Specializing Versatile Skill Libraries using Local Mixture of Experts

Anonymous Author(s)
Affiliation
Address
email

Abstract: A long-cherished vision in robotics is to equip robots with skills that match the versatility and precision of humans. For example, when playing table tennis, a robot should be capable of returning the ball in various ways while precisely placing it at a desired location. A common approach to model such versatile behavior is to use a Mixture of Experts (MoE) model, where each expert is a contextual motion primitive. However, learning such MoEs is challenging as most objectives force the model to cover the entire context spaces, which prevents specialization of the primitives resulting in rather low quality components. Starting from maximum entropy reinforcement learning (RL), we decompose the objective into optimizing an individual lower bound per mixture component. Further, we introduce a curriculum by allowing the components to focus on a local context region, enabling the model to learn highly accurate skill representations. To this end, we use local context distributions that are adapted jointly with the expert primitives. Our lower bound advocates an iterative addition of new components, where new components will concentrate on local context regions not covered by the current MoE. This local and incremental learning results in a modular MoE model of unprecedented accuracy and versatility, where both properties can be scaled by adding more components on the fly. We demonstrate this by an extensive ablation and on two challenging simulated robot skill learning tasks.

Keywords: Curriculum, Versatile Skill Learning, Episodic Hierarchical RL

1 Introduction

Human motor skills are precise and versatile, which allows us to perform motor tasks in different ways while achieving a consistent movement quality. For example, when playing table tennis, we can hit the ball in various ways while still targeting a specific landing point of the ball on the opponent's side of the table. Another example is grasping an object which lies behind an obstacle. We can grasp even small objects precisely with different grasp types while avoiding the obstacle. Such versatile skills are crucial if we want to employ robots in unstructured and dynamically changing environments. Such skills, often represented as movement primitives, were already successfully learned for challenging robot learning tasks, e.g., the ball-in-a-cup [1, 2] task, by a variety of policy search algorithms [3]. Yet, most of these algorithms cannot find multiple, versatile, and precise solutions to the multi-modal solution space, as they usually assume a Gaussian policy. Thus, they can only cover one mode.

In this paper, we model versatile behavior with contextual skill libraries of motion primitives [4, 5], formalized by Mixtures of Experts (MoEs). Here the context defines task properties, e.g., reaching different goal positions or different friction parameters of an object to manipulate [6]. Given a context, the MoE first selects a component, i.e., a motion primitive, to execute. Subsequently, the component adjusts the primitive's parameters such that the underlying task is solved. Such models are already commonly used [7, 8]. Yet, the quality and versatility of the learned skill libraries remains limited using existing algorithms. Most algorithms are based on expectation-maximization (EM) [9] techniques [7], and suffer from local optima and mode averaging [9] if not enough components are available. This mode averaging also prevents the single components from specializing in a

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

local context region. Moreover, existing algorithms do not explicitly optimize the versatility of the library. Hence, they often yield degenerated libraries with only a single mode. We propose a new objective for learning contextual, precise, and versatile MoE models based on a maximum entropy formulation. This objective also introduces a learnable context distribution, which provides a curriculum for the MoE model and can be decomposed into individual updates for the components and their related local context distributions. This allows the components to specialize in local regions of the context space and prevents mode averaging. For an efficient adaptation of the curriculum, we model it as a mixture of Gaussians, where each component is related to a primitive of the MoE policy. Due to the curriculum, the MoE does not have to cover the whole context space during training, which prevents the averaging effects that negatively affect most other approaches. Yet, not covering the whole context space leads to poor performance for some contexts, which is also undesirable. Thus, we introduce a heuristic-free mechanism to add new components during training until the whole context space is covered. Additionally, we can detect and remove components that converged to sub-optimal solutions. Hence our algorithm learns skills that cover the whole context space while still being highly precise and versatile using a modular approach that allows automatic adaptation of the model complexity.

We demonstrate that we can learn versatile and precise contextual motor skills in simulated beer pong and table tennis environments while still covering the context space properly and learning precise solutions. Moreover, we present ablation studies showing the importance of the single elements and hyperparameters of our algorithm and show the quality of our solution.

2 Related Work

Contextual Episodic Policy Search. Episodic policy search [3] aims at maximizing the expected return by optimizing the parameters of a controller, e.g., a motion primitive [10, 4]. Due to the lack of gradients, most approaches use stochastic search [11, 12, 13] over the parameter space. Researchers extend these approaches to the contextual setting [14, 15], improved data efficiency [6, 16], and addressed problems such as premature convergence [17, 15].

Klink et al. [2] introduce a curriculum into contextual policy search. By having an adaptable distribution over the contexts, they allow the agent to concentrate on easy-to-solve contexts first and then generalize to the whole context space. Yet, this distribution and the policy itself are uni-modal Gaussians, which does not allow learning versatile behavior. To learn such behavior, we use multi-modal mixture models to model both the curriculum over contexts and the policy.

