Are People Successful at Learning Sequential Decisions on a Perceptual Matching Task?

Reiko Yakushijin (yaku@cl.aoyama.ac.jp)
Department of Psychology, Aoyama Gakuin University, Shibuya, Tokyo, 150-8366, Japan

Robert A. Jacobs (robbie@bcs.rochester.edu)
Department of Brain & Cognitive Sciences, University of Rochester, Rochester, NY 14627, USA

Abstract

Sequential decision-making tasks are commonplace in our everyday lives. We report the results of an experiment in which human subjects were trained to perform a perceptual matching task, an instance of a sequential decision-making task. We use two benchmarks to evaluate the quality of subjects' learning. One benchmark is based on optimal performance as defined by a dynamic programming procedure. The other is based on an adaptive computational agent that uses a reinforcement learning method known as Q-learning to learn to perform the task. Our analyses suggest that subjects learned to perform the perceptual matching task in a near-optimal manner at the end of training. Subjects were able to achieve near-optimal performance because they learned, at least partially, the causal structure underlying the task. Subjects' learning curves were broadly consistent with those of model-based reinforcement-learning agents that built and used internal models of how their actions influenced the external environment. We hypothesize that, in general, people will achieve near-optimal performances on sequential decision-making tasks when they can detect the effects of their actions on the environment, and when they can represent and reason about these effects using an internal mental model.

Keywords: sequential decision making; optimal performance; dynamic programming; reinforcement learning

Introduction

Tasks requiring people to make a sequence of decisions to reach a goal are commonplace in our lives. When playing chess, a person must choose a sequence of chess moves to capture an opponent's king. When driving to work, a person must choose a sequence of left and right turns to arrive at work in a timely manner. And when pursuing financial goals, a person must choose a sequence of saving and spending options to achieve a financial target. Interest in sequential decision-making tasks among cognitive scientists has increased dramatically in recent years (e.g., Busemeyer, 2002; Chhabra & Jacobs, 2006; Fu & Anderson, 2006; Gibson, Fichman, & Plaut, 1997; Gureckis & Love, 2009; Lee, 2006; Sutton & Barto, 1998; Shanks, Tunney, & McCarthy, 2002).

Here, we are interested in whether people are successful at learning to perform sequential decision-making tasks. There are at least two ways in which the quality of learning can be evaluated. These ways differ in terms of the benchmark to which the performances of a learner are compared. One way uses a benchmark of optimal performance on a task. Analyses based on optimal performance are referred to as ideal observer analyses, ideal actor analyses, or rational analyses in the literatures on perception, motor control, and cognition, respectively. At each moment during training with a task, a learner's performance can be compared to the optimal performance for that task. If a learner achieves near-optimal performance at the end of training, then it can be claimed that the learner has been successful.

A second way of evaluating a learner is to compare the learner's performances with those of an adaptive computational agent that is trained to perform the same task. We consider here an agent that learns via "reinforcement learning" methods developed by researchers interested in artificial intelligence (Sutton & Barto, 1998). Cognitive scientists have begun to use reinforcement learning methods to develop new theories of biological learning (e.g., Busemeyer & Pleskac, 2009; Daw & Touretzky, 2002; Schultz, Dayan, & Montague, 1997; Fu & Anderson, 2006). To date, however, there are few comparisons of the learning curves of people and agents based on reinforcement learning methods. Because reinforcement learning is regarded as effective and well-understood from an engineering perspective, and as plausible from psychological and neurophysiological perspectives, the performances of agents based on this form of learning can provide useful benchmarks for evaluating a person's learning. If a person's performance during training improves at the same rate as that of a reinforcement-learning agent, then it can be argued that the person is a successful learner. If a person's performance improves at a slower rate, then the person is not learning as much from experience as he or she could learn. Experimentation is often required to identify the cognitive "bottlenecks" preventing the person from learning faster. Lastly, if a person's performance improves at a faster rate, then this suggests that the person is using information sources or information processing operations that are not available to the agent. A new, more complex agent should be considered in this case.

