Problem restructuring for better decision making in recurring decision situations

Noname manuscript No.
(will be inserted by the editor)

Problem Restructuring for Better Decision Making in Recurring Decision Situations

Avshalom Elmalech · David Sarne · Barbara J. Grosz

Received: date / Accepted: date

Abstract This paper proposes the use of restructuring information about choices to improve the performance of computer agents on recurring sequentially dependent decisions. The intended situations of use for the restructuring methods it defines are website platforms such as electronic marketplaces in which agents typically engage in sequentially dependent decisions. With the proposed methods, such platforms can improve agents' experience, thus attracting more customers to their sites. In sequentially-dependent-decisions settings, decisions made at one time may affect decisions made later; hence, the best choice at any point depends not only on the options at that point, but also on future conditions and the decisions made in them. This "problem restructuring" approach was tested on sequential economic search, which is a common type of recurring sequentially dependent decision-making problem that arises in a broad range of areas. The paper introduces four heuristics for restructuring the choices that are available to decision makers in economic search applications. Three of these heuristics are based on characteristics of the choices, not of the decision maker. The fourth heuristic requires information about a decision-makers prior decision-making, which it uses to classify the decision-maker. The classification type is used to choose the best of the three other heuristics. The heuristics were extensively tested on a large number of agents designed by different people with skills similar to those of a typical agent developer. The results demonstrate that the problem-restructuring approach is a promising one for improving the performance of agents on sequentially dependent decisions. Although there was a minor degradation in performance for a small portion of the agents, the overall and average individual performance improved

A. Elmalech
Computer Science Department, Bar-Ilan University, Ramat-Gan 52900, Israel. E-mail: elmalech@cs.biu.ac.il

D. Sarne
Computer Science Department, Bar-Ilan University, Ramat-Gan 52900, Israel. E-mail: sarned@cs.biu.ac.il

B. J. Grosz
School of Engineering and Applied Science, Harvard University, Cambridge MA 02138, USA. E-mail: grosz@eecs.harvard.edu

substantially. Complementary experimentation with people demonstrated that the methods carry over, to some extent, also to human decision makers. Interestingly, the heuristic that adapts based on a decision-maker's history achieved the best results for computer agents, but not for people.

Keywords Decision Making · Sequentially Dependent Decisions · Platform Design · Experimentation

1 Introduction

Some decisions people face are choices among a clear set of options, for instance the choice of which jam to buy at the supermarket or between purchasing a high-wind umbrella or a small collapsible one. We will refer to this type of decision as a "decision simpliciter" because it is not adorned by any accompanying decisions. Other decisions come in clusters, for example the choice of wearing a black suit and black shoes rather than a brown suit with brown shoes. This paper considers a third type of decision, which we call "sequentially dependent decisions". Decisions of this type occur in sequences with decision n being affected by some of decisions $1...n-1$. For instance, a decision about whether to take an umbrella will affect the decision of whether to take a bus or taxi if it begins to rain while walking toward the bus stop. With sequentially dependent decisions, each decision exerts both immediate and long-term influence. As a result, the best choice at any point depends not only on the options at that point, but also on future conditions and the decisions made in them [53]. It is thus more difficult to define (and to compute) metrics for goodness of decision or decision-making performance.

A significant body of work on decisions simpliciter has shown a range of cognitive influences on people's decision making that lead them to make decisions that are "suboptimal" in the sense that their choices do not align with their preferences [9]. For instance, people may be overwhelmed by too many choices [76], and they do better with opt-out than opt-in policies [85,3]. Other work has shown that preference elicitation for such decisions is difficult [28]. A variety of work on autonomous agents suggests that even though they do not share people's cognitive limitations and may have sufficient computational capabilities to determine theoretically optimal strategies, they also make suboptimal decisions simplicter, whether as a matter of game theoretic issues [12] or systems design uninformed by theory [69] or both. The increasing presence of agents operating as proxies for people, not only for decisions simpliciter (e.g., bidding on eBay), but also for sequential decision making situations (e.g., algorithmic trading) makes the design of effective agent strategies for sequentially dependent decisions an important multi-agent systems challenge.

Interestingly, although people do not always maximize utility in individual decisions, a range of work has shown that they are expected monetary value (**EMV**) maximizers, in situations of repeated decisions simpliciter [88,43,5,42,49, 66,89,19,57,46], suggesting a preference for EMV maximization and the use of the EMV-maximizing strategy when the decision making environment enables them to recognize this solution. As sequentially dependent decisions involve a sequence of decisions, these results provide a basis for presuming they maximize EMV for such decisions as well. Research on people's strategies for sequentially dependent

decisions shows, however, that their strategies do not align with EMV maximizing ones either for individual or for repeated sequentially dependent decisions [50]. This literature argues that the EMV strategy is complex and unintuitive, claiming that people's deviation from the EMV model's predictions results from an inability to compute the solution rather than from the model inadequately representing the decision problem or their preferences [22]. As a result, we take the evidence from repeated decisions simpliciter as a basis for using EMV-maximization as the standard of optimality.

This paper focuses on repeated sequentially dependent decisions for three reasons. First, computer agents are typically constructed for situations that re-occur, that is, for repeated rather than for single use. Second, many decisions involving uncertain outcomes are made repeatedly. Examples of such repeated sequentially dependent decisions include both internet applications such as gambling in virtual casinos, shopping for holiday gifts, searching for research papers on a specific topic, and "physical" applications such as chess tournaments [55], driving (e.g., the repeated selection among routes) [5] as well as testing and repair of complex automated systems [45]. Third, as discussed above, the assumption of EMV-maximization as the standard measure of optimality is better justified for repeated decision making than for one-shot decisions. We note, importantly, that the approach and the methods we have developed may be used in non-recurring decision situations since they do not depend on the recurrence (or obviously on EMV). To measure their effectiveness in such situations, would, however, require some measure of optimality or utility elicitation.

The paper describes novel methods for improving agent performance on sequentially dependent decisions and empirically evaluates their effect on agent decision making and their usefulness for guiding people in making decisions in sequentially dependent decisionsettings. The prototypical context of use of these methods are website platforms for electronic marketplaces, such as `alibaba.com`, `made-in-china.com` and `gobizkorea.com` that connect buyers with numerous different sellers. The sellers and their products are listed, but typically with only very partial price information (e.g., a wide range of prices). To obtain actual prices, buyers must query sellers directly, which they typically do through links on the website. For some websites (e.g., `autotrader.com` and `Yet2.com`), buyers must query sellers for supplementary information about opportunities (e.g., about the used cars on Autotrader or inventions and patents Yet2).[1] These electronic marketplaces compete, and to attract customers to their site, they aim to incorporate features that improve customers experience. The restructuring methods we investigate are designed to provide such a capability. Thus, the improved agent performance is intended to arise not by changing the agent design (which is problematic in these settings for reasons we discuss below), but rather by providing an environment in which a variety of agent strategies embodied in autonomous agents coming to such sites will perform better.

The paper explores the hypothesis that restructuring of the options presented for sequentially dependent decisions would enable better performance by computer agents. The goal of restructuring is to lead a decision maker to a sequence

[1] The Yet2 marketplace operates as an online platform that allows "sellers" (industrial firms, entrepreneurial ventures, research universities and individual inventors) to post their inventions, while "buyers" can search the listed inventions [23].

of choices that more closely align with the optimal strategy. This approach parallels recent developments in psychology and behavioral economics [85,3,76,11,86] that restructure a decision problem rather than attempting to change a person's decision making strategy directly. For instance, this work has proposed removing options to allow people to focus on an appropriate set of choices and characteristics of those choices, which would suggest, for instance, that a used car buyer is likely reach a better decision more quickly if presented with a smaller set of possible cars and only the most important characteristics of those cars.

We take a restructuring approach rather than attempt to find a way to get agents to use the optimal strategy for two reasons. First, many multi-agent settings, including the eCommerce examples we use in our test set, involve agents designed by, and working for, a variety of organizations. The strategies these agents used are pre-set and cannot be changed by an external system. The second reason is rooted in the complexity of sequentially dependent decisions for which the set of preferences are complex (as described in Section 3) and the optimal strategy is non-intuitive. In such situations, explaining the optimal strategy requires substantial effort and it may not be possible to persuade non-experts in decision making or its correctness and optimality.

The restructuring we investigate includes elimination of a subset of the alternatives presented to the agent and changing the values of some characteristics of an alternative. In particular, we adjust certain characteristics of each option to compensate for possible reasoning biases. As we describe later, restructuring for sequentially dependent decisions has several complexities not present in decisions simpliciter. In particular, whereas for decisions simpliciter, a choice may be removed simply on the basis of its own value, that is not possible for sequentially dependent decisions. Future uncertainties may turn a presently unappealing choice into a more appealing one. For instance, an investor might invest in stocks or in bonds. If the stock market is bullish, then eliminating the bonds option would force him to invest in stocks. If a few days later the stock markets collapses, the elimination of bonds option may become detrimental.

We tested the restructuring approach to sequentially dependent decisions on economic search problems [4,32,33,58,73,90][2], which are search problems involving both uncertainty and costs of information gain. Economic search models represent well a variety of activities, including many in which agents are likely to participate and which have a recurring nature (e.g., a consumer searching for a product, a saver searching for an investment). In economic search, an agent needs to choose among several opportunities, each of which has an associated distribution of gains; to obtain the actual gain out of this distribution, the agent incurs a cost. In such problems, agents need to take into consideration the trade-off between the cost of further exploration and the additional benefit of knowing more values [4,16,41]. Unlike typical AI problems of exploration—exploitation tradeoffs (e.g., multi-armed bandit problems), in economic search, the exploration yields a specific value (rather than simply narrowing the distribution) and the end goal is choice of an opportunity (rather than of an entity with which to continue interacting). It is also distinguishable in having a closed-form solution.

The EMV-maximizing strategy for economic search problems has many inherent counter-intuitive characteristics [90] that make explaining it conceptually

[2] These problems are known in other literatures as directed search with full recall [90].

challenging. Prior research has shown that people's strategy for economic search problems does not align with the EMV-maximizing search strategy [73, 58, 32, 10]. Thus, these problems provide a good test environment for the restructuring approach to sequentially dependent decisions.

