Table of Contents

Go Back

Learning from Experience with Delayed Feedback

Hazhir Rahmandad
Ph.D. Student, Sloan School of Management, M.I.T
E53-364A, 30 Wadsworth Ave.,
Cambridge, MA 02142, U.S.A
617-253-3865
hazhir@mit.edu

Nelson Repenning
Associate Professor, Sloan School of Management, M.I.T
E53-335, 30 Wadsworth Ave.,
Cambridge, MA 02142,
U.S.A
617-258-6889
nelson@mit.edu

John Sterman
Professor, Sloan School of Management, M.I.T
E53-351, 30 Wadsworth Ave.,
Cambridge, MA 02142, U.S.A
617-253-1951
jsterman@mit.edu

Abstract

Many important settings in individual and organizational life involve allocating resources between different types of activities with different delays between allocation and results. Examples include factory managers choosing to spend time on production now or on process improvement that may boost output later, and individuals choosing to get a job now or stay in school to get a better job later. Empirical studies show that learning is difficult in dynamic systems and people often fail to learn from experience in the presence of delays. Understanding the processes that hinder learning from experience is central to improving learning and decision-making. We use a formal model to examine the effect of time delays on learning from experience. The model represents a decision-maker engaged in a continuous time allocation task who learns from his performance as he tries to improve the payoff determined by his own actions. Our analysis shows that in an easy learning task where the payoff landscape is smooth and has only a single peak, we can still observe sub-optimal performance. Moreover, the decision maker can learn to believe that the sub-optimal performance is really the best she can do.

Keywords

Barriers to Learning, Reinforcement Learning, Modeling Learning

1-Introduction

Many important situations in individual and organizational life involve allocating resources between different types of activities with different delays between allocation and results. These situations, often involving tradeoffs between short term and long-term results, are found in diverse fields. For example, an individual must allocate her time between different activities, some satisfying her daily needs and some contributing to her long-term goals. At a more aggregate level of analysis, a factory can boost output in the short run by cutting maintenance, but in the long run, output falls as breakdowns increase. Other examples include learning and process improvement (Repenning and Sterman 2002) and environmental issues (Meadows et al., 1972). Often such settings entail a worse-before-better dynamic in which system performance falls in the short run but improves in the long run (and vice-versa); Forrester (1969) described such worse-before-better tradeoffs in urban and other public policy settings.

Individuals, organizations and societies often fail to learn from experience to improve their performance in these allocation and decision-making tasks. Studies show that learning in complex dynamic systems is often difficult (Sterman 1994, Paich and Sterman 1993, Diehl and Sterman 1995, Dörner 1996). Understanding the processes that hinder learning from experience is central to improving learning and decision-making.

Researchers have suggested a few factors that contribute to poor learning. One is misperceptions of feedback (Sterman 1994), including the difficulties people have in recognizing feedback loops, time delays, stocks and flows, nonlinearities and other structural elements common in complex dynamic systems. Other theories stress the complexity of the payoff landscape, focusing on the potential for people (and machine learning processes) to become stuck at a local optimum in a rugged landscape (Levinthal, 1997;Busemeyer, 1986). Our focus in this paper is the role of time delays in impeding learning. While studies show how time delays degrade decision-making quality (Brehmer, 1992; Paich and Sterman, 1993; Diehl and Sterman, 1995), there are few if any formal models examining how learning may be affected by the presence of time delays.

Our study was motivated by recent fieldwork in a manufacturing company. Repenning and Sterman (2002) found that managers learned the wrong lessons from their interactions with the workforce as a result of different time delays inherent in how various types of worker activity influence the system's performance. Specifically, managers seeking to meet production targets had two basic options: (1) increase process productivity and yield through better maintenance and investment of time and resources in improvement activity; or (2) pressure the workforce to "work harder" through overtime, speeding production, taking fewer breaks, and, most importantly, by cutting back on the time devoted to maintenance and process improvement. Though the study found, consistent with the extensive quality improvement literature, that "working smarter" provides a greater payoff than working harder, many organizations find themselves stuck in a trap of working harder, resulting in reduced maintenance and improvement activity, lower productivity, greater pressure to hit targets and thus even less time for improvement and maintenance (Repenning and Sterman 2001, 2002).

What makes these situations theoretically interesting, as well as of practical importance, is that people often consistently learn the wrong lessons from their experience, incorrectly interpreting the outcome feedback they receive as reinforcing the wisdom of a harmful course of action. Consider a manager facing a production shortfall. Pressuring the employees to work harder generates a short run improvement in output, as they reallocate time from improvement to production. The resulting decline in productivity and equipment uptime, however, comes only with a delay. It appears to be difficult for many managers to recognize and account for such delays, so they conclude that pressuring their people to work harder was the right thing to do, even if it is in fact harmful. Repenning and Sterman (2002) show how, over time, managers develop strongly held beliefs that pressuring workers for more output is the best policy, that the workers are intrinsically lazy and require continuous supervision, and that process improvement is ineffective or impossible in their organization—even when these beliefs are false.

In this paper we develop a formal model to examine the effect of time delays on learning from experience. The model represents a decision-maker engaged in a continuous-time resource allocation task. As in many real world tasks, the decision-maker must simultaneously allocate resources (make decisions) and try to learn from experience how to make better allocation decisions to achieve better outcomes.

We investigate the effect of different delays between resource allocation decisions (activities) and results (the payoff) on the ability of the decision-maker to find the optimal payoff. We draw on current literature on learning in psychology, game theory and attribution theory to model learning, and investigate four different learning procedures with different levels of sophistication, rationality, and information processing requirements. By including these different models, we can distinguish the effects of decision-making rationality from learning problems arising from delayed feedback. Our analysis shows that performance can still be significantly sub-optimal in very easy learning tasks, specifically, even when the payoff landscape is unchanging, smooth, and has a unique optimum. Moreover, the decision-maker can learn to believe that the sub-optimal performance is really the best she can do. The difficulty of learning in the presence of delays appears to be independent of the specific learning procedures we examined and is instead rooted in the delays between actions and outcomes.

In the next section we describe the structure of the model and discuss the different learning procedures in detail. The "Results and Analysis" section presents a base case demonstrating that all four learning procedures can discover the optimum allocation in the absence of action-payoff delays. Next we analyze the performance of the four learning procedures in the presence of action-payoff delays, followed by tests to examine the robustness of these results under different parameter settings. We close with discussion of the implications, limitations, and possible extensions.

2-The Model

Our model represents a decision-maker engaged in a continuous-time resource allocation task. The manager must allocate a fixed resource among different activities; each allocation generates a payoff. The payoff can depend on the lagged allocation of resources, and there may be different delays between the allocation of resources to each activity and its impact on the payoff. The decision maker receives outcome feedback about the payoff and past resource allocations (possibly after some reporting delays), and must attempt to learn from these actions how to adjust resource allocations to improve performance (Figure 1).

As a concrete example, consider a plant manager allocating the time of his employees (the resource) among three activities: production, maintenance and process improvement. These activities influence production, but with different delays. Time spent on production yields results almost immediately. There is a longer delay between a change in maintenance activity and machine uptime (and hence production). Finally, it takes even longer for process improvement activity to affect output.

The plant manager gains experience by seeing the results of his past decisions and seeks to increase production based on these experiences. He has some understanding of the complicated mechanisms involved in controlling production and he may be aware of the existence of different delays between each activity and observed production. Consequently, when evaluating the effectiveness of his past allocation decisions, he takes these delays into consideration (e.g., he does not expect last week's process improvement effort to enhance production today). However, his mental model of the production process may be imperfect, and there may be discrepancies between the length of the delays he perceives and the real delays.