We report the results of an experiment in which human subjects were trained to perform a perceptual matching task. This task was designed to contain a number of desirable features. Importantly, the perceptual matching task is an instance of a sequential decision-making task. Subjects made a sequence of decisions (or, equivalently, took a sequence of actions) to modify an environmental state to a goal state. In addition, efficient performance on the perceptual matching task required knowledge of how different properties of an environment interacted with each other. In many everyday tasks, people are required to understand the interactions, or "causal relations", among multiple components (Busemeyer, 2002; Gopnik &

Shulz, 2007). For example, when reaching for a coffee mug, a person must understand that forces exerted at the shoulder also influence the positions and velocities of the elbow, wrist, and fingers. To make an efficient movement, a person must use this knowledge of the causal interactions among motor components to design an effective motor plan.

Subjects' performances on the perceptual matching task were evaluated via two benchmarks. Using an optimization technique known as dynamic programming, optimal performance on this task was calculated. In addition, computer simulations of an adaptive agent were conducted in which the agent was trained to perform the perceptual matching task using a reinforcement learning method known as Q-learning (Sutton & Barto, 1998; Watkins, 1989). Comparisons of subjects' performances during training with optimal performance and with those of the adaptive agent suggest that: (i) subjects learned to perform the perceptual matching task in a near-optimal manner at the end of training; (ii) subjects learned, at least partially, the causal structure underlying the task; (iii) subjects' learning curves were consistent with those of model-based reinforcement-learning agents; and (iv) subjects may have learned by building and using mental models of how their actions influenced the external environment. Additional details and results are reported in Yakushijin & Jacobs (2010).

Experiment

Methods: Twenty-four undergraduate students at the University of Rochester participated in the experiment. Subjects were paid $10 for their participation. All subjects had normal or corrected-to-normal vision. Subjects were randomly assigned to one of six experimental conditions. Each condition included both training and test trials. Only the results of training trials are discussed here due to space limitations.

On a training trial, subjects performed a perceptual matching task which used visual objects from a class of parameterized objects known as "supershapes" (highly realistic but unfamiliar shapes; see Gielis, 2003). The parameters were latent (hidden) variables whose values determined the shapes of the objects. On each trial, subjects viewed a target object, a comparison object, and a set of six buttons (see left panel of Figure 1). Buttons were organized into three pairs, and each pair could be used to decrease or increase the value of an action variable. By pressing the buttons, subjects could change the values of the action variables which, in turn, changed the values of the parameters underlying the comparison object's shape which, in turn, changed the shape of the comparison object. Subjects' task was to press one or more buttons (i.e., to change the values of the action variables) to modify the shape of the comparison object until it matched the shape of the target object using as few button presses as possible.

An experimental condition was characterized by a specific set of causal relations among the latent shape parameters. For example, one such set is schematically illustrated in the right panel of Figure 1. Here, the three action variables are denoted A, B, and C. These variables are observable in the sense that subjects could directly and easily control their values through the use of the buttons. The values of the action variables determined the values of the shape parameters, denoted X, Y, and Z. Note that there are causal relations among the shape parameters. According to the network in Figure 1, if the value of X is changed, then this leads to a modification of Y which, in turn, leads to a modification of Z. The shape parameters determine the shape of the comparison object, whose perceptual features are denoted f_1, f_2, f_3, f_4, f_5, and f_6. The perceptual features used by a subject to assess the similarity of target and comparison object shapes may only be implicitly known by a subject, and may differ between subjects.

Importantly, to efficiently convert the comparison object's shape to the target object's shape (i.e., with the fewest number of button presses) often requires an understanding of the causal relations among the shape parameters. For instance, if the values of parameters X, Y, and Z all need to be modified, a person who does not understand the causal relations among shape parameters may decide to change the value of action variable C (thereby changing shape parameter Z), then the value of action variable B (thereby changing Y and Z), and finally the value of action variable A (thereby changing X, Y, and Z). In many cases, this will be an inefficient strategy. A person with good knowledge of the causal relations among the shape parameters knows that he or she can change the values of X, Y, and Z with a single button press that decreases or increases the value of action variable A. Thus, a good understanding of the causal relations among the shape parameters will lead to efficient task performance, whereas a poor understanding of the causal relations will lead to many more button presses than necessary.

