Know Your Enemy: Applying Cognitive Modeling in Security Domain

Vladislav D. Veksler (vdv718@gmail.com)
DCS Corp, U.S. Army Research Laboratory
Aberdeen Proving Ground, MD, USA

Norbou Buchler (norbou.buchler.civ@mail.mil)
U.S. Army Research Laboratory
Aberdeen Proving Ground, MD, USA

Abstract

Game Theory -based decision aids have been successfully employed in real-world policing, anti-terrorism, and wildlife conservation efforts (Tambe, Jiang, An, & Jain, 2013). Cognitive modeling, in concert with *model tracing* and *dynamic parameter fitting* techniques, may be used to improve the performance of such decision aids by predicting individual attacker behavior in repeated security games. We present three simulations, showing that (1) cognitive modeling can aid in greatly improving decision-aid performance in the security domain; and (2) despite the fact that individual attackers will differ in initial preferences and in how they learn, model parameters can be adjusted dynamically to make useful predictions for each attacker.

Keywords: cognitive modeling; game theory; behavioral game theory; strategy selection; agent simulation; model tracing

Introduction

Game Theory (GT) focuses on mathematical models of rational decision-making. In recent years, GT-based decision-aiding software has received significant attention for success in real-world security domain problems, such as scheduling patrols conducted by the US Coast Guard at multiple major US ports (Shieh et al., 2012), scheduling police patrols at major airports such as LAX (Pita et al., 2008), allocating federal air marshals on flights of US Air Carriers and several other applications (Tambe, 2011; Tambe et al., 2013). Success of GT approaches can be further improved by dropping the assumption that humans are optimally rational decision-makers, and by using cognitive modeling to predict adversary strategy selection.

Humans are not perfectly rational, rather, we are boundedly rational (Simon, 1972). Our ability to make optimal decisions is limited by available information, available time, and a myriad of cognitive constraints. There is a body of literature describing biases in human decisions (e.g., Kahneman, 2011), cultural preferences (e.g., Sample, 2015), and cognitive process interactions (e.g., Anderson, 2007) that can aid in predicting attacker decisions in real-world security problems.

Behavioral game theory is a modification of rational game theory informed by, "experimental evidence and psychological intuition" (Camerer, 2003, p. 465). Ultimately, the goal of behavioral game theory is to predict behavior and inform decisions in real-world strategic situations (Gächter, 2008). Whereas the success of normative game theory in security domain comes from providing efficient randomization of security plans and processes, behavioral game theory provides a more realistic view of human strategy selection based on a large body of empirical evidence, and argues for use of behavioral/cognitive models to predict human behavior.

Cognitive modeling is a method for predicting behavior based on known cognitive processes and biases. Computational cognitive models take the form of software that simulates human decisions on a given task. Computational cognitive models have been employed to account for game play in Prisoner's dilemma (Lebiere, Wallach, & West, 2000), rock-paper-scissors (West, Lebiere, & Bothell, 2006), and a collaborative foraging Geo Game (Reitter & Lebiere, 2011).

In this paper, computational cognitive models are employed to predict human behavior in security games. The rest of this paper describes a normative game theory approach to decision-making in the security domain, and suggests an alternative approach that employs cognitive modeling for selecting the best strategy in response to individual attacker's evolving preferences. We present three simulations highlighting the advantages of using cognitive modeling over normative game theory, and examine the use of model tracing and dynamic parameter fitting for predicting individual attacker's strategy selection.

Game Theory Approach to Decision Aids

Tambe et al. (2013) describe several successful applications of game theory decision aid software in real-world security games. Airport security, coast guard, and police officers employ this software to patrol for criminal activity. Animal preservation patrols are aided with this software in their efforts to control poaching. Tambe et al. (2013) focus on Stackelberg security games where the defender must perpetually defend a set of targets with limited resources, and the attacker can choose to attack a given target after observing defender actions. The general idea of picking an optimal mix of actions (i.e. mixed strategy) for the defender so as to decrease the chances of a successful attack applies across a much wider context (e.g., sports, cyber security, anti-terrorism).

