Modeling Agents through Bounded Rationality Theories

Avi Rosenfeld[1] and Sarit Kraus[2]

[1] Jerusalem College of Technology, Jerusalem 91160, Israel
rosenfa@jct.ac.il
[2] Bar-Ilan University, Ramat-Gan 52900, Israel
sarit@cs.biu.ac.il

Abstract. Effectively modeling an agent's cognitive model is an important problem in many domains. In this paper, we explore the search agents people wrote to operate within optimization problems. We claim that the overwhelming majority of these agents used strategies based on bounded rationality, even when optimal solutions could have been implemented. Particularly, we believe that many elements from Aspiration Adaptation Theory (AAT) are useful in quantifying these strategies. To support these claims, we present extensive empirical results from over a hundred agents programmed to perform in optimization problems involving solving for one and two variables.

1 Introduction

Realistic modeling of individual reasoning and decision making is essential for economics and artificial intelligence researchers [1, 3–6, 10, 12, 13]. In economics, validly encapsulating human decision-making is critical, for instance in predicting policy effects. Within the field of computer science, it is critical for mixed human-computer systems such as entertainment domains [3], Interactive Tutoring Systems [6], and mixed human-agent trading environments [4]. In these and similar domains, creating agents that effectively understand and/or simulate people's logic is particularly critical. In both economics and computer science the perspectives of unbounded rationality based on notions such as expected utility, game theory, Bayesian models, or Markov Decision Processes (MDP) [7, 9], have traditionally been the foundation for modeling agent's behavior.

While many important insights have been gained by these perspectives, it often does not provide a descriptively correct model of human decision-making. Indeed, previous research in experimental economics and cognitive psychology has shown that human decision makers often do not adhere to fully rational behavior. For example, Kahneman and Tversky [2] have shown that individuals often deviate from optimal behavior as prescribed by Expected Utility Theory. Furthermore, decision makers often do not know the quantitative structure of the environment in which they act. But even if the quantitative structure of the environment is known to the decision maker, finding the optimal path of actions is often a problem with intractable computational complexity [8]. Thus, even in the best of circumstances, modeling behavior based on full rationality may be impractical.

A research direction called Bounded Rationality initiated by Simon [15] focuses on investigating observed rationality of individuals in decision making and problem solving. Simon presumes that people – except in the most simple situations – lack the cognitive and computational capabilities to find optimal solutions. Instead they proceed by searching for non-optimal alternatives to fulfill their goals. Simon proposes that real-world decision makers satisfice rather than optimize seeking a "good enough" solution instead of the optimal one. In this tradition, Sauermann and Selten propose a framework called Aspiration Adaptation Theory (AAT) [10, 12] as a boundedly rational model of decision making.

Recent experimental evidence from economics researchers provides support that people apply boundedly rational procedures, such as Aspiration Adaptation Theory (AAT) [10, 12], in real-world domains [11]. In this paper, we provide empirical evidence that the computer agents people write to search within optimization problems also contain several key elements from these models. This realization allows us to effectively model the decisions by these agents. Additionally, we present several guidelines by which realistic agents can be built based on this result. We begin by presenting the basics of AAT.

2 Aspiration Adaptation Theory

Aspiration Adaptation Theory was proposed by Selten as a general economic model of nonoptimizing boundedly rational behavior [10, 12]. We frame this theory within the specifics of the optimization problems presented in the next section. We refer the reader to the full paper [12] for a complete and general presentation of this theory.

As originally formulated, AAT is not a learning theory or a learning algorithm. Instead, it presents an approach by which bounded agents address a complex problem, $\mathcal{G}$. The complexity within $\mathcal{G}$ prevents the problem from being directly solved and instead an agent creates m goal variables G_1 , ..., G_m as means of solving for $\mathcal{G}$. These goals are incomparable, and no aggregate utility function, known as a substitution rate in economics, can be constructed for combining the m goals into $\mathcal{G}$. The lack of a utility function may be because the problem is simply too complex to quantify such a function, or because the agent lacks the resources to properly calculate the aggregate utility function for $\mathcal{G}$. In attempting a solution, the agent has a group of s instrument variables which represent base actions that can be used to pursue the goal variables. We define a *strategy* x = (x_1, ..., x_s) as a combination of instrument values for the goal variables G_1 , ..., G_m. These values can and do typically change over the life of the agent. For example, assume a company has a goal $\mathcal{G}$ to be optimally profitable. Examples of goal variables might be to create more sales, create higher brand awareness or to invest in the company's production infrastructure. Here, the instrument variables may include lowering the product's price, investing more money in marketing, or hiring more skilled workers. The agent might have one strategy at the beginning of its operation (e.g. an opening sale to entice buyers) and then use a different strategy after the business matures. An action *A* is a rule for changing the strategy. Formally, we define x' = A(x) for every strategy $x_1, \ldots, x_s$. Examples of actions in this context include:

- Raising the product's price by 5%.

