Reinforcement Learning: Past, Present and Future*

Richard S. Sutton

AT&T Labs, Florham Park, NJ 07932, USA,
sutton@research.att.com, www.cs.umass.edu/~rich

Reinforcement learning (RL) concerns the problem of a learning agent interacting with its environment to achieve a goal. Instead of being given examples of desired behavior, the learning agent must discover by trial and error how to behave in order to get the most reward. RL has become popular as an approach to artificial intelligence because of its simple algorithms and mathematical foundations (Watkins, 1989; Sutton, 1988; Bertsekas and Tsitsiklis, 1996) and because of a string of strikingly successful applications (e.g., Tesauro, 1995; Crites and Barto, 1996; Zhang and Dietterich, 1996; Nie and Haykin, 1996; Singh and Bertsekas, 1997; Baxter, Tridgell, and Weaver, 1998). An overall introduction to the field is provided by a recent textbook (Sutton and Barto, 1998). Here we summarize three stages in the development of the field, which we coarsely characterize as the past, present, and future of reinforcement learning.

RL *past*, up until about 1985, developed the general idea of trial-and-error learning—of actively exploring to discover what to do in order to get reward. It was many years before trial-and-error learning was recognized as a significant subject for study different from supervised learning and pattern recognition. RL past emphasized the need for an active, exploring agent, as in the studies of learning automata and of the n-armed bandit problem. Another key insight of RL past was just the idea of a scalar reward signal as a simple but general specification of the goal of an intelligent agent, an idea which I like to highlight by referring to it as the *reward hypothesis*. The learning methods of RL past usually learned only a *policy*, a mapping from perceived states of the world to the action to take. This limited them to relatively benign problems in which reward was immediate and indicated (e.g., by its sign) whether the behavior was good or bad. Problems with delayed reward, or in which the best action much be picked out of several good actions (or the least bad out of several bad actions), could not be reliably solved until the ideas of value functions and temporal-difference learning were introduced in the 1980s.

The transition to RL present ($\approx$ 1985) came about by focusing on *value functions* and on a general mathematical characterization of the RL problem known as *Markov decision processes* (MDPs). The state-value function, for example, is the function mapping perceived states of the world to the expected total future reward starting from that state. Almost all sound methods for solving MDPs (that is, for finding optimal behavior) are based on learning or computing approximations to value functions, and the most efficient methods for doing this all

[*] The slides used in the talk corresponding to this extended abstract can be found at http://envy.cs.umass.edu/~rich/SEAL98/sld001.htm.

X. Yao et al. (Eds.): SEAL'98, LNCS 1585, pp. 195-197, 1999.

seem to be based on temporal differences in estimated value (as in dynamic programming, heuristic search, and temporal-difference learning). Although finding a policy to maximize reward is still the ultimate goal of RL, RL present is much more focused on the intermediating goal of approximating value, from which the optimal policy can be determined. RL present is also as much about *planning* using a model of the world as it is about learning from interaction with the world. Whether learning or planning optimal behavior, approximation of value functions seems to be at the heart of all efficient methods for finding optimal behavior. The *value function hypothesis* is that approximation of value functions is the dominant purpose of intelligence.

RL future has yet to happen, of course, but it may be useful to try to guess what it will be like. Just as RL present took a step away from the ultimate goal of reward to focus on value functions, so RL future may take a further step away to focus on the structures that enable value function estimation. Principle among these are representations of the world's state and dynamics. It is commonplace to note that the efficiency of all kinds of learning is strongly affected by the suitability of the representations used. If the right features are represented prominently, then learning is easy; otherwise it is hard. It is time to consider seriously how features and other structures can be constructed automatically by machines rather than by people. In RL, representational choices must also be made about states (e.g., McCallum, 1995), actions (e.g., Sutton, Precup, and Singh, 1998) and models of the world's dynamics (Precup and Sutton, 1998), all of which can strongly affect performance. In psychology, the idea of a developing mind actively creating its representations of the world is called *constructivism*. My prediction is that for the next tens of years RL will be focused on constructivism.

References

Baxter, J., Tridgell, A., Weaver, L. (1998). KnightCap: A chess progream that learns by combining TD(λ) with game-tree search. *Proceedings of the Fifteenth International Conference on Machine Learning*, pp. 28–36.

Bertsekas, D. P., and Tsitsiklis, J. N. (1996). *Neuro-Dynamic Programming.* Athena Scientific, Belmont, MA.

Crites, R. H., and Barto, A. G. (1996). Improving elevator performance using reinforcement learning. In *Advances in Neural Information Processing Systems 9*, pp. 1017–1023. MIT Press, Cambridge, MA.

McCallum, A. K. (1995) Reinforcement Learning with Selective Perception and Hidden State. University of Rochester PhD. thesis.

Nie, J., and Haykin, S. (1996). A dynamic channel assignment policy through Q-learning. CRL Report 334. Communications Research Laboratory, McMaster University, Hamilton, Ontario.

Precup, D., Sutton, R.S. (1998). Multi-time models for temporally abstract planning. *Advances in Neural Information Processing Systems 11*. MIT Press, Cambridge, MA.

Singh, S. P., and Bertsekas, D. (1997). Reinforcement learning for dynamic channel allocation in cellular telephone systems. In *Advances in Neural Information Processing Systems 10*, pp. 974–980. MIT Press, Cambridge, MA.

Sutton, R. S. (1988). Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44.

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

Sutton, R. S., Precup, D., Singh, S. (1998). Between MDPs and semi-MDPs: Learning, planning, and representing knowledge at multiple temporal scales. Technical Report 98-74, Department of Computer Science, University of Massachusetts.

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

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards.* Ph.D. thesis, Cambridge University.

Zhang, W., and Dietterich, T. G. (1996). High-performance job-shop scheduling with a time–delay TD(λ) network. In *Advances in Neural Information Processing Systems 9*, pp. 1024–1030. MIT Press, Cambridge, MA.