Game theory suggests that the defender's mixed strategy should be a distribution of actions that removes any incentive for the opponent to choose one action over another. For example, imagine a simple game where there are only two possible actions for the defender, $D1$ and $D2$, and two possible actions for the attacker, $A1$ and $A2$. Let us assume that attacker payoffs are probabilities of a successful attack, these probabilities/payoffs being as listed in Table 1 (when the de-

fender chooses $D1$, the probabilities of attacker success for actions $A1$ and $A2$ are .2 and .6, respectively; when the defender chooses $D2$ these probabilities are .5 and .3, respectively). In this scenario, if the defender always chose $D1$, a rational attacker would always play $A2$, winning 60% of the time. If the defender always chose $D2$, the attacker would always play $A1$, winning 50% of the time. If the defender played randomly, the attacker would have the incentive to always play $A2$, winning 45% of the time.[1] A GT-computed mixed strategy in this game would be for the defender to play $D1$ one third of the time, and $D2$ two thirds of the time. This would leave the attacker with no preference for either option: playing either $A1$ or $A2$ would only lead to a successful attack 40% of the time.

Table 1
Sample payoffs for the attacker in a security game, where there are only two possible actions for the defender, D1 and D2, and two possible actions for the attacker, A1 and A2.

	A1	A2
D1	.2	.6
D2	.5	.3

In real world security games there are many more actions, and payoffs must take into account much more than success/failure. For example, Kar, Fang, Fave, Sintov, and Tambe (2015) describe a scenario where the attacker, an animal poacher, is drawn not only by the success of a poaching effort, but also by animal density and travel time. That is, when humans play the security game as an attacker (i.e. animal poacher) they are more likely to choose an action that leads to less travel time and higher animal density, even when the risk of failure (capture) is high.

Although animal density and travel distance are characteristics of the task environment, these factors are also latent indicators of cognitive biases. Direct consideration of such cognitive biases and limitations in formal attacker models should improve the performance of decision aids in the security domain.

Cognitive Modeling Approach

There are many computational models that provide robust predictions of human behavior. For example, models of reinforcement learning provide robust accounts of human trial-and-error behavior and its neural correlates (e.g., Anderson, 2007; Fu & Anderson, 2006; Holroyd & Coles, 2002; Nason & Laird, 2005); models of declarative memory provide robust predictions of fact recall latency and probability (e.g., Anderson, 2007; Anderson & Reder, 1999; Mackintosh, 1983; Shanks, 1994); and skill acquisition models provide robust predictions of how people achieve experise (e.g., Chase & Simon, 1973; Gobet, 1998; Gobet et al., 2001). Furthermore, multiple individual process models can be integrated together to generate more complete and general predictions of behavior across many contexts (Anderson, 2007; Choi & Ohlsson, 2011; Gray, 2007; Veksler, Myers, & Gluck, 2014).

Cognitive models are typically held to account for behavior in the aggregate (average group behavior) rather than for individual differences. However, even in the absence of precise individual predicitons, general behavioral tendencies can be helpful in predicting likely behavior. This is not different from predicting large-scale events in physical sciences: fundamental principles of physics may not help us to predict exactly where a tsunami will hit, but it is useful to know that some locations are more probable than others.

Additionally, in repeated security games[2] cognitive models can be dynamically updated to provide better predictions of individual attacker's cognition and behavior by use of *model tracing* and *dynamic parameter fitting*. The model tracing technique comprises force-feeding a participant's experiences to the cognitive model. That is, if the participant and the model were to choose different strategies, model actions would be overwritten with participant actions in the model's memory. This method was employed in computerized instructional aids, "cognitive tutors", for students learning high school math in Pittsburgh (Anderson, Corbett, Koedinger, & Pelletier, 1995).Dynamic parameter fitting is used to adjust model parameters based on known data points, so as to make better individual predictions for future behavior. This method was employed to predict performance of individual F-16 pilot teams (Jastrzembski, Gluck, & Rodgers, 2009) and is employed in software that predicts optimal training schedules based on individual performance histories (Jastrzembski, Rodgers, Gluck, & Krusmark, 2014). The following simulations examine the use of these techniques for defender agent software in the security domain.