The six experimental conditions differed in the causal relations among the latent shape parameters X, Y, and Z. Two of the causal relations were "linear" structures (one parameter had a direct causal influence on a second parameter which, in turn, had a direct causal influence on a third parameter; e.g., $X \rightarrow Y \rightarrow Z$ or $Y \rightarrow X \rightarrow Z$), two of the relations were "common cause" structures (one parameter had direct causal influences on the two remaining parameters; e.g., $Y \leftarrow X \rightarrow Z$ or $X \leftarrow Y \rightarrow Z$), and two of the relations were "common effect" structures (two parameters had direct causal influences on a third parameter; e.g., $X \rightarrow Y \leftarrow Z$ or $Y \rightarrow X \leftarrow Z$).

An experimental session consisted of 7 blocks of trials where a block contained a set of training trials followed by a set of test trials. (Test trials evaluated subjects' one-step look-ahead knowledge; on a test trial, a subject decided if a comparison object could be converted to a target object using a single button press, and the subject did not receive feedback. Again, test trials are not discussed here.) Each set contained 26 trials, one trial for each possible perturbation of a target object shape to form an initial comparison object shape.

Results: Task Performances: As a benchmark for evaluating subjects' performances on training trials, we computed optimal performances on these trials using an optimization method known as dynamic programming (Bellman, 1957).

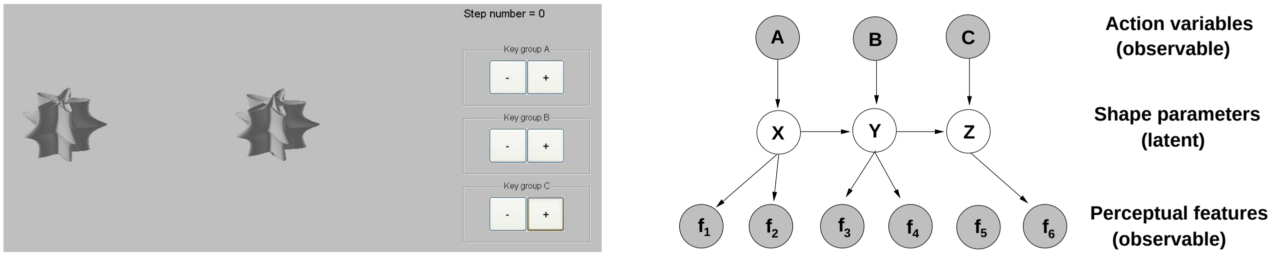

Figure 1: Left: Example of an experimental display. Right: Bayesian network representing the causal relations (in one of the experimental conditions) among the action variables, shape parameters, and perceptual features. For simplicity, the network does not represent the fact that subjects' button presses determined the values of the action variables.

In brief, dynamic programming is a technique for computing optimal solutions to multi-stage decision tasks. That is, dynamic programming finds the shortest sequences of actions that move a system from an initial state to a goal state when all states are fully observable. In the context of a training trial, the initial state corresponds to the initial values of the shape parameters X, Y, and Z for the comparison object, and the goal state corresponds to the values of the shape parameters for the target object. The dynamic programming algorithm is provided with full state information. This means that the algorithm knows the values of the comparison object's shape parameters at every time step. It also knows the state transition dynamics, meaning that it knows the causal relations among the shape parameters and, thus, knows how any button press will change the values of the shape parameters. Relative to our subjects, the dynamic programming algorithm is at an advantage. At the start of the experiment, our subjects did not know the values of the shape parameters or the causal relations among the parameters. Consequently, it would be impressive if subjects learned to perform the task as well as the dynamic programming algorithm.

We determined the optimal performances in the six experimental conditions via dynamic programming. Our analysis revealed that the range (1-5 steps or button presses) and the average length (2.54 steps) of the optimal action sequences were identical for all conditions. Thus, the conditions were well balanced in terms of their intrinsic difficulties.