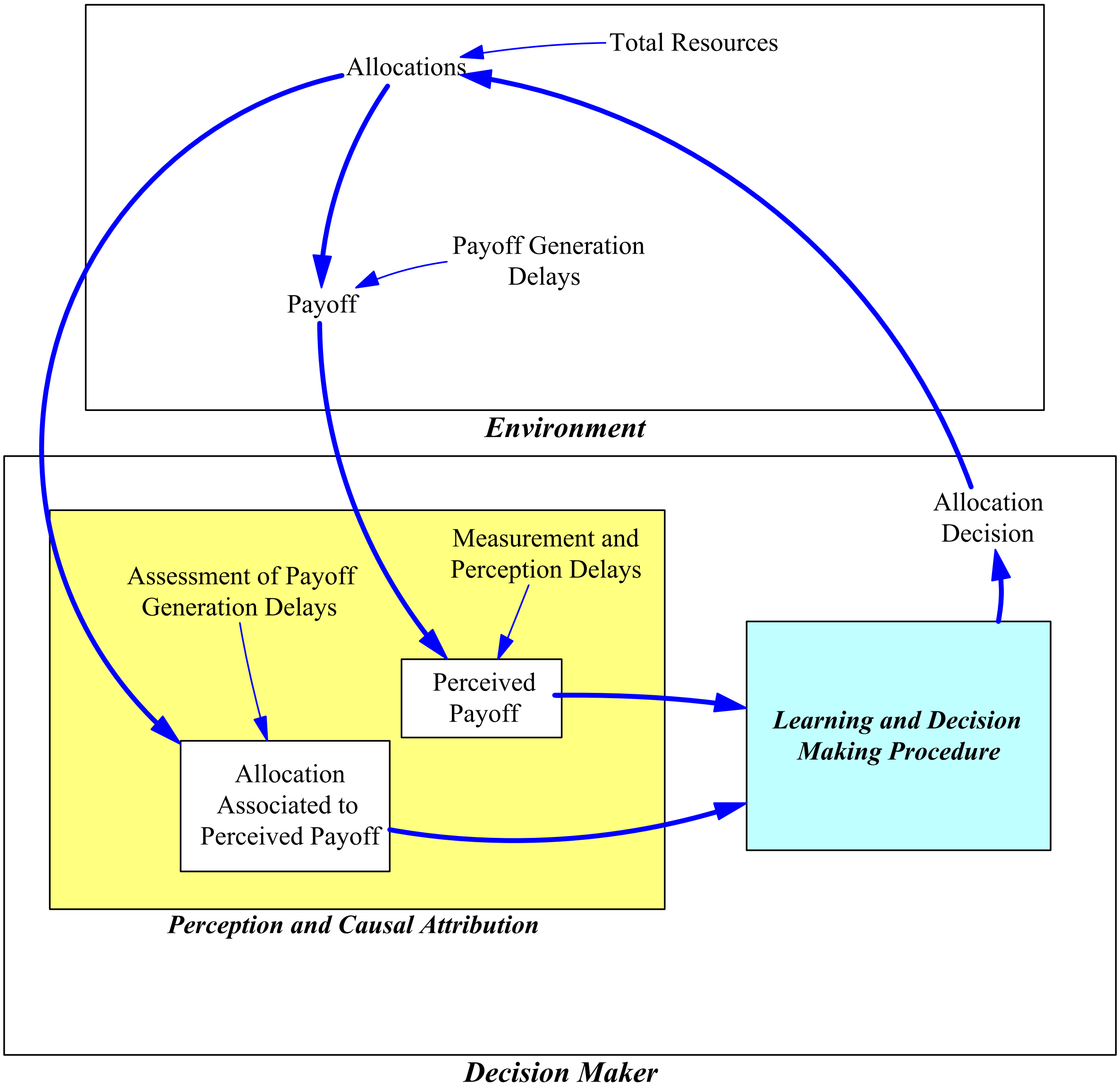
Total Resources
Allocations
Payoff Generation Delays
Payoff
Environment
Allocation Decision
Measurement and Perception Delays
Assessment of Payoff Generation Delays
Perceived Payoff
Learning and Decision Making Procedure
Allocation Associated to Perceived Payoff
Perception and Causal Attribution
Decision Maker

Figure 1- Learning Model Overview

2-1- Allocation and payoff

The decision maker continuously allocates a fraction of total resources to activity j of m possible activities at time t, $FR_j(t)$[i] where:

$$\sum_J FR_j(t) = 1 \qquad \textbf{For j:1,..., m} \qquad \textbf{(1)}$$

In our simulations we assume m = 3 activities, so the decision-maker has two degrees of freedom. Three activities keep the analysis simple while still permitting different combinations of delay and impact for each. Total resources, $R(t)$, are assumed to be constant so $R(t) = R$, and resources allocated to activity j at time t, $A_j(t)$ are:

$$A_j(t) = FR_j(t) * R \qquad \textbf{(2)}$$

These allocations can influence the payoff with some delay. The payoff at time t is determined by the Effective Allocation, $EA_j(t)$, which can lag behind A_j with a payoff generation delay ofT_j. We assume a pipeline delay for simplicity[ii]:

$$EA_j(t) = A_j(t - T_j) \qquad \textbf{(3)}$$

The payoff generation delays are fixed but can be different for each activity.
In our plant manager example, R is the total person hours available in one time period (day, week, etc),$FR_j(t)$ is the fraction of the workers' time the manager allocates to activity j (j: producing, maintenance, process improvement) and $A_j(t)$ is total person-hours/period spent on activity j. The delays in the impact of these activities on production (the payoff) are presumably ordered approximately as $0 \approx T_{production} < T_{maintenance} < T_{improvement}$.

For simplicity we assume the payoff, $PF(t)$, to be a constant-returns-to-scale, Cobb-Douglas function of the effective allocations:

$$PF(t) = \prod_J EA_j(t)^{\alpha_j} \qquad , \sum_j \alpha_j = 1 \qquad \textbf{(4)}$$

The Cobb-Douglas function provides a smooth payoff function with a single peak—yielding a very simple learning task compared to most real life situations. We selected this payoff function to concentrate solely on the effects of delays on learning, eliminating the problems in learning that arise in more complicated landscapes with multiple peaks or payoff landscapes that change over time.

2-2- Perception delays

The decision-maker perceives her own actions and payoff, then she uses this information to learn about the efficiency of different allocation profiles and to come up with better allocations. We assume decision makers account for the delays between past allocations and payoffs, but recognize that the decision maker's estimate of the length of these delays may not be correct.

In real systems it takes time to measure, report and perceive information such as activities and payoffs, so our model provides for the possibility that the perceived payoff, $PPF(t)$, differs from the actual payoff, $PF(t)$. However, to keep things simple and give learning algorithms the most favorable circumstances, here we assume these delays to be zero and measurement and perception to be fast and unbiased, so *PPF(t) = PF(t)*.

The decision-maker accounts for the delays between allocations and payoff based on her beliefs about the length of the payoff generation delays, τ_j. She attributes the current observed payoff, $PPF(t)$, to allocations she made τ_j periods ago, so the allocation attributed to the current payoff, $APF_j(t)$, is:

$$APF_j(t) = A_j(t - \tau_j) \quad \textbf{(5)}$$

The values of $APF_j(t)$ and $PPF(t)$ are used as inputs to the various learning and decision-making procedures representing decision maker's behavior.

With our plant manager, for example, the production record represents $PPF(t)$, and we assume current production is immediately available to and perceived by the manager. The manager understands that the current production rate is the result of the worker time he has allocated to production, maintenance and process improvement activities τ_1, τ_2 and τ_3 periods ago.

2-3- Learning and decision-making procedures

Having perceived her own actions and payoff streams, the decision-maker learns from her experience, that is, selects what she believes is a better set of allocations and updates her beliefs about the efficiency of different allocations (Figure 1). We developed four different learning algorithms to explore the sensitivity of the results to different assumptions about how people learn. The inputs to all of these modules are the perceived payoff and action associated with that payoff, $PPF(t)$ and $APF_j(t)$; the outputs of the algorithms are the allocation decisions $FR_j(t)$. The learning algorithms differ in their level of rationality, information processing requirements, and assumed prior knowledge about the shape of the payoff landscape. Here, rationality indicates decision-maker's ability to make the best use of the information available by trying explicitly to optimize her allocation decisions. Information processing capability indicates her cognitive capacity for keeping track of and using information about past allocations and payoffs. Prior knowledge about the shape of the payoff landscape determines her ability to use off-line cognitive search (Gavetti and Levinthal, 2000) to find better policies.