Simulation 1: Model Tracing

A cognitive model can predict general tendencies, but it is unlikely that a model will predict all decisions of a given individual, even on fairly simple tasks. Model predictions for each of the attacker's decisions contain an element of uncertainty, X. This simulation explores CM-based agent performance for varying sizes of X, and compares CM performance to a normative game theory approach.

For this simulation we employ a sample repeated security game where the attacker and defender have four strategies each, and attacker payoffs are probabilities of attacker success as represented in Table 2. Given this payoffs matrix, if the defender played a fixed strategy (e.g., always play $D2$), the attacker could find a corresponding strategy that would win 90% of the time (e.g., if defender always plays $D2$, then attacker should always play $A3$). If the defender was equally likely to choose any action, the attacker could optimize by always playing $A1$, winning 52.5% of the games. A defender agent based on GT would employ a mixed strategy, choosing actions $D1$, $D2$, $D3$, and $D4$ with probabilities .275, .240,

[1] Playing $A1$ against a random opponent would have a 35% chance of winning, and playing any mix of $A1$ and $A2$ would have a chance of winning that is between 35% and 45%.

[2] Repeated security games differ from "one shot" security games in that the attacker attempts multiple attacks in sequence.

Table 2
Payoffs for the attacker in a security game used for Simulations 1-3, where there are four possible actions for the defender {D1,D2,D3,D4}, and four possible actions for the attacker {A1,A2,A3,A4}.

		Attacker			
		A1	A2	A3	A4
Defender	D1	.15	.45	.50	.90
	D2	.55	.10	.90	.45
	D3	.50	.90	.15	.45
	D4	.90	.50	.50	.10

.275, and .210, respectively. This mixed strategy would leave the attacker without a preference, where any given action would have a 50% probability of success.

However, humans are not perfectly rational, and human attacker action preferences will change based on their experience. For example, if the attacker chooses $A1$ and happens to lose, they will be less likely to choose $A1$ in future attacks, regardless of whether $A1$ is ultimately a good choice. Conversely, if the attacker chooses $A1$ and happens to win, they will be more likely to choose $A1$ in future attacks, regardless of whether $A1$ is ultimately a poor choice.

More formally, after performing some action, A, the expected utility of this action, U_A, is incremented by the following term:

$$\Delta U_A = \alpha(R - U_A), \tag{1}$$

where α is the learning rate, and R is the value of the feedback (e.g. success/failure, reward/punishment). This type of learning (error-driven reinforcement learning) is a very robust principle of biological brains (e.g., Anderson, 2007; Bayer & Glimcher, 2005; Fu & Anderson, 2006; Kable & Glimcher, 2009).[3] The action chosen at each decision point is one with the highest value of the term $U + X$, where X represents exploratory tendencies plus all the unknown factors driving human decision-making at the given moment.

From the perspective of predicting attacker actions, decision-making can be thought of as a stochastic process, where the action with the highest U is the most likely to be selected. A defender agent based on CM would employ *model tracing* to keep track of U values for each attacker action, and then employ this knowledge to outguess the attacker. That is, once each game plays out, and actual attacker action in that game, A, and outcome of the game, R, are both known, the model of the attacker can be updated in accordance with Equation 1. For each consecutive game, CM would predict that the most likely attacker action is one with highest U, and makes a corresponding best choice (e.g., if the attacker in this simulation is likely to choose $A2$, defender will choose $D3$).

Let us assume, for now, that we know attacker learning rate (in this simulation we assume $\alpha = .2$, as is the default in the ACT-R cognitive architecture; Anderson, 2007), and their perceived rewards for success and failure (in this simulation we assume $R = 1$ for each win, and $R = -1$ for each loss), and the attacker has no prior preferences for any actions. Holding these variables constant allows us to answer what the effect of X will be – the lack of predictability in human decision-making.

