Reinforcement learning control

Andrew G Barto

University of Massachusetts, Amherst, USA

Reinforcement learning refers to improving performance through trial-and-error. Despite recent progress in developing artificial learning systems, including new learning methods for artificial neural networks, most of these systems learn under the tutelage of a knowledgeable 'teacher' able to tell them how to respond to a set of training stimuli. Learning under these conditions is not adequate, however, when it is costly, or even impossible, to obtain this kind of training information. Reinforcement learning is attracting increasing attention in computer science and engineering because it can be used by autonomous systems to learn from their experiences instead of from knowledgeable teachers, and it is attracting attention in computational neuroscience because it is consonant with biological principles. Recent research has improved the efficiency of reinforcement learning and has provided some striking examples of its capabilities.

Current Opinion in Neurobiology 1994, **4**:888–893

Introduction

While the core ideas of modern reinforcement learning come from theories of animal classical and instrumental conditioning (although the specific term 'reinforcement learning' is not used by psychologists), the influence of concepts from artificial intelligence and control theory has produced a collection of computationally powerful learning methods. Most adaptive artificial neural networks employ the learning paradigm called supervised learning, which emphasizes the role of training information in the form of desired, or 'target', network responses for a set of training inputs. In contrast, reinforcement learning emphasizes learning feedback that evaluates the learner's performance without providing standards of correctness in the form of behavioral targets. The simplest reinforcement learning methods follow the classical Law of Effect [1] that states that if an action is followed by a satisfactory state of affairs, or an improvement in the state of affairs, then the tendency to produce that action is strengthened, that is, reinforced, and if an action is followed an unsatisfactory state of affairs, then the tendency to produce that action is weakened. Because it can be significantly easier to obtain evaluative feedback than standards of correctness, reinforcement learning is attracting increasing attention in computer science and engineering as an approach to making robots more autonomous. In addition, because many researchers believe its principles are closely compatible with neural mechanisms, reinforcement learning is attracting increasing attention in computational neuroscience.

Klopf [2] put forward the hypothesis — which first appeared in a 1972 technical report — that individual neurons are capable of reinforcement learning, a hypothesis that has had considerable influence on modern approaches to reinforcement learning. In this article, I review these developments and discuss some of their implications for modeling neural control systems.

Supervised and reinforcement learning

It is not obvious that there are important differences between supervised and reinforcement learning. The widely known error backpropagation method for training artificial neural networks (e.g. see [3]) is an example of a supervised learning method. Training information consists of a set of input patterns, each paired with a target output pattern. The network's weights are adjusted based on a list of errors derived by comparing each output unit's actual activity with its target activity. This error list specifically tells each output unit how it should change its activity; the error backpropagation process makes similar information available to all the units in the network. In contrast to a list of errors, the training information used in reinforcement learning is evaluative feedback: it tells the learner whether or not, and possibly by how much, its behavior has improved or has gotten worse; or it provides a measure of the 'goodness' of the behavior; or it just provides an indication of success or failure. Evaluative feedback does not directly tell the learner what it should have done or how it should change its behavior. Instead of trying to match a standard of correctness (i.e. obtain errors of zero), a reinforcement learning system tries to maximize the goodness of be-

Abbreviation

TD—temporal difference.

