Intrinsically-Motivated Robot Learning of Bayesian Probabilistic Movement Primitives

Thibaut Kulak and Sylvain Calinon

Abstract— **We present an approach for internally-guided learning in the context of a multi-task robot skill acquisition framework. More specifically, we focus on learning a parametrized distribution of robot movement primitives by using active intrinsically-motivated learning. We focus on the case where the learning process is initialized with human demonstrations, and refined through experiences. Such approach aims at combining experiential and observational learning. We demonstrate the effectiveness of our approach on a waste throwing task with a simulated 7-DoF Franka Emika robot.**

I. Introduction

Intrinsically-motivated learning (a.k.a. curiosity-driven learning) has emerged as an efficient approach for autonomous lifelong learning in robots [1, 2]. It is inspired by the ability of humans to discover how to produce interesting effects in their environments [3]–[5]. In [5], psychologists suggested that exploration might be triggered and rewarded for situations that include novelty/surprise. They observed that the most rewarding situations were those with an intermediate level of novelty, between already familiar and completely new situations. This also seems to be confirmed by recent neuroscience studies showing that dopamine might be released, not only for predicting external rewards such as food, but also for internal rewards such as prediction errors [6]. This suggests that intrinsic motivation systems might be present in the brain, potentially by the presence of signals related to prediction errors.

Given this background, a way to implement an intrinsic motivation system might be to build a mechanism which can evaluate the degree of novelty of different situations from the point of view of a learning robot, and then designing an associated reward being maximal when these features are in an intermediate level. Maximizing this reward can then create an active exploratory behavior [1, 7].

In this work, we propose a Bayesian framework for intrinsically-motivated learning of robot movement primitives. Leveraging a few initial human demonstrations, we propose a way to choose actively which movement is going to improve the most the knowledge of the task. We demonstrate the usefulness of our approach on a simulated waste throwing task with a 7-DoF Franka Emika robot. Specifically, we evaluate the novelty of a robot movement in terms of the uncertainty associated to the object movement, and design a reward that is a tradeoff between this novelty and the proximity to previous demonstrations/trials.

II. Bayesian Movement Representation

In this section, we present the movement representation. We build upon the widely used framework of probabilistic movement primitives (ProMPs) [8], which we extend with a Bayesian perspective.

A. Probabilistic movement primitive (ProMP)

A ProMP is a probability distribution over trajectories built from a series of N demonstrations (trajectories) of length T and of D dimensions. A demonstration $\boldsymbol{\tau}_i \in \mathbb{R}^{(T\times D)}$ is approximated by a sum of M basis functions, which are often chosen as radial basis functions (RBF)

$$\boldsymbol{\tau}_i = \boldsymbol{\Phi}\boldsymbol{w}_i + \boldsymbol{\epsilon}, \qquad \text{with} \qquad \boldsymbol{\Phi} = \boldsymbol{\Phi}^{\text{1d}} \otimes \mathbb{I}_D, \tag{1}$$

where $\otimes$ represents the Kronecker product, $\boldsymbol{\epsilon}$ is zero-mean i.i.d. Gaussian noise, $\boldsymbol{w}_i$ of size $MD\times 1$ is the weight associated to the i^{th} demonstration, $\Phi^{\text{1d}}_{T\times M}$ is the basis function matrix with $\Phi^{1d}_{t,m} = \Phi_m(t)$ corresponding to the m^{th} basis function indexed at time t, and $\mathbb{I}_D$ is the identity matrix.

The weight vectors associated to each demonstration are learned through least squares with

$$\boldsymbol{w}_i = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{\tau}_i. \tag{2}$$

A probability distribution $p(\boldsymbol{w})$ can then be learned from the demonstrations $\{\boldsymbol{w}_i\}_{i=1}^N$, usually with a multivariate Gaussian or a GMM.

This probability distribution $p(\boldsymbol{w})$ can then be used for generalization/adaptation to different environments, typically by conditioning on trajectory keypoints.

B. Bayesian Gaussian Mixture Model (BGMM)

In this section, we present the learning of the joint distribution of weights with a BGMM. For conciseness purposes, we give here an overview of the approach, but the reader can refer to [9] for more details, where we proposed to use BGMM for active imitation learning of movement primitives.