– Lowering the product's quality by 10%
– Lowering product's price by 10% in conjunction with raising the marketing expenditure by 15%
– Making no change to the strategy

A finite number of n actions, $A_1 \ldots A_n$ are considered. The agent can only choose one action per time frame.

Despite its lack of utility to quantify rational behavior, the model provides several guidelines for how bounded agents will behave. Agents decide about goal variables as follows: The m goal variables are sorted in order of priority, or the variables' *urgency*. Each of the goal variables has a desired value, or the *aspiration level*, that the agent sets for the current period. The agent's search starts with an initial aspiration level and is governed by its *local procedural preferences*. The local procedural preferences prescribe which aspiration level is most urgently adapted upward if possible, second most urgently adapted upward if possible, etc. and which partial aspiration level is *retreated from* or adapted downward if the current aspiration level is not feasible. Here, all variables except for the goal variable being addressed are assigned values based on ceteris paribus, or all other goals being equal, a better value is preferred to a worse one. Borrowing from Simon's terminology [15] there is only an attempt to "satisfice" the goal values, or achieve "good enough" values instead of trying to optimize them. Note that this search approach is in contrast to traditional A.I. methods such as Hill-climbing or gradient descent learning techniques which can search for optimal values of all variables simultaneously [9].

While this theory has existed since the early 1960's [10], there are few empirical studies of how well it explains observed behavior [11]. As Selten's paper states, "AAT was meant as a useful point of departure for theory construction. The theory as it stands cannot claim to be a definite answer to the problem of modelling bounded rationality. Probably, one would need extensive experimental research in order to modify it in a way which fits observed behavior." [12]

In this paper, we study how AAT provides such a point of departure for studying optimizing search agents. While several differences do exist between AAT theory and the search agents we studied, overall we found many key similarities in their search process to those within this bounded rationality theory. To the best of our knowledge, this paper represents the first study that bridges between the fields of experimental economics and computer science to demonstrate the applicability of bounded rationality theory and in describing the model used by computer agents. Our next section details the exact methodology used to study the link between these fields.

3 Research Methodology

Central to our methodology is the strategy-method [13] from experimental economics. The assumption behind this method is that people can effectively implement their own strategies within a certain period of time, without additional aids such as handbooks or other information sources. Underlying this assumption is that people will execute a task to the best of their abilities if they are properly motivated, monetarily or otherwise.

In our study, all people writing agents were upper level undergraduates (seniors), masters and PhD computer science students and were given a firm deadline of 2 weeks to complete their agents. As motivation, we told the students that their grade was based on their agent's decisions. Once these programs were written, the mental model of the person's agent can be directly evaluated from its performance. This approach is well accepted in experimental economics and it had also begun to be used within artificial intelligence research [1] as well.

In order to ensure that people were able to effectively encapsulate their own strategies, several steps were taken. First, we took care to provide students a well designed Java framework in which methods were provided for most computations (e.g. finding the average of the agent's past performance). Thus, full knowledge of Java was not necessary, as strategies were meant to be encoded in only a few lines of code. This approach mimics the strategy-method variant previously used in economics [13] where people program their own strategies. The Java language was chosen as all students had experience using this programming language in multiple previous courses. Finally, after the first 2 week deadline, we required all students to submit a "draft" agent after which we reported back to all students with the relative performance of their agents to others in the group. The students were then allowed an additional week to fix bugs in their agents, or to improve their implementation without any penalty.

We used two tools to study the agent's design. First, we studied the code of the agent itself. By analyzing the agent's logic and comments, one can often understand the search process used. Additionally, after an assignment had been completed, we distributed questionnaires to the students themselves asking them directed questions about the strategy their agent used, confirming particulars of their approach. This allowed us to definitively determine what search mechanisms were used by these agent.

Our general methodology is as follows. We first study a relatively simple optimization problem, to understand the mental model used by agents to solve the problem. Next, we study progressively more difficult problems. Eventually, we hope to reach real-world types of optimization problems. By studying progressively less difficult problems first, we believe it will be easier to understand the general behavior of these agents and the applicability of our results. In this paper, we report on the first stage of this research.

Specifically, we studied two optimization problems – a commodity search problem where the optimal solution could be found based on solving for one cost instrument variable, and a more complicated domain where the optimal solution requires solving for price and quality instrument variables. In both domains an optimal solution can be constructed, and thus bounded rationality theories such as AAT theory are potentially unnecessary. However, the optimal solution is far from trivial in these domains. Thus, these problems allow us to explore issues surrounding the strategies and heuristics implemented by people's agents and questions of performance and optimality of these agents.

