surrogate, enabling the entire training pipeline to operate without any human subject data. Moreover, the proposed algorithm generates a real-time, continuous assistive torque profile directly from kinematic inputs that are readily measurable in practice, without relying on indirect mappings such as PD control [6].

III. METHOD

A. Human Walking Agent via Reflex-based Muscle Actuation

To train the exoskeleton controller within a musculoskeletal simulation, a human model was first required as an agent capable of walking in an environment. The accuracy of the generated gait kinematics is crucial, as they provide the observations for the deep RL algorithm in the exoskeleton control network. The more closely these simulated data resemble actual human gait, the more the sim-to-real deployment gap is bridged.

In this study, we employed a reflex-based controller to generate human gait. As demonstrated in [12], this approach effectively reproduces human kinematics with low energy expenditure in the planar case and produces more natural gait patterns compared to RL methods.

We used the HyFyDy simulation engine [18] to implement the reflex-based walking agent, owing to its realistic and accurate biomechanical modeling [12]. The musculoskeletal model was planar, comprising 9 degrees of freedom (DOFs) and 18 Hill-type musculotendon actuators and a gait controller activates each muscle through a state-dependent reflex-based mechanism based on Geyer and Herr [19]. Parameters such as reflex gains and offsets were optimized using the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [20] within SCONE [21], an open-source framework for predictive neuromuscular simulations. The objective function and optimization details were adopted from a prior study [22] to generate normal human gait. The update frequency of all kinematic data in the simulation is 100 Hz.

B. Deep Reinforcement Learning in Predictive Simulation

1) PPO Algorithm: After defining the human walking agent, the next step was to select a deep RL algorithm to train the exoskeleton controller. The controller must autonomously learn an optimal assistive torque policy by interacting with the agent. In this work, we adopted PPO [23] for its robustness, stability, and sample efficiency, as well as its proven success in robotics applications [24], [25].

PPO is a policy gradient method that seeks the optimal policy π_θ by maximizing expected cumulative rewards through iterative updates of policy parameters θ. It combines the advantages of actor-critic architectures, where the actor produces actions according to the policy and the critic estimates the value function to guide updates. PPO optimizes policy parameters by maximizing the clipped surrogate objective:

$$L^{\mathrm{CLIP}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t,\ \mathrm{clip}\left(r_t(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_t\right)\right] \tag{1}$$

Here, $r_t(\theta) = \dfrac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{old}}(a_t \mid s_t)}$ denotes the probability ratio measuring the policy update, and $\hat{A}_t$ is the advantage estimate provided by the critic. The constant ε limits the probability ratio and the clipping parameter prevents overly large updates, thus ensuring stable learning.

2) Exoskeleton Control Policy Design: The exoskeleton control policy $\pi_\theta(\mathbf{a}_{exo} \mid \mathbf{s}_{exo})$ was designed to assist human walking by mapping gait kinematics history states $\mathbf{s}_{exo}$, including joint angles, angular velocities, and previous actions, to continuous torque outputs $\mathbf{a}_{exo}$.

To enable the controller to capture temporal dependencies in gait, we incorporated a LSTM network as the actor within PPO. LSTMs have proven effective at modeling sequential features of locomotion [26]–[28], and their integration into PPO has shown promising results for legged locomotion tasks [29], [30]. This structure allows the controller to exploit proprioceptive histories and learn temporal walking patterns.

The actor network was implemented in PyTorch and consists of an LSTM encoder followed by Multilayer Perceptron (MLP) layers. Its input vector consisted of 10 time steps histories of hip angles, hip angular velocities, and hip torque actions for both legs, resulting in a 60-dimensional input. These inputs were processed through a single LSTM layer with 128 hidden units, followed by a multilayer perceptron with a hidden layer of 64 units and Rectified Linear Unit (ReLU) activation. The output layer applied a linear transformation with Tanh activation, scaled by a user-defined vector to match physical torque limits. The outputs corresponded to right and left hip assistive torques. The network architecture is shown in Fig. 2.

