An Architectural Integration of Temporal Motivation Theory for Decision Making

Paul S. Rosenbloom (Rosenbloom@USC.Edu)
Institute for Creative Technologies & Department of Computer Science, University of Southern California
12015 Waterfront Dr., Playa Vista, CA 90094 USA

Volkan Ustun (Ustun@ICT.USC.Edu)
USC Institute for Creative Technologies, 12015 Waterfront Dr., Playa Vista, CA 90094 USA

Abstract

Temporal Motivation Theory (TMT) is incorporated into the Sigma cognitive architecture to explore the ability of this combination to yield human-like decision making. In conjunction with *Lazy Reinforcement Learning* (LRL), which provides the inputs required for this form of decision making, experiments are run on a simple reinforcement learning task, a preference reversal task, and an uncertain two-choice task.

Keywords: Motivation; cumulative prospect theory; reinforcement learning; cognitive architecture

Introduction

Temporal Motivation Theory (TMT) weaves together threads from economics, decision making, sociology, and psychology concerned with modeling human motivation and its role in decision making (Steel & König, 2006). Although other forms of motivational theories have previously been incorporated into cognitive architectures (Bach 2009; Sun & Wilson, 2010), TMT provides a particularly intriguing point of departure due to how it already integrates together so many critical aspects. Its implementation within an architecture does, however, present several challenges, including the demands it places on how probabilities, utilities, and time are represented and processed.

The work described here incorporates TMT into the Sigma cognitive architecture (Rosenbloom, Demski & Ustun, 2016), which is capable of stretching to accommodate its various demands. TMT is then deployed in a *Lazy Reinforcement Learning* (LRL) context – where the policy is computed as needed from a learned fractional representation – for use in determining the values of actions/operators considered for selection. This is not the only context in which TMT could be applied. For example, it is likely also relevant in projection, a context that may better match a number of experimental setups, but reinforcement learning was chosen as the initial target due to its centrality in procedural learning.

The resulting combination was explored in a simple RL task plus two tasks that characteristically reveal non-rational choice behavior in humans: a preference reversal task and an uncertain two-choice task.

In the remainder of this article, the relevant aspects of TMT, the TMT/LRL combination, and Sigma are first introduced, and then followed by the implementation of TMT/LRL in Sigma, experimental results and a conclusion. The core result concerns how TMT can be integrated into a cognitive architecture to yield results that, at least qualitatively at this point, enable producing some of the major phenomena that motivate this theory.

Temporal Motivation Theory (TMT)

Temporal Motivation Theory arose as a combination of four prior theories from across multiple disciplines. *Picoeconomics* models the undervaluing of future rewards via hyperbolic discounting rather than classical exponential discounting (Ainslie, 1992). *Expectancy Theory* specifies the overall worth of a value in terms of its product with its expectancy, or probability (e.g., Vroom, 1964). *Cumulative Prospect Theory (CPT)* (Tversky & Kahneman, 1992), like *Prospect Theory (PT)* (Kahneman and Tversky, 1979) before it, models an approach/avoidance aspect of behavior by nonlinearly transforming both values (Figure 1; Equation 1) and expectancies (Figure 2; Equation 2), with different weights when positive (i.e., gains) versus negative (i.e., losses). However, it also goes a step beyond PT to handle stochastic dominance and large numbers of outcomes by using cumulative distribution functions (i.e., transforming the cumulative probabilities and then taking differences of the resulting neighbors). *Need Theory* shares some aspects of the previous theories, but also concerns itself with the specific needs that yield rewards. It furthermore proposes that, like the probabilities and utilities in Prospect Theory, the weighting of temporal distances differs for positive versus negative values (e.g., Dollard & Miller, 1950).

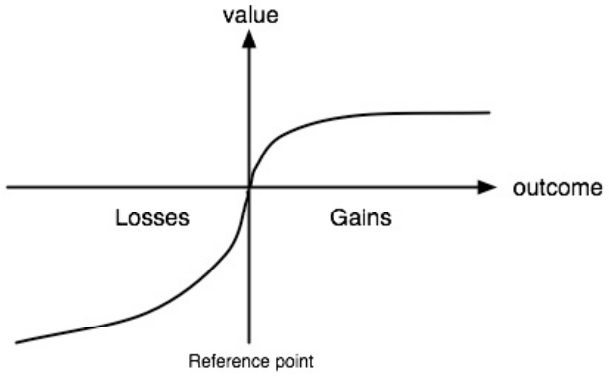

Figure 1: Shape of the (C)PT value transformation.
(from https://upload.wikimedia.org/wikipedia/commons/4/4e/Valuefun.jpg)

