Integrating Value Function-Based and Policy Search Methods for Sequential Decision Making

Shivaram Kalyanakrishnan and **Peter Stone**
Department of Computer Sciences, The University of Texas at Austin
{shivaram, pstone}@cs.utexas.edu

Sequential decision making from experience, or reinforcement learning (RL), has traditionally been cast as a Markov Decision Problem (MDP) (Sutton and Barto 1998). An MDP comprises a set of states S, a set of actions A, a reward function $R : S \times A \times S \rightarrow \mathbb{R}$, and a transition function $T : S \times A \times S \rightarrow [0, 1]$. The objective is to find a policy $\pi : S \rightarrow A$ that maximizes expected long-term reward. Indeed, when S and A are finite, it is possible to learn the optimal action value function $Q^* : S \times A \rightarrow \mathbb{R}$ efficiently, such as through temporal difference (TD) updates to a table of values (Brafman and Tennenholtz 2003; Watkins and Dayan 1992). The optimal policy π^* can then be derived as: $\pi^*(s) = \text{argmax}_a \, Q^*(s, a), \forall s \in S$.

The theoretical guarantee of learning the optimal policy does not extend to a predominant number of sequential decision making problems occurring in practice, which possess continuous (or very large) state spaces. In such problems, the function approximation scheme employed might preclude representation of the optimal policy; even with sufficient representations, learning algorithms can converge to fixed points that are sub-optimal (Bhatnagar et al. 2008; Perkins and Precup 2003). Just as insufficient representations can break the Markovian assumption, so can noise in the state signal, or partial observability. Coping with partial observability systematically, while well-studied, is yet to scale to complex tasks with high-dimensional, continuous state spaces (Cassandra, Kaelbling, and Littman 1994; Pineau, Gordon, and Thrun 2006).

Sequential decision making in real-world applications is bound to suffer from insufficient representations of the function approximator and noisy observations; the objective of learning in these tasks has to be rescaled from achieving optimal behavior to realizing policies with "high" expected long-term rewards in a "sample-efficient" manner. A natural approach for doing so is to learn an approximation of the (action) value function, as in the discrete case, from which a greedy policy is derived. Value function-based (VF) methods have indeed been successful on several complex tasks, including elevator control (Crites and Barto 1996), autonomic resource allocation (Tesauro et al. 2007), and robot soccer Keepaway (Stone, Sutton, and Kuhlmann 2005). However, it remains that even for simple two-state MDPs and with linear function approximation, the parameters $\mathbf{w}_{VF}$ learned to approximate the value function best (such as in terms of squared error) might yield policies with lower expected long-term rewards than what is achievable with the chosen representation (Baxter and Bartlett 2001). In other words, if $V(\pi_{\mathbf{w}})$ is the expected long-term reward of a policy π with parameters $\mathbf{w}$, then in general, $\mathbf{w}_{VF} \neq \text{argmax}_{\mathbf{w}} \, V(\pi_{\mathbf{w}})$.

For a given policy representation, determining $\mathbf{w}_{best} = \text{argmax}_{\mathbf{w}} \, V(\pi_{\mathbf{w}})$ is an unconstrained optimization problem. Policy search (PS) methods (Bhatnagar et al. 2008; Whiteson and Stone 2006) directly search the space of parameters to find "good" solutions $\mathbf{w}_{PS}$ with high values of $V(\pi_{\mathbf{w}_{PS}})$. In contrast with VF methods, they do not learn the value function as an intermediate step. Intuition suggests that this would make them less sensitive to poor function approximation and partial observability, while on the other hand, be less sample efficient.

