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

EMG-based Torque Prediction for Assistive Exoskeleton Control using Neural Networks with Bounded Generalization Error

Duy Hoang[1] Lucas Quesada[2] Bastien Berret[3] Olivier Bruneau[4] Laurent Fribourg[1]

Abstract— **Electromyography (EMG) signals are widely used in assistive exoskeleton control for predicting human joint torque due to their ability to extract muscle activations before movement onset. The standard procedure for learning the EMG-to-torque model involves training the model on a training set of EMG-torque data, followed by validating the model on a separate test set. The comparison between models is generally undertaken on the test set. However, the analysis of model performance on the data outside the test set remains unaddressed. The lack of a guarantee for unseen data reduces the reliability of EMG-to-torque models in practical exoskeleton control. In this paper, we address this issue by proposing a bounded-generalization-error neural network (BGNN) for EMG-based torque prediction. Using gradient descent to train the network, we formulate at each training step a theoretical upper bound on the generalization error, reflecting the prediction error across the entire data distribution, including unseen data beyond the test set. The NN training is terminated when this upper bound reaches its minimum, thereby achieving the tightest guarantee on the generalization error. Experimental results on torque prediction demonstrated that, while ensuring such a bounded generalization error, our method still gave results comparable to those of classical models. The use of our BGNN in assistive exoskeleton control was also tested with 13 participants on a pick-and-place task with an upper limb exoskeleton. Experimental results on assistive control revealed that our method can reduce human physical fatigue without compromising movement speed or accuracy compared to natural human movement characteristics, particularly for generalization in novel tasks.**

I. Introduction

Human torque estimation is crucial in many torque-based control strategies for robotic exoskeletons [1], [2], [3]. Several approaches have been explored utilizing different types of input signals to estimate human torque during robot operation. Typical input choices for the torque estimation model are mechanical torque sensors, mechanomyography [4], electromyography (EMG) [5], [6], or a combination of these modalities [7]. Compared to other sources, EMG signals are particularly advantageous due to their ability to extract information about muscle force production slightly before movement generation [2], [5]. This makes EMG a promising choice not only for human torque estimation but also for human torque prediction.

There are two main approaches for designing an EMG-based torque estimation model: methods based on neuromusculoskeletal models [6], [8], [9] and data-driven methods [5], [6], [10], [11]. Whereas model-based methods exploiting state-of-the-art knowledge about muscle neuromechanics require a long duration of training to identify individual biomechanical parameters, data-driven methods induce a shorter calibration time while offering a comparable estimation quality [5], [12]. For this reason, our work focuses on the class of data-driven methods. The ultimate goal of an EMG-to-torque model is to minimize the prediction error over a distribution $\mathcal{D}$ of all possible EMG-torque data (referred to as generalization error). Although numerous models can generate accurate data for inputs lying within an evaluated test set [5], [12], [13], no assessment is generally provided concerning the data outside the test set. In other words, there is no quantification of the generalization error. In contrast, we present here a model for which an upper bound on the generalization error is guaranteed. This property is particularly valuable for addressing safety concerns in EMG-based exoskeleton control, particularly in inter-task scenarios [14] where the robot executes tasks not represented in the training data of the EMG-to-torque model.

Motivated by [15], we develop an EMG-to-torque model using a neural network (NN), which is trained via the gradient descent (GD) algorithm. At each step of GD, we formulate an upper bound on the generalization error induced by the NN. The training is then stopped at the GD step that gives the minimum boundary on the generalization error. This provides us with an evaluation of model performance over the entire data distribution $\mathcal{D}$, which is generally ignored in both standard models in exoskeleton control [10], [11] and deep learning models [12], [13]. Additionally, the method of [15] allows us to achieve a reliable generalization error even with a simple NN architecture, as it does not impose assumptions on the number of NN neurons. This resolves the problem of high computational demands that limit the real-world implementation of deep learning models [12], [13] for exoskeleton control.

The main contributions of this paper lie in applying a novel NN training framework [15] to EMG-based torque modeling and evaluating its performance in the context of assistive exoskeleton control. We first examine the torque prediction quality of our method on a published dataset of 17 participants provided by [5], in comparison with state-of-the-art models [5], [10], [11], [12]. Subsequently, we investigate the

This work is supported by a grant from the "Fondation CFM pour la Recherche", through the Jean-Pierre Aguilar fellowship.

[1] Duy Hoang and Laurent Fribourg are with Université Paris-Saclay, CNRS, ENS Paris-Saclay, LMF, 91190 Gif-sur-Yvette, France.

[2] Lucas Quesada is with CEA List, 91120 Palaiseau, France.

[3] Bastien Berret is with Université Paris-Saclay, Inria, CIAMS, 91190 Gif-sur-Yvette, France.

[4] Olivier Bruneau is with Université Paris-Saclay, ENS Paris-Saclay, LURPA, 91190 Gif-Sur-Yvette, France.

