Learning and Optimizing Probabilistic Models for Planning under Uncertainty

R. van Bekkum, M. T. J. Spaan

Algorithmics Group, Department of Software Technology

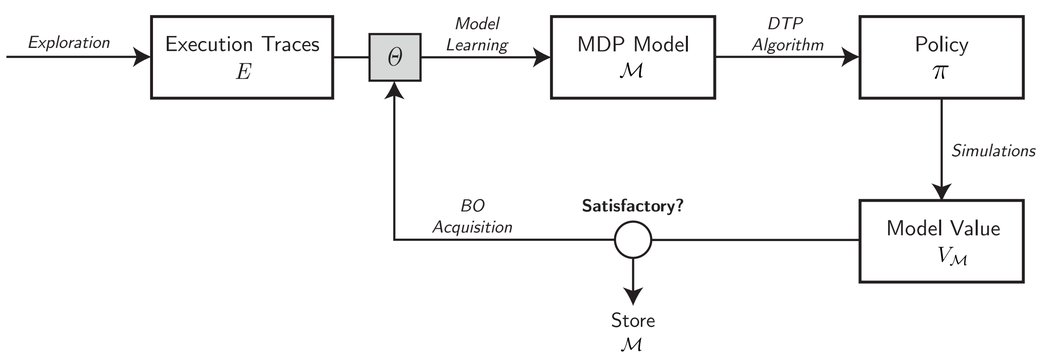
Abstract

Decision-theoretic planning techniques are increasingly being used to obtain (optimal) plans for domains involving uncertainty, which may be present in the form of the controlling agent's actions, its percepts, or exogenous factors in the domain. These techniques build on detailed probabilistic models of the underlying system, for which Markov Decision Processes (MDPs) have become the *de facto* standard formalism. However, handcrafting these probabilistic models is usually a daunting and error-prone task, requiring expert knowledge on the domain under consideration. Therefore, it is desirable to automate the process of obtaining these models by means of learning algorithms presented with a set of execution traces from the system. Although some work has already been done on crafting such learning algorithms, the state of the art lacks an automated method of configuring their hyperparameters, so to maximize the performance yielded from executing the derived plans.

In this work we present a solution that employs the *Bayesian Optimization* (BO) framework to learn MDPs autonomously from a set of execution traces, optimizing the expected value and performance in simulations over a set of tasks the underlying system is expected to perform. The approach has been tested on learning MDPs for mobile robot navigation, motivated by the significant uncertainty accompanying the robots' actions in this domain.

1 Base Framework

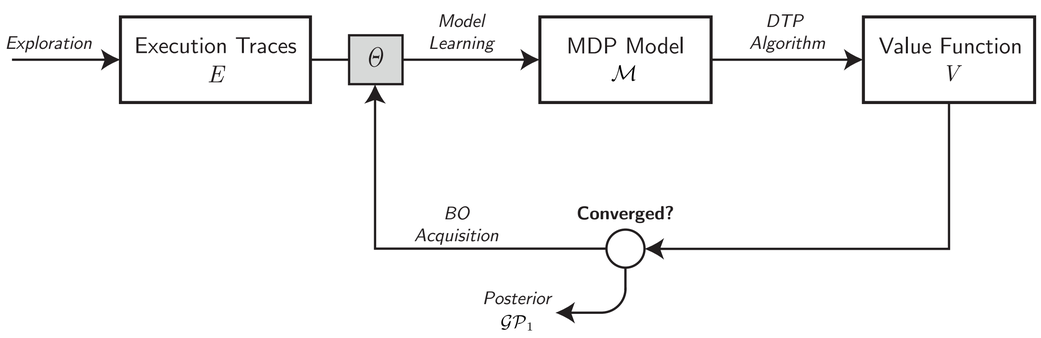
In our solution we employ BO to select hyperparameters θ for model learning algorithms towards a global maximizer of a performance measure $V_{\mathcal{M}}$ (made through time-expensive simulations) over a set of tasks the system is expected to perform.



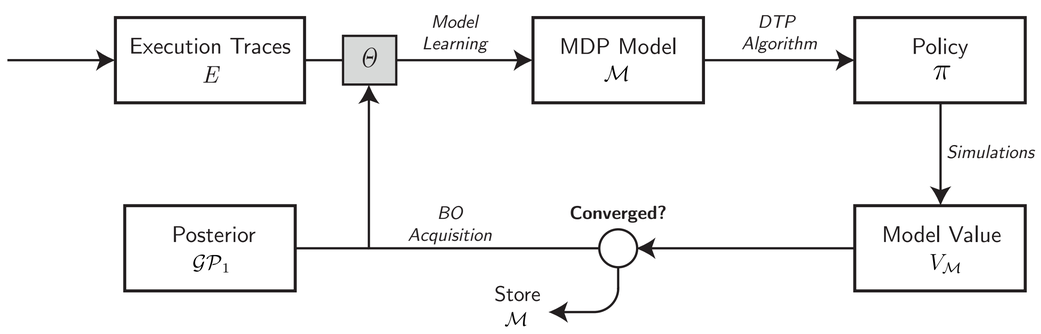
2 Multi-Phase Framework

To improve the cost-effectiveness, the solution is extended by defining two phases. In the first phase an optimization is done on a relatively cost-cheap performance measure based on the value functions computed from the learned MDPs. The resulting posterior is used to potentially speed up the optimization in the second phase by steering the BO acquisition of new samples.

