Pattern Recognition Letters 99 (2017) 67–76

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

# Learning assistive strategies for exoskeleton robots from user-robot physical interaction

Masashi Hamaya[a,b,*], Takamitsu Matsubara[a,c], Tomoyuki Noda[a], Tatsuya Teramae[a], Jun Morimoto[a]

[a] The Department of Brain Robot Interface, ATR-CNS, Kyoto, Japan
[b] The Graduate School of Frontier Bioscience, Osaka University, Osaka, Japan
[c] The Graduate School of Information Science, Nara Institute of Science and Technology, Nara, Japan

ARTICLE INFO

*Article history:*
Available online 10 April 2017

*Keywords:*
Exoskeleton robot
Human-robot physical interaction
Human-in-the- loop
Reinforcement learning

ABSTRACT

Social demand for exoskeleton robots that physically assist humans has been increasing in various situations due to the demographic trends of aging populations. With exoskeleton robots, an *assistive strategy* is a key ingredient. Since interactions between users and exoskeleton robots are bidirectional, the assistive strategy design problem is complex and challenging. In this paper, we explore a data-driven learning approach for designing assistive strategies for exoskeletons from user-robot physical interaction. We formulate the learning problem of assistive strategies as a policy search problem and exploit a data-efficient model-based reinforcement learning framework. Instead of explicitly providing the desired trajectories in the cost function, our cost function only considers the user's muscular effort measured by electromyography signals (EMGs) to learn the assistive strategies. The key underlying assumption is that the user is instructed to perform the task by his/her own intended movements. Since the EMGs are observed when the intended movements are achieved by the user's own muscle efforts rather than the robot's assistance, EMGs can be interpreted as the "cost" of the current assistance. We applied our method to a 1-DoF exoskeleton robot and conducted a series of experiments with human subjects. Our experimental results demonstrated that our method learned proper assistive strategies that explicitly considered the bidirectional interactions between a user and a robot with only 60 seconds of interaction. We also showed that our proposed method can cope with changes in both the robot dynamics and movement trajectories.

## 1. Introduction

Social demand for exoskeleton robots that physically assist humans has been increasing in various situations due to the demographic trends of aging populations. Applications have been proposed for augmenting able-bodied people [1–5], supporting physically challenged people [6–8], and rehabilitation [9–11]. In such exoskeletons, one key ingredient is a control method that generates robot actions as assistance based on user intentions: *an assistive strategy*. Human-robot interactions are generally bidirectional, where the robot provides an assist force to users and detects their reactions or movement intentions through sensors. Therefore, the assistive strategy design problem is complex and challenging.

Several assistive strategies have been proposed over the last decade. A typical approach is based on gravity compensation control [2,3,12,13], which effectively supports the load to maintain a static posture. Another popular strategy is the electromyography (EMG)-based method [14,15]. EMG-to-force models convert the subject's EMG signals to the joint torques of an assistive robot. For walking and balancing assistance, an inverted pendulum model with Center of Mass (CoM) and Zero Moment Point (ZMP) can derive stable gait patterns [8,16]. The adaptive oscillator-based strategy has also received much attention for assisting periodic movements [4,17–19]. An extension with a state-machine-based controller has also been proposed [20]. Most of these methods are based on independent models of users and robots. However, since the users and the exoskeletons physically interact in a bidirectional way, it might be desirable to explicitly consider such interactions in assistive strategy design.

Complex human-robot interaction has also been explored in various contexts. For user-robot collaboration tasks, movement primitives were learned from two interaction behaviors of users by Hidden Markov Model (HMM) [21] or Dynamic Movement

[*] Corresponding author.
*E-mail address:* hamaya@atr.jp (M. Hamaya).

http://dx.doi.org/10.1016/j.patrec.2017.04.007

Primitives (DMP) [22]. Moderes et al. applied an optimal control framework to find the appropriate robot impedance parameters for a human-robot cooperative reaching task [23]. For cooperative transportation tasks, some learning methods have also been explored. Medina et al. proposed an experience-driven robotic assistance control by HMM to learn the user's intention during task execution [24]. Rozo et al. presented a learning-from-demonstration framework for physical collaborative robot behaviors using task-parameterized Gaussian Mixture Models (GMMs) and optimal control for cooperative transportation tasks [25] and extended it with a stiffness estimation based on a convex optimization for the assembly of furniture mechanical structures [26]. Learning approaches from interaction data are often utilized not only for physical interactions but also for communication. Mitsunaga et al. applied a policy gradient RL method with which a robot can learn the proper distance between users and itself so that they feel comfortable [27]. Such research successfully learned the adaptive controllers of user behaviors from user-robot interaction data.

Based on the above successful studies of complex interaction designs, in this paper, we explore such a data-driven learning approach for designing assistive strategies for exoskeletons from user-robot physical interaction data. A few recent studies have applied learning methods for assistive strategy design on walking-aid robot control [28], robotic training for dart-throwing [29], and exoskeleton walking assistance [30,31]. However, two serious problems have not been sufficiently explored for applying learning methods for physical human-robot interactions. First, since collecting a large amount of interaction data imposes a heavy burden on users, long-term learning experiments cannot be conducted with users. In such studies, to reduce the required amount of interaction data, learning methods were applied only for a small number of parameters in pre-designed controllers. Second, the designing cost (or reward) functions are not straightforward. For typical autonomous robot control problems, the cost function is set with such task dependent information as the desired target locations or trajectories. However, for assistive scenarios using task-specific costs, this approach is inappropriate because the desired targets or trajectories must be determined by the user instead of the robot.

Instead of explicitly providing the desired trajectories in the cost function as references, our cost function only considers the user's muscular effort measured by electromyography signals (EMGs) to learn the assistive strategies. The key underlying assumption is that the user is instructed to perform the task by his/her own intended movements. Since EMGs are observed when the intended movements are achieved by the user's own muscle efforts rather than the robot's assistance, EMGs can be interpreted as the "cost" of the current assistance. Based on this assumption, we expect that a suitable assistive strategy for a user to perform an intended movement is learned by minimizing the EMG-based cost function without requiring the desired trajectories in the cost function.

We formulate the learning problem of assistive strategies as a policy search problem and exploit a data-efficient model-based reinforcement learning framework called Probabilistic Inference for Learning Control (PILCO) [32]. Our motivation to use PILCO is its data-efficiency property, which becomes crucial for such human-in-the-loop applications as assistive robotic devices. PILCO was compared to other model-free and model-based RL methods for a cart-pole swing-up task [32], and it outperformed the other RL algorithms by at least one order of magnitude. PILCO is also applicable for probabilistic continuous state-action systems that might fit human-in-the-loop exoskeletons, rather than approaches with deterministic system modeling and trajectory optimization [33].

In our preliminary study, we demonstrated that our method can efficiently learn proper assistive strategies with a simulated robot arm control task based on user EMGs [34]. In this paper, we applied our method to a real 1-DoF exoskeleton robot and thoroughly investigated its effectiveness for learning assistive strategies from user-robot interaction data.

This paper is organized as follows. In Section 2, we introduce how we formulated the assist policy learning problem by explicitly considering user-robot physical interaction. In Section 3, we explain how a data-efficient model-based reinforcement learning framework can be used in our assist policy learning problem. In Section 4, we present our experimental setup to evaluate our proposed method. In Section 5, we present our experimental results and discuss them in Section 6. Finally, in Section 7, we conclude this paper.

## 2. Problem formulation

This section formulates the learning problem of assistive strategy from the direct interactions shown in Fig. 1. We assume that the robot is physically coupled and securely attached to the user.

Since the robot's future state depends on its current state, its action, and the user's action, the robot dynamics can be written as follows:

$$\boldsymbol{s}_{t+1} = \boldsymbol{g}(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{v}_t) + \boldsymbol{\zeta}_t, \quad \boldsymbol{\zeta}_t \sim \mathcal{N}(0, \Sigma_\zeta), \tag{1}$$

where $\boldsymbol{s}_t$ is the robot's state (e.g., joint angles and velocities) and $\boldsymbol{u}_t$ is its action (e.g., joint torques or air pressures generated by pneumatic actuators). $\boldsymbol{v}_t$ is the user's action (e.g., muscle activations), and $\boldsymbol{\zeta}_t$ is an additive Gaussian noise that represents model uncertainty.