$$V_{CPT}^{+} = V^{\alpha}; \qquad V_{CPT}^{-} = -\lambda(-V)^{\beta} \tag{1}$$

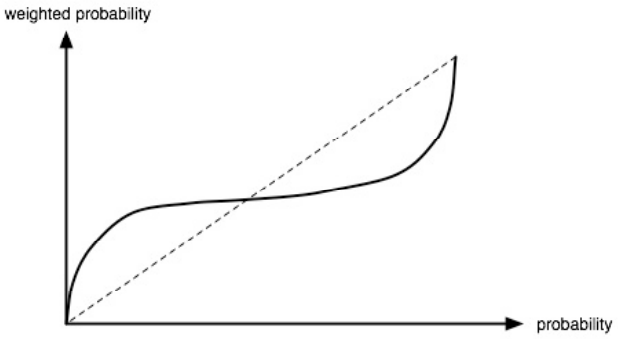

Figure 2: Shape of the (C)PT expectancy transformation.
(from https://upload.wikimedia.org/wikipedia/commons/9/90/Weightingfun.jpg)

$$E_{CPT} = \frac{E^C}{\left(E^C + (1-E)^C\right)^{1/C}} \quad \text{If } V \geq 0\ C=\gamma \text{ else } C=\delta \quad (2)$$

Putting this all together, except for the identification of specific needs and how they generate values, yields Equation 3 (Steel & König, 2006). It has two terms, for the sum over the gains and losses, respectively. The numerators blend Expectancy Theory and Cumulative Prospect Theory by multiplying the differences among nonlinearly transformed CPT expectancies (i.e., cumulative probabilities) times the nonlinearly transformed CPT values (i.e., rewards or utilities). The denominators blend Picoeconomics and Need Theory by linearly transforming temporal distances to rewards. All of the parameters other than the additive hyperbolic constant (Z) in the denominators are potentially distinct for gains versus losses.

$$Utility = \sum_{i=1}^{k} \frac{E^{+}_{CPT} \times V^{+}_{CPT}}{Z + \Gamma^{+}(T-t)} + \sum_{i=k+1}^{n} \frac{E^{-}_{CPT} \times V^{-}_{CPT}}{Z + \Gamma^{-}(T-t)} \quad (3)$$

TMT is also sometimes displayed in a simpler form, as the *procrastination equation* (Steel, 2010), as shown in Equation 4; however, it is Equation 3 that has been implemented in Sigma, along with a lazy form of reinforcement learning that provides the expectancies, values and times needed by it. Although Sigma has the ability to appraise the desirability of situations based on goal specifications (Rosenbloom, Gratch & Ustun, 2015), modeling of specific human(-like) needs has not yet been undertaken; so, instead, the results here are based on whatever rewards are appropriate for the tasks at hand.

$$Motivation = \frac{Expectancy \times Value}{Impulsiveness \times Delay} \quad (4)$$

TMT and Lazy Reinforcement Learning (LRL)

To use Equation 3 in choosing among actions, it must be applied to each action being considered. This in turn requires tracking probabilities and temporal distances for each reward received as a result of choosing each action. The policy – or *Q function* – learned via normal RL loses most of this information, just tracking a single number – corresponding to the projected discounted future utility – for each action.

An approach has been proposed for incrementally (or recursively) performing hyperbolic discounting so that time delays need not be explicitly tracked (Alexander & Brown, 2010), as with standard exponential discounting; however, it is complex enough on its own to justify putting off its consideration to future work. Still, keeping distinct the probabilities (E) and utilities (V) until they can be non-linearly transformed by CPT before being multiplied adds an extra layer of representational elaboration that is not supported by traditional Q functions.

A pair of algorithmic approaches to integrating (C)PT directly into RL can be found in Andriotti (2009) and L.A., et al. (2016), but here this additional complexity is handled by using *Lazy Reinforcement Learning (LRL)* to acquire a fractional *TMT-Q function* for states and actions, as well as a fractional *TMT-V function* for state values. LRL is like Lazy Q-Learning (Touzet, 2004) in avoiding eager combination of information for RL, but the former does learn distributions over rewards rather than using a full instance-based memory.