Figure 2 shows subjects' learning curves on training trials in the two experimental conditions with linear causal structures among shape parameters. Due to space limitations, we do not show results for conditions with common-cause and common-effect structures, though subjects in these conditions showed very similar results to subjects in linear structure conditions (Yakushijin & Jacobs, 2010). Eight subjects participated in linear structure conditions and, thus, the figure contains eight graphs. The horizontal axis of each graph gives the block number, and the vertical axis gives the average difference between the number of steps (i.e., button presses) used by a subject during a trial and the optimal number of steps for that trial as computed by the dynamic programming procedure. These graphs show a number of interesting features. Many subjects found the task to be difficult toward the start of the experiment and, thus, their performances were highly sub-optimal during this time period. However, every subject learned during the course of the experiment. Importantly, every subject achieved near-optimal performance at the end of training: The average difference between a subject's performance and the optimal performance at the end of training is less than 1/2 of a step (mean = 0.434; standard deviation = 0.324).

Results: Causal Learning: The data from the training trials show that subjects achieved near-optimal performances. These results are consistent with the idea that subjects learned about the causal relations among the latent shape parameters. Additional analyses of training and test trials, not described here due to space limitations, confirm that subjects did indeed learn (at least partially) about these causal relations, and that this knowledge played a role in their task performances. Details can be found in Yakushijin & Jacobs (2010).

Reinforcement Learning Agents

Above, our analysis of subjects' data used a benchmark of optimal performance based on dynamic programming. Although very useful, this analysis does not allow us to evaluate the quality of subjects' rates of learning. To do so, we use a different benchmark based on an adaptive computational agent that uses a reinforcement learning method known as Q-learning to learn to perform the perceptual matching task (Sutton & Barto, 1998; Watkins, 1989). Without going into the mathematical details, the reader should note that Q-learning is an approximate dynamic programming method (Si et al., 2004). It is easy to show that, under mild conditions, the sequence of decisions found by an agent using Q-learning is guaranteed to converge to an optimal sequence found by dynamic programming (Watkins & Dayan, 1992). Hence, the benchmarks based on dynamic programming and on Q-learning are related.

In a reinforcement learning framework, it is assumed that an agent attempts to choose actions so as to receive the most reward possible. The agent explores its environment by assessing its current state and choosing an action. After executing this action, the agent will be in a new state, and will receive a reward (possibly zero) associated with this new state.

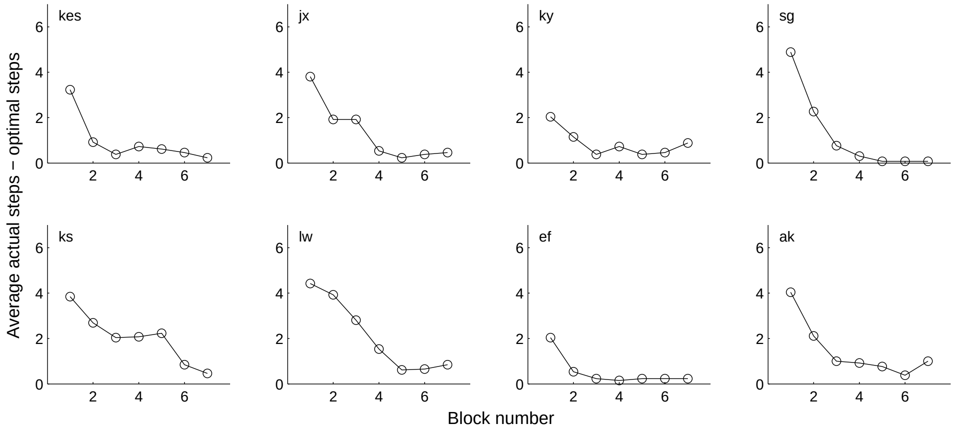

Figure 2: Subjects' learning performances on training trials in the two experimental conditions with linear causal structures among shape parameters (top row: $X \rightarrow Y \rightarrow Z$; bottom row: $Y \rightarrow X \rightarrow Z$).

The agent adapts its behavior in a trial-by-trial manner by noticing which actions tend to be followed by future rewards and which actions are not. To choose good actions, the agent needs to estimate the long-term reward values of selecting possible actions from possible states. Ideally, the value of selecting action a_t in state s_t at time t, denoted $Q(s_t, a_t)$, should equal the sum of rewards that the agent can expect to receive in the future if it takes action a_t in state s_t: $Q(s_t, a_t) = E[\sum_{k=0}^{\infty} \gamma_k \; r_{t+k+1}]$ where t is the current time step, k is an index over future time steps, r_{t+k+1} is the reward received at time $t+k+1$, and γ $(0 < \gamma \leq 1)$ is a term that serves to discount rewards that occur in the far future more than rewards that occur in the near future. An agent can learn accurate estimates of these ideal values on the basis of experience if it updates its estimates at each time step using the equation:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

where the agent makes action a_t in state s_t and receives reward r_{t+1}, and α is a step size or learning rate parameter (Sutton & Barto, 1998; Watkins, 1989).