3.1 Commodity Price Optimization

In the first optimization problem, we consider a problem where a person must minimize the price in buying a commodity (a television) given the following constraints. In this problem, a person must personally visit stores in order to observe the posted price of the commodity. However, some cost exists from visiting additional stores. We assume this cost is due to factors such as an opportunity cost with continuing the search instead of working at a job with a known hourly wage. For any given discrete time period, the person must decide if she wishes to terminate the search. At this point, we assume she can buy the commodity from any of the visited stores without incurring an additional cost. The goal of the agent is to minimize the overall cost of the process which is the sum of the product cost and the aggregated cost of the search.

From a strategic point of view, the game is played under a time constraint rather than against an opponent. An optimal solution to this optimization problem can be found as an instance of Pandora's problem [16] resulting in a stationary threshold below which the search should be terminated. Formally, we can describe this problem as follows:

We assume that there is a finite timeline $\mathcal{T} = \{1, 2, ..., k\}$. In each time step t, $t \leq k$ the agent observes a cost and needs to decide whether to end the search. All of the observed costs, regardless of the time step, are drawn from the same distribution. We denote c_t as the lowest price the agent observed up to and including the time period t (i.e., $c_t \leq c_{t-1}$). At the end of the game the agent's cost is $cost(t, c_t) = c_t + \lambda * t$, $\lambda > 0$. The agent's goal is to minimize this cost. As has been previously proven, the optimal strategy in such domains is as follows: exists $\bar{c}$ such that if $c_t \leq \bar{c}$ the agent should stop the search [16].

Intuitively, it seems strange that the decision as to whether the agent should stop the search does not depend on how much time is left, i.e., $\bar{c}$ does not depend on $k - t$. However, the reason for this is as follows. If the agent's overall expected benefit from continuing the search (i.e., the reduction in price that it will obtain) is lower than the overall cost due to the added search time, the agent clearly should not continue the search. Furthermore, it was proven that it is enough for the agent to consider only the next time period, i.e., it should stop the search if and only if the expected reduction in the price in the next time period is less than the cost of continuing one time period (λ) [16]. To understand why, consider the following sketch of the proof: Suppose, to the contrary, that the agent is in time step t, the expected benefit from continuing to $t + 1$ is less than λ, but it will still be beneficial for the agent to continue until time $t' > t + 1$. The agent should then continue the search to time $t' - 1$. However, it is given that $c_{t'-1} \leq c_t$. Thus, given that the price in each time period is drawn from the same probability, the relative expected reduction of the price when moving from $t' - 1$ to t' is smaller than the expected reduction when moving from t to $t + 1$. Nonetheless, the cost per time step, λ is the same. Thus, we demonstrate that if it is not beneficial for the agent to continue from t to $t + 1$, it is also not beneficial to continue from $t' - 1$ to t'; contradicting our assumption that it is beneficial to the agent to continue until time period t'.

In our implementation, the prices are distributed normally with a mean μ and a standard deviation σ. We denote by x the price for which the expected reduction in the price for one time period is equal to λ. For a given price p the benefit is $x - p$ and the probability[3] for p is

$$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2}$$

Given these definitions we must generally solve:

$$\int_0^x (x-p)\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2}dp = \lambda$$

In our specific implementation, $\mu = 1000$, $\sigma = 200$ and $\lambda = 15$. Thus we specifically solve,

$$\int_0^x (x-p)\frac{1}{200\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{p-1000}{200})^2}dp = 15$$

Solving this equations yields a solution of $x = 789$.

Note that as an optimal solution exists, ostensibly there is no need for bounded rationality theories such as AAT. However, we believe that not only people, but even the agents they write on their behalf, do not necessarily effectively harness a computer's computational power to find optimal strategies. Thus, we predict that agents will use non-optimal search strategies involving instrument variables such as the current price of the commodity (x_1) and the elapsed time (x_2) as measured by the number of visited stores.

3.2 Price and Quality Optimization

In the second problem, we consider an environment of a company with a monopoly for a certain product. The owners of the company must set several important variables that will impact how much money the company will make. In this environment, there are no external factors to these decisions. Thus, the outcome of these decisions is not influenced by factors such as how other companies perform, what are other people's decisions, or random effects.

We formally present this problem as follows: We assume that there is a finite timeline $\mathcal{T} = \{1, 2, ..., k\}$. The agent needs to set two instrument variables, the price and quality of the product for any $t \in \mathcal{T}$, denoted p_t and q_t, respectively. The values of p_t and q_t can be set to any positive integer. The profit of the agent at a given time t depends on its choice for the price and quality until the current time. We denote by $\bar{p}^t = (p_1, ..., p_t)$ and $\bar{q}^t = (q_1, ..., q_t)$, the price and qualities determined by the agent until time t. The profit of the agent at a given time, $PT(\bar{p}^t, \bar{q}^t)$ consists of the gross profit and the expenses due to the quality. The part of the profit that is influenced by the price is:

$$PG(\bar{p}^t) = \bar{p}_t^t e^{-(\bar{p}_t^t-\mu)^2/\lambda_p^1}$$

The part of the profit that is influenced by the quality is:

$$QG(\bar{q}^t) = \lambda_q^1 QG(t-1) + \lambda_q^2 QG(t-2) + \lambda_q^3\sqrt{\bar{q}_t^t}$$

[3] In the domain, when a negative price was drawn, we drew a new price. Since the probability of such an event is extremely small, we did not take it into consideration in our analysis.