The paper presents four problem restructuring heuristics for recurring sequentially dependent decision problems, three fixed and one adaptive. Each fixed heuristic manipulates the opportunities presented to an agent, either modifying the set of possibilities presented to the agent or modifying the distribution of gains for opportunities. The "information hiding", heuristic eliminates alternatives based on the likelihood that they will not actually be needed by the theoretical-optimal strategy. The "mean manipulation" heuristic restructures the distribution of gains associated with each alternative in a way that leads an agent that is influenced only by means (rather than the distribution of gains) to follow (more closely) the optimal strategy. The "random manipulation" eliminates all the alternatives except one; it turns the sequentially dependent decision into a degenerate decision simpliciter (a consequence of removing all but one alternative), thus forcing choosing the EMV maximizing alternative from the set of opportunities viewed as a decision simpliciter options. The adaptive heuristic, denoted "adaptive learner", requires repeated sequentially dependent decision interactions with the same agent. It uses the results of the prior interaction history to model the decision-maker's strategy and classify it according to a set of pre-defined strategies. It then applies the fixed restructuring heuristic that best matches the agent's classification.

The results of our experiments on economic search problems show that the use of restructuring results in a sequence of choices that more closely resemble, and are better aligned with, those produced by the theoretical-optimal ones for the corresponding non-revised settings. For most agents, restructuring substantially improves the expected outcome. The heuristics vary, however, in the extent to which they improve the performance of different types of agents (in terms of EMV). With the "mean manipulation" heuristic, some agents attained the maximum possible improvement, whereas the performance of others declined substantially. Not surprisingly, this variation depended on the extent to which the agents' strategies were mean-dependent. The "information hiding" heuristic, in contrast, never yielded the maximum possible improvement for any agent, but the average improvement it produced was greater than for mean manipulation, and the maximum degradation it engendered was substantially smaller. The "random manipulation" heuristic improved the performance of some agents, but substantially worsened the performance of others. The "adaptive learner" heuristic produced the best results. Both the frequency of detrimental effect and its magnitude was substantially minimized when the system was able to classify an agent according to strategy and use this classification in choosing among heuristics. To determine whether these heuristics although designed for agents would also be helpful to people in sequentially dependent decision situations, we also ran experiments with human subjects, using the information hiding and adaptive heuristics. Both heuristics substantially improved people's performance. In this case, though, the information hiding heuristic outperformed the adaptive heuristic. This result suggests that either people are insufficiently consistent in their decision making to enable an effective classification of the strategies they use or that the strategies they use are very different from those with which they equip their agents.

This paper makes two significant contributions. First, it demonstrates the effectiveness of problem restructuring for recurring sequentially dependent decisions. Second, it defines heuristics for problem restructuring and demonstrates their usefulness for the economic search domain, as a proof of concept for the ability to apply problem restructuring processes that perform effectively in general and for specific individuals.

In the next section, we motivate the use of expected-value as a measure for assessing the performance on recurring sequentially dependent decision problems. Section 3 presents formally the economic search problem, its optimal (EMV-based) solution and the complexities associated with recognizing the optimality of this strategy. The four heuristics along with the motivation for using each one are described in Section 4, and the details of the principles used to evaluate them are given in Section 5. Section 6 provides an extensive analysis of the agents' strategies, and Section 7 summarizes the results. Section 8 surveys relevant literature from several fields. Finally, we provide our conclusions and directions for future research in Section 9.

2 Performance Metrics for Sequentially Dependent Searching

An agent's performance on a sequentially dependent decision problem cannot be measured solely by how close the final choice comes to matching a simple preference function, but must take into account the inherent uncertainty about outcomes throughout the sequence of choices. This section briefly reviews the use of expected utility for assessing decisions simpliciter and challenges to it as an appropriate metric for human decision-making, discusses differences between people's decision making strategies in repeated play (i.e., when they make the same decision repeatedly over many opportunities) from decisions simpliciter, and then argues for EMV maximization as an appropriate measure of optimality for recurring sequentially dependent decisions.

Utility theory, and in particular expected utility maximization, provides an elegant model of decision making under uncertainty, with many variants proposed as limitations have been revealed [72, 65, 24, 18, 56, 59]. Its appropriateness as model of people's decision-making behavior has, however, been challenged since the 1950s [75, 78] with systematic violations of the model being shown for such problems as the Allais paradox [1] and Samuelson's colleague example [70].

Research in psychology and behavioral economics has identified various cognitive explanations of the differences between people's behavior and the "best decisions" according to utility theory, with the argument that people are "satisficers" rather than "optimizers" [78] being perhaps the best well known among AI researchers. More recent work includes evidence people prefer decision strategies that favor winning most of the time [34]. Others have considered models where agents maximize an individual utility which partially depends on the welfare of other agents [35]. Currently, the main alternative to expected-utility maximization as descriptive theories of choice under uncertainty are theories that incorporate decision weights that reflect the impact of events on the overall attractiveness of uncertain choices (e.g., see [82] for a survey on non-expected utility descriptive theories of choice under uncertainty). Among these decision-weighting theories are sign-dependent theories, including prospect theory [37], and rank-dependent theo-

ries [63], with the main features of both incorporated later in cumulative prospect theory (CPT) [87]. Recent experimental studies have demonstrated that cumulative prospect theory more accurately matches/predicts people's behavior than expected utility maximization [83,54]. A variety of new models that more closely match people's decision simpliciter have been proposed [6,67,68].

Research on repeated decisions simpliciter, however, shows that people's decision making alters when they are asked to choose repeatedly rather than just once. Expected monetary value, a specialization of the expected-utility model in which utilities are linear in outcomes [72], measures the average amount that a strategy would yield if repeated many times. It has been shown that in repeated-play settings people's strategies asymptotically approach the EMV strategy as the number of repeated plays increases [88,46].[3] In particular, the probability a person will prefer the option associated with the highest expected value is substantially greater than in single-play settings [57]. Furthermore, repeated-play reduces such disparities between people's choices and utility-maximizing model predictions as possibility and certainty effects [43,89] and ambiguity aversion [55]. The tendency of people to use expected-value considerations in repeated-play has been explained as an "aggregation of risk" phenomenon [36], meaning that when decision makers consider investment opportunities as part of a collection of gambles they will be less risk averse, though without violating their own risk preferences.

Interestingly, the strategies peoples use in making repeated sequentially dependent decisions do not align with the EMV maximizing strategy. Explanations in the literature for this deviation differ from those for the analogous differences for individual decisions simpliciter. For decisions simpliciter, the deviation has been attributed to non-EMV related preference or choice models. In the case of SDDs, however, prior work attributes peoples failure to use an EMV-maximizing strategy to the complexity and unintuitive nature of this strategy. For example, Dudey and Todd [22] investigated peoples strategies for classical optimal online stopping problems like the secretary problem [25]. They argue that although applying the EMV-maximizing strategy is simple, its structure is not immediately obvious and its derivation is not simple. They argue further that it is unlikely people will derive and use the optimal rule for this type of search problem. Lee [50] suggested that the optimal decision rule for stopping problems is beyond peoples cognitive capabilities and thus requires people to employ heuristic solutions. We thus take the evidence from repeated decisions simpliciter as a basis for assuming that people are EMV-maximization seeking for SDDs and use this strategy as the measure of optimality in analyzing our experimental results.

3 Modeling Choice as Structured Exploration

As the underlying framework for this research, we consider the canonical sequential economic search problem described by Weitzman [90] to which a broad class

[3] For example, it has been shown that people's behavior tends to converge towards expected value maximization when repeatedly facing Allais type binary choice problems [43,5]. This phenomena is also reflected in Samuelson's Colleague's decision to accept a series of 100 gambles versus his refusal to accept only one [70]. Much evidence has been given to a general phenomenon according to which most human participants accept risky gambles with positive expected values when the gambles will be played more than once but reject the corresponding single gamble [42,66,89,19].

of costly exploration problems can be mapped. In this problem, a searcher is given a number of possible available opportunities $B = \{B_1, ..., B_n\}$ (e.g., each opportunity represents a different store selling a product the searcher is interested in) from which she can exploit only one. The value v_i to the searcher of each opportunity B_i (e.g., the price in the store, and more generally: expense, reward, utility) is unknown. Only its probability distribution function, denoted $f_i(v)$, is known to the searcher. For exposition purposes we assume values represent rewards and towards the end of the section discuss the straightforward transition to the case where values represent an expense. The true value v_i of opportunity B_i can be obtained but only by paying a fee (e.g., the expense of driving to the store), denoted c_i, which might be different for each opportunity. Once the searcher decides to terminate her search (or once she has uncovered the value of all opportunities) she chooses the opportunity with the maximum value from the opportunities whose values were obtained. A strategy s is thus a mapping of a world state $W = (q, B' \subset B)$ to an opportunity $B_i \in B'$, the value of which should be obtained next, where q is the best (i.e., maximum) value obtained by the searcher so far and B' is the set of opportunities with values still unknown. ($B_i = \emptyset$ if the search is to be terminated at this point.) The expected-benefit-maximizing sequential search strategy s^* is the one that maximizes the expected value of the opportunity chosen when the process terminates minus the expected sum of the costs incurred in the search.

The search problem as formulated herein applies to a variety of real-world search situations. For example, consider the case of looking for a used car. Ads posted by prospective sellers may reveal little and leave the buyer with only a general sense of the true value and qualities of the car. The actual value of the car may be obtained only through a test drive or an inspection, but these incur a cost (possibly varying according to the car model, location, and so forth). The goal of the searcher is not necessarily to end up with the most highly valued car, since finding that one car may incur substantial overall cost (e.g., inspecting all cars). Instead, most car buyers will consider the tradeoff between the costs associated with further search and the marginal benefit of a better-valued opportunity. Table 1 provides mappings of other common search applications to the model. In this paper we consider costly search problems of recurring nature. Applications of this type are many and indeed a large portion of our daily routine may be seen as executing costly search processes: searching our closet for an outfit for the day, opening the refrigerator and choosing what to cook, searching for a parking space, etc. In all these examples the search itself takes time and may consume resources, resulting in a tradeoff between the benefit from the potential improvement in the quality of the results we may further obtain and the costs of the additional exploration.