We use four learning models we denote *Reinforcement*, *Myopic Search*, *Correlation* and *Regression*. The *Reinforcement* and *Myopic Search* algorithms assume a low level of rationality for the decision-maker, specifically, that she is unable to infer the general shape of payoff landscape and instead searches the neighborhood of her current allocations and adopts allocations she finds to be better than those currently used. These models also assume relatively low information processing capacity for the decision-maker has and no prior knowledge about the shape of the payoff landscape.

The correlation model assumes a higher level of rationality for the decision-maker in that she can infer the direction in which better allocation policies may be found by extrapolating observed performance data, based on the correlation between allocations and the payoff. It also includes higher information processing capacity assuming that the decision-maker engages in some calculations in order to infer the correlation between action and payoff. However the correlation method does not assume the decision maker has any prior knowledge about the shape of the payoff landscape.

In the Regression model the decision-maker is endowed with a high level of rationality. Every few time periods the decision maker runs a regression to estimate the best allocation policy based on all the information she has received so far. The regression method requires extensive information processing capacity to frequently perform regressions over the stored data and solve the equations for the optimal allocations given the estimated regression coefficients. Finally, the decision-maker is assumed to know the true shape of the payoff landscape (she is given a perfectly specified model and only has to estimate its parameters). Table 1 summarizes the characteristics of the different models on these dimensions.

Table 1- Sophistication and rationality of different learning models

Dimension ***Algorithm***	***Rationality***	***Information Processing Capacity***	***Prior Knowledge of Payoff Landscape***
Reinforcement	Low	Low	Low
Myopic Search	Low	Low	Low
Correlation	Medium	Medium	Low
Regression	High	High	High

In all the learning modules, the decision-maker has a mental representation of how important each activity is. We call these variables "Activity Value," $AV_j(t)$. The activity values are used to determine the allocation of resources to each activity (equations 12-14 explain how allocation of resources is determined based on Activity Values.) The main difference across the different learning algorithms is how these activity values are updated. Below we discuss the four learning algorithms in more details. The complete formulation of all the learning models can be found in the Appendix 1.

1- *Reinforcement learning:* In this method, the value (or attractiveness) of each activity is determined by (a function of) the cumulative payoff achieved so far by using that alternative. Attractiveness then influences the probability of choosing each alternative in the future. Reinforcement learning has a rich tradition in psychology, game theory and machine learning (Sutton and Barto 1998; Erev and Roth 1998). It has been used in a variety of applications, from training animals to explaining the results of learning in games and designing machines to play backgammon (Sutton and Barto, 1998)[iii].

In our model, each perceived payoff, $PPF(t)$, is associated with the allocations believed to be responsible for that payoff, $APF_j(t)$. We increase the value of each activity, $AV_j(t)$, based on its contribution to the perceived payoff. The increase in value depends on the perceived payoff itself, so a small payoff indicates a small increase in the values of different activities while a large payoff increases the value much more. Therefore large payoffs shift the relative weight of different activity values towards the allocations responsible for those better payoffs.

$$\frac{d}{dt}AV_j(t) =$$
$$PPF(t)^{Re\,inf\,orcementPower} * APF_j(t) - AV_j(t) / Reinforcement\ Forgetting\ Time \qquad \textbf{(6)}$$

Our implementation of reinforcement learning includes a forgetting process, representing the decision-maker's discounting of older information. Discounting old information helps the algorithm adjust the activity values better. Equation 6 shows the main formulation of the *Reinforcement* algorithm. Here both "Reinforcement Power" and "Reinforcement Forgetting Time" are parameters specific to this algorithm. "Reinforcement Power" indicates how strongly we feedback the payoff as reinforcement, to adjust the "Activity Values" and therefore determines the speed of converging to better policies. "Reinforcement Forgetting Time" is the time constant for depreciating the old payoff reinforcements.

Details of the model formulations are included in appendix 1. The *Reinforcement* method is a low information, low rationality procedure: it continues to do what has worked well in the past, adjusting only slowly to new information, and does not attempt to extrapolate from these beliefs about activity value to the shape of the payoff landscape or to use gradient information to move towards higher-payoff allocations.

2- *Myopic search*: In this method the decision maker explores neighboring regions of the decision space (at random). If a better set of resource allocations is found, it is selected as a goal; otherwise the current activity values are retained. This procedure is similar to the underlying process for most of the stochastic optimization techniques where, unaware of the shape of payoff landscape, the algorithm explores different possibilities and usually moves to better policies upon discovering them.

Optimization models assume decisions switch instantly to better policies, if found, for the next step. This assumption is also used in some behavioral models (Levinthal and March, 1982; Levinthal 97). In reality, decisions adjust slowly, due to the time required to perceive new information, make decisions, and implement them; that is, to the factors that lead to organizational inertia. To be behaviorally more realistic, we assume activity value, $AV_j(t)$, adjusts gradually towards the best currently known allocation, $AV_j^*(t)$ (Equation 7.) $AV_j^*(t)$ is the last allocation that improved the payoff from its recent average.

$$\frac{d}{dt}AV_j(t) = (AV_j^*(t) - AV_j(t)) \big/ \lambda \qquad \textbf{(7)}$$

where λ is the Value Adjustment Time Constant. The formulation details for *Myopic search* method can be found in the appendix 1. The myopic method is a low information, low rationality method. It compares the payoff from the current allocation to the results of a local exploration and does not attempt to compare multiple experiments; it does not use information about the payoffs in the neighborhood to make any inferences about the shape of the payoff landscape or even the local gradient. It is essentially conservative, retaining the current allocation until a better one is found.

3- *Correlation*: This method uses principles from attribution theories to model learning. In our setting, learning can be viewed as how people attribute different allocations to different payoffs and how these attributions are adjusted as new information about payoffs and allocations are continuously perceived. Several researchers have proposed different models for explaining how people make attributions. Lipe (1991) reviews these models and concludes that all major attribution theories are based on the use of counterfactual information. However it is difficult to obtain counterfactual information (information about contingencies that were not realized) so she proposes the use of covariation data as a good proxy. The correlation module is based on the hypothesis that people use the covariation of different activities with the payoff to make inferences about action-payoff causality.

In our model, the correlations between the perceived payoff and the actions associated with those payoffs let the decision-maker decide whether performance would improve or deteriorate if the activity increases. A positive (negative) correlation between recent values of $APF_j(t)$ and $PPF(t)$ suggests that more (less) of activity j will improve the payoff. Based on these inferences the decision-maker adjusts the activity values, $AV_j(t)$, so that positively correlated activities increase above their current level and negatively correlated activities decrease below the current level.

$$\frac{d}{dt}AV_j(t) = AV_j(t) * (f(Action\_PayoffCorrelation_j(t)) - 1)/\lambda$$

$$f(0)=1\ ,\ f'(x)>0 \qquad \textbf{(8)}$$

The formulation details for the correlation algorithm are found in Appendix 1, Table 3. At the optimal allocation, the gradient of the payoff with allocation will be zero (the top of the payoff hill is flat) and so the correlation between activities and payoff will be zero. Therefore the change in activity values will become zero and the decision-maker settles on the optimum policy. The correlation method is a moderate information, moderate rationality approach: more data is needed than required by the myopic or reinforcement methods to estimate the correlations among activities and payoff, and these correlations are used to make inferences about the local gradient so the decision maker can move uphill from the current allocations to allocations believed to yield higher payoff, even if these allocations have not yet been tried.

4- *Regression*: This method is a sophisticated learning model with significant information processing requirements. We assume that the decision-maker knows the correct shape of the payoff landscape and uses a correctly specified regression model to estimate the parameters of the payoff function.

By observing the payoffs and the activities corresponding to those payoffs, the decision-maker receives the information needed to estimate the parameters of the payoff function. To do so, after every few periods, she runs a regression using all the data from the beginning of the learning task[iv]. From these estimates the optimal allocations are readily calculated. For the assumed constant returns to scale, Cobb-Douglas function, the regression is:

$$\log(PPF(t)) = \log(\alpha_0) + \sum_j \alpha_j * \log(APF_j(t)) + e(t) \tag{9}$$

The estimates of α_j ,α_j^*, are evaluated every "Evaluation Period," *EP*. Based on these estimates, the optimal activity values, $AV_j^*(t)$, are given by:

$$AV_j^*(t) = Max(\alpha_j^* \Big/ \sum_J \alpha_j^*, 0) \tag{10}$$ [v]

The decision-maker then adjusts the action values towards the optimal values (see Equation 7).