1) Joint distribution: The joint distribution is defined by a mixture of K multivariate normal distributions (MVNs) with means $\boldsymbol{\mu}=\{\boldsymbol{\mu}_k\}_{k=1}^K$, precision matrices $\boldsymbol{\Lambda}=\{\boldsymbol{\Lambda}_k\}_{k=1}^K$ and mixing coefficients $\boldsymbol{\pi}=\{\pi_k\}_{k=1}^K$ as

$$p(\boldsymbol{w}|\boldsymbol{\pi},\boldsymbol{\mu},\boldsymbol{\Lambda}) = \sum_{k=1}^K \pi_k \mathcal{N}(\boldsymbol{w}|\boldsymbol{\mu}_k,\boldsymbol{\Lambda}_k^{-1}). \tag{3}$$

This work was supported by the ROSALIS project (Swiss National Science Foundation).

The authors are with the Idiap Research Institute, Rue Marconi 19, 1920 Martigny, Switzerland (e-mail: firstname.lastname@idiap.ch.)

A Normal-Wishart prior is used for means and precision matrices, and a Dirichlet prior is put on the mixing coefficients.

The means, the precision matrices and the mixing coefficients maximizing the posterior distribution are estimated using closed-form update equations similar to those of the Expectation-Maximization (EM) algorithm for the maximum likelihood solution, see Section 10.2.1 in [10] for further details. Also, they are available as parts of standard machine learning libraries (e.g., *scikit-learn* for Python).

Given N demonstrations $\boldsymbol{W} = \{\boldsymbol{w}_i\}_{i=1}^{N}$, the predictive density $p(\hat{\boldsymbol{w}}|\boldsymbol{W})$ of a new weight $\hat{\boldsymbol{w}}$ is equivalent to a mixture of multivariate t-distributions [10].

2) Conditional distribution: The weights represent the evolution of the state with time. For instance, the state can represent the joint angle values of a robot manipulator and the Cartesian position of an object. We can then condition on a particular value $\hat{\boldsymbol{w}}^i$ of an input dimension (e.g., dimensions representing the robot joint space) to get the conditional posterior predictive distribution $p(\hat{\boldsymbol{w}}^o|\hat{\boldsymbol{w}}^i, \boldsymbol{W})$ of an output dimension (e.g., dimensions representing the object), as in [10] (Section 10.2.3) and [9], namely a mixture of multivariate t-distributions.

C. Quantifying the uncertainties

We have shown in [9] that the conditional posterior predictive distribution encodes two types of uncertainties: the aleatoric uncertainty (possible variations of the task, the one learned with standard ProMPs) and the epistemic uncertainty (representing the lack of knowledge). We observed that the aleatoric uncertainty does not depend on the context $\hat{\boldsymbol{w}}^i$, while the epistemic uncertainty grows quadratically with it. Such a decomposition is particularly useful in the context of ProMPs, because we can have access to the aleatoric uncertainty to design minimal intervention control behaviors, or the epistemic uncertainty for quantifying the lack of knowledge of the model.

We propose to approximate the entropy of the epistemic part of the conditional posterior predictive distribution with the most common uncertainty measure, the Shannon entropy [10, 11]. The entropy of a mixture of multivariate t-distributions cannot be obtained analytically, so we approximate this mixture by a mixture of Gaussians using moment-matching. We propose to use the closed-form lower bound $H_{lower}(p^{ep}(\hat{\boldsymbol{w}}^o|\hat{\boldsymbol{w}}^i, \boldsymbol{W}))$ introduced in [12] for measuring the entropy of the Gaussian mixture, because it has been shown to be tight (see [9] for the complete equations).

We will now show how we can use the learned statistical model to build different active learning modalities.

III. Active learning modalities

In this section, we derive an intrinsically-motivated learning strategy. To facilitate the presentation of the approach, we will introduce the approach in the context of a specific robot experiment, where the aim is to learn to move an object to different positions. First, we present the task and the goal of the active learning framework. Then, we propose a method for active intrinsically-motivated learning.

A. Manipulation task

We present our approach in the context of learning to manipulate an object with a robot. The trajectory is composed of the robot joint states $\boldsymbol{\tau}^{\text{robot}}$ and the object position $\boldsymbol{\tau}^{\text{obj}}$, which implies that the ProMP weights $\boldsymbol{w}$ are a concatenation of robot weights $\boldsymbol{w}^{\text{robot}}$ and object weights $\boldsymbol{w}^{\text{obj}}$.

The goal of the task is to move the object to different desired final object positions $\boldsymbol{\tau}_{\text{des}}^{\text{obj},t=T}$. We denote the goal space $\mathcal{G}$ as the space of all desired final object positions we would like our robot to be able to generalize to. Formally, this means that there exists an unknown ground truth target distribution $p^{\text{GT}}(\boldsymbol{w}) = p^{\text{GT}}(\boldsymbol{w}^{\text{robot}}, \boldsymbol{w}^{\text{obj}})$ which can be used to generate robot movements $p^{\text{GT}}(\boldsymbol{w}^{\text{robot}}|\boldsymbol{\tau}_{\text{des}}^{\text{obj},t=T})$ that bring the object to the position $\boldsymbol{\tau}_{\text{des}}^{\text{obj},t=T}$.

