Oghogho Jr, Martin Princeton; Sharifi, Mojtaba; Vukadin, Mia; Chin, Connor; Mushahwar, Vivian K.; Tavakoli, Mahdi. Deep Reinforcement Learning for EMG-Based Control of Assistance Level in Upper-Limb Exoskeletons, International Symposium on Medical Robotics, Atlanta, USA, April 2022.

Deep Reinforcement Learning for EMG-based Control of Assistance Level in Upper-limb Exoskeletons

Martin Oghogho Jr, Mojtaba Sharifi, *Member, IEEE,* Mia Vukadin, Connor Chin, Vivian K. Mushahwar, *Member, IEEE,* Mahdi Tavakoli, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a deep reinforcement learning (DRL) method to control the assistance level of an upper-limb exoskeleton in real-time based on the electromyographic (EMG) activity of human muscles in 3D point-to-point reaching movements. The proposed autonomous assistive device would enhance the force exertion capability of individuals by resolving major challenges such as identifying scaling factors for personalized amplification of their effort and not requiring lengthy offline training/adjustment periods to perform their manual tasks comfortably. To this end, we employed the Twin Delayed Deep Deterministic Policy Gradient (TD3) method for rapid learning of the appropriate controller's gain values and delivering personalized assistive torques by the exoskeleton to different joints to assist the wearer in a weight handling task. A nonlinear reward function is defined in terms of the EMG activity level and the position deviation from the destination point to simultaneously minimize the muscle effort and maximize the positioning accuracy. This facilitates autonomous and individualized physical assistance by rapid exploration of reward values and adopting various action gains within a safe range to exploit the ones that maximize the reward. Based on experimental studies on an exoskeleton with soft actuators that we have developed, the proposed DRL method is able to identify the most appropriate assistive gain for each joint of the exoskeleton in real-time for the user with a fast rate of convergence (during the first two minutes). Optimum assistive gains are identified for each degree of freedom (DOF) in a 4 kg weight handling task in 3D space, which required less than 15% of the muscle contraction level (EMG activity).**

***Index Terms*—Deep reinforcement learning (DRL); twin delayed deep deterministic policy gradient (TD3); actor-critic method; assistive exoskeleton; EMG-based control**

This work was supported by the Natural Sciences and Engineering Research Council (NSERC), Canadian Institutes of Health Research (CIHR), and Canada Foundation for Innovation (CFI). *(Corresponding author: Mojtaba Sharifi)*

Martin Oghogho Jr, Mojtaba Sharifi, Mia Vukadin, Connor Chin, and Mahdi Tavakoli are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta T6G 1H9, Canada. (e-mail: martinpr@ualberta.ca, m.Sharifi@ualberta.ca, vukadin@ualberta.ca, cchin@ualberta.ca, mahdi.tavakoli@ualberta.ca)

Mojtaba Sharifi and Vivian K. Mushahwar are with the Department of Medicine, Division of Physical Medicine and Rehabilitation, University of Alberta, Edmonton, Alberta T6G 2E1, Canada. (e-mail: Vivian.Mushahwar@ualberta.ca)

Mojtaba Sharifi, Vivian K. Mushahwar, and Mahdi Tavakoli are with the Sensory Motor Adaptive Rehabilitation Technology (SMART) Network, University of Alberta, Edmonton, Alberta T6G 2E1, Canada.

Mojtaba Sharifi is with the Department of Mechanical Engineering, San Jose State University, San Jose, California 95192-0087.

I. Introduction

Workers doing manual tasks in different industries (construction, manufacturing, warehousing) experience 340 million occupational accidents annually worldwide [1]. Assistive robotic systems such as exoskeletons have been developed to help the above-mentioned individuals to prevent occupational injuries and musculoskeletal damage [2], [3]. Extensive research studies have been conducted regarding the control design of upper-limb exoskeletons using different classical and advanced strategies with pre-specified structure, gains, and parameters. However, it is still needed to enhance the autonomy of these systems while preserving the safety and comfort of the wearer for personalizing the exoskeleton behavior.

Human-robot interaction (HRI) is a research field that enables humans and robots to collaborate and achieve shared goals in applications as diverse as robot-assisted surgery, assistive and rehabilitation therapies, and manufacturing operations [4], [5]. One of the major difficulties in facilitating these collaborations is estimating the intention of the human operator during each specific task. In this regard, various approaches have been devised and assessed to resolve this challenge involving head pose and gesture detection [6], speech recognition [7], kinematic data (e.g., velocity) [8], force/torque measurement [9], and biological signals [10]. To control the exoskeletons, two kinds of sensory information have been employed to estimate the human intention, namely the HRI force/torque data and electromyography (EMG) signals of the human muscles. However, the force/torque transducers mounted onto the exoskeleton structure failed to isolate the active portion of the human effort from its passive component [11], [12], [13], [14]. On the other hand, the EMG signals generated by the muscle contractions can be measured as the index of the active human force [15].