On the other hand, the user's action is decided by the user's control policy that can be based on the robot's state, the robot's action, or the previous user's action. Thus, the user's control policy can be modeled:

$$\boldsymbol{v}_{t+1} = \boldsymbol{h}(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{v}_t) + \boldsymbol{\eta}_t, \quad \boldsymbol{\eta}_t \sim \mathcal{N}(0, \Sigma_\eta), \tag{2}$$

where $\boldsymbol{\eta}_t$ is an additive Gaussian noise.

By integrating them into one equation, human-robot integrated dynamics can be represented:

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t) + \boldsymbol{\xi}_t, \quad \boldsymbol{\xi}_t \sim \mathcal{N}(0, \Sigma_\xi), \tag{3}$$

where

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{s} \\ \boldsymbol{v} \end{bmatrix}, \quad \Sigma_\xi = \begin{bmatrix} \Sigma_\zeta & 0 \\ 0 & \Sigma_\eta \end{bmatrix}. \tag{4}$$

Based on the above system, we formulate our learning problem of assistive strategies. Our objective is to find a robot control policy (assistive strategy) $\pi : \pi(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{u}$ that minimizes the long-term cost:

$$J^\pi(\boldsymbol{\theta}) = \Sigma_{t=0}^{T} \mathbb{E}_{\boldsymbol{x}_t}[c(\boldsymbol{x}_t)], \ \ \boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0), \tag{5}$$

where $J^\pi$ evaluates the cost of $T$ steps, $\boldsymbol{\theta}$ is an adjustable parameter vector, so-called policy parameter, and $c(\boldsymbol{x}_t)$ is given as:

$$c(\boldsymbol{x}_t) = 1 - \exp\left(-\frac{1}{2\sigma_c^2}\boldsymbol{x}_t^\top \boldsymbol{T}\boldsymbol{x}_t\right)$$
$$\boldsymbol{T} = \begin{pmatrix} 0 & 0 \\ 0 & \boldsymbol{T}_v \end{pmatrix}, \tag{6}$$

where $\sigma_c$ is the width of the cost function and $\boldsymbol{T}$ is a diagonal matrix that expresses the weight of each element of the state in the cost function. This expression for Eq. (6) is suggested by the PILCO framework, analytically computes the expected cost over the policy, and makes the learned dynamics tractable. Note that unlike typical autonomous robot control problems, cost function $c(\boldsymbol{x}_t)$ does not incorporate such task dependent information as the desired target locations or trajectories. Instead, our cost function only considers the user's muscular effort that can be measured by

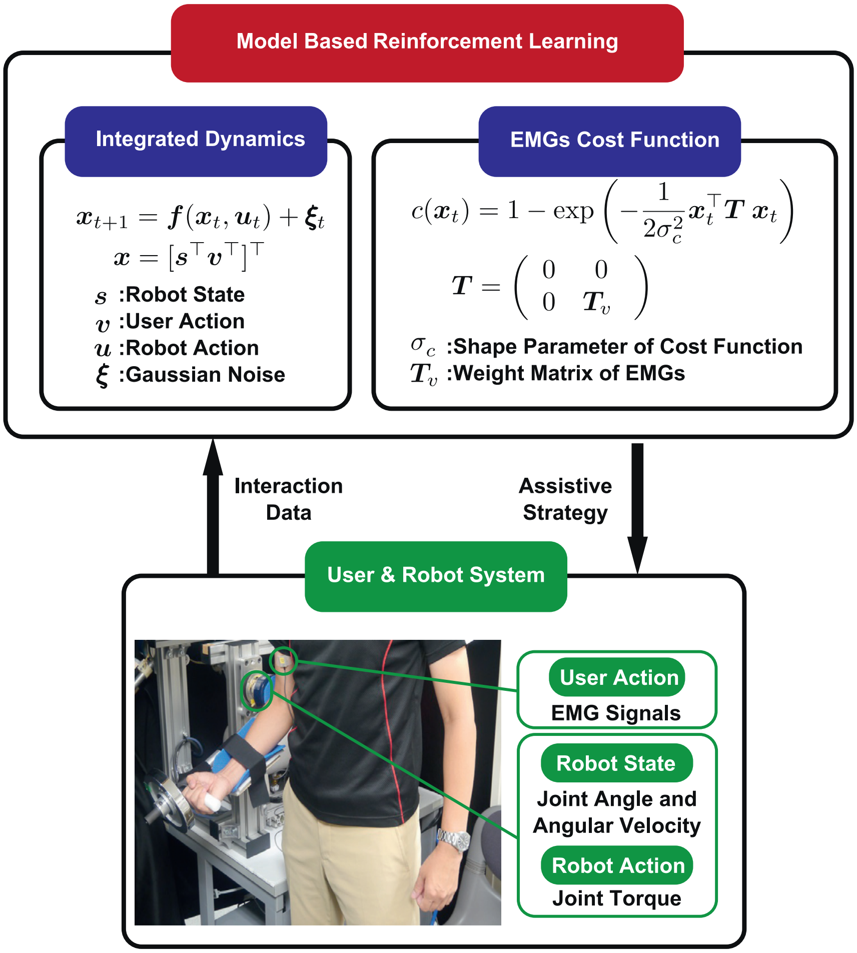
Model Based Reinforcement Learning
Integrated Dynamics
$x_{t+1} = f(x_t, u_t) + \xi_t$
$x = [s^\top v^\top]^\top$
$s$ :Robot State
$v$ :User Action
$u$ :Robot Action
$\xi$ :Gaussian Noise
EMGs Cost Function
$c(x_t) = 1 - \exp\left(-\frac{1}{2\sigma_c^2} x_t^\top T x_t\right)$
$T = \begin{pmatrix} 0 & 0 \\ 0 & T_v \end{pmatrix}$
$\sigma_c$ :Shape Parameter of Cost Function
$T_v$ :Weight Matrix of EMGs
Interaction Data
Assistive Strategy
User & Robot System
User Action
EMG Signals
Robot State
Joint Angle and Angular Velocity
Robot Action
Joint Torque

**Fig. 1.** Schematic diagram of our approach. We formulated assistive strategies based on user-robot integrated system and task-free EMGs cost function and adopted data-efficient model-based reinforcement learning to derive control policy.

EMGs because the robot needs to *assist* the user's voluntary motions rather than control them. Such a cost function design, which allows the user to lead the learning process, results in an appropriate assistive strategy [34].

The difficulty of solving the above problem reflects the modeling of such interaction dynamics. Due to the inclusion of the user action policy and the interaction effects between the user and the robot, it does not follow rigid body dynamics anymore. Moreover, since human subjects tend to become tired when experiments are conducted for a long period, collecting large-size data is cumbersome for learning systems.

## 3. Learning assistive strategies by reinforcement learning

We apply PILCO, a model-based policy search method [32], to the assistive problem. PILCO uses probabilistic non-parametric Gaussian processes to consider the uncertainty of models. Since PILCO analytically computes long-term predictions, policy evaluations, and policy improvements, it can perform critical, data-efficient learning. In this section, we briefly summarize PILCO. More details about it can be found [32].