 © Current Biology Ltd ISSN 0959-4388

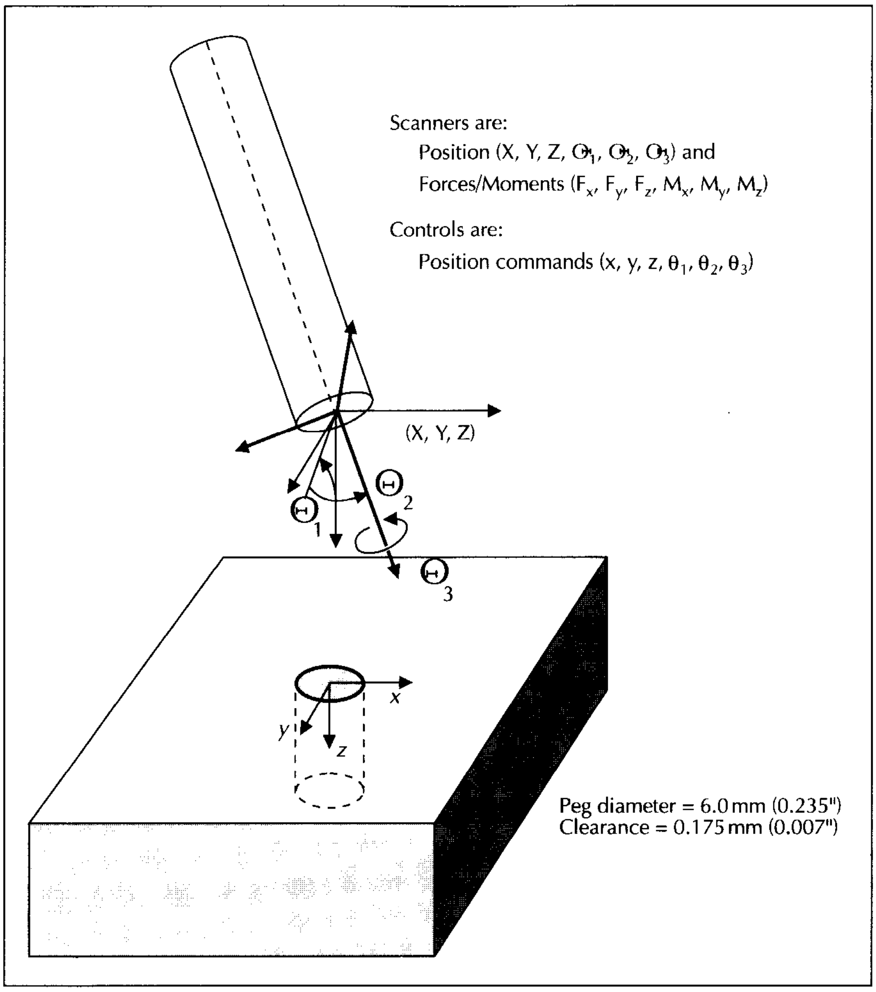

Fig. 1. A robot peg-insertion task. A peg is placed in the gripper of a six degrees-of-freedom robot. The robot's objective is to control the peg's position and orientation so as to insert it into the hole to a suitable depth (25 mm). The network controller receives continuous measurements of the peg's position (X, Y, and Z), orientation (Θ_1, Θ_2, and Θ_3), as well as measurements of forces (F_x, F_y, and F_z), along the three axes and moments (M_x, M_y, and M_z), about these axes. As a function of these measured quantities, the controller generates position commands (x, y, z, θ_1, θ_2, θ_3) to direct movement. For small clearances, this task is difficult due to significant inaccuracies in both the measured quantities and the robot's ability to move as directed. Reproduced with permission from [10].

havior as indicated by evaluative feedback. To do this, it has to actively try alternatives, compare the resulting evaluations, and use some kind of selection mechanism to guide behavior toward the better alternatives. Whereas supervised learning is instructional, reinforcement learning is selectional.

The power of a reinforcement learning system derives from the fact that the evaluation component, often called a 'critic', requires much less information and knowledge than a 'teacher' of a supervised learning system. It is possible to evaluate a system's behavior without knowing what the correct behavior would be. In fact, a critic does not even need access to the learning system's actions, as it can simply evaluate their consequences on some complex process. Moreover, it does not need to know anything about the mechanism by which the actions produce these consequences. On the negative side, because it relies on low-resolution training information, a reinforcement learning system can require a great amount of experience to show significant improvement. Another complication arises from the fact that actions can have delayed as well as immediate consequences, and evaluative feedback generally evaluates the consequences of all of the system's past behavior. How can a reinforcement learning system deal with complex webs of actions and their consequences occurring throughout time? This has been called the temporal credit assignment problem. General discussions of modern reinforcement learning are provided by references [4–7].

