INTRODUCTION

Reinforcement learning is a numerical approach to learning through interaction. An algorithmic agent is tasked with determining a policy that provides a large cumulative reward. There are two groups which the agent learns its policy: Model-free reinforcement learning or Model-based reinforcement learning.

- In model-free reinforcement learning, the agent ignore the dynamics of the environment. It only relies on its state to make decisions.
- In model-based reinforcement learning, the agent possesses a model. The agent uses this model to reason about the implications of its decisions and plan its behavior.

The model-based approach to reinforcement learning offers two significant advantages.

1. Domains in which acquiring experience is expensive. Model-based methods can leverage planning to do policy improvement without requiring further samples from the environment.
2. The regime in which capacity for function approximation is limited and the optimal value function and policy cannot be represented.

However, in model-based reinforcement learning, planning with an incomplete or an unfinished model of the environment has the potential to harm learning progress. Dyna-style reinforcement learning is a powerful approach for problems where not much real data is available.

Efficient decision making when interacting with an incompletely known world can be thought of as an online learning and planning problem. Each interaction provides additional information. The planning process should be repeated to take this into account. However, planning is a complex process; on large problems it not possible to repeat it on every time step.

There are two ideas underlying the Dyna architecture.

- Planning, taking action and learning are continuous, they work as fast as possible without waiting for each other.
- Learning and planning are similar. Planning in the Dyna architecture consists of using the model to generate imaginary experience and then processing the transitions of the imaginary experience by model-free reinforcement learning algorithms as if they had actually occurred.

This paper develops an explicitly model-based approach extending the Dyna architecture to linear function approximation. Dynastyle planning continues by generating imaginary experience from the model and then by applying model-free reinforcement learning algorithms to the imagined state transitions. Here, they introduce two versions of prioritized sweeping with linear Dyna and show their performance empirically on the Mountain Car and Boyan Chain problems. I will investigate these later.

References

Silver, D., Sutton, R. S., and Muller, M. Sample-based ¨ learning and search with permanent and transient memories. In Proceedings of the International Conference on Machine Learning, pp. 968–975, 2008.

Sutton, R. S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming, Proceedings of the Seventh International Conference on Machine Learning, pp. 216– 224.

http://proceedings.mlr.press/v119/abbas20a/abbas20a.pdf

https://arxiv.org/pdf/1206.3285.pdf