Policy Search for Motor Primitives

Jens Kober, Jan Peters

Many motor skills in humanoid robotics can be learned using parametrized motor primitives from demonstrations. However, most interesting motor learning problems require self-improvement often beyond the reach of current reinforcement learning methods due to the high dimensionality of the state-space. We develop an EM-inspired algorithm applicable to complex motor learning tasks. We compare this algorithm to several well-known parametrized policy search methods and show that it outperforms them. We apply it to motor learning problems and show that it can learn the complex Ball-in-a-Cup task using a real Barrett WAM™ robot arm.

1 Introduction

Humans demonstrate a large variety of complex motor skills in their day-to-day life. Their agility and adaptability to new control tasks remains unmatched by the millions of robots laboring on factory floors and roaming research labs. Achieving the abilities of learning new motor skills is an essential component in order to get a step closer to human-like motor skills. If future robots were able to acquire their basic tasks by imitating human demonstrations and, subsequently, self-improve by trial and error, such robot learning would result in more wide-spread robot application as well as large productivity gains in industry.

Recent progress in the area of machine learning has yielded several important tools for making progress towards this vision for the future. Dynamical system-based motor primitives [1] have enabled robots to learn complex tasks ranging from Tennis-swings [1] to legged locomotion [2] by imitation. However, most interesting motor learning problems are high-dimensional reinforcement learning problems often beyond the reach of standard methods. Nevertheless, in reinforcement learning, policy search, also known as policy learning, has become an accepted alternative of value function-based approaches [3]. In high-dimensional domains with continuous states and actions, such as robotics, this approach has previously proven superior as it allows the usage of domain-appropriate pre-structured policies, the straightforward integration of a teacher's presentation as well as fast online learning [3–5]. We develop a novel EM-inspired reinforcement learning algorithm particularly well-suited for dynamic motor primitives. We show that the presented algorithm works well when employed in the context of learning dynamic motor primitives in two different settings, i.e., the Underactuated Swing-Up [6] and the complex task of Ball-in-a-Cup [7]. Both tasks are achieved on a real Barrett WAM™ robot arm. Please also refer to the video on the first author's website[1].

2 Reinforcement Learning for Motor Primitives

Reinforcement learning of discrete motor primitives is a hard, specific type of learning problem where it is difficult to apply generic reinforcement learning algorithms [4]. For this reason, we focus largely on domain-appropriate reinforcement learning algorithms which operate on parametrized policies for episodic control problems.

In [8] we derive a framework of reward weighted imitation. Based on [9] we consider the expected return of an episode as an improper probability distribution. We maximize a lower bound of the logarithm of the expected return. Depending on the strategy of optimizing this lower bound the framework yields several well known policy search approaches: Episodic REINFORCE [10], the Policy Gradient Theorem [11], Natural Actor Critic approaches [4], a generalization of the Reward-Weighted Regression [5] as well as our novel Policy learning by Weighting Exploration with the Returns (PoWER) algorithm.

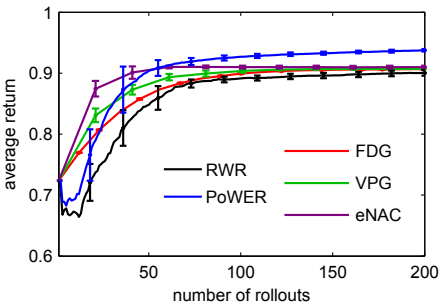

Figure 1: This figure shows the performance of all compared methods for the swing-up in simulation where the solid line represents the mean performance averaged over 20 learning runs with the error bars indicating the standard deviation. Policy learning by Weighting Exploration with the Returns (PoWER) clearly outperforms Finite Difference Gradients (FDG), 'Vanilla' Policy Gradients (VPG), the Episodic Natural Actor Critic (eNAC) and the adapted Reward-Weighted Regression (RWR) from 50 rollouts on and finds a significantly better policy in every run.

3 Application to Robotics

While the resulting algorithms are generally applicable, we demonstrate the effectiveness of the novel algorithm in the context of motor primitive learning for robotics. As a first evaluation, we will show that the novel PoWER algorithm

[1] http://www.kyb.mpg.de/~kober

Auszug aus: Künstliche Intelligenz, Heft 3/2009, ISSN 0933-1875, BöttcherIT Verlag, Bremen, **www.kuenstliche-intelligenz.de/order**