Both of these fractional functions explicitly represent a distribution over rewards received at each future temporal distance, while keeping separated values, expectancies and times – as necessary for TMT – and compressing the tree-structured searches into a linear structure that converts all rewards found at one temporal distance into a single distribution over those rewards (Figure 3a).

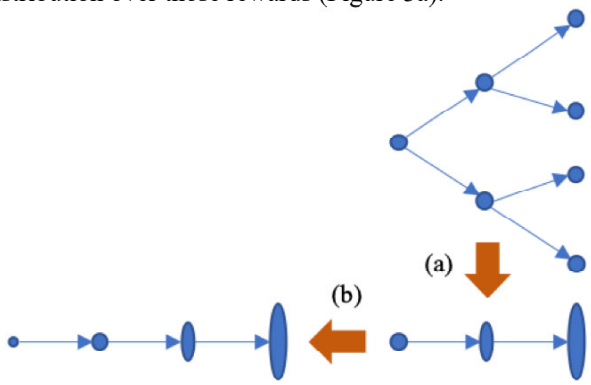

Figure 3: (a) Compressing tree of rewards into sequence of distributions over rewards; (b) Propagating sequence backward one state while shifting distributions forward.

The learning of these fractional functions occurs in an incremental manner analogous to what happens in standard RL, but the whole temporal structure is propagated backwards over action applications to be used in updating the memories for previous states (Figure 3b). In the process, the existing distributions are shifted one step forward in time, to reflect that they now correspond to states one step further in the future, and the temporal location 0 is opened up for learning about the reward at the current state.

Action selection then occurs via Equation 3, by computing Q from TMT-Q and using Boltzmann selection. In Sigma, Equation 3 is implemented in the architecture, whereas LRL involves changing parts of the knowledge traditionally used for RL. This latter works because RL in Sigma, although it leverages architectural mechanisms for things like gradient-

descent learning (Rosenbloom et al., 2013) and translation of mental images (Rosenbloom, 2011), is not itself an architectural mechanism (Rosenbloom, 2012).

Sigma and Learning TMT-V/TMT-Q

Sigma is defined in terms of two architectures, a cognitive architecture at the bottom of Newell's (1980) cognitive band and a graphical architecture at the top of his biological band. The latter started as a straightforward implementation of factor graphs with the sum-product algorithm (Kschischang, Frey & Loeliger, 2001), a general form of probabilistic graphical model (Koller & Friedman, 2009) that yields efficient computation over complex multivariate functions by decomposing them into products of simpler factors, mapping them onto graphs that have bidirectional links among variable and factor nodes, and computing over these graphs via message passing. However, in Sigma this has since been extended, e.g., by allowing unidirectional message passing to support additional cognitive structures such as rules and neural networks (Rosenbloom, Demski & Ustun, 2017).

The functions stored at factor nodes and sent along links are *regular region tensors*; i.e., *n*-dimensional structures in which each cell spans one or more values along each dimension, and every cell along any row in one dimension shares the same boundaries along the others. So, for example, TMT-V is a 3D tensor with dimensions for the state, reward, and temporal distance between the state and the reward (Figure 4); and TMT-Q is a 4D tensor with the addition of an action dimension. In both, there is a full distribution over the reward for each combination of values of the other dimensions. To slide the rewards to the right, the mental imagery operation of translation is used, while the now empty current (0) time step is initialized to uniform (Figure 5).

Reward \ Future Time (steps)	0	1	2	3-*max*
-1	.5	.1	.4	.333
0	.25	.6	.5	.333
1	.25	.3	.1	.333

Figure 4: A nominal 2D slice from TMT-V for future reward distributions at a single state. The final temporal regions at the right specify a uniform distribution over all possible but as yet unexplored future times.

Reward \ Future Time (steps)	0	1	2	3	4-*max*
-1	.333	.5	.1	.4	.333
0	.333	.25	.6	.5	.333
1	.333	.25	.3	.1	.333

Figure 5: 2D Slice from Figure 4 translated to the right, with the open column (0) initialized to uniform for learning from the perceived reward at the current state.