The regression model helps test how delays affect learning over a wide range of rationality assumptions. The three other models represent an individual ignorant about the shape of the payoff function, while in some cases decision-makers have at least partial understanding of the structure and functional forms of the causal relationships relating the payoff to the activities. Although in feedback-rich settings, mental models are far from perfect and calculation of optimal decision based on the understanding of the mechanisms exceeds our cognitive capabilities, the regression model offers an extreme case of rationality to test the robustness of our results.

The tradeoff between exploration and exploitation is a crucial issue in learning (Sutton and Barto, 1998; March, 1993). On one hand the decision-maker should explore the decision space by trying some new allocation policies if she wants to learn about the shape of the payoff function. On the other hand pure exploration leads to random allocation decisions with no improvement. By using the data from exploration, the decision-maker can focus on better policies that she has found and therefore improve her payoff. This exploitation policy is required if any improvement is to be perceived.

We use random changes in resource allocation to capture the exploration/exploitation issue in our learning models. The "Activity Values" represent the accumulation of experience and learning by the decision-maker. In pure exploitation, of the set of activity values, $AV_j(t)$, determine the allocation decision. Specifically, the fraction of resources to be allocated to activity j is:

$$FR_j'(t) = AV_j(t) \Big/ \sum_J AV_j(t) \tag{11}$$

The tendency of the decision-maker to follow this policy shows her tendency to exploit the experience she has gained so far. Deviations from this policy represent experiments to explore other regions of the payoff landscape. We multiply the activity values by a random disturbance to generate Operational Activity Values, $OAV_j(t)$'s, which are the basis for the allocation decisions.

$$OAV_j(t) = AV_j(t) * (1 + PN_j(t)) \tag{12}$$

$$FR_j(t) = OAV_j(t) \Big/ \sum_J OAV_j(t) \tag{13}$$

$PN_j(t)$ is a pink noise term specific to each activity. Pink noise (Sterman, 2000) generates a stream of random numbers with autocorrelation. In real settings, the decision-maker experiments with a policy for some time before moving to another. Such persistence is both physically required (decision makers cannot instantly change resource allocations), and provides decision makers with a large enough sample of data about each allocation to decide if a policy is beneficial or not. A high value of the autocorrelation time constant, or "Activity noise correlation time", δ, means the stream $PN_j(t)$ is highly autocorrelated and the random numbers do not change abruptly, representing decision makers who only slowly change the regions of allocation space they are exploring; a low value represents decision makers who jump quickly from one allocation to another in the neighborhood of their current policy.[vi]

The standard deviation of the noise term, which determines how far from the current allocations she explores, depends on where she finds herself on the payoff landscape. If her recent explorations have shown no improvement in the payoff, she concludes that she is near the peak of the payoff landscape and therefore extensive exploration is not required (alternatively, she concludes that the return to exploration is low and reduces experimentation accordingly). If, however, recent exploration has resulted in finding significantly better regions, she concludes that she is in a low-payoff region and there is still room for improvement so she should keep on exploring, so Var ($PN_j(t)$) remains large:

$$Var\ (PN_j(t)) = g\ (Recent\ Payoff\ Improvement(t)),$$
$$g(0)=Minimum\ Exploration\ Variance,\ g'(x)>0 \qquad \textbf{(14)}$$

Here "*Recent Payoff Improvement(t)"* is calculated by comparing the observed payoff to its recent average. If the current payoff is higher than recent payoff, *Recent Payoff Improvement* will be increased, if not, it will decay towards 0.

3- Results and Analysis

In this section we investigate the behavior of the learning model under different conditions. We first explore the basic behavior of the model and its learning capabilities when there are no delays. The no-delay case helps compare the capabilities of the different learning modules and provides a base against which we can compare the behavior of the model under other conditions. Next we introduce delays between activities and payoff and analyze the ability of the different learning modules to find the optimal payoff, for a wide range of parameters.

In running the model, the total resource, R, is 100 units and the exponents in the payoff function are set to 1/2, 1/3 and 1/6 for activities one, two and three, respectively. The optimal allocation fractions are therefore 1/2, 1/3 and 1/6. The decision-maker starts from a random allocation and tries to improve her performance in the course of 240 periods.[vii]

We report two aspects of learning performance: (1) How close to optimal the simulated decision-maker gets, and (2) how fast she converges to that level of performance, if she converges at all. Given the parameters for the payoff function, the optimal payoff is 36.37 and the percentage of this achieved by the decision-maker at the end of simulation is reported as Achieved Payoff Percentage. Monte-Carlo simulations with different random noise seeds for the exploration term (equation 12) and for the randomly chosen initial resource allocation give statistically reliable results for each scenario analyzed. The reported statistics are based on sets of simulations differing only in the stream of random numbers used in exploration and in the initial resource allocation.

To facilitate comparison of the four learning modules, they are run in parallel using the same noise seed for the initial allocations and the exploration terms in each. Therefore any differences in a given run are due only to the differences among the four learning procedures.

3-1- Base case

The base case represents an easy learning task where there are no delays between actions and the payoff. The decision-maker is also aware of this fact and therefore she has a perfect understanding of the correct delay structure. In this setting, we expect all the learning models to find the optimum solution. Figure 2 shows the trajectory of the payoff for each learning model, averaged over 100 simulation runs. The vertical axis shows the percentage of the optimal payoff achieved[viii] by each learning model.

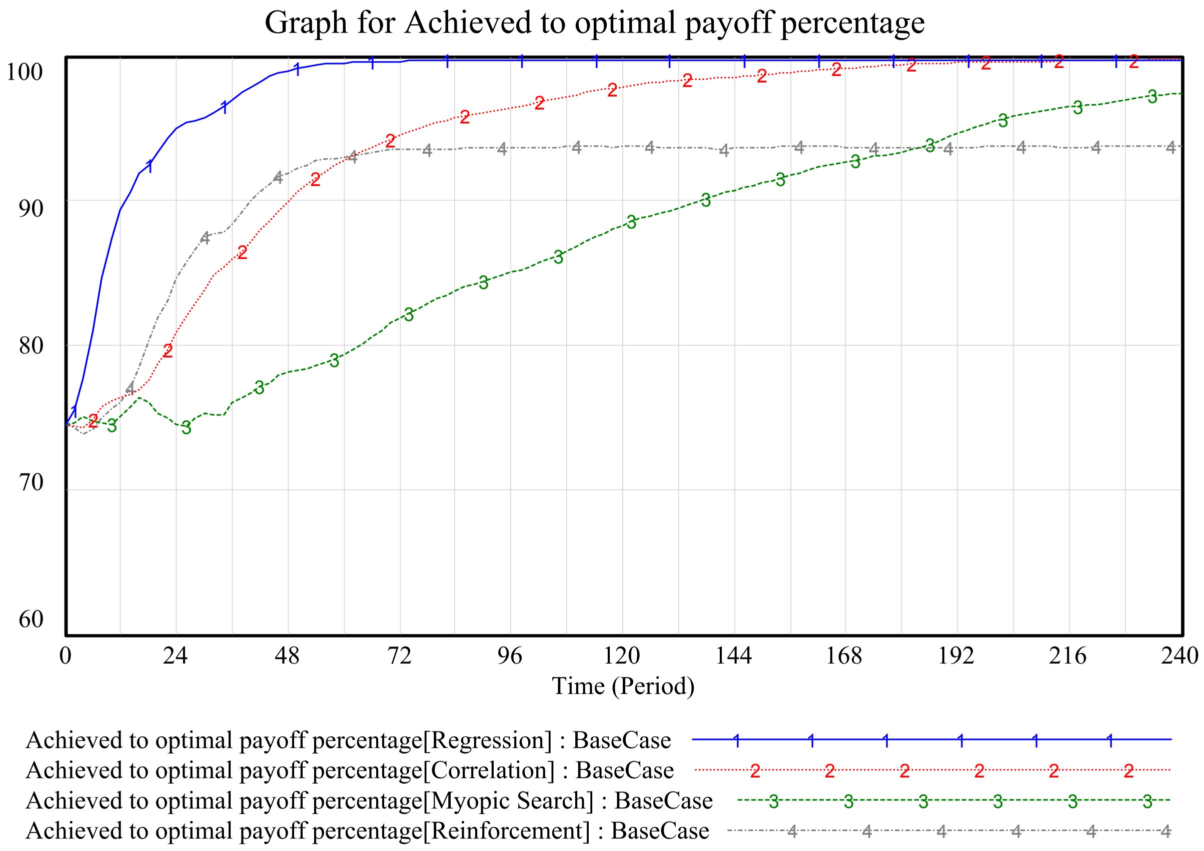

Figure 2- Payoff relative to optimal in the base case, averaged over 100 simulations

