Modeling Assistant's Autonomy Constraints as a Means for Improving Autonomous Assistant-Agent Design*

Nadav Kiril Altshuler
Bar-Ilan University
Israel
kirilis02@gmail.com

David Sarne
Bar-Ilan University
Israel
sarned@cs.biu.ac.il

ABSTRACT

In this paper we introduce and experimentally evaluate a new sub-optimal decision-making design to be used by autonomous agents acting on behalf of a user in repeated tasks, whenever the agent's autonomy level is continuously controlled by the user. This mode of operation is common and can be found whenever user's perception of the agent's competence is affected by the nature of the outcomes resulting from the agent's decisions rather than the optimality of the decisions made, e.g., in spam filtering, CV filtering, poker agents, and robotic vacuum cleaners as well as in newly arriving systems such as autonomous cars. Our proposed design relies on choosing the action that offers the best tradeoff between decision optimality and the influence over future allowed autonomy, where the latter is predicted using standard machine learning techniques. The design is found to be highly effective compared to following the theoretic-optimal decision rule, over various measures, through extensive experimentation with a virtual investment agent, making virtual investments on behalf of 679 subjects using Amazon Mechanical Turk.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**;

KEYWORDS

Human agent interaction

ACM Reference Format:

Nadav Kiril Altshuler and David Sarne. 2018. Modeling Assistant's Autonomy Constraints as a Means for Improving Autonomous Assistant-Agent Design. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

An important role of collaborative interfaces and many AI-based systems in general is supporting people in decision situations. This typically takes the form of providing people beneficial advice or suggesting a preferred course of action [13] and can be found in almost any aspect of our daily lives. For example, GPS-based navigation apps recommend their users a driving route (e.g., GoogleMaps and Waze), weight loss apps recommend a daily diet (e.g., Nutrino) and fitness apps recommend a preferred course of training (e.g., Runkeeper). With recent developments in AI technologies and the ever increasing adoption of such systems as mainstream, many of these systems have evolved to autonomously carrying out various types of decision-oriented tasks on behalf of their user rather than simply providing advice. These can either take the form of virtual agents (e.g., Poker bots that play on real money [7], automatic spam filters [10], automatic news feed generators [30] automatic CV filters [24]) or physical ones (e.g., autonomic car, vacuum robots).

While agents of the latter type are equipped with a wide set of functionalities enabling them to act fully autonomously, their decision space as well as the level of autonomy they may actually exhibit (i.e., the space of actions they may choose from) are often constrained by the user.[1] For example, a poker agent can be constrained by the amount of money it can bet on each round, a robotic vacuum cleaner can be constrained by virtual walls set by the user, limiting it to a sub-area of the apartment to be cleaned, a GPS system may be limited by the type of roads it may consider. These constraints typically hold whenever it is not clear how competent the agent is, or to what extent the agent's goals are aligned with the user's. The constraints change over time, based on the user's perception of the agent's competence as well as various other psychological factors that may influence her satisfaction from the agent.

The fact that the agent's autonomy may be constrained calls for a somehow different design paradigm for such agents. For years, decision support systems and advice-giving agents were designed to act optimally [6, 36], in the sense of maximizing some predefined measure (e.g., the expected time to get somewhere in the case of a navigation systems, or the coverage achieved per time unit in the case of a robotic vacuum cleaner). Alas, since people are bounded rational [23], and do not always recognize the optimality of the decisions made by the agent [20, 37], a sequence of sub-optimal decisions may result in a greater user satisfaction and allow the agent to act (almost) fully autonomously in the long run, in a way that the aggregated performance is improved overall. Prior work has shown that indeed in many cases suboptimal advising is a preferred choice for collaborative interfaces and virtual advisors [3, 12, 13, 28]. Still, the motivation for using suboptimal advice in prior work was the non-intuitive nature of the optimal solution, i.e., at times, sub-optimal solutions might seem more appealing to the user and hence are likely to be adopted to a greater rate, leading to an improved performance overall. As such, the solutions proposed relied on the tradeoff between the level of intuitiveness of different

*Produces the permission block, and copyright information

[1] We stress that while we use the terms "autonomy" and "level of autonomy", the idea has nothing to do with the concept of "adjustable autonomy" which commonly appears in literature in the context of transferring control between a user and an agent.

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

solutions and the expected value they encapsulated in case as the primary design principle (and consequently the autonomy level she allows).

In this work, we introduce a new type of sub-optimal decision making designs to be used in a different class of settings where it is the uncertain nature of the outcomes that affect the user's perception of the agent's competence. This is in fact the more common setting with agents that make decision on behalf of the user (rather than giving advices) as the user often does not get to observe the decision made but rather only receives information related to its outcome (e.g. algo-trading [22]), Here, even the optimal decision may result in poor actual outcomes, as outcomes are a priori uncertain.

Our proposed design, the adaptive design, relies on predicting the effect of different outcomes, resulting from choices made by the agent, over the extent of autonomy to which it is enabled. This is done using standard machine learning techniques. The design aims to balance between decision optimality and the resulting future autonomy of the agent, picking the action that maximizes some pre-defined tradeoff between the two. The idea is that over time, as the user's confidence in the agents' competence increases, the agent will gradually be able to make decisions that better align with the theoretical-optimal ones.

The effectiveness of the adaptive design was tested experimentally using an investment game in which an agent is making investments on behalf of the user and the user gets to control the amount of funds to be used by the agent in its investment, out of the total funds available, over time. The performance of the adaptive design was compared to the one achieved with an agent that acts optimally, over various measures such as the average autonomy granted to the agent over time, the user's average overall profit, user's expressed satisfaction and agent's profit. The evaluation used two different instances of the framework differing in the stocks available to the agent for investment. The analysis of the results suggests a statistically significant improvement in all measures when using the adaptive design.

In the following sections we provide a formal problem formulation. Section 3 details our adaptive agent design. The experimental framework and experimental design used for evaluation are given in Sections 4 and 5, followed by the analysis of the results. Finally, we provide a review of related work and our conclusions.