In our first set of simulations in which a reinforcement-learning agent was trained to perform the perceptual matching task, all "Q-values" were initialized to zero, the discount rate γ was set to 0.7, and the learning rate α was set to 0.45. In preliminary simulations, these values were found to be best in the sense that they led to performances that most closely matched human performances. At each time step, the state of the agent represented the difference in shape between the comparison and target objects. It was a three-dimensional vector whose elements were set to the values of the shape parameters for the comparison object minus the values of these parameters for the target object. Six possible actions were available to the agent corresponding to the six buttons that a subject could press to modify the action variables. The agent chose an action using an ε-greedy strategy, meaning that the agent chose the action a that maximized $Q(s_t, a)$ with probability $1 - \varepsilon$ (ties were broken at random), and chose a random action with probability ε. The value of ε was initialized to one, and then it was slowly decreased during the course of a simulation. As a result, the agent tended to "explore" a wide range of actions toward the beginning of a simulation, and tended to "exploit" its current estimates of the best action to take toward the middle and end of a simulation. If the agent chose an action that caused the comparison object to have the same shape as the target object, the agent received a reward of 100. Otherwise, it received a reward of -1. The agent performed the training trials of the experiment in the same manner as our human subjects—it performed 7 blocks of training trials with 26 trials per block. To accurately estimate the agent's performances during training, the agent was simulated 1000 times.

The results for experimental conditions using linear causal structures are shown in the left graph of Figure 3 (results for other conditions were similar). The horizontal axis plots the block number, and the vertical axis plots the average difference between the number of steps (i.e., actions or button presses) used by the agent or by human subjects during a trial and the optimal number of steps for that trial as computed by the dynamic programming procedure (as in Figure 2; the error bars in Figure 3 indicate the standard deviations). The solid line shows the data for the simulated agent, and the dotted line shows the data for our human subjects. Interestingly, the learning curves of the simulated agent and of the human subjects have similar shapes, though subjects learned faster than the agent at nearly all stages of training in all experimental conditions. Modifications of the agent by either using different values for the agent's parameters or by adding "eligibility traces" did not significantly alter this basic finding.

Why did subjects show better learning performances than the simulated agent? In the machine learning literature, a distinction is made between model-free versus model-based reinforcement learning agents. The agent described above is an instance of a model-free agent. Although model-free agents are more common in the literature, we hypothesized

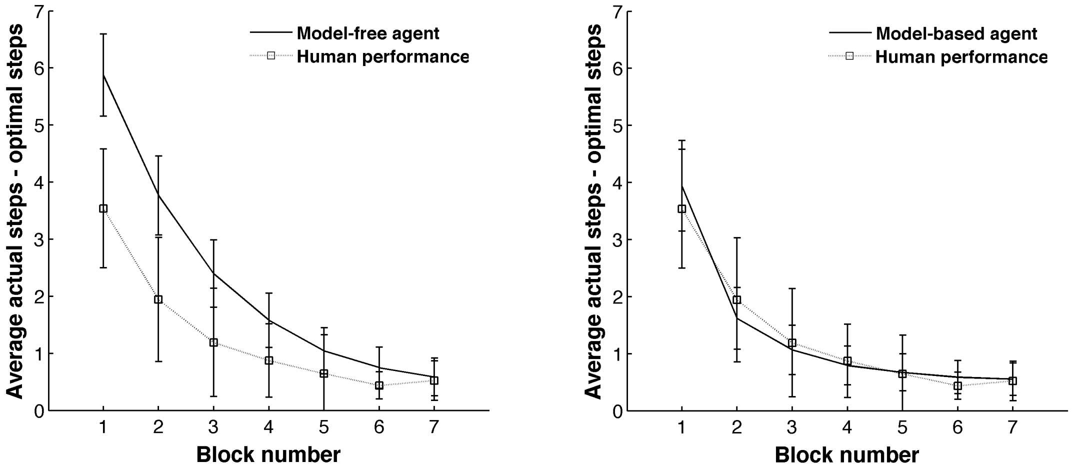

Figure 3: Left: Learning curves for the simulated agent trained via Q-learning (solid line) and for the human subjects (dotted line) in experimental conditions using linear causal structures (error bars plot standard deviations). Right: Identical to the left graph except that the simulated agent learned a model of how actions influenced the environment, and used this model to reason about good actions to take at each time step.