The critic network was constructed as a feedforward MLP, receiving the same 60-dimensional input and passing

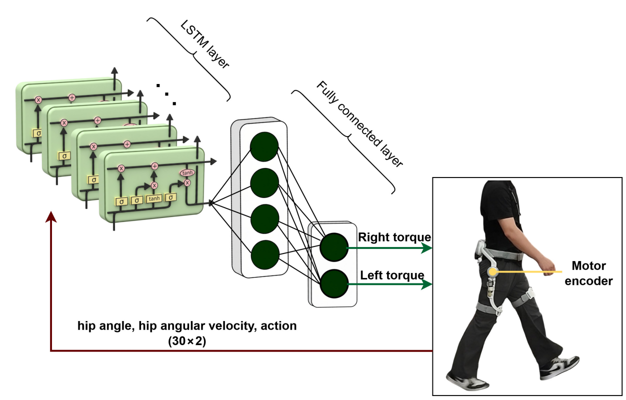


Fig. 2. Schematic illustration of the actor network functioning as the exoskeleton controller.

it through two hidden layers with 128 and 64 units, each with ReLU activation. The network output was a scalar representing the estimated state value.

The objective was to learn an optimal continuous policy π_θ^* for maximizing the discounted reward:

$$R_{exo} = r_{effort} + r_{vel} + r_s + r_{fall} \tag{2}$$

The reward function comprises several terms that reduce muscle effort and preserve balance. The subrewards were defined as follows and their weights and parameters are summarized in Table I.

Muscle Effort Reward (r_{effort}): Rewards the agent for minimizing muscle activation. Because muscle activation is a strong predictor of metabolic cost [31], [32], penalizing the muscle-activation term is expected to reduce real-world energy expenditure. Effort was calculated separately for each leg with equal weights, promoting symmetry without an explicit symmetry term.

$$r_{\text{effort}} = \sum_{*\in\{\text{L, R}\}} w_{\text{eff},*} \times \left(1 - \tanh\left[\frac{1}{N}\sum_{i=1}^{N}(\text{mus. activ.}_{*,i})^2\right]\right) \tag{3}$$

where N is the number of muscles per leg. L and R denote left and right legs, respectively.

Velocity Tracking Reward (r_{vel}): Encourages walking at the target speed, promoting steady locomotion.

$$r_{vel} = w_{\text{v}} \times (1 - \tanh(|v - v_{target}|)) \tag{4}$$

where v is the forward velocity and v_{target} is the target velocity.

Smoothness Penalty (r_s): Penalizes abrupt torque changes between time steps, promoting smoother profiles.

$$r_s = -\lambda_s\left[(a_R - a_{R,prev})^2 + (a_L - a_{L,prev})^2\right] \tag{5}$$

where a_{R} and a_{L} denote the current right and left hip actions, and $a_{\text{R,prev}}$ and $a_{\text{L,prev}}$ represent the actions applied to the hip joint actuators in the previous time step. λ_s is the smoothness coefficient.

Fall Penalty (r_{fall}): Applies a negative reward if vertical position of the center of mass (y_{com}) drops below a threshold, preventing falls.

$$r_{fall} = \begin{cases} -10, & \text{if } y_{com} < h_{com,min}, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

During training, domain randomization was applied by injecting Gaussian noise into hip angles and angular velocities. Specifically, zero-mean noise with standard deviations of 0.05 rad for angles and 0.1 rad/s for velocities was added to both legs, improving robustness to sensor uncertainty. The overview of learning process in presented in Fig. 3.

Simulations were executed through the SCONE Python API on a workstation with an AMD Ryzen™ Threadripper™ PRO 5955WX CPU, an NVIDIA® GeForce RTX™ 4090 GPU, and 128 GB RAM. The complete simulation required slightly under three hours to finish and PPO training hyperparameters are provided in Table II.