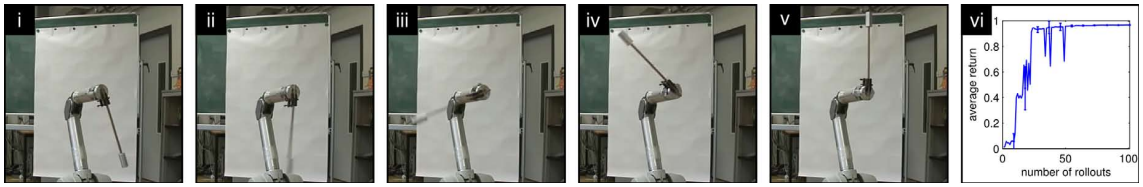

Figure 2: This figure shows the time series of the Underactuated Swing-Up where only a single joint of the robot is moved with a torque limit enforced by capping the motor current of that joint. The resulting motion requires the robot to (i) first move away from the target to reduce the maximal required torque during the swing-up in (ii-iv) and subsequent stabilization (v). The performance of the PoWER method on the real robot is shown in (vi).

outperforms many previous well-known methods such as 'Vanilla' Policy Gradients [4], Finite Difference Gradients [4], the Episodic Natural Actor Critic [4] and the generalized Reward-Weighted Regression [5] on a simulated Underactuated Swing-Up [6]. Real robot applications are done with our best benchmarked method, the PoWER method. Here, we first show PoWER can learn the Underactuated Swing-Up [6] even on a real robot. As a significantly more complex motor learning task, we show how the robot can learn a high-speed Ball-in-a-Cup [7] movement with motor primitives for all seven degrees of freedom of our Barrett WAM™ robot arm.

As policy representation, we employ dynamical system-based motor primitives [1] with added exploration. In Sections 3.1 and 3.2, we use imitation learning for the initialization. This is straightforward as the motor primitives are linear in parameters and is done with locally-weighted regression to solve for an imitation from a single example [1]. Subsequently, we improve the policy by reinforcement learning.

3.1 Underactuated Swing-Up

As simulated benchmark and for the real-robot evaluations, we employed the Underactuated Swing-Up [6]. Here, only a single degree of freedom is represented by a motor primitive. The goal is to move a hanging heavy pendulum to an upright position and stabilize it there with minimal motor torques. By limiting the motor current for that degree of freedom, we can ensure that the torque limits described in [6] are maintained and directly moving the joint to the right position is not possible. This process is illustrated in Figure 2 (i-v). The problem is similar to a mountain-car problem where the car would have to stop on top or experience a failure.

The applied torque limits and reward function are the same as in [6]. The motor primitive with ten parameters was initialized by imitation learning. Subsequently, we compared our novel algorithm to several established algorithms and could show that PoWER would outperform them. All open parameters were manually optimized. The results are given in Figure 1. As it turned out to be the best performing method, we then used it successfully for learning optimal swing-ups on a real robot. See Figure 2 (vi) for the resulting real-robot performance.

3.2 Ball-in-a-Cup on a Barrett WAM™

The most challenging application in this project is the children's game Ball-in-a-Cup [7] where a small cup is attached to the robot's end-effector and a small wooden ball is connected to this cup by a 40cm string. Initially, the ball is hanging down vertically. The robot needs to move fast in order to induce a motion at the ball through the string, swing it up and catch it with the cup, a possible movement is illustrated in Figure 3 (top row). The robot is controlled in joint space. The ball is tracked by a stereo vision system. The return is based on the minimal distance between the ball and the cup. The task is quite complex as a small perturbation of the initial condition or during the trajectory will drastically change the movement of the ball and hence the outcome of the rollout.

Due to the complexity of the task, Ball-in-a-Cup is even a hard motor learning task for children who often succeed only after observing a successful demonstration and self-improvement by trial-and-error. Mimicking how children learn to play Ball-in-a-Cup, we first initialize the motor primitives by imitation and, subsequently, improve them by reinforcement learning. We recorded the motions of a human player by kinesthetic teach-in in order to obtain an example for imitation as shown in Figure 3 (top row). From the imitation, the initial motor primitive parameters can be determined as well as the movement duration. As expected, the robot fails to reproduce the presented behavior successfully and reinforcement learning is needed for self-improvement. The robot regularly succeeds at swinging the ball into the cup after approximately 75 iterations. See Figure 3 (bottom row) for the learning performance.

4 Conclusion

We successfully applied the novel PoWER algorithm in the context of learning two tasks on a physical robot, i.e., the Underacted Swing-Up and Ball-in-a-Cup. Due to the curse of dimensionality, we require a good initial policy and cannot start with random policy parameters. Instead, we mimic the way children learn Ball-in-a-Cup and, first, present an example for imitation learning which is recorded using kinesthetic teach-in. Subsequently, our reinforcement learning algorithm takes over and learns how to move the ball into the cup reliably. After only realistically few episodes, the task can be successfully executed and the robot shows very good average performance with only successful trials. Future applications will include the optimization of forehands and backhands for table tennis.

References

[1] A. Ijspeert, J. Nakanishi, and S. Schaal. Learning attractor landscapes for learning motor primitives. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.

