Guided Policy Search for Sequential Multi-Task Learning

Fangzhou Xiong, Biao Sun, Xu Yang,
Hong Qiao, *Senior Member, IEEE,* Kaizhu Huang, *Member, IEEE,*
Amir Hussain, *Senior Member, IEEE,* and Zhiyong Liu, *Senior Member, IEEE*

***Abstract*—Policy search in reinforcement learning (RL) is a practical approach to interact directly with environments in parameter spaces, that often deal with dilemmas of local optima and real-time sample collection. A promising algorithm, known as guided policy search (GPS), is capable of handling the challenge of training samples using trajectory-centric methods. It can also provide asymptotic local convergence guarantees. However, in its current form, the GPS algorithm cannot operate in sequential multi-task learning scenarios. This is due to its batch-style training requirement, where all training samples are collectively provided at the start of the learning process. The algorithm's adaptation is thus hindered for real-time applications, where training samples or tasks can arrive randomly. In this paper, the GPS approach is reformulated, by adapting a recently proposed, lifelong-learning method, elastic weight consolidation (EWC). Specifically, Fisher information is incorporated to impart knowledge from previously learned tasks. The proposed algorithm, termed sequential multi-task, learning-guided policy search (SMT-GPS), is able to operate in sequential multi-task learning settings, ensuring continuous policy learning, without catastrophic forgetting. Pendulum and robotic manipulation experiments demonstrate the new algorithms efficacy to learn control policies for handling sequentially-arriving training samples, delivering comparable performance to the traditional, batch-based GPS algorithm. In conclusion, the proposed algorithm is posited as a new benchmark for the real-time RL and robotics research community.**

***Index Terms*—Reinforcement learning, guided policy search, sequential multi-task learning, elastic weight consolidation.**

I. Introduction

As a core component of artificial intelligence (AI), reinforcement learning (RL) offers the robotics community, a framework and set of tools for designing sophisticated and hard-to-engineer behaviors to interact with the realistic world. In other words, it enables robots, as agents, to autonomously seek optimal behaviors through trial-and-error learning. Further, instead of explicitly deriving a solution to this unresolved problem, an objective function is usually used to describe the learning task, and its associated feedback [1]. Generally, the agent in RL attempts to maximize long-term rewards, as a specific form of the objective function, in order to acquire optimal behaviors for performing the task.

Estimation of expected long-term rewards from raw experiences obtained in the learning process [2], requires use of traditional methods such as dynamic programming and temporal-difference (TD) learning. These can address challenges of filling the complete state-action space with data [3]. However, they cannot meet requirements of high-dimensional continuous state and action spaces that are particularly encountered in the robotics domain. Policy search, a subfield of RL, has been applied in robotics applications for a wide range of tasks, such as manipulation [4], grasping [5], and locomotion [6]. This scales application of RL into high dimensional continuous action spaces, using parameterized policies, to avoid bootstrapping introduced by traditional value-function approximations. Direct policy search methods can effectively deal with high-dimensional systems, whereas complex policies, with hundreds of parameters, frequently present a challenge for such methods, requiring many samples [3]. Additionally, policy search methods need to address the problem of sample complexity resulting from high-dimensional, continuous action spaces [3]. Furthermore, despite the development of deep reinforcement learning, policy search still tends to fall into poor local optima [7].

The guided policy search (GPS) method introduces trajectory optimization to mitigate the issue of sample efficiency, for guiding policy search away from local optima. This offers significant potential for learning robotic tasks with minimal trials. The approach mainly utilizes trajectory-centric optimization to generate suitable samples, and also guides the learning process to train complex, high-dimensional policies [8], [9], [10]. Mirror descent guided policy search (MDGPS) introduced by Montgomery et al. [10], considers GPS as approximating mirror descent. It provides a total bound for

Manuscript received June 29, 2017; revised November 25, 2017. This work is supported by NSFC grants U1613213, 61375005, 61503383, 61210009, 61627808, 91648205, 61702516, 61473236, National Key Research and Development Plan of China grant 2017YFB1300202, 2016YFC0300801, MOST grants 2015BAK35B00, 2015BAK35B01, Guangdong Science and Technology Department grant 2016B090910001, and Suzhou Science and Technology Program grant SYG201712, SZS201613. This work is also supported by the Strategic Priority Research Program of the CAS (Grant XDB02080003), Key Program Special Fund in XJTLU (KSF01), and the UK Engineering and Physical Sciences Research Council (EPSRC) grant no. EP/M026981/1.

F. Xiong, X. Yang, H. Qiao, and Z. Liu are with the State Key Lab of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Science, Beijing 100190, and School of Computer and Control, University of Chinese Academy of Sciences (UCAS), Beijing 100049, China.

B. Sun and H. Qiao are with University of Science and Technology Beijing, Beijing 100083, China. H. Qiao and Z. Liu are with CAS Centre for Excellence in Brain Science and Intelligence Technology (CEBSIT), Shanghai 200031, and Cloud Computing Center, Chinese Academy of Sciences, DongGuan, GuangDong 523808, China. K. Huang is with Department of EEE, Xi'an Jiaotong-Liverpool University, Ren'ai Road No. 111, SIP 215123 Suzhou, Jiangsu Province, China. A. Hussain is with the Division of Computing Science and Maths, School of Natural Sciences, University of Stirling, Stirling FK9 4LA, Scotland, UK.

Corresponding author: Zhiyong Liu (email: zhiyong.liu@ia.ac.cn).

global policy cost and an appropriate step size to enhance global policy. Recently, Chebotar et al. [11] extended this to a global policy sampling scheme, and introduced a KL-constrained path integrals (PI^2) approach. This enhanced its generalization capability, by increasing the diversity of training data. Nevertheless, current GPS schemes can only train policies with a batch mode for different tasks, and are known to struggle with challenges of incremental data processing, particularly in robotic applications [12], [13], [14]. Specifically, GPS methods will not work if all training tasks are presented sequentially, and not collectively made available during the early training period.

GPS agents can however, learn policies from streaming data for the case of a single task. Most RL algorithms, such as Q-learning [15], [2] and Sarsa [16], work in an online mode only for one task. On the other hand, there are a number of online learning models for solving multi-tasks [17], [18], [19]. However, the GPS approach is unable to handle different tasks that are not known apriori and specified sequentially, even though it can learn policies by acquiring trajectory information online. The ability to continually learn, without catastrophic forgetting, is of significant importance to enable effective interaction with the realistic world [20]. When applied to robotic applications for example, the agent has to meet strong real-time requirements that generally present higher demands for online learning scenarios. Specifically, the robot will be required to learn skills to handle sequential tasks in real-time, and rapidly adapt to the dynamic environment.

The problem of sequential multi-task learning in GPS has also been considered part of lifelong learning [21], [22], since the agent aims to add new task knowledge, while transferring knowledge between tasks. Lifelong learning, considered a general approach to efficiently learn consecutive tasks, has been explored for reinforcement learning for some time [23]. Recently, an efficient lifelong learning algorithm for policy gradient methods has been proposed [24], which adopts a linear function to represent the policy. However, these methods are currently limited in their application, and deep neural networks are increasingly becoming more popular, particularly for robotic manipulation environments. For multi-task domains in computer vision, Li et al.[25] recently introduced deep neural networks to address the problem of continuously learning new prediction tasks, without accessing training data for previously learned tasks. However, current neural network approaches have still not been able to fully implement continual learning, and there is also inevitable catastrophic forgetting associated with this mode of learning. In an attempt to enable agents to continuously learn without catastrophic forgetting, James et al. [26] recently proposed training of networks, using an elastic weight consolidation (EWC) algorithm, that can maintain expertise from previously learnt tasks. For gradient policy learning, a deep deterministic-policy gradient (DDPG) algorithm [27] has been proposed to continuously improve policy, whilst an agent explores its environment. Compared to batch algorithms, DDPG is capable of addressing tasks for continuous control, which could be explored as a form of sequential multi-task learning.