When there are no delays, all four learning models converge to the optimal resource allocation. For comparison, the average payoff achieved by a random resource allocation strategy is 74.5% of optimal (since the 100 simulations for each learning model start from randomly chosen allocations, all four models begin at an average payoff of 74.5%.)[ix]

An important aspect of learning is how fast the decision-maker converges to the allocation policy she perceives to be optimal. In some settings and with some streams of random numbers it is also possible that the decision-maker does not converge within the 240 period simulation horizon. Table 2 reports the payoffs, the fraction of simulations that have converged as well as the average convergence time[x] (for those that did) in the base case. The hypothesis H_0: μ=100% is not rejected for any of the algorithms at the 90% level and therefore one can conclude that all the algorithms find the optimal payoff.[xi] Essentially all simulations converge prior to the 240 period horizon and the average convergence times range from a low of 41 periods for the regression model to a high of 86 periods for the myopic model.

Table 2- Achieved Payoff, Convergence Time and Percentage Converged for the Base case

Learning Algorithm	**Regression**		**Correlation**		**Myopic**		**Reinforcement**	
Estimates ***Variable***	μ	σ	μ	σ	μ	σ	μ	σ
Achieved Payoff percentage at period240	99.67	1.225	99.78	0.817	97.42	4.75	93.81	7.38
Convergence Time	40.89	6.08	57.68	25.32	86.49	47.59	51.55	9.29
Percentage Converged	100		100		99		100	

3-2-The Impact of delays

Different programs of research on learning implicitly share a basic assumption that the decision-maker has a perfect understanding of any delay structures in the generation of payoffs. Under this conventional assumption, all our learning algorithms reliably find the optimum allocation and converge to it. In this section we investigate the results of relaxing this assumption.

A simple variation is to introduce a delay in the impact of one of the activities, leaving the delay for the other two other activities at zero. We simulate the model for 9 different values of the "Payoff Generation Delay" for activity one, T_1, ranging from 0 to 16 periods, while we keep the delays for other activities at 0. In our factory management example the long delay in the impact of activity 1 is analogous to process improvement activities that take a long time to bear fruit. Because activity one is the most influential in determining the payoff, these settings are expected to highlight the effect of delays more clearly.

We keep the perceived payoff generation delay for all activities, including activity one, at zero, corresponding to a decision maker who believes all activities affect the payoff immediately. Fieldwork has shown that managers may often fail to account correctly for delays between process improvement initiatives and their results (Repenning and Sterman, 2002).

Figure 3 shows the performance of four learning algorithms under these settings. Average results over 200 simulations are reported for higher confidence. Under each delay time the Average Payoff Percentage achieved by decision-maker at the end of 240 periods and the Percentage Converged [xii] are reported. In the "Average Payoff Percentage" graph, the line denoted "Pure Random" represents the case where the decision-maker selects her initial allocation policy randomly and sticks to that, without trying to learn from her experience.

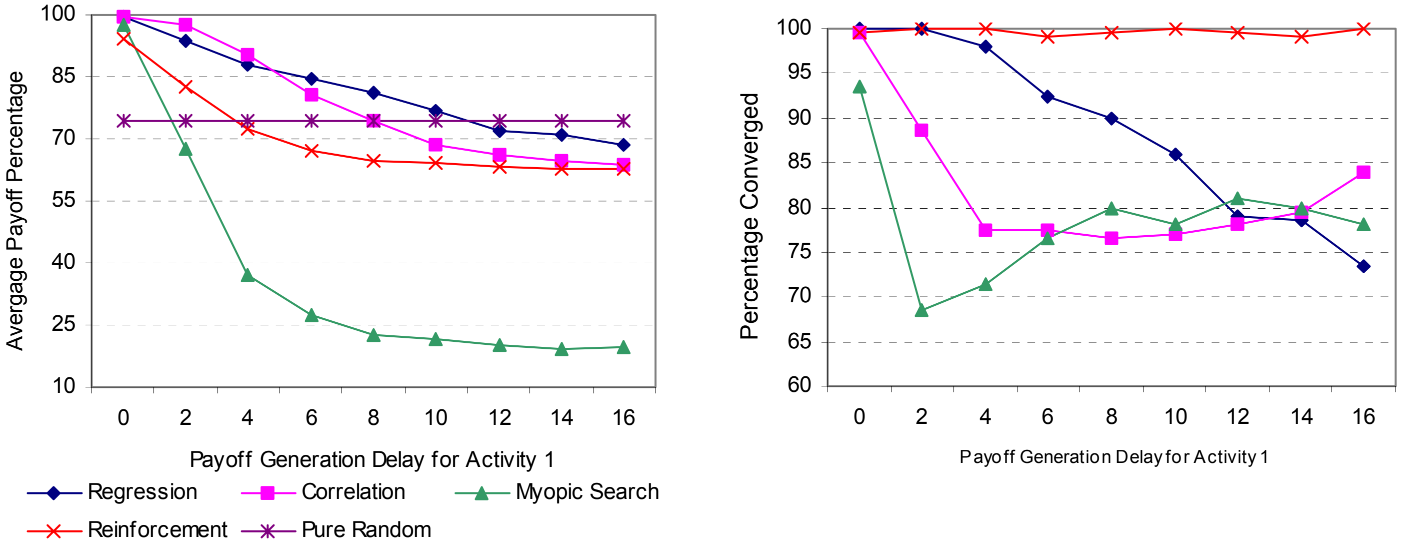

Figure 3- Average Payoff Percentage and Percentage Converged with different time delays for first activity

The following patterns can be discerned from these graphs:

- Learning is significantly hampered if the perceived delays do not match the actual delays. The pattern is consistent across the different learning procedures. As the true delay grows longer, all four learning procedures yield average performance worse than the random allocation policy (which yields 74.5% of optimal under this payoff function).
- Under biased delay perception, a significant fraction of simulation runs converge to sub-optimal policies. This means that the decision-maker usually ends up with some inferior policy concluding that this is the best payoff she can get, and stops exploring other regions of the payoff landscape, even though there is significant unrealized potential for improvement.
- Different learning algorithms show the same qualitative patterns. They also show some differences in their precise performance, where the more rational learning algorithms, regression and correlation, perform better. Nevertheless, even these higher rationality methods eventually underperform a random allocation policy as the delay grows.

In short, independent of our assumptions about the learning capabilities of the decision-maker, a common failure mode persists: when there is a mismatch between the true delay and the delay perceived by the decision-maker, learning is slower and the simulated decision maker frequently concludes that an inferior policy is the best she can achieve. Often, the decision makers reach equilibrium allocations that yield payoffs significantly lower than the performance of a completely random strategy. To explore the robustness of these results, we investigate the effect of important model parameters on the performance of each learning algorithm.

3-3-Robustness of results

Each learning model involves parameters whose values are highly uncertain. To explore the sensitivity of the results to these parameters we conducted sensitivity analysis over all the important parameters of each learning procedure.

We conducted a Monte Carlo analysis, selecting each of the parameters listed in Table 3 from a uniform distribution over the indicated range. We carried out 3000 simulations, using random initial allocations in each. Table 3 shows the parameters used in this analysis and their low, high and base values.

