Reinforcement learning for repetitive systems with discrete sensor information

S. Goossens, G. Pinte, W. Symens
FMTC (Flanders' Mechatronics Technology Centre)
Celestijnenlaan 300D, 3001 Heverlee, Belgium
Email: stijn.goossens@fmtc.be

M. Gagliolo, A. Rodriguez, A. Nowé
CoMo, VUB (Vrije Universiteit Brussel)
Pleinlaan 2, 1050 Brussels, Belgium
Email: mgagliol@vub.ac.be

1 Introduction

In most existing motion control algorithms, a reference trajectory is tracked based on a continuous measurement of the system's response. However, in some industrial applications, it is either not possible or too expensive to install a sensor which measures the system's output over the complete stroke. The motion is only detected at certain discrete positions. The control objective in these systems is often not to follow a complete trajectory accurately, but to realize a certain state of the output at the sensor locations (e.g. to pass by the sensor at a certain time, or to obtain a certain speed at the sensor location). Model-based control strategies are not suited for the control of these systems, since it is not straightforward to derive a reliable plant model, due to the lack of sensor data. Here we investigate the potential of a non-model-based learning strategy, Reinforcement Learning (RL), in dealing with this kind of discrete sensor information.

2 Reinforcement learning

RL problems [1] are a class of machine learning problems, where an agent must learn to interact with an unknown environment, using a "trial and error" approach. At a given timestep t, the agent may execute one of a set of *actions* $a \in \mathcal{A}$, possibly causing the environment to change its *state* $s \in \mathcal{S}$, and generate a (scalar) *reward* $r \in \mathbb{R}$. An agent is represented by a *policy*, mapping states to actions. The aim of a RL algorithm is to optimize the policy, maximizing the reward accumulated by the agent.

Simply stated, RL consists in learning from a teacher (the environment) who cannot tell us *what* to do next (the optimal policy), but only *how good* we are doing so far (the reward signal). It therefore offers a suitable framework for the control of systems with discrete sensor information. The desired state at the discrete sensor positions can be incorporated in the reward signal. The control output is represented as a parameterized signal and the action set consists of (continuous or discrete) adaptations of the parameters of this signal. After each time that the complete control signal is applied and the discrete sensor signals are available, the policy is updated based on the new reward signal.

3 Experimental validation

The potential of different RL techniques is validated on a set-up consisting of a linear motor and a moving mass mounted on a linear guide (Fig. 1). The position of the moving mass is monitored via a single discrete sensor set along this guide. When the motor is activated, the mass is pushed forward and slides to a certain position. The objective is to find the motor control signal which passes the mass in front of the sensor at a predefined time. On top of this, the dissipated energy should be minimized and the controller should be robust to system variations.

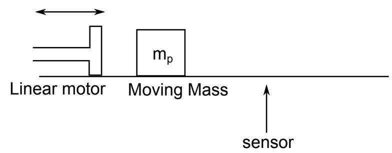

Figure 1: Sketch set-up, position measurement at one location

Several RL methods are evaluated on this setup. Two are based on value function estimation: SARSA [1], where actions are discretized, and Ex$< a >$ [2], where continuous action values are estimated using a non parametric model. Two are based on direct policy search, both representing their policy as a probability density function (pdf) over actions, which is updated during learning: a policy gradient method [3], where the actions are drawn from a Gaussian distribution; and a learning automaton [4], where the pdf over the actions is non-parametric.

4 Acknowledgement

This work has been carried out within the framework of the LeCoPro project (grant nr. 80032) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

References

[1] R. Sutton, A. Barto, "Reinforcement learning: An introduction," MIT Press, 1998.

[2] J.A. Martn, J. de Lope, "Ex$< a >$: An Effective Algorithm for Continuous Actions Reinforcement Learning Problems," Universidad de Madrid, 2009.

[3] D. Wierstra, A. Förster, J. Peters, J. Schmidhuber. "Recurrent Policy Gradients," Logic Journal of IGPL, 18:620-634, 2010.

[4] M.N. Howell, G.P. Frost, T.J. Gordon, Q.H. Wu, "Continuous action reinforcement learning applied to vehicle suspension control", Mechatronics, 7(3):263-276. 1997.