Our recent experimental study affirms this intuition (Kalyanakrishnan and Stone 2009a). On a suite of "grid world" tasks that can be varied for the size of the state space, action noise, expressiveness of function approximation, and state noise, we record the performance of Sarsa (Rummery and Niranjan 1994), a classical VF method, and the cross-entropy method (De Boer et al. 2005), a PS method. While Sarsa is able to quickly learn good policies when provided adequate function approximation and complete observability of state, the cross-entropy method, while it requires more training experiences by a few orders of magnitude, is able to yield significantly better policies under poor function approximation and high values of state noise. Sample efficiency and asymptotic performance are both desirable in practical sequential decision making problems. Extending our study, we test a simple scheme to integrate VF and PS methods to achieve these dual objectives.

In our method of integration, denoted VF+PS, we force PS to employ the same representation as VF; i.e., the action chosen from some state under PS is greedy with respect to action preferences (which would be estimates of action values under VF). The parameters learned by VF after a certain number of episodes of training are transferred to PS, which proceeds to refine them. Results show that over a broad

Submitted to the Multidisciplinary Symposium on Reinforcement Learning (MSRL 2009). Do not distribute.
The first author is a student.

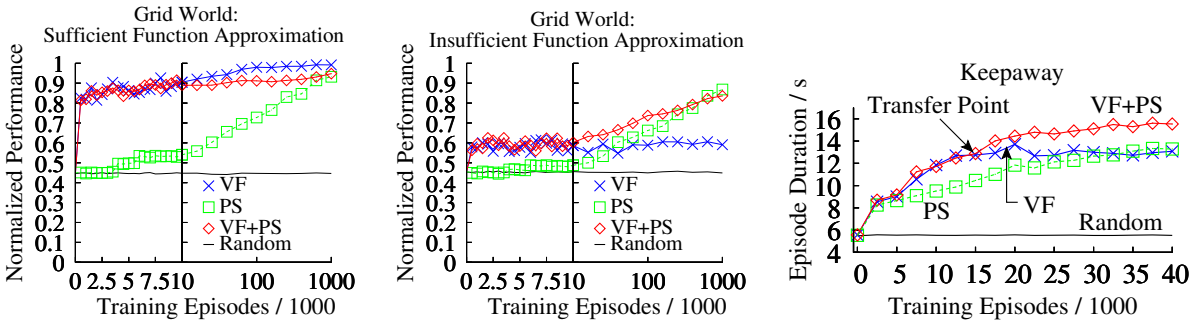

Figure 1: Combining VF and PS methods. In the grid world graphs, note the break in the x axis at 10,000 episodes, beyond which a log scale is adopted. For detailed descriptions, see original paper (Kalyanakrishnan and Stone 2009a).

range of settings in our grid world suite of tasks, and using a linear representation, VF+PS inherits the superior sample efficiency of VF as well as the high asymptotic performance of PS. Illustrative learning curves are shown in Figure 1. In Figure 1(a), the algorithms employ a representation capable of expressing the optimal policy; in Figure 1(b), the representation is severely impoverished. Under both settings, VF+PS (with parameters transferred at 10,000 episodes) is able to learn quickly and register a high asymptote, whereas VF and PS fail in one of these respects. We also observe this ability of VF+PS in the more complex benchmark task of robot soccer Keepaway (Stone, Sutton, and Kuhlmann 2005), when the representation used for learning is a single layer neural network for each action.

Finding $\mathbf{w}_{best} = \text{argmax}_{\mathbf{w}} V(\pi_{\mathbf{w}})$ under a given representation is crucial in practical applications of reinforcement learning. Whereas VF is not committed to finding $\mathbf{w}_{best}$ and PS is not sample efficient in trying to find it, VF+PS attempts to fulfil both criteria. We are working on adding to VF+PS the ability to intelligently decide when to switch from VF to PS (currently this is hand-coded). While our current implementation transfers the entire policy representation from VF to PS, a more general framework would allow PS to inherit the policy learned by VF without necessarily borrowing the representation. As preliminary work, our current implementation of VF+PS only considers Sarsa for VF and the cross-entropy method for PS: these methods do not represent the entire spectrum of VF and PS methods. We aim to consider in future work actor-critic algorithms, policy gradient methods, and VF methods using eligibility traces, all of which show some degree of resistance to deficient function approximation and partial observability.