We aim to learn this unknown joint distribution by combining imitation and intrinsically-motivated learning.

B. Intrinsically-motivated learning

We present here the learning modality, where the robot can try out a movement by itself and observe the environment changes in an open-ended manner. Namely, the robot chooses to execute a particular movement and observes the movement of the object. In contrast to imitation learning, one major advantage of intrinsically-motivated learning is that it does not require the presence of a human demonstrator.

We propose to select a robot movement based on how uncertain we are about the object movements it will cause. Formally, we would like to try the robot movement that maximizes the entropy of the epistemic part of the conditional distribution $p(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W})$, but this poses several problems. From a robotics point of view, doing so might pose safety problems as the movement retrieved might be very far from the underlying distribution $p^{\text{GT}}(\boldsymbol{w}^{\text{robot}})$ we aim to learn. From an active learning point of view, our active learning selection scheme is myopic and such criterion might select robot movements far away from the underlying distribution, i.e., where no generalization is required. For these reasons, we propose to use an information-density method [10]. Namely, we aim to find a robot movement that both has high information content (in the sense of the epistemic entropy), and that is close to the distribution of robot movements $p^{\text{robot}}(\boldsymbol{w}^{\text{robot}}|\boldsymbol{W})$:

$$\boldsymbol{w}^{\text{robot}^*} = \arg \max_{\boldsymbol{w}^{\text{robot}} \in \mathcal{W}^{\text{robot}}} \Big[H_{lower}\Big(p^{ep}(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W}) \Big) \\ + \beta p^{\text{robot}}(\boldsymbol{w}^{\text{robot}}) \Big], \tag{4}$$

where β is an hyperparameter weighting the relative importance of the two costs.

The full intrinsically-motivated learning algorithm is shown in Algorithm 1.

IV. Experiments

In this section, we show the usefulness of our intrinsically-motivated learning approach in the context of a waste throwing robotic task.

```
Algorithm 1: Active intrinsically-motivated learning
    Data: Movement database $\boldsymbol{W} = \{\boldsymbol{w}_i^{\text{robot}}, \boldsymbol{w}_i^{\text{obj}}\}_{i=1}^N$,
          robot movement space $\mathcal{W}^{\text{robot}}$
    Result: robot movement $\boldsymbol{w}^{\text{robot}^*}$ to execute

    Learn joint distribution of
      $p(\boldsymbol{w}|\boldsymbol{W}) = p(\boldsymbol{w}^{\text{robot}}, \boldsymbol{w}^{\text{obj}}|\boldsymbol{W})$ with BGMM;
    Calculate $p(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W})$;
    Isolate the epistemic uncertainty
      $p^{ep}(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W})$;
    Approximate the entropy of $p^{ep}(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W})$;
    Get the marginal distribution $p^{\text{robot}}(\boldsymbol{w}^{\text{robot}}|\boldsymbol{W})$ from
      $p(\boldsymbol{w}|\boldsymbol{W})$;

    Find $\boldsymbol{w}^{\text{robot}^*} =$
      $\arg\max_{\boldsymbol{w}^{\text{robot}} \in \mathcal{W}^{\text{robot}}} [H_{lower}(p^{ep}(\boldsymbol{w}^{\text{obj}}|\boldsymbol{w}^{\text{robot}}, \boldsymbol{W})) +$
      $\beta p^{\text{robot}}(\boldsymbol{w}^{\text{robot}})]$.
```

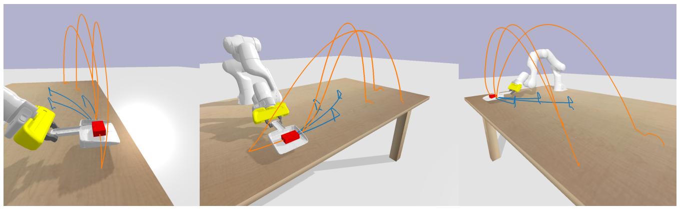

Fig. 1: Object trajectory for 6 demonstrations of the database (3 dynamic demonstrations in orange, and 3 non-dynamic demonstrations in blue).

A. *Waste throwing task*

We consider the task of throwing waste with a 7 DoF Franka Emika Panda robot simulated in pyBullet. This task is essential for the broader challenge of automatizing various forms of recycling. It is also relevant in diverse industrial applications requiring a robot to sort objects fast within a limited workspace.