For general population X is often estimated as gaussian noise with a mean of 0 and a standard deviation, σ, that varies between 5% and 25% of maximum reward values. We ran the simulation for σ values of .05, .15, and .25, averaging over 1024 simulation runs per parameter setting. The results of this simulation, displaying the number of prevented attacks over the course of 200 consecutive security games, are shown in Figure 1.

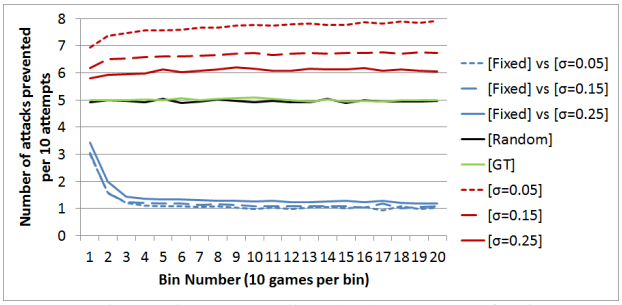

Figure 1. Simulation 1: Predicted advantages of using cognitive models in security games, given various levels of decision predictability. σ refers to uncertainty in the human attacker decisions, rather than in CM defender.

For all σ values in this simulation GT prevented 50.0% of the attacks (100.0 attacks prevented in 200 consecutive attempts). Depending on the predictability factor, CM prevented between 121.5 and 153.6 attempts on average (22%-54% more than GT). In addition to GT and CM, Figure 1 includes predicted baseline performances against random and fixed-strategy defenders. Predictably, a fixed-strategy defender does worst, losing in 75-90% of the games after about 10 initial games. Less obvious may be the fact that random-strategy agent performs almost as well as GT, winning on average 49.6% of the games, compared to 47.5% that we may have predicted against a perfectly rational agent.

The conclusion to draw from these simulation results is that humans are not normative decision makers, nor are we completely unpredictable; thus an approach that considers human cognition can perform better than normative GT. We can employ model tracing to improve defender performance in the security domain despite the fact that many attacker actions are not perfectly predictable. Finally, the less uncertainty there is in attacker behavior, the more attacks can be prevented. Thus, as we begin to account for a greater proportion of attacker cognition we can reduce the size of the X factor, and further improve defender performance.

[3]To be clear, the mechanism being described here, Reinforcement Learning, is only one of many cognitive processes that guides attacker behavior. In this paper we only focus on Reinforcement Learning in repeated security games as a clear and tractable example of how CM can aid in building better decision aids for real-world security domain problems.

Simulation 2: Preferred Strategies

Simulation 1 highlights the advantages of cognitive model predictions despite the uncertainty factor, X, in human decision-making. To isolate the effect of X we made a few assumptions, one of those being that the attacker has no initial action preferences. However, the four attacker strategies described in Simulation 1, Table 2 are not some arbitrarily named buttons $A1$-$A4$, but rather meaningful action-paths to the person(s) performing the attacks.

Let us assume, for example, that in the context of a cyber-attack, the hacker has two decisions: (1) whether to scan for vulnerabilities at a faster or a slower rate, and (2) whether to focus the attack on the main data server or on multiple perimeter machines. The hacker in this case has some risk aversion and believes that faster scans and attacking the main server present higher risks, resulting in initial perceived utilities for the *safer-perimeter*, *safer-main*, *faster-perimeter*, and *faster-main* options of +.10, +.05, 0.00, and -.10, respectively. We will refer to these initial utilities as preference set A, when they correspond to actions $A1$, $A2$, $A3$, and $A4$, respectively; and preference set B when they correspond to actions $A4$, $A3$, $A2$, and $A1$, respectively.

The question is, given that the attacker has some prior preferences, does it hurt the defender to assume that the attacker is a "blank slate"? Simulation revealed that CM would prevent about the same number of attacks against attackers with preference sets A or B as it would against blank slate attackers, see top of Figure 2. At worst, against an attacker with preference set B, σ = .05, CM prevents 2.3 attacks less in 200 attempts than against a blank-slate, σ = .05 attacker. The difference for each other attacker type is less than one attack in 200 attempts. GT agent performance remains at 50% regardless of attacker preferences.