Correspondence: Duy Hoang (e-mail: hoangduy@lmf.cnrs.fr).

assistance performance of the exoskeleton using our EMG-to-torque model in an experimental setup of 13 participants doing a pick-and-place task. To test the generalizability of the proposed method, an inter-task scenario is considered in which the pick-and-place movements are performed with a load of 1.5 kg while the NN training data is collected without the load. Results showed that, in terms of torque prediction, our method was comparable to other EMG-to-torque models on the test set while providing a guarantee on the data beyond this set. In terms of exoskeleton control, our method provided a suitable level of assistance as it maintained human natural speed and accuracy while reducing the human physical work. These findings suggest that bounded-generalization-error NN enables an effective EMG-based assistive exoskeleton, even in inter-task scenarios with a focus on usability and generalization.

II. Materials and Methods

A. Human torque prediction via neural network

1) NN design: We consider a single-output NN to predict the human torque for each active degree of freedom. The structure of the NN is illustrated in Fig. 1 with three main elements: input layer, hidden layer, and output layer. The input layer relates to the filtered EMG signals from d different muscles, represented as the input vector $\boldsymbol{x} \in \mathbb{R}^d$. The filtering process is performed using a fourth-order Butterworth band-pass filter (20–450 Hz), followed by centering, rectification, envelope extraction with a 3 Hz low-pass filter, and normalization by maximum voluntary contraction as [5], [16]. The hidden layer consists of m neurons, with individual r^{th} neuron weights denoted by $\boldsymbol{w}_r \in \mathbb{R}^d$ and hidden-layer weights by $\boldsymbol{W} = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m) \in \mathbb{R}^{d\times m}$. The output layer produces the predicted torque of the NN, with output-layer weights denoted by $\boldsymbol{a} = (a_1, \ldots, a_m) \in \mathbb{R}^m$. The NN output $\hat{y}$ is calculated as:

$$\hat{y} = f(\boldsymbol{W}, \boldsymbol{a}, \boldsymbol{x}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \zeta(\boldsymbol{w}_r^\top \boldsymbol{x}), \tag{1}$$

in which ζ is 1-Lipschitz activation function.

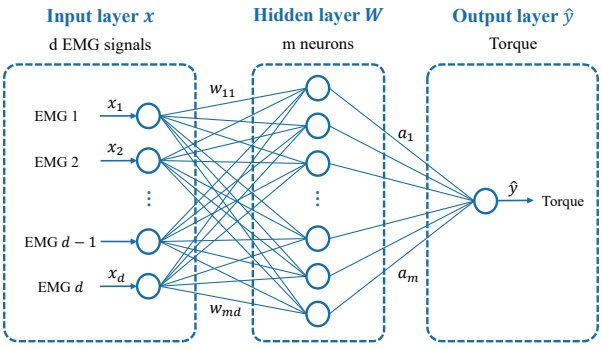

Fig. 1: Neural network EMG-to-torque model

The single-output NN with one hidden layer is used in this study since the generalization error bound in [15] is applied only for this architecture, and its low computational cost enables implementation on our robotic system.

2) NN training with bounded generalization error: We denote by $\mathcal{D}$ the data distribution containing all EMG-torque data of an individual subject. The NN is trained to give the minimal prediction error over $\mathcal{D}$ (referred to as generalization error or $L_{\mathcal{D}}$). However, we are not able to entirely access $\mathcal{D}$, we are only given a training set S consisting of n samples drawn independently and identically distributed from $\mathcal{D}$. To deal with this problem, the authors of [15] proposed formulating an upper bound on the generalization error of NN during the training (called $L_{\mathcal{B}}$), and the training process is stopped when $L_{\mathcal{B}}$ starts increasing. We briefly present here the results of [15] applied to the training of our NN EMG-to-torque model.

a) NN training by GD: We first initialize the hidden-layer weight to almost zero and the output-layer weight uniformly distributed in $\{-1, 1\}$. The output-layer weight is fixed during the training, while the hidden-layer weight is updated at each training step via the GD algorithm using the training dataset $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$:

$$\boldsymbol{w}_r(k+1) = \boldsymbol{w}_r(k) - \frac{\eta}{\sqrt{m}} \sum_{i=1}^{n} v_i(k) a_r \zeta'(\boldsymbol{w}_r^\top \boldsymbol{x}_i) \boldsymbol{x}_i \tag{2}$$

where $k = 0, 1, \ldots$ relates to the k^{th} step of GD, ζ' is the derivative of ζ, and $v_i(k) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \zeta(\boldsymbol{w}_r(k)^\top \boldsymbol{x}_i) - y_i$ corresponds to the error of the i^{th} training sample at step k.

b) Upper bound on generalization error: According to [15], the generalization error of the NN (1) is defined as:

$$L_{\mathcal{D}}(f) = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\ell(f(\boldsymbol{x}), y)],$$

where $\ell(\cdot,\cdot)$ is the elementary function. Here, we focus on the case that $\ell(\cdot,\cdot)$ is the absolute value of the prediction error, which yields:

$$L_{\mathcal{D}}(f) = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[|f(\boldsymbol{x}) - y|\right].$$

Let

$$c_k = \max_{r\in[m], k'\leqslant k} \|\boldsymbol{w}_r(k')\| \tag{3}$$

and consider two positive real numbers β and M satisfies:

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[\|\boldsymbol{x}\|^2\right] \leqslant \beta^2, \quad |f(\boldsymbol{x}) - y| \leqslant M \;\forall\; (\boldsymbol{x}, y) \sim \mathcal{D} \tag{4}$$

where $\mathcal{X} \subset \mathbb{R}^d$ is the input space distribution containing all possible instances of EMG signals $\boldsymbol{x}$. We hence define at each GD step:

$$L_{\mathcal{B}}(k) = \frac{1}{n} \sum_{i=1}^{n} |v_i(k)| + 4\beta\sqrt{\frac{m}{n}} c_k + 3M\sqrt{\frac{\log\frac{2}{\delta}}{2n}} \tag{5}$$

From [15], we have:

Proposition 1. *(cf. Section II of [15]) With probability at least $1-\delta$ over the sample S of size n, the generalization error $L_{\mathcal{D}}$ satisfies for all $k \in \mathbb{N}$:*

$$L_{\mathcal{D}}(f) \leqslant L_{\mathcal{B}}(k) \tag{6}$$

According to Proposition 1, we have $L_{\mathcal{B}}$ defined by (5) is an upper bound on the generalization error $L_{\mathcal{D}}$.

By definition, β represents an upper bound on the average input norm, whereas M indicates the worst prediction between the model $f(\boldsymbol{x})$ and the true output y. Therefore, from (5), it can be seen that larger values of β and M result in a larger upper bound on the generalization error $L_{\mathcal{B}}$.

c) Early stopping decision: We aim to stop the training when the generalization error boundary $L_{\mathcal{B}}$ reaches its minimum. The stopping step k^* is thus the largest k such that, for all $k' \in [k]$:

$$L_{\mathcal{B}}(k') < L_{\mathcal{B}}(k'-1). \tag{7}$$

We summarize our method in the following algorithm:

Algorithm 1. NN EMG-to-torque model training with bounded generalization error

1) Initialization:
   - Fix the number of input EMG signals d and the number of neurons m.
   - Initialize the NN weights to almost zero.
   - Calculate $L_{\mathcal{B}}(0)$ by (5).
2) NN training at step $k+1$:
   - Update NN weights via GD formula (2).
   - Calculate the generalization error upper bound $L_{\mathcal{B}}(k+1)$ using the formula (5).
3) Early stopping trigger:
   - If $L_{\mathcal{B}}(k+1) > L_{\mathcal{B}}(k)$, stop training.

B. Assistive exoskeleton control

Considering the dynamic model of the exoskeleton in the following Euler-Lagrange form:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{\tau}_e + \boldsymbol{\tau}_i \tag{8}$$

where $\boldsymbol{q} \in \mathbb{R}^N$ is the state vector of the robot representing the joint angles with corresponding joint velocity $\dot{\boldsymbol{q}}$ and joint acceleration $\ddot{\boldsymbol{q}}$; matrix $\boldsymbol{M} \in \mathbb{R}^{N\times N}$ denotes the inertia matrix, $\boldsymbol{C} \in \mathbb{R}^{N\times N}$ relates to the Centrifugal and Coriolis component, and vector $\boldsymbol{G} \in \mathbb{R}^N$ stands for the gravity terms of the robot; the vector $\boldsymbol{\tau}_e \in \mathbb{R}^N$ denotes the control signal and the vector $\boldsymbol{\tau}_i \in \mathbb{R}^N$ represents the human-robot interaction torque. The exoskeleton controller is designed to provide the required assistive torque for the upcoming movement of the human. As the EMG signals come prior to the movement, we set the output of the EMG-to-torque model as the desired interaction torque between the human and the robot, denoted by $\boldsymbol{\tau}_r$. This desired torque enables the robot to give full assistance. Indeed, the human-robot interaction torque $\boldsymbol{\tau}_i = \boldsymbol{\tau}_r$ means that the robot contributes to the full human torque demand for the imminent movement. Thus, the control signal $\boldsymbol{\tau}_e$ is calculated to minimize the difference between $\boldsymbol{\tau}_i$ and its reference $\boldsymbol{\tau}_r$. We use a PI corrector for this purpose, given by the following formula:

$$\boldsymbol{\tau}_{PI}(t) = K_P \boldsymbol{e}_{\boldsymbol{\tau}}(t) + K_I \int_{t_0}^{t} \boldsymbol{e}_{\boldsymbol{\tau}}(s)ds \tag{9}$$

where $\boldsymbol{e}_{\boldsymbol{\tau}} = \boldsymbol{\tau}_r - \boldsymbol{\tau}_i$, K_P and K_I are the proportional and integral control gains, respectively. To reduce the impact of the exoskeleton's gravity on the interaction, we also integrate a gravity compensator $\boldsymbol{\tau}_{GC} = \boldsymbol{G}(\boldsymbol{q})$ into the control loop as [17]. The final control signal for the exoskeleton $\boldsymbol{\tau}_e$ is obtained by:

$$\boldsymbol{\tau}_e(t) = \boldsymbol{\tau}_{PI}(t) + \boldsymbol{\tau}_{GC}(t) \tag{10}$$

The control diagram of the EMG assistance control strategy is illustrated in Fig. 2