In Sigma's cognitive architecture, *predicates* are used to define particular types of tensors for which working memory (WM) and/or long-term memory (LTM) factor nodes are to be created. The 3D tensor underlying Figures 4 and 5 is defined, in simplified syntax, as TMT-V(*x:location*, *f:future*, *r:reward*):1. There is one argument for each dimension, with a type for each that determines whether it is continuous, discrete (i.e., integers) or symbolic, and what its span is. All of the types are discrete here, but with differing spans. Argument r is underlined because the distributions are defined over it; that is, for each region defined by the other dimensions, there is a distribution over the rewards. The :1 denotes an LTM node should be defined with an initial uniform value of 1 (before normalization). There is no WM factor node here, so there is no temporary latching of values for this tensor. Instead, gradient-descent learning at the LTM node learns reward distributions from experience. Left for future work is consideration of the potential relationship of such LTM functions to Sigma's episodic memory (Rosenbloom,), which also learns histories of values at states from experience.

To define the overall structure of the graph in which such LTM nodes exist, Sigma supports the notion of a *conditional*, which blends the conditionality found in rules and in both probabilistic and neural networks. Conditionals are built from variablized patterns plus functions, with the functions yielding an additional form of LTM node from those in predicates. The patterns are defined over predicates and may take on the unidirectional forms found in rule *conditions* and *actions* and in neural networks, or the bidirectional form found in probabilistic and constraint networks (*condacts*). Figure 6, e.g., shows in simplified form a conditional that projects backward the state reward distributions – i.e., TMT-V – while using the mental imagery operation of translation (*f*+1) to convert Figure 4's distribution to Figure 5's.

```
CONDITIONAL TMT-V-Translated
  Conditions: Selected(operator:o)
              Location(x:x)
              Location*Next(x:nx)
              TMT-V(x:nx, f:f, r:r)
  Actions: TMT-V(x:x, f:f+1, r:r)
```

Figure 6: Conditional to translate from Figure 4 to Figure 5. The italicized argument values are variables.

Learning TMT-V here occurs via gradient descent at its LTM factor node, as driven by the messages reaching it from the conditional's action. The remainder of the LRL algorithm then includes an additional conditional that yields TMT-Q from TMT-V, and which is identical to the conditional in Figure 6 except for the action, and a simpler conditional that copies the reward at the current state to location 0 of TMT-V.

Computing Q from TMT-Q

The Q function isn't directly learned in Lazy RL, although by providing an LTM function for it, it would be possible to cache an evolving representation of it. Instead, the

architecture has been extended to compute Q from TMT-Q at decision time. The algorithm for this in Figure 7 implements TMT but also allows a full space of options: (1) nonlinearly transforming rewards according to PT, CPT, or not; (2) nonlinearly transforming probabilities as in CPT, or in PT, or not; and (3) discounting future rewards hyperbolically (as in TMT), exponentially (as in classical RL), or not. It thus provides a combinatoric space for experimentation.

```
1.  P ⟵ P - min(P)          ; Remove uniform aspect
2.  F,R ⟵ Normalize(F,R)
3.  When Transform?(R) or Transform?(P):
        R ⟵ Shatter(R)  ; Partition into unit regions
4.  When Cumulate?(P):
        P ⟵ Cumulate(P)
5.  When Transform?(P):
        P ⟵ Transform(P)
6.  When Cumulate?(P):
        P ⟵ P – P_prev      ; Uncumulate P
7.  R ⟵ Expected(P, [If Transform?(R) then
                        Transform(R) else R])
8.  When Discounted = H
        F ⟵ Hyperbolic(F)
9.  When Discounted = E
        F ⟵ Exponential(F)
10. ∑F
11. P ⟵ Scale-Down(P)  ; Scale down extremes
12. P ⟵ Exponentiate(P) ; Ensure values positive
13. Q ⟵ Remove-Unneeded-Slices
```

Figure 7: Algorithm to transform TMT-Q into Q. Rewards (R) may be transformed or not. Probabilities (P) may be cumulated or not and transformed or not. Futures (F) may be discounted hyperbolically (H), exponentially (E) or not.

Several aspects of this algorithm could do with a bit more explanation. In line 1 the removal of the uniform signal, by subtracting from each distribution the minimum probability in it, is not a standard part of TMT or CPT, but is necessary to appropriately use Sigma's learned distributions in transformations that are asymmetric around zero. Without this step, and with the standard loss-averse CPT parameters, decision making would be biased away from actions for which little has been learned, and which thus retain more of their initial uniform distribution. Ultimately, such a bias may prove to be appropriate as an implicit preference for actions about which the system is more certain, but it raises issues for Boltzmann selection in providing a strong bias for exploitation over exploration, so for now this bias is subtracted out, with further investigation of its possibly appropriate use left for future consideration.

