Explaining Human Behavior in Dynamic Tasks through Reinforcement Learning

Varun Dutt
Department of Social and Decision Sciences, Carnegie Mellon University, Pittsburgh, PA 15213, USA
Email: varundutt@cmu.edu

***Abstract*—Modeling human behavior in dynamic tasks can be challenging. As human beings possess a common set of cognitive processes, there should be certain robust cognitive mechanisms that capture human behavior in these dynamic tasks. This paper argues for a learning model of human behavior that uses a reinforcement learning (RL) mechanism that has been widely used in the fields of cognitive modeling, and judgement and decision making. The RL model has a generic decision-making structure that is well suited to explaining human behavior in dynamic tasks. The RL model is used to model human behavior in a popular dynamic control task called Dynamic Stock and Flows (DSF) that was used in a recent Model Comparison Challenge (MCC). The RL model's performance is compared to a winner model that won the MCC, that also uses the RL mechanism, and that is the best known model to explain human behavior in the DSF task. Results of comparison reveal that the RL model generalizes to explain human behavior better than the winner model. Furthermore, the RL model is able to generalize to human data of best and worst performers better than the winner model. Implications of this research highlight the potential of using experienced-based mechanisms like reinforcement learning to explain human behavior in dynamic tasks.**

***Index Terms*—dynamic tasks, best performer, worst performer, model comparison, model generalization, reinforcement learning**

I. Introduction

A common approach in the study of decision making involves observing human performance in a decision-making task followed by the development of a cognitive model that reproduces that behaviour and predicts new unobserved behaviour [1]. Usually, new conditions within the same task lead to the design of cognitive models with newer mechanisms, resulting in highly condition-specific models. These cognitive models might use specific mechanisms that might perform poorly to reproduce behaviour in closely related but slightly different conditions of the same task. Therefore, the practicality of this approach has been examined [1–3] and recent cognitive model designs have focused on using certain common set of cognitive mechanisms that are able to explain human behaviour in different conditions or variations of the same task [8-9].

A popular and common cognitive mechanism that is incorporated in many cognitive models is called reinforcement learning (RL), a computational approach to understanding and automating goal-directed learning and decision-making in dynamic tasks [4]. The RL mechanism is distinguished from other computational cognitive mechanisms by its emphasis on learning by an individual from direct interaction with individual's decision environment in the presence of an explicit goal and feedback, and without relying on any exemplary supervision [4]. The RL mechanism is simple to understand in the sense that a single propensity value (or experience) is updated in a task's condition as a weighted sum of the accumulated propensity in the previous trials and the outcome experienced by an agent in the last trial. Thus, a decision weight typically accounts for reliance on either an accumulated set of experiences, or on recent outcomes in a task.

In the recent past, the RL mechanism has been extremely popular and successful in explaining human behaviour in different dynamic tasks. For example, consider a dynamic repeated binary-choice task. In this task, people are asked to select between two alternatives repeatedly for many trials, each selection of one of the two alternatives affects people's earnings, and they receive immediate feedback on obtained outcomes as a consequence of their selection. Generally, in the repeated binary-choice task one of the alternatives is risky (with probabilistic rewards) and the other is safe (with deterministic rewards). A number of cognitive models like the RELACS [5], explorative sampler (with and without recency) [6], ACT-R [7], and instance-based learning [8-9] have been proposed to predict learning in different conditions of the repeated binary-choice task: with outcome feedback, with feedback about foregone payoffs, and with probability learning. All these models share the RL mechanism as a common mechanism to capture human learning and decision making in different conditions of the repeated binary-choice task.

Most recently, a model based upon the RL mechanism also won the Market Entry Competition (MEC), where the task entailed prediction of human choices in a 4-player binary-choice market-entry game [10] with

Manuscript received June 13, 2011; revised July 8, 2011; accepted July 8, 2011; *Journal of Advances in Information Technology*

doi:10.4304/jait.2.3.177-188

alternatives to enter and stay out of a risky market. Similarly, it has been demonstrated that a RL model provided useful predictions of choice rates in 12 repeated binary-choice games with unique mixed-strategy equilibrium [11]. Additional indications of the potential of cognitive models based upon the RL mechanism comes from the observed similarities of the basic reaction to feedback across a wide variety of human and animal species (e.g., [12-13]), and the discovery that the activity of certain dopamine neurons in human brain is correlated with one of the terms assumed by RL models [14].

This paper builds upon prior work and successes of the RL mechanism, and proposes a cognitive model that uses the RL mechanism to explain human behaviour in a popular dynamic control task, Dynamic Stock and Flows (DSF). The DSF task was used in the recently concluded 2009 Model Comparison Challenge competition (hereafter, MCC, see: www.cmu.edu/ddmlab/modeldsf) to compare different models of human behaviour developed in the task [15-16]. The RL model is first calibrated on certain set of calibration conditions in the DSF task and then the model is generalized to a different and novel set of generalization conditions in the same task. Model generalization is an important test for the non-specificity of a model compared to other models or benchmarks in different conditions of a task [17]. This paper demonstrates that the RL model is able to generalize to novel conditions of environment in the DSF task much better than an existing model that is based upon a popular cognitive architecture called ACT-R [3], [18-19]. The existing model that is based upon the ACT-R architecture (hereafter called the winner model) is the winner in the MCC and also uses the RL mechanism. In the MCC, the winner model outperformed 10 other competing models that used different mechanisms and approaches to model different conditions in the DSF task [15-16]. The winner model explicitly uses strategies and reinforcement learning among these strategies to make dynamic decisions (more details later in this paper).

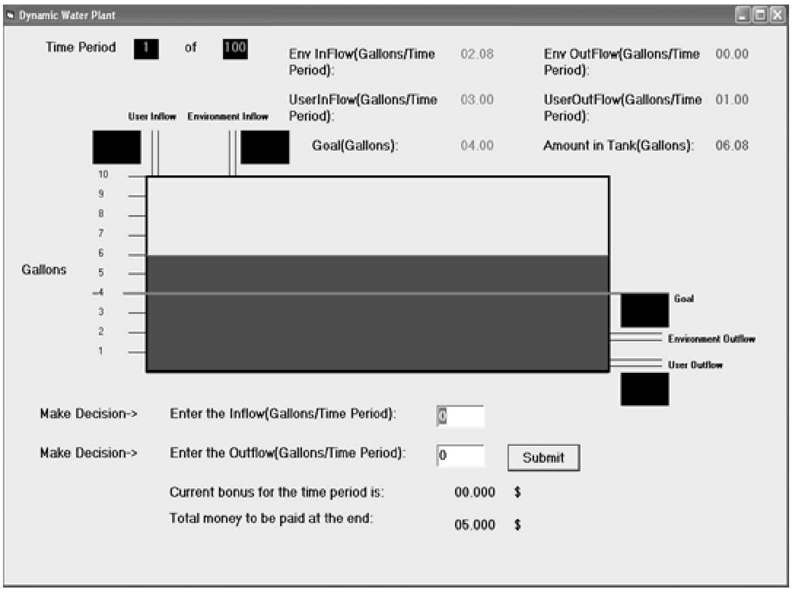
First, this paper explains the DSF task and different conditions used to calibrate and generalize models in the task. Second, the paper describes a computational RL model that is built upon the reinforcement-learning mechanism. The RL model is developed from verbal protocols provided in the calibration conditions in the DSF task, ideas from previous modelling work in the DSF task [20-21], and ideas from a popular expectancy – valence (EV) model in the Iowa Gambling Task (IGT) [22] (the EV model of IGT is also a reinforcement-learning model). Third, the paper provides details about the working of the winner model. Later, the winner model is used as a benchmark to compare the performance of the RL model. For this comparison, the paper presents results of running the calibrated winner and RL models in certain generalization conditions in the DSF task. Finally, the paper concludes by discussing the utility of experienced-based mechanisms like reinforcement learning to decision making in both the lab-based and real-world dynamic tasks.

II. Dynamic Stocks and Flows (DSF) Task

The DSF task is a generic dynamic control task that was designed to help understand human decision-making behaviour, and more concretely for this paper, to develop a RL model of human behaviour in dynamic tasks. The DSF task was also used in the MCC in which a number of models of human behaviour were developed and submitted to the competition.