2 PROBLEM FORMULATION

Our model considers an autonomous agent engaged in a repeated task on behalf of a user (e.g., navigating, investing money, making dietary choices).[2] We use $A = \{A_1, ..., A_n\}$ to denote the set of actions (or choices) available for the agent in the task. While the user's satisfaction from the agent's decisions is a priori unknown, the agent knows it is positively correlated with a primary well-defined "performance" measure X pertaining to the actual outcomes of the decisions it makes. The value of X resulting from picking any different action $A_i \in A$ is a priori uncertain and the agent is only acquainted with its underlying probability function, denoted $p_i(x)$ (or the probability distribution function $f_i(x)$ in the case of a continuous random variable). While the agent is not computationally bounded and can act autonomously without any continuous feedback from the user, the user has the option to restrict its *Autonomy Level* by constraining the action space available to the agent, as discussed in the former section. This is not to imply the user expects the agent to transfer back control by limiting autonomy but rather to say the agent is limited to picking its action from a reduced (possibly more controlled) space.

The model assumes the user has the same a-priori knowledge as the agent, but does not observe (or is not interested in) the choices made by the agent but rather only get to see the actual outcomes of these choices. Naturally these outcomes have a great effect on the user's satisfaction and the way she constrains the agent's autonomy level in consequent tasks.

The above maps to all the application domains mentioned in the former section as motivating examples. In navigation, actions correspond to picking among different applicable routes, performance is measured as time to get to destination, constraints are roads and areas to avoid or maximum number of segments to use and the sources of uncertainty are roads conditions, weather and traffic congestions along the road. In the investments domain, actions correspond to different allocations of the user's funds to stocks and bonds, performance is measured as the relative return on investment, constraints can be expressed as the maximum amount per investment or type of stocks to invest in and the uncertainty derives from the unstable behavior of stocks in the market. In dieting, actions are different recipes and food choices, performance is typically measured in terms of weekly weight loss, constraints can be the minimum or maximum requested calories per day or specific foods to avoid or include and the source of uncertainty is the inherent difference between people and the way different aspects, e.g., sport, may influence their diet.

The goal of the agent is thus to decide, on each instance of the task execution, on the action to be used, given the outcome of previous choices made and the resulting imposed constraints aiming to maximize some pre-defined measure such as user's profit, user's satisfaction and agent's profit.

3 PROPOSED APPROACH

This section outlines the principles of the proposed agent design. We begin by reviewing the optimal strategy for the agent if the user was a fully rational agent and then move on to the case of people.

Fully-rational user. If the user is fully rational then in the absence of any input other than the performance measure X of the choices made, and since there is a strong positive correlation between the performance measure and the user's satisfaction from the agent, it is optimal to pick the action a associated with the maximal expected performance. Formally: $max_{A_i} \sum x \cdot p_i(x)$.[3] This strategy maximizes the expected observed performance at any time, and hence will provide the maximum autonomy for the agent when carrying out

[2] Note that in this examples we refer to actually making the choice of a route, investment and daily diet, rather than generating few alternatives and requesting the user to choose from.

[3] This assumes linear correlation between the performance obtained and the user satisfaction and between user satisfaction and the degree of autonomy awarded in subsequent task. This is likely to be the case since the user is fully rational and the task is repeated. However, even in cases where the correlation is not linear all that needs to be changed is replacing x by the corresponding functions that capture the correlation between these variables.

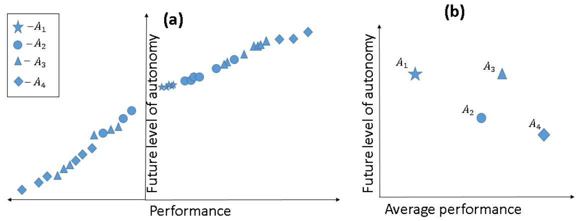

Figure 1: Future autonomy level as a function of performance: (a) for all outcomes; (b) for expected outcomes.

the subsequent task. We denote this strategy "theoretic-optimal" and use it as a baseline in our experimental evaluation.

Adaptive approach. While the theoretic-optimal approach will perform best with fully rational users, people are known to be bounded rational in the sense that their actions are influenced by various psychological effects [23]. In particular, the uncertainty associated with the outcomes may frequently lead to occasions where theoretically optimal actions result in poor outcomes, overwhelmingly influencing the constraints imposed by the user on the agent's future autonomy. Therefore the adaptive agent design aims to weigh in both the influence different outcomes of different choices will have on subsequent autonomy constraints and their performance contribution of as defined by the measure X.

The tradeoff between performance and the awarded autonomy is illustrated in Figure 1. Graph (a) of the figure schematically illustrates the level of autonomy to be granted (vertical axis) resulting from different outcomes of different choices (horizontal axis), given some specific history of outcomes. In this example we have four possible choices ($A_1, ..., A_4$), each associated with four possible outcomes.[4] Outcomes are ordered according to their performance measure, and the general relation to the awarded autonomy is positive - the greater the performance achieved, the greater the user satisfaction and consequently the greater the autonomy awarded to the agent in subsequent tasks. Still, from this graph it is difficult to derive preference (or dominance) relationships between different choices. These can be extracted based on outcomes aggregation as depicted in graph (b). Here, each data point represents the average position of each outcome (based on all its outcomes) in the bi-dimensional plane. The choice of average in this case is the result of the repeated nature of the interaction, as prior work provides much evidence that in repeated-play settings people's strategies asymptotically approach the expected monetary value (EMV) strategy as the number of repeated plays increases [5, 26]. From Graph (b) of Figure 1 we can derive some dominance relationships. For example, choice A_3 dominates A_1 and A_2, however there is no clear dominance relation between A_4 and the others (or between A_1 and A_2). Therefore, some tradeoff between these two aspects (average performance and average resulting autonomy) should be defined.

The adaptive approach, aims to pick the choice offering the maximum value among those available, according to such tradeoff. The idea is to stick to the choice that does not compromise much in terms of the performance achieved and yet is not likely to jeopardize the autonomy of the agent in subsequent tasks due to possible poor outcomes. This offers a general gradual improvement both in the performance achieved and the level of autonomy awarded along time. Meaning that over time, the method will gradually shift from more conservative choices that are required for influencing the user to award a greater autonomy to the agent, to more risky (and yet associated with better performance on average) choices.