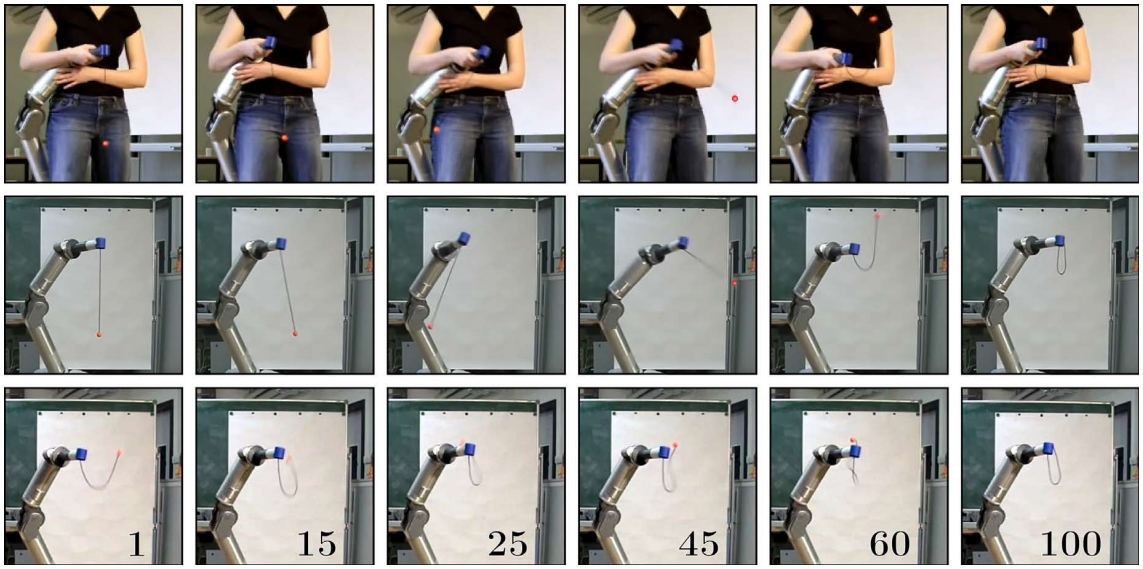

Figure 3: This figure shows a kinesthetic teach-in (top row, frames during a demonstration), the learned robot motion after convergence (middle row, frames during execution) as well as the learning progress (bottom row, the frames show the position of the ball closest to the cup during the indicated trial). After imitation learning the robot manages to reproduce the demonstrated motion quite accurately, but the ball misses the cup by several centimeters (bottom left, trial 1). After ca. 75 iterations of our PoWER algorithm (indicated by trials 15, 25, 45 and 60) the robot has improved its motion so that the ball goes in the cup (bottom right, trial 100).

[2] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. Learning from demonstration and adaptation of biped locomotion. *Robotics and Autonomous Systems (RAS)*, 47(2-3):79–91, 2004.

[3] J. Bagnell, S. Kadade, A. Ng, and J. Schneider. Policy search by dynamic programming. In *Advances in Neural Information Processing Systems (NIPS)*, 2003.

[4] J. Peters and S. Schaal. Policy gradient methods for robotics. In *International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[5] J. Peters and S. Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *International Conference on Machine Learning (ICML)*, 2007.

[6] C. G. Atkeson. Using local trajectory optimizers to speed up global optimization in dynamic programming. In *Advances in Neural Information Processing Systems (NIPS)*, 1994.

[7] Wikipedia, January 31, 2009. `http://en.wikipedia.org/wiki/Ball_in_a_cup`

[8] J. Kober and J. Peters. Policy Search for Motor Primitives in Robotics. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

[9] P. Dayan and G. E. Hinton. Using expectation-maximization for reinforcement learning. *Neural Computation*, 9(2):271–278, 1997.

[10] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

[11] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NIPS)*, 2000.

Contact

Jens Kober, Jan Peters
MPI for Biological Cybernetics, Dept. Schölkopf
Spemannstr. 38, 72076 Tübingen
Tel./Fax: +49 (0)7071 601-585/-552
{jens.kober,jan.peters}@tuebingen.mpg.de

Jens Kober is a Ph.D. student at the Robot Learning Lab (RoLL) at the Max-Planck Institute for Biological Cybernetics. He graduated from the Ecole Centrale Paris (ECP) and holds a German M.S. degree in Engineering Cybernetics from Stuttgart University.

Jan Peters is heading the Robot Learning Lab (RoLL) at the Dept. for Empirical Inference and Machine Learning of the Max Planck Institute for Biological Cybernetics. He is also an adjunct researcher of the Computational Learning and Motor Control Lab at the University of Southern California. He holds a Ph.D. in computer science from USC as well as four masters degrees all related to robotics and machine learning.