The objective in the DSF task is to reach and maintain a level of water in a tank at a fixed goal level over a number of trials. The level of water in the tank is the stock or accumulation, which increases with inflow and decreases with outflow across trials. There are two types of inflows and outflows in the DSF task: those that are exogenous (outside of a participant's control) and those that are endogenous (under a participant's control). The exogenous flows are called Environment Inflow (that increases the level of the stock without a participant's control) and the Environment Outflow (that decreases the level of stock without a participant's control). The endogenous flows are the User's (or participant's) Inflow and User's Outflow. The User Inflow and Outflow are the main decisions made by participants in each trial that increase or decrease the level of stock in the DSF task.

Fig. 1 presents the graphical user interface of the DSF task. In each trial (i.e., a decision point), participants observe the past trial's values of Environment Inflow and Outflow, the values of User Inflow and Outflow, the Amount of water in the tank (stock), and a Goal level. During each trial, participants can submit two values (including zero values) for the User Inflow and User Outflow, and click upon the Submit button. Participants might also receive a "bonus" monetary incentive in each trial in which the water level was close enough to the Goal level.


Dynamic Water Plant

Time Period 1 of 100

Env InFlow(Gallons/Time Period): 02.08

Env OutFlow(Gallons/Time Period): 00.00

UserInFlow(Gallons/Time Period): 03.00

UserOutFlow(Gallons/Time Period): 01.00

Goal(Gallons): 04.00

Amount in Tank(Gallons): 06.08

Make Decision-> Enter the Inflow(Gallons/Time Period): 0

Make Decision-> Enter the Outflow(Gallons/Time Period): 0

Current bonus for the time period is: 00.000 $

Total money to be paid at the end: 05.000 $

Figure 1. The Dynamic Stock and Flows (DSF) task.

An earlier laboratory study investigated how individuals controlled the DSF task over 100 trials of practice when the environment's inflow increased (positive slope) or decreased (negative slope) as a function of the number of trials [21]. In that study, participants played DSF for 100 trials starting at an initial water level of 2 gallons with the objective of maintaining the tank's water level at the 4 gallons goal level or within +/- 0.1 gallons of the goal. In one experiment, researchers used an Environment Inflow function that was either increasing (N=15, L+): 0.08 * (trial) + 2, or decreasing (N=11, L-): (-7.92/99) * (trial - 1) + 10. Environment Outflow was constant and set at 0 gallons/trial during all 100 trials. Both the increasing and decreasing functions resulted in an equal amount of Net Environmental Flow (i.e., Environment Inflow – Environment Outflow) into the tank over the course of 100 trials (= 604 gallons). Later in a separate experiment, researchers used a non-linear environment inflow function that was again either increasing (N=18, NL+): 5*LOG (trial), or decreasing (N=17, NL-): 5*LOG (101- trial). The Environmental Outflow was again kept constant and set at 0 gallons/trial during all 100 trials. These four functions (or conditions), i.e., L and NL, and their positive (+) and negative (-) sloped analogues were used in the RL model for the purpose of calibrating the model to human data. The exact same calibration functions were also provided in the MCC to participants to calibrate their models. Calibration meant to optimize a set of parameters in a model such that the mean-squared error between model data and human data upon a dependent measure is minimized.

Furthermore, three novel generalization functions were used in the DSF task to test the ability of the calibrated RL model to generalize to and explain human data in the novel conditions. These generalization functions were called Seq2, Seq2Noise, and Seq4. In all these functions, Environment Outflow was maintained at 0 gallons/trial. In Seq2, the Environmental Inflow function repeated a sequence of 1, 5, 1, 5…for 100 trials; while in Seq2Noise, the environmental inflow function was defined as 1+/-1, 5+/-1, 1+/-1, 5+/-1...for 100 trials. Thus, the final sequence could be 0/2, 4/6, 0/2, 4/6...etc. However, this sequence was created one-time and all participants were run on the same random sequence (thus, there was no variability between participants in the sequence). The +1 or -1 noise was distributed 50/50 over trials when the sequence was created one-time. Similarly, in Seq4, the environmental inflow function repeated a sequence of 0, 4, 2, 6... for 100 trials. All three generalization functions started with 4 gallons of water in the tank and had a goal of 6 gallons with a total of 100 trials. Again, the exact same generalization functions were used in the MCC to test the submitted models.

III. The Reinforcement-Learning (RL) Model

In this section, a model that is based upon the RL mechanism is detailed along with its structure and its building process. The RL model is expected to represent cognitive processes by which participants go about keeping control of the water level (stock) in the DSF task under different and unknown functions for Environment Inflow and Environment Outflow. Here an approach is followed that is traditional in cognitive modelling, which involves a "matching process" between the model and human data, helping to fine-tune the model with human behaviour as shown in data collected from experiments [23]. The closeness of the RL model's data to human data is calculated with estimates of trend (R^2) and the deviation (Root Mean Squared Error, RMSE; [23]) over a dependent measure. These are also the estimates used to evaluate the RL model with the best known winner model in the DSF task (the parameters of the winner model were originally calibrated by its creator using the same set of estimates).

A. The RL Model's Structure

The RL model is developed using the MS Office Excel® 2010 software (the model is available upon request from the author). To develop the RL model, the author made use of the following: the averages of participants' inflow and outflow decisions; comparisons of participants' inflow and outflow decisions to stock; and, environment inflow values across the four calibration functions, L+, L-, NL+, and NL-. The author also drew upon observations from verbal protocols collected from four participants [24]. These verbal protocols were collected as part of an earlier study [21] and were available to participants that participated in the MCC. One protocol was collected from each of the four calibration functions: L+, L-, NL+, and NL-. In addition, ideas on how to divide a participant's attention in the DSF task to one of the User Inflow and User Outflow controls was derived from the expectancy – valence (EV) model. The EV model is a reinforcement-learning model that incorporates ideas on weighted attention to different attributes in a dynamic task [22].

The structure of the DSF task and the RL model is shown in Fig. 2, using common terminology from the system's literature [25]. In Fig. 2, the DSF task is represented by the Stock rectangle, the User and Environment Inflow, and the User and Environment Outflow. The Environment Inflow could be one of the different calibration or generalization functions as defined above in the DSF task, and the Environment Outflow is zero in all trials across all functions (described above). Furthermore, the Discrepancy and Environment Net Flow variables are the probable attributes in the DSF task that participants might use to decide upon values to put in the User Inflow and User Outflow controls (see Fig. 2). This fact is because these attributes were clearly visible and available to participants in the DSF task's interface (see Fig. 1).

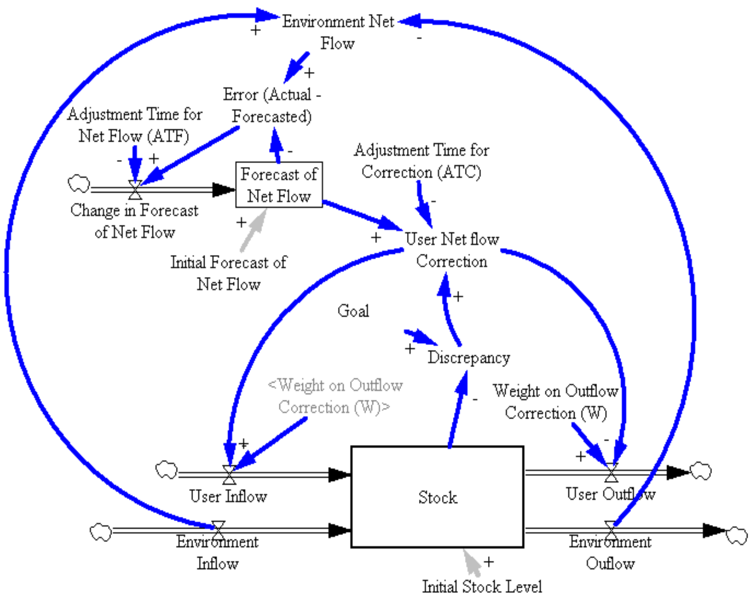

Environment Net Flow
Error (Actual - Forecasted)
Adjustment Time for Net Flow (ATF)
Forecast of Net Flow
Adjustment Time for Correction (ATC)
Change in Forecast of Net Flow
Initial Forecast of Net Flow
User Net flow Correction
Goal
Discrepancy
<Weight on Outflow Correction (W)>
Weight on Outflow Correction (W)
User Inflow
Stock
User Outflow
Environment Inflow
Environment Ouflow
Initial Stock Level

Figure 2. A system's representation of the DSF task and the RL model. The Stock, User Inflow, User Outflow, Environment Inflow, and Environment Outflow variables form a part of the DSF task. The Error, Adjustment Time for Net Flow (ATF), Change in forecast of Net Flow, Initial Forecast of Net Flow and Forecast of Net Flow constitute the parts forming the reinforcement-learning mechanism in the model (for more details refer to main text).

