Biologically-Inspired Deep Reinforcement Learning of Modular Control for a Six-Legged Robot

Kai Konen, Timo Korthals, Andrew Melnik, Malte Schilling

I. Introduction

Deep Reinforcement Learning (DRL) has been applied successfully in an increasing number of areas, ranging from computer games towards robotic control [1]. While such Deep Learning approaches have shown to produce viable solutions, these are still struggling when we try to scale them up towards complex bodies or when these approaches should be applied in more natural contexts that require behavioral flexibility. In these areas, there is still a major discrepancy between current state-of-the-art DRL approaches and biological systems [5] that excel at adaptive behavior. Therefore, insights on the structure of biological control systems appears as a promising approach for guiding DRL approaches [2].

Our goal is to address as an example the complex control problem of a six-legged walking agent. Such a problem is difficult, as there are many coupled degrees of freedom that have to be controlled in a coordinated way. Such a problem appears difficult to solve through a basic DRL approach as the space for exploration is growing way too large. Therefore, we will take a modular and hierarchical control approach for locomotion that is inspired by the structure of walking control as found in animals [6]: First, the complete control problem is divided into smaller sub-problems. Research on insects have found a decentralized control approach that suggests as a basic building block a division into individual leg controllers. Secondly, this induces a hierarchy. There is a higher level that selects a particular behavior to perform. A lower level takes care of motor control on the joint level. We will introduce such a modular structure into a neural network controller and train this system using reinforcement learning.

The modularity of this approach and in particular the decentralized aspect set this apart from other current hierarchical DRL approaches [4]. Each module only has to control, on the lower level, a small number of coupled degrees of freedom (this is comparable to an approach as are motor synergies and which we applied in a DRL approach in [3]). On the higher level that deals with selection of motor behaviors, a form of coordination is required. In the presented approach, we will use a fixed form of coordination and analyze learning performance. In the future, we want to extend this towards a research project that is also taking concurrency on the higher level into account: each leg should have its own individual controller which has as a scope the current state, local sensory input and some additional influences for coordination, but only from neighboring legs [7].

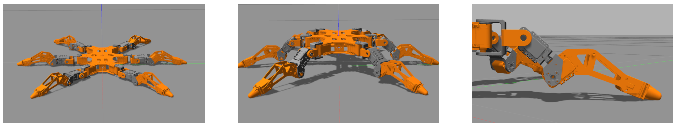

Fig. 1: Shown is the simulated PhantomX robot lying down, standing up, and a leg under a DoF lock.

In this article, we want to present the general setup and early results of our approach: First, we will introduce the simulated six-legged robot. Secondly, we will show as a simple example for reinforcement-based learning for such an agent how the agent can learn standing up when employing a modular approach. But, furthermore, how walking requires a form of coordination of leg controllers.

II. DRL for a Hexapod Robot

The goal of our approach is learning walking behavior for a six-legged robot. We are using Gazebo and the Open Dynamics Engine to simulate the PhantomX hexapod robot [9]–[11]. The PhantomX robot has six legs, each leg consists of three links and each link's position is controlled by a single joint controller. Overall the robot consists of 18 degrees of freedom. Movement is controlled by a ROS-Node which applies torque to each joint.

The task is to learn a continuous controller and we are applying Deep Deterministic Policy Gradient (DDPG) [8] for learning. DDPG is an off-policy algorithm for continuous action spaces. It is an actor-critic method which learns a policy and a Q-function. The actor is used to learn a policy and the critic evaluates the actions taken by the actor.

Using one DDPG agent to control all joints simultaneously depicts a complex learning challenge. With a continuous action space of 18 dimensions and a continuous observation space of 42 dimensions, the agent is not able to learn a well performing policy for a task as walking forward. To reduce the complexity, we took a modular approach. First, learning as a module a controller for a single leg. Such a structure was trained in a simple standing up task. In this task, all legs can act the same way and this one modular controller was used in multiple (but all identically structured instances) to produce behavior.

This research was supported by the Cluster of Excellence Cognitive Interaction Technology 'CITEC' (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG).

The authors are with the Center of Excellence 'Cognitive Interaction Technology', Bielefeld University (Germany). Corresponding Email: mschilli@techfak.uni-bielefeld.de

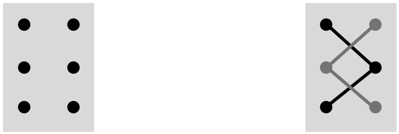

Fig. 2: Two controller layouts are shown (controllers are shown as black dots). Left: Single leg - one agent controls each leg, imposed to all legs. Right: Tripod - control for two legs is learned, imposed to groups of three legs each.

But such an approach would not work for locomotion, as in walking legs are not simply producing the exact same behavior. Instead, they are working together in different groups. Therefore, as a second approach, we trained a set of two individual leg controllers, each of which was applied on three diagonal legs. This fixed structure of legs that act together can be observed in fast walking insects (Fig. 2).

For the single leg controller the action space consisted of two joint torques — the femur and tibia that are actuated in standing up. The coxa was fixed at a set position. The observation space was reduced to the joint angles and velocities of femur and tibia of a single leg, as well as height, roll and pitch of the torso. The actions where then imposed to all six legs.

The tripod approach consisted of an overall action space of six joint torques: Coxa, femur and tibia torques for two leg groups. The actions where imposed to a set of three legs each. The observation space consisted of coxa, femur and tibia joint velocity and angle for each leg, as well as height, roll and pitch of the torso.