The sequential economic search problem has a relatively simple expected-benefit-maximizing solution. Nevertheless, despite its simplicity the solution is non-intuitive. A simplified version of an example from Weitzman [90] may be used to illustrate its non-intuitive nature. This problem involves a research department that has been assigned the task of evaluating two alternative technologies to produce a commodity. The benefits of each are uncertain and can only be known if a preliminary analysis is conducted. Since both technologies are used to produce the same commodity, eventually no more than one technology would actually be used. Table 2 summarizes the relevant information for decision-making. Technology β, if eventually used, will yield a total benefit of 100 with a probability of 0.5 and

Application	Goal	Opportunity	Value	Search cost	Source of uncertainty
Parking	Minimize time to park and walk to destination	Street to park in	Distance from destination	Time spent; fuel consumed	Uncertainty regarding the availability of parking in the street and exact distance of parking within street (if available)
Hitchhiking	Minimize time to get to destination	different cars offering rides	time it takes to get to destination (based on the car's planned route)	Time it takes till the next car shows up	Uncertainty regarding destinations of approaching cars
Product purchase	Minimize overall expense	Store	Product price	Time spent; communication and transportation expenses	Uncertainty regarding the sale price in each store

Table 1 The mapping of real-life applications to the sequential search problem.

Technology	β	ω
Cost	15	20
Reward	(0.5,100) , (0.5,55)	(0.2,240) , (0.8,0)

Table 2 Information for a simplified example.

a benefit of 55 with a probability of 0.5, whereas the alternative technology ω, if used, will deliver a benefit of 240 with a probability of 0.2 and with a probability of 0.8 will not yield any benefit. Research and development, which is mandatory for eliminating uncertainty, costs 15 for the β technology and 20 for ω.

The problem is to find a sequential search strategy which maximizes the expected overall benefit. Carrying out the required research and development only for β or only for ω is essentially better than not researching either technology. The expected value of researching β is $-15+[0.5(100)+0.5(55)] = 62.5$, whereas for ω it is $-20+[0.2(240)+0.8(0)] = 28$. Thus, at least one technology should be further explored. The next logical question is which technology should be researched first? When reasoning about which alternative to explore first, one may notice that by any of the standard economic criteria, β dominates ω: technology β has a lower development cost, a higher expected reward, a greater minimum reward and less variance. Consequently, most people would guess that β should be researched first [90]. However, it turns out that the optimal sequential strategy is to develop ω first and if its payoff turns out to be zero then develop β.

Figure 1 depicts the appropriate decision tree for the problem above. Suppose β is developed first. If the payoff turns out to be 55, it would then be worthwhile to develop ω, since the expected value of that strategy would be $-20+[0.2(240)+0.8(55)] = 72$, which is greater than the value of not developing ω at that point, i.e., 55. Even if β had a payoff of 100, then developing ω would be favorable, since the expected benefit would be $-20+[0.2(240)+0.8(100)] = 108$. Therefore the expected value of an optimal policy after first developing β would be $-15+[0.5(72)+0.5(108)] = 75$. A similar calculation for an optimal policy after first

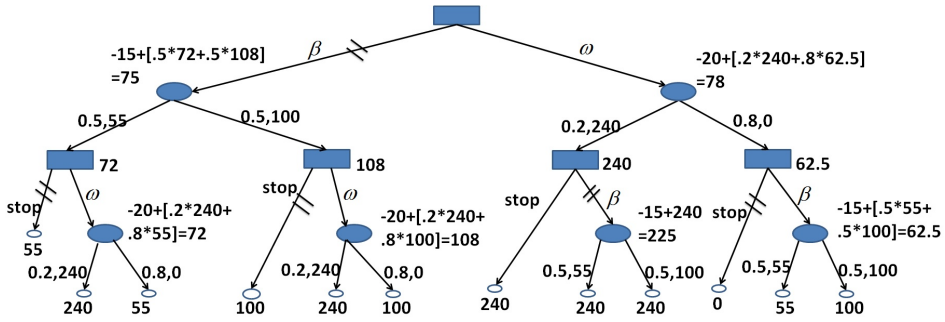


Fig. 1 Solution to a simplified example.

developing ω would be $-20 + [0.2(240) + 0.8(-15 + [0.5(100) + 0.5(55)])] = 78$. Thus, the optimal policy for this example has a counter-intuitive property whereby ω should be developed first.

Solving the sequential economic search problem using a decision tree requires considering all possible combinations of the opportunities whose value has not been obtained yet, the best value obtained so far and the opportunity which value needs to be reviewed next. Fortunately, it turns out that there is a simple way to extract the optimal solution for the sequential search problem. The solution is based on setting a reservation value (a threshold) denoted r_i for each opportunity B_i. The reservation value to be used should satisfy (see proof in [90]):[4]

$$c_i = \int_{x=r_i}^{\infty} (x - r_i) f_i(x) dx. \tag{1}$$

Intuitively, r_i is the value where the searcher is precisely indifferent: the expected marginal benefit from obtaining the value of the opportunity exactly equals the cost of obtaining that additional value. The searcher should always choose to obtain the value of the opportunity associated with the highest reservation value and terminate the search once there is no remaining opportunity associated with a reservation value greater than the highest value obtained so far (or all values have already been obtained).

Consider the example in Table 2. The reservation values of the two technologies can be extracted from (due to the discrete nature of the values): $c_\beta = \sum_{x \geq r_\beta}(x - r_\beta)P(x)$, resulting in $r_\beta = 70$, for β, and $c_\omega = \sum_{x \geq r_\omega}(x - r_\omega)P(x)$, resulting in $r_\omega = 140$, for ω. Therefore ω should be developed first and if its real value is below 70 (i.e., if zero) then β should be developed next.

One important and non-intuitive property of the above solution is that the reservation value calculated for an opportunity does not depend on the number and properties of the other opportunities, but rather on the distribution of the value of the specific opportunity and the cost of obtaining it. Another interesting characteristic of the optimal solution is the fact that it often favors risky opportunities (with low costs of obtaining their values) [90]. The explanation for this is that the great improvement in the opportunity value that can potentially be achieved

[4] The proof of optimality given in [90] holds also for the case where values are defined based on a discrete probability function $P_i(x)$, as in the example given above. In this case, the calculation of the reservation value r_i is given by $c_i = \sum_{x \geq r_i}(x - r_i)P_i(x)$.

outweighs the cost incurred, despite the relatively low probability of achieving its most beneficial values. These opportunities tend to have higher reservation values and hence need to be explored relatively early according to the optimal strategy.

Economic search problems have the exact characteristics of the type of problems we are interested in as described in the introduction: They are sequential in nature and the findings resulting from current decisions affect the choice of terminating the process or which opportunity to check in the future. They are widely applicable and people are likely to experience them in various forms on a daily basis. Furthermore, their solution has many non-intuitive properties. Nevertheless, as evidenced in the results section (and in prior literature [14, 69]), both people and the agents they program fail to follow the optimal strategy when engaged in sequential economic search. Supplying the theoretic-optimal solution for the economic search to the searcher is not always feasible. As discussed in former sections, agents are typically programmed with a pre-set strategy. As for people, they are likely not to trust the individual who gives them the optimal strategy (e.g., a passenger is often reluctant to accept a route suggested by a taxi driver; a house seeker often does not trust a real-estate broker's suggestions). It may require extensive effort to convince the searcher of the optimality of the proposed solution. A possible way to persuade a searcher that a strategy for the economic search problem is optimal is by giving her the optimality proof, but in our case that is relatively complex and requires strong mathematical background.[5] Another possible way to persuade an individual that a strategy is optimal is to calculate the expected value of every possible sequence of decisions and compare it with the expected outcome of the optimal strategy. However, in our case the number of possible sequences for which the expected outcome needs to be calculated is theoretically infinite in the case of continuous value distributions or exponential (combinatorial) for discrete probability distribution functions. A sequence defines the opportunity explored at each stage given the set of values received for previously explored opportunities. For N opportunities and V possible values upon which the distribution of values is defined, there are $O(N!) * |V|^N$ such sequences. In this case the strategy cannot even be expressed as its representation is exponential (unless we use the set of reservation values to represent the strategy, which, again, needs to be initially argued). Thus, both these methods for proving optimality have substantial overhead and require much effort in the case of economic search. In contrast, the problem restructuring approach can improve performance without requiring such overhead. The use of restructuring is completely transparent to the searcher, and does not require any direct interaction with her.

Before continuing, we present the problem in its dual expected-expense-minimizing version, on which our experimental design is based. In this case, opportunity values represent a payment, and the searcher attempts to minimize the expected overall expense, defined as the expected value of the opportunity chosen when the process terminates plus the expected sum of the costs incurred in the search. The optimal strategy in this case is to set a reservation value r_i for each opportunity B_i, satisfying:

$$c_i = \int_{x=-\infty}^{r_i} (r_i - x) f_i(x) dx \tag{2}$$

[5] The proof of optimality as given in [90] is four pages long and extensively uses mathematical manipulations.

The searcher should always choose to obtain the value of the opportunity associated with the minimum reservation value and terminate the search once there is no remaining opportunity associated with a reservation value lower than the minimum value obtained so far (or all values have already been obtained).

4 Problem Restructuring Heuristics

In this section, we define four problem-restructuring heuristics which we used in our investigations: Information Hiding, Random Manipulation, Mean Manipulation and Adaptive Learner. The first three assume that no prior information about the searcher is available and apply a fixed set of restructuring rules (differing in the information they present to the searcher). The fourth heuristic uses information from a searcher's prior searches to classify her and decides which of the other three heuristics to use. We emphasize that the need to initially supply the decision-maker the full set of alternatives available to her precludes trivial solutions that force the searcher to follow the optimal sequence.[6]

For a restructuring heuristic to be considered successful, it needs not only to improve the overall (cross-searchers) performance, but also to avoid significantly harming the individual performance of any searchers. It is notable that the removal of an opportunity or its change is risky, because if the opportunity is needed eventually, its absence degrades performance. Similarly if the opportunity is required in its original form, the searcher will fail to identify it as such. Still, as reflected by the results reported in the following sections, an intelligent use of the restructuring heuristics manage to substantially improve overall performance with only very few searchers suffering a slight performance degradation individually.