An overview of the simulated setup can be seen in Fig. 1. The goal of the task is to be able to generate robot movements that bring a simulated can to different desired positions within a goal space $\mathcal{G}$. The particularity of this goal space is that, for a part of it, it is possible to bring the object with a non-dynamic movement because the desired final position is in the reachable robot workspace. However, for the rest of the goal space, the final desired object position is outside of the robot workspace, so that it requires the robot to throw the can with a dynamic movement. For benchmarking and reproducibility purposes, we build our experiments on a precomputed database of demonstrations. We create 200 non-dynamic demonstrations and 260 dynamic demonstrations using an oracle, that we gather in a database of demonstrations $\mathcal{D}$. In Fig. 1, we illustrate the can trajectory for three dynamic demonstrations and three non-dynamic demonstrations. In Fig. 2, we show the final can positions in our database, with the blue color representing the non-dynamic demonstrations and the orange color representing the dynamic demonstrations.

The trajectories of our database encode the robot movement at a frequency of 240Hz, with $T = 639$ timesteps, representing movements of about 3 seconds. We choose a 10-dimensional state space containing the 7 joint angle values of the robot, and the 3-dimensional Cartesian position of the can. In all experiments, we use $N = 30$ Gaussian radial basis functions[1] (RBFs) for ProMP. The width of the RBFs are set as $h = (\frac{T-1}{N})^2$, and the centers $\{c_m\}_{m=1}^D$ are evenly spaced between $-2h$ and $T+2h$. We choose a diagonal covariance matrix prior, with a standard deviation of 0.1 for the ProMP weights, and a mean concentration prior of 0.0001. We use a maximum number of 5 Gaussians, or strictly less than the number of demonstrations if there are less than 6 demonstrations. Other hyperparameters of the BGMM are the default hyperparameters of the *scikit-learn* library.

The maximization procedure in the active intrinsically-motivated learning is performed using a Bayesian optimization algorithm: the Tree-Structured Parzen Estimator approach (TPE) [13], implemented in the Python package hyperopt. A maximal number of iterations of 100 is used in the algorithm. As the space of possible robot movements is of high dimension (30 basis functions × 7 joint angles), we perform the search on the first two principal components of $\{\boldsymbol{w}_i^{\text{robot}}\}_{i=1}^N$, found by principal component analysis (PCA) The search space that we use is then the marginal distribution $p(\boldsymbol{w}^{\text{robot}})$ projected to the 2-dimensional PCA subspace.

We introduce an objective metric for evaluating our learning algorithm: the task cost, which is simply a ℓ_2 norm between the final object position and the desired object position, averaged over the goal space. In practice, we compute this task cost by computing the maximum *a posteriori* robot movement given a goal chosen over a uniform grid of 5×5 goals in the goal space, execute those 25 movements in simulation, and average the ℓ_2 norms between the final object positions and the desired object positions. Such a metric presents the advantage of being directly representative of the quality of the learned task. It is important to note here that this metric based on an external reward is used only for comparison, and not by our active learning algorithm.

B. *Intrinsically-motivated learning*

We present here the results of our intrinsically-motivated learning method. First, we would like to emphasize quantitatively the need for combining imitation learning and intrinsically-motivated learning for this waste throwing task. Namely, we want to show that using intrinsically-motivated learning can effectively reduce the task cost. We show in Fig. 3 the task cost (averaged over 20 demonstrations) for:

- 10 random demonstrations;
- 10 random demonstrations + 20 active intrinsically-motivated trials;
- 30 random demonstrations.

[1] Namely: $\Phi_m(t) = \frac{\phi_m(t)}{\sum_{n=1}^D \phi_n(t)}$ with $\phi_m(t) = \exp(-\frac{(t-c_m)^2}{2h})$.

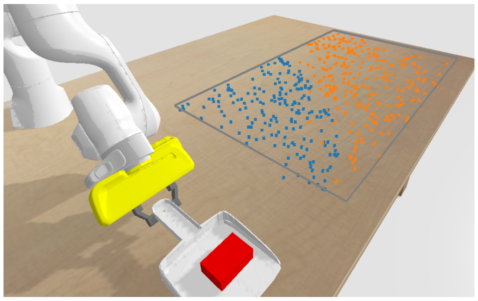

Fig. 2: Desired final object positions. The grey rectangle represents the goal space $\mathcal{G}$. Blue/orange dots show the final object position of respectively the non-dynamic/dynamic demonstrations of the database.

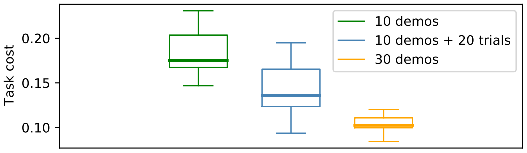

Fig. 3: Influence of demonstrations for intrinsically-motivated learning strategy.