The verbal protocols revealed that participants clearly understood their objective to reduce the differences between the observed level of water stock and the goal level (this difference is called the Discrepancy variable in the RL model in Fig. 2). The Discrepancy, however, was only one of the many variables that participants used to make their User Inflow and User Outflow decisions. The protocols also revealed that within the first few trials, participants recognized that the unknown Environment Outflow did not change from the zero value in the experiment, and hence there was only a change in the Environment Inflow in each trial, which added water in the tank. For example, a participant in the collected verbal protocol for the NL+ function was making User Inflow and Outflow decisions for the seventh trial and had 4.9 gallons of water in the tank. She clearly indicated that "*...as the Environment has only added water to the tank in the previous trials, this time it will again and I expect the amount the Environment will add will be around 3.9 gallons.*" The participant later removed 5 gallons from the tank by using only the User Outflow decision and keeping 0 gallons/Trial in the User Inflow decision. It was clear that participants cared about the net flow into the DSF's water stock more than their User Inflow and User Outflow decisions. Participants, as part of making the User Inflow and User Outflow decisions, were also trying to forecast the net flow of the Environment into the stock in the DSF task. The forecast of net flow is an expectation that forms a part of the reinforcement-learning mechanism and is represented by the Forecast of Net Flow variable in the RL model (see Fig. 2).

Thus, according to the verbal protocols, participants' User Inflow and User Outflow decisions appeared to be determined directly by the Forecast of Net Flow and the Discrepancy variables. The verbal protocols revealed that participants attempted to correct for an observed Discrepancy over the course of a number of trials rather than instantaneously. Verbal protocols indicated that participants were amazed by the variability of the unknown Environment Net Flow (i.e., Environment Inflow – Environment Outflow) and could not correctly estimate the exact amount of water stock, the Environment was going to add in the next trial (this is reflected by the Adjustment Time for Net Flow as part of the RL mechanism in the model in Fig. 2). For example, the participant in the NL+ function (described above), decreased the water level to 3.8 gallons at the beginning of the eighth trial; she further reduced the water level to 4.0 gallons in the ninth trial, anticipating correctly this time that the Environment would add 4.2 gallons in the current trial. The same participant took 9 trials from the beginning to make the water level in the tank meet the goal level for the first time. Hence, participants corrected for the Discrepancy by balancing their Forecast of Net Flow and Discrepancy over the course of many trials.

Although for the participant in the NL+ function this course of correction took nine trials, it is clear from the collected data that it was only around the seventh trial that the participant began to make a real attempt in correcting the Discrepancy and understanding the mechanics of the DSF task. The time from the seventh trial to the ninth trial, i.e., two trials, is the time constant or the Adjustment Time for Correction (ATC) in the RL model that participants take to reduce the Discrepancy to an acceptable range around the goal level.

The Adjustment Time for Net Flow (ATF) is a parameter that accounts for a participant's memory of past Environment Net Flow values (i.e., the experience gained by participants about the Environment Net flow values in the past trials). The Forecast of Net Flow as well as the Error in Forecast (Fig. 2) can be explained by the RL mechanism with a memory parameter ATF, as described above. The RL mechanism is a time-series averaging technique that requires three pieces of information to generate a forecast for the Environment Net Flow: 1) The previous estimated forecast (F_{t-1}); 2) the last observed value of Environment Net Flow (A_{t-1}); and, 3) Adjustment Factor (Alpha or α). Alpha is a factor that indicates how much of the most recent miss from the last observed value needs to be incorporated into the next forecast to bring it closer to the last observed value. Alpha is popularly taken to be, EXPONENT (-1/ATF) (see [10] for other formulations for ATF). In the DSF task, for a trial t, the reinforcement learning is performed on the Environment Net Flow and Forecast of Net Flow values which were seen in the previous trial, t-1, respectively (as shown in Fig. 1, the previous trial's Environment Inflow and Outflow information was available on the DSF task's graphical interface).

The heart of the model consists of a balancing loop to determine the User Net flow Correction in the DSF task in each trial. In the absence of Environment Outflow, the Environment Inflow into the stock causes the stock to move away from the Goal in every trial. This causes an increase in Discrepancy. The increase in Discrepancy beyond the acceptable goal range causes a User Net Flow Correction to account for the increase in Discrepancy in a number of trials, ATC. At this point, the User Net Flow Correction is also affected by the Forecast of Net Flow (Fig. 2). The Forecast of Net Flow, due to reinforcement learning, is estimated from past experiences of the Environment Net Flow value (which is accounted by α parameter). The Environment Net Flow is directly affected by the Environments Inflows and Outflows. Part of the User Net Flow Correction determines the User Inflow, and the other part determines the User Outflow, where both parts are weighted by the User correction weight to Inflow and Outflow (W). The concept of using an attention weight or W parameter to split the User Net Flow Correction is derived from its use in the EV model for the Iowa Gambling Task (IGT) [22]. In the EV model of the IGT, the W parameter weights the attention to gains or losses observed by participants in the task. Similarly, in the RL model in this study, the impact of User Net Flow Correction in User Inflow and User Outflow is weighted by the W parameter. The human attention is a limited resource and it will be weighted in favour of one of the User Inflow or User Outflow controls in the DSF task and the W parameter will help to achieve this attention shift in the RL model. The resultant value of the User Inflow and User Outflow after accounting for the User Net Flow Correction is such that it causes the stock to decrease, and hence User Inflow and User Outflows move the stock in a direction opposite to that caused by the Environment Inflow, bringing the stock closer to the goal.

Thus, the User Net Flow Correction of trial t-1 is used to calculate the contribution to User Inflow and User Outflow weighted by the attention parameter W as

$$\text{User Inflow}_{t} = \text{User Net Flow Correction}_{t-1} * (1 - W), \quad (1)$$

$$\text{User Outflow}_{t} = \text{User Net Flow Correction}_{t-1} * (W). \quad (2)$$

Where, W is between 0 and 1. As seen above, the User Net Flow Correction at time t-1 is computed by using the Forecast of Net Flow as

$$\text{User Net Flow Correction}_{t-1} = \text{Forecast of Net Flow}_{t-1} + A_{t-1}. \quad (3)$$

The A_{t-1} is defined as

$$A_{t-1} = \text{IF (Discrepancy}_{t-1} > 0.1 \text{ OR Discrepancy}_{t-1} < -0.1) \text{ THEN } -\text{Discrepancy}_{t-1} / \text{ATC ELSE } 0. \quad (4)$$

The Discrepancy_{t-1} is defined as

$$\text{Discrepancy}_{t-1} = \text{Goal} - \text{Stock}_{t-1}. \quad (5)$$

The Forecast of Net Flow (see Fig. 2) is derived using the RL mechanism as

$$\text{Forecast of Net Flow}_{t-1} = \alpha * \text{Environment Net Flow}_{t-2} + (1- \alpha) * \text{Forecast of Net Flow}_{t-2}. \quad (6)$$

Where, α (= EXP (-1/ATF)) is between 0 and 1. The Environment Net Flow is defined based upon the Environment Inflow and Environment Outflow as

$$\text{Environment Net Flow}_{t-1} = \text{Environment Inflow}_{t-1} - \text{Environment Outflow}_{t-1}. \quad (7)$$

Once the User Inflow and User Outflow have been computed in a trial, the next trial's stock in the DSF task is computed by the basic stock equation as

$$\text{Stock}_{t} = \text{Stock}_{t-1} + \text{Environment Inflow}_{t} - \text{Environment Outflow}_{t} + \text{User Inflow}_{t} - \text{User Outflow}_{t}. \quad (8)$$

In the above equations, the $\text{Environment Net Flow}_{0} = 0$, $\text{Forecast of Net Flow}_{0} = 0$, and Stock_{0} = Initial Stock Level (which was 2 gallons for calibration functions and 4 gallons for the three generalization functions). Also, the use of *if - then – else* in (4) is provided to ensure that the corrections to the User Inflow and User Outflow are only applied when the Discrepancy from the Goal is outside the goal range of +/- 0.1 about the goal level (this was also the range assumed in the DSF task for human experiments).

