Machine Learning, 49, 107–109, 2002

Introduction

SATINDER SINGH baveja@cs.colorado.edu
Syntek Capital, New York, NY 01009, USA

This is the third special issue of *Machine Learning* on the subject of reinforcement learning (the first and second special issues were edited by Richard Sutton in 1992 and Leslie Kaelbling in 1996 respectively). The field of reinforcement learning continues to grow, attracting ideas and participants not only from AI and machine learning but also from neuroscience, cognitive science, operations research, and control. More than a decade of the resulting research has led to great progress in the theoretical underpinnings of the field, much of it derived from the theory of dynamic programming and the associated frameworks of Markov Decision Processes (MDPs), semi-MDPs and partially observable MDPs (POMDPs) (see texts by Sutton and Barto (1998) and Bertsekas and Tsitsiklis (1996) for excellent overviews).

Much has been accomplished, and yet, of course, much remains to be done. I will take advantage of this guest editorial to outline my general views on three of the open issues that are key to further rapid progress in reinforcement learning, and then turn to very briefly survey the papers in this special issue.

Some key open issues

The most common formal representations of reinforcement learning problems are those of MDPs, semi-MDPs (to deal flexibly with continuous time), and POMDPs (to deal with hidden state). While these state-based representations have been tremendously useful in gaining theoretical confidence in and understanding of our basic algorithms through convergence results, they are also severely limited by their use of unstructured or at best propositional representations. Indeed, these POMDP representations have proven intractable not only in theory but also in practise. Recently, factored and other structured representations of state and observation common in the literature on graphical models for inference have begun to be incorporated in POMDP-based reinforcement learning problems. While this work is ongoing and encouraging, intractability and inapplicability continues to be a problem except in special cases. In my view, therefore, developing alternatives to POMDP/state based representations is crucial to the further development of the field. In this regard, there are at least three representational ideas already known in the literature that warrant greater and further attention: history-based representations (e.g., McCallum, 1995), deictic representations (e.g., Whitehead & Ballard, 1991), and test-based representations (e.g., Rivest & Schapire, 1994; Jaeger, 2000). Developing these (and other yet to be thought of) ideas further may help break us free from the tether of state-based representations and perhaps even lead the way towards incorporating the higher-order logic based representations common in AI.

Another recent line of work that has some bearing on representations of reinforcement learning problems is the work on temporal and structural abstraction (also called "hierarchical reinforcement learning"). There has been progress in developing some basic formalisms, theory, and algorithms, e.g., the work on options by Sutton, Precup, and Singh (1999), on HAMs by Parr and Russell (1998), and on MAXQ by Dietterich (1998). Nevertheless some significant open issues remain: how does an agent learn these abstract representations from experience, how does it learn environment-models in these representations, and furthermore how does it learn models for many different ways of behaving from a relatively small amount of experience (called off-policy learning). Providing practical and scalable solutions to these issues is crucial for reinforcement learning to deliver on its promise of building agents that can learn through interaction in complex, dynamic and uncertain environments.

Finally, another key thrust for the field should be to build up a set of publicly-available software tools, best-practise guidelines, and other resources that allow experts in application domains to use reinforcement learning to solve their problems. This is key to expanding the scope and influence of our field, because in the absence of such resources, the signature application of the field remains the 1995 work of Tesauro on TD-gammon (though of course there continue to be some successes in the areas of control problems, operations research, robotics, AI, and game playing).

In this issue

It is my pleasure to have assembled (with Leslie Kaelbling's help) a set of eleven papers that are representative of the exciting current work in the field of reinforcement learning. Two of the eleven are on applications of reinforcement learning methods: Tong and Brown formulate admission control and routing in multimedia networks as a semi-Markov decision process and use Q-learning with state aggregation to learn a better policy than available heuristic policies, while Mcgovern, Moss and Barto apply and compare rollout (Monte-Carlo) and other temporal difference methods to the problem of optimizing block instruction scheduling on a pipelined architecture, showing improved performance over a commercial scheduler.