Versatile Skill Learning. The Hierarchical Relative Entropy Policy Search (HiREPS) algorithm [7] extends the classical Relative Entropy Policy Search (REPS) approach [18] to MoEs, which allows learning versatile skills in a contextual episodic policy search setting. In a similar approach, Layered Direct Policy Search (LaDiPS) [8] also uses MoE policies, but builds on Model-Based Relative Entropy Policy Search (MORE) [17] instead of REPS. Both HiREPS and LaDiPS address the same problems as our approach, yet there are also considerable differences. First, HiREPS jointly optimizes the whole mixture model and introduces an additional ad hoc constraint, which is crucial to obtain versatile and well-performing solutions. LaDiPS uses separate updates for the different parts of the mixture but also relies on ad hoc constraints to enforce versatility. In contrast, for our approach, the objective and separate updates of the individual mixture parts follow naturally from the maximum entropy formulation. Second, neither HiREPS nor LaDiPS uses a curriculum for training. Thus, in both approaches, the MoE always has to cover the whole context space. Hence the components cannot specialize. We show in our experiments that the specialization of components due to the curriculum is crucial to obtain highly accurate skill libraries.

Versatile skill learning has also been considered in the evolutionary strategy community, most notably with the Map-Elite algorithm [19], in which behavioral descriptors are defined and used to distinguish the different learned motions. However, the algorithm can not be easily applied to the contextual setting where different controller parameters should be chosen in different situations.

Variational Inference. Our algorithm is also based on several recent advances in variational inference [20, 21, 22]. It is well known that maximum entropy RL is equivalent to inference in an appropriate probabilistic model [23]. Similar to previous works [24, 25], we exploit this relation and draw inspiration from recent research into variational inference and density estimation for Gaussian mixture models and MoEs [20, 21, 22]. We reformulate the lower bound objectives introduced in

those approaches for our maximum entropy RL setting and extend them with a curriculum for the mixture components.

Related Step-Based Approaches. In the step-based setting, the policy does not learn a function from contexts to parameters of an episodic controller but directly maps from system states to actions. Practitioners often use deep neural networks to parameterize step-based policies, giving rise to the field of deep RL. In this context, versatile policy learning is also a very active research area [26, 27, 28]. Similar to us, these approaches build on a maximum entropy objective, yet, they need to train an explicit discriminator to model the responsibilities, i.e., the probability of a skill given the state and, potentially, the action. In our approach, these responsibilities are available in closed form. While the approaches from [26, 27] assume the responsibilities are only depending on the state, both, [28] and our approach assume they depend on the action as well.

Researchers also worked on introducing curricula into deep RL. In a first approach Ghosh et al. [29] proposed partitioning the initial state distribution using clustering. They then learn individual policies for each partition while keeping the partitioning fixed. Strictly, this is not a curriculum as the partitioning is not adjusted, yet it still allows specialization of the individual policies in different regions of the state space. To automatically generate and adapt a curriculum for deep RL approaches, Klink et al. [30] extended their approach from the episodic setting [2] to the step-based setting. Yet, neither of these approaches is addressing versatility.

All these approaches are relevant for our work, as they use similar ideas and try to solve similar problems. Yet, we again want to emphasize the fundamental differences between the step-based and episodic settings that make a direct comparison difficult. A first example is exploration. While in the step-based case we usually explore the action space using uncorrelated noise, the episodic setting uses correlated distributions directly in the parameter space of the controller [3]. Another example are non-Markovian rewards, i.e., rewards that depend on the history of state and actions. Such rewards are often easier to define but only usable in the episodic setting, a benefit we exploit in our experiments.

3 Specializing Versatile Mixture of Expert Models

In this section, we derive a lower bound to optimize each component and its corresponding context distribution independently, such that the resulting MoE model can find skills that are precise and versatile. Our mixture of experts model is given as $\pi(\boldsymbol{\theta}|\mathbf{c}) = \sum_o \pi(o|\mathbf{c})\pi(\boldsymbol{\theta}|\mathbf{c}, o)$, where $\pi(o|\mathbf{c})$ is the gating distribution, assigning a probability to component o given the context $\mathbf{c}$ and $\pi(\boldsymbol{\theta}|\mathbf{c}, o)$ is the distribution for component o, which adapts the motion primitive's parameters $\boldsymbol{\theta}$ to the given context $\mathbf{c}$. In order to introduce the local context distributions $\pi(\mathbf{c}|o)$ for a component o, we can use Bayes' Rule to rewrite our policy as

$$\pi(\boldsymbol{\theta}|\mathbf{c}) = \sum_o \frac{\pi(\mathbf{c}|o)\pi(o)}{\pi(\mathbf{c})}\pi(\boldsymbol{\theta}|\mathbf{c}, o). \tag{1}$$

The context distribution $\pi(\mathbf{c})$ is given as $\pi(\mathbf{c}) = \sum_o \pi(\mathbf{c}|o)\pi(o)$. We model the components $\pi(\boldsymbol{\theta}|\mathbf{c}, o)$ as a linear conditional Gaussian distribution and the component-wise state distribution $\pi(\mathbf{c}|o)$ as a Gaussian. The prior weights $\pi(o)$ define a categorical distribution. Please note that we do not show the full derivations throughout this section and refer the interested reader to Appendix A for a detailed derivation.