### 3.1. Model learning

For target dynamics modeling, PILCO uses Gaussian process regression [35] where $(\boldsymbol{x}_t, \boldsymbol{u}_t) \in \mathbb{R}^{D+F}$ is the training input and $\boldsymbol{\Delta}_t = \boldsymbol{x}_{t+1} - \boldsymbol{x}_t \in \mathbb{R}^D$ is the training output. It typically uses the following kernel function:

$$k(\tilde{\boldsymbol{x}}_p, \tilde{\boldsymbol{x}}_q) = \sigma_f^2 \exp\left(-\frac{1}{2}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)^\top \boldsymbol{\Lambda}^{-1}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)\right) + \delta_{pq}\sigma_\xi^2, \tag{7}$$

where $\tilde{\boldsymbol{x}} := [\boldsymbol{x}^\top, \boldsymbol{u}^\top]$, $\boldsymbol{\Lambda}$ is a precision matrix that expresses the characteristic length and $\sigma_f$ is the bandwidth parameter. These parameters are learned with $n$ training inputs $\tilde{\boldsymbol{X}} = [\tilde{\boldsymbol{x}}_1, ..., \tilde{\boldsymbol{x}}_n]$ and targets $\boldsymbol{y} = [\boldsymbol{\Delta}_1, ..., \boldsymbol{\Delta}_n]$.

The predictive distribution of $\boldsymbol{x}_{t+1}$ is analytically given as follows:

$$p(\boldsymbol{x}_{t+1}|\boldsymbol{x}_t, \boldsymbol{u}_t) = \mathcal{N}(\boldsymbol{x}_{t+1}|\boldsymbol{\mu}_{t+1}, \boldsymbol{\Sigma}_{t+1}), \tag{8}$$

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{x}_t + \mathbb{E}_f[\boldsymbol{\Delta}_t], \quad \boldsymbol{\Sigma}_{t+1} = \mathrm{Var}_f[\boldsymbol{\Delta}_t], \tag{9}$$

where

$$\mathbb{E}_f[\boldsymbol{\Delta}_t] = m_f(\tilde{\boldsymbol{x}}_t) = \boldsymbol{k}_*^\top(\boldsymbol{K} + \sigma_\xi^2\boldsymbol{I})^{-1}\boldsymbol{y} = \boldsymbol{k}_*^\top\beta \tag{10}$$

$$\mathrm{Var}_f[\boldsymbol{\Delta}_t] = k_{**} - \boldsymbol{k}_*^\top(\boldsymbol{K} + \sigma_\xi^2\boldsymbol{I})^{-1}\boldsymbol{k}_*. \tag{11}$$

Here, $\boldsymbol{k}_* := k(\tilde{\boldsymbol{X}}, \tilde{\boldsymbol{x}}_t)$, $k_{**} := k(\tilde{\boldsymbol{x}}_t)$, and $\beta := (\boldsymbol{K} + \sigma_\xi^2\boldsymbol{I})^{-1}\boldsymbol{y}$, where $\boldsymbol{K}$ is a kernel matrix, each of whose element follows $K_{ij} = k(\tilde{x}_i, \tilde{x}_j)$.

### 3.2. Control policy

We employed the following control policy:

$$\tilde{\pi}(\boldsymbol{x}_*, \boldsymbol{\theta}) = \sum_{i=1}^{N} k(\boldsymbol{m}_i, \boldsymbol{x}_*)(\boldsymbol{K} + \sigma_\pi^2\boldsymbol{I})^{-1}\boldsymbol{t} = k(\boldsymbol{M}, \boldsymbol{x}_*)^\top\boldsymbol{\alpha}. \tag{12}$$

Here, $\boldsymbol{\alpha} = (\boldsymbol{K} + \sigma_\pi^2\boldsymbol{I})^{-1}\boldsymbol{t}$, where $\boldsymbol{x}_*$ is the test input, $\boldsymbol{t}$ is a training target, $\boldsymbol{M} = [\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N]$ are the centers of the Gaussian basis functions, $\sigma_\pi^2$ is noise variance, and $k$ is a kernel function. Policy parameter $\boldsymbol{\theta}$ is composed of $\boldsymbol{M}$**,** $\boldsymbol{t}$, and the scale of the kernel functions $k$ in Eq. (12). To make it possible to learn suitable assistive strategies even for different tasks and users, we used a policy model with a high expressive capability of a variety of functions, as shown in Eq. (12), among multiple applicable choices [32]. This function is a kernel regression (or a deterministic Gaussian process regression), which allows us to represent a variety of complex nonlinear maps between the state and the action [35]. In Section 5.2, we show how greatly different policies are learned for different users by the same function shown in Eq. (12). For safe user-robot interactions, base policy $\pi$ with a control limit is designed as

$$\pi(\boldsymbol{x}_*, \boldsymbol{\theta}) = \boldsymbol{u}_{\max}\sigma(\tilde{\pi}(\boldsymbol{x}_*, \boldsymbol{\theta})), \tag{13}$$

where $\boldsymbol{u}_{\max}$ is a maximum output and $\sigma(x) \in [0, 1]$ is a squashing function.

### 3.3. Policy evaluation

To evaluate the control policy, we need to compute long-term cost $J^\pi$. Although it cannot be obtained analytically due to the Gaussian process model's complexity, PILCO employs a reasonable approximation scheme with an analytic moment matching technique.

To predict $\boldsymbol{x}_{t+1}$, PILCO assumes that distribution $p(\tilde{\boldsymbol{x}}_t) = p(\boldsymbol{x}_t, \boldsymbol{u}_t)$ is Gaussian and calculates $p(\boldsymbol{\Delta}_t)$ as follows:

$$p(\boldsymbol{\Delta}_t) = \iint p(f(\tilde{\boldsymbol{x}}_t)|\tilde{\boldsymbol{x}}_t)p(\tilde{\boldsymbol{x}}_t)\mathrm{d}f\mathrm{d}\tilde{\boldsymbol{x}}_t. \tag{14}$$

Eq. (14) is calculated analytically. PILCO also assumes that mean $\boldsymbol{\mu}_{\boldsymbol{\Delta}}$ and covariance $\boldsymbol{\Sigma}_\Delta$ of distribution $p(\boldsymbol{\Delta}_t)$ are known. Then the mean and covariance of $p(\boldsymbol{x}_{t+1}) = \mathcal{N}(\boldsymbol{x}_{t+1}|\boldsymbol{\mu}_{t+1}, \boldsymbol{\Sigma}_{t+1})$ are obtained:

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \boldsymbol{\mu}_{\boldsymbol{\Delta}}, \tag{15}$$

$$\boldsymbol{\Sigma}_{t+1} = \boldsymbol{\Sigma}_t + \boldsymbol{\Sigma}_{\boldsymbol{\Delta}} + \mathrm{cov}[\boldsymbol{x}_t, \boldsymbol{\Delta}_t] + \mathrm{cov}[\boldsymbol{\Delta}_t, \boldsymbol{x}_t]. \tag{16}$$

Based on this prediction distribution, expected value $\mathbb{E}_{\boldsymbol{x}_t}[c(\boldsymbol{x}_t)]$ can be computed analytically:

$$\mathbb{E}_{\boldsymbol{x}_t}[c(\boldsymbol{x}_t)] = \int c(\boldsymbol{x}_t)\mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)\mathrm{d}\boldsymbol{x}_t \tag{17}$$

$$= 1 - |\boldsymbol{I} + \Sigma_t\boldsymbol{T}|^{-1/2} \times \exp\left(-\frac{1}{2}(\boldsymbol{\mu}_t - \boldsymbol{x}_t^d)^\top\tilde{\boldsymbol{S}}(\boldsymbol{\mu}_t - \boldsymbol{x}_t^d)\right), \tag{18}$$

$$\tilde{\boldsymbol{S}} := \boldsymbol{T}(\boldsymbol{I} + \Sigma_t\boldsymbol{T})^{-1}. \tag{19}$$

With the above equations, we can analytically compute the approximation of $J^\pi$.