A robot control example

Some of the features of reinforcement learning are clearly illustrated by a recent robot learning system that uses an artificial neural network trained via reinforcement learning to control a robot arm in inserting a peg into a hole (Fig. 1) [8–10]. The task's difficulty arises from the considerable uncertainty in both the sensations and in the execution of motion commands resulting from errors and noise. Even with the peg carefully lined up with the hole at the start, the tight clearance between the peg and the hole (about 0.175 mm for a 6.0 mm diameter peg) and the lack of precision in the robot (an inexpensive and imprecise robot arm) make it very difficult to follow any pre-planned trajectory without missing the hole or jamming the peg. Thus, the problem is not to learn a transformation from workspace coordinates to joint angles, or to control the

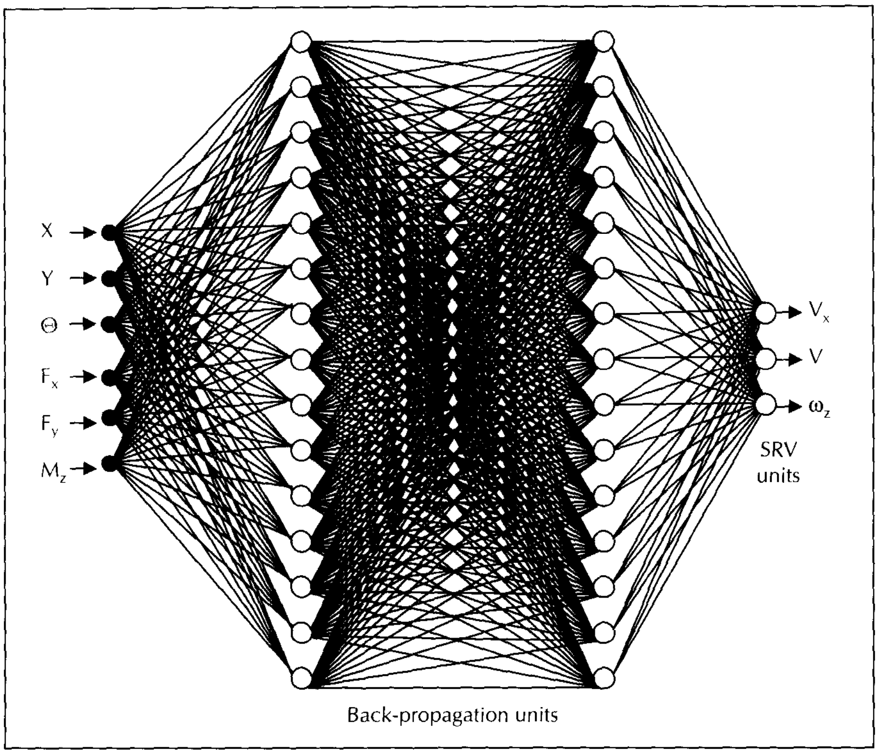

Fig. 2. An artificial neural network used for the peg-insertion task. This network is configured for a two-dimensional version of the peg-insertion task. Its inputs are the sensed position, orientation, force, and moment information, and its outputs command robot movement. The network's output units are stochastic real-valued (SRV) units [11], which adjust their weights using a reinforcement learning rule. The remaining weights of the network are adjusted through the use of the error-backpropagation method [3]. Reproduced with permission from [46].

robot to move along a given free-space trajectory (the preoccupation of many applications of artificial neural networks to robot control). These abilities are already built into the robot. The problem is to learn how to react to real sensations of position and force so as to be able to actually insert the peg into the hole.

