Learning to Walk in Every Direction with the TBR-Learning algorithm

Antoine Cully\textsuperscript{1,2} and Jean-Baptiste Mouret\textsuperscript{1,2}

\textsuperscript{1}Sorbonne Universités, UPMC Univ Paris 06, UMR 722, ISIR, F-75005, Paris, France
\textsuperscript{2}CNRS, UMR 7222, ISIR, F-75005, Paris, France
mouret@isir.upmc.fr

Introduction

Legged robots are versatile machines that can outperform wheeled robots on rough terrain (Raibert, 1986), for instance in exploration or rescue missions. Their versatility is, however, tempered by their mechanical and control complexity, which makes them prone to mechanical damages and difficult to control robustly (Raibert, 1986; Bongard et al., 2006; Koos et al., 2013a). A promising way to compensate for these two weaknesses is to let robots discover on their own the best way to move in the current situation. A legged robot can thus cope with an unexpected terrain or with mechanical damages by learning a new walking gait (Bongard et al., 2006; Koos et al., 2013a), in the same way as animals can learn to limp with a sprained ankle.

Reinforcement learning (Kohl and Stone, 2004; Tedrake et al., 2005) and evolutionary algorithms (Zykov et al., 2004; Chernova and Veloso, 2004; Hornby et al., 2005) have been investigated to discover walking gaits for physical robots. Nevertheless, most of these investigations are limited to straight, forward walking, whereas a robot that only walks along a straight line is obviously unable to accomplish any mission. Only a handful of works deal with controllers able to turn or to change the walking speed. In these cases, controllers are successively evaluated on each possible direction (Mouret et al., 2006), or learned with an incremental process (Kodjabachian and Meyer, 1998). Compared to learning a simple controller, these two approaches significantly increase the learning time and the complexity of the search process.

In the present paper, we describe the Transferability-based Behavioral Repertoire Evolution (TBR-Evolution), a new learning algorithm that allows a robot to learn to walk in every direction in a single run of evolutionary algorithm. This algorithm combines the BR-Evolution algorithm (Cully and Mouret, 2013), which creates a behavioral repertoire in a single run, with the transferability approach (Koos et al., 2013b), which minimizes the number of evaluations on a physical robot when evolving controllers thanks to a simulator. A behavioral repertoire is a collection of simple controllers, where each of them reaches one position. An external planning algorithm can thus successively pick up repertoire’s controllers to generate a desired trajectory. Since TBR-Evolution does not depend on the controllers’ architecture, it can be exploited in combination with most of the previous work about gait evolution.

Experiments

We evaluate the TBR-Evolution on a physical hexapod robot with 18 degrees of freedom (fig. 2B). The behaviors on the physical robot are assessed on-board thanks to a RGB-D sensor coupled with a state-of-the-art SLAM algorithm (Endres et al., 2012). For each TBR-Evolution experiment, a population of 100 solutions evolves during 3000 generations in simulation. After only 60 tests on the robot (one every 50 generations), the algorithm generates a collection with a
Figure 2: (A) Typical collection of controllers obtained with the TBR-Evolution algorithm. (Left) Endpoints in simulation of each controllers of the collection. The red lines are the final orientations of the robot. (Center) Estimated transferability map. The estimated transferability of each behavior of the collection is considered within a radius of 5 cm. (Right) Execution on the physical robot. we selected 30 controllers in the repertoire (square) and executed them on the physical robot (circles). The size and the color of the markers are proportional to their accuracy. (B) Illustration of 5 typical trajectories obtained with the TBR-Evolution.

The median number of 375 controllers (min=353, max=394, five replications) allowing the robot to walk in every direction (fig. 2B). The periodical tests aim to drive the process to solutions that work similarly in simulation and in reality (i.e. crossing the reality gap, Koos et al. (2013b)). One of these collections is pictured in figure 2A.

The archive covers a large portion of the space around the robot (fig. 2A) and has an actual transferability value below 15 cm (difference between endpoints in the simulation and in reality. Fig. 1, left), which is consistent with the observations of the transferability map (Fig. 2, center) and not very large, taking into account the SLAM precision, the size of the robot and the looseness in the joints.

The source-code of our experiments and a supplementary video can be downloaded from: http://pages.isir.upmc.fr/evorob_db/

Acknowledgements
This work has been funded by the ANR Creadapt project (ANR-12-JS03-0009) and a DGA scholarship to A. Cully.

References