### 3.4. Policy improvement with analytic gradient

Policy parameter $\boldsymbol{\theta}$ is optimized by minimizing $J^\pi(\boldsymbol{\theta})$. Gradient $\partial J^\pi(\boldsymbol{\theta})/\partial\boldsymbol{\theta}$, which can be computed analytically using the chain-rule because of the policy evaluation's analytic expression, is expressed by $\varepsilon_t := \mathbb{E}_{\boldsymbol{x}_t}[c(\boldsymbol{x}_t)]$:

$$\frac{\mathrm{d}J^\pi(\boldsymbol{\theta})}{\mathrm{d}\boldsymbol{\theta}} = \sum_{t=1}^{T}\frac{\mathrm{d}\varepsilon_t}{\mathrm{d}\boldsymbol{\theta}},$$

$$\frac{\mathrm{d}\varepsilon_t}{\mathrm{d}\boldsymbol{\theta}} = \frac{\mathrm{d}\varepsilon_t}{\mathrm{d}p(\boldsymbol{x}_t)}\frac{\mathrm{d}p(\boldsymbol{x}_t)}{\mathrm{d}\boldsymbol{\theta}} := \frac{\partial\varepsilon_t}{\mathrm{d}\boldsymbol{\mu}_t}\frac{\mathrm{d}\boldsymbol{\mu}_t}{\partial\boldsymbol{\theta}} + \frac{\partial\varepsilon_t}{\mathrm{d}\boldsymbol{\Sigma}_t}\frac{\mathrm{d}\boldsymbol{\Sigma}_t}{\partial\boldsymbol{\theta}}. \tag{20}$$

Therefore, such a standard gradient-based non-convex optimization method as BFGS can be applied to find locally optimal policy parameter $\boldsymbol{\theta}$.

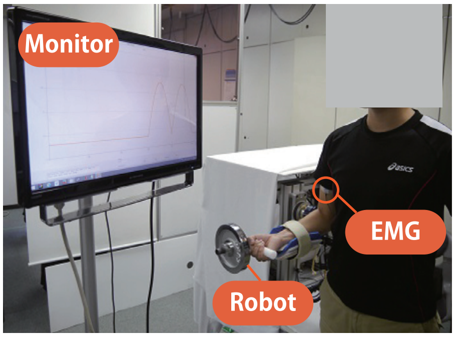

**Fig. 2.** Experimental setup: Subjects tracked reference trajectories presented on monitor by moving their elbow joint where forearm was physically attached to 1-DoF exoskeleton robot.

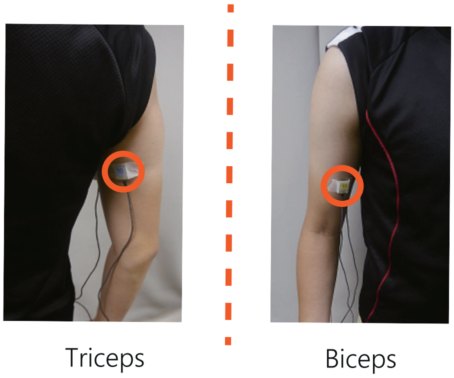

**Fig. 3.** EMGs electrode placement: We measured biceps and triceps EMGs.

## 4. Experimental setup

We developed an experimental platform based on a 1-DoF upper-limb elbow-joint exoskeleton robot (Fig. 2). Subjects followed the target joint trajectory that was presented on a monitor by moving their elbow joint where the forearm was physically attached to the 1-DoF exoskeleton robot. While the subjects were tracking the reference trajectories, we measured the interaction data and used them only to learn the assistive strategies. The subjects were instructed as follows. If the assistance made by the robot was helpful for achieving the task, they should relax and rely on it. On the other hand, if the assistance was unhelpful, the subjects should actuate their own muscles to achieve the task. We first conducted a joint-angle tracking task with five subjects to show the learning performance of our proposed method. Then we investigated whether the same learning system can cope with different experimental setups with one of the five subjects.

### 4.1. EMG measurements

We placed two EMGs sensors on the subject's forearm biceps and triceps (Fig. 3) to measure their physical effort. The sampling time was 0.004 s. The measured signals were rectified and low-pass filtered with a cutoff frequency of 2.0 Hz with a second-order Butterworth filter.

### 4.2. 1-DoF exoskeleton robot

The robot was driven by a pneumatic artificial muscle (PAM) actuator (FESTO Inc.), the link length was 0.4 m, and it weighed 1.7 kg (Fig. 4). The robot was equipped with a handle, and the subject was tightly secured to it. The low-level control period was 0.004 s. The input pressure was low-pass filtered with a cutoff fre-

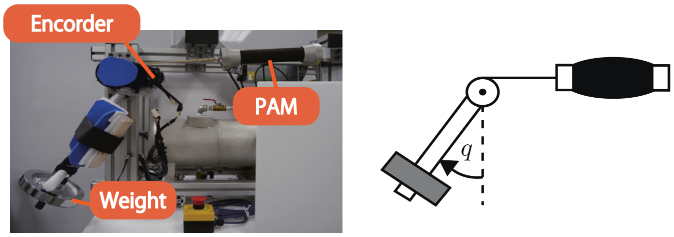

**Fig. 4.** 1-DoF upper-limb exoskeleton robot driven by pneumatic artificial muscle (PAM).

quency of 10 Hz to avoid sudden pressure changes. We evaluated our proposed method with two different load weights (2.5 and 3.75 kg) and with two different reference trajectory amplitudes (2.5 and 2.8 rad), as shown in Fig. 5. The 2.5-kg load weight with a reference trajectory amplitude of 2.5 rad was the default experimental setup. We first evaluated our proposed learning method on this default setup with five subjects and then investigated whether the assistive strategy learning system can cope with different experimental setups. We applied our learning method to two different experimental setups: with the heavier load weight (3.75 kg) or with the larger trajectory amplitude (2.8 rad).

### 4.3. Parameter settings in RL

We utilized PILCO open-source code [36]. The subject action and robot state in Eq. (3) were $\boldsymbol{x} = [q\ \dot{q}\ E_b\ E_t]^\top$, where $q$ was the robot's joint angle, $\dot{q}$ was the angular velocity, and $E_b$ and $E_t$ were the filtered-biceps and -triceps EMGs. $u$ was the desired pressure input to the PAMs. The initial joint angle was 1.3 rad (Fig. 5). The control period of PILCO was 0.2 s, and the prediction horizon was 4.0 s. The low-level control period was 0.004 s, and we used zero-order hold until the next control time step of PILCO. The weight of the cost function was $T_v = \mathrm{diag}(0.2\ 0.4)$, and the shape parameter of cost function $\sigma_c$ was 0.5 in Eq. (5). The number of basis functions $N$ in Eq. (12) was 10. The number of total policy parameters was 54. Maximum output $u_{\max}$ in Eq. (13) was 0.65 MPa.

One learning session was composed of five initial system identification trials and ten learning trials for each task. This one learning session corresponded to a total of 60 seconds and 300 points of data, and the sampling period was 0.2 s. On the initial system identification trials, the subject tried the task, and the robot generated random pressure. Its angle, its angular velocity, and the subject's EMG signal data were collected. In the learning trials, the robot learned the model from the input pressure and the measured data and optimized the policy parameters. The learned parameters were applied, the subject performed the task, and the robot collected new data. For the initial trials, we used the random pressure input around a periodic input pattern as $u_{\mathrm{init}} \sim \mathcal{N}(\mu_u, \sigma_u^2)$, where $\mu_u = u_{\max}|\sin(0.5\pi t)|$ and $\sigma_u^2 = 0.05$. After the 60 seconds initial system identification and learning trials, subjects tried the given joint-angle tracking task ten times with the learned assist control output. In addition, for comparison, subjects also conducted the task ten times without assistance.