In this paper, we reformulate the GPS method in an efficient and scalable manner, based on a sequential multi-task learning mechanism. The aim is to incrementally build a predictive model from data sequences, without catastrophic forgetting [28]. As noted earlier, current GPS approaches can only handle scenarios where data from all tasks is simultaneously made available during the early training stage, which constitutes an impractical constraint for consecutive task learning. By exploiting and adapting the recently developed EWC algorithm [26], we propose incorporation of Fisher information, to protect weights that are important for previous tasks, while learning the new task at hand. To some extent, this also overcomes catastrophic forgetting, in our proposed approach to sequential multi-task learning.

In summary, the main contribution of this paper is novel formulation of a GPS based framework, and its algorithmic implementation, termed sequential multi-task, learning-guided policy search (SMT-GPS). The proposed SMT-GPS algorithm can effectively utilize consecutive task information, enabling agents to accomplish new tasks incrementally, without forgetting those learned previously. This is demonstrated through learning control policies for two dynamical systems, specifically, upward swinging pendulum and peg insertion tasks.

The rest of the paper is organized as follows: Section II gives a brief review of background and related work. Section III presents formulation of the proposed framework, its algorithmic implementation and theoretical analysis. Comparative experimental results are presented and discussed in section IV. Finally, concluding remarks and future work suggestions are outlined in section V.

II. Background and Related Work

The agent's goal in RL is to seek a policy π to complete a specific task in an environment. At each time step t, the agent observes a state x_t and selects an action according to policy $\pi(u_t|x_t)$, producing a state transition according to dynamics $p(x_{t+1}|x_t, u_t)$.

For the policy search method, it aims to optimize a parameterized policy $\pi_\theta(u_t|x_t)$ over action u_t conditioned on the state x_t. Given stochastic dynamics $p(x_{t+1}|x_t, u_t)$ and cost function $\ell(x_t, u_t)$, the goal is to minimize the expected cost:

$$J(\theta) = \sum_{t=1}^{T} \mathbb{E}_{\pi_\theta}[\ell(x_t, u_t)] \tag{1}$$

where the notation $\pi_\theta(\tau)$ is overloaded to denote the marginals of $\pi_\theta(\tau) = p(x_1) \prod_{t=1}^{T} p(x_{t+1}|x_t, u_t)\pi_\theta(u_t|x_t)$ with a trajectory $\tau = \{x_1, u_1, ..., x_T, u_T\}$. The standard approach to policy search is computing the gradient $\nabla J(\theta)$ and using it to improve $J(\theta)$ [3].

A. Guided Policy Search

Simply put, the gist of GPS is to utilize a series of local controllers $p(u|x)$ to optimize global policy π_θ, represented by a deep neural network, that can describe a broad range of behaviors. These local controllers are used to generate guiding samples that can guide policy search to regions of high rewards. Thus, GPS can efficiently train this deep neural

Algorithm 1 MDGPS

```
for Optimizing for successful pegging do
    for position $i \in \{0, ..., M\}$ do
        C-step: $p_i \leftarrow argmin_{p_i} \mathbb{E}_{p_i(\tau)}[\sum_{t=1}^{T} \ell(x_t, u_t)]$
        $s.t.\mathrm{D}_{KL}(p_i(\tau)||\pi_\theta^{'}(\tau)) \leq \epsilon$
    end for
    S-step:
    $\pi_\theta \leftarrow argmin_\theta \sum_{t,i,j} \mathrm{D}_{KL}(\pi_{\theta(u_t|x_{t,i,j})}||p_i(u_t|x_{t,i,j}))$
    (via supervised learning)
end for
```

network with fewer samples than direct policy search [5]. The minimization of expected cost can be rewritten as the following constrained problem:

$$\min_{p,\pi_\theta} \mathbb{E}_p[\ell(\tau)] \ s.t. \ p(u_t|x_t) = \pi_\theta(u_t|x_t) \ \forall x_t, u_t, t. \tag{2}$$

A variant of GPS, the mirror decent guided-policy search (MDGPS) algorithm [7], splits global policy optimization into several local policy optimizations, in order to estimate $\nabla J(\theta)$. There are two loops in this particular algorithm, as shown in Algorithm 1. The inner loop (S-step) conducts local policy optimizations, while the outer loop (S-step) is a global policy optimizer which makes use of whole samples collected from the C-step.

During the C-step, the MDGPS algorithm uses a time-varying, linear Gaussian controller $p(u_t|x_t) \sim N(K_t x_t + k_t, C_t)$ as the local controller. The iterative linear-quadratic regulators (iLQR) algorithm is employed to calculate all terms in $p(u_t|x_t)$ at different conditions [5]. For estimation of dynamics, the MDGPS adopts a time-varying, linear Gaussian function to fit these as: $p(x_{t+1}|x_t, u_t) = N(f_{xt} + f_{ut}u_t + f_c t, F_t)$, where the Gaussian mixture model is used to estimate the dynamic model [10].

Finally, the S-step is set to optimise the global policy by introducing a deep neural network to mimic local policies generated at each condition. This converts the RL formulation into a supervised learning problem and traditional methods can be employed to optimize the global policy.

However, the MDGPS scheme requires all local policies at different conditions to support the training for global policy. In other words, the agent cannot learn continuously when conditions are given sequentially, as part of sequential multi-task learning. Hence, the learning algorithm will require reformulation in order to enable incremental task completion at a new condition, instead of starting from scratch. Specifically, there is a need to learn policies in an incremental manner, and hence avoid strict requirements of acquiring all conditions together at the initial learning stage.

B. Elastic Weight Consolidation

General artificial intelligence (AI) capabilities are known to be particularly difficult to realize in real-world settings. This is due to the requirement for agents to continuously learn and remember previously learnt tasks [29]. Nevertheless, researchers have proposed a range of methods aimed at realizing such learning capabilities. Recently, James et al. proposed a novel elastic weight consolidation (EWC) algorithm, exploiting task-specific synaptic consolidation, as a potential solution to continuous learning [26]. The EWC approach applies neural networks to adjust the learning process on certain weights, in accordance with the importance of previous tasks. A brief review of this state-of-the-art method is next presented.

Assuming there are a sequence of tasks to learn, for simplicity, we only consider two tasks A and B here. Generally speaking, the agent will employ gradient-descent based learning of parameters θ_A^*, to complete task A, after having been trained for this task only. When it comes to task B, the agent is required to train parameters θ, in order to complete both these tasks. The EWC algorithm proposes to maintain knowledge of task A, by optimizing parameters θ to remain in a region of low error for task A, centered around θ_A^*. Specifically, given the training data set $D = D_A \cup D_B$ (where D_A and D_B represent the training data for task A and B, respectively), the conditional probability $p(\theta|D)$ can be computed from the prior probability of parameters $p(\theta)$ and probability of the data $p(D|\theta)$ (by applying Bayes rule and the equation $D = D_A \cup D_B$):

$$\log p(\theta|D) = \log p(D_B|\theta) + \log p(\theta|D_A) - \log p(D_B). \tag{3}$$

According to [26], the objective function $\ell(\theta)$ in EWC aims to minimize

$$\ell(\theta) = \ell_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta_{A,i}^*)^2. \tag{4}$$

where the loss-function $\ell(\theta)$ represents the negative of log-probability of data, given the parameters (that is, $-\log p(D|\theta)$); $\ell_B(\theta)$ denotes the negative log-likelihood, or loss-function of task B; the Fisher information matrix F carries information about task A; i represents sets for each neural network parameter; and λ quantifies relative importance between the old task A and new task B.