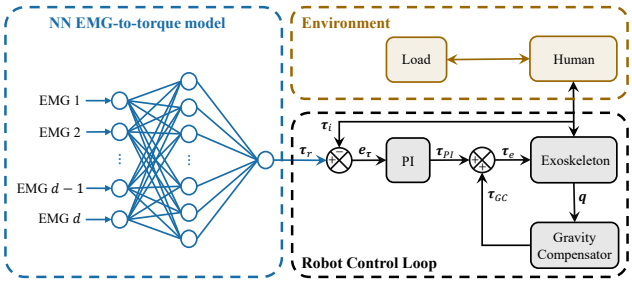

Fig. 2: Control diagram of the EMG assistance. The output torque of the EMG-to-torque model is considered as the desired torque $\boldsymbol{\tau}_r$. A PI corrector $\boldsymbol{\tau}_{PI}$ is applied to minimize the difference between the desired torque $\boldsymbol{\tau}_r$ and the current human-robot interaction torque $\boldsymbol{\tau}_i$. The exoskeleton's gravity is compensated by a gravity compensator $\boldsymbol{\tau}_{GC}$. The sum $\boldsymbol{\tau}_e$ of $\boldsymbol{\tau}_{PI}$ and $\boldsymbol{\tau}_{GC}$ is calculated at the end and sent to the robot as the control signal.

To measure different variables for the calculation of the control signal $\boldsymbol{\tau}_e$, we use MiniWave sensors (Cometa, Bareggio MI, Italy) for the EMG, a force/torque sensor located at the robot's end effector for the human-robot interaction $\boldsymbol{\tau}_i$, and the exoskeleton's incremental encoders for the joint angles $\boldsymbol{q}$. The reference desired torque $\boldsymbol{\tau}_r$ and the control signal $\boldsymbol{\tau}_e$ are updated every millisecond (1 kHz frequency).

C. Experimental Design

We consider two experiments to validate: (i) the performance of the proposed NN in torque prediction and (ii) the effectiveness of the proposed NN EMG-to-torque model. Two upper limb movements are involved in the experiments, including elbow and shoulder flexion/extension. Accordingly, two NNs were designed to predict the human torque, one for the elbow and one for the shoulder. Each NN contains $m = 32$ neurons in the hidden layer, $d = 8$ EMG signals as the input, corresponding to 8 muscles: brachioradialis, brachialis, the long, medial, and lateral triceps, and the anterior, posterior, and medial deltoids as recommended by [18]. We use $\zeta(\cdot) = tanh(\cdot)$ as the activation function and $\eta = 10^{-4}$ as the learning rate. The NN is trained according to Algorithm 1. Since the training process is terminated based on the evolution of the upper bound on the generalization error $L_{\mathcal{B}}$, no validation set is required. This allows us to benefit the entire dataset to train the NN.

We hereafter identify two parameters β and M, as specified in (4). The parameter β is estimated by considering the mean $\mu(\cdot)$ and standard deviation $\sigma(\cdot)$ of $\|\boldsymbol{x}\|^2$ for all $\boldsymbol{x}$ taken from the input training dataset $\mathcal{X}_S$:

$$\beta = \sqrt{\mu_{\boldsymbol{x}\in\mathcal{X}_S}(\|\boldsymbol{x}\|^2) + \sigma_{\boldsymbol{x}\in\mathcal{X}_S}(\|\boldsymbol{x}\|^2)} \tag{11}$$

Next, under the $tanh(\cdot)$ activation function, the NN output is limited between $[-\sqrt{m}, \sqrt{m}]$. Thus, we denote y_m as the maximum torque in the training set and scale the training torque by $\sqrt{m}/y_m$, thereby also keeping it within $[-\sqrt{m}, \sqrt{m}]$. This allows us to set $M = 2\sqrt{m}$. Following the scaling step, the NN output is accordingly rescaled to obtain the final prediction of the human torque (see Fig. 3)

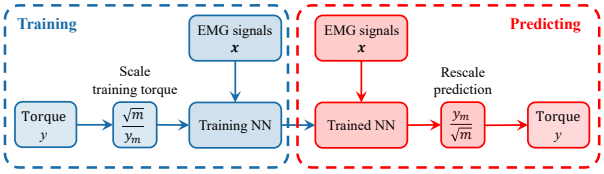

Fig. 3: The training torque is scaled by $\sqrt{m}/y_m$ before training the NN. After training, the output of the trained NN is rescaled by $y_m/\sqrt{m}$ to get the final human torque prediction.

1) Experiment on human torque prediction: In the first experiment, we compare our bounded-generalization-error neural network (BGNN) with the multivariate linear regression (MVLR) model [10], the synergy-based (SYN) model [11], the nonlinear mapping (NLMap) model [5], and the long short-term memory (LSTM) neural networks [12] in single joint torque prediction of the elbow. All the considered models were examined on the same dataset as used by [5]. The creation of the dataset is described in [5]. The dataset contains EMG-torque data of 17 participants. For each participant, 10 trials of elbow movements were performed during which the EMG signals and human torque were captured. We divided these 10 trials into a training set of 5 trials and a test set of the remaining 5 trials. Each considered model was trained on the training set and tested on the test set of the same participant.