Recently, classical control methods have been implemented using EMG signals, such as model predictive control, impedance control, and adaptive control for various HRI applications such as assistive exoskeletons [16], [17], where EMG signals were used to estimate the muscle force and motion intention. Nevertheless, the obtained estimations suffered from muscle fatigue, calibration inaccuracy, variation of muscle-skin conductivity, and electrode positioning [18],

[19]. Accordingly, artificial intelligence (AI), and specifically machine learning (ML), were adopted to play an important role in controlling the exoskeletons based on EMG signals. In most of these studies, artificial neural networks (ANN) were utilized to facilitate more accurate torque estimation compared to model-based strategies [20], [19]. A three-layer Back Propagation Neural Network (BPNN) was adopted in [20] to extract the filtered EMG features of the arm muscles and identify their relationship to the elbow joint angle. The aforementioned ANN provided appropriate estimations in learning the user intention but required considerable time for training in offline mode and neglected the muscle effort and time-varying human behavior during the task [21]. As a result, personalized and online forms of learning such as DRL algorithms can be integrated into the control system to resolve these issues [20], [21], [22].

DRL algorithms have been implemented on robotic systems to enhance their autonomy in performing different tasks without the necessity of mathematical modeling for the robot and its environment [23]. Further improvements and modifications in reinforcement learning algorithms led to more advanced methods such as Vanilla Policy Gradient (VPG) [24], Deep Q-Network (DQN) [25], Deterministic Policy Gradient (DPG), Trust Region Policy Optimization (TRPO) [26], Proximal Policy Optimization (PPO) [27]. On-policy methods such as PPO and TRPO tend to show slower learning rates than off-policy methods (e.g., DPG, DQN). Amongst commonly used off-policy DRL methods in robotic applications, Deep Deterministic Policy Gradient (DDPG) addressed issues of high variance faced in VPG and showed faster learning performance compared to VPG and TRPO in reaching and pick & place tasks [28], [29]. Despite DDPG's extensive usage in robotic applications, there were still areas for further improvements to avoid overestimation in predicting the reward values and to address low stability in learning convergence because of adopting sub-optimal policies [30]. As a result, the Twin-Delayed DDPG (TD3) was developed to reduce the growth of convergence errors due to the reward overestimation, improving the stability conditions, and regularizing the action noise [30], [31], [32], [33], [34].

In this study, the Twin-Delayed DDPG (TD3) algorithm is employed for the first time to intelligently control an upper-limb exoskeleton based on the EMG activity of the wearer's muscles. The proposed strategy aims to identify the user's intention, and supply the necessary assistance in response to their muscle effort in real-time in order to comfortably conduct their desired task. Online training of the TD3 algorithm results in a rapid and stable performance of learning for the appropriate gains in an assisitve control scheme that delivers personalized exoskeleton torques via analyzing EMG signals and motion trajectories of the HRI system. To this end, a nonlinear reward function is introduced in terms of the EMG activity level and the trajectory deviation for point-to-point reaching tasks. This reward shaping facilitates the minimization of muscle effort while ensuring high movement accuracy for the human user, without requiring prior information about the human-exoskeleton modeling for gain scheduling.

II. TD3 Structure in Deep Reinforcement Learning

DRL is a subset of ML that takes optimal actions learned by a continuously improving agent on an environment through maximizing the accumulated reward. The agent's behavior can be policy-based (actor) whereby given a state, an action is determined, $\pi : S \rightarrow A$. Alternatively, its behavior can be value-based (critic) whereby it estimates the reward value from possible actions in each state, and its policy is then derived from this estimation. The value or quality of each set of states and actions is defined by a Q function, $Q : S \times A \rightarrow R$. In order to optimize the agent's behavior, one can take advantage of both methods (value-based and policy-based) by combining them to create an actor-critic strategy. The actor network receives the state or observation from the environment and outputs the best action to be implemented. However, the critic network evaluates both the action and the observation as the inputs and estimates the reward value as a result of each action on the environment, as shown in Fig. 1. The difference between this estimated reward value and the actual reward value from the environment is called the Temporal Difference (TD) error. This error is fed back to the actor and critic through a backpropagation process to improve the actor's next decision as well as the critic's next reward estimation.

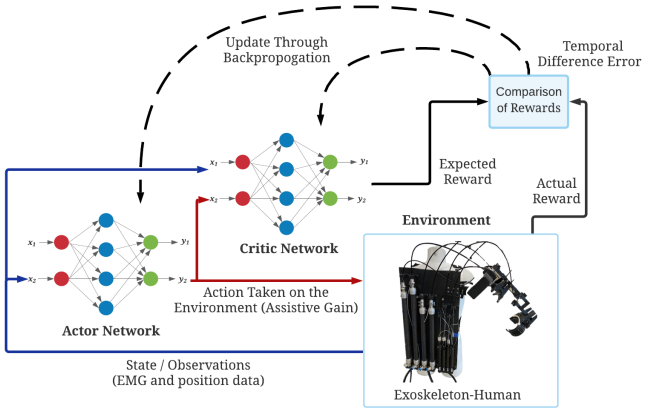

Fig. 1. Actor-critic RL architecture: the policy structure is considered to be the actor, the estimated value function is represented as the critic, and the environment in which the actor-critic network acts on