Furthermore, if ATC has a large value in the RL model, then participants make very small changes to the User Net flow Correction in each trial and thus participants take a lot of time to reduce the Discrepancy to 0 in the DSF task. On the other hand, a small value of ATC means rapid changes to the User Net Flow Correction in each trial where participants are able to rapidly reduce the Discrepancy to 0 in the DSF task. Similarly, if α has a value greater than 0.5 (i.e., ATF has a value of about 1.5 trials), then participants consider a greater contribution of the previous trial's Environment Net Flow rather than their accumulated experience of Forecast of Net Flow values. In contrast, a value of α that is less than 0.5 (i.e., ATF has a value less than 1.5 trials) means that participants depend upon their accumulated experience of Forecast of Net Flow values more than the Environment Net Flow value in the previous trial. Thus, if participants use accumulated experience, then we expect the empirically determined value of α to be less than 0.5 and ATF to be less than 1.5 trials. Finally, if the value of the attention parameter W is greater than 0.5, then the User Net flow Correction increases the User Outflow more than the User Inflow and there is a net decrease in the stock in the DSF task. However, if the value of W parameter less than 0.5, then the User Net flow Correction increases the User Inflow more than the User Outflow and there is a net increase in the stock in the DSF task.

IV. The Winner Model

The winner model was one among 10 other models submitted to the MCC that was organized by Carnegie Mellon University [15-16], [18]. The winner model was the model that generalized best in the MCC where the model obtained the lowest RMSE and highest R^2 values on the generalization functions among all models submitted to the competition. In order to test the efficacy of the RL model in this paper, the winner model is used as a benchmark. The use of the winner model is simply because it is the best known model to explain human behaviour in the DSF task and thus provides an excellent benchmark to the RL model. Furthermore, the winner model, like the RL model, contains explicit strategies and reinforcement learning among these strategies for performing in different conditions in the DSF task.

Each trial in the winner model produces a prediction of the slope of Environment Inflow by using one of the two explicit strategies, *calculating* or *estimating*, in a trial. Furthermore, the model monitors the success or utility experienced after executing a strategy (success is nothing but the Discrepancy in the DSF task). It commits to memory, in each trial, an explicit ACT-R chunk encoding the type of strategy used (*estimating* or *calculating*) and the value of the success criterion (chunks are basic units of experiences stored in memory). The choice of strategy is based upon the success criterion and uses reinforcement learning. Thus, to choose a strategy in a trial, the model initiates a memory retrieval, requesting a strategy of any type, with success criterion 0.0 (i.e., one that led to a value of Discrepancy = 0, in one of the past trials). The 0.0 success criterion is the value of the retrieval attributes used to retrieve strategy chunks from memory using a similarity mechanism. The ACT-R architecture's similarity, activation, and blending mechanisms retrieve a strategy chunk that is considered most promising in a trial (these mechanisms are particular instances on reinforcement learning). Specifically, a blended chunk for each strategy is calculated in a trial in the model. Blending (a weighted averaging technique that multiplies the success criterion in a chunk with the probability of retrieval of the chunk) takes into account prior experiences and implies a bias for more recent experiences of using a strategy [26-27]. Noise in activation of a strategy chunk leads to explorative behaviour and noise is assumed to have a stronger effect for the early trials than the later trials in the winner model.

The *calculating* strategy remembers the last trial's Environment Inflow value (i.e., *t-2* trial's value) precisely while making the decision in the t^{th} trial. Then, a calculation is attempted on the basis of the latest value of Environment Inflow that is available on the DSF's interface (the value available on the DSF interface is the *t-1* trial's value). Then, the *t-1* and *t-2* values of the Environment Inflow are used to derive the slope of the Environment Inflow in the t^{th} trial. However, the implementation of the addition and subtraction procedures assumes more reliable addition than subtraction, and a more reliable subtraction A – B, if A > B [15-16], [18].

The *estimating* strategy differs from the *calculating* strategy in that the exact value of the Environment Inflow in the *t-2*th trial is not retained by the model in its working memory precisely, but stored in and retrieved from memory. The last stored Environmental Inflow value is retrieved from memory (where the noise in retrieval may lead to inexact retrievals from memory). Then, the model determines the slope of the Environment Inflow just like the *calculating* strategy, i.e., the difference between the retrieved Environment Inflow and the current Environment Inflow that is shown on the DSF's interface.

For both strategies, the determined slope of the Environment Inflow is again stored as a chunk in memory. These estimates of the slope in slope chunks are then blended once again and a blended slope chunk is finally used to determine the value of the t^{th} trial's Environment Inflow. The value of the Environment Inflow is then used to determine the value of the User Inflow and User Outflow in the t^{th} trial in the DSF task based upon whether water needs to be added or removed from an existing water stock. At the start of the model, the memory of the model is prepopulated with initial slope chunks with a value of slope determined randomly from a uniform distribution between -10 and +10 [15-16], [18]. The next section compares and contrasts between mechanisms used in both the RL and winner models.

V. Qualitative Model Comparison on Parameters and Mechanisms

Upon a comparison, the RL and winner models seem to use the same number of parameters: α (or ATF), ATC, and W, in the RL model; and, *T*, *d*, and *s* parameter in the winner model. However, in addition to these three parameters, the winner model uses the blending mechanism twice (once in the *estimating* strategy and the other time in determining the value of the slope chunk to use). The winner model also makes a number of retrievals from memory, and uses two explicit strategies: *calculating* and *estimating*. In addition, in the winner model, there exist additional hidden parameters to calibrate the production (if – then rule) execution times and times to retrieve chunks from memory (production execution is controlled by a parameter in ACT-R and time to retrieve a chunk is an inverse function of the chunk's activation and controlled by two additional parameters in ACT-R [3], [19]). Furthermore, there might also be a problem with the plausibility of human participants using specific strategies that have been assumed as part of the winner model, namely, *calculating* and *estimating*. In the past, researchers have shown that the use of a model that is based upon explicit instantiation of strategies is only justified when the modeller has prior knowledge of the use of such strategies among participants and that participants actually use such strategies in the first place [28].

Although the RL model (discussed above) could also be classified as using implicit strategies in its working, these strategies were motivated from verbal protocols and other observations in human data. As you would recall, the reinforcement-learning mechanism in the RL model uses a combination of values of the accumulated Forecast of Net Flow and the Environment Net Flow from the last trial. Thus, in the RL model, the combination of the experience of accumulated forecast of a quantity and the most recent values of the same quantity as a strategy is sufficient to explain human behaviour.

In summary, upon evaluating both these models, the winner model seems to be more complex in its processing compared to the RL model. Also, unlike the RL model, the winner model uses explicit definition of strategies in its working and instantiation that do not seem to be motivated from verbal protocols and observations in human data. In the next section, we turn towards estimating the best set of parameters in the RL and winner models.

VI. Calibration of Model Parameters

This section reports the calibration procedure used for determining the best value of parameters in the RL and winner models. For the RL model, the model parameters that need to be determined includes: α (or ATF), ATC, and W. The best value of these three parameters was determined using the four calibration functions for Environment Inflow in the DSF task: L+, L-, NL+, and NL-.