Fig. 2 shows the multi-layer network trained for peg insertion (actually, this network was used for a two-dimensional version of the problem, but the network for the three-dimensional case was similar). Inputs give information about the current position of the peg computed from the sensed joint positions and the force and moment sensations produced by a wrist force sensor. Outputs are velocity commands in workspace coordinates. To train this network using supervised learning one would have to supply it with target outputs for a rich set of cases, but the point here is that the target outputs are not known; if they were, learning would not be necessary. Instead, the network learns from a critic that rewards it whenever progress is made toward the inserted position without generating wrist forces exceeding a threshold. The critic does not need to know the appropriate motion commands.

Each of the network's output units adjusts its weights using a reinforcement learning rule [11]. A random component in unit activity makes the network 'explore' its activity space. When a reward occurs just after the network emits a particular output pattern in the presence of some input pattern, each output unit's weights are adjusted to move its activity in the direction in which it was perturbed by the random component. Once these weights are adjusted, the usual error back-propagation process is used to adjust the weights of the hidden units. This has the effect of increasing the probability that future network responses to that input pattern (and similar patterns) will be closer to the output just emitted. Another part of the learning rule decreases the amount of exploratory activity as learning proceeds so that the network learns to react to each sensed situation with an appropriate motion command signal. After about 150 peg-insertion trials, the robot was consistently able to perform successful insertions. With additional trials, insertion time decreased and force threshold violations were eliminated, showing that the system had learned a mapping from sensations to the movement commands producing skilled peg insertion. Similar results were obtained for a square peg and square hole [10]. Other examples of reinforcement learning applied to robot control are described in references [12–14] (simulated robots) and [15•,16–18] (real robots).

Delayed rewards

Much recent progress in reinforcement learning concerns the problem of delayed rewards. Because the critic in the peg-insertion task described above was able to assess the progress of the system's behavior throughout learning, the network only had to learn how to act so as to produce immediate rewards. In other problems, the critic only rewards the learner when a final goal is reached. Both types of problems are formulated as problems in which a reward can (but may not) occur at any time, and where the learner tries to maximize a measure of the cumulative amount of reward received over time. In these problems, it can make sense to forgo short-term reward in order to achieve more reward over the long term. The theory of learning in these problems is based on the theory of optimal control [19–21], especially the computational method known as dynamic programming [7,22•].

The approach receiving the most attention focuses on learning processes by which the critic can learn how to improve its ability to evaluate behavior. These are often called adaptive critic methods [23], examples of which are temporal difference, or TD, methods [24] and Q-learning [25,26]. The basic idea is that valid predictions of reward should themselves be rewarding. Thus, an action that improves the likelihood of obtaining reward in the future, as predicted by the critic, is reinforced. With these methods, learning does not have to wait until a final goal is achieved. This mimics the phenomenon of secondary, or acquired, reinforcement observed in animal learning [27].

A remarkable demonstration is provided by a computer program using an adaptive critic to learn to play expert-level backgammon [28•,29]. This program, called TD-Gammon, started with little backgammon knowledge and yet learned to play near the level of the world's top grandmasters. The only rewards were generated at the ends of games won by the program. Using a multilayer artificial neural network, the adaptive critic learned over many games to estimate, for each board position, the probability that the program would win from that position. These estimates were used to guide the program's choice of moves. The enormous number of possible positions (more than 10^{20}) and the large number of possible choices for each move (about 400, taking into account the dice rolls), make it computationally infeasible to compute an optimal playing strategy and prevent the deep search methods successful in computer chess-playing from working well. Because backgammon can be viewed as a kind of control problem in which a player is trying to control a complex non-linear stochastic system, the success of TD-Gammon has given added impetus to research seeking similar success with other large, complex control tasks [30].