The underlying principle of policy gradients is to improve the actions' probabilities that lead to higher returns and reduce those that lead to lower ones until an optimal policy is achieved. DDPG which is a commonly used DRL method has its limitations such as unstable convergence, overreliance on hyperparameters for each task, and overestimation of the Q-values. Accordingly, Twin Delayed DDPG (TD3) was introduced to make major contributions to the deep deterministic

policy gradient method by resolving the above-mentioned issues as described below [30].

III. Implementation of TD3 for EMG-based Control of Exoskeleton

The objective of the TD3 algorithm in the application of exoskeleton control is to observe the motion response and the human EMG activity (s_t) and take appropriate actions (a_t) in order to achieve higher reward values (r_t). Therefore, we want to maximize the total rewards from a given set of actions on the exoskeleton. As described above, TD3 is an actor-critic strategy and the actor policy maps the states from the elbow and shoulder joints together with their muscle activities to specific assistive gains for the actuators, using a deterministic policy. The double critic networks, $Q(s,a)$ which are functions of the states and actions learn how to approximate the reward from each action using the Bellman's equation [30] in terms of the discounted sum of expected future TD errors:

$$\begin{aligned} Q_\theta(s,a) &= r_t + \gamma\,\mathbb{E}\left[Q_\theta(s_{t+1},a_{t+1}) - \delta_t\right] \\ &= r_t + \gamma\,\mathbb{E}\left[r_{t+1} + \gamma\,\mathbb{E}\left[Q_\theta(s_{t+2},a_{t+2})\right]\right] - \delta_t \\ &= \mathbb{E}_{s_i\sim p_\pi, a_i\sim\pi}\left[\sum_{i=t}^{T}\gamma^{i-t}(r_i-\delta_i)\right] \end{aligned} \tag{1}$$

where $\left[\sum_{i=t}^{T}\gamma^{i-t}(r_i-\delta_i)\right]$ describes the discounted sum of returns, γ is the discounted factor, $Q_\theta(s,a)$ is the differentiable function approximator used to estimate the value function with the parameter θ, and $\mathbb{E}$ is the expectations from distributions of both the states and actions following the policy π_θ. In the training process, the actor and two critic networks are first initialized with random parameters (ϕ, θ_1, θ_2). Training a policy using actor and critic networks can result in the divergence of the network behavior and causing instability. Therefore, target networks (with ϕ', θ_1', θ_2') are initialized as actor and critic networks copies to improve learning stability and reduce divergence when updating the target values. A replay buffer is also initialized to record the agent's experiences as $e_t(s_t,a_t,r_t,s_{t+1})$, which are later sampled randomly for learning. An action is then taken using the Markov decision process (MDP) [35] and an exploration noise is added to this action to explore different areas of possible positive rewards modeled using the following equation.

$$a \sim \pi(s) + \epsilon\,,\;\; \epsilon \sim \mathcal{N}(0,\sigma) \tag{2}$$

where a is the action, π is the employed policy, and ϵ is the exploration noise. Based on the action taken, the observations (consisting of the joint angles and EMG signals) are obtained in real-time at each time step. Also, the critic network predicts the expected return by taking the minimum of the two value functions. Then, the information is stored in a replay buffer (s,a,r,s'). Once a time step has been carried out the system is trained for several iterations by sampling a mini-batch of stored transitions from the replay buffer. Then, an action is taken with added noise based on the transitions and a target policy smoothing is applied. This is performed by clipping the noisy action to prevent it from being too far from its original value:

$$\tilde{a} \leftarrow \pi_{\phi'}(s) + \epsilon,\;\; \epsilon \sim clip\left(\mathcal{N}(0,\tilde{\sigma}) - c, c\right) \tag{3}$$

Then, the target Q values from the double critic networks are computed employing the smallest value of the two networks. The loss function is calculated for the two critic networks by computing the mean squared error (MSE) between each critic and target Q value. The critic is then optimized using backpropagation.

$$\begin{aligned} &y \leftarrow r + \gamma\, min_{i=1,2}\, Q_{\theta_i'}(s',\tilde{a}), \\ &\theta_i \leftarrow\; min_{\theta_i} N^{-1}\sum\left(y - Q_{\theta_i}(s,a)\right)^2 \end{aligned} \tag{4}$$

where r is the reward and y describes the fixed objective obtained as a result of the updated target networks. The actor policy is also updated and its loss function is computed by obtaining the mean of the Q values from the critic networks. And the actor network is optimized using backpropagation as in the critic network:

$$\nabla_\phi J(\phi) = N^{-1}\sum \nabla_a Q_{\theta_1}(s,a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s) \tag{5}$$

in which $\nabla_\phi J(\phi)$ is the gradient of the measured performance $J(\phi)$ of the target policy π_ϕ Finally, the target networks are updated alongside the delayed actor network using soft (gradual) update:

$$\begin{aligned} \theta_i' &\leftarrow \tau\theta_i + (1-\tau)\theta_i' \\ \phi' &\leftarrow \tau\phi + (1-\tau)\phi' \end{aligned} \tag{6}$$

where τ is the soft update coefficient that is selected to provide stable values in the value network before it updates the policy network. Note that fast rates of this update can lead to a divergent behavior that should be avoided.