Formally, we use $\theta(H_i, x)$ to denote the prediction of the autonomy level to be awarded to the agent in following tasks. The function takes as an input the entire set of prior observations of the performance achieved by the agent (i.e., the actual outcomes resulting from choices made by the agent) in the former i tasks carried out and the autonomy level set by the user in each consecutive round (encoded by H_i)[5] and a possible performance outcome x of the current choice to be made by the agent. In order to extract $\theta(H_i, x)$ it is required to have some data on the autonomy level awarded by people to the agent given different histories, on which any standard machine learning technique can be used in order to produce the necessary prediction. Based on $\theta(H_i, x)$ we can calculate the expected autonomy across all possible outcomes of a choice a, denoted $\theta_a(H_i)$: $\theta_a(H_i) = \sum \theta(H_i, x)p(x)$.[6] The immediate expected performance if picking action A_j, denoted $E_{A_j}[x]$, is $E_{A_j}[x] = \sum x p_{A_j}(x)$. Therefore the agent needs to consider the tradeoff between the two, denoted $G(\theta_{A_j}(H_i), E_{A_j}[x])$ (where the latter is an increasing function of its two parameters). The preferred choice $A_j \in A$ when the current encoded history is H_i is thus: $a = argmax_{A_j} G(\theta_{A_j}(H_i), E_{A_j}[x])$.

In our experiments we used an equal weight for $\theta(H_i, x)$ and $E_{A_j}[x]$, as this is a natural choice and the goal was merely to provide a proof of concept. In general, the optimal tradeoff between the two to be used is domain dependent and may require empirical investigation. Giving a substantial weight to the immediate performance will yield a strategy that is close to the theoretical-optimal strategy specified above, thus suffers from the same problems discussed above. On the other hand, giving substantial weight to the level of autonomy obtained is likely to result in lack of progress as far as actual performance is concerned - a rather mediocre (yet highly conservative) choice will be made over and over again.

4 EXPERIMENTAL FRAMEWORK

To evaluate the adaptive approach, we used an experimental framework called "the investment game". In this multi-round game the user starts with an initial budget, representing her initial wealth, and can allocate some of it to the agent to invest on her behalf. Both the agent and the user are acquainted with the different investment opportunities and the underlying distribution of gains based on which their profit is determined.[7]

The agent has complete freedom to invest the amount it receives in any of the opportunities available. Its autonomy level is defined as the amount it receives for investment out of the user's total funds. At the end of each round, the user gets to see the results of the investments made by the agent. Meaning that while she does

[4]For example, assume each choice is a different stock, and the different outcomes are the possible returns from each. The greater the return the less likely it is that the user will limit the agent in the amount of money to be used in future investments.

[5]Former autonomy levels are needed in order to learn how different outcomes influence subsequent autonomy restrictions for this specific user.

[6]For continuous distributions the calculation is: $\int \theta(H_i, x)f(x)dx$.

[7]In real-life the agent can estimate the underlying probability distribution of return based on analysts' estimations and overall market dynamics.

not become acquainted with the specific investment made, the user realizes the profit (or loss) resulting from the investment.

For its service, the agent charges the user a commission (expressed as a percentage of the total amount allocated by the user for investment) in each round. The choice of charging a commission was made for two reasons. First, it enables distinguishing the user's goal from the agent's goal—the agent gets to collect its commission even if the outcome is poor (e.g., loss). Thus, the user cannot trivially assume that she should allocate all her funds to the agent (as if the agent is fully "on her side") but rather should base her decision on her impression from the agent's investment outcomes. Second, this is the common practice with many of the (physical) agents making investments on behalf of a user (e.g., a stock broker firm or a pension plan). Having said the above, we stress that gaining autonomy is critical for the agent, as even though the commission is always paid, its absolute size depends on the amount allocated by the user for investment.

While admittedly limiting, relying on a single testbed is relatively common in the area of advice provisioning and human-agent interaction in general, especially when the proposed method suggests a paradigm shift in design, and the goal is to provide a proof of concept for the new approach (e.g., [18, 28]). Furthermore, our use of the investment game offers many advantages in evaluating the adaptive method. First, most people are familiar with the application domain as it maps to various real-life settings where an individual delegates others to make investments on her behalf: e.g., allocating funds to a pension plan, allocating available funds from a checking account to a saving account or a CD and the management of a stock portfolio by hedge funds and professional brokers. Furthermore, many of these latter funds allocation examples are of a repeated nature, and the user gets to decide on the level of autonomy for the agent in terms of the amount allocated for investment as well as specifying guidelines for investment. People's familiarity with the domain is important as it reduces the amount of explanations needed regarding the game rules and significantly reduces the chance of subjects failing to understand the game flow and objectives. Second, in this game the performance measure is well defined (the return on investment) and highly correlated with user satisfaction. The fact that we are dealing with stocks enables generating and experimenting with various highly different settings within that framework, e.g., by controlling the number of different stocks available for investment and their corresponding probability distribution of returns. Finally, as mentioned above, the small commission charged enables distinguishing the agent's goal from the user's goal.

5 EXPERIMENTAL DESIGN

In the following paragraphs we provide a detailed description of the specific parameter values used for the experiments with the framework, specific choices made in the implementation of the learning module of our agent, the interaction with participants and the different treatments used in our experiments.

Framework Implementation. We implemented the investment game as an interactive website, using *ASP.NET* for the server side and *Html*, *css* and *javascript* for the client side so that participants could interact with the system using a relatively simple graphic interface. Participants' initial budget was set to \$100. The commission charged by the agent was set to $\alpha = 0.2\%$.

Agents Implementation. We designed the agents such that they invested all the money they were allocated in one stock. This choice was made for two primary reasons. First, every investment in a mixture of stocks can be seen as investing in one stock that adheres to an equivalent joint distribution (just like hedge funds often mimic the behavior of a certain sector of stocks), hence this does not affect the generality of the solution. Second, and perhaps more importantly, this restriction enabled us to focus in a reduced number of investment alternatives, as the agent only had to pick the stock to invest in rather than evaluating a combinatorial number (or an infinite number of) allocation alternatives.

Two types of agents were implemented for the experiments. The first is the theoretic-optimal agent that according to the guidelines given earlier in the paper always picked the stock associated with the maximum expected return. The second agent was implemented according to our adaptive approach. In order to develop an effective prediction model for the level of autonomy to be awarded based on past outcomes, we first implemented a "Random" agent, that randomly picked the stock to invest in. We used this agent to experiment with 495 people, yielding a total of 7546 data samples, each encoding the level of autonomy awarded in a given round and the history of returns resulting from investments made till that round. This data was used as an input to three classic machine learning methods (all implemented using the Python scikit-learn library [34]): Neural Networks [40] (used with 2 hidden layers, with 100 neurons in each and the rectified linear unit function ('relu') which returns $f(x) = \max(0, x)$), SVM [17] with the 'rbf' kernel function ($e^{-\gamma \|x-x'\|^2}$), using the default parameters of the 'sickit.learn' package), and Random Forest [41].