Phase 1

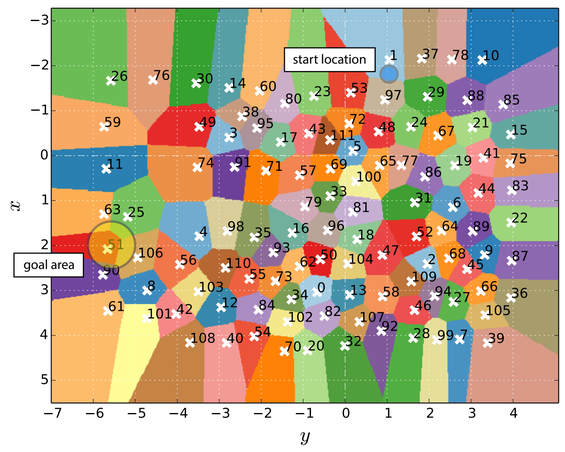


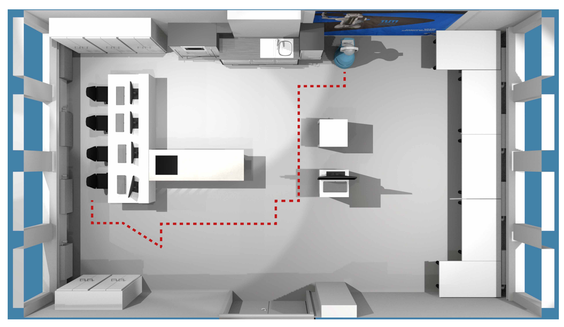
Phase 2



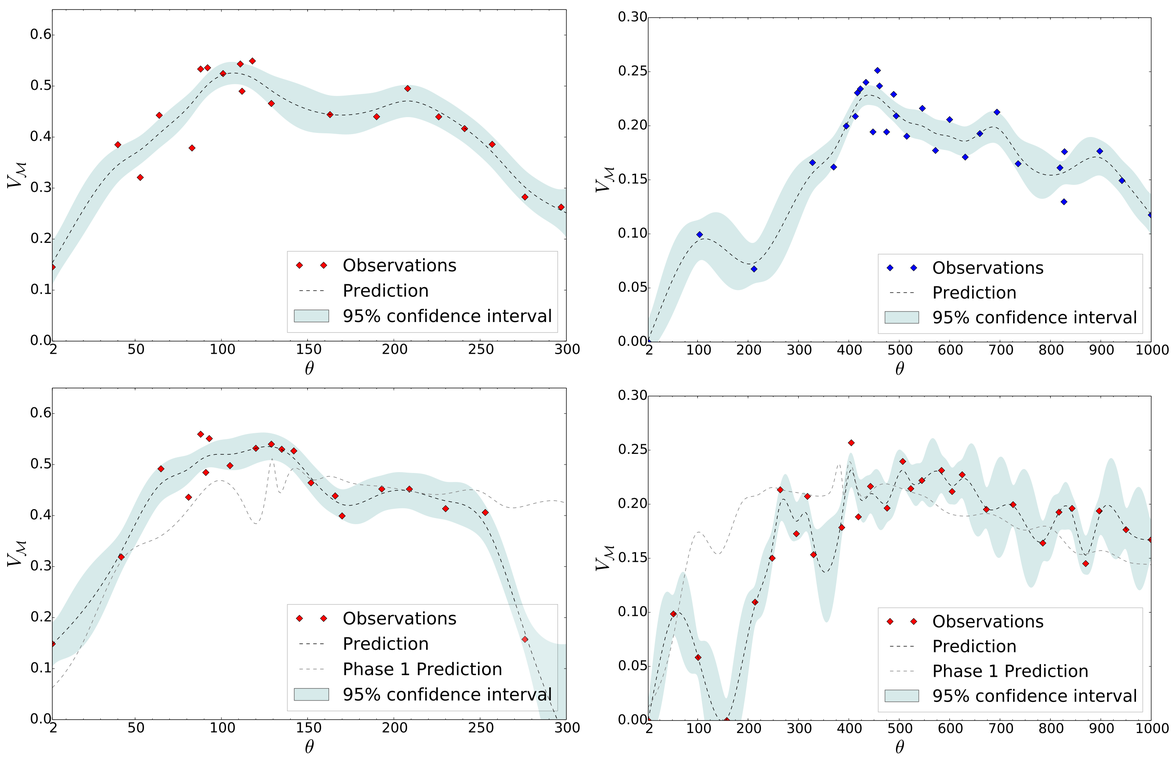
3 Experimental Setup and Results

For our experiments an implementation has been made for path planning in mobile robot navigation, where a mobile robot is controlled by learned MDPs in simulation environments inside the *Morse* robotics simulator. Based on execution traces from a random action policy, MDPs are learned based on clustering and maximum likelihood algorithms.





The plots below show GP posteriors resulting from BO in some of our experiments from which we can identify the hyperparameters θ most likely to maximize the system's performance.



4 Conclusions

- First Item
- Second Item
- Third Item