A. Reward Shaping for Assistive Control

A reward function is introduced to identify the best value of assistive gain for each joint (DOF) of the exoskeleton. This gain generates the assistive torques delivered to the human limb that is proportional to the normalized EMG activities of the corresponding muscles:

$$T_{a_i} = K_{a_i}\, EMG_{n_i} \tag{7}$$

Here, T_{a_i} is the delivered assistive torque, K_{a_i} is the learned gain for joint i, and EMG_{n_i} is the normalized EMG activity of the corresponding muscle. In this regard, a reward function is defined such that the TD3 algorithm learns to adjust the assistive gain so as to reduce the user's effort while smoothing the response motion trajectory. To this end, a penalty is considered for the obtained gains or actions that result in large overshoot in joint positions, in addition to the ones that necessitate higher EMG magnitudes. Accordingly, the wearer's muscle effort and the motion response's overshoot are minimized simultaneously by learning the best actions (assistive gains). The TD3 algorithm receives the

EMG signals and the maximum joint positions by which the agent computes the best action for the next trial. Therefore, the reward function is shaped to realize the above-mentioned trade-off between the muscle effort and trajectory convergence as follows:

$$\mathcal{R} = -\sum \left(\mu \, |e_i|^2 + \lambda \, EMG_{n_i}^4 + \mathcal{R}_p \right) \tag{8}$$

where μ and λ are the scaling coefficients. e is the overshoot of the response trajectory for joint i, EMG_{n_i} is the normalized EMG value of the corresponding muscle and $\mathcal{R}_p$ is a threshold penalty.

To make the learning process more useful and safe, high and low thresholds are set for the actions taken by the agent such that the implemented assistive gains stay within an appropriate range. If a suggested gain was outside of the threshold range, a high penalty would be applied to the reward function. These limits were determined based on the usability of the exoskeleton and the safety of interaction with the human.

$$\mathcal{R}_p = \begin{cases} \mathcal{P} & K_{a_i} \leq K_{a_{min}} \\ 0 & K_{a_{min}} \leq K_{a_i} \leq K_{a_{max}} \\ \mathcal{P} & K_{a_i} \geq K_{a_{max}} \end{cases} \tag{9}$$

where K_{a_i} is the action taken by the agent and implemented on the exoskeleton and $\mathcal{P}$ is the penalty value that affects the reward (8). $K_{a_{min}}$ and $K_{a_{max}}$ are the minimum and maximum thresholds of the assistive gains for this application, which are specified for each exoskeleton and wearer based on their mechanical characteristics. Gain values lower than $K_{a_{min}}$ would necessitate large EMG activities to generate adequate assistive torques and gain values higher than $K_{a_{max}}$ would result in sudden motions with considerable overshoots even with small EMG signals.

In this implementation, the assistive gain acts as the agent's action in TD3 learning algorithm, while the states are represented by EMG activity of the muscle and overshoot of the position. These states are optimized through the use of the reward function (8).

IV. Experimental Studies

The proposed learning-based assistive control strategy using TD3 algorithm was tested experimentally on an upper-limb exoskeleton with soft artificial muscle actuators (Fig. 2). The experiments were designed for a point-to-point weight handling task in 3D space, performed by the dominant arm of an able-bodied male participant (age 33). The wearer performed this task with different gain values suggested by the TD3 algorithm and different trials are assessed to explore action and reward values. The actor and critic networks were trained to identify the best actions (assistive gain K_a for each muscle and joint) result in maximization of the reward score.

A. Hardware Implementation

The proposed DRL method was implemented using Python (Python Software Foundation, USA) to generate assistive gains and the EMG-based controller was performed in real-time with 1 msec sampling rate having a C++ code uploaded on an ESP32 microcontroller (Espressif Systems, China). The connection between Python and the microcontroller was facilitated by a TCP/IP communication protocol employing the openAI's Gym environment. Accordingly, the main Python script for learning was in communication with the Gym environment creating a socket to finally transfer data to the C++ framework. As seen in Fig. 2, the exoskeleton's joints were actuated by fluidic muscles DMSP-20-RM-CM (Festo Corporate, Esslingen, Germany), and Omega electro-pneumatic transducers EP211-X120-10V (Omega Engineering Inc., USA) were employed to regulate the pressure of these pneumatic soft actuators. For measuring the exoskeleton position, quadrature optical encoders (HEDM-5500 B12, Broadcom Inc., US) were attached to the shoulder and elbow joints. Three SX230-1000 (Biometrics Ltd, United Kingdom) surface EMG sensors were placed along the medial deltoid, anterior deltoid, and biceps brachii muscles (Fig. 2). The EMG signals were first rectified and then passed through a second-order Butterworth low-pass filter with a cutoff frequency of 8 Hz for smoothing the EMG signals in addition to the sensor's built-in 460 Hz low-pass filter.