The starting point for the learning included a set of 32 possible features: current round number, current amount of funds, return achieved in the last i rounds (where i is feature's parameter, values checked between 5 and 13), agent's autonomy level in the last i rounds (where i is feature's parameter, values checked between 5 and 13), number of rounds in which the user lost money, and the average gain percent of the first i rounds (where i is feature's parameter, values checked between 1 and 10). To obtain the best combination of features and their parameters we trained all models with all possible combinations of features and their parameters. The training was done using the cross validation method [14] with 20 iterations (test-train ratio was 80%-20%), checking the mean error on the test set. The error was calculated as the absolute distance between the prediction and the correct answer.

The mean error as a function of the amount of samples using the three learning methods is shown in Figure 2(a). From the graph we see that all methods perform quite well, even with a relatively moderate number of observations used as an input. This is highly encouraging, given the substantial number of parameters constituting the input space (as each instance encodes the outcomes of all prior rounds and the level of autonomy awarded). Based on the results obtained we concluded that the best model in our case is Random Forest. The best features subset found for this method includes: the current round number, the current amounts of funds

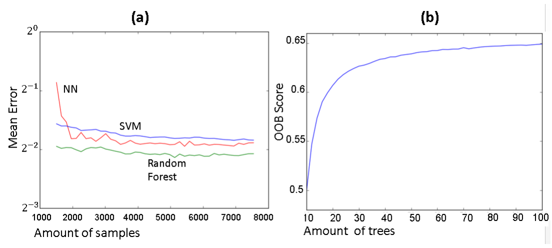

Figure 2: Training charts: (a) Mean error as a function of sample's amount; and (b) OOB Error as a function of the number of trees in the Random Forest method.

available to the user, the gains (in percentages) obtained in the last five rounds and the autonomy level awarded in the last five rounds. Figure 2(b) depicts the OOB (out of bag) error [8] obtained with this method while training is performed using the full set of samples as a function of the number of the trees used. From the graph we concluded that 70 trees are sufficient as beyond that number the improvement achieved is quite moderate. We emphasize that the above learning process is standard and by all means it is not one of the contributions of this work.

The future autonomy level and the immediate performance measures were given equal weights in $G(\theta_{A_j}(H_i), E_{A_j}[x])$. This arbitrary choice, as explained in previous sections, was made primarily because our goal was to prove a concept. Specifically, in our case the two measures had an approximately close range (immediate performance ranged between 1.06 and -0.12 and naturally the future autonomy level ranges between 0 and 1), hence linear combination with equal coefficients seemed natural.

Choice Availability. We extracted two sets of choices available to the agents, denoted $S1$ and $S2$ respectively, each emulating a different market structure, enabling a richer experimentation. Each set had 8 stocks the agent could invest in. The stocks' return distributions were generated based on real stocks data, collected from Yahoo Finance through the "matplotlib.finance" Python library [46]. For each stock we calculated the return over 20 different (randomly picked) pairs of dates, where pairs for $S1$ were picked within the first half of 2016 and those of $S2$ within the second half of 2016. This choice was made since we found a greater variance both within the individual return and within the average return between different stocks in the latter period. Naturally, both in $S1$ and $S2$, a stock with a higher expected return is also associated with a greater risk of loss.

Measures. In order to evaluate the effectiveness of the agents in the different experimental treatments we used several complementing measures, each offering a different perspective. The first measure is the User's Profit, i.e., the cumulative amount of money each user ended up with. The second measure is the Autonomy Level, i.e., the percentage of funds transfered by the user to the agent for investment, out of her total funds. The third measure is the Agent's Profit, defined as the cumulative amount of money the agent received as a commission throughout the experiment with any given participant.[8] Much like the second measure, this measure corresponds to the level of autonomy warranted to the agent by the user, however the latter gives more weight to absolute larger amounts.

[8]As discussed earlier, this measure is only partially correlated with the user's profit, since the agent charges the user regardless of whether the investment ended up in a gain or a loss.

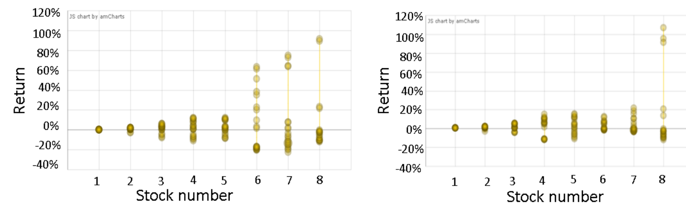

Figure 3: The choice of eight stocks available to the agent in $S1$ (right) and $S2$ (left) and the distribution of their returns.

While the above measures quantify performance, it is possible that they are not fully correlated with User's Satisfaction [28]. Therefore we also used subjective users reporting as a measure for their satisfaction. To this end, we asked users to specify at the end of the experiment whether they were satisfied with the investment agent, whether will recommend this agent to a friend and whether will use this agent again (if could). Each of the three questions relate to user satisfaction from a slightly different aspect. For example, the answer to the third question reflects user's loyalty and there is a direct link between loyalty and user satisfaction [21].

Interaction with Subjects. Participants were recruited and interacted with through the crowd-sourcing framework of Amazon Mechanical Turk (AMT). AMT has proven to be a well established method for data collection of tasks which require human intelligence to complete [33]. Each participant received thorough instructions of the game rules, the compensation terms and her goal in the game and were asked to engage in practice rounds until stating that they understood the game rules (with a strict requirement for playing at least three practice rounds). In order for the users to become acquainted with the different options available to the agent and their possible outcomes, participants were equipped with a chart depicting stocks and their predicted return distribution (see Figure 3). Prior to moving on to the actual games, participants had to correctly answer a short quiz, making sure they fully understand the game and the compensation method. Finally, participants were requested to play a sequence of 20 rounds. To encourage truthful participation, in addition to a show-up fee (the basic "HIT") participants were also awarded a bonus which was linearly correlated with the amount of virtual dollars they managed to accumulate throughout the game.

Experimental Treatments. Participants were assigned to one of four treatments, differing in the agent used (theoretic-optimal and adaptive) and the set of stocks used ($S1$ and $S2$). To prevent any carryover effect a "between subjects" design was used, assigning each participant to one treatment in one experimental framework only. In order to have a better control over the experiments and minimize the influence of drawings from the underlying distribution functions over the results obtained we pre-generated for each stock in each framework, using the data collected from YAHOO finance, 10 sequences of returns over 20 periods corresponding to the performance of the stock over the course of the game.[9]

[9]This data is downloadable from: https://tinyurl.com/InvestmentGameData.