## 5. Results

### 5.1. Performance of learned assistive strategies

Fig. 6 shows the mean of the EMGs and the tracking errors with five subjects. The EMGs were normalized by the percentages of maximum voluntary contraction (%MVC): %MVC $= E/E_{\max}$, where $E = \frac{1}{k}\Sigma_k e(k)$ and $e(k)$ is the rectified and low-pass filtered EMG at time step $k$. $E_{\max}$ was the maximum value of $e$ during the task execution. The gray bar indicates without assistance and the red one indicates with learned assistance. We conducted a statistical analysis between the two scenarios by paired t-tests. In Fig. 6(a), the mean of the biceps %MVC with the learned assistance was significantly lower than without assistance ($p < 0.05$). The muscle activities of the triceps were roughly constant and much lower than the biceps when we used the learned assistive strategy. Therefore, we did not compare the triceps activities, although they varied during the learning sessions and were useful for monitoring the human subject action to derive the control output of the exoskeleton robot. We show the tracking performance of the reference trajectory with and without learned assistive strategies in Fig. 6(b). The absolute means of the tracking error were not significantly different ($p = 0.94$). Therefore, we found that the subjects' muscle activities were reduced with the learned assistive strategies while they achieved similar tracking performance.

Fig. 7 shows the reference and the mean of the actual trajectories, the mean learned pressure input, and the mean biceps and triceps EMGs over ten test trials of one of the five subjects with the learned assistive strategy. In Fig. 7(a), the dashed line is the desired trajectory, and the solid line is the actual trajectories. The subject followed the desired trajectory with the learned assist control output of the exoskeleton robot. Fig. 7(b) shows the learned outputs. In Fig. 7(c) and (d), the blue line shows the biceps, and the green line shows the triceps EMGs. The subject generated large bicep activity at the beginning of the upward elbow movements due to the limitations of the actuators and the uncertainty of the user behaviors and EMGs. The triceps EMGs were basically constant.

### 5.2. Learned assistive strategies and interaction models

Fig. 8 shows the learned assistive strategies and interaction models with the Gaussian processes. Fig. 8(a) expresses the learned assistive strategies (pressure) given the robot's angle, angular velocity, and EMGs signals. (b) and (c) express the changes of the biceps and triceps EMGs between current and one-step-ahead times due to the robot assist pressure inputs at different joint angles when a subject lifted his arm. To visualize the assistive strategies and the interaction models on a 2D plane, we set the current angle and the angular velocity equal to the reference trajectory, and the biceps and the triceps activities were set as mean EMG values during the learning trials. As shown in Fig. 8(a), the learned assistive strategy depends not only on the robot's state but also on the user EMGs. Therefore, this resembles a shared control policy rather than a robot autonomous control policy. In Fig. 8(b), the changes of the biceps EMGs decreased as the angle and pressure increased. The subject was more relaxed when the robot properly assisted him. On the other hand, as shown in Fig. 8(c), the changes of the triceps EMGs were higher in the low-angle (around 1.3 rad) and high-pressure (around 0.65 MPa) regions. With this condition, the triceps increased intensively because the subject activated his triceps EMGs to reduce the tracking error when the robot generated excessive pressure.

Fig. 9(a) shows the learned assistive strategies for other four subjects. For subject A, they generated large pressure at around 2.0 rad and 0.06 mV, and the maximum pressure was smaller than the others. For subjects B, C, and D, the pressures increased as the angles increased. For subject B, the pressures were greatly generated at around 0 mV, and for subject C, they were greatly generated at around 0.06 mV. Fig. 9(b) shows the one-step changes of the biceps. For subjects A, B, and D, large EMGs were observed at around 1.3 rad. For subject C, this value shifted at around 2.0 rad. Fig. 9(c) shows the one-step changes of the triceps EMGs. Commonly for all the subjects, the EMGs tended to be larger as the angles increased around 2.4 rad.

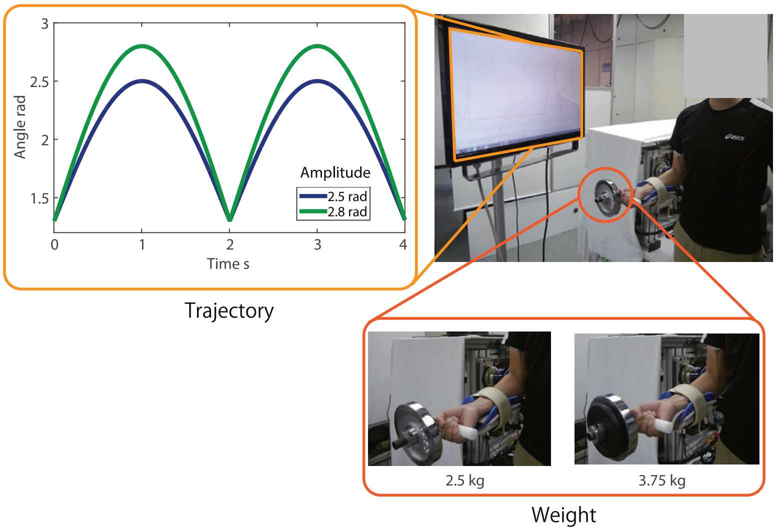

**Fig. 5.** Joint-angle tracking tasks. We evaluated our proposed method with two different load weights (2.5 and 3.75 kg) and with two different reference trajectory amplitudes (2.5 and 2.8 rad).

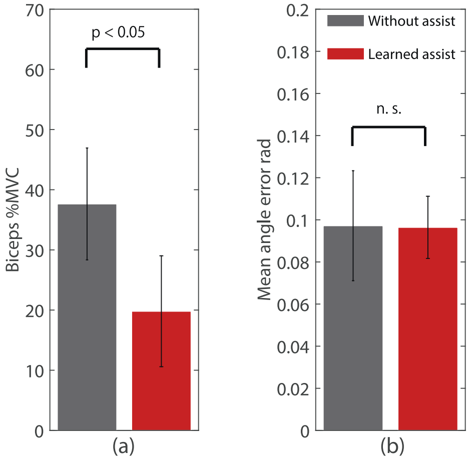

**Fig. 6.** Comparison between with and without learned assist under default experimental setup: (a) Biceps EMGs and (b) Tracking error. Biceps activities with learned assistance were significantly lower than without assistance ($p < 0.05$).

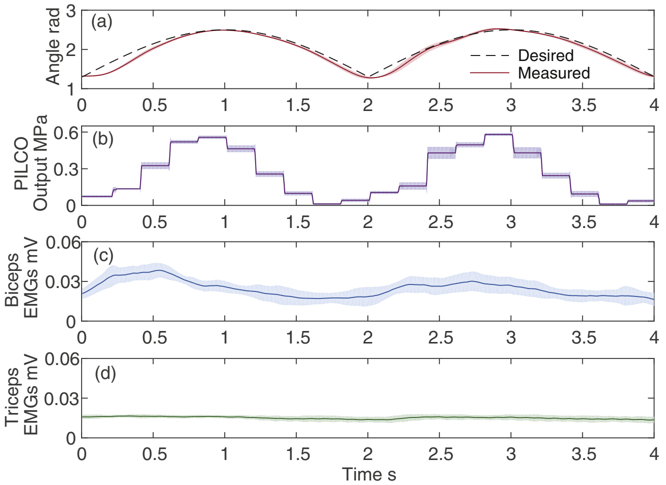

**Fig. 7.** Trajectories during elbow-joint movements with learned assistive strategy: (a) Reference, mean, and variance of measured trajectories; (b) Mean and variance of learned pressure; (c) and (d) Mean and variance biceps and triceps EMGs over ten test trials of one subject.

In summary, these results demonstrated the effectiveness of the learned assistive strategies for reducing the subject EMGs by explicitly considering the interaction model.

### 5.3. Learning process

Fig. 10 shows the accumulated long-term cost of one subject in each trial. The subject tried the learning session three times. The error bars show the standard deviations of the cost over the sessions. The accumulated cost quickly decreased and converged at around six trials.

Fig. 11 shows the learning process of the parametrized policy for one subject. Since the policy has 54 parameters, we visualized the policy maps of the 1st, 5th, and 10th trials on the 2D plane. According to the learning progress, the policy map became clearer and steeper.

### 5.4. Application to different experimental setups

First, we show the results under an experimental condition with a heavier load weight. In Fig. 12(a), we compared the biceps activities. Significant differences ($p < 0.001$) were observed between with and without the learned assistive strategy. Interestingly, as depicted in Fig. 12(b), the absolute tracking errors significantly decreased when we used the learned assistive strategy with $p < 0.005$. The tracking errors also decreased because generating precise elbow movements with more weight was harder than with less weight and the assist control input of the exoskeleton robot made this task easier.