To calibrate these parameters in the RL model, a constraint-based optimization procedure was followed which could be defined as

Objective:

$$\text{Min }\{\text{Sum of average RMSE}_{\text{Discrepancy}} \text{ in 4 Environment Inflow functions L+, L-, NL+, and NL-}\}$$

Subject to,

$$0 \leq \alpha \leq 1,\ \{\text{dimensionless}\}$$
$$0 \leq \text{ATC} \leq 100,\ \{\text{trial}\}$$
$$0 \leq \text{W} \leq 1,\ \{\text{dimensionless}\}$$

Thus, the aim of the optimization procedure is to find the best values of the three parameters (above) such that it would result in the minimum value for the sum of the average RMSE over the Discrepancy between the RL model's data and human data across the four calibration functions. The average RMSE for Discrepancy is evaluated by using average Discrepancy across 100 trials in the DSF task, where the Discrepancy is averaged over all human and model participants for each of the 100 trial points. The number of model participants used was exactly the same as the number of human participants in four different calibration functions (these were reported in the DSF task section for different functions above). The lower bound value of the three constraints is defined to be 0, as these parameters cannot be negative (and a negative value will be meaningless). The upper bound value of ATC constraint is defined to be 100, as that is the maximum trial value across different functions in the DSF task. The α and W parameters are weights in the equations of the RL model and thus these parameters can only contain real value between 0 and 1. To carry out the actual optimization, a genetic algorithm program was used [29]. The genetic algorithm tries out different combinations of the three model parameters to minimize the RMSE between the model's average Discrepancy and the corresponding human's average Discrepancy. The best-fitting model parameters are the ones for which the RMSE in the objective function will be minimized. The stopping rule for the algorithm in the RL model's optimization was set at 10,000 trials of different combinations of the three parameters. This stopping rule value is extremely large and thus ensures a very high level of confidence in the optimized parameter values obtained (for more details on the genetic algorithm program refer to [9]).

The parameters of the winner model were already optimized using the four calibration functions, L+, L-, NL+, and NL-, by its creator at the time of submitting the model to the MCC [15-16], [18]. Thus, the winner model was used "as is" to compare it to the calibrated RL model in the DSF task. The next section reports the best values of the parameters from the RL and winner models.

VII. Calibration Results

The optimization of the RL model resulted in a low value of 10.55 gallons for the RMSE averaged across the four calibration functions. The individual RMSE and R^2 in different calibration functions are detailed in Table I. Table I also details the best values for the model parameters (ATF, ATC, and W). The best values of these parameters seem to have an interesting effect. The ATC value is about 3 trials in the RL model and this ATC value is much closer to the two trial value that was also found in human data of the collected verbal protocol in the NL+ function (reported above). Thus, the RL model appears to provide a close representation to the observations found in human data. Furthermore, the value of α in the RL model is about half of 0.5. Thus, the model predominantly bases its User Inflow and User Outflow decisions on the past experiences of the Environment Net Flow values rather than on the last trial's (or most recent) Environment Net Flow value. Furthermore, the value of W parameter is very high and close to 1.0. This means that the model understands the dynamics of the DSF task (a non-zero Environment Inflow and a zero Environment Outflow in different calibration functions) and like human participants in verbal protocols, it primarily uses the User Outflow than the User Inflow. Thus, the model tries to bring the stock level back to the goal by removing the water stock that is added by the Environment Inflow in each trial. The similarity between the behaviour of the model and human data highlight the fact that the RL model is a plausible account of human behaviour in the DSF task.

TABLE II.
THE RL MODEL WITH CALIBRATION RESULTS TO HUMAN DATA IN THE L+, L-, NL+, AND NL- CALIBRATION FUNCTIONS.

Function	RMSE	R^2	Best Parameter Values

L+	2.24	0.00	ATF = 0.75
			(α = 0.27)
			ATC = 3.30
			W = 0.97

L-	1.75	0.85	
NL+	31.02	0.03	
NL-	6.17	0.20	
Average	10.30	0.27	

The winner model was already calibrated by its creator on the four calibration functions using the RMSE dependent measure. The calibrated values of the *T*, *d*, and *s* parameters were reported to be 0.35, 0.50, and 0.25, respectively [15-16], [18]. The similarity was determined based upon a difference similarity function (i.e., one which computed the difference between the value of the DSF task's attributes and slots of strategy chunks in memory). The retrieval constraint used in the winner model was to retrieve chunks with a success criterion of 0.0 (i.e., Discrepancy = 0).

The parameters reported in Table I are used to test the RL model on the three generalization functions in the DSF task: Seq2, Seq2Noise, and Seq4 (described next). The results of running the calibrated RL model in the generalization functions is compared to those obtained from the winner model. The winner model was originally run by the organizers of the MCC on the generalization functions and those results were directly used in this study.

VIII. MODEL GENERALIZATION

An important test for a model's ability to explain human behaviour independent of different conditions in a task is generalization [17]. The focus in this section is to test the ability of the RL model to generalize to novel Environment Inflow and Outflow functions.

The calibrated RL model was tested for generalization on the three different generalization functions: Seq2, Seq2Noise, and Seq4. The RL model was run on the three generalization functions using a set of 20 model participants per function and the optimized values of model's parameters (that was calibrated and reported in Table 1). These are the same number of human participants that were collected in a lab-based experiment on the three generalization functions (as part of the MCC). The average Discrepancy was recorded from human and model data over 100 trials (thus, the Discrepancy was averaged over 20 model and human participants for each of the 100 trial points). Later, the RMSE and R^2 were evaluated between the 100 model and human average Discrepancy values across the 100 trials. Similarly, for the winner model, a set of 20 model participants per generalization function were run by the organizers of the MCC. The RMSE and R^2 were evaluated between the winner model and human average Discrepancy values across the 100 trials (thus, the Discrepancy was averaged in the exact same way to that for the RL model and human data).

TABLE I.
THE RL AND WINNER MODELS WITH THEIR PERFORMANCE ON THE THREE GENERALIZATION FUNCTIONS.

Model	Function	RMSE	R^2
RL	Seq2	5.88	0.03
RL	Seq2Noise	2.33	0.72
RL	Seq4	0.80	0.84
RL	Average	3.01	0.53
Winner	Seq2	3.98	0.02
Winner	Seq2Noise	12.48	0.00
Winner	Seq4	4.12	0.05
Winner	Average	6.86	0.02

Table II details the value of RMSE and R^2 (on average Discrepancy) for the RL model in comparison to human data in the three generalization functions. Also provided, are the RMSE and R^2 (on average Discrepancy) for the winner model in comparison to human data in the generalization functions.

In Table II, upon comparing the winner model with the RL model, one finds that the RL model performs better than the winner model consistently on the R^2 measure in all three generalization functions. It is only in the Seq2 function that the RMSE of the winner model is a shade better than the RL model. In fact, upon comparing the average values of the two measures across all three test functions, one finds that the RL model is much better in its explanation of human behaviour compared to the winner model (smaller RMSE and higher R^2 values). Therefore, the best model in the DSF task does not fare as well as the RL model during generalization.

Fig. 3 shows the average Discrepancy generated from the RL and winner models to the average Discrepancy observed in human data across the three generalization functions, Seq2, Seq2Noise and Seq4. As seen in Fig. 3, although the RL model is similar to the winner model in its explanation of human data in the Seq2 function, the RL model is better than the winner model in the Seq2Noise and Seq4 functions. This observation is because the winner model overestimates the average Discrepancy in human data in the Seq2Noise and Seq4 functions (the dotted model line is clearly above the bold human data line); however, such an underestimation of human average Discrepancy is absent in the RL model curves. Most probably, the reason for this consistent overestimation of human average Discrepancy in the winner model is because of the working of one of the two strategies, *calculating* and *estimating*, that the model consistently uses based upon reinforcement learning.