The reason why initial preference sets A and B do not greatly disturb the model tracing approach has to do with the nature of error-driven learning (Rescorla & Wagner, 1972). This long-established principle of human learning suggests that the more surprising (i.e. unexpected) a given outcome is, the greater the change in the human (or simulated) brain. Thus, if human subjective utility for some action is 0.5 and the model assumes that utility to be 0.0, and the actual outcome value after that action was performed was 1.0, the change in action-utility in the model would be half of that in the human. After just a few experiences the model and human action-utilities would begin to converge, and model predictions would become more accurate.

A problem may occur in the instances where initial human preferences are strong enough that some actions are never even attempted. For example, let us examine preference sets C and D that are five times as strong as preference sets A and B, with initial perceived utilities for the *safer-perimeter*, *safer-main*, *faster-perimeter*, and *faster-main* options of +.50, +.25, 0.00, and -.50, respectively. Given these preference sets, an attacker (especially one with a low X factor) will be very unlikely to choose the *faster-main* option.

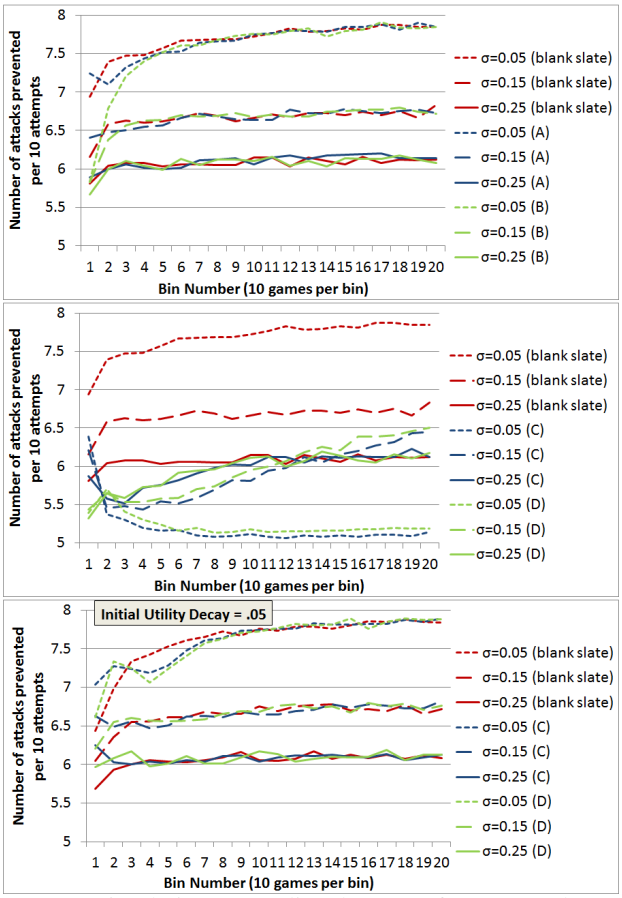

Figure 2. Simulation 2: Predicted CM performance when attacker has biases prior to the first attack. Each trend-line represents CM performance against a different attacker type (attackers differ by uncertainty factors and initial preferences). Preference sets A, B, C, and D are represented in figure legends as (A), (B), (C), and (D) respectively. Each data point represents an average over 1024 simulation runs.

This would throw off model predictions. A CM defender would still perform better than GT against attackers with such preferences, but much worse than it would against attackers with no initial preferences, see middle graph in Figure 2.

To account for potential extreme negative preferences, we can add Initial Utility Decay (IUD), dynamically adjusting initial action-utilities for any actions that are not being attempted by the attacker. The assumption is that if the attacker is not choosing a given option, then that option must have a lower subjective utility for them. For current simulation purposes we will employ linear decay, decrementing U_A by 0.05 at each decision point for every action A that the attacker does not exercise.