Once the EWC has learned appropriate parameters, θ for solving these two tasks, it moves to a third task C. Consequently, equation (4) is used again to learn new parameters, both to complete this new task C, and also keep neural network parameters close to the learned parameters for completed tasks A and B.

III. The Proposed Method

A. A GPS based Framework for Sequential Multi-Task Learning

In this section, we propose a reformulation of the conventional GPS algorithm based on a modified EWC mechanism, and present a general framework for sequential multi-task learning.

The basic GPS setting aims to study incremental learning of policies for solving a task at different conditions, that are provided sequentially instead of being provided together. The traditional GPS fails to work in the former setting, since it needs to learn global policies with all task conditions. This fundamental limitation of the GPS can be attributed to the structure of interaction between global policy (S-step in

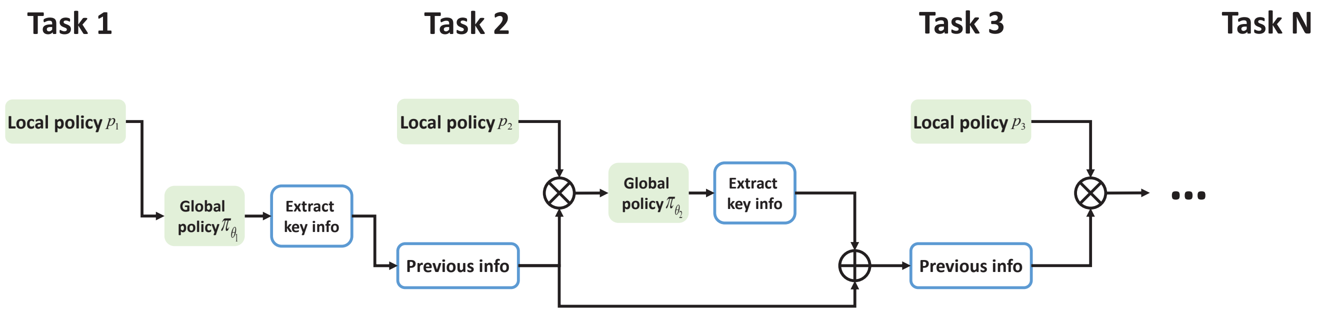


Fig. 1: The proposed GPS based framework for sequential multi-task learning.

Algorithm 1) and local policies (C-step in Algorithm 1), which pre-limits the sequential multi-task learning mode. In other words, in order to learn tasks incrementally, the agent should be able to separate mutual effects between global and local policy optimizations when encountering a new task condition, failing which it would affect both the S-step and C-step outcomes. An online approach will require the global policy to be learned asynchronously from each single local policy. Therefore, in contract to the traditional GPS algorithm, we propose optimize a single local policy directly in the C-step as follows:

$$p \leftarrow \underset{p}{\operatorname{argmin}}\, \mathbb{E}_p[\sum_{t=1}^{T} \ell(x_t, u_t)], \tag{5}$$

This breaks the limited relationship between the global policy and local policies, where optimization for local policies will not be influenced by the global policy. Nevertheless, the traditional GPS can quickly and efficiently train global policy that benefits from local policies at different task conditions. As a compensation mechanism for the absence of these interactions, instead of optimizing local policies with a fixed number in [10], we propose to optimize local policies continuously until they can complete the task at the current task condition. Thus, we only select "successful samples" generated by these trajectories that complete the task to execute the next step. Global policy optimization is readily carried out with these "successful samples" generated by local polices. Additionally, in order to learn a task continuously at different task conditions, the global policy needs to remember all previously learned policies and generalize to complete the new task, in an incremental manner.

Figure 1 illustrates our proposed GPS based framework for sequential multi-task learning. The local policy p_i is generally optimized with iterative linear-quadratic regulators (iLQR) [5] or the path integrals (PI^2) method [11]. The global policy π_{θ_i} usually adopts a deep neural network to represent a broad range of behaviors. Further, the current task information is evaluated by an information extraction approach, such as the Fisher information matrix. This records the second derivative of the loss near a minimum, with the guarantee of positive semidefiniteness. The parameters $\oplus$ and $\otimes$ represent methods of feature information fusion and incremental learning (such as the EWC algorithm), respectively.

Our proposed formulation enables global policy to perform a task at the current task condition, and at the same time, remember previously learned tasks without catastrophic forgetting. Next, we present an algorithmic implementation of our proposed framework for sequential multi-task learning.

B. *A new algorithmic implementation for sequential multi-task learning*

In this subsection, a sequential multi-task, learning-guided policy search (SMT-GPS) algorithm is proposed to tackle the problem of tight coupling between the global and local policies. Specifically, a modified EWC algorithm is developed to combine previous results with current information, by employing a Fisher information matrix to impart knowledge of previously learned tasks.

Algorithm 2 summarizes our proposed method. Initially, the previous policy p is set to null, owing to the absence of a previous task. In the inner loop (lines 3-7), the agent aims to learn local policies individually at a given task condition, where an iLQR algorithm is utilized to fit dynamics and optimize local policies. As for the outer loop (lines 8-9), the agent applies local policies collected from the inner loop, to optimize global policy. This employs a variant of the EWC algorithm to complete optimization under different task conditions. More precisely, we use those trajectories that can complete the task, to generate "successful samples" and employ the sample set D_m to optimize global policy. This is totally different from the traditional GPS scenario, where all samples are collected to carry out optimization. Further, the ability to continuously learn at different task conditions is realized by this modified EWC algorithm (corresponding to operation $\otimes$ in Figure 1), where different Fisher information matrixes are fused, with a sum operation over task conditions encountered to-date (corresponding to operation $\oplus$ in Figure 1). In particular, a variable weight parameter λ_i is introduced to measure importance of different task conditions, which is different from the constant parameter λ employed in primary EWC settings. Next, we utilize this modified EWC algorithm to formulate the following optimization problem:

$$\begin{aligned}\pi_\theta \leftarrow \operatorname*{argmin}_\theta \sum_{t=1}^{T} \mathrm{D}_{KL}(\pi_\theta(u_t|x_{t,m})||p_m(u_t|x_{t,m})) \\ + \frac{\lambda_i}{2}\sum_{i=1}^{m}(\theta-\theta_i)^T F_i(\theta-\theta_i).\end{aligned} \tag{6}$$

According to (6), we train a neural network with gradient descent learning, to optimize global policy for the agent to learn different tasks continuously. This equates to learning sequential multi-tasks without catastrophic forgetting, and constitutes a novel algorithmic implementation of our proposed framework.

Algorithm 2 SMT-GPS

```
Initialize: $p \leftarrow null$
for condition $m = 1$ to $M$ do
    for iteration $k \in 1, \ldots, K$ do
        Generate samples $D_i = \tau_{i,j}$ by running $p_i$
        Fit linear-Gaussian dynamics $p_i(x_{t+1}|x_t, u_t)$ using samples in $D_i$
        Optimize local policy:
            $p \leftarrow \operatorname{argmin}_p \mathbb{E}_p[\sum_{t=1}^{T} \ell(x_t, u_t)]$
    end for
    Collect “successful samples” by running $p_i$, and record as $D_m$
    Optimize global policy with $D_m$:
    $\pi_\theta \leftarrow \operatorname{argmin}_\theta \sum_t \mathrm{D}_{KL}(\pi_\theta(u_t|x_{t,m})||p_m(u_t|x_{t,m})) + \frac{\lambda_i}{2}\sum_{i=1}^{m}(\theta-\theta_i)^T F_i(\theta-\theta_i)$
end for
```

C. Theoretical Analysis

In this section, we present a theoretical analysis of global policy cost and provide its upper bound. It is shown that the global policy limitlessly approaches local policies that have already completed tasks. Equivalently, the global policy is able to complete tasks at different task conditions.