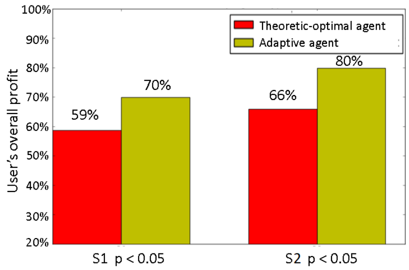

Figure 4: User profit as a function of the agent and stocks set used sets $S1$ and $S2$.

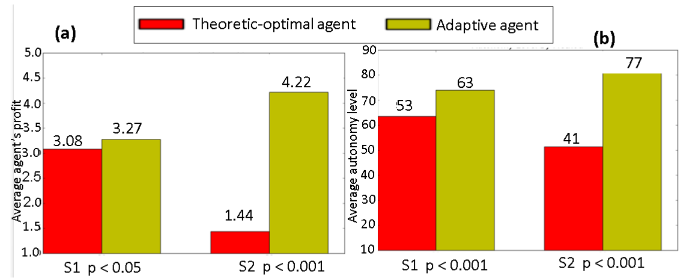

Figure 5: (a) average agents' profit; and (b) average autonomy level.

6 RESULTS

Overall we had 679 participants taking part in our experiments, with 317 experimenting with the adaptive agent and 362 experimenting with the theoretic-optimal. Participants ranged in age (21-70), gender (67.3% men and 32.7% women), education, and nationality (50.8% from US, 27.4% from India and the rest from numerous other countries), with a fairly balanced division between treatments.

In the following paragraph we present the results' analysis in a comparative manner (adaptive agent versus the theoretical-optimal agent), according to the profit users managed to achieve, the agent's profit, user satisfaction and the level of autonomy the agent gained in its decisions. Statistical significance is calculated based on the one way Mann-Whitney Wilcoxon (MWW) test [29] which is a non-parametric test (hence it is the most suitable for our case), with alternative hypothesis that one population tends to have larger values than the other [31]. We note that the same statistical-significance results were obtained with the one way t-test, except for a single comparison, of a lesser importance, related to agent's profit when using the set $S1$ for which we received $p_{t-test} = 13\%$).

User's Profit. Figure 4 depicts the average (cross-users) cumulative user's profit in the four treatments. From the figure we observe that with both stocks sets the adaptive method led to a greater user profit, and in both cases the difference is statistically significant ($p < 0.05$). We note that a bound to the theoretical maximum user profit that can be obtained is 90% with $S1$ and 100% with $S2$ (achieved when always investing all money available to the user in the stock associated with the maximum expected return). Therefore based on the results reported in Figure 4 we can say that the use of the adaptive method reduced the difference between the user's expected profit achieved with the theoretic-optimal agent and the highest-achievable profit through the use of any agent by at least 35% and 41% for $S1$ and $S2$, respectively.

Further analysis of individual profits (i.e., break down to specific users) reveals that, generally, the improvement achieved in the average user's profit does not result from an increase in the profit of some at the expense of others. Meaning, that individual gain generally improved and there is no specific group of individuals whose gain actually worsened. Therefore, in summary, we conclude that the adaptive agent outperforms the theoretic optimal agent as far as user's profit is concerned.

Agent's Profit. Figure 5(a) depicts the average (cross-users) agent's cumulative profit (i.e., commissions charged) in the four treatments. As mentioned earlier, this measure is only partially correlated with the user's profit, since the agent charges the user regardless of whether the investment ended up in a gain or a loss.

With both stock sets the adaptive agent managed to make a greater profit compared to the one the theoretic-optimal agent made, and in both cases the difference is statistically significant ($p < 0.05$). We note that the theoretical maximum agent-profit, i.e., if the user always transfers all her accumulated funds to the agent for investment and the agent is always picking the stock associated with the maximum expected return, is \$4.1 with $S1$ and \$5.0 with $S2$. Therefore based on the results reported in Figure 5(a) we can say that the adaptive agent reduced the difference between the profit the theoretic-optimal agent managed to achieve and the theoretical achievable one by 18.6% and 78%, for $S1$ and $S2$, respectively.[10] We assume that the reason for the huge differences between $S1$ and $S2$ lies in the differences between stocks sets. As mentioned before the stocks in $S2$ offer larger potential gains and yet reflect a greater variance in their returns. Therefore there is a greater chance to run into a poor outcome in $S2$, resulting in a substantial decrease in the agent's autonomy level and consequently in its gain.

Overall, despite the fact that agent's profit and user's profit are not fully correlated, we can see that the adaptive agent achieved significantly better results in both measures. Thus, the improvement in user's profit is not at the expense of the agent's profit.

User's Satisfaction. Figure 6 depict the average subjective user satisfaction reportings received when experimenting with the two agents for the two stock sets. From the graphs we observe a dramatic improvement (statistically significant with $p < 0.001$) in all three user satisfaction measures when using the adaptive agent compared to the case of using the theoretic-optimal agent. The improvement by itself was quite expected, as the user's average profit using the adaptive agent is greater than with the theoretic-optimal agent (see Figure 4). However while the magnitude of the absolute and relative improvements in user's average profit is somehow moderate, the corresponding magnitude of improvement in user satisfaction is quite overwhelming: with $S1$ stocks user satisfaction doubled with the use of the adaptive method and with $S2$ stocks it tripled and even quadrupled (in the recurring use measure).

Interestingly, when using the theoretic-optimal agent the user satisfaction reported with $S1$ stocks was greater than the satisfaction reported with $S2$. This is somehow counter-intuitive, as the average user profit obtained when using this agent in the $S1$ treatment is smaller than the profit obtained in the $S2$ treatment. This

[10]This is once again a lower bound for the improvement obtained, as explained above.