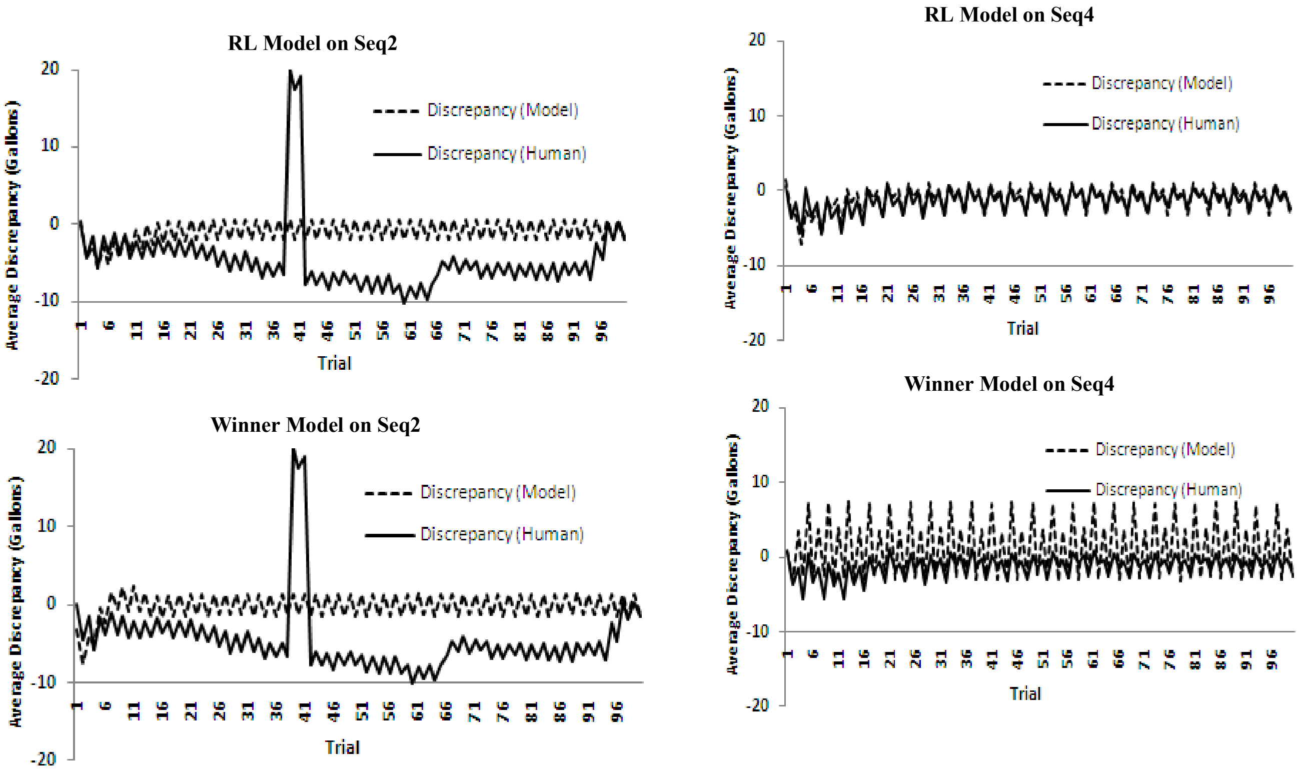
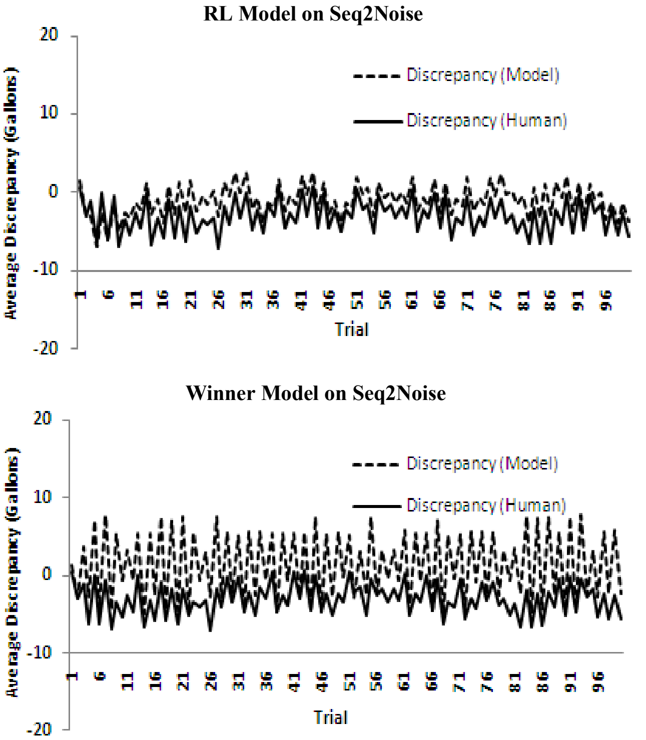


Figure 3. Average Discrepancy for the RL and winner models over different generalization functions. The Y axis represents the average Discrepancy (i.e., Goal – Stock). The X axis represents trials from 1 to 100.



VIII. Generalization to Best and Worst Performers

Another method of testing the ability of a model is to test how well the model is able to explain behaviour of best and worst performing human participants. In order to understand the generality of the RL model, it was compared to the human data for the best and worst human performers across the three generalization functions: Seq2, Seq2Noise, and Seq4. The RL model was also compared to the winner model in its ability to explain human behaviour for best and worst performers. In all these comparisons, the parameters in the RL and winner models were left at values that were obtained by calibrating these models.

For this analysis, the best human performer in a generalization function is a participant whose absolute value of the average Discrepancy across the 100 trials is the least compared to all other participants. Similarly, the worst human performer in a generalization function is a participant whose absolute value of the average Discrepancy across the 100 trials is the most compared to all other participants. Table III reports the comparison of the RL and winner models to human data of the best and worst human participants in terms of RMSE and R^2 values for the average Discrepancy across the 100 trials in the three generalization functions.

In Table III, the RL model explains human behaviour of the worst and best human performers better than the winner model. As expected, the explanation of human behaviour is not so good for the worst human performer from both the RL and winner models, but reasonably good for the best human performer (this conclusion is based upon the average RMSE and R^2 values reported in Table III).

TABLE III.
THE RL AND WINNER MODELS WITH BEST AND WORST PERFORMERS ON THE THREE GENERALIZATION FUNCTIONS.

Model	Function	Performer	RMSE	R^2
RL	Seq2	Best	1.83	0.05
RL	Seq2	Worst	144.72	0.07
RL	Seq2Noise	Best	1.77	0.36
RL	Seq2Noise	Worst	49.01	0.01
RL	Seq4	Best	2.51	0.00
RL	Seq4	Worst	6.24	0.19
RL	Average	Best	2.03	0.14
RL	Average	Worst	66.66	0.09
Winner	Seq2	Best	1.70	0.06
Winner	Seq2	Worst	145.33	0.03
Winner	Seq2Noise	Best	4.45	0.21
Winner	Seq2Noise	Worst	50.92	0.00
Winner	Seq4	Best	4.48	0.02
Winner	Seq4	Worst	8.12	0.11
Winner	Average	Best	3.54	0.09
Winner	Average	Worst	68.12	0.05

IX. Discussion

This study was aimed at testing the generality of a popular experienced-based reinforcement-learning mechanism in its ability to explain human behaviour in dynamic tasks. The study proposed a simple model of a dynamic task based upon the reinforcement-learning (RL) mechanism. The RL model was calibrated on a set of calibration functions in a popular dynamic control task called Dynamic Stock and Flows (DSF). Later, the model was generalized to a new set of generalization functions in the DSF task. The model was compared in its ability to explain human behaviour with a winner model that was developed by its creator using the RL mechanism in the ACT-R architecture and that was the best known model in the DSF task. Results revealed that the RL model generalized better than the winner model to explain human behaviour.

Reinforcement learning is a simple mechanism that depends upon a balance between past experience gained from a task's attribute and the most recent observations of the same attribute in the task [4]. Thus, it is a form of learning that is most suited to dynamic tasks where participants make repeated decisions and learn from the outcomes and feedback of these decisions [30]. Because the reinforcement-learning mechanism is suited to how humans learn in dynamic tasks by a process of trial-and-error, the mechanism seems to well in different conditions or variations of a dynamic task.

Moreover, the success of reinforcement learning in this paper is not limited to DSF task alone as the mechanism has been found to be robust in explaining human behaviour in economic choice tasks with uncertainties in the task environment. For example, recent research has highlighted the role of experience-based decisions in economic choice behaviour where the dominant mechanism to explain human choice in these economic games has been reinforcement learning [31]. In a recently concluded Technion Prediction Tournament (TPT), two large experiments were run examining different problems involving a choice between two alternatives, one uncertain and risky, and the other certain and safe. The first experiment entailed a set of calibration problems and a second experiment a set of generalization problems. Both sets of problems were drawn randomly from the same space. The problems challenged other researchers to calibrate their models on the calibration set and then predict human choice results on the generalization set. The main result from the repeated dynamic decisions part of that investigation was an indication of a clear advantage of models that explicitly assume human behaviour to be governed by experience-based mechanisms like reinforcement learning [31]. In fact, the winner of the competition was an instance-based model that made dynamic decisions based upon experience gained in the task i.e., using the Instance-based learning (IBL) theory [32]. The instance-based learning mechanism is very close to the RL mechanism as both depend upon accumulated experiences to make decisions.