Without loss of generality, given task conditions for training, we assume that the probabilities of $\pi_\theta(x)$ and $p(x)$ follow different distributions, yet both are bounded as $0 < \alpha < \pi_\theta(x), p(x) < \beta < 1$. Further, each weight parameter λ_i satisfies the constraint $0 < \lambda_i < 1$.

1) The State Distribution Difference:

Given $\epsilon_t = \max_{x_t} \mathrm{D}_{KL}(\pi_\theta(u_t|x_t)||p(u_t|x_t))$, the state distribution difference satisfies: $||\pi_\theta(x_t) - p(x_t)||_1 \leq \epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}$.

According to [10], [30], we can express the state distribution $p(x_t)$ as:

$$p(x_t) = [\prod_{t'=1}^{t}(1-\sqrt{2\epsilon_{t'}})][\pi_\theta(x_t) - \widetilde{p}(x_t)] + \widetilde{p}(x_t), \tag{7}$$

where $\widetilde{p}(x_t)$ is some other distribution, and the probability $1-\sqrt{2\epsilon_{t'}}$ implies that $p(x_t)$ and $\pi_\theta(x)$ take the same action at time step t.

Applying a second-order Taylor series in KL divergence, with an assumption of $\Delta\theta \to 0$

$$\begin{aligned}\mathrm{D}_{KL}(q_\theta(z)||q_{\theta+\Delta\theta}(z)) &\approx (\mathbb{E}_z[\log q_\theta(z)] - \mathbb{E}_z[\log q_\theta(z)]) \\ &\quad - \mathbb{E}_z[\nabla \log q_\theta(z)]\Delta\theta \\ &\quad - \frac{1}{2}\Delta\theta^T \mathbb{E}_z[\nabla^2 \log q_\theta(z)]\Delta\theta \\ &= \frac{1}{2}\Delta\theta^T \mathbb{E}_z[-\nabla^2 \log q_\theta(z)]\Delta\theta \\ &= \frac{1}{2}\Delta\theta^T F \Delta\theta,\end{aligned} \tag{8}$$

where $F = \mathbb{E}_z[-\nabla^2 \log q_\theta(z)]$. The proof for this approximation can be found in [31]. A previously learned global policy represented by $\pi^{'}_{\Theta}$, can be readily used to represent previous local policies at different task conditions, specifically, for each policy p_i, $p_i = \pi^{'}_{\Theta}$. Following optimization of a global policy by applying (6), we can obtain a new global policy π_θ situated in the neighborhood of $\pi^{'}_{\Theta}$, represented as $\pi_\theta = \pi^{'}_{\Theta-\Delta\Theta}$. Thus at each task condition i, by substituting $\theta = \Theta - \Delta\Theta$ and applying $\Delta\theta = \Delta\Theta \to 0$, we can rationally derive the following:

$$\begin{aligned}\mathrm{D}_{KL}(\pi_\theta(x_{t,i})||p(x_{t,i})) &= \mathrm{D}_{KL}(\pi^{'}_{\Theta-\Delta\Theta}(x_{t,i})||\pi^{'}_{\Theta}(x_{t,i})) \\ &= \mathrm{D}_{KL}(\pi^{'}_{\theta}(x_{t,i})||\pi^{'}_{\theta+\Delta\Theta}(x_{t,i})) \\ &= \mathrm{D}_{KL}(\pi^{'}_{\theta}(x_{t,i})||\pi^{'}_{\theta+\Delta\theta}(x_{t,i})) \\ &= \frac{1}{2}\Delta\theta^T F \Delta\theta \\ &= \frac{1}{2}(\theta-\theta_i)^T F_i(\theta-\theta_i),\end{aligned} \tag{9}$$

The above corresponds to the second term of the optimization problem in (6). Thus, the optimization problem for global policy can be rewritten as:

$$\begin{aligned}\pi_\theta \leftarrow \operatorname*{argmin}_\theta \sum_t \mathrm{D}_{KL}(\pi_\theta(u_t|x_{t,m})||p_m(u_t|x_{t,m})) \\ + \lambda_i \sum_i \mathrm{D}_{KL}(\pi_\theta(x_{t,i})||p(x_{t,i})).\end{aligned} \tag{10}$$

Consequently, the state distribution difference $Dis = ||\pi_\theta(x_t) - p(x_t)||_1$ at time step t and task condition m can be expressed as follows:

$$
\begin{aligned}
Dis &= || \, \mathrm{D}_{KL}(\pi_\theta(u_t|x_t)||p(u_t|x_t)) \\
&\quad + \lambda_i \sum_{i=1}^{m} \mathrm{D}_{KL}(\pi_\theta(x_{t,i})||p(x_{t,i})) - p(x_t)||_1 \\
&\le ||\epsilon_t + \lambda_i \sum_{i=1}^{m} \mathrm{D}_{KL}(\pi_\theta(x_{t,i})||p(x_{t,i})) - p(x_t)||_1 \\
&\le ||\epsilon_t + \mathrm{D}_{KL}(\pi_\theta(x_t)||p(x_t)) - p(x_t)||_1 \\
&\le ||\epsilon_t + \mathrm{D}_{\chi^2}(\pi_\theta(x_t)||p(x_t)) - p(x_t)||_1 \\
&\le ||\epsilon_t + \frac{(\pi_\theta(x_t) - p(x_t))^2}{p(x_t)} - p(x_t)||_1 \\
&= ||\epsilon_t + \frac{2\pi_\theta(x_t)[\pi_\theta(x_t) - \widetilde{p}(x_t)]}{p(x_t)} \\
&\quad \cdot [1 - \prod_{t'=1}^{t}(1 - \sqrt{2\epsilon_{t'}})] - \frac{\pi_\theta(x_t)^2}{p(x_t)}||_1 \\
&\le ||\epsilon_t + \frac{4[1 - \prod_{t'=1}^{t}(1 - \sqrt{2\epsilon_{t'}})]}{p(x_t)}||_1,
\end{aligned} \tag{11}
$$

where the second step follows from the definition $\epsilon_t = \max_{x_t} \mathrm{D}_{KL}(\pi_\theta(u_t|x_t)||p(u_t|x_t))$, the third step follows $\mathrm{D}_{KL}(\pi_\theta(x_t)||p(x_t)) = \max_{x_{t,i}} \mathrm{D}_{KL}(\pi_\theta(x_{t,i})||p(x_{t,i}))$ and $\sum_{i=1}^{m} \lambda_i = 1$, the fourth and fifth steps follow from the conclusion $\mathrm{D}_{KL}(p(x)||q(x)) \le \mathrm{D}_{\chi^2}(p(x)||q(x))$ (presented in [32]), the sixth step follows (7), and the last inequality comes from the fact that $0 < \pi_\theta(x) < 1$ and $||\pi_\theta(x_t) - \widetilde{p}(x_t)|| \le 2$ for discrete distributions. For the continuous case, the result can be obtained through the limit of an infinitely fine discretization. Next, it is noted that

$$
\prod_{t'=1}^{t}(1 - \sqrt{2\epsilon_{t'}}) \ge 1 - \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}, \tag{12}
$$

so we can have:

$$
||\pi_\theta(x_t) - p(x_t)||_1 \le ||\epsilon_t + \frac{4\prod_{t'=1}^{t}\sqrt{2\epsilon_{t'}}}{p(x_t)}||_1. \tag{13}
$$