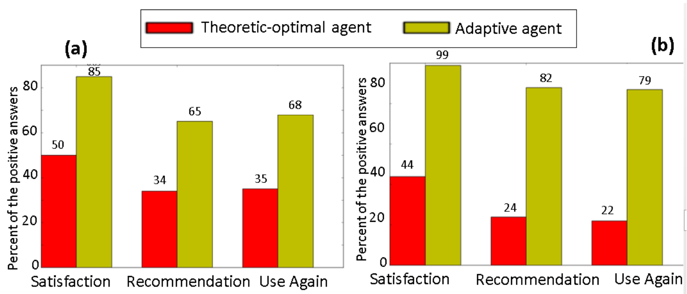

Figure 6: Percent of positive answers in the user satisfaction questions for the theoretic-optimal agent and the adaptive agent.

is yet another evidence to the claim that the expected performance and user's satisfaction are not perfectly correlated. With the use of the adaptive method agent, substantially greater satisfaction was reported in the $S2$ treatment compared to the reports in the $S1$ treatment. Meaning that the greater the risk in the stocks set (or, in general, the greater the variance in the outcomes resulting from different choices available to the agent) the greater the absolute satisfaction of the user from the Adaptive method agent. Similarly, the greater the risk in the stocks set the greater the improvement in the user's satisfaction due to switching from the theoretic optimal agent to the Adaptive method agent (tripling and quadrupling user satisfaction compared to doubling it, in our experiments).

Autonomy Level. The analysis of the autonomy level granted to the agent aims to shed some light over the source of improvement achieved with the adaptive agent compared to when using the theoretic-optimal agent as reported in the former paragraphs. As discussed in the former section, we had two measures that directly relate to the level of autonomy granted to the agent by the user. The first is the agent's profit, which is a constant percentage out of the amount of funds transferred to it for investment, and indeed with the use of the adaptive agent this measure of autonomy increased.

The second measure is the portion of the funds transferred to the agent for investment out of the user's total accumulated funds (i.e., the percentage transferred). This measure is more informative, in the sense that it is not sensitive to differences in the amount of funds available to the user due to the results of prior investments made. Figure 5(b) depicts the average percentage of the funds the user made available to the agent for investment, out of the total she had at the time, for the four experimental treatments. As can be seen from the figure, the adaptive agent was allowed to invest a greater percentage of the funds available to the user, compared to the theoretic-optimal agent, both in $S1$ and in $S2$. Both improvements are statistically significant ($p < 0.001$). Interestingly, when switching from $S1$ to $S2$, we observe a decrease in the portion of the funds the user allows the agent to invest (from 54% to 41%) when using the theoretic-optimal agent, whereas with the adaptive agent we observe an increase (from 64% to 77%). A possible explanation for this is that as the variance in outcomes increases (with the transition to $S2$) the use of the theoretic-optimal agent results in a decrease in the autonomy granted as the agent keeps investing in a stock that often yields poor returns (though its expected return is the highest) and the user becomes reluctant to allocate

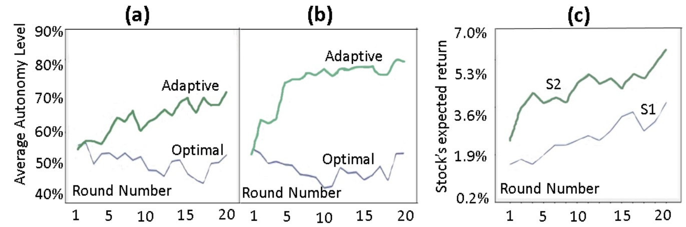

Figure 7: Mean autonomy level as a function of round number: (a) $S1$ and (b) $S2$, and the expected return of the invested stock as a function of round number (Graph c).

funds. With the adaptive method the agent manages to overcome the problem by starting with more conservative stocks in order to maintain a good level of autonomy in future investments and then gradually shifting to stocks offering a greater expected return (and yet naturally more risky), as the effect of poor outcomes on the level of autonomy reduces. This is also supported by the analysis of Figure 7(c) as discussed in the analysis of the choice of the stock to be used. Overall, this latter finding related to the reversed effect reflected in the transition from $S1$ to $S2$ aligns well with the findings reported as part of the user satisfaction analysis, and in particular the reversed change in user's satisfaction in the transition from $S1$ to $S2$ with the theoretic-optimal agent and the adaptive one.

Graphs (a) and (b) of Figure 7 depict the average autonomy level enabled to the two agents in $S1$ and $S2$ based on the different game rounds. The starting point of both agents (i.e., in round 1) is the same - users allocate slightly less than 60% of their initial funds, both in $S1$ and $S2$, as at this point there is no history that can influence the user. With the theoretic-optimal agent the level of allowed autonomy seems to slightly decrease along rounds, possibly due to some disappointment from poor outcomes and lack of improvement in the average return over time. With the adaptive agent, however, we actually observe an increase in the autonomy level as the game progresses. This latter finding, which is explained better below through the analysis of the choice of the stock used, has a very important implication: as the number of rounds increases the difference between levels of the autonomy awarded to the two agents are likely to increase, favoring the adaptive agent. These behaviors are consistent cross stock sets.

Choice of a Stock. Finally, we present Figure 7(c), which depicts the average return of the stock picked by the adaptive agent in the different rounds.[11] From the figure we observe that the expected return increases over time. Meaning that the agent is gradually switching to more risky, and yet of greater expected return, stocks. The graph captures the essence of the evolution in the choices made by the adaptive agent and together with the analysis given above unfolds the reasons for its success - by considering the effect of different outcomes over the level of autonomy to be awarded in subsequent investment rounds, the agent manages to establish a high level of autonomy, which in turn enables it to gradually shift to better paying stocks, this time however without compromising its future autonomy in case of a poor outcome.

[11]The average return with the theoretic-optimal agent is fixed along rounds, as the same stock is picked over and over again.

7 RELATED WORK

Autonomous agents acting on behalf of a user can be found in various domains (e.g., trading in Energy Markets [25], automated planning and scheduling [43]) and forms (e.g., multi-purpose assistants [27], decision support systems [32] and various others [4, 11]). Common to most designs of such agents, that the agent is fully autonomous, hence follows an optimal strategy in order to maximize some pre-defined measure of performance. Unlike with this very long line of work, in our case, the agent's autonomy is inherently constrained by the user, and the extent of the constraints placed are continuously influenced by its actions and their outcomes. This difference is critical, as human users typically do not adhere to rigid models of rationality and are easily influenced by various external factors and biased towards certain conclusions [23, 42]. This has been used in various designs of different user interfaces [2, 12, 19, 20]. Using the theoretical-optimal strategy in our case thus can (and does, according to the results presented in the former section) result in inferior overall performance.