3.1 Maximum Entropy Skill Learning with Curriculum

We consider a maximum entropy objective [23] for episodic policy search

$$\max_{\pi(\boldsymbol{\theta}|\mathbf{c})} \mathbb{E}_{p(\mathbf{c})}\left[\mathbb{E}_{\pi(\boldsymbol{\theta}|\mathbf{c})}\left[\mathrm{R}(\mathbf{c}, \boldsymbol{\theta})\right] + \alpha \mathrm{H}\left[\pi(\boldsymbol{\theta}|\mathbf{c})\right]\right],$$

where $p(\mathbf{c})$ is the task specific context distribution and $\pi(\boldsymbol{\theta}|\mathbf{c})$ is our MoE policy. The reward maximization enforces preciseness, the entropy bonus enforces versatility. However, the standard maximum entropy objective does not allow each mixture component to create its own curriculum.

Inspired by the work from Klink et al. [2], we extend the objective to

$$\max_{\pi(\mathbf{c},\boldsymbol{\theta})} \mathbb{E}_{\pi(\mathbf{c})}\left[\mathbb{E}_{\pi(\boldsymbol{\theta}|\mathbf{c})}\left[\mathrm{R}(\mathbf{c}, \boldsymbol{\theta})\right] + \alpha \mathrm{H}\left[\pi(\boldsymbol{\theta}|\mathbf{c})\right]\right] - \beta \mathrm{KL}\left(\pi(\mathbf{c}) \parallel p(\mathbf{c})\right), \tag{2}$$

where α and β are scaling parameters. We are now able to optimize the joint distribution $\pi(\mathbf{c}, \boldsymbol{\theta})$, where the Kullback-Leibler (KL) term ensures that the context distribution $\pi(\mathbf{c})$ recovers the task specific context distribution $p(\mathbf{c})$. Since we generally want to cover the whole context space and therefore consider each context equally important, we assume $p(\mathbf{c})$ to be uniformly distributed in a given interval. Note that this objective is similar to the negative joint I-projection used in variational inference. Here, we also exploit the properties of the I-projection for learning the context distribution – the I-projection is mode seeking instead of mode-averaging and therefore allows for a specialization on a local context area. Yet, the given objective is difficult to optimize for mixture models as the sum over the mixture components is appearing inside the log terms of the entropy and the KL. However, we can derive a similar lower bound as in [20] by inserting our policy model from Equation (1) into Objective (2) and applying the identities $\log \pi(\boldsymbol{\theta}|\mathbf{c}) = \log \frac{\pi(\boldsymbol{\theta}|\mathbf{c},o)\pi(o|\mathbf{c})}{\pi(o|\mathbf{c},\boldsymbol{\theta})}$, $\log \pi(\mathbf{c}) = \log \frac{\pi(\mathbf{c}|o)\pi(o)}{\pi(o|\mathbf{c})}$ and obtain a new objective

$$\begin{aligned}\max_{\pi(\mathbf{c},\boldsymbol{\theta})} \sum_o \pi(o)\mathbb{E}_{\pi(\mathbf{c}|o),\pi(\boldsymbol{\theta}|\mathbf{c},o)} \left[\mathrm{R}(\mathbf{c},\boldsymbol{\theta}) - \alpha \log \pi(\boldsymbol{\theta}|\mathbf{c},o) + \alpha \log \pi(o|\mathbf{c},\boldsymbol{\theta})\right] \\ - \sum_o \pi(o)\mathbb{E}_{\pi(\mathbf{c}|o)} \left[\beta \log \pi(\mathbf{c}|o) + \beta \log \pi(o) - (\beta - \alpha) \log \pi(o|\mathbf{c})\right]. \end{aligned} \tag{3}$$

In Sections 3.2 and 3.3, we describe the interpretation of the individual terms of this objective in detail. Yet, we already make two important observations. First, we see that the objective contains a sum over the components such that individual component updates are possible. However, the log responsibilities $\log \pi(o|\mathbf{c},\boldsymbol{\theta}) = \log \frac{\pi(\boldsymbol{\theta}|\mathbf{c},o)\pi(o|\mathbf{c})}{\pi(\boldsymbol{\theta}|\mathbf{c})}$ and $\log \pi(o|\mathbf{c}) = \log \frac{\pi(\mathbf{c}|o)\pi(o)}{\pi(\mathbf{c})}$ still hinder us from optimizing each component $\pi(\boldsymbol{\theta}|\mathbf{c},o)$ and its corresponding context distribution $\pi(\mathbf{c}|o)$ independently. We will introduce a lower bound in the following section that does not have this limitation. Second, we see that the contexts $\mathbf{c}$ are sampled from the distribution $\pi(\mathbf{c}|o)$ for each $\pi(\boldsymbol{\theta}|\mathbf{c},o)$. Since $\pi(\mathbf{c}|o)$ is a Gaussian distribution, it is ensured that $\pi(\boldsymbol{\theta}|\mathbf{c},o)$ only sees samples from a local region during optimization - ensuring that each $\pi(\boldsymbol{\theta}|\mathbf{c},o)$ specializes in a local context region without the need to cover multiple context modes.