The results of employing a CM defender agent with initial utility decay are shown at bottom of Figure 2. As the figure suggests, initial utility decay greatly aids in accounting for strong initial preferences, resulting in almost no difference (less than 1 attack in 200 attempts) between defending against

blank-slate attackers and those with preference sets C and D. This improvement comes at a slight cost against a blank-slate attacker. CM performance with IUD is slightly worse than that without IUD against a blank-slate attacker (red lines at bottom and top of Figure 2, respectively), preventing 1.3, 0.4, and 0.4 attacks less in 200 attempts against attackers with σ values of .05, .15, and .25, respectively.

IUD is just one potential method for dynamic parameter adjustment in cognitive modeling. There are undoubtedly better alternatives to overcoming the problem of initial preferences other than linear IUD. The focus here is not in finding the optimal method, but rather in highlighting the benefits of using cognitive modeling in the security domain. This simulation suggests that even when individual attackers have unpredictable initial preferences, CM preferences can be adjusted dynamically without incurring a significant loss in performance.

Simulation 3: Learning Rate

Simulations 1 and 2 explore how CM-based defender agent performance in repeated security games is affected by decision unpredictability and initial preferences of human attackers. One other variable from Equation 1 that we have yet to discuss is the learning rate, α. This simulation focuses on CM performance given different attacker learning rates.

Table 3
Number of attacks prevented in 200 attempts by CM defender (IUD=.05). Each data point represents an average over 1024 simulations. α and σ refer to learning rate and uncertainty in the human attacker decisions. GT-based defender performance for all attackers is 100.0.

	CM learning rate				
	0.1	0.2	0.3	0.4	auto
α=0.1, σ=0.05	139.0	140.8	138.1	136.0	140.4
α=0.2, σ=0.05	139.4	152.5	152.3	151.6	151.9
α=0.3, σ=0.05	133.5	152.8	155.0	154.8	154.5
α=0.4, σ=0.05	131.6	150.1	154.4	154.7	153.9
α=0.1, σ=0.15	117.3	117.6	117.5	116.4	117.1
α=0.2, σ=0.15	130.9	133.0	133.0	132.2	132.3
α=0.3, σ=0.15	136.0	140.8	141.3	141.6	140.8
α=0.4, σ=0.15	136.1	142.6	144.4	144.4	143.9
α=0.1, σ=0.25	110.6	110.8	110.5	109.8	110.4
α=0.2, σ=0.25	120.5	121.6	121.5	121.4	121.3
α=0.3, σ=0.25	127.4	129.9	130.5	130.3	129.6
α=0.4, σ=0.25	131.3	134.6	135.9	136.0	135.1

Table 3 displays CM (IUD=.05) performance with different assumed learning rates, playing 200 consecutive security games against attackers with different actual learning rates.[4] In general, the higher the attacker learning rate (i.e., the more of a factor their experiences are in their decision-making), the easier it is to predict attacker decisions and prevent future attacks. Depending on the attacker, the range of differences between the best and worst CM performance was as high as 23 attacks in 200 attempts; though assuming a higher attacker learning rate does not hurt CM defender performance as much as assuming a lower learning rate.

The "auto" learning rate (right-most column) represents a CM defender that dynamically adjusts the learning rate parameter prior each decision. That is, the "auto" CM agent adjusts its assumption about the attacker learning rate depending on which assumed learning rate would result in a greater number of correct predictions for all known attacker decisions. As implemented in this simulation, the "auto" agent contrasts prediction history of CM agents with assumed learning rates of .1, .2, .3, and .4, and mimics the next decision of the agent with the best prediction rate.

Dynamically fitting the learning rate parameter in this way does not guarantee that the attacker learning rate will be correctly inferred at any given decision point, mostly due to the uncertainty in attacker decisions. Thus, the "auto" CM performance cannot be as good as that of an omniscient agent. However, the "auto" learning rate produces near-optimal performance, see Figure 3.