The use of sub-optimal strategies in the design of collaborative agents can be found in recent prior literature dealing with advising agents [3, 13, 28]. Here, however, the choice of the sub-optimal advice is the result of the non-intuitiveness of the optimal advice. In our case the reason for not using the theoretical-optimal action is the influence over the future level of autonomy enabled to the agent. Therefore, the considerations made have nothing to do with the nature (or intuitiveness) of the actions taken but rather the potential outcomes of different actions. Another work [9] shows that imperfect advisors may benefit in greater user's trust and better performance, yet our research focus on an autonomous agents that chooses between several options rather than giving binary device.

Much research has been focused on the question of when to transfer decision-making control from the agent to the user [39], typically in the context of "adjustable autonomy". For example, in cases where the agent is unable to preform the task with complete autonomy [47], when the sensibility and importance of the task make human intervention crucial [15, 44] or when the agent can benefit from having the user guide the enumeration of subspaces of the full problem space (e.g., in planning and scheduling) whenever full enumeration is impractical [1]. The main difference between this line of work to ours is that in adjustable autonomy, when in control, the agents will try to act optimally in terms of expected outcomes while our design picks "suboptimal" actions, taking into account the tradeoff between future constraints to be imposed on its actions and the possible outcomes (performance-wise) of different actions.

In a way, the reliance of the adaptive design on the influence different actions will have over future autonomy constraints can be seen as an implicit modeling of some notion of user trust or user satisfaction. Modeling different aspects of user's trust in agents, and computer systems in general, is a widely studied topic due to its importance to the design of computer software [16, 35, 38, 45]. Still, these works concern primarily methods modeling and enhancing trust and to the best of our knowledge none of these deal with managing the tradeoff between trust (or the influence over the agent's future autonomy level) and the optimality of the strategy used by the agent in repeated settings where the agent's autonomy level is continuously constrained by the user.

8 DISCUSSION, CONCLUSIONS AND FUTURE WORK

The encouraging results reported in the former section suggest that indeed the adaptive method is a highly effective alternative to the use of the optimal strategy for agents that preform repeated tasks on behalf of the user—not only did the user's expected profit in our experiments increase but also statistically significant improvement was obtained in user satisfaction and agent's profit. Furthermore, the learning part, which is the only additional overhead induced by the new design, required a moderate number of observations in order to converge. Overall, the results show that the adaptive agent manages to steadily progress both in terms of the autonomy users awarded it and the quality of the choices made. The adaptive design is thus an important contribution to this fast-growing research field and ought to be considered whenever designing agents aiming to act on behalf of the user.

Naturally, an agent that has the privilege to learn is likely to benefit from this edge. Still, in our case the agent's learning is limited to the level of autonomy it is likely to be granted. This by itself is not enough to warrant good performance since, as discussed in Section 3, giving substantial weight to the level of autonomy obtained is likely to result in lack of progress as far as actual performance is concerned as a rather mediocre choice will be made over and over again. Meaning that autonomy can be maximized simply by continuously picking the safest stock (one that is least likely to lose), alas, this will result in poor average performance, making the learning detrimental. We note that in preliminary experiments carried out while working on this research we have used such strategy and the results reflected the exact phenomenon described above. Therefore learning by itself does not contribute much to performance (and hence no a priori advantage can be attributed to this aspect with respect to the performance achieved by the adaptive agent). Instead, it is the continuous gradual progress along the autonomy and performance dimensions that accounts for the improvement achieved with the adaptive method.

We emphasize that the performance achieved by the adaptive agent in our experiments is a lower bound to the performance one may achieve with the adaptive method, as we were using an arbitrary tradeoff between predicted autonomy level and the value encapsulated in the action taken, as our goal was to provide a proof of concept. The development of methods for determining the proper tradeoff between the two as a function of the domain the agent is operating it, can be highly beneficial and we hope will be addressed in future work. The method description given in Section 3 provides some general guidelines, pointing to how the change in weight is likely to affect performance. Some ideas we believe to be promising in that context are the use of decaying weight for the autonomy granted and bi-threshold-based approaches, though the effectiveness of these require extensive experimentation to evaluate.

ACKNOWLEDGMENTS

This research was partially supported by the ISRAEL SCIENCE FOUNDATION (grant No. 1162/17) and the ISF-NSFC joint research program (grant No. 2240/15).

REFERENCES

[1] George Alexander, Anita Raja, and David J Musliner. 2008. Controlling deliberation in a markov decision process-based agent. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 461–468.

[2] Shani Alkoby and David Sarne. 2017. The Benefit in Free Information Disclosure When Selling Information to People. In *Proceedings of the Thirty-First Conference on Artificial Intelligence (AAAI)*. 985–992.

[3] Amos Azaria, Zinovi Rabinovich, Sarit Kraus, Claudia V. Goldman, and Ya'akov Gal. 2012. Strategic Advice Provision in Repeated Human-Agent Interactions. In *Proceedings of the Twenty-Sixth Conference on Artificial Intelligence (AAAI)*.

[4] Amos Azaria, Ariella Richardson, and Avi Rosenfeld. 2016. Autonomous agents and human cultures in the trust–revenge game. *Autonomous Agents and Multi-Agent Systems* 30, 3 (2016), 486–505.

[5] Brigid Barron. 2003. When smart groups fail. *The journal of the learning sciences* 12, 3 (2003), 307–359.

[6] Pratyush Bharati and Abhijit Chaudhury. 2004. An empirical investigation of decision-making satisfaction in web-based decision support systems. *Decision support systems* 37, 2 (2004), 187–197.

[7] Darse Billings. 2006. *Algorithms and assessment in computer poker*. Ph.D. Dissertation. University of Alberta.

[8] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.

[9] Cody Buntain, Amos Azaria, and Sarit Kraus. 2014. Leveraging Fee-Based, Imperfect Advisors in Human-Agent Games of Trust.. In *AAAI*. 916–922.

[10] Gordon V Cormack et al. 2008. Email spam filtering: A systematic review. *Foundations and Trends® in Information Retrieval* 1, 4 (2008), 335–455.

[11] Celso M de Melo, Stacy Marsella, and Jonathan Gratch. 2018. Social decisions and fairness change when peopleâĂŹs interests are represented by autonomous agents. *Autonomous Agents and Multi-Agent Systems* 32, 1 (2018), 163–187.

[12] Avshalom Elmalech, David Sarne, and Barbara J. Grosz. 2015. Problem restructuring for better decision making in recurring decision situations. *Autonomous Agents and Multi-Agent Systems* 29, 1 (2015), 1–39.