Second, we show the results with an experimental condition with a larger reference trajectory amplitude. In Fig. 13(a), we compared the biceps activities. Significant differences ($p < 0.001$) were observed between with and without the learned assistive strategy. As in Fig. 13(b), in this experimental setup with a larger reference trajectory amplitude, we interestingly observed significant reduction in the triceps activities with $p < 0.001$. This is probably because a faster movement is required to track the reference trajectory with a larger amplitude in the same time period. We did not observe a significant difference in the tracking errors ($p = 0.59$).

In summary, our method learned the assistive strategies even for different experimental settings in the robot dynamics and the shape of the reference trajectories with the same assist learning

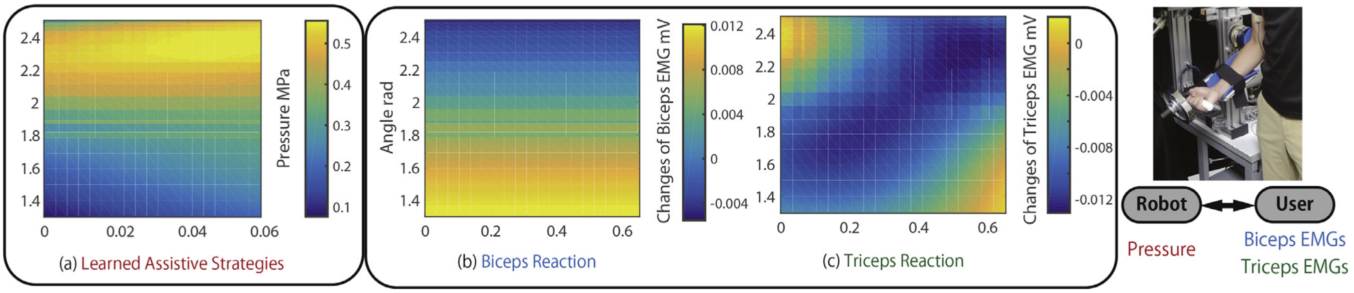

**Fig. 8.** (a) Learned assistive strategies when subject lifted his arm: (b) Changes of biceps EMGs, and (c) Changes of triceps EMGs.

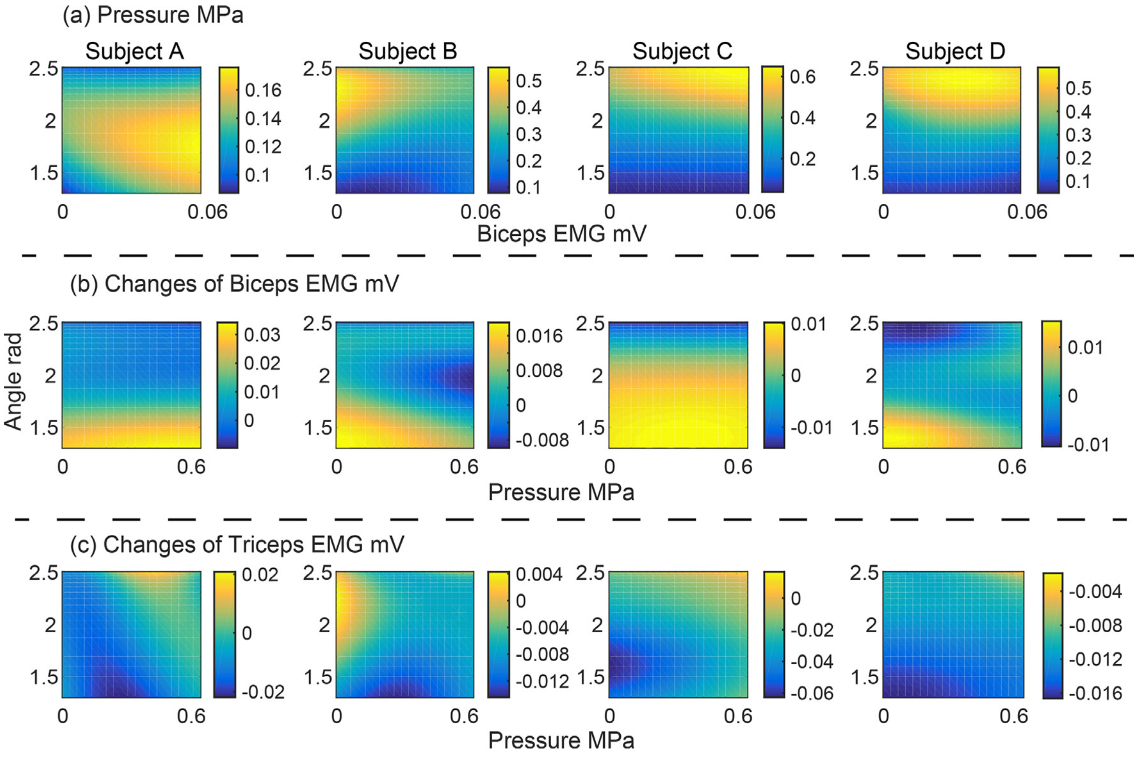

**Fig. 9.** (a) Learned assistive strategies for different subjects: (b) Changes of biceps and (c) Changes of triceps EMGs.

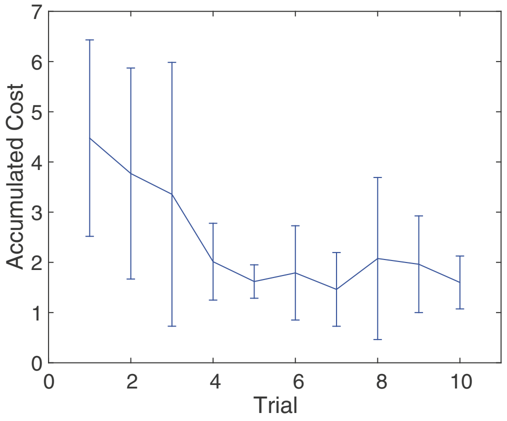

**Fig. 10.** Accumulated cost in each trial. It quickly decreased and converged at around six trials.

strategy. This result suggests the usefulness of our learning method for different assistive control applications.

Finally, we conducted additional experiments to investigate the effects of different parameters in the cost function for the learning process. We prepared five more parameters (Table 1) and compared the performance of the learned assistive strategies between the same and different parameters. As shown in Fig. 14, the resulting EMGs in the Biceps were not statistically significantly different ($p = 0.0965$). This result suggests that our method may not be sensitive for such parameter settings.

**Table 1**
Parameters of cost function.

| | Weight $\boldsymbol{T}_{\mathbf{v}} = \mathrm{diag}(\cdot)$ | Shape $\sigma_c$ |
|---|---|---|
| 1 | 0.2, 0.4 | 0.5 |
| 2 | 0.2, 0.2 | 0.5 |
| 3 | 0.4, 0.2 | 0.5 |
| 4 | 0.2, 0.4 | 0.3 |
| 5 | 0.2, 0.4 | 0.7 |

## 6. Discussion

Our experimental results suggest that our approach is relevant for learning assistive strategies because of its sample efficiency. The proper assistive strategies of a 1-DoF robot for trajectory tracking tasks were learned only with 60 seconds human-