TABLE I
REWARD FUNCTION PARAMETERS

Parameters	Values
$w_{\text{eff,L}}$	0.5
$w_{\text{eff,R}}$	0.5
w_v	0.2
$h_{com,min}$	0.8
v_{target}	1.3
λ_s	0.1

TABLE II
HYPERPARAMETERS USED IN TRAINING

Parameters	Values	Parameters	Values
Discount Factor	0.99	Clip Threshold	0.2
Entropy	0.01	Learning Rate	0.0003
Epochs	20	Memory Buffer	4096
Max Iteration	90000	Batch Size	256

IV. EXPERIMENTS

A. Implementation of Learned Exoskeleton Controller

The trained actor network parameters, representing the learned exoskeleton controller, were deployed directly to a portable hip exoskeleton for evaluation on able-bodied participants. The robot is a lightweight 4-DOF wearable hip-assistive device weighing 2.9 kg, designed to support bilateral hip flexion–extension (FE) and abduction–adduction (AbAd) movements [33]. In this study, the two FE degrees of freedom were utilized as active joints driven by actuators to deliver assistive torque, while the two AbAd degrees of freedom were configured as passive joints to allow natural lateral hip movement without actuation. To ensure accurate hip joint angle and velocity sensing, motor-embedded encoders provide real-time leg-motion feedback. High-level control, executed on the onboard Arduino Portenta X8 computer, performs assistive-torque planning, in which the trained LSTM-PPO actor network is hard-coded to map the measured kinematic states directly to the desired assistive torques. Mid-level control is managed by motor drivers with integrated current sensors to regulate torque output precisely, and low-level control at each actuator ensures accurate motor commands. All sensing and actuation components operate at a 100 Hz control frequency, enabling timely detection of movement intentions and delivering up to 15 Nm of peak torque to assist the user.

B. Assessment

1) Metabolic Cost of Walking: To evaluate the efficiency of the proposed LSTM-PPO controller, we measured the metabolic energy expenditure during treadmill walking at a constant speed, both with and without exoskeleton assistance as in Fig. 4. For a fair comparison, we benchmarked our approach against a state-of-the-art conventional control method, Delayed Output Feedback Control (DOFC) [34], due to its demonstrated effectiveness in improving human gait.

a) Participants: Four healthy male participants (age: 24.0 ± 0.8; weight: $67.7 \pm 5.0kg$; height: $173.0 \pm 3.39cm$) with no known medical conditions participated in the metabolic cost experiment, which was approved by the Korea

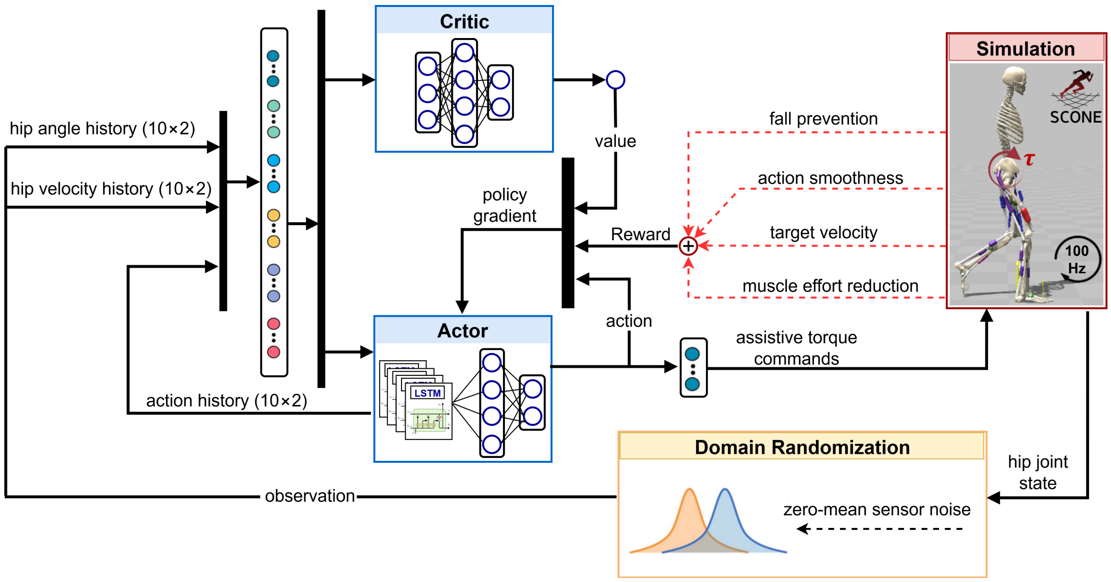


Fig. 3. Overview of learning process of exoskeleton controller in predictive simulation.