The profit at time t is

$$PT(\bar{c}^t, \bar{q}^t) = PG(\bar{p}^t) * QG(\bar{q}^t) - \gamma \bar{q}_t^t$$

The profit of the entire time period is: $\sum_{t \in \mathcal{T}} PT(\bar{p}^t, \bar{q}^t)$

All of the constants but μ and γ are known to the agent. In our experiments we set the constants λ_p^1 = 1000, $\lambda_q^1 = 0.7$, $\lambda_q^2 = 0.3$, and $\lambda_q^3 = 0.4$. For initialization purposes, q_{-1} and q_0 are set to 0. The value for μ is a randomly selected integer from a uniform distribution between 25 and 75 and γ is a randomly selected integer from a uniform distribution between 40 and 60. Finally, we set k=50 indicating that the company will only exist for 50 time periods.

The goal of the agent is to maximize the company's profit over the course of one trial. The agent operates within a one-shot environment which resets the values of μ and γ after every trial. Thus, no learning could be performed between these trials to learn the values of μ and γ. However, throughout one trial, for every time period the agent was given the values of $PT(\bar{c}^t, \bar{q}^t)$, $PG(\bar{c}^t)$, $QG(\bar{q}^t)$

Note that in this environment as well, an optimal solution can be found. Here, the only problem parameters with unknown values are μ and γ. It is possible to construct a table offline with the optimal values of price and quality given all possible permutations for μ and γ. Note that the size of this table will be 50 (as per k) times 50 (as per as possible values for μ) times 20 (as per as possible values for γ). Once this table has been constructed, the online agent only needs to identify what the values for μ and γ are so it may use the optimal prelearned values. As such, an optimal solution is as follows: In the first time period, the agent uses a predetermined value for γ that will yield the highest average profit. Once the agent observes the company's profit after the first iteration, it is able to solve for the unknown value of μ. In the second iteration we can similarly solve for γ as it is known to be the only remaining variable. After this point, the agent sets the price and quality for every remaining time step as per the prelearned optimal values for these values of μ and γ. Alternatively, another optimal solution involves first solving for the unknown value for γ in the first iteration, for μ in the second iteration, and using the prelearned optimal values after this point.

As an optimal solution is again possible in this domain, bounded rationality theories such as AAT seem irrelevant. However, we generally believe that two significant factors contributed to the student's inability to optimally solve these problems. First, both problems were verbally presented and thus students needed to properly model the problems before solving them. Second, even after these problems were properly modeled, correctly solving for these problem parameters was far from trivial and required significant algebraic knowledge. As a result, we again hypothesize that people will use non-optimal strategies here involving search within instrument variables of the company's price (x_1) and quality (x_2) parameters.

While the commodity search optimization problem and the slight more complex monopoly domain are relatively simple, the similarities and differences between them allows us to generalize our findings. The one commodity search problem is characterized by complete information, but a certain level of randomness (non-deterministic behavior) exists in what the price of the commodity is in a given store. Also note that the optimal solution involves making a decision based on the current price alone. In the first domain, other instrument variables, such as the number of visited stores or the length of the time horizon, are not part of the optimal solution. In contrast, the monopoly game is characterized by deterministic functions with two unknown parameters (μ and γ). While this problem is more straightforward, the introduction of a second variable makes the problem seemingly more complex. Furthermore, the optimal value for quality changes over time, and can only be found by solving for γ. Nonetheless, both domains are generalized representations of real-world problems [1, 11] and thus serve as good domains for studying the models of search agents.

4 Results and Analysis

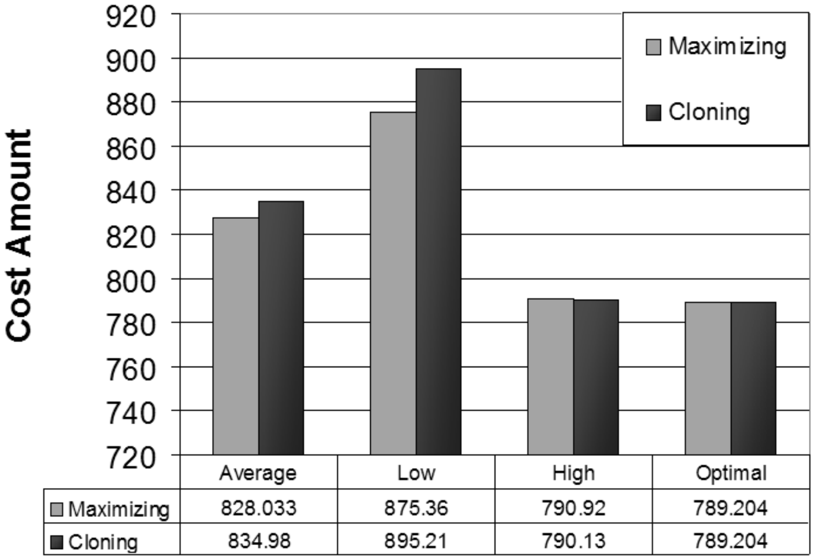
We studied how people's agents performed in the above commodity search and monopoly domains. Within the commodity search domain, we studied the agents from 41 senior undergraduate and graduate students. Within the monopoly domain, we studied the agents from a different group of 57 senior undergraduate and graduate computer science students.