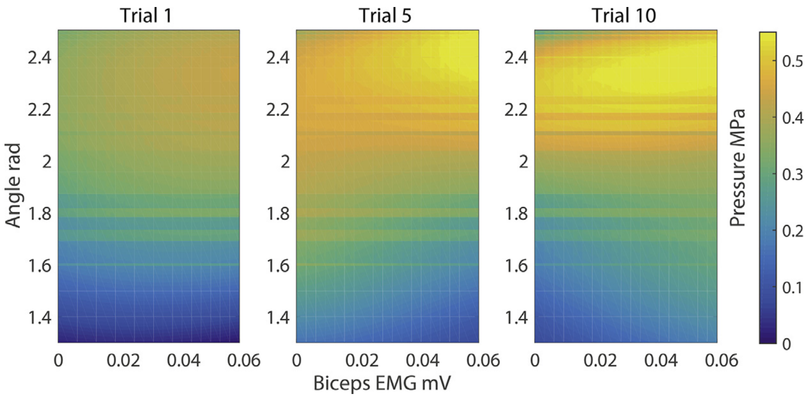

**Fig. 11.** Changes of policy map of one subject. Policy map became clearer and steeper.

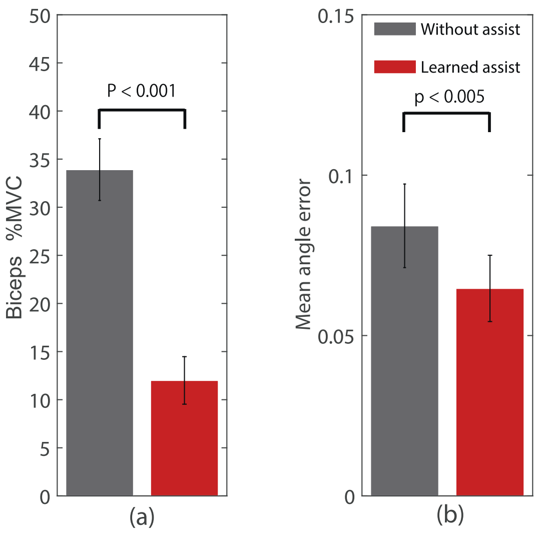

**Fig. 12.** Comparison between with and without learned assist under experimental setup with heavier load weight: (a) Biceps EMGs and (b) Tracking error. Biceps activities with learned assistance were significantly lower than without assistance ($p < 0.001$). Interestingly, absolute tracking error significantly decreased if we used learned assistive strategy ($p < 0.005$).

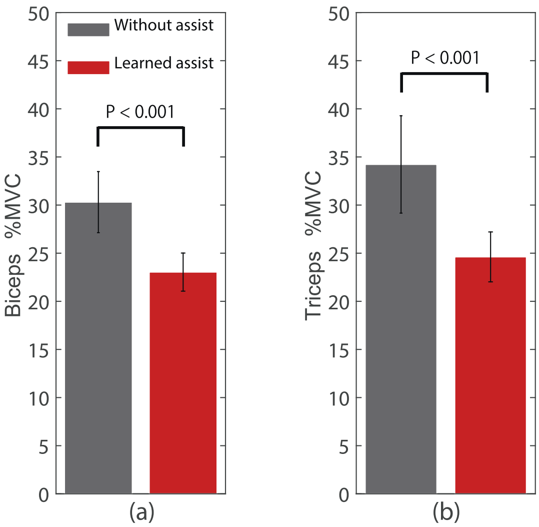

**Fig. 13.** Comparison between with and without learned assist under experimental setup with larger reference trajectory amplitude: (a) Biceps EMGs. Biceps activities with learned assistance were significantly lower than without assistance ($p < 0.001$). (b) Triceps EMGs. We also observed significant reduction in triceps activities ($p < 0.001$).

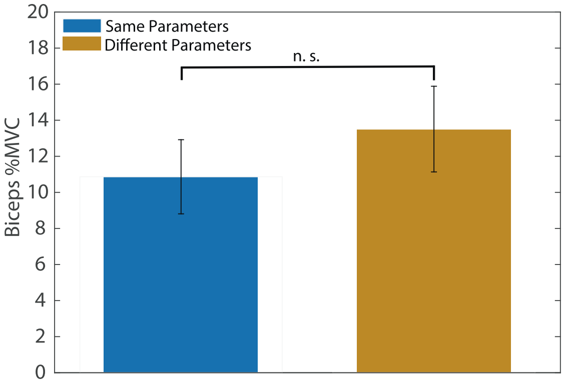

**Fig. 14.** Performance comparison of learned assistive strategies between same and different parameters. Biceps EMGs were not statistically significantly different $p > 0.05$.

robot interaction data. Although other existing approaches also reported their sample efficient learning, they are based on sample-inefficient, model-free reinforcement learning. Thus, the sample efficiency was achieved by utilizing task-specific knowledge in the policy design. For example, in both dart-throwing training assistance [29] and exoskeleton walking assistance [31], policies were carefully designed with only a few parameters to be learned. On the other hand, our framework with model-based reinforcement learning learned 54 parameters from scratch, which is evidence of its relevance for learning assistive strategies. Even though our proposed method certainly reduced user EMGs, the amount of EMGs did not become zero for the following two possible reasons: 1) inconsistency of user behaviors, which were not consistent among multiple trials even in the same task, and 2) uncertainty in the EMGs.

In our experimental task, the predictive horizon was clearly given from the task period. Thus, it is not a turning parameter. However, for more complex tasks where the horizon is not explicitly given from the task, it becomes another turning parameter which should be set properly for the task.

## 7. Conclusion

We directly learned assistive strategies from interactions between users and a robot. First, we formulated a learning problem of assistive strategies. To reduce the required number of interactions between a user and the robot to learn the assist policy, we applied a data-efficient, model-based reinforcement learning framework. To verify the effectiveness of our proposed method, we conducted a series of experiments. The results show that our method learned such a proper assistive strategy to reduce the user EMGs even for changes in the robot dynamics and the shape of the reference trajectories with the same task-free EMG-based cost function.

Our method can be extended to learn multiple task assistive strategies, but the learned strategies are task-dependent. Future work will extend our method using multi-task RL, which can be generalized even for unseen tasks from multiple sets of human-robot interaction data across different tasks. Generally, RL frameworks suffer from the curse of dimensionality. To extend the scalability of our framework for higher dimensional systems, we might utilize such dimensionality reduction techniques as synergies for both the user and robot. Since human muscles are not always activated independently, muscle synergies can be used to reduce the state dimension of humans [37]. On the other hand, robot control based on synergies has also been explored. Cunha et al. showed that only two synergies could construct the locomotive movements of a bipedal robot [38]. By exploiting both the user's and the robot's synergies, we might be able to extend our framework to apply our lower-limb exoskeleton robot with multiple DoFs [10,39]. Another future work will extend our framework to learn an assistive strategy that reduces fatigue in long-term runs by designing a fatigue-based cost function using a fatigue estimation method [40–42].

## Acknowledgments

This study was supported by the Strategic Research Program for Brain Sciences from the Japan Agency for Medical Research and Development, AMED, by JSPS KAKENHI JP16H06565, by NEDO, by MIC-SCOPE, by the Development of Medical Devices and Systems for Advanced Medical Services from AMED, by the ImPACT Program of Council for Science, Technology and Innovation (Cabinet Office, Government of Japan), and by the "Research and development of technology for enhancing functional recovery of elderly and disabled people based on non-invasive brain imaging and robotic assistive devices," the Commissioned Research of National Institute of Information and Communications Technology (NICT), JAPAN.