3.2 Lower-Bound Decomposition for Component Distributions

In order to maximize the objective in Equation (3) for each component $\pi(\boldsymbol{\theta}|\mathbf{c},o)$ individually, we can first extract the terms which only depend on $\pi(\boldsymbol{\theta}|\mathbf{c},o)$ as

$$\max_{\pi(\boldsymbol{\theta}|\mathbf{c},o)} \sum_o \pi(o)\mathbb{E}_{\pi(\mathbf{c}|o),\pi(\boldsymbol{\theta}|\mathbf{c},o)} \left[\mathrm{R}(\mathbf{c},\boldsymbol{\theta}) - \alpha \log \pi(\boldsymbol{\theta}|\mathbf{c},o) + \alpha \log \pi(o|\mathbf{c},\boldsymbol{\theta})\right]. \tag{4}$$

Similar to [20], we can obtain a tight lower-bound by introducing an auxiliary distribution $\tilde{\pi}(o|\mathbf{c},\boldsymbol{\theta}) = \pi(o|\mathbf{c},\boldsymbol{\theta})$ resulting in

$$\max_{\pi(\boldsymbol{\theta}|\mathbf{c},o)} \sum_o \pi(o)\mathbb{E}_{\pi(\mathbf{c}|o),\pi(\boldsymbol{\theta}|\mathbf{c},o)} \left[\mathrm{R}(\mathbf{c},\boldsymbol{\theta}) + \alpha \log \tilde{\pi}(o|\mathbf{c},\boldsymbol{\theta})\right] + \alpha \mathrm{E}_{\pi(o),\pi(\mathbf{c}|o)} \left[\mathrm{H}\left(\pi(\boldsymbol{\theta}|\mathbf{c},o)\right)\right]. \tag{5}$$

The auxiliary distribution is fixed during the optimization and can be computed according to the last policy model allowing us to update each component independently. The newly introduced Objective (5) is a lower bound to the Objective (4), which is tight after the update of the auxiliary distribution, guaranteeing the convergence of the algorithm. The objective in Equation (5) is an augmented version of the standard maximum entropy RL objective and thus can be optimized with any suitable Policy Search algorithm. In this work we modify contextual Model-Based Relative Entropy Policy Search (MORE) [14], such that it can deal with the maximum entropy objective.

3.3 Lower-Bound Decomposition for Context Distributions and Prior Weights

In order to update $\pi(\mathbf{c}|o)$, we extract the relevant terms from Objective (3)

$$\max_{\pi(\mathbf{c}|o)} \sum_o \pi(o)\mathbb{E}_{\pi(\mathbf{c}|o)} \left[L_c(o,\mathbf{c}) + (\beta - \alpha) \log \tilde{\pi}(o|\mathbf{c})\right] + \beta \mathrm{E}_{\pi(o)} \left[\mathrm{H}\left(\pi(\mathbf{c}|o)\right)\right], \tag{6}$$

where $L_c(o,\mathbf{c}) = \int_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta}|\mathbf{c},o) \left[\mathrm{R}(\mathbf{c},\boldsymbol{\theta}) - \alpha \log \pi(\boldsymbol{\theta}|\mathbf{c},o) + \alpha \log \tilde{\pi}(o|\mathbf{c},\boldsymbol{\theta})\right] d\boldsymbol{\theta}$ and $\tilde{\pi}(o|\mathbf{c}) = \pi(o|\mathbf{c})$ is a second auxiliary distribution. Note, that we used the auxiliary distribution $\tilde{\pi}(o|\mathbf{c},\boldsymbol{\theta})$

as part of the modified reward function $L_c(o, \mathbf{c})$. Also note, that we approximate the integral over $\boldsymbol{\theta}$ with a single sample, since we assume to have only one $(\mathbf{c}, \boldsymbol{\theta})$ sample. As we face the same problem as in Section 3.2 regarding the log-responsibilities $\log \pi(o|\mathbf{c})$, we already introduced a second auxiliary distribution $\tilde{\pi}(o|\mathbf{c}) = \pi(o|\mathbf{c})$ to obtain a tight lower bound. After the optimization step (Equation 6), we obtain the optimal solution $\pi(\mathbf{c}|o)^*$ and tighten the bound by setting $\tilde{\pi}(o|\mathbf{c}) = \pi(o|\mathbf{c})^*$ and $\tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta}) = \pi(o|\mathbf{c}, \boldsymbol{\theta})^*$. This lower bound corresponds to a modified maximum entropy objective and thus can be updated with any suitable policy search method. We use an adjusted version of MORE [17], which is also used in [20].

Finally, we can formulate the objective for updating the component weights $\pi(o)$, which resembles a lower bound of the original Objective (3) and corresponds to the highest hierarchy in our update scheme. The objective is given as

$$\max_{\pi(o)} \sum_{o} \pi(o) \left[\mathbb{E}_{\pi(\mathbf{c}|o)} \left[L_c(o, \mathbf{c}) + (\beta - \alpha) \log \tilde{\pi}(o|\mathbf{c}) \right] + \beta \mathrm{H}\left(\pi(\mathbf{c}|o)\right) \right] + \beta \mathrm{H}\left(\pi(o)\right) \tag{7}$$

The objective in Equation (7) is a maximum entropy objective for discrete action distributions and can be optimized with any suitable Policy Search method. We use REPS [18] throughout this work.