4.1 Non-Optimal Performance

In both domains, a minority of the agents did in fact exhibit performance near or close to that of the optimal agent. However, the vast majority of these the agents deviated significantly from optimal behavior.

As per previous work by Sarne et al. [1], the 41 agents from the commodity search domain were divided into 14 “maximizing” agents and 27 “cloning” agents. The “maximizing” agents were written by students who were asked to create as high performance as possible (optimal). The “cloning” agents were written by people who were instructed to mimic their own personal strategies. As one focus of the experiment was how effectively people could clone their own strategies, the maximize group served as the control group with a ratio of 1:2. As random effects do exist in this environment, we ran each of these agents 50 times, and averaged the agent's performance. We then compared the average performance from the “cloning” and “maximizing” groups, the best performing agent from each of these groups, and the worst performing agent from each of these groups. Finally, we compared the performance of the optimal agent to the agents the students wrote.

Figure 1 shows the performance of these agents. Note that the goal of this domain was to minimize the search price. As such, the low search cost of the best agent (column 3) closely approximated the performance of the optimal agent (column 4). It is important to also note that the average cost of both the “cloning” and “maximizing” agents (approximately 830 units) were quite far from the optimal agent (approximately 790) with p-values testing for significance being much less than 0.0001. However, the differences between the “cloning” and “maximizing” agents were not significant (p-value 0.48). These results imply that most people asked to write optimal agents fall well short of this amount, and do in fact, closely replicate their own non-optimal strategies. This result validates the use of the strategy method [13] from experimental economics as people were typically successful in implementing their own strategies.



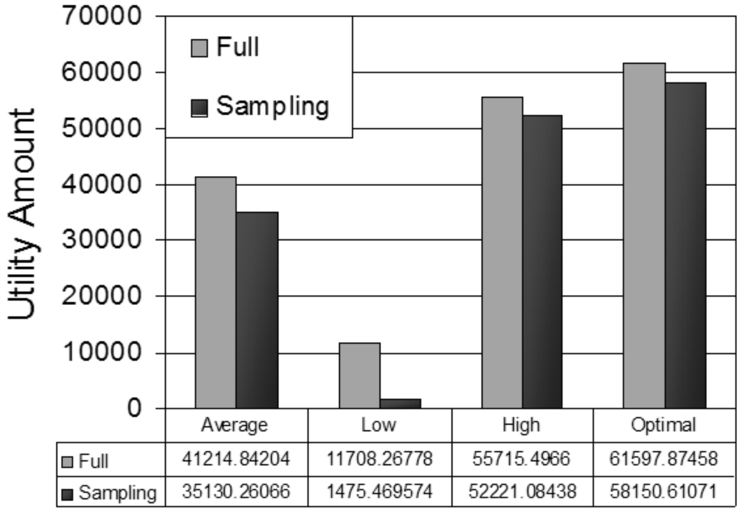
	Average	Low	High	Optimal
Maximizing	828.033	875.36	790.92	789.204
Cloning	834.98	895.21	790.13	789.204

Fig. 1. Comparing the Average, Worst, and Best Utility Value of “Maximizing” and “Cloning” Commodity Search Agents to Optimal Values. Lower costs are better.

Once we verified that the strategy method could be applied to agents written to act within optimizing problems, we studied a second group of 57 maximizing agents written for the monopoly domain. We again studied the average, highest, and lowest performance across the agents, and compared this performance to that of the optimal agent. In this domain, many different values were possible for the previously described price and quality functions. As such, we applied two different evaluation approaches. Within the *Full* evaluation, we studied the average performance of the agent across all possible permutations of price and quality. In the *Sampling* evaluation, we studied the average performance of the agents across six randomly selected value pairs of these values. Realistically, the second type of evaluation seems more appropriate as people typically build small numbers of companies. However, the fuller evaluation in the *Full* group is useful for statistical testing.

Figure 2 displays the performance of the agents from the monopoly domain. Note that again in this domain, the performance of the best agents (third column) in both the *Full* and *Sampling* evaluation methods reached near optimal levels. However, the agents’ average performance (first column) again fell well short of the optimal (p-values between the optimal performance and the Full and Sampling evaluation groups were both well below 0.0001). This again strongly supports the claim that people’s agents typically fall far short of optimal values. Note that in the more complex monopoly domain, the average agent’s performance was over 30% less than the optimal value, while in the simpler commodity search problem, this difference was closer to only 5%.