Institute of Science and Technology Institutional Review Board (KIST-202310-HR-003). Walking speed was fixed at 4.9 km/h, and the peak hip assistive torque was set to approximately 8 Nm for both LSTM-PPO and DOFC condition (with 0.2 s time-delay). The experimental protocol is summarized in Table III and the order of algorithm exposure was randomized across subjects to mitigate ordering effects.

b) Measurement: Metabolic energy consumption was measured using the K5 portable metabolic system (COSMED, Italy). The metabolic rate (MR) was computed using a modified Brockway equation [35] and normalized by body weight. Net metabolic rate (NMR) was calculated following the procedure described in [36].

2) Gait Speed Adaptability: To test speed adaptability, assistance torques were applied at multiple walking speeds. Participants walked at 3, 4, and 5 km/h under LSTM-PPO assistance condition. Each trial lasted one minute, with the last 30 seconds used to compute root mean squared (RMS) torque.

V. RESULTS AND DISCUSSION

A. Metabolic cost reduction

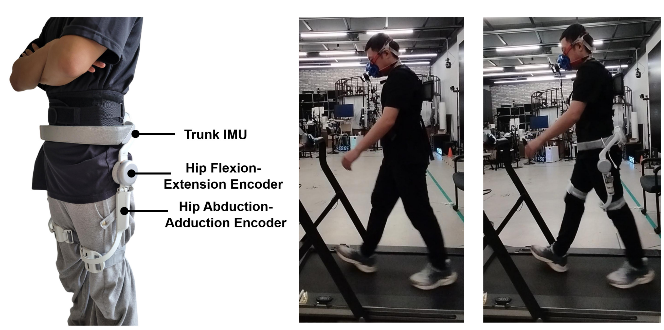
As reported in Table IV, our proposed LSTM-PPO algorithm, trained in simulation, reduced the metabolic cost of walking in all subjects compared to walking without the exoskeleton, with an average reduction of 9.1%. This reduction of metabolic cost is greater than that achieved by the DOFC algorithm, which averaged 5.8%. In addition, Fig. 5 shows a significant difference (p-value < 0.05) in the net metabolic rate values between LSTM-PPO algorithm assistance and no exo condition. These results demonstrate that our AI-powered algorithm outperforms conventional control methods under similar experimental conditions.

TABLE III
EXPERIMENTAL PROTOCOL

Condition	Stand	Free1	Rest	Exo-RL	Rest	Exo-DOFC	Rest	Free2	Stand
Time (min)	5	6	5	6	5	6	5	6	5

The reductions observed with both LSTM-PPO and DOFC are not competitive compared to those of state-of-the-art exoskeletons reported in the literature [6], [34]. Because our focus was on AI performance, we intentionally reduced the magnitude of the assistive torque, which makes the DOFC performance appear relatively small compared to the outcomes reported in [34]. However, the primary goal of this experiment was to validate the feasibility of our learned controller on a real exoskeleton and to evaluate its relative performance against a conventional control approach within a single experimental protocol, which had not been conducted in previous studies.

We also investigated metabolic cost reduction within the predictive simulation framework to compare with our experimental findings. Simulations were run with and without



(a) (b) (c)

Fig. 4. Robot configuration and assessment process. (a) Lower-limb wearable hip exoskeleton MoonWalk.Omni. (b) Metabolic cost measurement without exoskeleton. (c) Metabolic cost measurement with exoskeleton.