To summarize, we split the initial objective in Equation (3) into different hierarchies, allowing us to optimize the different terms in our mixture model individually. Starting by first updating the components using the maximization problem in Equation (5), we can optimize for the component-wise state distributions $\pi(\mathbf{c}|o)$ with Objective (5) after tightening the bound. Finally, we update the weight distribution $\pi(o)$. Intuitively, $\tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta})$ and $\tilde{\pi}(o|\mathbf{c})$ specify how probable it is that samples $(\mathbf{c}, \boldsymbol{\theta})$ and $(\mathbf{c})$ were generated from component o. Thus, low probability values lead to less reward (see Equations (5) and (6)), forcing the components to concentrate on uncovered context and context-action spaces.

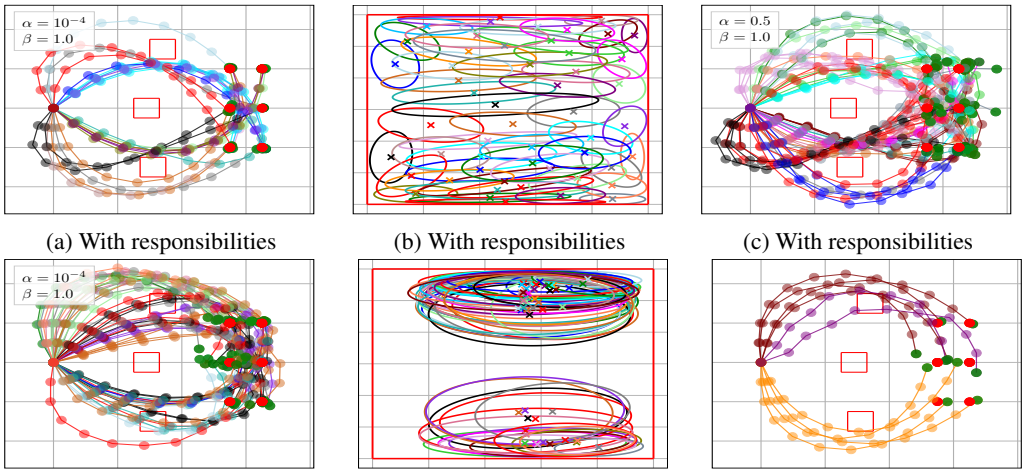
Algorithmic Details and Addition of Components. We initialize our algorithm with only one component and incrementally add components, and its corresponding context distribution, randomly. We fix all components except for the newly added one and optimize it for K iterations to let it discover new solutions in yet undiscovered context regions. After K iterations we can add a new component and repeat the procedure. Due to the additional reward induced by the auxiliary distribution $\tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta})$ the new component will focus on undiscovered solutions and due to $\tilde{\pi}(o|\mathbf{c})$ the context distribution will cover uncovered areas of the context space. Note that such a simple adding procedure is only possible due to the mode-seeking properties of the I-projection, as the components do not need to average over multiple modes but can specialize on a local context region. We also fix the weights $\pi(o)$ to be uniformly distributed among all components, since otherwise components which are already fully trained might dominate the optimization. By allowing to fine-tune all components every H iterations, the already added components can adjust to the newly added ones. After finishing adding components, we update the weights $\pi(o)$ at the end of our optimization procedure. We describe the full details of our algorithm in Appendix B.

4 Experiments

In this section, we empirically analyze our method. First, we investigate the importance of the different parameters and distributions of the objectives derived in Section 3. In the second part, we investigate the learned skills on versatility and preciseness on the two sophisticated robot beer pong and robot table tennis experiments. Please note that experimental details for the following section can be found in Appendix C.

4.1 Ablation Studies

We investigate the importance of the auxiliary distributions $\tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta})$ and $\tilde{\pi}(o|\mathbf{c})$ on a 10-link planar reaching task with two-dimensional context space. For this purpose, we update the component distributions $\pi(\boldsymbol{\theta}|\mathbf{c}, o)$ and the corresponding context distributions $\pi(\mathbf{c}|o)$ by simply optimizing the objectives in (5, 6) with i) considering the responsibilities (as given by our algorithm) and ii) by setting $\log \tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta})$ and $\log \tilde{\pi}(o|\mathbf{c})$ to zero. Furthermore, we showcase the differences between learned solutions of HiREPS and our method on the 10-link planar task.



(a) With responsibilities (b) With responsibilities (c) With responsibilities

(d) Without responsibilities (e) Without responsibilities (f) HiREPS

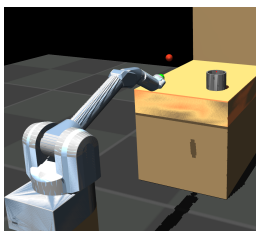
Figure 1: **Importance of auxiliary distributions and HiREPS comparison.** We show a snapshot of the learned policies of the planar reaching task for a 2-dim context space by considering $\log \tilde{\pi}(o|\mathbf{c}, \boldsymbol{\theta})$, $\log \tilde{\pi}(o|\mathbf{c})$ ((a)+(b)+(c)) and neglecting the auxiliary distributions ((d)+(e)). By considering these terms we can learn more diverse solutions (a) + (c) and cover the whole 2d-context space (b), whereas the solutions of the policy trained by neglecting the terms show nearly the same, partially invalid solutions (d) and are not able to cover the whole context space (e) leading to poor generalization performance. The solutions by HiREPS (f) indicate less versatility compared to the solutions (a) + (c). Note that each color indicates a different component.