4.1 The Information Hiding Heuristic

This heuristic removes from the search problem opportunities for which the probability that they will need to be explored according to the optimal strategy s^* is less than a pre-set threshold α. By removing these opportunities, we prevent the searcher from exploring them early in the search, yielding a search strategy that is better aligned with the optimal strategy in the early, and more influential, stages of the search. While the removal of alternatives is likely to worsen the performance of searchers that use the optimal strategy, the expected decrease in performance is small; the use of the threshold guarantees that the probability that these removed opportunities would actually be required if the optimal search were to be conducted will remain relatively small. Formally, for each opportunity B_i we calculate its reservation value, r_i, according to Equation 2. The probability of needing to explore opportunity B_i according to the optimal strategy, denoted P_i, is the probability that all the opportunities explored prior to B_i (i.e., those associated with a reservation value lower than r_i) will yield values greater than

[6] For example, at each stage of the search disclosing to the searcher the opportunity that needs to be explored next according to optimal search strategy and terminate the process (e.g., by disclosing an empty set) when the value obtained so far is below the lowest reservation value of the remaining opportunities.

r_i, thus it is given by $P_i = \prod_{r_j \leq r_i} P(v_j \geq r_i)$. The heuristic therefore omits every opportunity B_i $(i \leq n)$ for which $P_i \leq \alpha$ from the problem instance.

4.2 The Mean Manipulation Heuristic

This heuristic attempts to overcome people's tendency to overemphasize mean values, reasoning about this one feature of a distribution rather than the distribution itself more fully [58]. When relying on mean values, searchers typically choose to explore the opportunity in which the expected net value is minimal (and lower than the best (i.e., lowest) value obtained so far). We denote this search strategy the "naive mean-based greedy search". Formally, denoting the mean of opportunity B_i by μ_i, searchers using the naive mean-based greedy strategy calculate the value $w_j = \mu_j + c_j$ for each opportunity $B_j \in B$ and choose to explore the value of opportunity $B_i = argmin_{B_j}\{w_j | B_j \in B^{'} \cap w_j \leq v\}$ where v is the lowest value obtained so far (and terminate the exploration when $B_i = \emptyset$).

The heuristic restructures the problem in a way that the value w_i, which a searcher using the naive mean-based greedy search will assign each opportunity B_i in the restructured problem, will be equal to the reservation value r_i calculated for that opportunity in the original problem. This ensures that the choices made by searchers that use the naive mean-based greedy search strategy for the restructured problem are fully aligned with those of the optimal strategy for the original problem. The restructuring is based on assigning a revised probability distribution function $f_i^{'}$ to each opportunity B_i $(0 < i \leq n)$, such that $w_i^{'} = r_i$, where r_i is the reservation value calculated according to Equation 2. The restructuring of f_i is simple, as it only requires the allocation of a large mass of probability around $\mu_i^{'}$ (the desired mean which satisfies $w_i^{'} = r_i$). The remaining probability can be distributed along the interval such that $\mu_i^{'}$ does not change.

4.3 The Random Heuristic

This heuristic is ideal for searchers who do not really see any benefit in exploring. These searchers choose one opportunity and terminate the search immediately thereafter. The opportunity picked can be the first, the last or any other in the set (including a random selection) or the one that complies with pre-defined criteria, e.g., the one with the lowest net expectancy (value plus exploration cost). This has evidence in experimental economics where it has been shown that some people stop after one observation. For example, some automobile shoppers are reported to check (and buy from) one dealer only [44] and some proportion of consumers buy from the first store they visit [20]. In such cases, where the searcher does not see any benefit in exploring, the ability to improve performance through problem restructuring is limited since the heuristic can merely leave only the choice associated with the lowest net expectancy ($B_i = argmin_{B_j}\{\mu_j + c_j | B_j \in B'\}$) in the set of choices (opportunities), and remove the remaining ones.

4.4 The Adaptive Learner Heuristic

The adaptive learner heuristic attempts to classify the strategy of a searcher and uses this classification to determine the best problem restructuring method to apply from a pre-defined set. We use "class-representing agents" for classification purposes. Each class-representing agent employs a strategy that best represents strategies of a specific class. The class-representing agent can thus be used with any problem instance, returning as an output the sequence used and the accumulated cost it obtained. For some cases, the class-representing agent may have several typical behaviors and thus will return a vector of possible outputs. For example in the class of searchers that make a single selection only, the selection of any single opportunity equally represents the class. In this case n results will be received as output, each correlated with picking another opportunity. A searcher will be classified as belonging to a class based on the similarity between her search and the search exhibited by the class representing agent for the same problem instances (in a "nearest neighbor"-like classification). The measure of similarity we use is the achieved performance over the same set of problems. Specifically, the searcher is classified as belonging to the class for which the relative difference between the searcher's performance and its representing-agent's performance is minimal, and below a threshold γ. Otherwise, if the minimal relative difference is above the threshold, the searcher is classified as belonging to a default class. Once the searcher is classified, the restructuring heuristic that is most suitable for that class can be applied. The use of the threshold γ assures that for any searcher that cannot be accurately classified, a default restructuring heuristic is used, one that guarantees that the possible degradation in the performance of the searcher is minor.

Algorithm 1 Classifier for Adaptive Learner

Input: O - Set of prior problem instances and the searcher's performance in each instance.
S - Set of class representing agents.
$Threshold$ - classification threshold.
Output: s^* - the classification of the searcher ($null$ represents no classification or no previous data).

Initialization: $d_s \leftarrow 0 \; \forall s \in S$
for every $s \in S$ **do**
  for every $o \in O$ **do**
    $d_s \leftarrow d_s + min_i \left(\frac{|Performance_{searcher}(o) - Performance_s(o)[i]|}{Performance_s(o)[i]} \right)$
  end for
end for
if $min \left(\frac{d_s}{|O|} | s \in S \right) \leq Threshold$ **then**
  return $argmin_s(d_s)$
else
  return $null$
end if

Our adaptive learner heuristic's classifier is given in pseudo-code in Algorithm 1. It receives a set of prior problem instances, O, which also contains the searcher's performance (i.e., its overall cost) in each instance and a set S of strategy classes as input. The function $Performance_s(o)$ returns a vector in which each element is a possible performance of the class-representing agent $s \in S$ when given the

problem instance o.[7] The function $Performance_{searcher}(o)$ is the performance of the searcher being classified when encountered problem o (this value is part of the input in O). For each strategy class s the classifier calculates the relative difference between the performance of the representative agent of that class and the performance of the searcher that needs to be classified (Step 4). In case the representing agent returns a vector, as discussed above, it uses the minimum difference between $Performance_{searcher}(o)$ and any of the vector elements. This is repeated for any instance $o \in O$ (Steps 3-5). The algorithm then returns (Step 8) the class to which the searcher belongs (or *null*, if none of the distance measures are below the threshold set). Based on the strategy returned we apply the restructuring heuristic which is most suitable for this strategy type (or the default restructuring if *null* is returned).

The results we report in this paper for the adaptive learner heuristic are based on a set S of three strategy classes: optimal strategy, naive mean-based greedy search strategy and random strategy. We use the optimal strategy to represent the class of strategies that are better off without any problem restructuring. It is noteworthy that despite having the optimal strategy as being this class representative, none of the agents that were evaluated used this strategy, as reported in Section 6. Nonetheless, when it comes to representing strategies that will suffer from input restructuring, the optimal strategy is a natural pick. For a searcher that cannot be classified as belonging to one of these three strategies, we use the information hiding restructuring heuristic. To produce the functionality $Performance_s(o)$ required in Algorithm 1, we developed the following three class-representing agents:

- *Optimal Agent*. This agent follows the theoretical-optimal search strategy described in Section 3.
- *Mean-based Greedy Agent*. This agent follows the naive mean-based greedy search strategy described in 4.2.
- *Random Agent*. This agent picks one opportunity and then terminates its search as described in 4.3. Since there are numerous possible selection rules for picking a single opportunity, this agent returns a vector of size $|o|$ where the i-th element gives the performance of picking only the i-th opportunity.

The more observations of prior searcher behavior given to the classifier the better the classification it can produce, and consequently the better the searcher's performance is likely to be after the appropriate restructuring is applied. Obviously, an even greater improvement in performance could be obtained if the searcher's decision-making logic were available (i.e., the agent's design and code in the case of an agent searcher) rather than merely results of prior searches. In this case, by having the agent execute the search using each of the restructuring heuristics, the one with which the agent performs best would be immediately revealed. Nevertheless, in most cases, having access to the agent's code is inapplicable. Consequently, the most the heuristic can count on is the searcher's performance in prior searches.

[7] While most representing agents return a single performance output, some representing agents (i.e., the agent representing the class of searchers that are satisfied with a single selection, as described above) return a vector of possible outcomes.

5 Empirical Investigation of Restructuring

In this section we describe the principles of the agent-based and human-based methodology used for this research. We continue by presenting the two domains used for the empirical investigation of our approach. We then describe the four sets of problems and the measures that were used to evaluate the performance of our heuristics. Finally we present a detailed description of the two populations and the experiments used to assess the effectiveness of input restructuring in our domains.

5.1 Agent-Based and Human-Based Methodology

We tested the effectiveness of the general approach and the specific restructuring heuristics with both agents and people. The use of agents in this context offers several important advantages. First, from an application perspective, the ability to improve the performance of agents in search-related applications and tasks, especially in eCommerce, could significantly affect future markets. The importance and role of such agents have been growing rapidly. Many search tasks are delegated to agents that are designed and controlled by their users (e.g., comparison shopping bots). Many of these agents use non-optimal strategies. Once programmed, their search strategy cannot be changed externally, but it can be influenced by restructuring of the search problem. Second, from a methodological perspective, the use of agents enables the evaluation to be carried out over thousands of different search problems, substantially improving the statistical quality of the result. Third, the use of agents eliminates human computational and memory limitations as possible causes of inefficiency. The inefficiency of the agents' search is fully attributable to their designs, reflecting agent designers' limited knowledge of how to reason effectively in search-based environments rather than cognitive processing.

The experiments with people as decision-makers complement the experimentation with agents in several respects. First, it can validate or invalidate the effectiveness of different heuristics, which have been proven to be useful in extensive experimentation with agents, when applied to people. The level of similarity between peoples' strategies and agents' strategies is not conclusive; some work suggests a relatively strong correlation between the behaviors of the two [14], while in other work the agents have been reported to act different than people to some extent [26, 69]. Second, it is notable that while agents can be programmed only by people with programming skills, the experiments with people as decision makers enable testing the approach with the general population. Finally, while agents tend to use a consistent strategy, people often tend to alternate between different strategies and rules of thumb [84, 81]. This imposes great challenges on heuristics such as the adaptive learner, and hence the evaluation with people is crucial.