In a related line of work, we show that VF and PS methods can be applied to learn separate components of team behavior in the Keepaway task. Whereas previous successful results in Keepaway (including the one reported earlier) limit learning to an isolated, infrequent decision that amounts to a turn-taking behavior among players (PASS), we expand the agents' learning capability to include the more ubiquitous action of moving without the ball (GETOPEN) (Kalyanakrishnan and Stone 2009b). Credit assignment to individual actions in GETOPEN is less straightforward than for PASS; we find it expeditious to learn GETOPEN using policy search, whereas PASS is well-suited to TD learning (Stone, Sutton, and Kuhlmann 2005). Results show that not only does learning GETOPEN match well-tuned hand-coded policies, but importantly, that PASS and GETOPEN can be learned simultaneously in an interleaved manner. This serves as an instance of VF and PS methods solving different subproblems in a complex task, which is also important for scaling RL to increasingly complex problems.

References

Baxter, J., and Bartlett, P. L. 2001. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research* 15:319–350.

Bhatnagar, S.; Sutton, R.; Ghavamzadeh, M.; and Lee, M. 2008. Incremental natural actor-critic algorithms. In Platt, J.; Koller, D.; Singer, Y.; and Roweis, S., eds., *Advances in Neural Information Processing Systems 20*. Cambridge, MA: MIT Press. 105–112.

Brafman, R. I., and Tennenholtz, M. 2003. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research* 3:213–231.

Cassandra, A. R.; Kaelbling, L. P.; and Littman, M. L. 1994. Acting optimally in partially observable stochastic domains. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, volume 2, 1023–1028. Seattle, Washington, USA: AAAI Press/MIT Press.

Crites, R. H., and Barto, A. G. 1996. Improving elevator performance using reinforcement learning. In Touretzky, D. S.; Mozer, M.; and Hasselmo, M. E., eds., *Advances in Neural Information Processing Systems 8, NIPS, Denver, CO, November 27-30, 1995*, 1017–1023. MIT Press.

De Boer, P. T.; Kroese, D. P.; Mannor, S.; and Rubinstein, R. 2005. A tutorial on the cross-entropy method. *Annals of Operations Research* 134(1):19–67.

Kalyanakrishnan, S., and Stone, P. 2009a. An empirical analysis of value function-based and policy search reinforcement learning. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems*. To appear.

Kalyanakrishnan, S., and Stone, P. 2009b. Learning complementary multiagent behaviors: A case study. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems*. To appear as short paper.

Perkins, T. J., and Precup, D. 2003. A convergent form of approximate policy iteration. In S. Becker, S. T., and Obermayer, K., eds., *Advances in Neural Information Processing Systems 15*. Cambridge, MA: MIT Press. 1595–1602.

Pineau, J.; Gordon, G.; and Thrun, S. 2006. Anytime point-based approximations for large pomdps. *Journal of Artificial Intelligence Research* 27:335–380.

Rummery, G. A., and Niranjan, M. 1994. On-line Q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR 166, Cambridge University Engineering Department.

Stone, P.; Sutton, R. S.; and Kuhlmann, G. 2005. Reinforcement learning for RoboCup-soccer keepaway. *Adaptive Behavior* 13(3):165–188.

Sutton, R. S., and Barto, A. G. 1998. *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.

Tesauro, G.; Jong, N. K.; Das, R.; and Bennani, M. N. 2007. On the use of hybrid reinforcement learning for autonomic resource allocation. *Cluster Computing* 10(3):287–299.

Watkins, C. J. C. H., and Dayan, P. 1992. Q-learning. *Machine Learning* 8(3-4):279–292.

Whiteson, S., and Stone, P. 2006. On-line evolutionary computation for reinforcement learning in stochastic domains. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 1577–84.