TABLE IV

SUBJECT DATA, EXOSKELETON ASSISTIVE TORQUE, AND METABOLIC COST MEASUREMENT

Subject No.	Age	Weight (kg)	Assist mode	Exo assistive torque τ_{max} (Nm)	Metabolic cost measurement NMR Reduction (Exo vs Free, %)
1	23	66	LSTM-PPO	7.7	**7.9**
			DOFC	7.4	1.2
2	24	69	LSTM-PPO	7.8	**11.6**
			DOFC	7.5	6.9
3	25	74	LSTM-PPO	6.8	4.2
			DOFC	7.0	**5.4**
4	24	62	LSTM-PPO	7.7	**12.7**
			DOFC	7.9	9.5
Mean (±SD)	24.0 ± 0.8	67.7 ± 5.0	LSTM-PPO	7.5 ± 0.5	**9.1** ± **3.9**
			DOFC	7.4 ± 0.4	5.8 ± 3.5

assistance, and the cost of transport (COT) reduction was calculated using the definition provided in [37], implemented in SCONE software. The result showed a 17.8% reduction in metabolic cost during walking in simulation. This reduction aligns with the experimental trend, though the difference in magnitude arises from the differences in simulation and experiment condition.

B. Gait kinematics and assistive torque profile

Reflex-based controllers are known to produce realistic gait kinematics, particularly for the hip joint, though their accuracy is somewhat lower for other joints such as the ankle [12]. The gait data obtained from the robot's motor encoders, including angle and velocity, support this observation, as shown in Fig. 6.

For a quantitative comparison between gait kinematics in simulation and experiment, we used a normalized cross-correlation method described in [38] to measure shape agreement. Both hip angle and angular velocity demonstrated strong cross-correlation values: $R = 0.97$ for hip angle and $R = 0.83$ for hip angular velocity. This high degree of correlation gives confidence that these data can serve as reliable inputs for intelligent robot control.

Although real-world signals inevitably contain noise and conditions are less ideal than in simulation, we addressed this issue by applying domain randomization [39] to bridge the sim-to-real gap. To further verify consistency between simulation and experiment, we compared the generated assistive torque profiles. As shown in Fig. 6, the assistive torque applied to the model in the simulation and to the human body in the experiment exhibit a strong similarity in both shape and timing. The high cross-correlation value $R = 0.93$ supports this claim. This comparison, which has not been carried out in previous studies, is important for demonstrating how effectively we have addressed the gap between simulation and reality.

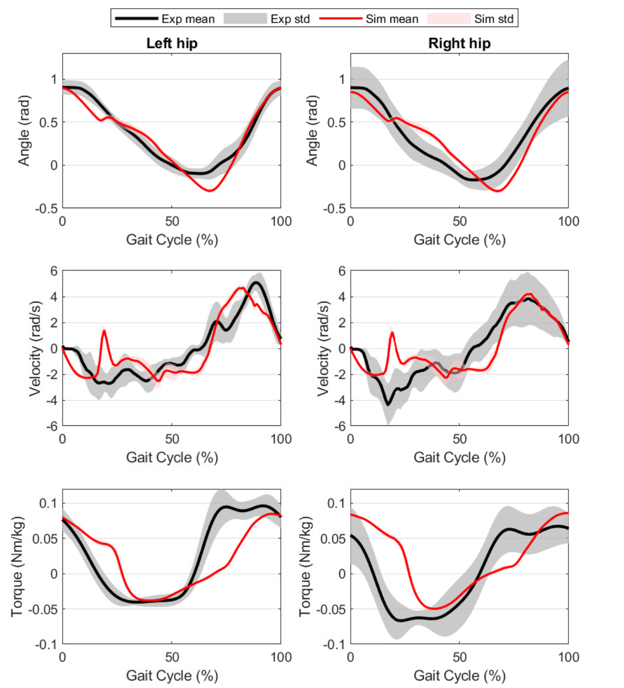


Fig. 6. Comparison of hip joint angle, velocity, and assistive torque between simulation and experiment.