This seems to imply that as problems become progressively harder, bounded agents seem to perform progressively further from the optimal values.



	Average	Low	High	Optimal
Full	41214.84204	11708.26778	55715.4966	61597.87458
Sampling	35130.26066	1475.469574	52221.08438	58150.61071

Fig. 2. Comparing the Average, Worst, and Best Utility Value of People's Monopoly Agents to Optimal Values. Higher utilities are better.

4.2 Elements of AAT to Quantify Behavior

Our goal was not just to verify the non-optimality of people's agents, but to generalize what non-optimal strategies are in fact being used. To the best of our knowledge, this paper represents the first of its kind that demonstrates that many agents designed to solve optimizing problems in fact implemented strategies consistent with bounded rationality, and specifically key elements of AAT.

It is important to note that several key differences exist between classic AAT theory, and the behavior exhibited by the search agents in the domains we studied. First, AAT assumes that the m goal variables used to solve $\mathcal{G}$ are incomparable. Here, we consider optimization problems where some function between $\mathcal{G}$ and the m goal variables clearly exists, but we hypothesize the agent will not attempt to calculate it due to its bounded nature. Second, AAT is based on the premise that the agent's search will be based on an *aspiration scale* which sorts the m goal variables and attempts to satisfice values for these goals. As we consider concrete optimization problems, it is more natural for agents to consider optimizing the instrument variables that constitute the basis of these goals rather than the more abstract general goal variables. For example, we would expect an agent in the monopoly domain to focus on variables such as price and quality instead of higher level abstract goals such as “brand awareness” or “company infrastructure”.

Nonetheless, we hypothesize that bounded search agents would still exhibit three key elements inspired by AAT. First, we expect the agents to prioritize certain instrument variables to be solved first. This concept parallel’s AAT’s concept of urgency between its m goal variables. Second, we expect the agent to stop its search within a given instrument variable once it has satisficed this value. Finally, we expect that at times an agent will change its satisficing threshold during the search process. This parallels AAT’s concept of retreat and should be expected if the agent deems its original goal is no longer realistic. For example, within the commodity search domain an agent may begin by first attempt to find a commodity below a certain threshold price. However, assuming a certain time elapses (or a number of stores have been visited), it may revise downward this threshold as being a “good enough” solution. In the monopoly domain, the agent might set a priority between which variable, price or quality will be searched for first. After this variable has been satisficed, the agent will then proceed to the other variable.

Note that these three qualities are not an optimal optimizing approach. These problem solving approaches make no attempt to calculate the optimal solution through means such as solving for the unknown parameters within the problems (see the previous section). Furthermore, in contrast to more traditional A.I. methods [9], they do they attempt a simultaneous search on multiple variables.

To study this point we constructed a short list of several questions by which we determined the model of the agents. These questions included: How many variables did the agents attempt to solve for? What were they? Was a search process used to set these instrument variables, or was a predefined strategy (independent of actual performance) used instead? If search was used, were the variables searched for simultaneously, or sequentially? If sequential search was used, did the agent revisit variables after it had originally set a value for it (such as to retreat, or revise downward the previously set threshold value). We recognize that despite the wealth of log files and strategy descriptions provided by the agents’ authors, at times some ambiguity may exist in an agent’s model as how to answer these questions. To overcome this issue, we had a total of three people judge each of the agents in our results. Of these three people, two were not authors on this paper, and thus had no bias.

Instrument Variable	Judge 1	Judge 2	Judge 3	Average
Price	6	6	7	6.33
Stores Visited	5	5	5	5
Both w/ Retreat	30	30	29	29.67

Table 1. Goal Variables in the Commodity Search Domain

Table 1 presents the number of agents categorized by each of these judges in the commodity search problem. This table depicts how the 3 judges categorized the number and search variables of the agents. The optimal solution within this problem involves search within only one variable, the current price of the commodity. Nonetheless, the judges found that on average 6.33 of the 41 agents made decisions based on this variable alone (see row 1, column 4). While none of these students actually used the optimal strategy (buy if the price in the current store is less than 789 – as generally dictated by [16] within our implementation), several of the students did have similar strategies such as buy if the price is less than 800. Another 5 students actually used another search variable, the number of stores visited, to make decisions (see row 2). For these students, the strategy was to visit a predetermined number of Y stores and to buy the commodity in the cheapest store from the group of Y. Both of these strategies only contained one instrument variable. As such, they can be viewed as basic search strategies involving only one variable (e.g. search until price $< X$, or visit Y stores and buy in the cheapest store).