Versatility is Induced by the Auxiliary Distributions. The toy task for this ablation is a 10-link planar reaching task (see Figure 1a), where the robot has to reach the red dots with its end-effector in a 2d context-space while avoiding the rectangle-shaped obstacles. By adding 60 components, we have trained both versions (i and ii) over 15 seeds/trials and have chosen the best performing parameter constellations (Fig. 1a for i), Fig. 1d) for ii)). We then picked the first model and sampled for each of the shown six contexts vectors (red dots) 100 samples and plotted the corresponding mean of each sampled component. The results indicate the importance of the auxiliary distributions. The plots for i) (Fig. 1a + Fig. 1c) show versatile – several modes to the same context – and precise – small distance of end-effector (green dot) to goal (red dot), while avoiding obstacles– solutions while fully covering the context space (Fig. 1b). A trend of covering more modes with higher α can also be seen in (Fig. 1c). For the case without the responsibilities ii) (Fig. 1d), the solutions are not precise and invalid since the planar robot reaches the target through the obstacles. This effect is since the context space is not fully covered (Fig. 1e) by the component-wise context distributions, such that the model has to extrapolate the solutions from components that are not trained for the contexts shown in red dots.

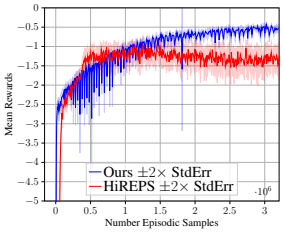
Comparison to HiREPS. We performed a hyperparameter optimization on the same 10-link planar reacher task and picked the best performing model among the 15 seeds/trials. For the same contexts as before, we show the mean solutions of the sampled components in Fig. 1f. Note that the sampling procedure was the same as before – for each context we sampled 100 components. It can be observed that HiREPS has less diverse solutions since only three components among its 60 components were chosen in total. Furthermore, slight deviations of the end-effector from the target context can also be observed.

4.2 Robotic Experiments

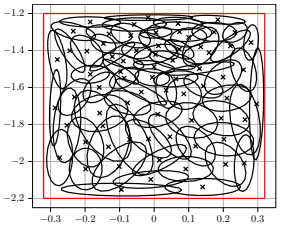
We test and compare our algorithm on a Beer Pong and Table Tennis simulation environment in MuJoCo [31], where we have chosen HiREPS [7] as baseline. We believe that HiREPS is a suitable algorithm to compare, since it is a contextual policy search algorithm and considers optimizing a mixture model, while learning versatile skills. Throughout our experiments we choose probabilistic movement primitives (ProMP) [4] as policy representation, where the weight parameters of the ProMP as well as the execution duration of the motion primitive is learned. We consider non-



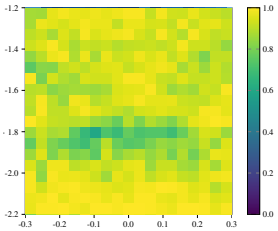
(a) Beer Pong Environment



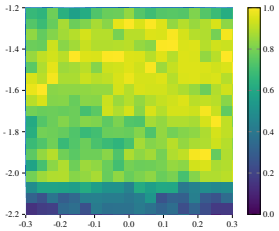
(b) Rewards



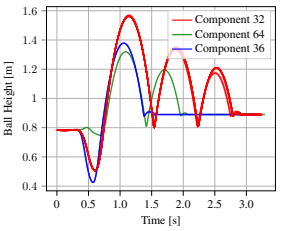
(c) Coverage of Context Space



(d) Success Rates Ours



(e) Succes Rates HiREPS



(f) Exemplary Ball Trajectories

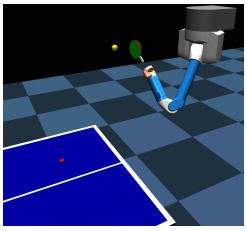
Figure 2: **Beer Pong Experiment.** In the Beer Pong experiment (a), the robot has to throw the red ball into a cup. The learned mixture of expert (MoE) models show a high quality of the skills, reflected in the high reward value (b) and dense coverage of the 2-dimensional context space, i.e., the position of the cup with the component-wise context distributions (c). Comparing the success rates (92% in average) of our approach (d) and HiREPS (75% in average) (e) shows that our approach performs better throughout the context space. Here, we consider a trial successful if the ball goes in the cup. Although our method enforces versatility only in the joint space, this also results in versatile ball trajectories for a given context (f).

markovian rewards, in which the reward depends on the history of state and actions. This type of reward function is not applicable to common step-based RL methods which build on markovian properties. As for analyzing the results of the experiments, we focus on the questions i) how does our algorithm perform compared to the baseline, ii) are we able to cover the whole context space and iii) are we able to learn versatile skills.