Given the lower bound α for $p(x_t)$, we can obtain the upper bound for $||\pi_\theta(x_t) - p(x_t)||_1$ which is

$$
||\pi_\theta(x_t) - p(x_t)||_1 \le \epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}. \tag{14}
$$

2) The Global Policy Cost:

For the state-distribution difference $||\pi_\theta(x_t) - p(x_t)||_1 \le \epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}$, we can set a bound for the global policy cost as follows:

$$
\begin{aligned}
\sum_{t=1}^{T} \mathbb{E}_{\pi_\theta(x_t,u_t)}[l(x_t,u_t)] \le \sum_{t=1}^{T} [&\mathbb{E}_{p(x_t,u_t)}[l(x_t,u_t)] \\
&+ [\epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}]L(x_t,u_t) \\
&+ \sqrt{2\epsilon_t}L(x_t,u_t)],
\end{aligned} \tag{15}
$$

where $L(x_t,u_t) = \max_{x_t,u_t} l(x_t,u_t)$.

In the first step, we specify a bound on the cost of global policy at time step t according to:

$$
\begin{aligned}
\mathbb{E}_{\pi_\theta(x_t,u_t)}[l(x_t,u_t)] &= \langle \pi_\theta(x_t,u_t), l(x_t,u_t) \rangle \\
&= \langle \pi_\theta(x_t,u_t) - p(x_t)\pi_\theta(u_t|x_t), \\
&\quad l(x_t,u_t) \rangle \\
&\quad + \langle p(x_t)\pi_\theta(u_t|x_t), l(x_t,u_t) \rangle \\
&= \langle \pi_\theta(u_t|x_t)[\pi_\theta(x_t) - p(x_t)], l(x_t,u_t) \rangle \\
&\quad + \langle p(x_t)[\pi_\theta(u_t|x_t) - p(u_t|x_t)], \\
&\quad l(x_t,u_t) \rangle \\
&\quad + \mathbb{E}_{p(x_t,u_t)}[l(x_t,u_t)] \\
&\le ||\pi_\theta(x_t) - p(x_t)||_1 L(x_t,u_t) \\
&\quad + ||\pi_\theta(u_t|x_t) - p(u_t|x_t)||_1 L(x_t,u_t) \\
&\quad + \mathbb{E}_{p(x_t,u_t)}[l(x_t,u_t)] \\
&\le \mathbb{E}_{p(x_t,u_t)}[l(x_t,u_t)] \\
&\quad + [\epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}]L(x_t,u_t) \\
&\quad + \sqrt{2\epsilon_t}L(x_t,u_t),
\end{aligned} \tag{16}
$$

where $L(x_t,u_t) = \max_{x_t,u_t} l(x_t,u_t)$, and the proof for $\max_{x_t} ||\pi_\theta(u_t|x_t) - p(u_t|x_t)||_1 \le \sqrt{2\epsilon_t}$ was presented in [30].

Next, summing the above quantity over all time t, we get:

$$
\begin{aligned}
\sum_{t=1}^{T} \mathbb{E}_{\pi_\theta(x_t,u_t)}[l(x_t,u_t)] \le \sum_{t=1}^{T} [&\mathbb{E}_{p(x_t,u_t)}[l(x_t,u_t)] \\
&+ [\epsilon_t + 4\alpha \prod_{t'=1}^{t} \sqrt{2\epsilon_{t'}}]L(x_t,u_t) \\
&+ \sqrt{2\epsilon_t}L(x_t,u_t)].
\end{aligned} \tag{17}
$$

This bound on the cost of global policy illustrates that for the case of low cost local policies, we will eventually reduce the cost of global policy $\pi_\theta(u_t|x_t)$. In our setting, local policies adopted in (6) ensure they are capable of performing tasks by being trained on “successful samples”, or equivalently, the cost for local policy is kept particularly small. Noting $\epsilon_t \le \epsilon$ in the C-step of Algorithm 1, and by choosing a small enough ϵ, we can keep the difference between global and local polices arbitrarily small so as to learn sequential multiple tasks without catastrophic forgetting.

IV. Simulation Illustration

In this section, we employ the proposed SMT-GPS algorithm to learn control policies for two dynamical systems shown in Figure 2 and Figure 3, specifically, a pendulum swinging upwards, and a peg insertion environment. By generating multiple tasks, through varying the initial position of each system (illustrated in Table I), a series of tasks are used to evaluate the algorithm. First, a pendulum experiment is carried out, to demonstrate the feasibility of the SMT-GPS approach to continue learning without catastrophic forgetting, in contrast to a conventional RL algorithm. Next, a robot manipulation experiment is conducted to explicitly illustrate the ability of sequential multi-task learning in SMT-GPS, in comparison with a traditional GPS based method.

TABLE I: System parameters used for different tasks.

	Task1	Task2	Task3	Task4
Pendulum (θ)	0.55π	0.7π	0.85π	π
Peg Insertion ([x,y])	$[-0.12, -0.12]$	$[-0.12, 0.08]$	$[0.12, 0.08]$	$[0.12, -0.12]$

TABLE II: Results of DDPG and SMT-GPS.

Testing Task No.	DDPG		SMT-GPS	
	Average Loss (l)	Final Position (θ)	Average Loss (l)	Final Position (θ)
1	-267.7	0.0257	-244.8	0.0035
2	-440.1	0.0251	-401.6	0.0047
3	-639.4	0.0246	-579.0	0.0052
4	-866.2	0.0243	-777.1	0.0058

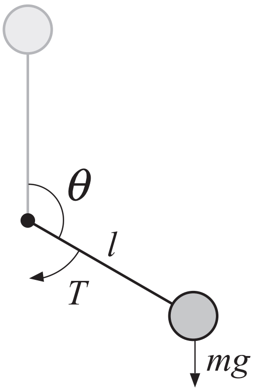


Fig. 2: Pendulum.

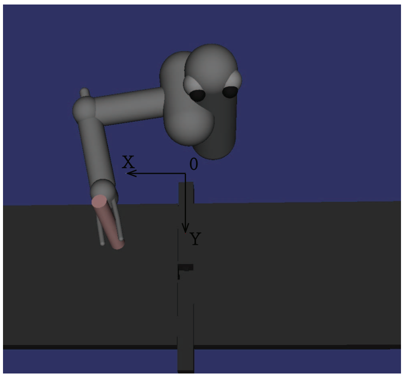


Fig. 3: Peg Insertion.

A. Pendulum control

1) Dynamical system:

The controller of the pendulum aims to swing the pendulum several times to build up momentum to make the pendulum upright. It also needs to decelerate the pendulum early enough to prevent it from falling over. If the maximal load torque mgl is greater than the maximal output torque u^{max}, a non-trivial solution results for this one degree of freedom system. The state comprises angles and velocities relative to the target position. The goal is to study a policy for controlling the pendulum swinging upwards. For each task, if the final position of the pendulum is close to upright position ($\theta < 0.1\pi$), the task is considered to be successful. The cost function at state x_t, for action u_t, is given by:

$$\ell(x_t, u_t) = \frac{1}{2}w_u||u_t||^2 + \frac{1}{2}w_x||p_{x_t} - p^*||^2 + \frac{1}{2}w_v||v_{x_t} - v^*||, \tag{18}$$

where p_{x_t} and v_{x_t} are the position and velocity of pendulum at state x_t respectively, p^* and v^* represent the target information for position and velocity of the pendulum, and w_u, w_x and w_v are weighting parameters. This cost function encourages low energy actions for target pendulum positions.