The shattering along the reward dimension in line 3 and the later removal of unnecessary slices in line 13 both relate to how Sigma's regular region tensors work. The shattering ensures that there is a separate region for each distinct reward value – even when they have the same probabilities – enabling computations for cumulative probabilities to happen appropriately. The removal of unneeded slices at the end then reaggregates regions when all adjacent pairs of them along any dimension have the same probabilities.

The core of (C)PT is transforming both probabilities (line 5) and rewards (which occurs in line 7, in the process of determining the expected value for each future region). With full CPT, the probabilities need to be cumulated before they are transformed (line 4) and then uncumulated after (line 6). This is then followed, if desired, by discounting, either hyperbolically (line 8) or exponentially (line 9); and summing the expectations over future times (line 10).

The key remaining steps are required by Sigma rather than TMT or CPT. The core is the need to exponentiate the results (line 12) so that Sigma's decision procedure receives a fully non-negative Q distribution. In support of this, distributions with extreme values that would lead to either underflow or overflow when exponentiated are scaled down (line 11).

The elements added in this algorithm to fit TMT into Sigma may affect the exact numerical values yielded, in comparison to pure TMT, but the intent, as evaluated in the next section, is that the same qualitative phenomena will result.

Experiments

Tversky and Kahneman (1992) provided a standard set of parameters for use in CPT: $\alpha=\beta=.88$; $\gamma=.61$; $\delta=.69$; and $\lambda=2.25$. These values are used throughout these experiments, without attempting to tune them further to any differences resulting either from the implementation in Sigma or the specific tasks or models. We are not aware of standard values for the other TMT parameters in Equation 3, so the following values were used: $\Gamma^{+}=\Gamma^{-}=1$; and $Z=.5$. Although Z is often shown as 1, it was found to be easier to display preference reversal on a small grid with a smaller value. The one remaining parameter is a value of .95 for exponential discounting, when used.

RL Task

The first experiment used a simple RL task – of finding a goal location in a 1D corridor (Figure 8) – to determine whether RL would still work when TMT is injected into the decision-making process; and, further, to see if there might be any interesting differences among 12 of the variations enabled by the algorithm in Figure 7, omitting only those with no discounting, as they would not learn anything of interest here.

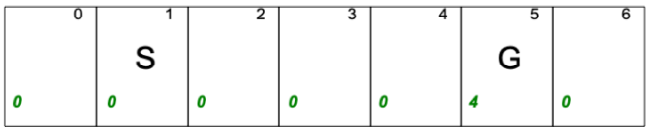

Figure 8: Simple RL Task Domain with starting and goal locations, and a reward of 4 at the goal location.

Each variation was run for 5 repetitions of 200 trials each. The average number of decisions per repetition ranged from 4428 (with either form of expectancy/probability transformation, no value transformation, and exponential discounting) up to 5510 (with no expectancy/probability

transformation, a value transformation, and hyperbolic discounting). However, all of the variations did learn, with even the slowest yielding a Q function favoring movement to the right over movement to the left at locations 1-4. The faster cases simply had greater differences between the values for moving right versus moving left.

To explore whether there were any interesting differences among the 12 variations, a three-factor ANOVA was run over the number of decisions per repetition for the three dimensions/factors that define the space of variations. This yielded four results with $p<.05$, for the three single factors plus the interaction between expectancy/probability transformations and discounting strategy.

There are three levels of expectancy/probability transformation, with CPT and PT having almost identical means, of 4911 and 4909, but with the no transformation case jumping up to 5410. There are two levels for value transformation, with transformation (5130) being slightly outperformed by no transformation (5023). There are also two levels of discounting, with hyperbolic discounting (5373) being outperformed by exponential discounting (4780). The one significant pairwise interaction is due to exponential discounting significantly outperforming hyperbolic discounting with expectancy/probability transformations, but performing similarly with no transformation.

Given the rationality, and thus the a priori expectation of optimality, of standard RL – that is, exponential discounting with no transformations – the one real surprise here is the gain over this from expectancy/probability transforms. Although these results are very preliminary, they do suggest further investigation may be worthwhile from an RL perspective.