Computational neuroscience

Although most reinforcement learning research is motivated by a desire to synthesize artificial learning systems, increasing effort is being made to relate these systems to neural mechanisms. Reinforcement learning is consistent with the existence of diffuse modulatory systems in the brain that could broadcast reward signals to diverse structures. Researchers have proposed reinforcement learning models for the influence of such modulatory signals during development [31] and for the role of the basal ganglia in motor skill learning [32,33•,34]. Ljunberg *et al.* [35] present data that suggests that dopamine-producing cells in the basal ganglia respond in a manner consistent with the behavior of a TD-adaptive critic method [33•]. Furthermore, the confluence of cortical and dopamine projections on the dendrites of striatal spiny neurons suggests a three factor rule for synaptic plasticity [34]. Such a rule is necessary to implement reinforcement learning at the cellular level. Whereas a Hebbian rule adjusts synaptic strength based on the correlation of pre- and post-synaptic activity, a third factor — the reward signal — is necessary for reinforcement learning: that is, synaptic efficacy changes only when the correlation is closely followed by reward [2].

Other neural models postulate a role for reinforcement learning in the primate premotor cortex for learning how to trigger movements on the basis of visual stimuli [36] and in providing a more realistic alternative to error backpropagation in a network model of cortical area 7a [37]. Several purely behavioral models have been proposed that invoke TD mechanisms to explain a wide range of classical conditioning data [38–40]. Discussions of reinforcement learning models from a biological perspective can be found in references [35,38,41,42].

Conclusion

The increasing interest in reinforcement learning is due to its applicability to learning by autonomous robots. Although both supervised and unsupervised learning can play essential roles in learning, these paradigms by themselves are not general enough for learning while acting in dynamic and uncertain environments. In uncharted territory, where one would expect learning to be most beneficial, a system has to learn from its experiences rather than from a knowledgeable teacher. Among the topics being addressed by current reinforcement learning research are understanding how exploratory behavior is best introduced and controlled [43], learning when the environment state cannot be observed [44], and the design of modular and hierarchical learning systems [14,45].

References and recommended reading

Papers of particular interest, published within the annual period of review, have been highlighted as:
- • of special interest
- •• of outstanding interest

1. Thorndike EL: *Animal Intelligence*. Darien, Connecticut: Hafner; 1911.

2. Klopf AH: *The hedonistic neuron: a theory of memory, learning, and intelligence*. Washington, DC: Hemishere; 1982.

3. Rumelhart DE, Hinton GE, Williams RJ: **Learning internal representations by error propagation.** In *Parallel distributed processing: explorations in the microstructure of cognition*, vol 1: *Foun-*

dations. Edited by Rumelhart DE, McClelland JL. Cambridge, Massachusetts: Bradford Books/MIT Press; 1986:318–362.

4. Barto AG: **Some learning tasks from a control perspective.** In *1990 Lectures in complex systems.* Edited by Nadel L, Stein DL. Redwood City, California: Addison-Wesley; 1991:195–223.

5. Barto AG: **Reinforcement learning and adaptive critic methods.** In *Handbook of intelligent control: neural, fuzzy, and adaptive approaches.* Edited by White DA, Sofge DA. New York: Van Nostrand Reinhold; 1992:469–491.

6. Sutton RS (Ed): **A special issue of machine learning on reinforcement learning.** *Machine Learning* 1992, **8**. (Also published as: Sutton RS (Ed): *Reinforcement learning.* Boston: Kluwer Academic Press; 1992.)

7. Werbos PJ: **Approximate dynamic programming for real-time control and neural modeling.** In *Handbook of intelligent control: neural, fuzzy, and adaptive approaches.* Edited by White DA, Sofge DA. New York: Van Nostrand Reinhold; 1992:493–525.

8. Gullapalli V, Grupen RA, Barto AG: **Learning reactive admittance control.** In *Proceedings of the 1992 IEEE conference on robotics and automation.* Los Alamitos, California: Computer Society Press; 1992:1475–1480.

9. Gullapalli V: **Learning control under extreme uncertainty.** In *Advances in neural information processing systems 5.* Edited by Hanson SJ, Cohen JD, Giles CL. San Mateo, California: Morgan Kaufmann; 1993:327–334.

10. Gullapalli V, Barto AG, Grupen RA: **Learning admittance mappings for force-guided assembly.** In *Proceedings of the 1994 international conference on robotics and automation.* Los Alamitos, California: Computer Society Press; 1994:2633–2638.