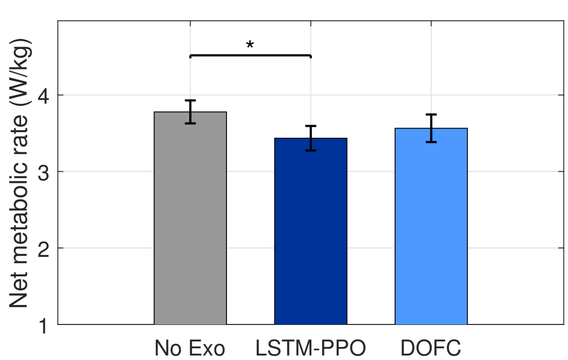


Fig. 5. Comparison of net metabolic rate. (* p-value < 0.05)

C. Gait speed adaptability

Although the exoskeleton controller was trained in simulation under a single fixed gait speed, it demonstrated

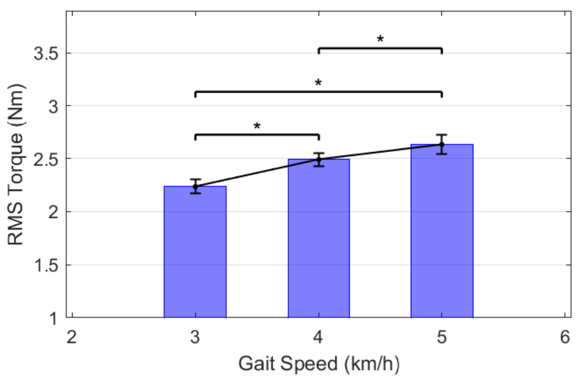


Fig. 7. Speed adaptability of torque assistance in LSTM-PPO algorithm. Horizontal brackets represent significant differences ($^{*}p$-value < 0.05)

adaptability to various unseen speeds during the experiment. Biologically, an increase in walking speed leads to greater demands for hip torque and power during gait [40]. We calculated the RMS value of assistive torque at three different gait speeds to evaluate the ability of the LSTM-PPO algorithm to adjust assistance as walking speed changes continuously. Fig. 7 illustrates the speed adaptability of the proposed algorithm, showing a significant difference (p-value < 0.05) across walking speeds. This capability eliminates the need for handcrafted rules in the algorithm, unlike traditional control approaches [36].

VI. CONCLUSIONS

In this study, we proposed a framework capable of learning to generate optimal assistance during walking with an exoskeleton through predictive musculoskeletal simulation. Predictive simulation enables the production of realistic human gait using reflex-based muscle actuation controllers while providing reliable human kinematic states to the RL-based exoskeleton controller. We leveraged the simplicity of the reflex-based approach to place greater focus on training the exoskeleton controller. Specifically, we proposed an LSTM-PPO algorithm that combines the ability of LSTM neural networks to capture temporal dependencies with the robustness of the PPO algorithm. This approach allows for the use of a simple reward function that does not rely on reference tracking terms and is entirely experiment-free.

We deployed the learned controller on a hip exoskeleton to evaluate its performance in assisting human gait. Experimental results showed that the proposed algorithm reduced the metabolic cost of walking more effectively than conventional control approaches in a single experiment. Furthermore, the LSTM-PPO algorithm demonstrated adaptability to various gait speeds. In addition, despite being based on a simplified planar model, the comparison of human kinematics and applied assistive torque between simulation and experiment revealed a strong correlation.

In this study, we used a planar 9-DOF musculoskeletal model, which may limit the generalizability of the simulation outcomes due to the neglect of frontal-plane control. Future work could extend the proposed framework to 3D musculoskeletal models to evaluate whether they achieve better performance than 2D models. In addition, the algorithm could be applied to other lower-limb exoskeletons with more complex kinematics than the hip joint. It may also be applied to individuals with gait impairments, as predictive simulation enables the modeling of various pathological gait patterns.

REFERENCES

[1] S. A. Murray, K. H. Ha, C. Hartigan, and M. Goldfarb, “An assistive control approach for a lower-limb exoskeleton to facilitate recovery of walking following stroke,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 23, no. 3, pp. 441–449, 2014.