Most recently, the analysis and investigation of the applicability and generality of reinforcement learning has been extended to team-based economic games where the uncertainty in the environment is both a function of the task as well as the decisions made by other participants in a team [10]. For example, it has been shown that in the MEC that used team-based market-entry games, the best models among different submissions were those that were either based upon the RL mechanism or were simple variants of the RL mechanism (e.g., the SAW and I-SAW models). In fact, the winner of the competition was a model that is based upon RL mechanism [10].

In connected research, it has also been shown that a model that was developed upon the IBL theory and one that uses past set of experiences to make choice decisions (an idea similar to RL), is also the one that generalizes well to novel task conditions in repeated binary-choice tasks. For example, researchers have shown that a single IBL model that uses past experiences to make decisions generalizes accurately to choices in a repeated-choice task, a binary-choice probability-learning task, and a repeated-choice task with a changing probability of outcomes as a function of trials [8]. Also, a variant of the same experience-based IBL model performs equally well to explain human choices in both the repeated and sampling binary-choice tasks [8].

Another important criterion on model specificity is the plausibility of different mechanisms used in models that explain human behaviour in dynamic tasks. Thus, researchers have highlighted the success of specific strategies in models that led to equally good predictions of human behaviour as the experienced-based and RL like models [28]. The important questions to pose here are the following: do humans actually use the strategies that are assumed as part of the strategy models? How well does a modeller know about the use of the assumed strategies among participants in a task? For example, the best known winner model in the DSF task used the *calculating* and *estimating* strategies to explain human behaviour in the task. But one still does not know whether humans actually make use of such strategies in their play in the DSF task. The important point to realize is that any set of sub-optimal strategies could do well to explain the sub-optimal human behaviour without one knowing which ones humans actually follow in a task [28]. In a situation, where there is little clue to the structure and nature of strategies that humans adopt, researchers have suggested the use of verbal protocols and observations in human data to motivate strategies (as assumed in the RL model).

Furthermore, the use of RL mechanisms in models has wide application to the real-world tasks that are outside the realm of lab-based settings and human experiments [4]. The RL mechanism has been applied to board games like backgammon and checkers; elevator control; robo-soccer; dynamic channel allocation; and, inventory problems to name a few [33]. A classic example is of a chess grandmaster that is about to make a move: the choice for a piece is informed by both the *experience* of planning and anticipating the counter-response and the *immediate* judgements of the intuitive desirability of particular positions. Similarly, in the animal world, a gazelle calf struggles to its feet minutes after being born. Half an hour later it is running at 30 miles per hour [4]. The RL mechanism also seems to appeal to some researchers who are interested in finding high-quality approximate solutions to large-scale stochastic-planning problems that are important for industry and government [34].

However, like with many theoretical mechanisms that exist today, the RL mechanism has also its limits and boundaries. An important consideration in RL for a modeller is the exploration – exploitation (EE) trade off [36]. In the EE trade off, a RL-agent (or a model participant) needs to explore the task environment in order to assess the task's outcome structure. After some exploration, the agent might have found a set of apparently rewarding actions. However, how can the agent be sure that the found actions were actually the best? Hence, when should an agent continue to explore or else, when should it just exploit its existing knowledge? Similarly, there could be credit-assignment problems in using the RL mechanisms in dynamic tasks where the outcome is only known after a long delay. For example, a robot in a room will normally perform many moves through its state-action space where immediate rewards are (almost) zero and where more relevant events are rather distant in the future. How does one propagate the effect of a delayed reinforcement reward/outcome to all states and actions that have had an effect on the reception of the reinforcement? Some such problems are also found in the real world: the consequences of our wait-and-see actions for climate change in the status-quo have only a delayed reinforcement in terms of adverse consequences we might face in the future [35]. Careful assumptions in reinforcement-learning models on stopping rules and newer techniques like annealing hold a great promise to help alleviate such problems [36].

Lastly, the potential for research and use of reinforcement learning mechanism is immense for the community that researches in artificial intelligence (AI) and games. Some key problems that are at the forefront of researchers in AI include the EE trade off, optimal values, constructions of the learning rate (or Alpha) parameter, and extension of reinforcement learning in Markov processes and non-stationary environments [33]. Thus, reinforcement learning provides a formal framework defining the interaction between an agent and his decision environment in terms of states, actions, and outcome/rewards. This framework is intended to be a simple way of representing essential features of different problems (captured through games) that society faces day-to-day without an explicit assumption of use of strategies [4].

ACKNOWLEDGMENT

The author is thankful to Cleotilde Gonzalez, Director, Dynamic Decision Making Laboratory, Carnegie Mellon University for her comments on an initial version of the manuscript. Furthermore, the author is grateful to the organizers of the Modelling Comparison Challenge competition for providing human data on the winner model for this manuscript. This research was supported by a 2008-2009 Steinbrenner Fellowship from Carnegie Mellon University to the author.

REFERENCES

[1] N. L. Cassimatis, P. Bello, and P. Langley, "Ability, Parsimony and Breadth in Models of Higher-Order Cognition," *Cognitive Science*, vol. 32, no. 8, pp. 1304 – 1322, December 2008.

[2] A. Newell, "You can't play 20 questions with nature and win: Projective comments on the papers of this symposium," in *Visual Information Processing*, W. G. Chase, Ed. New York: Academic Press, 1973, pp. 283–308.

[3] J. R. Anderson and C. L. Lebiere, "The Newell test for a theory of cognition," *Behavioral & Brain Science,* vol. 26, no. 5, pp. 587–637, April 2004.

[4] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Boston, MA: MIT Press, 1998.

[5] I. Erev and G. Barron, "On adaptation, maximization and reinforcement learning among cognitive strategies," *Psychol. Rev.*, vol. 112, pp. 912–931, October 2005.

[6] I. Erev, E. Ert, and E. Yechiam, "Loss aversion, diminishing sensitivity, and the effect of experience on repeated decisions." *J. Behav. Decis. Mak.*, vol. 16, pp. 215–233, December 2008.

[7] T. C. Stewart, R. West, and C. Lebiere, "Applying cognitive architectures to decision making: How cognitive

theory and the equivalence measure triumphed in the Technion Prediction Tournament," in *Proc. 31st Annu. Meeting of the Cog. Science Society*, Amersterdam, 2009, pp 561-566.

[8] T. Lejarraga, V. Dutt, and C. Gonzalez, "Instance-based learning: A general model of decisions from experience in repeated binary choice," *J. Behav. Decis. Mak.*, in press.

[9] C. Gonzalez and V. Dutt, "Instance-based learning: Integrating decisions from experience in sampling and repeated choice paradigms," *Psychol. Rev.*, in press.

[10] I. Erev, E. Ert, and A. E. Roth, "A choice prediction competition for market entry games: An introduction," *Games*, vol. 1, pp. 117-136, May 2010.

[11] I. Erev and A.E. Roth, "Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria," *Amer. Eco. Rev.*, vol. 88, pp. 848-881, September 1998.

[12] B. F. Skinner, *The behavior of Organisms.* New York, NY: Appleton-Century-Crofts, 1938.

[13] S. Shafir, T. Reich, E. Tsur, I. Erev, and A. Lotem, "Perceptual accuracy and conflicting effects of certainty on risk-taking behavior," *Nature*, vol. 453, pp. 917-920, June 2008.

[14] W. Schultz, "Predictive reward signal of dopamine neurons," *J. of Neurophys.*, vol. 80, pp. 1-27, July 1998.

[15] C. Lebiere, C. Gonzalez, and W. Warwick, "A comparative approach to understanding general intelligence: Predicting cognitive performance in an open-ended dynamic task," in *Proc. of the 2nd Artificial General Intelligence Conference (AGI-09)*, Amsterdam, 2009, pp. 103 – 107.