B. Experimental Trials

The human-exoskeleton interaction is shown in Fig. 2, where the user performed a weight handling task in 3D space by making shoulder abduction-adduction (AA), shoulder flexion-extension (FE), and elbow flexion-extension (FE). For each test, the user was asked to grab a 4 kg weight and move his arm from the resting configuration to a final target level for placement of the weight. This target level was chosen to be close to the middle point of the range of joints' motion to demonstrate the positioning performance of the exoskeleton's controller, which corresponds to the indicated markers on the tables for weight handling. The wearer would receive an assistive torque by the exoskeleton proportional to the EMG activity of the medial deltoid, anterior deltoid, and long head of the biceps brachii. This proportional gain was adjusted through DRL for each trial of this task. By exploring the response of this HRI having various gain values (actions), the maximum reward score was exploited to facilitate minimum overshoot (deviation) from the final destination point and optimize the muscle effort (level of EMG). Final overshoot from the target position was measured using the positional data from the quadrature encoders. The employed parameters to train the TD3 networks for learning appropriate assistive gains for each human user are obtained from [30]. Each episode of learning consisted of 5 trials where each trial took an average of 4.9 seconds.

As seen in Fig. 3, the trained TD3 networks were able to identify the optimum value of the assistive gain K_a for each

(a) (b)

Fig. 2. Upper-limb exoskeleton with soft actuators worn by a human user for implementation of the proposed DRL method for learning personalized assitive gains: (a) front view and (b) back view

joint of the exoskeleton after 26 trials (within 2 minutes). Accordingly, these optimum gains were learned as 55.1 ± 5.5 Nm, 67.6 ± 8.4 Nm, and 54.2 ± 7.3 Nm for the elbow FE, shoulder AA, and shoulder FE motions to assist the biceps brachii, medial deltoid, and anterior deltoid muscles, respectively, based on their EMG activities. This learning performance was achieved by exploring assistive gains from 25 to 130 Nm to investigate the most appropriate HRI behavior based on the reward definition (8). As demonstrated in Figs. 4, 5 and 6, the optimum assistive gains for three DOFs of the exoskeleton produced a balance between the muscle activities and the accuracy of the motion planning.

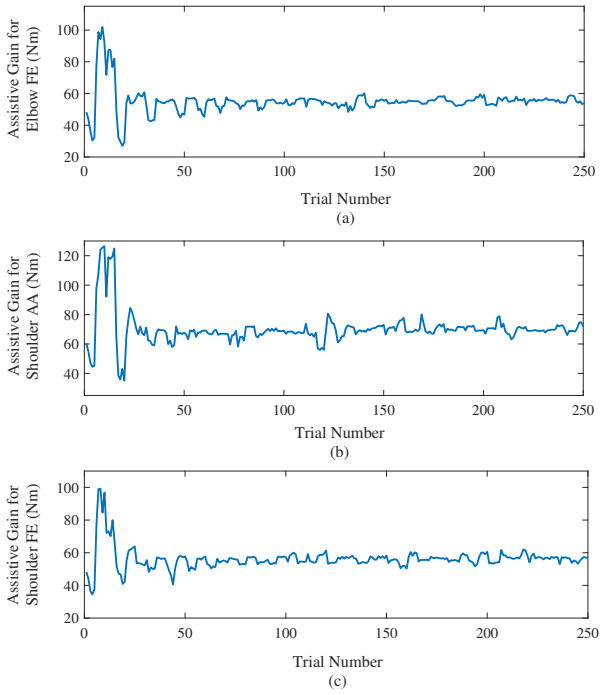

Fig. 3. Learning of the assitive gain K_a for (a) elbow FE, (b) shoulder AA, and (c) shoulder FE motions