2) Experiment on EMG assistance: This experiment was conducted to validate the effectiveness of the assistive control strategy using the BGNN. The robot was employed to support human elbow and shoulder flexion/extension movements in a parasagittal plane. Two NNs were trained to predict the human shoulder and elbow torque, respectively. The controller design was given in Section II-B. We refer to this control mode as EMG assistance (EA). The performance of our EA will be compared with natural human movements (no-exoskeleton mode NE) and the transparent mode (TR), where the exoskeleton tracks human movements without causing any modifications [17].

a) Participants: The experiment was undertaken on 13 healthy participants (5 females) with an average age of 24.69 ± 1.75 years old, average height of 1.67 ± 0.08 m, and average weight of 64.85 ± 11.25 kg. Participants with neurological or motor impairments are excluded from the study, as the proposed method is intended for industrial applications with healthy users. Each participant provided written informed consent before the experiment. The experimental protocol was approved by the local ethics committee.

b) Task: Each participant performed 2 sessions involving elbow and shoulder flexion/extension in a parasagittal plane. The first session was to gather individual data for NN

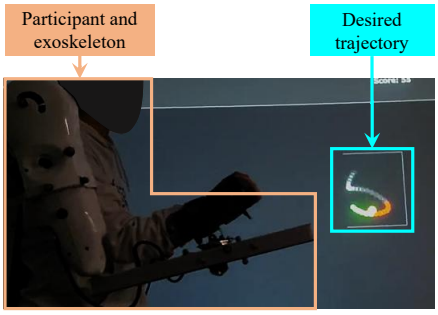

Fig. 4: Experimental setup for session 1, participants followed trajectories projected on a screen while wearing the ABLE exoskeleton. The task is undertaken without the load.

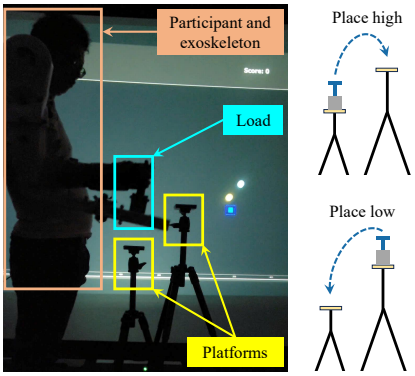

Fig. 5: Experimental setup for session 2, participants picked the load from the lower platform and placed it on the higher platform (place high - PH), then picked the load from the higher platform and placed it on the lower platform (place low - PL).

training. We follow the same methodology provided by [5] for this step. The participant followed a trajectory projected on a screen while wearing the ABLE exoskeleton [19] (see Fig. 4). A total of 5 trials related to 5 different trajectories were recorded for each subject in this session. After the data was collected, we trained the NNs and started the second session. For this session, the participant was asked to perform pick-and-place movements with a load of 1.5 kg between two platforms (see Fig. 5). Two movement types were considered: picking the load from the low platform and placing it on the high platform (place high – PH), and picking the load from the high platform and placing it on the low platform (place low – PL). Further details of the experimental protocol can be found in [16]. Each participant performed the task subsequently under 3 conditions (NE, TR, and EA) in random order. For each condition, 25 PH and 25 PL movements were undertaken. At the end of each condition, the participant answered a feedback questionnaire on their perception of the task (see Section III-B.3 for details).

3) Statistical analysis: We conduct the statistical analysis to report the significant effect of: (i) EMG-to-torque models on torque prediction quality and (ii) control conditions on control performance. The normality of the data is first checked by the Shapiro-Wilk test. If the data distribution is normal, a repeated measures analysis of variance (rm-ANOVA) is conducted to report the differences between

the factors (EMG-to-torque models or control conditions). A post-hoc t-test with Holm-Bonferroni correction is then applied to determine significance levels. If the data distribution is not normal, the Friedman test is used instead of the rm-ANOVA, and the post-hoc Wilcoxon-Nemenyi test is applied instead of the t-test. For all tests, differences between factors were considered significant if $p < 0.05$.

III. Results

A. Results on human torque estimation

The considered modes are compared on three evaluation metrics: the normalized mean absolute error (NMAE), the normalized upper bound on the generalization error over the entire data distribution (NLB), and Pearson's correlation coefficient R. The selection of the NMAE and NLB for the evaluation is based on the fact that generalization error is guaranteed in terms of the absolute error by Proposition 1.

Model	NMAE	NLB	R
BGNN	7.617$\pm$ 1.099	19.286 $\pm$ 1.550	0.957$\pm$ 0.016
MVLR [10]	8.746 $\pm$ 1.469	Not given	0.936 $\pm$ 0.028
SYN [11]	9.536 $\pm$ 1.578	Not given	0.927 $\pm$ 0.030
NLMap [5]	6.338 $\pm$ 1.181	Not given	0.966 $\pm$ 0.015
LSTM [12]	5.032 $\pm$ 0.682	Not given	0.981 $\pm$ 0.005

TABLE I: Average results on torque estimation of different models across 17 subjects in NMAE: Normalized mean absolute error, NLB: Normalized upper bound on mean absolute error over the entire data distribution of individual participant, and R: Pearson's correlation coefficient.

Results on the test set showed that our BGNN induced significantly less NMAE than MVLR and SYN ($p < 0.001$), but significantly more NMAE than NLMap or LSTM ($p < 0.01$). The same trend was observed regarding the R coefficient as NN had a significantly bigger R than MVLR and SYN ($p < 0.001$), but significantly smaller R than NLMap and LSTM ($p < 0.01$).