2) Results and Discussion:

In the section, we employ the dynamical system to evaluate our proposed SMT-GPS algorithm. It is benchmarked against a well-known RL algorithm, termed deep deterministic policy gradient (DDPG), which continuously improves the policy by training a deterministic policy. For each task, we execute 100 steps to generate a trajectory sample, and collect 10 samples during each session (corresponding to one iteration in Algorithm 2). The SMT-GPS and DDPG algorithms are implemented to optimize policy for the sequentially specified tasks. When an agent completes learning at task i, experiments are evaluated a total of fifty times for tasks $1, 2, \cdots, i$.

Table II shows that both algorithms are capable of completing previously learned tasks $1, 2, \cdots, i$, when learning a new task i. This is evidenced by the final position of the pendulum being close to the target upright position ($\theta \approx 0$). Further, this indicates the proposed algorithm has some ability to overcome catastrophic forgetting.

To compare the two algorithms in more detail, we investigate the average loss accumulated in 100 executing steps, which indirectly describes the final state of pendulum. As seen in Table II, after training at all 4 tasks, SMT-GPS achieves similar results to DDPG when testing at those 4 tasks. However, the proposed SMT-GPS can finish the task sequentially, and delivers less loss than DDPG, both in terms of loss function and final position of the pendulum. More importantly, SMT-GPS only utilizes samples generated at the current task condition in order to learn the control policy. On the other hand, the DDPG requires samples in different tasks to be randomly presented to train policy. Since previous task samples need to be collected to update policy, when faced with a new task, the DDPG places a higher demand, both in the manner tasks appear and space samples are stored.

B. Peg Insertion

1) Dynamical system:

This robot manipulation experiment requires controlling a seven degree of freedom 3D simulated arm with the MuJoCo simulation environment [33], to insert a tight-fitting peg into a hole. The state consists of joint angles, velocities, and end-effector positions relative to the target position. For each task, if the distance d between the current state and goal position is smaller than a baseline 0.06 (as shown in Figure 4), the task is considered to be successful. The cost function presented in [5] is:

$$\ell(x_t, u_t) = \frac{1}{2} w_u ||u_t||^2 + w_p \ell_{12}(p_{x_t} - p^*), \tag{19}$$

where u_t is the robot action, p_{x_t} is the position of end effector for state x_t, p^* is the desired end effector position, and the norm $\ell_{12}(z)$ is calculated by $\frac{1}{2}||z||^2 + \sqrt{\gamma + z^2}$ which corresponds to the sum of an ℓ_2 and ℓ_1 norm. This cost function comprises two terms, the first weighted by w_u to encourage low energy actions and the other weighted by w_p to enable the peg to reach target hole precisely.

In this section, the SMT-GPS and MDGPS are employed to conduct comparative experiments. For the SMT-GPS, the global policy for each task is represented by a fully connected neural network, with the structure $[26-100-100-7]$. In each manipulation task, only "successful samples" that represent successful trajectories for completing the task are collected, to train the neural network global policy. As for the MDGPS, environment settings described in [10] are employed.

Specifically, for each task, a trajectory sample is generated for 100 steps, and 5 samples collected during each iteration. The SMT-GPS and MDGPS algorithms are applied to optimize the policy iteratively. When the agent is learning task i, the experiment is evaluated at previously visited tasks $1, 2, \cdots, i$.

A further three sub-experiments are carried out to evaluate the efficacy of the proposed SMT-GPS method. The first experiment aims to demonstrate the effectiveness of the proposed method, by comparing with the same neural network, but without employing Fisher information. Subsequently, a comparative experiment of SMT-GPS with MDGPS is designed to demonstrate the formers sequential multi-task learning capability. Finally, we utilize Fisher information to carry out a concrete analysis of the comparative efficacy of the proposed SMT-GPS. Comparative results are presented and discussed in the next section.

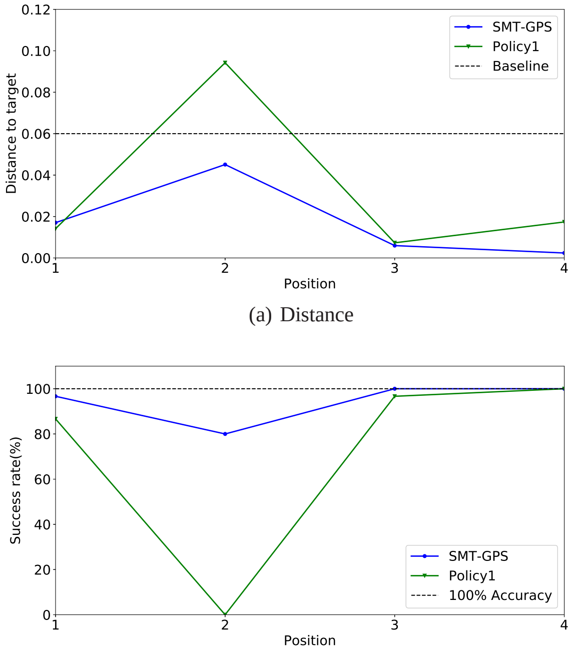
2) *Results and Discussion:*

a) *Overcoming catastrophic forgetting:*

In this experiment, in order to analyze algorithms from the perspective of storing previous information, a new policy (denoted "Policy1") is constructed within the same neural network framework. However, Fisher information from previous tasks is not exploited here, whilst the agent learns a new task. In other words, the training for "Policy1" only depends on the current task information, with the exception of neural network parameters inherited directly from training previous tasks.

First, we evaluate the SMT-GPS and "Policy1" at 200 positions randomly selected around the 4 initial tasks. Experimental results presented in Figure 4 illustrate that once the agent has learned the control policy for 4 tasks, the SMT-GPS can almost complete peg insertion at all test tasks, both in terms of the distance to target and success rate. However, "Policy1" fails in some areas around the training tasks.

Results show that the SMT-GPS algorithm outperforms "Policy1" for completing the insertion task. This is due to use of the EWC algorithm to optimize the SMT-GPS approach. However, "Policy1" is capable of self-optimizing only at the current task without taking Fisher information of previous tasks into account. Specifically, "Policy1" lacks the second term on the right-hand of equation (6). Thus, equipped with the EWC algorithm which utilizes previous task information, the proposed SMT-GPS is able to complete different sequentially presented tasks in this experiment. Further, it can exploit previously learned information without catastrophic forgetting, that is, it has the ability to learn knowledge continuously.



(a) Distance

(b) Accuracy

Fig. 4: The result of comparing SMT-GPS and "Policy1".

b) *Sequential multi-task learning capability:*

In this section, experiments are carried out using the proposed SMT-GPS approach and the MDGPS algorithm, in order to evaluate the comparative effectiveness of their multi-task learning capabilities.