Table 3- Parameter settings for sensitivity analysis

Parameter	Algorithm	Low	High	Base	Description
Payoff Generation Delay[activity 1] T_1	All	0	16	0	How long on average it take for resources allocated to activity 1 to become effective and influence the payoff
Perceived Payoff Generation Delay[activity1] τ_1	All	0	16	0	The decision maker's estimate of the delay between resources allocated to activity 1 and the payoff.
Action Noise Correlation Time δ	All	1	9	3	The correlation time constant for the pink noise used to model exploration
Value Adjustment Time Constant λ	All	3	17	10	How fast the activity value system moves towards the indicated policy
Action Lookup Time Horizon	Correlation	2	10	6	The time horizon for calculating correlations between an activity and the payoff
Sensitivity of Allocations to correlations	Correlation	0.05	0.8	0.2	How strongly the allocations respond to differences among the correlations between each activity and the payoff
Reinforcement Forgetting Time	Reinforcement	3	17	10	The time constant for forgetting past activity values.
Reinforcement Power	Reinforcement	4	20	12	How strongly the differences in payoff will be reflected in action value updates.
Evaluation Period *EP*	Regression	1	9	3	How often a new regression is conducted to recalculate the optimal policy

Simple graphs and tables are not informative about these multidimensional data. We investigate the results of this sensitivity analysis using regressions with three dependent variables: Achieved Payoff Percentage, Convergence Time, and Probability of Convergence. For each of these variables and for each of the learning algorithms, we run a regression over the independent parameters in that algorithm, listed in Table 3. The regressors also include the Absolute Perception Error, which is the absolute difference between the Payoff Generation Delay and Perceived Payoff Generation Delay. Having both Absolute Perception Error and Payoff Generation Delay makes the perceived influence time almost redundant so we omit it from the independent variables. OLS is used for the Achieved Payoff Percentage and Convergence Time; logistic regression is used for the Convergence Probability. Tables 4–8 show the regression results. All the models are significant at p < 0.001.

An important trend persists in this data: increasing the absolute difference between the real delay, "Payoff Generation Delay," and the perceived delay, "Perceived Payoff Generation Delay," always decreases the achieved payoff significantly (Table 5, row 4). The coefficient for this effect is large and highly significant, indicating the persistence of the effect across different settings of parameters and different learning models. It is interesting to note that the fraction of runs that converge does not necessarily decrease with higher values of Absolute Perception Error: the regression model shows a negative relationship between perception error and convergence, but the correlation and myopic search algorithms show a positive trend in this relationship. In fact, even in the case of the regression model, close examination of the simulations with high absolute perception error indicates that a significant percentage of them do converge. The interpretation is that the decision maker not only fails to find the optimum allocations, but also concludes that a significantly sub-optimal payoff is the best that she can do and ceases exploration and search for better allocations.

Table 4- Summary Statistics for Dependent variables

Variable	Observations	Mean	Std Dev	Median
Achieved Payoff Percentage[Rgr]	3000	84.5	22.5	92.9
Achieved Payoff Percentage[Crr]	3000	76.5	23.9	85.0
Achieved Payoff Percentage[Myo]	3000	62.8	26.2	67.3
Achieved Payoff Percentage[PfR]	3000	76.6	22.0	84.0
Convergence Time[Rgr]	2836	106.2	37.0	98.9
Convergence Time[Crr]	1694	116.6	61.3	88.3
Convergence Time[Myo]	2404	112.2	51.6	96.2
Convergence Time[PfR]	2981	81.2	22.7	76.1
Convergence Probability[Rgr]	3000	0.945	0.227	1.0
Convergence Probability[Crr]	3000	0.564	0.496	1.0
Convergence Probability[Myo]	3000	0.801	0.399	1.0
Convergence Probability[Pfr]	3000	0.994	0.079	1.0

Table 5- Regression for Achieved Payoff Percentage

Variable \ Algorithm		Regression		Correlation		Myopic		Reinforcement	
1	Adj. R^2\ Model DF	0.226	5	0.315	6	0.070	4	0.028	5
2	Intercept	85.5	<.0001	88.4	<.0001	58.1	<.0001	79.8	<.0001
3	Payoff Generation Delay [a1]	0.788	<.0001	-1.65	<.0001	-0.109	0.274	-0.145	0.090
4	Absolute Perception Error	-2.63	<.0001	-1.82	<.0001	-1.24	<.0001	-0.723	<.0001
5	Log (Correlation Time)	0.346	0.244	-0.060	0.84	-0.206	0.587	-1.60	<.0001
6	Value Adjustment Time	0.583	<.0001	1.62	<.0001	1.28	<.0001		
7	Reinforcement Forgetting Time							0.291	0.003
8	Evaluation Period	0.062	0.691						
9	Correlation Slope at Origin			0.814	<.0001				
10	Action Lookup Time Horizon			-23.1	<.0001				
11	Reinforcement Power							0.189	0.026

Table 6- Regression for Convergence Time

Variable \ Algorithm		Regression		Correlation		Myopic		Reinforcement	
1	Adj. R^2\ # Data point	0.0249	2836	0.307	1693	0.077	2404	0.29	2981
2	Intercept	84.1	<.0001	95.4	<.0001	132.4	<.0001	80.3	<.0001
3	Payoff Generation Delay [a1]	1.76	<.0001	3.93	<.0001	1.99	<.0001	2.18	<.0001
4	Absolute Perception Error	4.13	<.0001	-1.16	0.0001	-0.981	0.0001	-0.481	<.0001
5	Log (Correlation Time)	-4.48	<.0001	4.12	<.0001	-0.939	0.266	-5.30	<.0001
6	Value Adjustment Time	-0.820	<.0001	-4.37	<.0001	-2.74	<.0001		
7	Reinforcement Forgetting Time							-0.319	0.0001
8	Evaluation Period	0.810	0.002						
9	Correlation Slope at Origin			-0.422	0.430				
10	Action Lookup Time Horizon			111.5	<.0001				
11	Reinforcement Power							0.014	0.857

Table 7- Logistic Regression for Convergence Probability

Variable \ Algorithm		Regression		Correlation		Myopic		Reinforcement	
1	Wald\Percent Concordant	147.4	78.4	551	78.7	225	70.6	22.2	77.7
2	Intercept	4.52	<.0001	1.23	<.0001	0.672	0.000	8.51	<.0001
3	Payoff Generation Delay [a1]	0.047	0.003	-0.054	<.0001	-0.062	<.0001	-0.242	0.001
4	Absolute Perception Error	-0.259	<.0001	0.040	0.0001	0.047	0.0001	0.026	0.662
5	Log (Correlation Time)	-0.083	0.23	-0.379	<.0001	-0.239	<.0001	-0.596	0.006
6	Value Adjustment Time	0.003	0.871	0.151	<.0001	0.166	<.0001		
7	Reinforcement Forgetting Time							0.088	0.147
8	Evaluation Period	-0.010	0.780						
9	Correlation Slope at Origin			0.066	0.000				
10	Action Lookup Time Horizon			-4.24	<.0001				
11	Reinforcement Power							-0.032	0.522

Table 8- 95% Wald confidence interval estimates for Odds Ratio

Variable \ Algorithm		Regression		Correlation		Myopic		Reinforcement	
95% Confidence Interval		Low	High	Low	High	Low	High	Low	High
1	Payoff Generation Delay [a1]	1.02	1.08	0.930	0.964	0.921	0.960	0.684	0.902
2	Absolute Perception Error	0.739	0.806	1.02	1.06	1.02	1.07	0.915	1.150
3	Log (Correlation Time)	0.632	1.12	0.391	0.524	0.515	0.714	0.117	0.704
4	Value Adjustment Time	0.963	1.05	1.14	1.19	1.15	1.21		
5	Reinforcement Forgetting Time							0.970	1.23
6	Evaluation Period	0.922	1.06						
7	Correlation Slope at Origin			1.03	1.12				
8	Action Lookup Time Horizon			0.010	0.022				
9	Reinforcement Power							0.877	1.07

The analysis shows that the introduction of delay between resource allocations and their impact causes sub-optimal performance, slower learning, and frequent convergence to sub-optimal states for all learning models over a wide range of parameters.