However, considering the data outside the test set, no guarantees were provided by MVLR, SYN, NLMap, or LSTM. On the other hand, our method provided a theoretical upper bound on the mean absolute error over the entire data distribution of each participant. This is confirmed by the fact that the NMAE on the test set was consistently lower than the corresponding subject's NLB.

B. Results on EMG assistance

The NE, TR, and our EA are compared based on their impacts on muscle activation, movement kinematics, energy expenditure during the task, and subjective questionnaires.

1) Impacts on muscle activation: As shown in Fig. 6, on average, our EA mode reduced muscle activation compared to both NE and TR conditions in all muscles except the triceps medial head. Statistical analysis revealed that our EA induced significantly lower muscle activation than NE and TR in the brachialis and brachioradialis ($p < 0.05$). For other muscles, no main difference was found between EA and NE. However, a significant reduction in muscle activation was found in EA compared to TR in terms of the triceps long head ($p < 0.01$), triceps lateral head ($p < 0.01$), anterior deltoid ($p < 0.05$), and posterior deltoid ($p < 0.001$). Considering the triceps medial head, although EA gave the highest muscle activation, there was no main effect within the conditions.

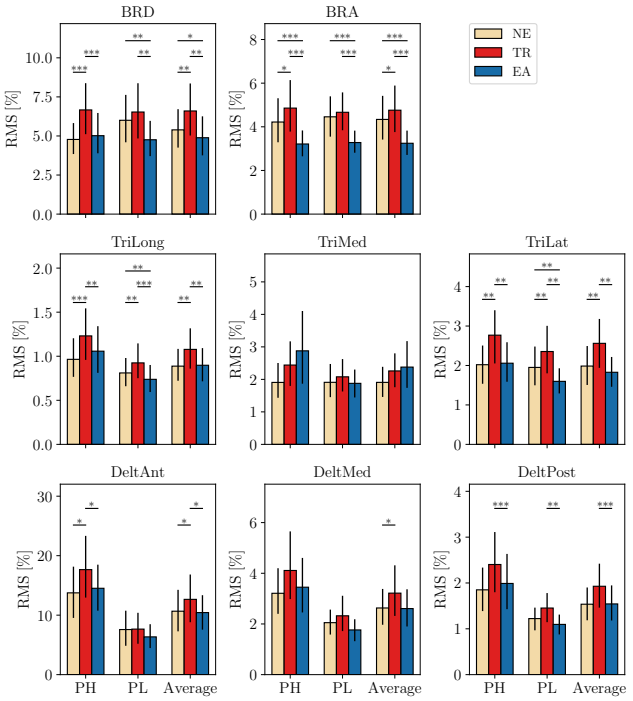

Fig. 6: Results on muscle activation across 13 participants. Considered muscles: BRD = Brachioradialis, BRA = Brachialis, TriLong = Triceps long head, TriMed = Triceps medial head, TriLat = Triceps lateral head, DeltAnt = Anterior deltoid, DeltMed = Medial deltoid, DeltPost = Posterior deltoid. Motion phases: PH = Place high, PL = Place low. Statistical significant is indicated by: * < 0.05, ** < 0.01, and *** < 0.001.

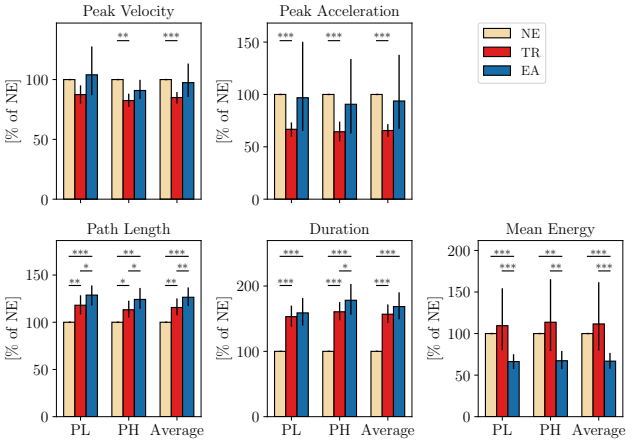

Fig. 7: Results on human kinematics across 13 participants. Motion phases: PH = Place high, PL = Place low. Statistical significant is indicated by: * < 0.05, ** < 0.01, and *** < 0.001.

2) Impacts on human kinematics and energy: Fig. 7 describes the impact of the 3 considered control conditions on human kinematics and energy. A significant alteration in human kinematics was reported when the task was performed with the exoskeleton in the EA and TR conditions. Both EA and TR significantly increased the total path length

and the duration of the movement ($p < 0.01$). However, compared to NE, our EA mode provided a similar level of movement acceleration with no significant difference. In contrast, TR induced significantly lower peak velocity and peak acceleration ($p < 0.001$) than NE, which was consistent with the results observed in [17], [20]. Considering the energy exerted by participants during the task, the EA resulted in the lowest energy consumption, with a significant reduction compared to both NE and TR ($p < 0.001$).