TABLE III: Results of comparing SMT-GPS and MDGPS

Algorithm	Distance to Target	Average Cost	Success Rate
SMT-GPS	0.009096	-584.2872	0.9667
MDGPS	0.007466	-587.5530	0.9833

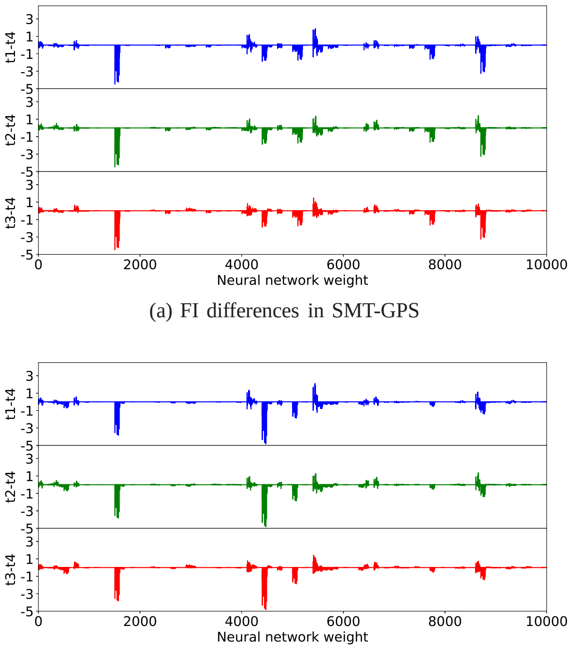
In the MDGPS algorithm, all samples of different tasks are presented together at the beginning of training, in order to analyse policies in a batch way. For SMT-GPS, the agent learns policies based only on current task samples, after it completes previous tasks. We test these two algorithms on a total of 120 different tasks, with initial positions randomly selected within the square area constructed by associated training tasks. In other words, both algorithms are evaluated at 30 similar yet different tasks separately, generated around 4 different training tasks.

Results are presented in Table III which show that the proposed SMT-GPS algorithm can attain comparable performance to MDGPS in three aspects, including distance between end-effector position and target position, the action cost and success rate of peg insertion. However, the SMT-GPS only relies on current task samples, which is totally different from the MDGPS whose training samples for all tasks need to be presented in advance. In other words, the proposed SMT-GPS can be seen to complete multiple tasks sequentially without requiring whole task information, and can also achieve better results at the neighborhood of these 4 tasks.

c) Fisher information analysis:

Finally, we carry out an experiment to further analyze the concrete influence of retaining previous information, with a form of Fisher information (FI) incorporated in the EWC algorithm.

As before, we use the "Policy1" method as a contrastive method, and train policies with the same settings as in the first subsection. Since FI describes the accuracy of estimated posterior probability for each task parameter, we now make a comparison between different tasks in terms of their FI values.



(a) FI differences in SMT-GPS

(b) FI differences in "Policy1"

Fig. 5: The result about FI difference on each weight in layer 3 for comparing SMT-GPS and "Policy1" methods.

For simplicity, the neural network weights in the third layer (i.e. the second hidden layer) are used for illustration. The outcomes are calculated in three scenarios, showing FI differences between each of the first three tasks with the fourth task, corresponding to the top ($t1 - t4$), middle ($t2 - t4$) and bottom ($t3 - t4$) sub-figures in Figure 5, respectively. As can be seen, the FI differences in 3 sub-figures achieve similar results since the agent can complete each of the previous tasks when learning the current task. Here, FI is employed to record important information of previous tasks, which can be considered a way of communicating information between different learning tasks. It can also be seen that, for the case of the SMT-GPS algorithm, FI differences present smaller values compared to"Policy1". For instance, there are significant differences around the weight 4500, which show that "Policy1" is not capable of learning a perfect global parameter to represent the previous parameters for each task. Therefore, we conclude that the proposed SMT-GPS method is able to retain key weights for previously learned tasks. In other words, it can recall previous task information to avoid catastrophic forgetting, whilst executing a new learning task.

V. Conclusions

In this paper, we proposed a novel GPS based framework for sequential multi-task learning. It enables agents to continuously learn policies for different tasks, without catastrophic forgetting. In particular, an algorithmic implementation, termed SMT-GPS, has been realized, and comparatively evaluated on two dynamical systems, specifically, an upward-swinging pendulum and peg insertion environments. These demonstrate the algorithms ability to both remember previous task policies and incrementally learn, new task-specific knowledge. Use of well-trained local policies, optimised by "successful samples representing successful completion of trajectories, enable the SMT-GPS to address the problem of catastrophic forgetting. The latter is of significant importance, for enabling effective interaction with the real world.

Further, the agents global policy employs a modified EWC algorithm to perform self-optimization at different task conditions. Here, Fisher information is introduced to represent parameters for previous tasks. Thus, the agent can generate a successful policy for completing all encountered tasks. In contrast to traditional batch algorithms employed in RL, such as GPS, the proposed SMT-GPS is capable of learning policies incrementally, without requiring all learning tasks to be presented in advance. The new algorithm is thus posited as a new benchmark method, for the real-time RL and robotics research community.

For future work, the proposed framework can be extended by introducing deep neural networks, to effectively deal with visual inputs. This could enable agents to complete tasks and learn continuously in more complex environments. Further, exploring other learning models for the SMT-GPS, such as learning to reach different target positions for the same task setting, is another challenging future work direction.

Acknowledgments

This work is supported by NSFC grants U1613213, 61375005, 61503383, 61210009, 61627808, 91648205, 61702516, 61473236, National Key Research and Development Plan of China grant 2017YFB1300202, 2016YFC0300801, MOST grants 2015BAK35B00, 2015BAK35B01, Guangdong Science and Technology Department grant 2016B090910001, and Suzhou Science and Technology Program grant SYG201712, SZS201613. The work is also supported by the Strategic Priority Research Program of the CAS (Grant XDB02080003), Key Program Special Fund in XJTLU (KSF01), and the UK Engineering and Physical Sciences Research Council (EPSRC) grant no. EP/M026981/1. The authors are grateful to the anonymous reviewers for their insightful comments and suggestions, which helped improve the quality of this paper.

REFERENCES

[1] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, pp. 1238–1274, Aug. 2013.

[2] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, vol. 1. MIT press Cambridge, 1998.

[3] M. P. Deisenroth, G. Neumann, J. Peters, *et al.*, "A survey on policy search for robotics," *Foundations and Trends® in Robotics*, vol. 2, no. 1–2, pp. 1–142, 2013.

[4] M. Kalakrishnan, L. Righetti, P. Pastor, and S. Schaal, "Learning force control policies for compliant manipulation," in *International Conference on Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ*, pp. 4639–4644, IEEE, Sep. 2011.

[5] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *Journal of Machine Learning Research*, vol. 17, no. 39, pp. 1–40, 2016.

[6] G. Endo, J. Morimoto, T. Matsubara, J. Nakanishi, and G. Cheng, "Learning cpg-based biped locomotion with a policy gradient method: Application to a humanoid robot," *The International Journal of Robotics Research*, vol. 27, pp. 213–228, Feb. 2008.

[7] S. Levine and V. Koltun, "Guided policy search," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 1–9, Jun. 2013.

[8] S. Levine and V. Koltun, "Variational policy search via trajectory optimization," in *Advances in Neural Information Processing Systems*, pp. 207–215, 2013.

[9] S. Levine and P. Abbeel, "Learning neural network policies with guided policy search under unknown dynamics," in *Advances in Neural Information Processing Systems*, pp. 1071–1079, 2014.

[10] W. H. Montgomery and S. Levine, "Guided policy search via approximate mirror descent," in *Advances in Neural Information Processing Systems*, pp. 4008–4016, 2016.

[11] Y. Chebotar, M. Kalakrishnan, A. Yahya, A. Li, S. Schaal, and S. Levine, "Path integral guided policy search," in *Robotics and Automation (I-CRA), 2017 IEEE International Conference on*, pp. 3381–3388, IEEE, Jul. 2017.

[12] S. Adam, L. Busoniu, and R. Babuska, "Experience replay for real-time reinforcement learning control," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 2, pp. 201–212, 2012.