that a model-based reinforcement learning agent may provide a better account of our subjects' performances. Model-based agents typically learn faster than model-free agents, albeit with greater computational expense. Based on real-world experiences, a model-based agent learns an internal model of how its actions influence the environment. The agent updates its Q-values from both real-world experiences with the environment and from simulated experiences with the model (see Sutton and Barto, 1998, for details).

In our simulations, the model was an artificial neural network. Its six input units corresponded to the six possible actions or key presses (an action variable could either increase or decrease in value, and there were three action variables). Its nine output units corresponded to the nine possible influences on the comparison objects' shape parameters (a shape parameter could either increase in value, decrease in value, or maintain the same value, and there were three shape parameters). The network did not contain any hidden units.

When updating its Q-values, the model-based agent used 'prioritized sweeping' (Moore & Atkeson, 1993). This is an efficient method for focusing Q-value updates to state-action pairs associated with large changes in expected reward. Large changes occur, for example, when the current state is a non-goal state and the agent discovers a previously unfamiliar action that leads to a goal state. Large changes also occur when the current state is a non-goal state, and the agent discovers a new action that leads to a new non-goal state known to lie on a path toward a goal state.

In brief, our simulations used prioritized sweeping as follows. At each moment in time, the model-based agent maintained a queue of state-action pairs whose Q-values would change based on either real or simulated experiences. For each update based on a real experience, there were up to N updates based on simulated experiences. The items on the queue were prioritized by the absolute amount that their Q-values would be modified. For example, suppose that at some moment in time, state-action pair (s^*, a^*) had the highest priority. Then $Q(s^*, a^*)$ would be updated. If performing this update on the basis of simulated experience, the agent used the model to predict the resulting new state. In addition, the agent also used the model to examine changes to the Q-values for all state-action pairs predicted to lead to state s^*, known as predecessor state-action pairs. These predecessor state-action pairs were added to the queue, along with their corresponding priorities.

The simulations with the model-based agent were identical to those with the model-free agent. However, the model-based agent used different parameter values. Its discount rate γ was set to 0.3, its learning rate α was set to 0.05, and N, the number of Q-value updates based on simulated experiences for each update based on a real experience, was set to 5. In preliminary simulations, these values were found to be best in the sense that they led to performances that most closely matched human performances.

The combined results for the experimental conditions using linear causal structures are shown in the right graph of Figure 3 (once again, results for the other experimental conditions were similar). The learning curves of the model-based agent are more similar to those of human subjects than the curves of the model-free agent. Indeed, the curves of the model-based agent and of the human subjects are nearly identical. Our findings suggest (but do not prove) that subjects may have achieved near-optimal performances on the perceptual matching task by building internal models of how their actions influenced the external environment. By using these models to reason about possible action sequences, subjects quickly learned to perform the task.

Conclusions

Sequential decision-making tasks are commonplace in our everyday lives. Here, we studied whether people were successful at learning to perform a perceptual matching task, an instance of a sequential decision-making task. We used two benchmarks to evaluate the quality of subjects' learning. One benchmark was based on optimal performance as defined by a dynamic programming procedure. The other was based on an adaptive computational agent that used Q-learning to learn to perform the task. Overall, our analyses suggest that subjects learned to perform the perceptual matching task in a near-optimal manner. When doing so, subjects learned, at least partially, the causal structure underlying the task. In addition, subjects' learning curves were broadly consistent with those of model-based reinforcement-learning agents that built and used internal models of how their actions influenced the external environment.