Preference Reversal

The second experiment concerned a temporal form of *preference reversal* (Ainsley, 1992), an example of humans engaging in non-rational decision making, where shifting a pair of possible rewards further away in time, even when their relative distances to the starting point remain the same, reverses which reward is preferred. A typical example compares two decision situations in which the subject can either receive \$10 on one day versus \$100 one year from that day. If the decision occurs on that first day, then the immediate gratification of \$10 may be preferred to \$100 in a year. However, if the decision occurs 10 years prior to the first day, the \$100 will be preferred. With exponential discounting, a year's worth of discounting has the same proportional impact whether or not that year is now or 10 years in the future. However, with hyperbolic discounting the effect is smaller the further into the future the rewards are.

An analog of this task was developed in a slightly wider version of the 1D corridor in Figure 8, with the start location at 5, and rewards of 1 and 3 either at "near" locations (4 and 7) or "far" locations (2 and 9). For 1 repetition of 200 trials, every variant with exponential discounting failed as expected to exhibit preference reversal, with all showing a strong preference for the larger reward whether near or far. For example, for standard RL – although still in the form of LRL – with no transformations, the left versus right Q values at location 5 were <.14, .86> for near and <.16, .84> for far. For hyperbolic discounting, all cases instead showed preference reversal, except for those with no value transformation. For full TMT, for example, the left versus right Q values at location 5 were <.62, .38> for near and <.41, .59> for far. In the exceptional cases – i.e., with no value transformation – the constant preference for the larger value is likely due to there being a wider difference between their untransformed values versus their transformed values, and thus requiring more distant "far" locations to show preference reversal. In a follow-on experiment with rewards of 2 and 4 and locations of 1 and 10, this case did in fact exhibit preference reversal.

Two-Choice Task

The third experiment compares a choice between a single fixed reward versus a gamble between two rewards with known probabilities. It stresses the non-rational consequences of transforming expectations/probabilities and values/utilities, and thus is particularly relevant to (C)PT.

A narrower adaptation of the 1D corridor in Figure 8 is used here, with a start location of 1, a move left leading to the gamble at location 0, and a move right leading to the fixed reward at location 2. As both payoffs are at the same distance, the form of temporal discounting becomes irrelevant here. However, instead of using the Q function for decision making, the probability of selecting the gamble is set to .9, with only a .1 chance of selecting the fixed reward, so that, even when the gamble is very skewed, such as .99 versus .01, there will be enough experience with the rare alternative.

Figure 9 shows the results for those cases from Table 3 of Tversky & Kahneman (1992), where the uncertain choice has one 0 reward and one positive one (either 50, 100, 200 or 400). The Human data is taken directly from that table for these fifteen data points, whereas the Sigma data is from the simulation, and the TMT data is directly from the equations. It can be seen from this figure that the shapes of the three curves are roughly the same, although both the Sigma and TMT curves are somewhat lower than the Human curve throughout much of the midrange of the probabilities.

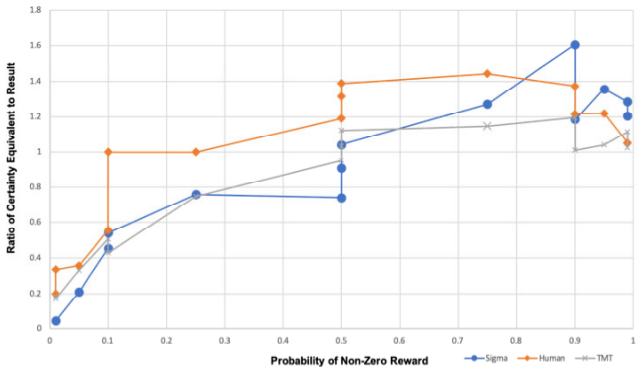

Figure 9: Comparison of Sigma (simulated), Human and TMT (analytical) data for uncertain alternatives of 0 and a non-zero (positive) reward.

Conclusion

In this article, one key aspect of motivation – how it impacts decision making in human(-like) intelligence – has been approached architecturally, by adding Temporal Motivation Theory (TMT) to Sigma's decision procedure. A lazy form of reinforcement learning (RL) that is implemented by modifying Sigma's standard knowledge-driven (plus gradient-descent learning) approach to RL provides the values, expectancies and times required by TMT.

Experiments explored whether this combination could still learn appropriately in a simple RL task, and whether it yields human-like results in preference reversal and two-choice tasks. The answer to these questions is yes, although only qualitatively at this point. The one big surprise was that in the simple RL task, adding in the expectancy/probability transformation improved the learning. Although this is a very preliminary result, it is worth looking into further.