We can see that, starting from 10 initial demonstrations, 20 intrinsically-motivated learning trials can improve the model. We can notably see that 20 intrinsically-motivated trials reduce the task cost half as well as 20 additional demonstrations. This shows that intrinsically-motivated learning can be used to reduce the burden of the human demonstrator by reducing the number of demonstrations s/he will be asked. Namely, Fig. 3 shows that intrinsically-motivated learning seems to be a good learning modality to be combined with imitation learning.

We propose now a baseline to compare our intrinsically-motivated learning method with:

- Random: This baseline computes the marginal $p(\boldsymbol{w}_{\text{robot}}|\boldsymbol{W})$ from the BGMM, and samples a robot movement from it. This seems like a reasonable baseline which already uses the correlations in the observed robot movements, and samples meaningful robot movements that are close to the observed demonstrations.

In Fig. 4, we show the performance of our method compared to this baseline, averaged over 20 experiments, and starting from 5 or 10 randomly sampled initial demonstrations. We can observe that our method presents a clear improvement over the baseline in both cases. Namely, the baseline deteriorates the task cost across the iterations, whereas our method permits to reduce the task cost, as observed in Fig. 3 (the mean task cost is reduced by around 20% after 10 autonomous trials in both cases). The deterioration of the task cost with the random approach can be explained by the fact that sampling from the marginal distribution of the robot movements at each iteration might end up with samples that are quite far from the original distribution, hence not useful for the task.

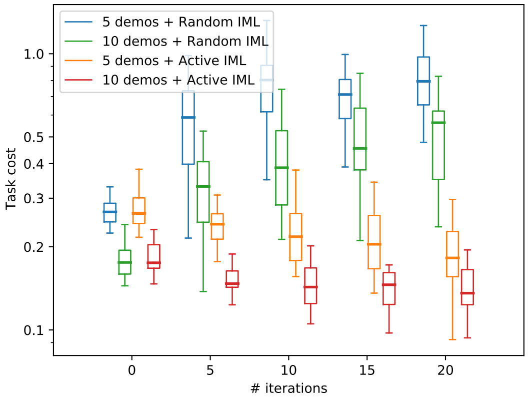

Fig. 4: Evaluation of intrinsically-motivated learning strategy (task cost in logarithmic scale).

V. Conclusion

In this work, we proposed a Bayesian representation of robot movements by extending the widely-used framework of probabilistic movement primitives. With this Bayesian representation, we proposed an intrinsically-motivated learning criterion, and showed its robustness on a waste throwing task with a 7-DoF simulated Franka Emika Panda robot.

The fundamental element of our method lies in that we model the joint distribution of the movement, and therefore can learn the model with demonstrations and/or autonomous robot trials. This permits us to leverage the variations observed in the human demonstrations for intrinsically-motivated learning.

References

[1] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE transactions on evolutionary computation*, vol. 11, no. 2, pp. 265–286, 2007.

[2] J. Schmidhuber, "Formal theory of creativity, fun, and intrinsic motivation (1990–2010)," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 3, pp. 230–247, 2010.

[3] R. W. White, "Motivation reconsidered: The concept of competence," *Psychological review*, vol. 66, no. 5, p. 297, 1959.

[4] E. L. Deci and R. M. Ryan, *Intrinsic Motivation and Self-Determination in Human Behavior*. Springer US, 1985.

[5] D. E. Berlyne, *Conflict, arousal, and curiosity.* McGraw-Hill Book Company, 1960.

[6] J. C. Horvitz, "Mesolimbocortical and nigrostriatal dopamine responses to salient non-reward events," *Neuroscience*, vol. 96, no. 4, pp. 651–656, 2000.

[7] J. Marshall, D. Blank, and L. Meeden, "An emergent framework for self-motivation in developmental robotics," *International Conference on Development and Learning*, 2004.

[8] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann, "Probabilistic movement primitives," in *Advances in Neural Information Processing Systems (NIPS)*, 2013, pp. 2616–2624.

[9] T. Kulak, H. Girgin, and S. Odobez, J.-M.and Calinon, "Active learning of Bayesian probabilistic movement primitives," *IEEE Robotics and Automation Letters*, 2021.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, 2006.

[11] C. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[12] A. Kolchinsky and B. Tracey, "Estimating mixture entropy with pairwise distances," *Entropy*, vol. 19, no. 7, p. 361, 2017.

[13] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Advances in neural information processing systems*, 2011, pp. 2546–2554.