5.2 Domains for Empirical Evaluation

To evaluate the four heuristics and our hypothesis that performance in search-based domains can be improved by restructuring the search problem we used two different domains. The first, called the "repairman problem", is more natural

Domain	Opportunity B_i	Value distribution $f_i(x)$	Search cost c_i
Job assignment	a server	server's queue length	time it takes to query the server to learn about its current queue length
Repairman problem	repairman	charge requested by repairman	cost of querying the repairman to learn about the requested charge

Table 3 Mapping of the two domains used in the experiments to the sequential search problem.

for human decision-makers as it relates to a somewhat "daily" search task. The second, called "job-assignment", was used for agents, as it is based on a computer environment and seems more "natural" for agents that strategy developers develop.

Job-assignment is a classic server-assignment problem in a distributed setting that can be mapped to the general search problem introduced in this paper. The problem considers the assignment of a computational job for execution to a server chosen from a set of N homogeneous servers. The servers differ in the length of their job queue. Only the distribution of each server's queue length is known. To learn the actual queue length of a server, it must be queried, which is a time consuming task (server-dependent). The job can eventually be assigned only to one of the servers queried (i.e., a server cannot be assigned unless it has first been queried). The goal is to find a querying strategy that minimizes the overall time until the execution of the job commences. The mapping of this problem to the sequential search problem in its cost minimization variant is straightforward (see Table 3). Each server represents an opportunity where its queue length is its true value and the querying time is the cost of obtaining the value of that opportunity.

The "repairman problem" was used for experiments with human decision-makers. In the "repairman problem" the searcher considers N repairmen who offer a service the searcher needs. The charge (price) of each repairman is associated with some distribution of values. In order to reveal a repairman's price, the searcher must pay a query fee. This problem is a cost minimization problem, as the searcher attempts to minimize her total expense, which is the sum of the querying fees and the price paid.

5.3 Tested Sets

We used four different sets of problems to evaluate the restructuring heuristics. To simplify the search problem representation, distributions in all sets were formed as multi-rectangular distribution functions. In multi-rectangular distribution functions, the interval is divided into n sub intervals $\{(x_0, x_1), (x_1, x_2), .., (x_{n-1}, x_n)\}$ and the probability distribution is given by $f(x) = \frac{p_i}{x_i - x_{i-1}}$ for $x_{i-1} < x < x_i$ and $f(x) = 0$ otherwise, $(\sum_{i=1}^{n} P_i = 1)$. The optimal search strategy for the problem with multi-rectangular distribution functions is given in the appendix. The benefit of using a multi-rectangular distribution function is its simplicity and modularity, in the sense that any distribution function can be modeled through it with a small number of rectangles. The first set of problems consisted of 5000 problems, each generated as follows:

- Number of opportunities - randomly picked from the range $(2, 20)$.
- Search cost of each opportunity - randomly picked from the range $(1, 100)$.
- Distribution function assigned to each opportunity - generated by randomly setting an interval and probability for each rectangle and then normalizing all rectangles so that the distribution is properly defined over the interval $(0, 1000)$.
- Number of rectangles in a distribution - randomly picked from the range $(3, 8)$.

This set was used as the primary set for experiments with agents. The choice of the specific ranges was made to enable a rich set of problem variants: the cost of searching was between one to three orders of magnitudes smaller than the maximum value of an opportunity, and the number of opportunities substantially varied.

The second and the third sets of problems also contained 5000 problems and were generated with different distributions of values and querying costs, as follows:

- Set 2: "Increased Costs". This set differed from the first set of problems in the interval from which opportunities' search costs were drawn, which in this case was $(1, 300)$, resulting in an increased ratio between the search cost and possible values of different opportunities. This increase in search costs should have made searchers more reluctant to resume their search at any point.
- Set 3: "Increased Variance". This set differed from the first set of problems in the sense that the multi-rectangular distribution function in this case was defined over the interval $(1000, 10000)$, resulting in a substantial increased variance in opportunity values. This change should have made searchers more eager to resume their search at any point.

The fourth set contained 100 different problems that were used primarily for the experiments with people as decision makers. The problems were generated in a manner similar to the procedure described above with minor adjustments. First, since it is difficult for people to handle a large number of opportunities, the set contained a fixed number of 8 opportunities ("repairmen"). Second, the multi-rectangular distribution function (which was still defined over the interval $(0, 1000)$) now had only four rectangles. This is mainly because it is difficult for people to digest a complex distribution with a large number of rectangles.

5.4 Performance Measures

Two complementary measures were used to evaluate the different heuristics: (1) relative decrease in the overall costs; (2) relative reduction in search inefficiency. Formally, we use $t_{opt}(o)$ to denote the expected overall cost of the optimal search strategy for problem $o \in O$ and $t_{rst}(o, s)$ and $t_{\neg rst}(o, s)$ to denote the total costs of a searcher $s \in S$ with the restructured and non-restructured versions of problem respectively (where O is the set of problem instances and S is the set of searchers evaluated).[8] The first measure, calculated as $(t_{\neg rst}(o, s) - t_{rst}(o, s))/t_{\neg rst}$, relates directly to the cost saved, which depends on the problem instance o, since $t_{\neg rst}(o, s)$ can vary substantially. The second measure, calculated as $(t_{\neg rst}(o, s) - t_{rst}(o, s))/(t_{\neg rst}(o, s) - t_{opt}(o))$, takes into account that the overall search cost

[8] The notation S is thus augmented to consider any set of searchers, rather than just the class-representing agents as before.

using either the restructured or original data is bounded by the performance of the optimal agent. It thus highlights the efficiency of the heuristic in improving performance.

The above measures are applicable to a specific problem instance and specific searcher (either agent or person). The aggregative equivalents of these measures that apply to the general population are defined as:

$$\text{social performance improvement}= \frac{\sum_{s\in S}\sum_{o\in O} t_{\neg rst}(o,s) - \sum_{s\in S}\sum_{o\in O} t_{rst}(o,s)}{\sum_{s\in S}\sum_{o\in O} t_{\neg rst}(o,s)}. \quad (3)$$

$$\text{social reduction in search inefficiency}= \frac{\sum_{s\in S}\sum_{o\in O} t_{\neg rst}(o,s) - \sum_{s\in S}\sum_{o\in O} t_{rst}(o,s)}{\sum_{s\in S}\sum_{o\in O} t_{\neg rst}(o,s) - |S|\sum_{o\in O} t_{opt}(o)}. \quad (4)$$

These two measures apply directly to the overall change in the performance of the population as a whole (i.e., the social change). A negative value for either measure represents an increase in the overall search cost (in the case of (3)) or in the search inefficiency (in the case of (4)).

Two additional measures were used for agents:

$$\text{individual performance improvement}= \frac{\sum_{o\in O}(t_{\neg rst}(o,s) - t_{rst}(o,s))}{\sum_{o\in O} t_{\neg rst}(o,s)}. \quad (5)$$

$$\text{individual reduction in search inefficiency}= \frac{\sum_{o\in O}(t_{\neg rst}(o,s) - t_{rst}(o,s))}{\sum_{o\in O} t_{\neg rst}(o,s) - \sum_{o\in O} t_{opt}(o)}. \quad (6)$$

The latter measures apply to the average individual agent improvement (cross-problems). The reason for calculating these measures only for agents is two-fold: first, the human subjects were limited by the number of search problems in which they could engage (e.g., due to the experiment's length and the need to stay focused for relatively long time). Second, as we discuss later in further detail, human subjects were used in "between-subjects" design, hence no individual person had both $t_{\neg rst}$ and t_{rst} results. Based on (5) and (6) we can also define the cross-agents average individual improvement, calculated as:

$$\text{average individual performance improvement}= \frac{1}{|S|}\sum_{s\in S}\frac{\sum_{o\in O}(t_{\neg rst}(o,s) - t_{rst}(o,s))}{\sum_{o\in O} t_{\neg rst}(o,s)}. \quad (7)$$

$$\text{average reduction in individual search inefficiency}= \frac{1}{|S|}\sum_{s\in S}\frac{\sum_{o\in O}(t_{\neg rst}(o,s) - t_{rst}(o,s))}{\sum_{o\in O} t_{\neg rst}(o,s) - \sum_{o\in O} t_{opt}(o)}. \quad (8)$$

For each evaluated heuristic the maximum decrease in individual average performance was recorded. This is due to the fact that an important requirement for a successful heuristic is that it does not substantially worsen any of the agents' individual performance.

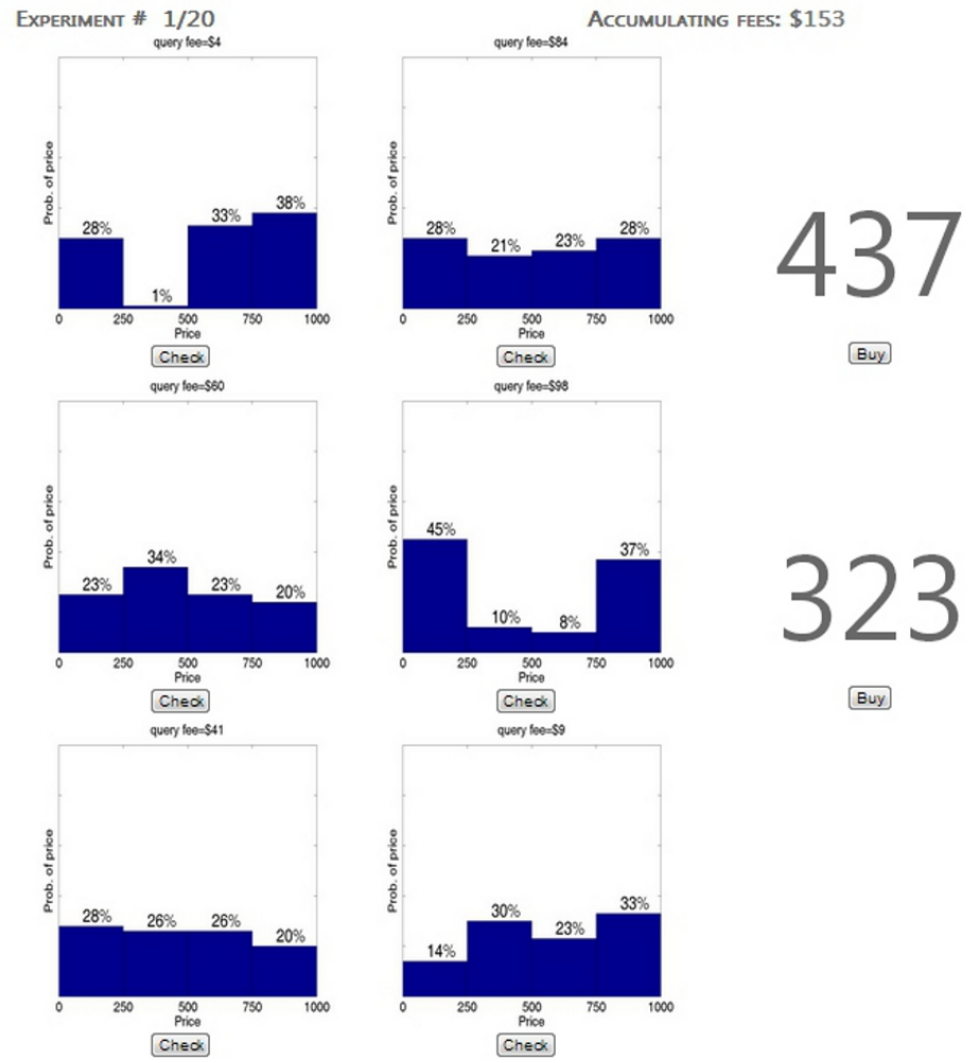
5.5 Evaluating the Usefulness of Heuristics for Agents