11. Gullapalli V: **A stochastic reinforcement algorithm for learning real-valued functions.** *Neural Networks* 1990, **3**:671–692.

12. Lin L-J: **Self-improving reactive agents based on reinforcement learning, planning and teaching.** *Machine Learning* 1992, **8**:293–321.

13. Prescott T, Mayhew J: **Adaptive local navigation.** In *Active vision.* Edited by Blake A, Yuille A. Cambridge, Massachusetts: MIT Press; 1992:203–215.

14. Tham CK, Prager RW: **A modular Q-learning archiecture for manipulator task decomposition.** In *Machine learning: proceedings of the eleventh international workshop.* Edited by Cohen WW, Hirsch H. San Francisco: Morgan Kaufmann; 1994:323–327.

15. • Kaelbling LP: *Learning in embedded systems.* Cambridge, Massachusetts: MIT Press; 1993.
Kaelbling addresses reinforcement learning applied to problems encountered by robots embedded in complex changing environments. Some of the first results using reinforcement learning with a real mobile robot are described.

16. Maes P, Brooks RA: **Learning to coordinate behaviors.** In *Proceedings of the eighth national conference on artificial intelligence.* Menlo Park: AAAI Press/MIT Press; 1990:796–802.

17. Mahadevan S, Connell J: **Automatic programming of behavior-based robots using reinforcement learning.** *Artif Intell* 1992, **55**:311–365.

18. Miller WT, Scalera SM, Kim A: **Neural network control of dynamic balance for a biped walking robot.** In *Proceedings of the eighth Yale workshop on adaptive and learning systems.* New Haven: Center for Systems Science, Yale University; 1994:156–161.

19. Barto AG, Sutton RS, Watkins CJCH: **Learning and sequential decision making.** In *Learning and computational neuroscience: foundations of adaptive networks.* Edited by Gabriel M, Moore J. Cambridge, Massachusetts: MIT Press; 1990:539–602.

20. Barto AG, Sutton RS, Watkins C: **Sequential decision problems and neural networks.** In *Advances in neural information processing systems 2.* Edited by Touretzky DS. San Mateo, CA: Morgan Kaufmann; 1990:686–693.

21. Sutton RS, Barto AG, Williams RJ: **Reinforcement learning is direct adaptive optimal control.** In *Proceedings of the 1991 American control conference.* Evanston, Illinois; American Automatic Control Council; 1991:2143–2146.

22. • Barto AG, Bradtke SJ, Singh SP: **Learning to act using real-time dynamic programming.** *Artif Intell* 1994, in press.
This article establishes the connection between certain reinforcement learning methods and the theory of asynchronous dynamic programming. It contains extensive review material, focusing on the relationship between reinforcement learning and methods of artificial intelligence.

23. Barto AG, Sutton RS, Anderson CW: **Neuron-like elements that can solve difficult learning control problems.** *IEEE Trans Systems Man Cybernetics* 1983, **13**:835–846. (Reprinted in *Neurocomputing: Foundations of Research.* Edited by Anderson JA, Rosenfeld E. Cambridge, Massachusetts: MIT Press; 1988:535–549.)

24. Sutton RS: **Learning to predict by the method of temporal differences.** *Machine Learning* 1988, **3**:9–44.

25. Watkins CJCH: **Learning from delayed rewards [PhD thesis].** Cambridge, UK: Cambridge University; 1989.

26. Watkins CJCH, Dayan P: **Q-learning.** *Machine Learning* 1992, **8**:279–292.

27. Mackintosh NJ: *Conditioning and Associative Learning.* New York: Oxford University Press; 1983.

28. • Tesauro G J: **Practical issues in temporal difference learning.** *Machine Learning* 1992, **8**:257–277.
This paper describes the backgammon learning system discussed in this review.