3) Impacts on human perception: Table II summarizes the results on feedback questionnaires. Results indicated that the EA controller induced significantly less physical fatigue than TR ($p < 0.01$) and achieved a score comparable to NE, with no statistically significant difference. The same observation was obtained in fast as EA was significantly faster than TR ($p < 0.05$) but remained similar to NE. In terms of mental tiring and accuracy, although NE gave the best score (the least mental tiring and the most accurate), no important difference was found between the control conditions. In contrast, considering the simplicity, participants agreed that it was significantly simpler to perform the task in NE compared to both TR and EA ($p < 0.05$).

Question	Score		
	NE	TR	EA
Were you physically tiring? Higher score = More tiring	2.4 ± 1.1	3.6 ± 0.7	2.5 ± 0.8
Were you mentally tiring? Higher score = More tiring	1.1 ± 0.2	1.2 ± 0.4	1.3 ± 0.6
Were you fast? Higher score = Faster	3.6 ± 1.0	2.6 ± 0.6	3.4 ± 1.0
Were you accurate? Higher score = More accurate	4.5 ± 0.7	3.9 ± 1.0	3.9 ± 1.0
Was the task simple? Higher score = More simple	4.6 ± 0.5	4.2 ± 0.6	4.2 ± 0.7

TABLE II: Results on feedback questionnaires across 13 participants. Each participant rated their performance with a score ranging from 1 to 5 for each question per condition.

C. Discussion

This work aimed to evaluate the use of the BGNN for EMG-to-torque modeling in assistive exoskeleton control. For this purpose, we conducted the experiments to validate: (i) the torque prediction quality of BGNN, and (ii) the assistive performance of an EMG-based exoskeleton control strategy using BGNN.

Results on EMG-to-torque prediction showed that, considering the performance on the test set, our BGNN lay in the middle of the trade-off between model simplicity and model accuracy, as it achieved superior performance than linear models like MVLR and SYN but underperformed the nonlinear models like NLMap and LSTM. However, considering the entire data distribution (including data beyond the test set), our BGNN provided a theoretical guarantee on the upper bound of the generalization error, whereas this was ignored in both NLMap and LSTM. The absence of such guarantees undermines the reliability of NLMap and LSTM in real-world exoskeleton applications, as they may produce unsafe torque predictions when confronted with novel tasks. By contrast, our approach enhances the trustworthiness of the NN EMG-based torque prediction in assistive exoskeleton control by ensuring the model's performance even under unseen data.

For the assistive exoskeleton experiment, we first emphasized the applicability of our model in real-world implementation, where deep architectures like LSTM are impeded by their computational demand. Then, we underlined the generalizability of our method in an inter-task setup where the BGNN trained with load-free movement was used to support load-carrying movements. The control experiment revealed that our EA control strategy effectively assisted the human in carrying the load. Evidence for this conclusion comes from the decreased muscle activation and lower mean energy expenditure observed with EA compared to NE and TR (see Fig. 6 and 7). Subjective assessments via the questionnaires also suggested that EA caused less physical fatigue than TR and a similar level to NE. No significant difference was found between our EA and human natural movement in terms of mental fatigue, fast, and accuracy. While these results pointed to the acceptability of human–robot interaction, the final subjective questionnaire revealed that performing the task with the exoskeleton was more challenging than without it. This might be due to the fact that all participants had no experience with the exoskeleton. In this case, long-term training is recommended to help the participant familiarize themselves with the robot [21], [22], thereby simplifying the task.

IV. Conclusion

This paper proposed a data-driven approach for predicting human joint torque using EMG signals and NNs. Unlike previous studies, our method established a theoretical upper bound on the generalization error, guaranteeing the reliability of the model over the entire data distribution (even outside the test set). Two experiments were undertaken to verify the effectiveness of our method in both human torque prediction and assistive exoskeleton control. The results on torque prediction revealed that our BGNN achieved comparable performance to other state-of-the-art models while ensuring an upper bound on the generalization error. On the other hand, the results on assistive control showed that our control strategy with BGNN provided an attenuation of human physical effort while preserving the speed and accuracy of natural movement. The generalization ability of our BGNN was also emphasized in an inter-task context, as the task executed by the participant was different from the task used to train the NN.

While our study focused on with/without load movements in a parasagittal plane, future extension could be considered to examine the generalizability of the proposed method on more complex movements (such as 3D trajectories with different speeds and varying loads [23], [24]). To do this, further research on extending the framework of bounded generalization error to multi-layer and multi-output neural networks is required.

REFERENCES

[1] T. Proietti, V. Crocher, A. Roby-Brami, and N. Jarrasse, "Upper-limb robotic exoskeletons for neurorehabilitation: A review on control strategies," *IEEE reviews in biomedical engineering*, vol. 9, pp. 4–14, 2016.

[2] 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.

[3] B. Treussart, F. Geffard, N. Vignais, and F. Marin, "Controlling an upper-limb exoskeleton by EMG signal while carrying unknown load," in *2020 IEEE international conference on Robotics and automation (ICRA)*. IEEE, 2020, pp. 9107–9113.

[4] C. Caulcrick, W. Huo, W. Hoult, and R. Vaidyanathan, "Human joint torque modelling with MMG and EMG during lower limb human-exoskeleton interaction," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7185–7192, 2021.