The evaluation used agents designed by computer science students in a core Operating Systems course. While this group does not represent people in general, it reasonably represents future agent developers who are likely to design the search logic for eCommerce and other computer-aided domains. As part of a regular course assignment, each student programmed an agent that received a list of servers with their distribution of waiting times and their querying costs (times) as input. The agents had to query the servers (using a proxy program) to learn their associated waiting time and then choose one of these queried servers to execute a (dummy) program. This search problem expanded an existing assignment that was focused on probabilistic assignment, making it a more complete assignment. The students were told that: (a) the agent would be run repeatedly with 5000 different problem instances generated according to the guidelines described in 5.3; (b) the agent's performance would be determined as its average expected-expense in all runs; and (c) their grade for the strategy part of the assignment would be determined based on their agents performance, linearly. As part of their assignment, students provided documentation describing the algorithms they used to search for a server.

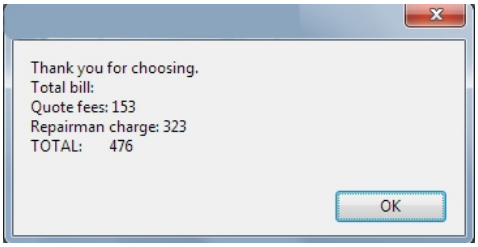
An external proxy program was used to facilitate communication with the different servers. The main functionality of the proxy was to randomly draw a server's waiting time, based on its distribution, if queried, and to calculate the overall time that had elapsed from the beginning of the search until the dummy program began executing (i.e., after assigned to a server and waiting in that server's queue). Each agent in our experiments was evaluated using all problems from all four sets.

5.6 Evaluating the Usefulness of Heuristics for People's Searches

The evaluation infrastructure for the experiments with people as decision makers was developed as a JavaScript web-based application, emulating an exploration problem with 8 repairmen, each associated with a different distribution of service fees (repairman's charge) and a cost for obtaining the true price (represented as a "query fee", for querying a repairman). Figure 2 (top panel) presents a screenshot of the system's GUI. In this example, the fee distribution of six repairmen is represented by the bar-graphs and for the other two repairmen the actual fee is given (i.e., already revealed). Querying a repairman is done by clicking the "Check" button below its distribution, in which case the true value of the repairman becomes known and the query fee of that repairman is added to the accumulated cost. The game terminates when clicking the "Buy" button, which is available only for repairmen whose values have been revealed. After "buying" the subject is shown a short summary of the overall expense, divided into the accumulated exploration cost and the amount paid for the service itself (see buttom panel in Figure 2).



(a)



(b)

Fig. 2 Screenshots of the system designed for the experiments with people as decision-makers.

The subjects were recruited using Amazon's Mechanical Turk service [2].[9] When logged in, the subjects received a short textual description of the experiment, emphasizing the exploration aspects of the process, its recurring nature and the way costs are accumulated. Next, they had to play a series of practice games in order to make sure they understood the above. Participants had to play at least three practice games; they could continue practicing until they felt ready to start the experiment. After completing the practice stage participants had to pass a qualification test, making sure they understand the recurring nature of the game and that performance will be based on the average achieved. Then, participants were told they were about to play the 20 "real" games. At this point the system randomly drew 20 problems from its fourth problem set to be played sequentially. To prevent the carryover effect, a "between subjects" design was used, i.e., the same restructuring was used in all of the 20 problems presented to a given subject.[10] To motivate players to exhibit efficient exploration behavior, and possibly also extend their practice session, they were told that they will receive a bonus that linearly depends on their average expense over the 20 games. As a means of precaution, the time it took each participant to make each selection and the overall time of each game played was logged, and the results of participants with unusual low times were removed from the analysis. Overall, 120 people participated in our experiments; 40 played in a non-restructured environment, 40 played with the information-hiding restructuring heuristic and 40 played with the adaptive learner restructuring heuristic.[11] The reason the mean manipulation heuristic was not included in the experiments with people is that people (unlike agents) are likely to fail in calculating the mean values, thus this heuristic is likely to be much less effective than others. As for the random manipulation, the performance of this heuristic with people does not require designated experimentation, as the performance $t_{rst}(o, s)$ is subject-independent (since there is only one opportunity presented to the searcher) and the performance $t_{\neg rst}(o, s)$ is recorded in any case as part of the experimental design. We would like to emphasize that while not tested "stand-alone", both the mean manipulation and the random manipulation heuristics were implemented as part of the adaptive learner heuristic and used for its evaluation with people.

6 Strategy Analysis

In this section we present and analyze the different strategies that were embedded in the agents by the strategy designers. The analysis reveals great variance both in the strategy structure and the parameters that the strategies used. In particular, all of the strategies differed from the optimal strategy. The analysis of people's strategies is beyond the scope of this paper for two main reasons. First, while asking people to document or express their search strategy as part of the experiment is possible, relying on these statements is very problematic. Evidence of discrepancies

[9] For a comparison between AMT and other recruitment methods see [61].

[10] Of course the between-subject design raises the question of wiping out the individual heterogeneity in search behavior that can be quite large. Still, if we used a within-subject design we would have been affected by learning.

[11] Since each participant in AMT has a unique ID, connected to a unique bank account, it is possible to block the same ID from participating more than once in a given experiment.

Agent ID	mean	variance	median	cost	random	subset	accumulated cost	probability
1-6,17	v			v				
7,11,12, 14-16,18, 19,24-26, 28,29,30	v			v		v		
59-61					v	v		
36-38	v					v		
39-58,62-63,65-67						v		
8,9,10	v			v			v	
13,21,22	v	v		v				
20,31	v	v		v		v		
23	v			v				v
27	v	v				v		
68-71	v			v	v	v		
32,33	v				v			
34,35			v	v		v		
64					v			
72							v	v

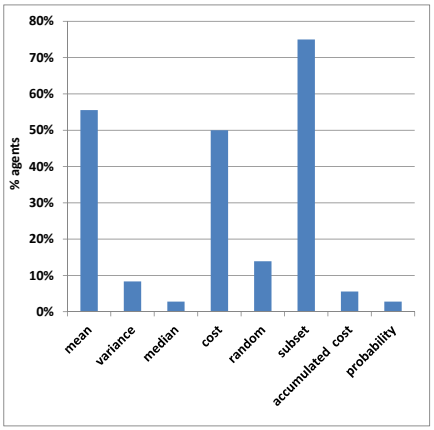


Fig. 3 Strategy characteristics according to the manual classification (based on code and documentation). The right graph depicts the percentage of agents that used in their strategy each characteristic.

between actual and reported human behavior is a prevalent theme in research [30, 7, 40]. Second, reverse engineering of the collected data into a strategy is infeasible in our case due to the richness of the problems used and the limited number of observations collected for each person, as well as the large amount of different possible behaviors.

Overall, we had 72 agent strategies, each developed by a different person. The strategies used for the agents reveal several characteristics along which agent designs varied in our population of programmers. In this section, we use two techniques to classify the agents' strategy-characteristics. The first is based on manual classification of agents according to search strategy characteristics reflected in the documentation that was supplied by the strategy designers; the second technique is based on the agents' performance. The manual classification (based on documentation and code review) revealed substantial diversity in the agents' strategies that is mostly captured by the following essential characteristics (see Figure 3 for the level of use of each characteristic):[12]

- "mean", "variance", and "median" - correspond to taking into consideration to some extent the expected value, the variance and the median value of each server, respectively, in the agent's strategy.
- "cost" - indicates taking into consideration the time (cost) of querying servers.
- "random" - indicates the use of some kind of randomness in the decision making process of the agent.
- "subset" - indicates a preliminary selection of a subset of servers for querying.
- "accumulated cost" - indicates the inclusion of the cost incurred so far (i.e., "sunk cost") in the decision making process.
- "probability" - indicates the use of the probability of finding a server with a lower waiting time than the minimum found so far in the agent's considerations.

Several interesting observations can be made based on the use of these characteristics, as reflected in Figure 3. First, many of the agents use the mean waiting

[12] See Appendix for a detailed list of some of the more interesting strategies used.

time of a server as a parameter that directly influences their search strategy, even though the optimal strategy is not directly affected by this parameter (cf. Section 3). This is also the case with the variance and median. In particular, the use of randomization and the accumulated cost as a consideration has no correlation with and plays no role in the optimal strategy. Finally, a substantial number of agents (34 of the 72) do not take into account the cost of search in their strategy. One possible explanation for this phenomena is that the designers of these strategies considered cost to be of very little importance in comparison to the mean waiting times.

We emphasize that the dominance of the mean-use characteristic in the agents' strategies can be considered an evidence for the tendency of people towards EMV-maximization. More than 55% of the agents' strategies took into consideration mean calculations. In particular, 97% (corresponding to 31 of the 32 agents that did not base their selection on querying a single server only) implemented some level of mean computation and 16% (5 out of the 32) fully implemented the pure-mean strategy.