The learned assistive gain of $K_a = 55.1$ Nm for the shoulder FE resulted in 35% reduction in EMG activity for the anterior deltoid muscle compared to the one with $K_a = 38$ Nm (Fig. 4a) and a smoother motion trajectory with negligible overshoot in comparison with 14% in the case of $K_a = 98$ Nm (Fig. 4b). Therefore, the obtained gain magnitude from DRL provided a trade-off between the muscle effort and motion accuracy in a point-to-point reaching task. Similarly, the overshoot magnitude in the shoulder AA motion decreased by 48% via lowering the assistive gain from $K_a = 125$ Nm to the learned value of $K_a = 67.6$ Nm (Fig. 5a), while the EMG activity of the medial deltoid muscle increased only by 11% (Fig. 5b). Also for the elbow FE motion, the learned gain of $K_a = 55.1$ Nm resulted in 45.6% less EMG activity for the biceps brachii muscle compared to the case of $K_a = 24$ Nm and 42.7% less overshoot in the motion trajectory than the one obtained for $K_a = 102$ Nm, as illustrated in Fig. 6. Therefore, the level of assistance delivered to the user was optimized using the TD3 algorithm based on the muscle effort (EMG) and movement data for different DOFs of the exoskeleton in this point-to-point weight handling task. The average values of the normalized EMG activity for the anterior deltoid, medial deltoid, and biceps brachii muscles using the learned K_a values were identified as 0.141, 0.083, and 0.127 (less than 15%) in Figs. 4, 5 and 6, respectively. The different K_a values were then tested during a more complicated two-point pick and place task.The trajectory of the elbow FE joint showed that the learned K_a value gave the user the highest placement accuracy during the task (Fig. 7).

V. Conclusions and Implications

In this paper, a new DRL-based control method was developed and tested to deliver physical assistance intelligently based on EMG signals using upper-limb exoskeletons. For this purpose, Twin Delayed Deep Deterministic Policy Gradient (TD3) was utilized for fast learning of the appropriate

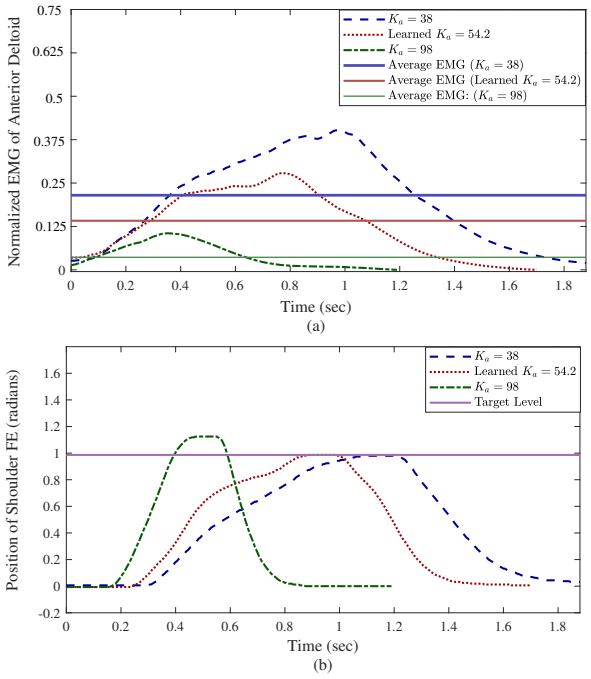

Fig. 4. (a) Normalized EMG activity of anterior deltoid muscle, and (b) the position response for the corresponding shoulder FE joint with low, learned and high assistive gains ($K_a = 38$, $K_a = 54.2$, and $K_a = 98$ Nm)

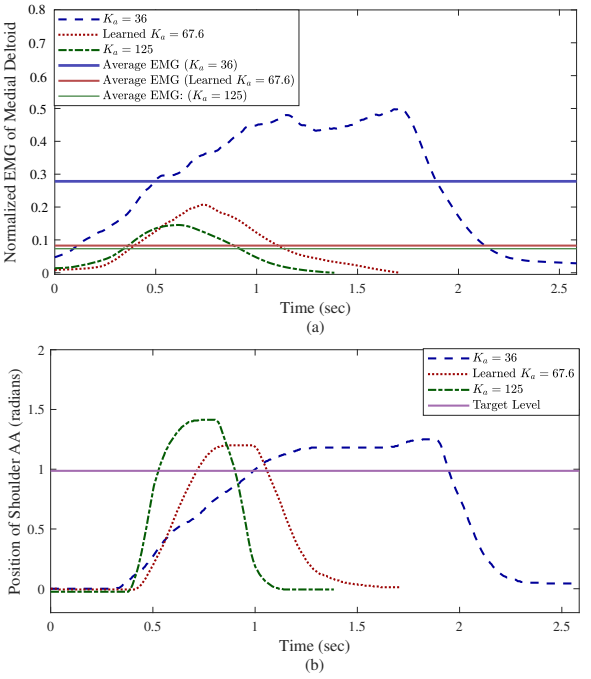

Fig. 5. (a) Normalized EMG activity of medial deltoid muscle, and (b) the position response for the corresponding shoulder AA joint with low, learned and high assistive gains ($K_a = 36$, $K_a = 67.6$, and $K_a = 125$ Nm)

controller's gains and providing personalized torques to assist the wearer in point-to-point movements. To balance the muscle effort and reaching accuracy, a reward function was defined in terms of the EMG activities and the maximum overshoot of the position. The proposed autonomous system was able to learn the optimum action gains without prior