[13] M. J. Powell and A. D. Ames, "Towards real-time parameter optimization for feasible nonlinear control with applications to robot locomotion," in *American Control Conference (ACC), 2016*, pp. 3922–3927, IEEE, 2016.

[14] M. Li, Y. Li, S. S. Ge, and T. H. Lee, "Adaptive control of robotic manipulators with unified motion constraints," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 1, pp. 184–194, 2017.

[15] C. J. C. H. Watkins, *Learning from delayed rewards*. PhD thesis, King's College, Cambridge, 1989.

[16] S. P. Singh and R. S. Sutton, "Reinforcement learning with replacing eligibility traces," *Recent Advances in Reinforcement Learning*, pp. 123–158, 1996.

[17] F. Fernández and M. Veloso, "Probabilistic policy reuse in a reinforcement learning agent," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pp. 720–727, ACM, 2006.

[18] H. B. Ammar, E. Eaton, P. Ruvolo, and M. Taylor, "Online multi-task learning for policy gradient methods," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1206–1214, Jun. 2014.

[19] P. Yang, K. Huang, and C.-L. Liu, "Geometry preserving multi-task metric learning," *Machine learning*, vol. 92, no. 1, pp. 133–175, 2013.

[20] A. Saha, P. Rai, H. DaumÃ, S. Venkatasubramanian, *et al.*, "Online learning of multiple tasks and their relationships," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 643–651, 2011.

[21] S. Thrun, "Lifelong learning algorithms," *Learning to learn*, vol. 8, pp. 181–209, 1998.

[22] W. Hao, J. Fan, Z. Zhang, and G. Zhu, "End-to-end lifelong learning: a framework to achieve plasticities of both the feature and classifier constructions," *Cognitive Computation*, pp. 1–13, 2017.

[23] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, and T. M. Mitchell, "Toward an architecture for never-ending language learning," in *Association for the Advance of Artificial Intelligence*, vol. 5, p. 3, 2010.

[24] P. Ruvolo and E. Eaton, "Ella: An efficient lifelong learning algorithm," in *International Conference on Machine Learning*, pp. 507–515, 2013.

[25] Z. Li and D. Hoiem, "Learning without forgetting," in *European Conference on Computer Vision*, pp. 614–629, Springer, 2016.

[26] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, p. 201611835, 2017.

[27] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[28] J. Lu, S. C. Hoi, J. Wang, P. Zhao, and Z.-Y. Liu, "Large scale online kernel learning," *Journal of Machine Learning Research*, vol. 17, no. 47, p. 1, 2016.

[29] S. Legg and M. Hutter, "Universal intelligence: A definition of machine intelligence," *Minds and Machines*, vol. 17, no. 4, pp. 391–444, 2007.

[30] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 1889–1897, 2015.

[31] R. Pascanu and Y. Bengio, "Revisiting natural gradient for deep networks," *arXiv preprint arXiv:1301.3584*, 2013.

[32] A. Sayyareh, "A new upper bound for kullback-leibler divergence," *Applied Mathematical Sciences*, vol. 67, pp. 3303–3317, 2011.

[33] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pp. 5026–5033, IEEE, 2012.

Fangzhou Xiong received the bachelor's degree in automation from Sun Yat-sen University, Guangzhou, China in 2015. He is now a Ph.D. student in Institute of Automation, Chinese Academy of Sciences, Beijing, China, advised by Prof. Zhiyong Liu.

His key research interest is machine learning and he is currently studying reinforcement learning and multi-task learning.

Biao Sun received the B.S. and M.S. degerees from University of Science and technology Beijing, China in 2015 and 2018, respectively.

His research interests include robot manipulation and computer vision.

Xu Yang is an assistant professor at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

His research interests include computer vision, graph algorithms, and robotics.

Hong Qiao (SM'06) received the B.S. and M.S. degrees in Engineering both from Xi'an Jiaotong University, Xi'an, China, in 1986 and 1989, respectively, the M.Phil. degree from the University of Strathclyde, Strathclyde, U.K. in 1997, and the Ph.D. degree from De Montfort University, Leicester, U.K., in 1995.

She held teaching and research positions with Universities in the U.K. and Hong Kong, from 1990 to 2004. She is currently a '100-Talents Project' Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China. Her current research interests include robotic manipulation, robotic vision, bio-inspired intelligent robot and brain-like intelligence.

She was the first to propose the concept of an attractive region in environment (ARIE) in strategy investigations, which has been successfully applied for robot assembly, robot grasping, and part recognition - reported in *Advanced Manufacturing Alert* (New York, NY, USA: Wiley, 1999). She is a Member of the Administrative Committee of the IEEE Robotics and Automation Society (2014-2016, 2017-2019), the IEEE Medal for Environmental and Safety Technologies Committee (2014-2018), and the RAS Long Range Planning Committee (2016-2017). She is on the Editorial Boards of 5 IEEE Transactions and the Editor-in-Chief of Assembly Automation.

Kaizhu Huang works as Head of department of Electrical and Electronic Engineering and Professor in Xi'an Jiaotong-Liverpool Univ. Before that, he was an Associate Professor at National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA). Dr. Huang was a student of the Special Class for Gifted Youth at Xi'an Jiaotong Univ. and received the B.Sc. degree in Engineering in 1997. He received the M.Sc. degree in Engineering from CASIA in July 2000 and the Ph.D. degree from The Chinese Univ. of Hong Kong (CUHK) in 2004. He worked as a research scientist in Fujitsu $R\&D$ Centre from 2004 to 2007. During 2008 and 2009, he was a research fellow in CUHK and a researcher at Univ. of Bristol, U.K. His research interests include machine learning, pattern recognition, and neural information processing. He has published more than 120 papers including over 50 international journal papers. He also received a lot of international awards such as Asia Pacific Neural Network Society Young Investigator Award in 2011.

Amir Hussain (SM97) received the BEng (1st Class Honours with distinction) and PhD in Electronic and Electrical Engineering from the University of Strathclyde in Glasgow, UK, in 1992 and 1997 respectively.

He joined the University of Stirling in Scotland, UK, in 2000, where he is full Professor of Computing Science, He is also founding Director of the Cognitive Big Data Informatics (CogBiD) Laboratory, and Head of the Data Science Research Group. Professor Hussains research interests are cross-disciplinary and industry focused, aimed at pioneering brain-inspired, cognitive Big Data technology for solving complex real-world problems. In 2017, he was ranked, in an independent survey (published in Elseviers Information Processing and Management Journal), as one of worlds top two most productive researchers in sentiment analysis (since 2000). He has (co)authored more than 300 publications, including 120+ journal papers and over a dozen Books.

He has led major multi-disciplinary research projects funded by national and European research councils, local and international charities and industry, and supervised more than 30 PhDs to-date. He is founding Editor-in-Chief of two journals: Cognitive Computation (Springer), and Big Data Analytics (BioMed Central). He also serves as Associate Editor of several other leading journals including, the IEEE Transactions on Neural Networks and Learning Systems, IEEE Computational Intelligence Magazine and the IEEE Transactions on Emerging Topics in Computational Intelligence.

He has served as Invited Speaker/Organizing Committee (co)Chair for over 50 top international Conferences, including General co-Chair for the forthcoming IEEE World Congress on Computational Intelligence (WCCI'2020) - the world's largest event on computational intelligence. He is Vice-Chair of the Emergent Technologies Technical Committee of the IEEE Computational Intelligence Society, and Chapter Chair of the IEEE UK & RI Industry Applications Society Chapter.

Zhi-Yong Liu is a professor at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

His research interests include machine learning, pattern recognition, computer vision, and bioinformatics.