[2] 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.

[3] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino, “The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury,” Am. J. Phys. Med. Rehabil., vol. 91, no. 11, pp. 911–921, 2012.

[4] M. Sharifi, S. Tripathi, Y. Chen, Q. Zhang, and M. Tavakoli, “Reinforcement learning methods for assistive and rehabilitation robotic systems: A survey,” IEEE Trans. Syst., Man, Cybern., Syst., early access, 2025.

[5] A. I. Károly, P. Galambos, J. Kuti, and I. J. Rudas, “Deep learning in robotics: Survey on model structures and training strategies,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 51, no. 1, pp. 266–279, Jan. 2021.

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

[7] B. Luo, Z. Wu, F. Zhou, and B. C. Wang, “Human-in-the-loop reinforcement learning in continuous-action space,” IEEE Trans. Neural Netw. Learn. Syst., vol. 35, no. 11, pp. 15735–15744, Nov. 2023.

[8] J. I. Han, J. H. Lee, H. S. Choi, J. H. Kim, and J. Choi, “Policy design for an ankle-foot orthosis using simulated physical human–robot interaction via deep reinforcement learning,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 30, pp. 2186–2197, Nov. 2022.

[9] M. Wang and S. Luo, “AI-computing, deep reinforcement learning-based predictive human-robot neuromechanical simulation for wearable robots,” Appl. Intell., vol. 55, no. 6, pp. 1–19, Jun. 2025.

[10] K. G. Gerritsen, A. J. van den Bogert, M. Hulliger, and R. F. Zernicke, “Intrinsic muscle properties facilitate locomotor control—a computer simulation study,” Mot. Control, vol. 2, no. 3, pp. 206–220, 1998.

[11] C. T. John, F. C. Anderson, J. S. Higginson, and S. L. Delp, “Stabilisation of walking by intrinsic muscle properties revealed in a three-dimensional muscle-driven simulation,” Comput. Methods Biomech. Biomed. Eng., vol. 16, no. 4, pp. 451–462, Apr. 2013.

[12] P. Schumacher, T. Geijtenbeek, V. Caggiano, V. Kumar, S. Schmitt, G. Martius, and D. F. Haeufle, “Natural and robust walking using reinforcement learning without demonstrations in high-dimensional musculoskeletal models,” arXiv preprint arXiv:2309.02976, 2023.

[13] N. Ratnakumar, K. Akbaş, R. Jones, Z. You, and X. Zhou, “Predicting sit-to-stand motions with a deep reinforcement learning based controller under idealized exoskeleton assistance,” Multibody Syst. Dyn., pp. 1–18, 2024.

[14] O. Kayan and H. Yalcin, “Learning to walk on a human musculoskeletal model wearing a knee orthosis via deep reinforcement learning,” in Proc. 5th Int. Congr. Human-Comput. Interact., Optim. Robot. Appl. (HORA), Jun. 2023, pp. 1–4.

[15] Q. Zhang, V. Nalam, X. Tu, M. Li, J. Si, M. D. Lewek, and H. H. Huang, “Imposing healthy hip motion pattern and range by exoskeleton control for individualized assistance,” IEEE Robot. Autom. Lett., vol. 7, no. 4, pp. 11126–11133, Oct. 2022.

[16] 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 Trans. Syst., Man, Cybern., Syst., early access, 2024.

[17] G. Bingjing, H. Jianhai, L. Xiangpan, and Y. Lin, “Human–robot interactive control based on reinforcement learning for gait rehabilitation training robot,” Int. J. Adv. Robot. Syst., vol. 16, no. 2, pp. 1–13, Mar. 2019.

[18] T. Geijtenbeek, "The Hyfydy simulation software," 2021. [Online]. Available: https://hyfydy.com

[19] H. Geyer and H. Herr, "A muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 18, no. 3, pp. 263–273, Jun. 2010.