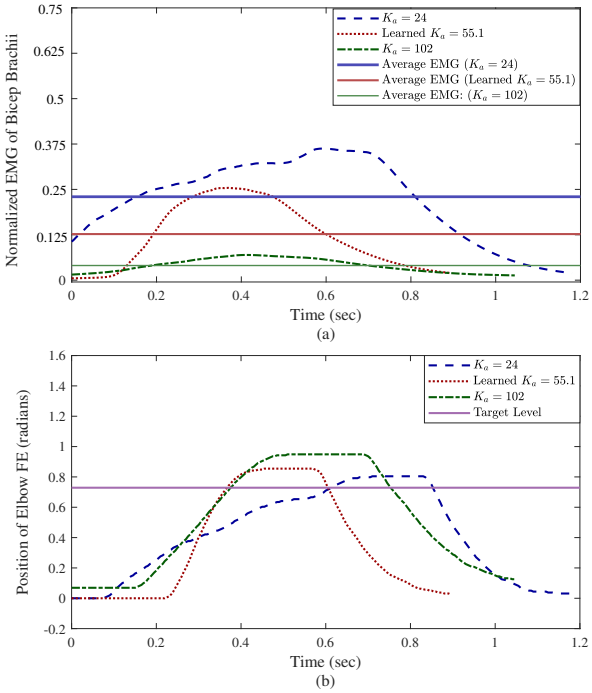

Fig. 6. (a) Normalized EMG activity of biceps brachii muscle, and (b) the position response for the corresponding elbow FE joint with low, learned and high assistive gains ($K_a = 24$, $K_a = 55.1$, and $K_a = 102$ Nm)

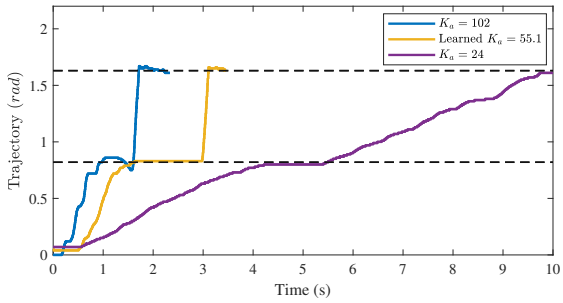

Fig. 7. Elbow FE trajectory for a two-point pick and place task

information about the passive human-exoskeleton dynamics and the active muscular capability of the wearer. In experimental evaluations, the TD3 agent was able to identify the optimum assistive gains in under 2 minutes of online learning, and resulted in performing a 4 kg weight handling task with less than 15% of the muscle contractions. The empirical results showed that having this intelligent strategy in EMG-based control of the exoskeleton would optimize the muscle effort and enhance the motion response of the HRI system. In future studies, further improvements in the DRL algorithm will be beneficial to minimize the number of trials and accelerate learning for more challenging and complicated tasks. This autonomous controller can be employed for learning appropriate assistive torques for individuals with various physical capabilities and neurological conditions in different tasks.

REFERENCES

[1] "The enormous burden of poor working conditions," Jul 2020. [Online]. Available: https://www.ilo.org/moscow/areas-of-work/occupational-safety-and-health/WCMS_249278/lang--en/index.htm

[2] M. Lazzaroni, S. Toxiri, D. G. Caldwell, S. Anastasi, L. Monica, E. D. Momi, and J. Ortiz, "Acceleration-based assistive strategy to control a back-support exoskeleton for load handling: Preliminary evaluation," in *IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, 2019, pp. 625–630.

[3] R. Ramon, C. Nataros, T. Yi, L. Lagos, A. Avarelli, and O. Bai, "Hotcell worker assistive robotic exoskeleton design and control," in *IEEE International Symposium on Measurement and Control in Robotics (ISMCR)*, 2019, pp. A1–3–1–A1–3–4.

[4] M. Sharifi, S. Behzadipour, H. Salarieh, and M. Tavakoli, "Assist-as-needed policy for movement therapy using telerobotics-mediated therapist supervision," *Control Engineering Practice*, vol. 101, p. 104481, 2020.

[5] C. Rossa, M. Najafi, M. Tavakoli, and K. Adams, "Robotic rehabilitation and assistance for individuals with movement disorders based on a kinematic model of the upper limb," *IEEE Transactions on Medical Robotics and Bionics*, vol. 3, no. 1, pp. 190–203, 2021.

[6] R. Stiefelhagen, C. Fugen, R. Gieselmann, H. Holzapfel, K. Nickel, and A. Waibel, "Natural human-robot interaction using speech, head pose and gestures," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, 2004, pp. 2422–2427.

[7] J. Kennedy, S. Lemaignan, C. Montassier, P. Lavalade, B. Irfan, F. Papadopoulos, E. Senft, and T. Belpaeme, "Child speech recognition in human-robot interaction: evaluations and recommendations," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, 2017, pp. 82–90.

[8] J. Lanini, H. Razavi, J. Urain, and A. Ijspeert, "Human intention detection as a multiclass classification problem: Application in physical human–robot interaction while walking," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4171–4178, 2018.