The second classification method that was applied is based on the agents' performance with the first set of 5000 problems. Figure 4 depicts the average performance of each agent with that set. The vertical axis represents the average overall time until execution and the horizontal axis is the agent's number (for clarity of presentation the agents are ordered according to their performance, thus the IDs are not correlated with those given in Figure 3). The two horizontal lines in the figure represent the performance of an agent searching according to the optimal strategy and an agent that randomly queries and picks a single server. As can be seen from the figure, none of the agents' strategies reached the level of performance of the optimal strategy. The average overall time (cost) obtained by the agents is 446.8, while that of the optimal agent is 223.1. Furthermore, many of the strategies (40 of 72) did even worse than the "random" agent.

From Figure 4, it seems that other than the group (40 - 71) most of the agents cannot be classified merely based on average performance. Therefore, an attempt was made to identify clusters of agents, based both on agents' performance and their design characteristics as given in Figure 3. The following clusters emerged from this assessment:

- The naive mean-based greedy search strategy and its variants (agents 3-7).
- Mean-based approaches that involve preliminary filtering of servers according to means and costs (agents 14-16).
- A variation of the naive mean-based greedy search strategy that also takes the variance of each server as a factor (agents 21-22).
- Querying the two servers with the lowest expected queue length and assigning the job to the one with the minimum value found (agents 24-26).
- Assigning the job to the first/last/a random server (agents 40-71).

For many agents, similarities in performance could not be explained by resemblances among the strategies themselves. In fact, the above classification covers only 45 of the 72 agents. The remaining agents, even when associated with an average performance close to the former agents', do not share any strategy characteristic with them, giving evidence to large number of different strategies used. Even in cases, where agents used different variants of the same basic strategy, the differences among the variants resulted in substantial performance differences.

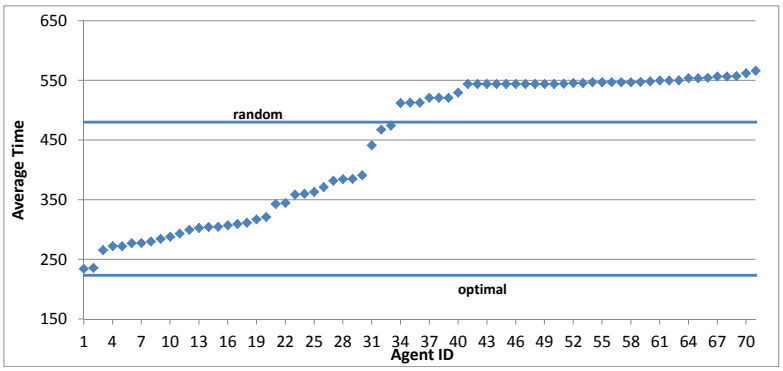


Fig. 4 The individual average performance in the original (non-restructured) problems of the first set (to make the results easier to follow, agents' IDs represent the relative order according to the individual performance).

We extended the performance-based classification by considering the agents' rankings. Figure 5 compares the agents' performance time-wise and rank wise. The time-wise measure represents an agent's average time (i.e., cost equivalent) over the first set of problems. The rank-wise measure represents an agent's average relative ranking (where 1 is the best rank). This measure was calculated by extracting each agent's rank (based on its performance compared to the other agents' performance) in each problem, and taking the average ranking of the agent over the entire set. In general, as observed from the graph, the time-wise and the rank-wise measures are correlated. Three areas of inconsistency between the two can be observed in the figure (marked with circles). They relate to the agents associated with similar individual average performance, however with very different average rankings (1A vs. 1B, 3A vs. 3B) and vice versa (2A vs. 2B). A drill-down analysis based on the strategy documentation of the agents associated with these inconsistencies reveals the reasons for these variations. The most interesting inconsistency is between clusters "1A" and "1B" which had relatively good performance. All agents belonging to these two groups are mean-based; the difference between them is in their "stopping rule". The agents belonging to group "1A" terminate their search if the queue length of the **recently** explored server is smaller than the sum of any of the servers' queue length and its querying time. The "stopping rule" of the agents belonging to group "1B", on the other hand, is to terminate the search if the **minimal** queue length found so far is smaller than the sum of any of the servers' queue length and its querying time. When the agents from group "1A" are evaluated according to their average time their performance is close to the performance of the naive mean-based agents, however the rank-based measure reveals that they perform worse than the naive mean-based agents. The average-time-based measure is unable to differentiate between the two groups because the difference in their performance is not significant. This is because the additional exploration by agents of group "1A" adds very little to the average (time-wise). The changes, though small in absolute values, are consistent and thus affect ranking to an extent that differentiate this group from the naive mean-based agents.

The second major inconsistency is between agents of groups "2A" and "2B". Based on the average ranking of these agents, one would expect that they would

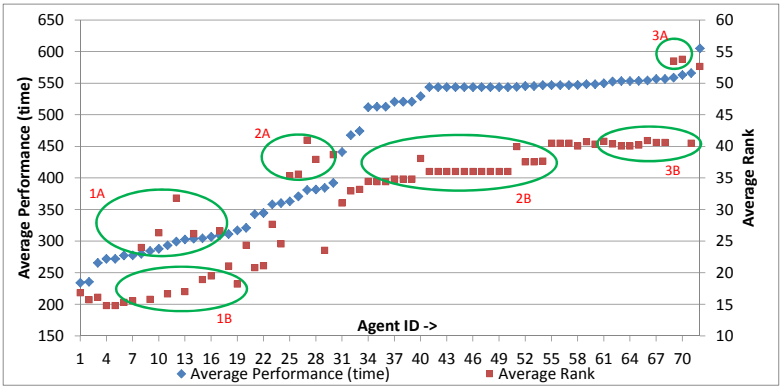


Fig. 5 Individual average performance (primary vertical axis) versus individual average ranking (secondary vertical axis).

have similar average performances. However, the performance measure reveals substantial differences in performance. Common to all agents in those groups is the search extent; in the worst case they all query three servers. The difference is that the agents from "2B" always query three servers whereas the agents from "2A" first check if the expectancy of the next server is lower than the minimal queue length revealed so far. While the effect of this difference on the agents' performance is substantial, its effect on the ranking is negligible. This is because of the frequency in which the two search sequences differ — the number of times that agents from group "2A" end up querying less than three servers is moderate, hence the effect on ranking is minor. However, the performance difference in those cases is substantial, resulting in a noticeable effect over the agents' average performance.

The last observable inconsistency relates to agents of groups "3A" and "3B". Agents from group "3B" randomly choose a server and assign the job to it, whereas the agents from group "3A" choose the server with the maximum query time (e.g., hoping that the more "expensive" will yield the best result) and assign it the job. In this case, the difference in performance is small but consistent, hence the difference in ranking is substantial.

7 Results

In this section we report the results of the different restructuring heuristics based on experimenting with agents and with people.

7.1 Agents' Results

The following paragraphs detail the performance achieved by each heuristic, based on the set of 72 agents. It is noteworthy, as reported in the former section, that from the 72 agents, a significant number (40) based their strategy on querying a single server only. While this selection rule is legitimate, the fact that it is used by a substantial number of agents may suggest a great influence over the results.

Therefore, for each heuristic we also provide the average performance measures when calculated for the subset of agents that use richer strategies, showing that even for that subset alone the same results hold.

The results in this section are based on the first set of 5000 search problems. Later we show that these results carry over to the second and third sets ("Increased variance" and "Increased search costs").

7.1.1 Information Hiding

The threshold α used for removing alternatives from the problem instance is a key parameter affecting the "Information Hiding" heuristic. Figure 6 depicts the average time until execution (over 72 agents, for the 5000 problem instances of the first problem set) for different threshold values. For comparison purposes, it also shows the average performance on the non-restructured set of problems, which corresponds to $\alpha = 0$ (a separate horizontal line). The shape of the curve has an intuitive explanation. For small threshold values, an increase in the threshold increases agent performance as it further reduces the chance for a possible deviation from the optimal sequence. Since the probability that the removed servers are actually needed is very small, the negative effect of the removal of options is negligible. However, as the threshold increases, the probability that an opportunity that was removed is actually needed by the optimal strategy substantially increases, and thus a greater performance decrease is experienced.

As can be observed from Figure 6, the optimal (average-time-minimizing) threshold, based on the 5000 problem instances of the first set, is $\alpha = 10\%$, for which an average time of 407.5 is obtained. This graph also shows that for a large interval of threshold values around this point — in particular for $3\% \leq \alpha \leq 30\%$ — the performance level is relatively similar. Thus, the improvements obtained are not extremely sensitive to the exact value of α used; relatively good performance may be achieved even if the α value used is not exactly the one which yields the minimum average time. In fact, any threshold below $\alpha = 62\%$ results in improved performance in comparison to the performance obtained without the use of this restructuring heuristic (i.e., with $\alpha = 0$). Figure 7 complements Figure 6 by depicting the average performance of the optimal search strategy when used with the information hiding heuristic, with different values of α. As discussed above, the use of the information-hiding heuristic results in performance degradation whenever an opportunity that was removed is actually required for the searcher according to the optimal search sequence. The figure illustrates that, for low values of α (up to $\alpha = 16\%$), the degradation in the optimal search strategy's performance is relatively moderate. This result strengthens the applicability of the information-hiding heuristic, as for the relevant α-values, even if the searcher uses the optimal search strategy, the performance degradation is mild.

We analyzed the opportunities that were removed when Information Hiding was applied. The analysis revealed that those opportunities that were removed were not necessarily the more certain ones nor the ones with the low expected payoff. In fact, the distribution of mean and variance of these opportunities is very similar to the one characterizing all opportunities in general. Hence there is no general rule of thumb for what opportunities will be removed most often solely based on these two measures.

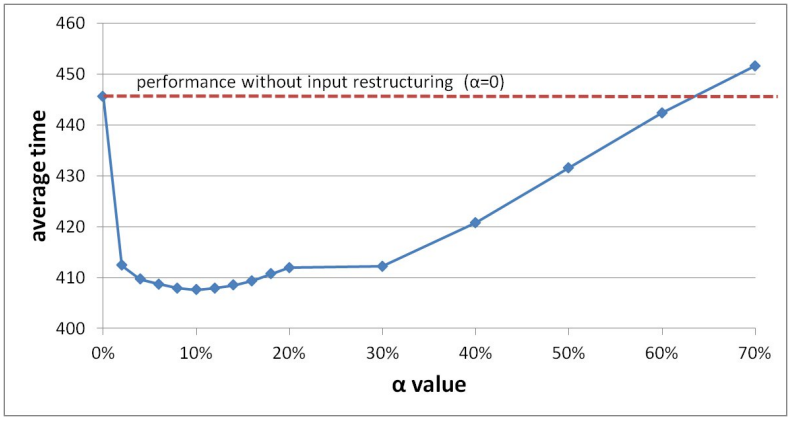


Fig. 6 The effect of α on the average performance (cross-agents) in "information hiding".