3-4- Analysis of behavior

The results show that learning is slow and ineffective when the decision maker underestimates the delay between an activity and its impact on performance. It may be objected that this is hardly surprising because the underlying model of the task is mis-specified (by underestimation of the delays). A truly rational decision maker would not only seek to learn about better allocations, but would also seek to test her assumptions about the true length of the delays between actions and their impact; if her initial beliefs about the delay structure were wrong, experience should reveal the problem and lead to a correctly specified model. We do not attempt to model such second-order learning here and leave the resolution of this question to further research. Nevertheless, the results suggest such sophisticated learning is likely to be difficult. First, estimating delay length and distributions is difficult and requires substantial data; in some domains such as the delay in the capital investment process it took decades for consensus about the length and distribution of the delay to emerge (see Sterman 2000, Ch. 11 and references therein). Second, the experimental research suggests people have great difficulty recognizing and accounting for delays, even when information about their length and contents is available and salient (Sterman 1989a, 1989b, 1994); here there are no direct cues to indicate the length, or even the presence, of delays. Third, the outcome feedback decision makers receive may not signal that there is a problem with the initial assumptions about the delay structure. Our results show that if the decision-maker starts with an inaccurate estimate of the delays, she may not only fail to find the optimal allocations, but will also find that her explorations around the neighborhood of the best allocation she can find, reveal no improvement. She then concludes that she has found the best possible allocation and ceases to explore. Having concluded that she has found the optimal allocation, and without any external reference point to indicate how far short of optimal she still is, there are no signals in the environment to suggest that the problem is the misperception of the delay times between activities and results.

Learning may be difficult even when the delays are correctly perceived by the decision-maker, especially in the Correlation, Myopic, and Reinforcement algorithms, where we assume the decision maker does not have a perfectly specified model of the payoff landscape. When there are no delays, the decision-maker moves from the current allocation to allocations exploration reveals to yield higher payoff. However, if she perceives some delay between activity and payoff and tries to account for those delays in her attributions, she may end up with out of date information for changing the value of different activities. Specifically, her conclusions about the gradient of the payoff landscape may be dated and point in the wrong direction. Consider the case where the Perceived Payoff Generation Delay is 3 periods and equals the true value of the Payoff Generation Delay (no misperception). In this case the decision-maker properly attributes the payoff to her decisions of 3 periods ago and correctly finds out in which direction she could have changed the allocation decision 3 periods ago to improve the payoff. However, during the intervening 3 periods she has been exploring different policies and therefore the indicated direction of change in policy may no longer indicate the best direction to move.

The discussion above suggests why a decision maker might not be able to learn the optimal policy and fail to discover the misperception of the delay structure. However, it does not explain why the results show frequent convergence to sub-optimal policies (rather than behavior in which the decision maker continuously wanders around in the payoff landscape or oscillates between different sub-optimal allocations).

To illustrate, consider again the plant manager example. Our plant manager starts his job underestimating the delays involved in process improvement activities, e.g., suppose he expects these programs to become influential after three months, while the actual time required for workers to experiment with new ideas and successfully implement them is actually about one year. To boost production he tries different policies. For example he may put pressure on workers to work harder (allocating more resources to production) or he may implement new process improvement plans (allocating resources to process improvement). Implementing a process improvement plan, he expects to observe the benefits after three months, so he attributes the increase or decrease in production, as observed after this period, to the improvement plan he had started. However, after three months, production is still lower than before the improvement plan was implemented, because worker hours have been allocated to improvement, thus cutting production effort, while the benefits of the improvement activity have not yet been realized. Observing that production has not recovered, the manager starts to revise his initial belief that improvement activity will boost performance and begins to conclude that the improvement program is not working. Having made such an attribution, if he then pushes the employees to work harder (allocating resources to producing), he will boost the payoff and learns that reducing investment in improvement is the way to boost production. His experience may then suggest that he cut maintenance and process improvement even further. Eventually, however, cuts in improvement and maintenance reduce production as equipments fail and quality drops. The manager then finds that output falls. If the reallocation of resources is slow enough, the drop in output as worker time devoted to production increases will cause the manager to perceive that there is an optimum allocation (an allocation in which change in any direction yields inferior results;) He settles on this allocation policy and stops exploring different policies. He concludes that he has found the best balance between production effort, maintenance, and improvement when in fact he is systematically under-investing in improvement and maintenance.

4- Conclusions

Two main patterns of behavior were highlighted in the analysis. First, a range of formal learning models can learn to make optimal decisions when there are no delays between resource allocations and their impact on the payoff. However, introducing a delay between the allocation of resources to an activity and the impact of that activity on performance causes significant deterioration in the ability of the models to learn; indeed, in many cases, the simulated decision-maker never finds the optimal solution and instead settles on an inferior allocation. The analysis of results sheds some light on how people can learn the wrong lessons from experience. Further, the results are robust to wide variation in the rationality and sophistication of the learning methods we tested, from a myopic method using little information and making no assumptions about the shape of the payoff landscape through a sophisticated method that uses all available data and uses perfect knowledge of the payoff landscape. Unrecognized delays between decisions and their impact can distort the outcome feedback people receive so that learning procedures that work well when there are no delays, fail.

The results also suggest it is important to investigate how people can learn about the delay structure of a system. Managers do not start their job with perfect estimates of the delays between different actions and their payoffs. Decision makers face a triple task: First, they must use the information available to them to decide what to do right now. Second, they decide how they might make better decisions in the future; such learning is conditioned on their current mental model of the decision setting. Third, they must learn about flaws in their mental model and revise it (in this case, revising their estimates of the length and distribution of any delays in the environment, but more generally, recognizing and accounting for feedback processes, stocks and flows, nonlinearities, and other elements of dynamic complexity). All these decisions and learning activities go on simultaneously. Challenging the mental models underlying our interpretation of events is difficult. It is likely to happen only when a decision-maker repeatedly fails to find any way to reconcile and interpret the information she believes to be relevant (at best; research suggests mental models are highly resistant to change, and even condition the information people perceive, so that potential anomalies that might lead to new conceptions are not even recognized. See the discussion in Sterman 2000, Ch. 1). Our analysis shows how delays can lead the decision-maker to converge on sub-optimal policies, not only learning incorrect lessons from experience, but also closing the window of opportunity for learning about the flaws in the underlying mental model.

These mechanisms have been documented empirically in the case of a manufacturing company by Repenning and Sterman (2001). They conclude:

" The most important implication of our analysis is that our experiences often teach us exactly the wrong lessons about how to maintain and improve the long-term health of the systems in which we work and live."

Our research also has methodological implications. We introduce several formal models of the learning process appropriate for continuous time settings and continuous decision variables. Despite the prevalence of such settings in the real world, the vast majority of formal models of human learning assume discrete time and/or discrete decision spaces. Our models also assume different degrees of rationality, information use, and computational power. We show how the exploration/exploitation tradeoff can be modeled, including persistence in exploratory experiments and an endogenous propensity to carry out such experiments based on the perceived potential for improvement. There are many opportunities to extend this line of modeling to generate and test explicit theories of learning in dynamic decision-making environments.

References

Brehmer, B.1992. "Dynamic Decision-Making - Human Control of Complex-Systems." Acta Psychologica **81**(3): 211-241.
Busemeyer, J. R., K. N. Swenson and A. Lazarte, .1986. "An Adaptive Approach to Resource Allocation." Organizational Behavior and Human Decision Processes **38**, 818-841.

Diehl, E. and J. Sterman.1995. "Effects of Feedback Complexity on Dynamic Decision Making." Organizational Behavior and Human Decision Processes **62**(2): 198-215.

Dörner, D. 1996. The Logic of Failure, New York: Henry Holt

Erev, I. and A. E. Roth .1998. "Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria." American Economic Review **88**(4): 848-881.

Forrester, J. W., 1969, Urban Dynamics. Waltham, MA: Pegasus Communications

Gavetti, G. and D. Levinthal .2000. "Looking Forward and Looking Backward: Cognitive and Experiential Search." Administrative Science Quarterly **45**: 113-137.

Levinthal, D. A. .1997. "Adaptation on rugged landscapes." Management Science **43**(7): 934-950.

Levinthal, D.A. and J.G. March,.1981."A Model of Adaptive Organizational Search" Journal of Economic Behavior and Organization, **2**: 307-333

Lipe, M. G., .1991."Counterfactual Reasoning as a Framework for Attribution Theories" Psychological Bulletin **109**(3): 456-571.