The remaining papers in this special issue are mainly a combination of new algorithms and theoretical results. In one such paper, Ormoneit and Sen consider a kernel-based approach to approximating value functions from sampled data. They prove that under smoothness assumptions on the payoff and reward functions such an approach has the following important and hitherto unavailable property: the quality of the estimated value function improves with increasing amount of data and eventually leads to optimal performance. In another paper, Tsitsiklis and VanRoy tackle the debate over the appropriateness of the common practise of using the discounted-case formulation for problems where the average-case formulation is more natural. The usual justification is that an average-case formulation can be approximated well by the discounted-case formulation with a discount factor close to one, and discounted-case algorithms tend to be more stable than their average-case counterparts. On the other hand, using a discount-factor very close to one can introduces instabilities into the learning process. In this issue Tsitsiklis and VanRoy prove that at least for the case of policy evaluation with linear function approximation there is little transient or asymptotic difference between the two cases provided care is taken to choose the right scaling parameters.

Kearns, Mansour and Ng provide new algorithms based on sparse sampling for solving large-scale MDPs. They show that given a generative model for an arbitrary MDP, their algorithm performs on-line, near-optimal planning with a per-state running time that has *no dependence* on the number of states, but an *exponential dependence* on the planning horizon. This work adds a dramatically different point in the space of reinforcement learning algorithms and the different tradeoffs they make in their dependence on size of state-action space and planning horizon. In an another paper, Kearns and Singh provide a new algorithm that addresses the exploration-exploitation tradeoff faced by any algorithm for learning to solve unknown MDPs. Their algorithm achieves near-optimal return in time polynomial in the size of the MDP as well as the mixing time of the optimal policy (in the average-case formulation) or the horizon-time of the MDP (in the discounted-case formulation).

Boyan provides a new derivation of Least-Squares Temporal Difference (LSTD) that generalizes it to all values of the eligibility trace parameter (λ). This derivation and analysis also provides an interesting and intuitive interpretation of LSTD (a direct learning method that does not build an explicit model) as a model-based reinforcement learning method. Millan, Posenato, and Dedieu propose extensions of Q-learning to deal with continuous state-action spaces using ideas from self-organizing maps and test out their algorithms on robot navigation tasks. Mihatsch and Nueneier propose, analyse, and test a new family of modified Q-learning algorithms with a continuous parameter that determines how risk-seeking or risk-averse the resulting policy is going to be. Munos and Moore provide and test new criteria for splitting cells in powerful new variable resolution approaches to state abstraction in continuous-state deterministic control problems. Foster and Dayan introduce a new class of approaches to structural abstraction in reinforcement learning. They propose using unsupervised, mixture model, learning methods to extract component structure from optimal value functions for several related tasks and show that the learned structure can be exploited for more efficient reinforcement learning.

References

Bertsekas, D. P., & Tsitsiklis, J. N. (1996). *Neuro-dynamic programming*. Belmont, MA: Athena Scientific.

Dietterich, T. G. (1998). The MAXQ method for hierarchical reinforcement learning. In *Machine Learning: Proceedings of the Fifteenth International Conference* (pp. 118–126). San Mateo, CA: Morgan Kaufman.

Jaeger, H. (2000). Observable operator models for discrete stochastic time series. *Neural Computation, 12*, 1371–1398.

McCallum, A. K. (1995). Reinforcement learning with selective perception and hidden state. Doctoral dissertation, Department of Computer Science, University of Rochester.

Parr, R., & Russell, S. (1998). Reinforcement learning with hierarchies of machines. In *Advances in neural information processing systems 11*. Cambridge. MA: MIT Press.

Rivest, R. L., & Schapire, R. E. (1994). Diversity-based inference of finite automata. *Journal of the ACM, 41*, 555–589.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Sutton, R. S., Precup, D., & Singh, S. (1999). Between MDPs and Semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence, 112*, 181–211

Tesauro, G. J. (1995). Temporal difference learning and TD-Gammon. *Communications of the ACM, 38*, 58–68.

Whitehead, S. D., & Ballard, D. H. (1991). Learning to perceive and act by trial and error. *Machine Learning, 7:1*, 45–83.