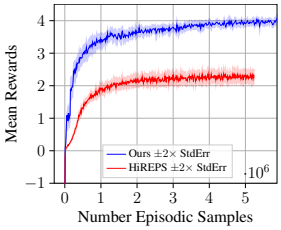
We start with the Beer Pong task (see Fig. 2a). The goal of the Barret WAM robot is to throw the red ball into the cup on the table. The 2-dim contexts resemble the position of the cup on the table. We incrementally add 70 components in our method, while HiREPS directly starts with 70 components. We have run both algorithms over 20 seeds/trials and report the performance in Fig. 2b[1], where we plot the mean reward with two times the standard error. Our algorithm allows to cover the whole context space with the learned component-wise distributions (see Fig. 2c) resulting in a high end reward. This performance is reflected in Fig. 2d, where we have divided the context space into a fine grid and sampled 100 times for each context a component from which we have executed the mean. We repeated that for all 20 different models and plot the mean success rate of throwing the ball into the cup. We did the same procedure for HiREPS in Fig. 2e. We can observe that HiREPS has much more darker rectangles, showing that the success rate in these context regions is low. Although, versatility is encouraged in the joint space of the robot, different joint trajectories often yield different ball trajectories. Given one context, in Fig. 2f we show the z-coordinates of the ball trajectories over time resulting from sampling 20 times from the MoE model. Each component leads to different ball trajectories, resulting in different number of "ball-jumps" on the table.

As a second robotic experiment, we consider learning versatile skills for playing table tennis (Fig. 3a), where the robot has to return different incoming balls to desired targets on the opponent's table side. We consider a four dimensional context space, including ball's initial serve position and ball's target landing position (right and left half of table Fig. 3c). For both parts of the context, we fix the z position and vary the x and y coordinates. We incrementally add 50 components in our method, while HiREPS directly starts with 50 components. We have run both algorithms over 20 seeds/trials

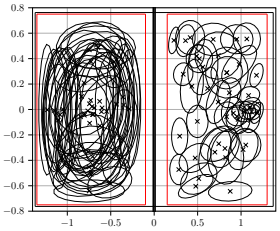
[1]Note that we have excluded the lastly added component from testing for obtaining the true performance, since this component is not fully trained yet and would not be chosen by the model if we would not assume $\pi(o)$ to be uniformly distributed.



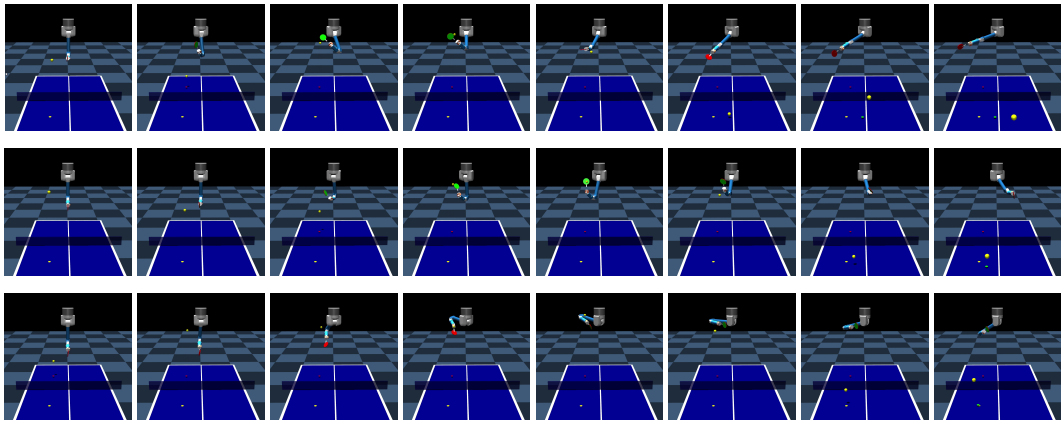
(a) Table Tennis Environment



(b) Rewards



(c) Coverage of Context Space



(d) Versatile striking incoming ball with same context

Figure 3: **Table Tennis Experiment.** The Table Tennis simulation environment (a). The learned mixture of expert (MoE) models show a high quality of the skills, reflected in the high reward value (b) and dense coverage of the 4-dimensional context space, i.e., the target position (left side) of the outgoing ball and the landing position (right side) of the incoming ball (c). Versatile strikings for a fixed context. The robot is able to hit the (yellow) ball in various ways, also indicated through the different colors of the racket sides (d). Note that the red and yellow dots on the table are markers for the serving and desired landing position respectively.

and report the performance in Fig. 3b[1], where we plot the mean reward with two times the standard error. The component-wise context distributions are spread among the context space (Fig. 3c) and allow each component to locally specialize on a context region. This high coverage of the context space allows for a high reward achievement, outperforming HiREPS. In Fig. 3d, we show three different striking skills sampled from our trained MoE model, for a fixed context, i. e. fixed serving and desired ball position. The first two skills use the green side of the racket (can be regarded as forehand) while the third skill uses the red side of the racket (backhand), which shows that our method is able to learn totally different skills, similar to human players.

5 Conclusion

We proposed a new objective for learning contextual and versatile Mixtures of Experts (MoE) models. We based our objective on a maximum entropy formulation to increase the versatility of the solutions and introduced a curriculum to allow the components to specialize. Our formulation also allows for easy online adaptation of the model complexity during training. We conducted an ablation to show the importance of the individual parts of our objective. Further, we showed that our method learns precise and versatile solutions, which outperform the baseline on sophisticated simulated robotic tasks. This work aims at presenting a mathematically well-founded method and demonstrates its general feasibility on various challenging tasks. Currently, the major drawback of our approach is sample efficiency, as we do not share experience between the components. We intend to address this issue in future work, e.g., by intra-option learning. Another direction for future research is extending the approach to more complex models, such as non-linear mixtures of experts.