[13] Avshalom Elmalech, David Sarne, Avi Rosenfeld, and Eden Shalom Erez. 2015. When Suboptimal Rules.. In *In proc. of AAAI*. 1313–1319.

[14] Gene H Golub, Michael Heath, and Grace Wahba. 1979. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics* 21, 2 (1979), 215–223.

[15] Michael A Goodrich, Dan R Olsen, Jacob W Crandall, and Thomas J Palmer. 2001. Experiments in adjustable autonomy. In *Proc. of IJCAI*. 1624–1629.

[16] Jones Granatyr, Nardine Osman, João Dias, Maria Augusta Silveira Netto Nunes, Judith Masthoff, Fabrício Enembreck, Otto Robert Lessing, Carles Sierra, Ana Maria Paiva, and Edson Emílio Scalabrin. 2017. The Need for Affective Trust Applied to Trust and Reputation Models. *ACM Comput. Surv.* 50, 4 (Aug. 2017), 48:1–48:36.

[17] Steve R Gunn et al. 1998. Support vector machines for classification and regression. *ISIS technical report* 14 (1998), 85–86.

[18] Galit Haim, Ya'akov Gal, Bo An, and Sarit Kraus. 2017. Human-computer negotiation in a three player market setting. *Artificial Intelligence* 246 (2017), 34–52. https://doi.org/10.1016/j.artint.2017.01.003

[19] Chen Hajaj, Noam Hazon, and David Sarne. 2017. Enhancing comparison shopping agents through ordering and gradual information disclosure. *Autonomous Agents and Multi-Agent Systems* 31, 3 (2017), 696–714.

[20] Chen Hajaj and David Sarne. 2017. Selective opportunity disclosure at the service of strategic information platforms. *Autonomous Agents and Multi-Agent Systems* 31, 5 (2017), 1133–1164.

[21] Roger Hallowell. 1996. The relationships of customer satisfaction, customer loyalty, and profitability: an empirical study. *International journal of service industry management* 7, 4 (1996), 27–42.

[22] Terrence Hendershott, Ryan Riordan, et al. 2009. Algorithmic trading and information. *Manuscript, University of California, Berkeley* (2009).

[23] Daniel Kahneman. 2000. A psychological point of view: Violations of rational rules as a diagnostic of mental processes. *Behavioral and Brain Sciences* 23, 5 (2000), 681–683.

[24] Rémy Kessler, Juan Manuel Torres-Moreno, and Marc El-Bèze. 2007. E-Gen: automatic job offer processing system for human resources. In *proc. of MICAZ*. 985–995.

[25] Wolfgang Ketter, Markus Peters, and John Collins. 2013. Autonomous Agents in Future Energy Markets: The 2012 Power Trading Agent Competition.. In *Proc. of AAAI*.

[26] Alexander Klos, Elke U Weber, and Martin Weber. 2005. Investment decisions and time horizon: Risk perception and risk behavior in repeated gambles. *Management Science* 51, 12 (2005), 1777–1790.

[27] Mario Konecki, Matjaš Gams, and Nikola Ivković. 2015. MIA: A multi-purpose intelligent assistant. In *Proc. of MIPRO*. 1109–1113.

[28] Priel Levy and David Sarne. 2016. Intelligent Advice Provisioning for Repeated Interaction.. In *Proc. of AAAI*. 842–849.

[29] Henry B Mann and Donald R Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics* (1947), 50–60.

[30] Alberto Messina, Roberto Borgotallo, Giorgio Dimino, D Airola Gnota, and Laurent Boch. 2008. Ants: A complete system for automatic news programme annotation based on multimodal analysis. In *Proc. of WIAMIS*. 219–222.

[31] Nadim Nachar et al. 2008. The Mann-Whitney U: A test for assessing whether two independent samples come from the same distribution. *Tutorials in Quantitative Methods for Psychology* 4, 1 (2008), 13–20.

[32] Bakhta Nachet and Abdelkader Adla. 2014. An agent-based distributed collaborative decision support system. *Intelligent Decision Technologies* 8, 1 (2014), 15–34.

[33] Gabriele Paolacci, Jesse Chandler, and Panagiotis G Ipeirotis. 2010. Running experiments on amazon mechanical turk. *Judgment and Decision Making* 5 (2010), 411–419.

[34] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[35] Isaac Pinyol and Jordi Sabater-Mir. 2013. Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review* 40, 1 (01 Jun 2013), 1–25.

[36] Daniel J Power and Ramesh Sharda. 2009. Decision support systems. In *Springer handbook of automation*. Springer, 1539–1548.

[37] Avi Rosenfeld and Sarit Kraus. 2009. Modeling Agents through Bounded Rationality Theories.. In *Proc. of IJCAI*. 264–271.

[38] Jordi Sabater and Carles Sierra. 2005. Review on computational trust and reputation models. *Artificial intelligence review* 24, 1 (2005), 33–60.

[39] Paul Scerri, David V Pynadath, and Milind Tambe. 2002. Towards adjustable autonomy for the real world. *Journal of Artificial Intelligence Research* 17, 1 (2002), 171–228.

[40] Jürgen Schmidhuber. 2015. Deep learning in neural networks: An overview. *Neural networks* 61 (2015), 85–117.

[41] Mark R Segal. 2004. Machine learning benchmarks and random forest regression. *eScholarship Repository. University of California.* (2004).

[42] Herbert A Simon. 1972. Theories of bounded rationality. *Decision and organization* 1, 1 (1972), 161–176.

[43] Evan Sultanik, Pragnesh Jay Modi, and William C Regli. 2007. On Modeling Multiagent Task Scheduling as a Distributed Constraint Optimization Problem.. In *Proc. ofIJCAI*. 1531–1536.

[44] Ioannis Tsamardinos, ME Pollack, et al. 1999. Adjustable Autonomy for a Plan Management Agent. In *Proc. of AAAI*.

[45] Ananth Uggirala, Anand K Gramopadhye, Brain J Melloy, and Joe E Toler. 2004. Measurement of trust in complex and dynamic systems using a quantitative approach. *International Journal of Industrial Ergonomics* 34, 3 (2004), 175–186.

[46] Yuxing Yan. 2014. *Python for Finance*. Packt Publishing Ltd.

[47] Shlomo Zilberstein. 2015. Building Strong Semi-Autonomous Systems.. In *Proc. of AAAI*. 4088–4092.