29. Tesauro GJ: **Temporal difference learning in TD-Gammon.** *Communications of the ACM* 1994, in press.

30. White DA, Sofge DA (Eds): *Handbook of intelligent control: neural, fuzzy, and adaptive approaches.* New York: Van Nostrand Reinhold; 1992.

31. Montague PR, Dayan P, Nowlan SJ, Pouget A, Sejnowski TJ: **Using aperiodic reinforcement for directed self-organization during development.** In *Advances in neural information processing systems 5.* Edited by Hanson SJ, Cohen JD, Giles CL. San Mateo, California: Morgan Kaufmann; 1993:969–976.

32. Barto AG: **Adaptive critics and the basal ganglia.** In *Models of information processing in the basal ganglia.* Edited by Houk JC, Davis J, Beiser D. Cambridge, Massachusetts: MIT Press; 1995:215–232.

33. • Houk JC, Adams JL, Barto AG: **A model of how the basal ganglia generates and uses neural signals that predict reinforcement.** In *Models of information processing in the basal ganglia.* Edited by Houk JC, Davis J, Beiser D. Cambridge, Massachusetts: MIT Press; 1995:249–270.
The authors present a hypothesis about how circuitry and cellular mechanisms in the basal ganglia could implement a TD adaptive critic computation.

34. Wickens J: **Striatal dopamine in motor activation and reward-mediated learning: steps toward a unifying model.** *J Neural Transm* 1990, **80**:9–31.

35. Ljunberg T, Apicella P, Schultz W: **Responses of monkey dopamine neurons during learning of behavioral reactions.** *J Neurophysiol* 1992, **67**:145–163.

36. Fagg AH, Arbib MA: **A model of primate visual-motor conditional learning.** *Adaptive Behavior* 1992, **1**:3–37.

37. Mazzoni P, Andersen RA, Jordan MI: **A more biologically plausible learning rule for neural networks.** *Proc Natl Acad Sci USA* 1991, **88**:4433–4437.

38. Hampson SE: *Connectionist Problem Solving: Computational Aspects of Biological Learning.* Boston: Birkhauser; 1989.

39. Klopf AH: **A neuronal model of classical conditioning.** *Psychobiology* 1988, **16**:85–125.

40. Sutton RS, Barto AG: **Time-derivative models of Pavlovian reinforcement.** In *Learning and Computational Neuroscience:*

Foundations of Adaptive Networks. Edited by Gabriel M, Moore J. Cambridge, Massachusetts: MIT Press; 1990:497–537.

41. Barto AG: **From chemotaxis to cooperativity: abstract exercises in neuronal learning strategies.** In *The computing neuron.* Edited by Durbin R, Maill R, Mitchison G. Reading, Massachusetts: Addison-Wesley; 1989:73–98.

42. Werbos PJ: **Building and understanding adaptive systems: a statistical/numerical approach to factory automation and brain research.** *IEEE Trans Systems Man Cybernetics* 1987, **17**:7–20.

43. Thrun SB: **The role of exploration in learning control.** In *Handbook of intelligent control: neural, fuzzy, and adaptive approaches.* Edited by White DA, Sofge DA. New York: Van Nostrand Reinhold; 1992:527–559.

44. Singh SP, Jaakkola T, Jordan MI: **Learning without state-estimation in partially observable Markovian decision problems.** In *Maching learning: proceedings of the eleventh international workshop.* Edited by Cohen WW, Hirsch H. San Francisco: Morgan Kaufmann; 1994:284–292.

45. Singh SP: **Transfer of learning by composing solutions for elemental sequential tasks.** *Machine Learning* 1992, **8**:323–339.

46. Barto AG, Gullapalli V: **Neural networks and adaptive control.** In *Neuroscience: from neural networks to artificial intelligence, research notes in neural computation,* vol 4. Edited by Rudomin P, Arbib MA, Cervantes-Perez F, Romo R. Berlin: Springer-Verlag; 1993:483.

AG Barto, Department of Computer Science, University of Massachusetts, Amherst, Massachusetts 01003, USA.