The cognitive science literature now contains several studies of human performance on sequential decision-making tasks. Some studies have suggested that human performance is optimal, whereas other studies have suggested the opposite. To date, our field does not have a good understanding of the factors influencing whether people will achieve optimal performance on a task. Future research will need to focus on this critical issue. Previous articles in the literature suggested that perceptual aliasing (Stankiewicz et al., 2006) or the existence of actions leading to large rewards in the short-term but not the long-term (Neth, Sims, & Gray, 2006; Gureckis & Love, 2009) seem to be factors leading to sub-optimal performance. Here, we propose a new understanding of when people will (or will not) achieve optimal performance. We hypothesize that people will achieve near-optimal performance on sequential-decision making tasks when they can detect the effects of their actions on the environment, and when they can represent and reason about these effects using an internal mental model.

Acknowledgments

This work was supported by a Grant-in-Aid for Scientific Research (#20730480) from the Japan Society for the Promotion of Science, by a research grant from the Air Force Office of Scientific Research (FA9550-06-1-0492), and by a research grant from the National Science Foundation (DRL-0817250).

References

Bellman, R. (1957). *Dynamic Programming*. Princeton, NJ: Princeton University Press.

Busemeyer, J. R. (2002). Dynamic decision making. In N. J. Smelser & P. B. Baltes (Eds.), *International Encyclopedia of the Social and Behavioral Sciences*. Oxford, UK: Elsevier Press.

Busemeyer, J. R. & Pleskac, T. J. (2009). Theoretical tools for understanding and aiding dynamic decision making. *Journal of Mathematical Psychology*, *53*, 126-138.

Chhabra, M. & Jacobs, R. A. (2006). Near-optimal human adaptive control across different noise environments. *The Journal of Neuroscience*, *26*, 10883-10887.

Daw, N. D. & Touretzky, D. S. (2002). Long-term reward prediction in TD models of the dopamine system. *Neural Computation*, *14*, 2567-2583.

Fu, W.-T. & Anderson, J. R. (2006). From recurrent choice to skill learning: A reinforcement-learning model. *Journal of Experimental Psychology: General*, *135*, 184-206.

Gibson, F. P., Fichman, M., & Plaut, D. C. (1997). Learning in dynamic decision tasks: Computational model and empirical evidence. *Organizational Behavior and Human Decision Processes*, *71*, 1-35.

Gielis, J. (2003). A generic geometric transformation that unifies a wide range of natural and abstract shapes. *American Journal of Botany*, *90*, 333-338.

Gopnik, A. & Shulz, L. (2007). *Causal Learning: Psychology, Philosophy, and Computation*. New York: Oxford University Press.

Gureckis, T. M. & Love, B. C. (2009). Short-term gains, long-term pains: How cues about state aid learning in dynamic environments. *Cognition*, *113*, 293-313.

Lee, M. D. (2006). A hierarchical Bayesian model of human decision-making on an optimal stopping problem. *Cognitive Science*, *30*, 1-26.

Moore, A. & Atkeson, C. (1993). Prioritized sweeping: Reinforcement learning with less data and less real time. *Machine Learning*, *13*, 103-130.

Neth, H., Sims, C. R., & Gray, W. D. (2006). Melioration dominates maximization: Stable suboptimal performance despite global feedback. *Proceedings of the 28th Annual Meeting of the Cognitive Science Society*.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*, 1593-1598.

Shanks, D. R., Tunney, R. J., & McCarthy, J. D. (2002). A re-examination of melioration and rational choice. *Journal of Behavioral Decision Making*, *15*, 233-250.

Si, J., Barto, A. G., Powell, W. B., & Wunsch, D. (2004). *Handbook of Learning and Approximate Dynamic Programming*. Piscataway, NJ: Wiley-IEEE.

Stankiewicz, B. J., Legge, G. E., Mansfield, J. S., & Schlicht, E. J. (2006). Lost in virtual space: Studies in human and ideal spatial navigation. *Journal of Experimental Psychology: Human Perception and Performance*, *32*, 688-704.

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

Watkins, C. J. C. H. (1989). Learning From Delayed Rewards. Unpublished doctoral dissertation. Cambridge, UK: Cambridge, University.

Watkins, C. J. C. H. & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*, 279-292.

Yakushijin, R. & Jacobs, R. A. (2010). Are people successful at learning sequential decisions on a perceptual matching task? Manuscript submitted for journal publication.