Surprisingly, approximately 73% of the agents (see row 3 column 4, average 29.67 of 41) use combination strategies. For these students, the strategy was to immediately buy the commodity if the price was less than X, otherwise, they visit a maximum of Y stores and buy in the store with the cheapest found price. While non-optimal, this strategy has key elements of AAT. Originally, agents search based on price alone. However, if the desired price is not found, the agent downward revises its aspiration. This can be seen as being similar to the retreat process within AAT's goal variables. Note that the use of *urgency* here is not even justified based on optimal behavior as the second goal variable (Stores Visited) is not even part of the optimal solution! Furthermore, the combination strategy of settling on a price after visiting Y stores if the no commodity was found with a price less than X is a good example of *retreat* values. Here, the price less than X is aspired for. However, assuming this cannot be found after visiting Y stores, the satisficing threshold within this variable is revised and a lower value is accepted.

Case	Variable 1	Variable 2	Judge 1	Judge 2	Judge 3	Average	AAT?
1	Optimal	Optimal	0	0	0	0	No
2	Simultaneous Search	Simultaneous Search	9	13	8	10	No
3	Predetermined	Predetermined	3	3	4	3.33	No
4	Search Price	Search Quality	9	14	23	15.33	Yes
5	Search Quality	Search Price	4	0	1	1.67	Yes
6	Search Price	Predetermined Quality	27	22	15	21.33	Trivial Search
7	Search Quality	Predetermined Price	0	1	0	0.33	Trivial Search
8	Alternating	Alternating	5	4	6	5	Yes

Table 2. Comparing the Agent Strategies within the Monopoly (Price and Quality) Domain

Similarly, the results in Table 2 present the analysis of the three judges of the monopoly agents' cognitive models. None of the students used the optimal strategy to solve for both price and quality variables (line 1). A number of agents did simultaneously search for both the price and quality variables (line 2), and a smaller number of students did use predetermined heuristics for setting both variables (e.g. price always equals 10 and quality equals time elapsed). Both of these strategies do not contain elements of AAT as no urgency exists between variables (in line 2), or no search for goal variables is performed (in the predefined case).

In the vast majority of the agents (approximately 77%) search was conducted with elements of AAT. For most agents (lines 4 & 6) the price variable was searched for first (e.g. had the highest urgency), after which quality was either searched for (in line 4) or set by a predetermined function (in line 6). A very small number of agents had the opposite *aspiration scale* with quality being searched for first (in lines 5 & 7). Similarly, only a relative small number of agents (line 8) consistently alternated between searching for price and quality. In this approach, an agent would set one value (say price), in the next time frame search for the optimal value of the second goal (quality), only to return back to the first goal variable (price) and adjust downward (retreat from) its original value. The reason why fewer agents made use of retreat in this domain seems as follows. According to AAT theory, retreat occurs once an agent realizes it must change its aspiration based on the infeasibility of satisfying multiple goals. In the monopoly game students typically did not see any infeasibility element and therefore did not retreat between variables. However, in the commodity search problem, students (albeit wrongly) saw infeasibility and therefore retreated back on their values. Thus they made use of *retreat* variables to refine their aspirations.

We hope to further study what specific mechanisms were used by agents to determine when it had satisficed a given goal variable. This direction is motivated by several points in Tables 3 & 4. For example, note that in the simpler cost search domain agents revisited previous values (and revised the other goal variable downwards through retreat), but in the more complicated domain they typically did not (except for the agents in line 8 of Table 2). Instead, most agents in the monopoly domain first satisficed the price value (albeit typically with a non-optimal value), and then tried to satisfice the quality goal, never to return to the price goal. Second, note that most differences between the judges in Table 2 revolved around differences in classifying an agent's model as one that is predefined or based on search (see differences in lines 4 & 6). We instructed the judges to assume an agent used search if it changed or retreated from its goal because of previous performance, but not search if it changed its goal because of some predefined value. For example, if an agent perceived that its performance dropped because it raised its quality value, and then decided to lower its quality value, search was used. However, if the agent decided to lower its quality value, even by the same amount, because of some predefined constant function, they were instructed to categorize the agent as having a predefined strategy without search. This definition is based on Learning Directional Theory (LDT) [14] whereby agents change the search process for goals based on previous performance. However, questions arose in cases where goal variables seemed to be intertwined (e.g. an agent set its quality as a function of the price value it was searching for). Additionally, this definition required the judge to read the agent's code and strategy files, and not just observe its performance. Furthermore, many monopoly agents that simultaneously searched for both goal variables (line 2 of Table 2) and thus were not classified as using AAT often instead used the boundedly rational model of LDT to search for these values. Consequently, we are currently studying if LDT can be extended to better describe why and how goal variables are satisficed, and when an agent will reorder its *aspiration scale* to revisit previous goal variables during search.