[5] L. Quesada, D. Verdel, O. Bruneau, B. Berret, M.-A. Amorim, and N. Vignais, "EMG-to-torque models for exoskeleton assistance: a framework for the evaluation of in situ calibration," *The International Journal of Robotics Research*, p. 02783649251414884, 2026.

[6] L. Zhang, Z. Li, Y. Hu, C. Smith, E. M. G. Farewik, and R. Wang, "Ankle joint torque estimation using an EMG-driven neuromusculoskeletal model and an artificial neural network model," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 564–573, 2020.

[7] Q. Zhang, K. Lambeth, Z. Sun, A. Dodson, X. Bao, and N. Sharma, "Evaluation of a fused sonomyography and electromyography-based control on a cable-driven ankle exoskeleton," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 2183–2202, 2023.

[8] K. R. Holzbaur, W. M. Murray, and S. L. Delp, "A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control," *Annals of biomedical engineering*, vol. 33, pp. 829–840, 2005.

[9] Y. Zhou, J. Li, S. Zuo, J. Zhang, M. Dong, and Z. Sun, "An online estimating framework for ankle actively exerted torque under multi-DOF coupled dynamic motions via sEMG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2024.

[10] C. Camardella, M. Barsotti, D. Buongiorno, A. Frisoli, and V. Bevilacqua, "Towards online myoelectric control based on muscle synergies-to-force mapping for robotic applications," *Neurocomputing*, vol. 452, pp. 768–778, 2021.

[11] M. Hamaya, T. Matsubara, J.-i. Furukawa, Y. Sun, S. Yagi, T. Teramae, T. Noda, and J. Morimoto, "Exploiting human and robot muscle synergies for human-in-the-loop optimization of EMG-based assistive strategies," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 549–555.

[12] L. Zhang, D. Soselia, R. Wang, and E. M. Gutierrez-Farewik, "Estimation of joint torque by EMG-driven neuromusculoskeletal models and LSTM networks," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 3722–3731, 2023.

[13] P. Xia, J. Hu, and Y. Peng, "EMG-based estimation of limb movement using deep learning with recurrent convolutional neural networks," *Artificial organs*, vol. 42, no. 5, pp. E67–E77, 2018.

[14] M. Hamaya, T. Matsubara, T. Noda, T. Teramae, and J. Morimoto, "Learning task-parametrized assistive strategies for exoskeleton robots by multi-task reinforcement learning," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5907–5912.

[15] D. Martin Xavier, L. Chamoin, and L. Fribourg, "Early stopping strategy using neural tangent kernel theory and Rademacher complexity," in *2025 American Control Conference (ACC)*. IEEE, 2025, pp. 1301–1306.

[16] L. Quesada, D. Verdel, O. Bruneau, B. Berret, M.-A. Amorim, and N. Vignais, "Less is more: uncompensated gravity torques for intuitive EMG-based assistance with a robotic exoskeleton," *bioRxiv*, 2025. [Online]. Available: https://www.biorxiv.org/content/early/2025/11/24/2025.11.21.689589

[17] D. Verdel, A. Farr, T. Devienne, N. Vignais, B. Berret, and O. Bruneau, "Human movement modifications induced by different levels of transparency of an active upper limb exoskeleton," *Frontiers in Robotics and AI*, vol. 11, p. 1308958, 2024.

[18] L. Quesada, D. Verdel, O. Bruneau, B. Berret, M.-A. Amorim, and N. Vignais, "EMG feature extraction and muscle selection for continuous upper limb movement regression," *Biomedical Signal Processing and Control*, vol. 103, p. 107323, 2025.

[19] P. Garrec, J.-P. Friconneau, Y. Measson, and Y. Perrot, "ABLE, an innovative transparent exoskeleton for the upper-limb," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 1483–1488.

[20] S. Bastide, N. Vignais, F. Geffard, and B. Berret, "Interacting with a "transparent" upper-limb exoskeleton: A human motor control approach," in *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2018, pp. 4661–4666.

[21] M. Dufraisse, J. Cegarra, J.-J. A. Kouadio, I. Clerc-Urmès, and L. Wioland, "From unknown to familiar: An exploratory longitudinal field study on occupational exoskeletons adoption," *Applied Ergonomics*, vol. 122, p. 104393, 2025.

[22] G. Diamond-Ouellette, A. Telonio, T. Karakolis, J. Leblond, L. Bouyer, and K. Best, "Exploring the change in metabolic cost of walking before and after familiarization with a passive load-bearing exoskeleton: A case series," *IISE Transactions on Occupational Ergonomics and Human Factors*, vol. 10, no. 3, pp. 161–172, 2022.

[23] M. Oghogho, M. Sharifi, M. Vukadin, C. Chin, V. K. Mushahwar, and M. Tavakoli, "Deep reinforcement learning for emg-based control of assistance level in upper-limb exoskeletons," in *2022 International Symposium on Medical Robotics (ISMR)*. IEEE, 2022, pp. 1–7.

[24] P. Sedighi, X. Li, and M. Tavakoli, "EMG-based intention detection using deep learning for shared control in upper-limb assistive exoskeletons," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 41–48, 2023.