[20] C. Igel, T. Suttorp, and N. Hansen, "A computational efficient covariance matrix update and a (1+1)-CMA for evolution strategies," in Proc. 8th Annu. Conf. Genetic Evol. Comput., Jul. 2006, pp. 453–460.

[21] T. Geijtenbeek, "Scone: Open source software for predictive simulation of biological motion," J. Open Source Softw., vol. 4, no. 38, p. 1421, 2019.

[22] C. F. Ong, T. Geijtenbeek, J. L. Hicks, and S. L. Delp, "Predicting gait adaptations due to ankle plantarflexor muscle weakness and contracture using physics-based musculoskeletal simulations," PLoS Comput. Biol., vol. 15, no. 10, p. e1006993, Oct. 2019.

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

[24] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," Sci. Robot., vol. 5, no. 47, p. eabc5986, Oct. 2020.

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

[26] B. Su and E. M. Gutierrez-Farewik, "Gait trajectory and gait phase prediction based on an LSTM network," Sensors, vol. 20, no. 24, p. 7127, Dec. 2020.

[27] B. Ren, Z. Zhang, C. Zhang, and S. Chen, "Motion trajectories prediction of lower limb exoskeleton based on long short-term memory (LSTM) networks," Actuators, vol. 11, no. 3, p. 73, Feb. 2022.

[28] M. Luo, X. Dong, H. Yu, M. Zhang, X. Wu, W. Kobsiriphat, J.-X. Wang, and W. Cao, "Lateral walking gait phase recognition for hip exoskeleton by denoising autoencoder-LSTM," Comput. Struct. Biotechnol. J., vol. 28, pp. 50–62, 2025.

[29] E. R. Rush, C. Heckman, K. Jayaram, and J. S. Humbert, "Neural dynamics of robust legged robots," Front. Robot. AI, vol. 11, p. 1324404, 2024.

[30] I. Radosavovic, T. Xiao, B. Zhang, T. Darrell, J. Malik, and K. Sreenath, "Real-world humanoid locomotion with reinforcement learning," Sci. Robot., vol. 9, no. 89, p. eadi9579, 2024.

[31] O. M. Blake and J. M. Wakeling, "Estimating changes in metabolic power from EMG," SpringerPlus, vol. 2, no. 1, p. 229, 2013.

[32] R. E. Bruns, P. Vos, and R. D. Wedge, "Electromyography as a surrogate for estimating metabolic energy expenditure during locomotion," Med. Eng. Phys., vol. 109, p. 103899, Apr. 2022.

[33] H. Shin, S. Kim, S. Kim, J. Lee, J. Kim, and K. Kim, "Multi-modal locomotion mode recognition in the real world for robotic hip complex exoskeletons," IEEE Robot. Autom. Lett., early access, 2025.

[34] 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 Trans. Robot., vol. 35, no. 4, pp. 1055–1062, Aug. 2019.

[35] J. M. Brockway, "Derivation of formulae used to calculate energy expenditure in man," Hum. Nutr. Clin. Nutr., vol. 41, no. 6, pp. 463–471, Nov. 1987.

[36] K. Seo, J. Lee, Y. Lee, T. Ha, and Y. Shim, "Fully autonomous hip exoskeleton saves metabolic cost of walking," in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2016, pp. 4628–4635.

[37] J. M. Wang, S. R. Hamner, S. L. Delp, and V. Koltun, "Optimizing locomotion controllers using biologically-based actuators and objectives," ACM Trans. Graph., vol. 31, no. 4, pp. 1–11, Jul. 2012.

[38] T. A. Wren, K. P. Do, S. A. Rethlefsen, and B. Healy, "Cross-correlation as a method for comparing dynamic electromyography signals during gait," J. Biomech., vol. 39, no. 14, pp. 2714–2718, Oct. 2006.

[39] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Sep. 2017, pp. 23–30.

[40] M. Grimmer, "Powered lower limb prostheses," Ph.D. dissertation, Technische Universität Darmstadt, Darmstadt, Germany, 2015.