Chapter 16
Particle Swarm Optimization for Model Predictive Control in Reinforcement Learning Environments

Daniel Hein
Technische Universität München, Germany

Alexander Hentschel
AxiomZen, Canada

Thomas A. Runkler
Siemens AG, Germany

Steffen Udluft
Siemens AG, Germany

ABSTRACT

This chapter introduces a model-based reinforcement learning (RL) approach for continuous state and action spaces. While most RL methods try to find closed-form policies, the approach taken here employs numerical online optimization of control action sequences following the strategy of nonlinear model predictive control. First, a general method for reformulating RL problems as optimization tasks is provided. Subsequently, particle swarm optimization (PSO) is applied to search for optimal solutions. This PSO policy (PSO-P) is effective for high dimensional state spaces and does not require a priori assumptions about adequate policy representations. Furthermore, by translating RL problems into optimization tasks, the rich collection of real-world-inspired RL benchmarks is made available for benchmarking numerical optimization techniques. The effectiveness of PSO-P is demonstrated on two standard benchmarks mountain car and cart-pole swing-up and a new industry-inspired benchmark, the so-called industrial benchmark.

DOI: 10.4018/978-1-5225-5134-8.ch016

INTRODUCTION

This chapter focuses on a general reinforcement learning (RL) setting with continuous state and action spaces. In this domain, the policy performance often strongly depends on the algorithms for policy generation and the chosen policy representation (Sutton & Barto, 1998). In the authors' experience, tuning the policy learning process is generally challenging for industrial RL problems. Specifically, it is hard to assess whether a trained policy has unsatisfactory performance due to inadequate training data, unsuitable policy representation, or an unfitting training algorithm. Determining the best problem-specific RL approach often requires time-intensive trials with various policy configurations and training algorithms. In contrast, it is often significantly easier to train a well-performing system model from observational data, compared to directly learning a policy and assessing its performance.

The main purpose of the present contribution is to provide a heuristic for solving RL problems which employs numerical online optimization of control action sequences. As an initial step, a neural system model is trained from observational data with standard methods. However, the presented method also works with any other model type, e.g., Gaussian process or first principal models. The resulting problem of finding optimal control action sequences based on model predictions is solved with particle swarm optimization (PSO), because PSO is an established algorithm for non-convex optimization. Specifically, the presented heuristic iterates over the following steps. (1) PSO is employed to search for an action sequence that maximizes the expected return when applied to the current system state by simulating its effects using the system model. (2) The first action of the sequence with the highest expected return is applied to the real-world system. (3) The system transitions to the subsequent state and the optimization process are repeated based on the new state (go to step 1).

As this approach can generate control actions for any system state, it formally constitutes an RL policy. This PSO policy (PSO-P) deviates fundamentally from common RL approaches. Most methods for solving RL problems try to learn a closed-form policy (Sutton & Barto, 1998). The most significant advantages of PSO-P are the following. (1) Closed-form policy learners generally select a policy from a user-parameterized (potentially infinite) set of candidate policies. For example, when learning an RL policy based on tile coding (Sutton, 1996), the user must specify partitions of the state space. The partition's characteristics directly influence how well the resulting policy can differentiate the effect of different actions. For complex RL problems, policy performances usually vary drastically depending on the chosen partitions. In contrast, PSO-P does not require a priori assumptions about problem-specific policy representations, because it directly optimizes action sequences. (2) Closed-form RL policies operate on the state space and are generally affected by the *curse of dimensionality* (Bellman, Adaptive Control Processes: A Guided Tour, 1962). Simply put, the number of data points required for a representative coverage of the state space grows exponentially with the state space's dimensionality. Common RL methods, such as tile coding, quickly become computationally intractable with increasing dimensionality. Moreover, for industrial RL problems it is often very expensive to obtain adequate training data prohibiting data-intensive RL methods. In comparison, PSO-P is not affected by the state space dimensionality because it operates in the space of action sequences.

From a strictly mathematical standpoint, PSO-P follows a known strategy from nonlinear model predictive control (MPC): employing online numerical optimization in search for the best action sequences. While MPC and RL target almost the same class of control optimization problems with different methods, the mathematical formalisms in both communities are drastically different. Particularly, the authors find that the presented approach is rarely considered in the RL community. The main contribution of