In addition to what has already been mentioned with respect to relevant future work, there are a number of other issues worth further follow up. One is exploring more complex RL tasks that require the learning of longer sequences of future time steps, and thus considering whether Lazy Reinforcement Learning (LRL) continues to be sufficiently efficient. A second is the use of TMT in projection – i.e., lookahead search – as an alternative to its use in reinforcement learning. A third is looking more deeply at a broader range of human tasks, and specifically at whether with appropriate parameter searches good quantitative fits can be produced. A fourth and final topic is incorporating architectural models of specific motivations/needs that can appropriately and automatically provide many of the rewards/values required for human-like decision making.

Acknowledgments

The work described in this article was sponsored by the U.S. Army. Statements and opinions expressed may not reflect the position or policy of the United States Government, and no official endorsement should be inferred.

References

Ainslie, G. (1992). *Picoeconomics: The Strategic Interaction of Successive Motivational States within the Person.* New York: Cambridge University Press.

Alexander, W. H. & Brown, J. W. (2010). Hyperbolically discounted temporal difference learning. *Neural Computation*, *22*, 1511-1527.

Andriotti, G. K. (2009). *Prospect Theory Multi-Agent Based Simulations for Non-Rational Route Choice Decision Making Modelling*. PhD Thesis, University of Würzburg.

Bach, J. (2009). *Principles of Synthetic Intelligence, Psi: An Architecture of Motivated Cognition.* New York, NY: Oxford University Press.

Dollard, J., & Miller, N. E. (1950). *Personality and Psychotherapy: An Analysis in Terms of Learning, Thinking, and Culture.* New York, NY: McGraw-Hill.

Kahneman, D. & Tversky, A. (1979). Prospect Theory: An Analysis of Decision under Risk (PDF). *Econometrica*, *47*, 263–291.

Koller, D. & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press.

Kschischang, F. R., Frey, B. J. & Loeliger, H.-A. (2001). Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*. *47*, 498-519.

L.A., P., Jie, C., Fu, M., Marcus, S. & Csaba, S. (2016). Cumulative Prospect Theory meets Reinforcement Learning: Prediction and Control. In *Proceedings of Machine Learning Research*, *48*, 1406-1415.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Rosenbloom, P. S. (2011). Mental imagery in a graphical cognitive architecture. *Proceedings of the 2nd International Conference on Biologically Inspired Cognitive Architectures* (pp. 314-323).

Rosenbloom, P. S. (2012). Deconstructing reinforcement learning in Sigma. *Proceedings of the 5th Conference on Artificial General Intelligence* (pp. 262-271).

Rosenbloom, P. S., Demski, A., Han, T., & Ustun, V. (2013). Learning via gradient descent in Sigma. *Proceedings of the 12th International Conference on Cognitive Modeling* (pp. 35-40).

Rosenbloom, P. S., Demski, A. & Ustun, V. (2016). The Sigma cognitive architecture and system: Towards functionally elegant grand unification. *Journal of Artificial General Intelligence*, *7*, 1-103.

Rosenbloom, P. S., Demski, A. & Ustun, V. (2017). Toward a neural-symbolic Sigma: Introducing neural network learning. *Proc. of the 15th Annual Meeting of the International Conference on Cognitive Modeling*.

Rosenbloom, P. S., Gratch, J. & Ustun, V. (2015). Towards emotion in Sigma: From Appraisal to Attention. *Proceedings of the 8th Conference on Artificial General Intelligence* (pp. 142-151).

Steel, P. (2010). *The Procrastination Equation: How to Stop Putting Things Off and Start Getting Stuff Done*. Toronto, Ontario: Random House Canada.

Steel, P. & König, C. J. (2006). Integrating Theories of Motivation. *Academy of Management Review*, *31*, 889-913.

Sun, R. & Wilson, N. (2010). Motivational processes within the perception-action cycle. In V. Cutsuridis, A. Hussain & J. G. Taylor (Eds.), *Perception-Action Cycle: Models, Architectures, and Hardware*. New York, NY: Springer.

Touzet, C. F. (2004). Distributed lazy Q-learning for cooperative mobile robots. *International Journal of Advanced Robotic Systems*, *1*, 5-13.

Tversky, A. & Kahneman, D. (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, *5*, 297–323.

Vroom, V. H. (1964). *Work and Motivation.* New York, NY: Wiley.