## References

[1] K. Yamamoto, K. Hyodo, M. Ishii, T. Matsuo, Development of power assisting suit for assisting nurse labor, JSME Int'l J. Series C 45 (3) (2002) 703–711.
[2] S. Karlin, Raiding iron man's closet, IEEE Spectr. 48 (8) (2011) 25.
[3] K. Kazerooni, A. Chu, R. Steger, That which does not stabilize, will only make us stronger, Int'l J. Rob. Res. 26 (1) (2007) 75–89.
[4] X. Zhang, M. Hashimoto, SBC for motion assist using neural oscillator, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2009, pp. 659–664.
[5] A. Asbeck, S. De Rossi, I. Galiana, Y. Ding, C. Walsh, Stronger, smarter, softer: next-generation wearable robots, IEEE Rob. Autom. Mag. 21 (2014) 22–33.
[6] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, Y. Sankai, Intention-based walking support for paraplegia patient with robot suitHAL, Adv. Rob. 21 (12) (2007) 1441–1469.
[7] R.J. Farris, H.A. Quintero, M. Goldfarb, Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals, IEEE Trans. Neural Syst. Rehabil. Eng. 19 (6) (2011) 652–659.
[8] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, H. van der Kooij, Design and control of the mindwalker exoskeleton, IEEE Trans. Neural Syst. Rehabil. Eng. 23 (2015) 277–286.
[9] H. Kwa, J. Noorden, M. Missel, T. Craig, J. Pratt, P. Neuhans, Development of the IHMC mobility assist exoskeleton, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2009, pp. 2556–2562.
[10] S.-H. Hyon, J. Morimoto, T. Matsubara, T. Noda, M. Kawato, XoR: hybrid drive exoskeleton robot that can balance, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2011, pp. 2715–2722.
[11] A.U. Pehlivan, D.P. Losey, M.K. O'Malley, Minimal assist-as-needed controller for upper limb robotic rehabilitation, IEEE Trans. Rob. 32 (1) (2016) 113–124.
[12] S. Banala, S.K. Agrawal, A. Fattah, V. Krishnamoorthy, W.-L. Hsu, J. Scholz, K. Rudolph, Gravity-balancing leg orthosis and its performance evaluation, IEEE Trans. Rob. 22 (6) (2006) 1228–1239.
[13] C.J. Walsh, K. Endo, H. Herr, A quasi-passive leg exoskeleton for load-carrying augmentation, Int'l J. Humanoid Rob. 4 (3) (2007) 487–506.
[14] H. Kawamoto, S. Kanbe, Y. Sankai, Power assist method for HAL-3 using EMG-based feedback controller, in: Proc. Int'l Conf. on Systems, Man and Cybernetics, 2003, pp. 1648–1653.
[15] C. Fleischer, C. Reinicke, G. Hommel, Predicting the intended motion with EMG signals for an exoskeleton orthosis controller, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2005, pp. 2029–2034.
[16] T. Kagawa, Y. Uno, Gait pattern generation for a power-assist device of paraplegic gait, in: Proc. IEEE Int'l Symposium on Robot and Human Interactive Communication, 2009, pp. 633–638.
[17] R. Ronsse, N. Vitiello, T. Lenzi, J. van den Kieboom, M. Carrozza, A. Ijspeert, Adaptive oscillators with human-in-the-loop: Proof of concept for assistance and rehabilitation, in: Proc. IEEE/RAS-EMBS Int'l Conf. on Biomedical Robotics and Biomechatronics, 2010, pp. 668–674.
[18] T. Matsubara, D. Uto, T. Noda, T. Teramae, J. Morimoto, Style-phase adaptation of human and humanoid biped walking patterns in real systems, in: Proc. IEEE/RAS Int'l Conf. on Humanoid Robots, 2014, pp. 128–133.
[19] T. Matsubara, A. Uchikata, J. Morimoto, Spatiotemporal synchronization of biped walking patterns with multiple external inputs by style–phase adaptation, Biol. Cybern. 109 (6) (2015) 597–610.
[20] T. Yan, M. Cempini, C.M. Oddo, N. Vitiello, Review of assistive strategies in powered lower-limb orthoses and exoskeletons, Rob. Auton. Syst. 64 (2015) 120–136.
[21] D. Lee, C. Ott, Y. Nakamura, Mimetic communication with impedance control for physical human-robot interaction, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2009, pp. 1535–1542.
[22] H.B. Amor, G. Neumann, S. Kamthe, O. Kroemer, J. Peters, Interaction primitives for human-robot cooperation tasks, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2014, pp. 2831–2837.
[23] H. Modares, I. Ranatunga, F.L. Lewis, D.O. Popa, Optimized assistive human-robot interaction using reinforcement learning, IEEE Trans. Cybern. 46 (3) (2016) 655–667.
[24] J.R. Medina, M. Lawitzky, A. Mortl, D. Lee, S. Hirche, An experience-driven robotic assistant acquiring human knowledge to improve haptic cooperation, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2011, pp. 2416–2422.
[25] L. Rozo, D. Bruno, S. Calinon, D.G. Caldwell, Learning optimal controllers in human-robot cooperative transportation tasks with position and force constraints, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2015, pp. 1024–1030.
[26] L. Rozo, S. Calinon, D.G. Caldwell, P. Jimenez, C. Torras, Learning physical collaborative robot behaviors from human demonstrations, IEEE Trans. Rob. 32 (3) (2016) 513–527.
[27] N. Mitsunaga, C. Smith, T. Kanda, H. Ishiguro, N. Hagita, Adapting robot behavior for human–robot interaction, IEEE Trans. Rob. 24 (4) (2008) 911–916.
[28] W. Xu, J. Huang, Y. Wang, H. Cai, Study of reinforcement learning based shared control of walking-aid robot, in: Proc. IEEE/SICE Int'l Symposium on System Integration, 2013, pp. 282–287.
[29] C. Obayashi, T. Tamei, T. Shibata, Assist-as-needed robotic trainer based on reinforcement learning and its application to dart-throwing, Neural Netw. 53 (2014) 52–60.
[30] R. Huang, H. Cheng, H. Guo, Q. Chen, H.-T. Tran, X. Lin, Interactive learning for sensitivity factors of a human-powered augmentation lower exoskeleton, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2015, pp. 6409–6415.
[31] R. Huang, H. Cheng, H. Guo, Q. Chen, X. Lin, Hierarchical interactive learning for a human-powered augmentation lower exoskeleton, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2016, pp. 257–263.
[32] M.P. Deisenroth, D. Fox, C.E. Rasmussen, Gaussian processes for data-efficient learning in robotics and control, IEEE Trans. Pattern Recognit. Mach. Intell. 37 (2015) 408–423.
[33] T. Teramae, T. Noda, S.H. Hyon, J. Morimoto, Modeling and control of a pneumatic-electric hybrid system, in: Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, 2013, pp. 4887–4892.
[34] M. Hamaya, T. Matsubara, T. Noda, T. Teramae, J. Morimoto, Learning assistive strategies from a few user-robot interactions: Model-based reinforcement learning approach, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2016, pp. 3346–3351.
[35] C. Rasmussen, C. Williams, Gaussian Processes for Machine Learning, Springer, 2006.
[36] M.P. Deisenroth, A. McHutchon, J. Hall, C.E. Rasmussen, Pilco web site, 2013, http://mlg.eng.cam.ac.uk/pilco/
[37] G. Rasool, K. Iqbal, N. Bouaynaya, G. White, Real-time task discrimination for myoelectric control employing task-specific muscle synergies, IEEE Trans. Neural Syst. Rehabil. Eng. 24 (1) (2016) 98–108.
[38] T. Cunha, P.M. Vieira, K. Costa, C.P. Santos, Looking for motor synergies in darwin-OP biped robot, in: Proc. IEEE Int'l Conf. on Robotics and Automation, 2016, pp. 1776–1781.

[39] B. Ugurlu, C. Doppmann, M. Hamaya, P. Forni, T. Teramae, T. Noda, J. Morimoto, Variable ankle stiffness improves balance control: experiments on a bipedal exoskeleton, IEEE Trans. Mechatron. 21 (1) (2016) 79–87.

[40] A.F. Mannion, P. Dolan, Electromyographic median frequency changes during isometric contraction of the back extensors to fatigue., Spine 19 (11) (1994) 1223–1229.

[41] P. Bonato, S.H. Roy, M. Knaflitz, C.J. de Luca, Time-frequency parameters of the surface myoelectric signal for assessing muscle fatigue during cyclic dynamic contractions, IEEE Trans. Biomed. Eng. 48 (7) (2001) 745–753.

[42] L. Peternel, N. Tsagarakis, D. Caldwell, A. Ajoudani, Adaptation of robot physical behavior to human fatigue in human-robot co-manipulation, in: Proc. IEEE/RAS Int'l Conf. on Humanoid Robots, 2016, pp. 489–494.