[9] M. Sharifi, "Impedance control of non-linear multi-dof teleoperation systems with time delay: absolute stability," *IET Control Theory & Applications*, vol. 12, pp. 1722–1729(7), 2018.

[10] N. P. Reddy and V. Gupta, "Toward direct biocontrol using surface emg signals: Control of finger and wrist joint models," *Medical engineering & physics*, vol. 29, no. 3, pp. 398–403, 2007.

[11] X. Liu and Q. Wang, "Real-time locomotion mode recognition and assistive torque control for unilateral knee exoskeleton on different terrains," *IEEE/ASME Transactions on Mechatronics*, 2020.

[12] M. Sharifi, J. K. Mehr, V. K. Mushahwar, and M. Tavakoli, "Adaptive cpg-based gait planning with learning-based torque estimation and control for exoskeletons," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8261–8268, 2021.

[13] J. K. Mehr, M. Sharifi, V. K. Mushahwar, and M. Tavakoli, "Intelligent locomotion planning with enhanced postural stability for lower-limb exoskeletons," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7588–7595, 2021.

[14] M. Sharifi, J. K. Mehr, V. K. Mushahwar, and M. Tavakoli, "Autonomous locomotion trajectory shaping and nonlinear control for lower-limb exoskeletons," *IEEE/ASME Transactions on Mechatronics*, Published online, 2022.

[15] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, "Techniques of emg signal analysis: detection, processing, classification and applications," *Biological procedures online*, vol. 8, no. 1, pp. 11–35, 2006.

[16] Q. Wei, Z. Li, K. Zhao, Y. Kang, and C.-Y. Su, "Synergy-based control of assistive lower-limb exoskeletons by skill transfer," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 2, pp. 705–715, 2020.

[17] T. Teramae, T. Noda, and J. Morimoto, "Emg-based model predictive control for physical human–robot interaction: Application for assist-as-needed control," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 210–217, 2017.

[18] S. Qiu, W. Guo, D. Caldwell, and F. Chen, "Exoskeleton online learning and estimation of human walking intention based on dynamical movement primitives," *IEEE Transactions on Cognitive and Developmental Systems*, 2020.

[19] K. Gui, H. Liu, and D. Zhang, "A practical and adaptive method to achieve emg-based torque estimation for a robotic exoskeleton," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 2, pp. 483–494, 2019.

[20] G. C. Luh, J. J. Cai, and Y. S. Lee, "Estimation of elbow motion intension under varing weight in lifting movement using an emg-angle neural network model," in *International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 2, 2017, pp. 640–645.

[21] Q. Wu, B. Chen, and H. Wu, "Neural-network-enhanced torque estimation control of a soft wearable exoskeleton for elbow assistance," *Mechatronics*, vol. 63, p. 102279, 2019.

[22] N. Sacchi, G. P. Incremona, and A. Ferrara, "Deep reinforcement learning of robotic prosthesis for gait symmetry in trans-femoral amputated patients," in *29th Mediterranean Conference on Control and Automation (MED)*, 2021, pp. 723–728.

[23] Y. P. Pane, S. P. Nageshrao, J. Kober, and R. Babuška, "Reinforcement learning based compensation methods for robot manipulators," *Engineering Applications of Artificial Intelligence*, vol. 78, pp. 236–247, 2019.

[24] "Vanilla policy gradient." [Online]. Available: https://spinningup.openai.com/en/latest/algorithms/vpg.html

[25] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, "Deep q-learning from demonstrations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[26] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *International conference on machine learning*. PMLR, 2015, pp. 1889–1897.

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

[28] N. Lambert, "Deep rl case study: Chaotic gradients," Jan 2020. [Online]. Available: https://towardsdatascience.com/deep-rl-case-study-policy-based-vs-model-conditioned-gradients-in-rl

[29] A. Franceschetti, E. Tosello, N. Castaman, and S. Ghidoni, "Robotic arm control and task training through deep reinforcement learning," *arXiv preprint arXiv:2005.02632*, 2020.

[30] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1587–1596.

[31] S. Dankwa and W. Zheng, "Twin-delayed ddpg: A deep reinforcement learning technique to model a continuous movement of an intelligent robot agent," in *Proceedings of the 3rd International Conference on Vision, Image and Signal Processing*, 2019, pp. 1–5.

[32] ——, "Modeling a continuous locomotion behavior of an intelligent agent using deep reinforcement technique." in *IEEE 2nd International Conference on Computer and Communication Engineering Technology (CCET)*, 2019, pp. 172–175.

[33] D. Machalek, T. Quah, and K. M. Powell, "Dynamic economic optimization of a continuously stirred tank reactor using reinforcement learning," in *American Control Conference (ACC)*, 2020, pp. 2955–2960.

[34] J. Li and T. Yu, "Deep reinforcement learning based multi-objective integrated automatic generation control for multiple continuous power disturbances," *IEEE Access*, vol. 8, pp. 156 839–156 850, 2020.

[35] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.