Overall, several conclusions can be drawn from the results in both of these domains. First, nearly all agents written to “maximize” performance fell far short of doing so. Within the commodity search domain 30 of 41 agents of all agents used strategies consistent with elements of AAT’s *urgency* and *retreat* concepts, while the remaining agents considered a trivial search case where only one goal variable was searched for. Within the monopoly domain, on average approximately 44 of 57 agents used AAT based strategies. Thus, we conclude that optimal approaches cannot properly model most people’s agents, and bounded rationality models such as AAT should be used instead.

5 Conclusions and Future Work

In this paper we report our findings about the model used by people’s agents to operate in two general optimization problems. These problems can be generalized to many real-world domains [1, 11] and thus our findings contribute two significant findings for Artificial Intelligence researchers. First, we empirically demonstrate that people, or even the agents they write on their behalf, are poor optimizers. Even when we explicitly asked two different groups of over 70 people to write agents to optimally solve a problem, and an optimal solution existed, they instead chose to use approaches that fell well short of optimal behavior. Thus, one must conclude that encapsulating human behavior based on optimal strategies is not effective for certain domains. Second, we find that key elements of Aspiration Adaptation Theory (AAT) do effectively encapsulate many people’s search strategies. However, AAT was original formulated for domains where utility cannot be measured and thus did not make any guarantees about performance, or how close to optimal this behavior is [12]. Thus, the results in this paper is particularly important, and indicates the importance and greater generality of using bounded rationality models even in problems where optimal solutions exist and its applicability to search problems.

While the focus of this paper is quantifying the cognitive model of agents, our results lead us to the following conclusions about how to write agents that better interact with people or simulate human performance. First, optimal methods should not be used as they do not realistically encapsulate most human’s behavior. Instead, bounded methods should be created such as those based on AAT. In understanding the strategies used by people, we propose that a small pilot be used based on the strategy method [13]. This should identify the ordering (urgency) for search variables and a range of aspiration values within these variables. For example, in the domains we studied, such a pilot would clearly identify price as the variable first searched for in both domains. Finally, any pilot should be focused on the range of aspired for values in each goal variables such that some distribution can be constructed to realistically model the range of problem solving approaches.

For future work, several directions are possible. First, while we found that people’s strategies fell short of optimal behavior in both optimization problems we studied, we assume that people will write rational and optimal agents in simpler problems. We hope to study the level of problem complexity that motivates people to abandon optimal solutions for those based on bounded rationality. Second, we hope to study how effective the above general conclusions are in simulating human behavior in these and other domains. Finally, we also hope to study how people interact with agents based on AAT versus those based on optimal or other predefined heuristic strategies. Specifically, we hope to study agent-human interactions in an emergency response domain. We are hopeful that AAT and other theories of bounded rationality can be applied to these and other agent-based domain problems.

References

1. M. Chalamish, D. Sarne, and S. Kraus. Programming agents as a means of capturing self-strategy. In *AAMAS '08*, pages 1161–1168, 2008.
2. D. Kahneman and A. Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47:263–291, 1979.
3. Pattie Maes. Artificial life meets entertainment: lifelike autonomous agents. *Commun. ACM*, 38(11):108–114, 1995.
4. Efrat Manisterski, Raz Lin, and Sarit Kraus. Understanding how people design trading agents over time. In *AAMAS '08*, pages 1593–1596, 2008.
5. Yohei Murakami, Kazuhisa Minami, Tomoyuki Kawasoe, and Toru Ishida. Multi-agent simulation for crisis management. In *IEEE Workshop on Knowledge Media Networking (KMN02)*, 2002.
6. Yohei Murakami, Yuki Sugimoto, and Toru Ishida. Modeling human behavior for virtual training systems. In *AAAI*, pages 127–132, 2005.
7. John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.
8. David V. Pynadath and Milind Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *JAIR*, 16:389–423, 2002.
9. S. J. Russell and Norvig. *Artificial Intelligence: A Modern Approach (Second Edition)*. Prentice Hall, 2003.
10. Heinz Sauermann and Reinhard Selten. Anspruchsanpassungstheorie der unternehmung. *Zeitschrift fr die Gesamte Staatswissenschaft*, 118:577–597, 1962.
11. R. Selten, S. Pittnauer, and M. Hohnisch. Experimental results on the process of goal formation and aspiration adaptation. In *Technical Report – Bonn Laboratory for Experimental Economics*, 2008.
12. Reinhard Selten. Aspiration adaptation theory. *Journal of Mathematical Psychology*, 42:1910–214, 1998.
13. Reinhard Selten, Michael Mitzkewitz, and Gerald R. Uhlich. Duopoly strategies programmed by experienced players. *Econometrica*, 65(3):517–556, 1997.
14. Reinhard Selten and Rolf Stoecker. End behavior in sequences of finite prisoner's dilemma supergames. *J. of Economic Behavior and Organization*, 7:47–70, 1986.
15. Herbert A. Simon. *Models of Man*. John Wiley & Sons, New York, 1957.
16. Martin L Weitzman. Optimal search for the best alternative. *Econometrica*, 47(3):641–654, 1979.