



Project Type _____

- Master Thesis
- Bachelor Thesis
- Research Project

Supervisors _____

-  Onur Celik
-  celik@kit.edu

Difficulty _____

Algorithmic



Math



Application



Deep Reinforcement Learning of Versatile Behaviour

Description

Classic Reinforcement Learning (RL) aims to maximize the expected reward. These algorithms aim at finding one possible solution, which optimizes this task. However, it is often desired to find more versatile behavior that can solve a given task in different ways.



Figure 1: Grasping a cup in different ways [1]

An example is a robot, grasping a cup in different ways, e.g. grasping from right or from left, or grasping at different points of the cup to lift it.

Maximum Entropy Reinforcement Learning addresses this problem by adding an entropy term to the objective, which ensures that besides the average reward, also the entropy of the policy is maximized. This entropy-term leads to more versatile solutions.

Yet, we also need more complex, hierarchical policy architectures to represent versatile behavior. Typically, a Gaussian policy is used that can naturally only represent a single behavior plus uncorrelated noise. Hence, we want to use more complex policy structures, such as a Gaussian Mixture Model (GMM). A GMM for example is structured as single Gaussians (components) and a gating policy (e.g. a Softmax distribution). With this gating policy the agent is able to choose different components. A component represents a policy which can solve the task in a certain way. Introducing the gating variable as a latent random variable of the overall policy makes the optimization intractable.

However, it is possible to optimize each component of the policy independently by using a variational decomposition of the objective [2]. This decomposition reduces the optimization of a Mixture Model to optimizing each mixture component as well as the gating separately. For these individual optimization problems, we will use Soft-Actor-Critic (SAC) [3], which is a policy optimization algorithm addressing the maximum entropy objective.

Tasks

The tasks in this project will involve:

- **Literature review:** First, getting to know to the Reinforcement Learning algorithms with a detailed literature review in Hierarchical Reinforcement Learning and Policy Search Methods
- **Mathematical Derivation:** Clean and detailed derivation of the algorithm is of great importance
- **Implementing the Algorithm**
- **Evaluating:** We will evaluate our algorithm on challenging robotics tasks, such as reordering/manipulating an object into different positions and orientations
- **Comparison:** We will also choose baseline algorithms which we want to compare to. One baseline will cover single Soft-Actor-Critic without the hierarchy. To make a fair comparison, we will also use algorithms which exploit hierarchical structures.

References

- [1] Rosen Diankov | research and projects. <http://www.programmingvision.com/research.html>. Accessed: 2020-2-19.
- [2] Oleg Arenz, Mingjun Zhong, and Gerhard Neumann. Trust-region variational inference with gaussian mixture models. *arXiv preprint arXiv:1907.04710*, 2019.
- [3] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv preprint arXiv:1801.01290*, 2018.