[16] W. Walter, V. Dutt, K. A. Gluck, and D. Reitter, "Results and lessons learned from the 2009 DSF model comparison challenge," in *Proc. of the of the 19th Conference on Behavior Representation in Modeling and Simulation*, Charleston, SC, 2010, pp. 270-271.

[17] J. R. Busemeyer and Y. M. Wang, "Model comparisons and model selections based on generalization criterion methodology," *J. of Mathemat. Psychol.*, vol. 44, pp. 171-189, March 2000.

[18] D. Reitter, "Two-level, multi-strategy memory based control," in *talk at the Dynamic Stocks and Flows Modeling Challenge Symposium (competition winner)*, 9th *International Conference on Cognitive Modeling (ICCM)*, Manchester, 2009, CD-ROM.

[19] J. R. Anderson and C. Lebiere, *The atomic components of thought*. Mahwah, NJ: Erlbaum, 1998.

[20] V. Dutt and C. Gonzalez, "Slope of inflow impacts dynamic decision making," in *Proc. of the 25th International Conference of the System Dynamics Society,* Boston, 2007, pp. 81 (and CD-ROM).

[21] C. Gonzalez and V. Dutt, "Learning to control a dynamic task: A system dynamics cognitive model of the slope effect," in *Proc. of the 8th International Conference on Cognitive Modeling*, Ann Arbor, MI, 2007, pp. 61 – 66.

[22] J. R. Busemeyer and J. C. Stout, "A contribution of cognitive decision models to clinical assessment: Decomposing performance on the Bechara gambling task," *Psychol. Assessment*, vol. 14, pp. 253-262, September 2002.

[23] C. D. Schunn, and D. Wallach, "Evaluating goodness-of-fit in comparison of models to data," in *Psychologie der Kognition: Reden and Vorträge anlässlich der Emeritierung,* W. Tack, Ed. Saarbrueken, Germany: University of Saarland Press, 2005, pp. 115-154.

[24] A. K. Ericsson, and H. A. Simon, *Protocol analysis: Verbal reports as data*. Cambridge, MA: MIT Press, 1993.

[25] J. D. Sterman, *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Cambridge, MA: Irwin/McGraw-Hill, 2000.

[26] C. Lebiere, "The dynamics of cognition: An ACT-R model of cognitive arithmetic," Ph.D. dissertation, Dept. of Computer Science, Carnegie Mellon Univ., Pittsburgh, PA, 1998.

[27] C. Lebiere, "A blending process for aggregate retrievals," in *Proc. of the 6th ACT-R Annual Workshop*, Fairfax, VA, 1999, ACT-R website.

[28] C. Gonzalez, C. V. Dutt, A. Healy, M. Young, and L. Bourne, "Comparison of instance and strategy models in ACT-R," *in Proc. of the 9th International Conference on Cognitive Modeling – ICCM2009*, Manchester, 2009, CD-ROM.

[29] Evolver, *Evolver User Reference Handbook*. Evolver Version 4.0 - Professional Edition, Ithaca: NY: Palisade Corporation, 1998.

[30] J. D. Sterman, "Learning In and About Complex Systems," *Syst. Dyn. Rev.*, vol. 10, pp. 291-330, December 1994.

[31] I. Erev, E. Ert, A. E. Roth, E. Haruvy, S. M. Herzog, R. Hau, R. Hertwig, T. Stewart, R. West, and C. Lebiere, "A choice prediction competition: Choices from experience and from description," *J. Behav. Decis. Mak.,* vol. 23, pp.15–47, January 2010.

[32] C. Gonzalez, J. F. Lerch, and C. Lebiere, "Instance-based learning in dynamic decision making," *Cogn. Scien.,* vol. 27, pp. 591-635, August 2003.

[33] M. Hutter. (2010, May 30). *Research Projects of Marcus Hutter* [Online]. Available: http://www.hutter1.net/official/projects.htm#rl(URL), 2010.

[34] A. G. Barto, "Reinforcement learning in the real world," in *Proc. of the 2004 IEEE International Joint Conference on Neural Networks*, Amherst, 2004, pp. 1661 vol. 3.

[35] V. Dutt and C. Gonzalez, "Why do we want to delay actions on climate change? Effects of probability and timing of climate consequences," *J. Behav. Decis. Mak.*, in press.

[36] S. Singh and M. Kearns, "Near-Optimal Reinforcement Learning in Polynomial Time," *Machine Learning journal*, vol. 49, pp. 209-232, November 2002.

Varun Dutt is a Post-Doctoral Fellow, Department of Social and Decision Sciences, Carnegie Mellon University. He has Ph.D. degree in Engineering and Public Policy and a Master's degree in Software Engineering from Carnegie Mellon University. Prior to coming to Carnegie Mellon, Varun worked as a software engineer in India's top IT firm, Tata Consultancy Services. He is also the Knowledge Editor of the English daily, *Financial Chronicle* and has written more than 200 articles on technology, policy, sustainability, and entrepreneurship. His current research interests focus on environmental decision making, dynamic decision making, situation awareness, cyber situation awareness, and modeling of human behavior. Varun has published in top journals of international repute which include, *Journal of Behavioral Decision Making, Journal of Applied Cognitive Psychology*, *Games*, *Computers in Human Behavior*, *Climatic Change*, and *Psychological Review*. He has also presented at more than 40 international conferences world over.

Call for Papers and Special Issues

Aims and Scope

JAIT is intended to reflect new directions of research and report latest advances. It is a platform for rapid dissemination of high quality research / application / work-in-progress articles on IT solutions for managing challenges and problems within the highlighted scope. JAIT encourages a multidisciplinary approach towards solving problems by harnessing the power of IT in the following areas:

- **Healthcare and Biomedicine** - advances in healthcare and biomedicine e.g. for fighting impending dangerous diseases - using IT to model transmission patterns and effective management of patients' records; expert systems to help diagnosis, etc.
- **Environmental Management** - climate change management, environmental impacts of events such as rapid urbanization and mass migration, air and water pollution (e.g. flow patterns of water or airborne pollutants), deforestation (e.g. processing and management of satellite imagery), depletion of natural resources, exploration of resources (e.g. using geographic information system analysis).
- **Popularization of Ubiquitous Computing** - foraging for computing / communication resources on the move (e.g. vehicular technology), smart / 'aware' environments, security and privacy in these contexts; human-centric computing; possible legal and social implications.
- **Commercial, Industrial and Governmental Applications** - how to use knowledge discovery to help improve productivity, resource management, day-to-day operations, decision support, deployment of human expertise, etc. Best practices in e-commerce, e-commerce, e-government, IT in construction/large project management, IT in agriculture (to improve crop yields and supply chain management), IT in business administration and enterprise computing, etc. with potential for cross-fertilization.
- **Social and Demographic Changes** - provide IT solutions that can help policy makers plan and manage issues such as rapid urbanization, mass internal migration (from rural to urban environments), graying populations, etc.
- **IT in Education and Entertainment** - complete end-to-end IT solutions for students of different abilities to learn better; best practices in e-learning; personalized tutoring systems. IT solutions for storage, indexing, retrieval and distribution of multimedia data for the film and music industry; virtual / augmented reality for entertainment purposes; restoration and management of old film/music archives.
- **Law and Order** - using IT to coordinate different law enforcement agencies' efforts so as to give them an edge over criminals and terrorists; effective and secure sharing of intelligence across national and international agencies; using IT to combat corrupt practices and commercial crimes such as frauds, rogue/unauthorized trading activities and accounting irregularities; traffic flow management and crowd control.

The main focus of the journal is on technical aspects (e.g. data mining, parallel computing, artificial intelligence, image processing (e.g. satellite imagery), video sequence analysis (e.g. surveillance video), predictive models, etc.), although a small element of social implications/issues could be allowed to put the technical aspects into perspective. In particular, we encourage a multidisciplinary / convergent approach based on the following broadly based branches of computer science for the application areas highlighted above:

Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the "Call for Papers" to be included on the Journal's Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal's style, together with all authors' contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. "Special Issue: Selected Best Papers of XYZ Conference".
- Sending us a formal "Letter of Intent" for the Special Issue.
- Creating a "Call for Papers" for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal's style, together with all authors' contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at http://www.academypublisher.com/jait/.