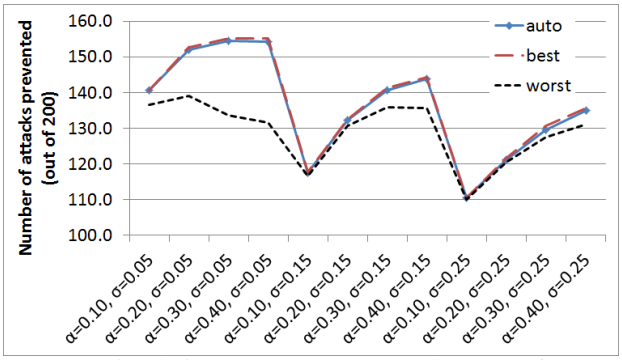

Figure 3. Simulation 3: Best, worst, and auto performance from Table 3. α and σ refer to learning rate and uncertainty in the human attacker decisions, rather than in CM defender.

To be clear, the "auto" learning rate method employed here is not the only method for adjusting model parameters. The focus here, however, is not on finding the optimal method for dynamic parameter fitting, but rather on highlighting the availability of techniques in cognitive modeling, such as model tracing and dynamic parameter fitting, for providing individual/team -tailored decision predictions that can be of great use in the security domain and beyond.

Summary & Discussion

In recent years, game theory -based decision-aid software has received significant attention for success in real-world security domain problems, such as scheduling patrols conducted by the US Coast Guard and police, allocating federal air marshals on flights, and major anti-poaching efforts. Behavioral game theory points out that normative approaches provide unrealistic predictions of human choice, and suggests the use of

[4]Results in Table 3 and Figure 3 are based on games against a blank-slate agent, but all effects hold against agents with initial preferences.

behavioral/cognitive models. In this paper we focus on the use of cognitive modeling to improve on the success of normative game-theory approaches in the security domain.

We present three simulations that highlight the potential advantages of employing cognitive models for predicting attacker decisions. The simulations suggest that (1) cognitive modeling provides performance advantages over normative game theory approaches, and (2) model parameters can be adjusted dynamically to make useful predictions about each individual human attacker despite the fact that each individual attacker may have different preferences and learning abilities.

The presented simulation results provide encouraging evidence of potential usefulness of cognitive models in the context of real-world security problems. Despite the fact that simulations in this paper are based on robust behavioral phenomena, the presented results should be taken as theoretical in nature, requiring further empirical validation. In future work we plan to gather human data and validate current simulation results.

In the current paper we only focus on a single cognitive process, as an example of how cognitive modeling may be employed in this domain. There is a wide array of established cognitive models beyond what we could explore in this paper. Integration of more models in CM-based decision aids would help in reducing the uncertainty factor, further improving the rate of prevented attacks.

Acknowledgements

This work was funded under Cooperative Agreement Number W911NF-09-2-0053.

References

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford ; New York: Oxford University Press.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive Tutors: Lessons Learned. *The Journal of the Learning Sciences*, *4*(2), 167–207. doi: 10.1207/s15327809jls0402_2

Anderson, J. R., & Reder, L. M. (1999). The fan effect: New results and new theories. *Journal of Experimental Psychology-General*, *128*(2), 186–197.

Bayer, H. M., & Glimcher, P. W. (2005). Midbrain dopamine neurons encode a quantitative reward prediction error signal. *Neuron*, *47*(1), 129–141.

Camerer, C. (2003). *Behavioral Game Theory: Experiments in Strategic Interaction*.

Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive psychology*, *4*(1), 55–81.

Choi, D., & Ohlsson, S. (2011). Effects of multiple learning mechanisms in a cognitive architecture. In L. Carlson, C. Hoelscher, & T. F. Shipley (Eds.), *Proceedings of the thirty-third annual meeting of the cognitive science society* (pp. 3003–3008). Boston, MA: Cognitive Science Society.

Fu, W. T., & Anderson, J. R. (2006). From recurrent choice to skilled learning: A reinforcement learning model. *Journal of Experimental Psychology: General*, *135*(2), 184–206.

Gächter, S. (2008). Behavioral Game Theory. In *Blackwell handbook of judgment and decision making* (pp. 485–503).

Gobet, F. (1998). Expert memory: a comparison of four theories. *Cognition*, *66*(2), 115–152.

Gobet, F., Lane, P. C. R., Croker, S., Cheng, P. C. H., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, *5*(6), 236–243.