References

[1] J. Kober and J. Peters. Policy search for motor primitives in robotics. In *Learning Motor Skills*, pages 83–117. Springer, 2014.

[2] P. Klink, H. Abdulsamad, B. Belousov, and J. Peters. Self-paced contextual reinforcement learning. In *Conference on Robot Learning*, pages 513–529. PMLR, 2020.

[3] M. P. Deisenroth, G. Neumann, J. Peters, et al. A survey on policy search for robotics. *Foundations and trends in Robotics*, 2(1-2):388–403, 2013.

[4] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann. Probabilistic movement primitives. *Advances in neural information processing systems*, 26:2616–2624, 2013.

[5] A. J. Ijspeert, J. Nakanishi, and S. Schaal. Learning attractor landscapes for learning motor primitives. Technical report, 2002.

[6] A. Kupcsik, M. Deisenroth, J. Peters, and G. Neumann. Data-efficient generalization of robot skills with contextual policy search. In *Proceedings of the AAAI conference on artificial intelligence*, volume 27, 2013.

[7] C. Daniel, G. Neumann, and J. Peters. Hierarchical relative entropy policy search. In *Artificial Intelligence and Statistics*, pages 273–281. PMLR, 2012.

[8] F. End, R. Akrour, J. Peters, and G. Neumann. Layered direct policy search for learning hierarchical skills. pages 6442–6448, 05 2017. doi:10.1109/ICRA.2017.7989761.

[9] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[10] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert. Learning movement primitives. In *Robotics research. the eleventh international symposium*, pages 561–572. Springer, 2005.

[11] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001.

[12] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber. Natural evolution strategies. *The Journal of Machine Learning Research*, 15(1):949–980, 2014.

[13] S. Mannor, R. Y. Rubinstein, and Y. Gat. The cross entropy method for fast policy search. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 512–519, 2003.

[14] V. Tangkaratt, H. van Hoof, S. Parisi, G. Neumann, J. Peters, and M. Sugiyama. Policy search with high-dimensional context variables. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[15] A. Abdolmaleki, D. Simões, N. Lau, L. P. Reis, and G. Neumann. Contextual direct policy search. *Journal of Intelligent & Robotic Systems*, 96(2):141–157, 2019.

[16] R. Pinsler, P. Karkus, A. Kupcsik, D. Hsu, and W. S. Lee. Factored contextual policy search with bayesian optimization. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7242–7248. IEEE, 2019.

[17] A. Abdolmaleki, R. Lioutikov, N. Lua, L. Paulo Reis, J. Peters, and G. Neumann. Model-based relative entropy stochastic search. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, pages 153–154, 2016.

[18] J. Peters, K. Mulling, and Y. Altun. Relative entropy policy search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, 2010.

[19] A. Cully, J. Clune, D. Tarapore, and J.-B. Mouret. Robots that can adapt like animals. *Nature*, 521(7553):503–507, May 2015. doi:10.1038/nature14422. URL `https://hal.archives-ouvertes.fr/hal-01158243`.

[20] O. Arenz, G. Neumann, and M. Zhong. Efficient gradient-free variational inference using policy search. In *International conference on machine learning*, pages 234–243. PMLR, 2018.

[21] O. Arenz, M. Zhong, and G. Neumann. Trust-region variational inference with gaussian mixture models. *Journal of Machine Learning Research*, 21(163):1–60, 2020.

[22] P. Becker, O. Arenz, and G. Neumann. Expected information maximization: Using the i-projection for mixture density estimation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ByglLlHFDS.

[23] S. Levine. Reinforcement learning and control as probabilistic inference: Tutorial and review. *arXiv preprint arXiv:1805.00909*, 2018.

[24] G. Neumann et al. Variational inference for policy search in changing situations. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pages 817–824, 2011.

[25] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870. PMLR, 10–15 Jul 2018. URL http://proceedings.mlr.press/v80/haarnoja18b.html.

[26] B. Eysenbach, A. Gupta, J. Ibarz, and S. Levine. Diversity is all you need: Learning skills without a reward function. *CoRR*, abs/1802.06070, 2018. URL http://arxiv.org/abs/1802.06070.

[27] S. Kumar, A. Kumar, S. Levine, and C. Finn. One solution is not all you need: Few-shot extrapolation via structured maxent rl. *arXiv preprint arXiv:2010.14484*, 2020.

[28] T. Osa, V. Tangkaratt, and M. Sugiyama. Discovering diverse solutions in deep reinforcement learning. *arXiv preprint arXiv:2103.07084*, 2021.

[29] D. Ghosh, A. Singh, A. Rajeswaran, V. Kumar, and S. Levine. Divide-and-conquer reinforcement learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJwelMbR-.

[30] P. Klink, C. D' Eramo, J. R. Peters, and J. Pajarinen. Self-paced deep reinforcement learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9216–9227. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/68a9750337a418a86fe06c1991a1d64c-Paper.pdf.

[31] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi:10.1109/IROS.2012.6386109.