To evaluate the different controllers, two tasks were chosen. Standing up: The torso of the robot had to reach a specified height and stay at that height for the remainder of the episode. Walking forwards: The robot had to cover as much distance forward as possible in a set time frame. As an auxiliary task, a small reward was given for lifting the torso off the ground. The observation space consisted of coxa, femur and tibia joint velocities and angles for two legs, as well as height and yaw of the torso and the distance from the initial robot position.

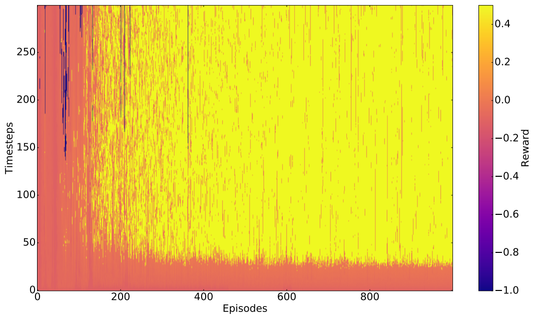

Fig. 3: Reward over time during training. On the horizontal axis, individual training episodes are given. Vertical axis shows progressing of time during an episode. Color coding represents the reward for a point in time during that episode.

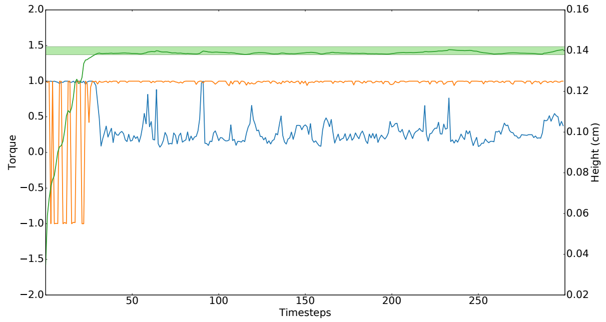

Fig. 4: Test of the controller for standing up at the end of training. Shown are tibia and femur torques (blue and orange, left axis) and the height and goal height of the robot (green, plotted on the right axis).

III. Results and Conclusion

With the single leg controller, the agent was able to learn a policy for the standing up task (Fig. 3). The agent learned to reduce the DoFs and simplify the control problem by locking different joints by applying maximum torque (as seen in Fig. 1). During the task, the agent locked the tibia joint once he had reached the goal height and only adjusted the femur joint torque to balance changes in height (Fig. 4, for a video see [12]). The results of four training session with 1000 episodes at 300 timesteps each are shown in table I.

	T 1	T 2	T 3	T 4	T 5	T 6
ØReward	0.323	0.271	-0.068	0.006	0.068	0.006
SD	0.271	0.280	0.081	0.0255	0.237	0.291

TABLE I: Shown are the mean reward and standard deviation of each training session. The reward range was chosen to be between -1 and 0.5. Training 1-4 were single leg controllers, training 5 and 6 were using the tripod arrangement.

The single leg and tripod controller converged against similar policies for the standing up task. Albeit the tripod training did not converge as quickly and the agent needed more time to reach the goal height, it learned to use the joint lock mechanism and applied similar torques to the different femur and tibia joints.

For the walking forward task, only the tripod controller was tested. The learned policy was able to move the robot forward, but only in a limited fashion. It learned a swinging trajectory for the legs to move itself forward, but failed to properly keep the torso off the ground, resulting in a crawling motion (not shown in detail here).

Our results show that a modular approach—compared to controlling every joint independently—increases the ability to learn motions like standing up and walking forward. In the future, we will use a flexible hierarchical structure and modular controllers for each leg (controlling coxa, femur and tibia joint torques). This reduces the action space per agent and omits observations not directly necessary for individual legs.

REFERENCES

[1] Arulkumaran K., Deisenroth M.P., Brundage M., Bharath A.A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Process. Mag.* **34**, 26-38.

[2] Hassabis D., Kumaran D., Summerfield C., Botvinick M. (2017). Neuroscience-Inspired Artificial Intelligence. *Neuron*, **95**(2), 245–258.

[3] Kidziński Ł. et al. (2018). Learning to Run Challenge Solutions: Adapting Reinforcement Learning Methods for Neuromusculoskeletal Environments. In: Escalera S., Weimer M. (eds) The NIPS '17 Competition: Building Intelligent Systems. The Springer Series on Challenges in Machine Learning. Springer, Cham.

[4] Kulkarni T., Narasimhan K., Saeedi A., Tenenbaum J. (2016). Hierarchical deep rein- forcement learning: Integrating temporal abstraction and intrinsic motivation. In NIPS, p. 36753683.

[5] Neftci E.O. and Averbeck B.B. (2019). Reinforcement learning in artificial and biological systems. *Nature Machine Intelligence*, **1**.

[6] Schilling M., Hoinville T., Schmitz J., Cruse H. (2013). Walknet, a bio-inspired controller for hexapod walking. *Biol. Cybern.* **107**(4), 397–419.

[7] Schilling M., Melnik A. (2018). An Approach to Hierarchical Deep Reinforcement Learning for a Decentralized Walking Control Architecture In: Biologically Inspired Cognitive Architectures 2018. Proc. of the Ninth Annual Meeting of the BICA Society. Samsonovic AV (Ed); Advances in Intelligent Systems and Computing, 848:272–282. Springer, Cham.

[8] Lillicrap T., Hunt J., Pritzel A., Heess N., Erez T., Tassa Y., Silver D., Wierstra D. (2016). Continuous control with deep reinforcement learning, 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico.

[9] https://github.com/kkonen/phantomx_description

[10] https://github.com/kkonen/phantomx_control

[11] https://github.com/kkonen/phantomx_gazebo

[12] https://github.com/kkonen/phantomx_gazebo/blob/master/standing_up.mp4