March, J.G., .1991."Exploration and Exploitation in Organizational Learning" Organization Science, **2**(1): 71-87.

Meadows, D. H., D.L. Meadows, J. Randers and W. Behrens, .1972. The Limits to Growth. New York: Universe Books

Paich, M. and J. D. Sterman .1993. "Boom, Bust, and Failures to Learn in Experimental Markets." Management Science **39**(12): 1439-1458.

Repenning, N.P. and J. D. Sterman. 2001. "Nobody ever gets credit for fixing problems that never happened: Creating and sustaining process improvement." California Management Review 43(4): 64-88.

Repenning, N. P. and J. D. Sterman. 2002. "Getting Quality the Old Fashion: Self-Confirming Attributions in the Dynamics of Process Improvement. Administrative Science Quarterly, Forthcoming

Sterman, J.D. .2000. Business Dynamics: Systems Thinking and Modeling for a Complex World. Chicago, IL: Irwin/McGraw Hill

Sterman, J. D. .1994. "Learning in and about complex systems." System Dynamics Review **10**(2-3): 91-330.

Sterman, J. D. (1989a). "Misperceptions of Feedback in Dynamic Decision Making." *Organizational Behavior and Human Decision Processes* **43**(3): 301-335.

Sterman, J. D. (1989b). "Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment." *Management Science* **35**(3): 321-339.

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

Appendix 1- Mathematics of the model

This appendix includes the formulation details for all four learning algorithms: Regression, Correlation, Myopic Search and Reinforcement.

Table 1- Reinforcement Learning algorithm

Reinforcement Learning Algorithm	
Variable	**Formulation**
$\frac{d}{dt}AV_j(t)$	$AR_j(t)-FR_j(t)$
$AR_j(t)$	$PPF(t)^{ReinforcemenetPower} * APF_j(t)$
$FR_j(t)$	$AV_j(t)\,/\,Reinforcement\ Forgetting\ Time$

Table 2- Myopic Search algorithm

Myopic Search Algorithm	
Variable	**Formulation**
$\frac{d}{dt}AV_j(t)$	$(AV_j^*(t)-AV_j(t))\big/\lambda$ if PPF(t) ≥ Historical Payoff(t) , else *0*
Historical Payoff(t)	*Smooth(PPF(t), Historical Averaging Time Horizon)*
$AV_j^*(t)$	$FractionofActions_j(t) * \sum AV_j(t)$ Where $FractionofActions_j(t)$ is: $\frac{APF_j(t)}{\sum APF_j(t)}$

Table 3- Correlation algorithm

Correlation algorithm	
Variable	**Formulation**
$\frac{d}{dt}AV_j(t)$	$(AV_j^*(t)-AV_j(t))\big/\lambda$
$AV_j^*(t)$	$AV_j(t) * f(Action\_PayoffCorrelation_j(t))$ Where $f(x)$ is: $2.(\frac{e^{2*x*\gamma}}{1+e^{2*x*\gamma}})$ and γ is Correlation Slope at origin

$Action\_PayoffCorrelation_j(t)$	$\frac{Action\_PayoffCo\,\text{var}\,iance_j(t)}{\sqrt{ActionVariance_j(t)} * \sqrt{PayoffVariance(t)}}$
$Action\_PayoffCo\,\text{var}\,iance_j(t)$	smooth($ChangeinAction_j(t)$ **Change in Peyoff(t),Action Lookup Time Horizon)* Where smooth function is defined by: $\frac{d}{dt}(smooth(x,\tau)) = \frac{x - smooth(x,\tau)}{\tau}$, Initial value of $smooth(x,\tau)$ is equal to initial x
$ActionVariance_j(t)$	smooth($ChangeinAction_j(t)^2$ *,Action Lookup Time Horizon)*
Payoff Variance	smooth($ChangeinPayoff_j^{\,2}(t)$ *,Action Lookup Time Horizon)*
Change in Action[j]	$APF_j(t) - smooth(APF_j(t), ActionLookupTimeHorizon)$
Change in Payoff	*PPF(t)- smooth(PPF(t),Action Lookup Time Horizon)*

Table 4- Regression algorithm

Regression Algorithm	
Variable	**Formulation**
$\frac{d}{dt}AV_j(t)$	$(AV_j^*(t) - AV_j(t))\big/\lambda$
$AV_j^*(t)$	$Max(\alpha_j^*(s)\big/\sum_J \alpha_j^*(s), 0)$, *s = EP* INT(t/EP)* where EP is the evaluation period (the decision maker runs the regression model every EP time periods).
$\alpha_j^*(s)$	The $\alpha_j^*(s)$ are the estimates of α_j in the following OLS regression model: $\log(PPF(t)) = \log(\alpha_0) + \sum_j \alpha_j * \log(APF_j(t)) + e(t)$ The regression is run every Evaluation Period (EP) periods, using all data between time zero and the current time (every time step is an observation).

[i] The model is formulated in continuous time but simulated by Euler integration with a time step of 0.125 period. Sensitivity analysis shows little sensitivity of the results to time steps < 0.2 periods.

[ii] Different types of delay, including first- and third-order Erlang delays, were examined; the results were qualitatively the same.

[iii] For a brief history of reinforcement learning idea and some well-known applications see chapter 1 of Sutton and Barto (1998).

[iv] A more realistic formulation discounts older data in favor of new data to do the regression to account for the possible changes in the environment and the payoff function. Assuming the payoff function to be constant during the learning task, this consideration makes no significant difference in our case.

[v] Because α_j^*'s are not bound to positive values, the resulting $AV_j^*(t)$'s are adjusted to a close point on the feasible action space in case of negative α_j^*. This adjustment keeps the desired action values ($AV_j^*(t)$'s) feasible (positive).

[vi] We set the mean of $PN_j(t) = 0$, so the decision-maker has no bias in searching different regions of the landscape. We also truncate the values of PN so that $PN_j(t) \geq -1$ to ensure that $OAV_j(t) \geq 0$ (Equation 12).

[vii] The simulation horizon is long enough to give the decision-maker opportunity to learn, while keeping the simulation time reasonable. In the example of the factory managers choosing among production, maintenance and improvement, the length of a period might be one month, so the 240 period horizon would represent 20 years, ample time to examine how much managers learn from their work experience.

[viii] Note that by optimum we mean the best payoff one can achieve over the long-term by pursuing any policy. However this does not need to be the highest possible payoff. For example, if there is a one period delay between activity 1 and its impact on the payoff and the two other activities are instantaneously influential (0 delay), the decision-maker can allocate all 100 units to activity 1 during the current period and allocate all the resource between two other activities during the next period. Under these conditions, she can achieve higher than optimum payoff for the next period, at the expense of getting no payoff this period (because activities 2 and 3 receive no resources) and the period after the next (because activity 1 receives no resources in that period). A constant returns to scale payoff function prevents such policies from yielding higher payoffs than the constant allocations in longterm.

[ix] Equal rather than random initial allocations gives qualitatively similar results, as shown here:

Learning Model	**Regression**		**Correlation**		**Myopic**		**Reinforcement**	
Estimate **Variable**	**μ**	**σ**	**μ**	**σ**	**μ**	**σ**	**μ**	**σ**
Achieved payoff percentage at period 240	99.69	1.03	99.93	0.033	98.77	2.13	95.80	3.95
Convergence Time	40.29	3.95	43.34	5.28	80.09	45.32	47.04	5.93
Percentage Converged	100		100		98		100	

[x] The criterion for determining when a simulation has converged follows the following logic: We consider a particular learning procedure to have converged when the variance of the payoff falls below 1% of its recent (exponential) average. If later the variance increases again, to 10% of its average at the time of convergence, we reset the convergence time and keep looking for the next instance of convergence.

[xi] In case of Myopic search and Reinforcement algorithms, a few runs ended up in low payoff values while most of the runs converged to values very close to 100%, as a result their average is around 98% and 95% while with higher variance H_0 is not rejected.

[xii] Convergence times are (negatively) correlated with the fraction of runs that converge and therefore are not graphed here.

Back to the Top