Gray, W. D. (Ed.). (2007). *Integrated models of cognitive systems*. New York: Oxford University Press.

Holroyd, C. B., & Coles, M. G. H. (2002). The neural basis. of human error processing: Reinforcement learning, dopamine, and the error-related negativity. *Psychological Review*, *109*(4), 679–709.

Jastrzembski, T. S., Gluck, K. A., & Rodgers, S. (2009). Improving military readiness: A state-of-the-art cognitive tool to predict performance and optimize training effectiveness. In *The interservice/industry training, simulation, and education conference (i/itsec)*.

Jastrzembski, T. S., Rodgers, S. M., Gluck, K. A., & Krusmark, M. A. (2014). *Predictive performance optimizer*. Google Patents.

Kable, J. W., & Glimcher, P. W. (2009). The Neurobiology of Decision: Consensus and Controversy. *Neuron*, *63*(6), 733–745. doi: 10.1016/j.neuron.2009.09.003

Kahneman, D. (2011). *Thinking, fast and slow*. Macmillan.

Kar, D., Fang, F., Fave, F. D., Sintov, N., & Tambe, M. (2015). A Game of Thrones: When Human Behavior Models Compete in Repeated Stackelberg Security Games. In *2015 international conference on autonomous agents and multiagent systems* (pp. 1381–1390). International Foundation for Autonomous Agents and Multiagent Systems.

Lebiere, C., Wallach, D., & West, R. L. (2000). A memory-based account of the prisonerâĂŹs dilemma and other 2x2 games. In *Proceedings of international conference on cognitive modeling* (pp. 185–193).

Mackintosh, N. J. (1983). *Conditioning and associative learning*. Oxford, New York: Clarendon Press, Oxford University Press.

Nason, S., & Laird, J. E. (2005). Soar-RL: Integrating reinforcement learning with Soar. *Cognitive Systems Research*, *6*, 51–59.

Pita, J., Jain, M., Ordónez, F., Portway, C., Tambe, M., Western, C., & Kraus, S. (2008). ARMOR Security for Los Angeles International Airport. In *Twenty-third aaai conference on artificial intelligence* (pp. 1884–1885).

Reitter, D., & Lebiere, C. (2011). Towards cognitive models of communication and group intelligence. In *Proceedings of the 33rd annual meeting of the cognitive science society, boston* (pp. 734–739).

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. In P. W. F. Black AH (Ed.), *Classical conditioning ii: Current research and theory* (pp. 64–99). New York: Appleton Century Crofts.

Sample, C. (2015). *Cyber + Culture Early Warning Study* (Tech. Rep.). CERT. doi: CMU/SEI-2015-SR-025

Shanks, D. (1994). Human associative learning. In N. J. Mackintosh (Ed.), *Animal learning and cognition*. San Diego, CA: Academic Press.

Shieh, E., An, B., Yang, R., Tambe, M., Baldwin, C., DiRenzo, J., ... Meyer, G. (2012). Protect: A deployed game theoretic system to protect the ports of the united states. In *11th international conference on autonomous agents and multiagent systems* (pp. 13–20).

Simon, H. A. (1972). Theories of bounded rationality. In C. B. McGuire & R. Radner (Eds.), *Decision and organization* (pp. 161–176). msterdam: Elsevier.

Tambe, M. (2011). *Security and Game Theory* (Vol. 9781107096). doi: 10.1017/CBO9780511973031

Tambe, M., Jiang, A. X., An, B., & Jain, M. (2013). Computational game theory for security: Progress and challenges. In *Aaai spring symposium on applied computational game theory*.

Veksler, V. D., Myers, C. W., & Gluck, K. A. (2014). SAwSu: An Integrated Model of Associative and Reinforcement Learning. *Cognitive Science*, *38*(3), 580–598.

West, R. L., Lebiere, C., & Bothell, D. J. (2006). Cognitive architectures, game playing, and human evolution. *Cognition and multi-agent interaction: From cognitive modeling to social simulation*, 103–123.