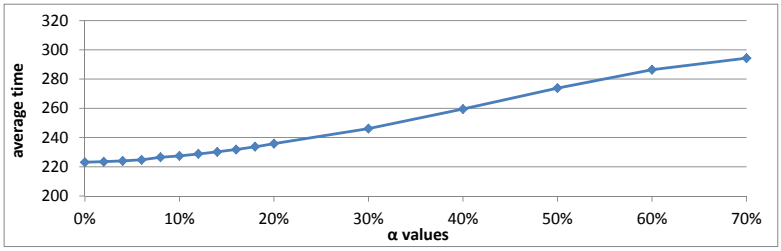


Fig. 7 The effect of α on the average performance of the optimal search strategy, when information hiding is applied.

Figure 8 depicts the individual reduction in search inefficiency (according to Equation 6) of each agent with $\alpha = 10\%$. As shown, this heuristic managed to decrease the search inefficiency of 64 of the 72 agents. The average reduction in individual search inefficiency (according to (8)) is 15.5% and the average individual performance improvement (according to (7)) is 7.8%. The maximum individual reduction in search inefficiency that was obtained is 80.2%. Alongside the improvement in the agents' performance, this heuristic also worsened the performance of some of the agents. The maximum decrease in individual performance was found for an agent whose search inefficiency increased by 12.1%, which corresponds to a 2.2% degradation in its individual performance measure (according to (5)).

Similar results are obtained when agents that query and choose only a single opportunity were excluded. In this case the average reduction in individual search inefficiency is 14.9% and the average individual performance improvement is 7.9%. The maximum individual reduction in search inefficiency is 80.2% as before.

The main advantages of the information hiding heuristic are therefore that it improves the performance of most agents and that even in cases in which an individual agent's performance degrades, the degradation is relatively minor. Thus, the heuristic is a good candidate for use as a default problem-restructuring heuristic whenever there is no information about the searcher's strategy or an agent cannot be classified accurately.

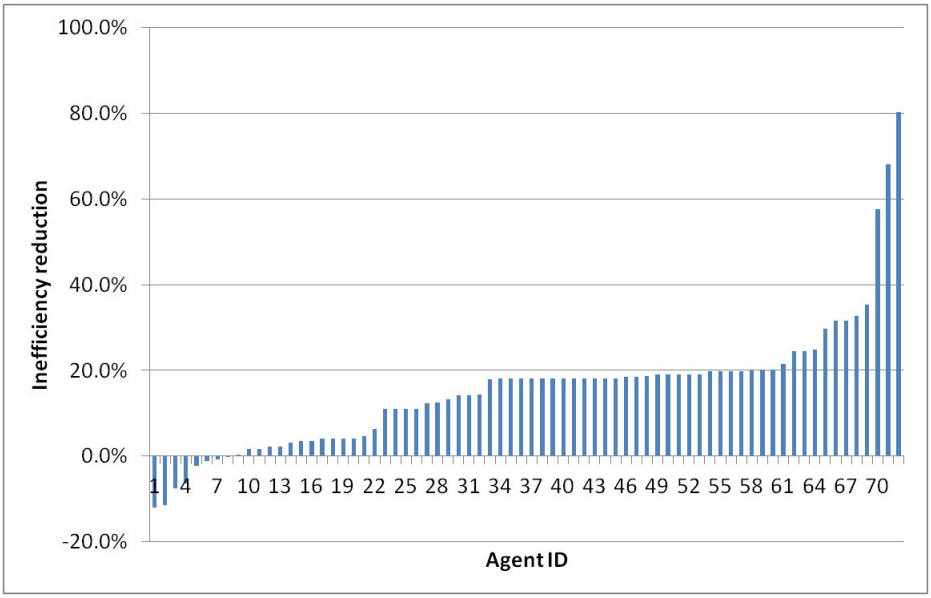


Fig. 8 Average reduction in the individual search inefficiency of information hiding for $\alpha = 10\%$ (agent's ID represents its order rather than the specific agent's identification).

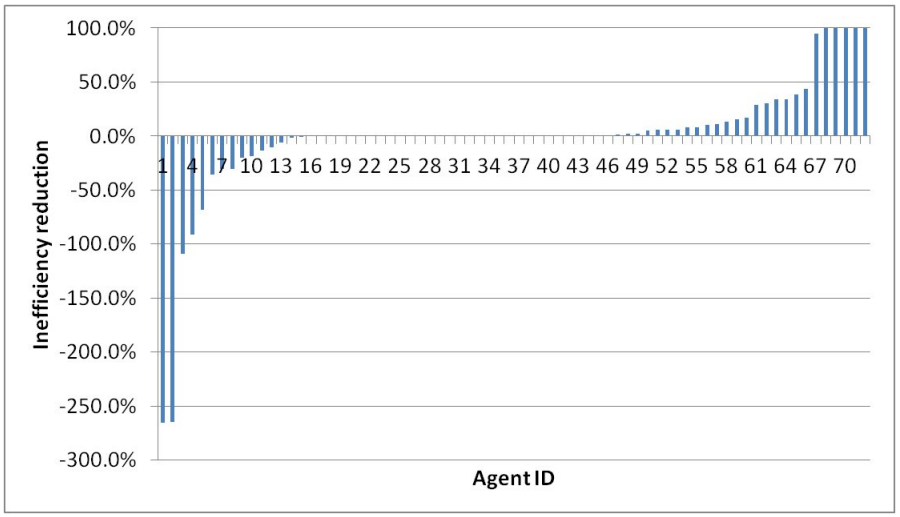


Fig. 9 Average reduction in the individual search inefficiency of Mean Manipulation (agent's ID represents the order rather than the specific agent's identification).

7.1.2 Mean Manipulation

Figure 9 depicts the average reduction in search inefficiency of each agent when using the "Mean Manipulation" heuristic. With this heuristic, the search inefficiency of five of the agents was eliminated almost entirely (the last five agents to

the right). These agents used variants of the naive mean-based greedy search strategy. Other agents also benefited from this heuristic and substantially reduced the overhead associated with their inefficient search. These agents also use mean-based considerations, to some extent, in their search strategy.

This heuristic has a significant downside, though — 15 agents did worse with the mean manipulation heuristic, 12 of them substantially worse. For these agents the search inefficiency increased by up to 250%. Overall, the average individual search inefficiency with the mean manipulation slightly increased (by 0.6% according to (8)), while the average individual performance improvement (according to (7)) was 1.5%. This example of an improvement by one measure which is reflected as a degradation according to another is a classic instance of Simpson's paradox [80].

Similar results are obtained when excluding agents that query and choose only a single opportunity. In this case the average individual search inefficiency increased by 5.8%, and the average individual performance degraded by 3.4%

The substantial increase in search overhead for some of the agents makes this heuristic inappropriate for general use. It is, however, very useful when incorporated into an adaptive mechanism that attempts to identify agents that use mean-based strategies and applies this restructuring method to their input (as in the case of our adaptive learner heuristic).

7.1.3 Random Manipulation

Figure 10 depicts the average reduction in search inefficiency of each agent when using the "Random Manipulation" heuristic. This heuristic improved the performance of 47 agents (a decrease in the individual inefficiency of 50% for most of them according to (6)). The remaining 25 agents (all of them from the group of the 32 agents that do not limit the evaluation to one opportunity only) did substantially worse with this restructuring heuristic; for two of them the individual search inefficiency increased by more than 1000%. Overall, the average individual search inefficiency increased by 38.8% (according to (8)), and the average individual performance was improved by 17.9% (according to (7)), which is again explained by Simpson's paradox.

As expected, the results with this heuristic are substantially worse when excluding agents that query and choose only a single opportunity. In this case the individual search inefficiency increases in average by 150.6%, and the individual performance degrades by 19.1%.

7.1.4 Adaptive Learner

The adaptive learner was found to be the most effective heuristic among the four. Figure 11 depicts the average reduction in search inefficiency for each agent when using the adaptive learner heuristic (with a threshold of $\gamma = 7\%$ and $\alpha = 10\%$ for information hiding). As observed in the figure, the improvement in most agents' individual search inefficiency is very close to the best result obtained with any of the former three heuristics, indicating that the adaptive learner managed to successfully assign each agent with the most suitable restructuring method. Overall, the average reduction in individual search inefficiency (according to (8)) is 43.7%. The average individual performance improvemant (according to (7)) is 20.3%. Two

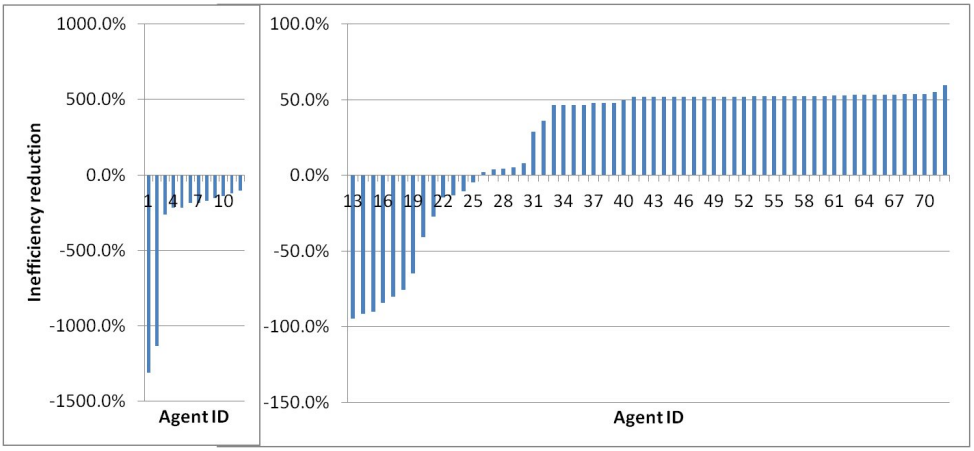


Fig. 10 Average reduction in the individual search inefficiency of Random Manipulation (agent's ID represents the order rather than the specific agent's identification). For exposition purposes, the data is given as two graphs. The left graph contains the 12 agents whose performance worsened by more than 100% and the right graph contains the remaining agents (hence the different scale of the vertical axis).

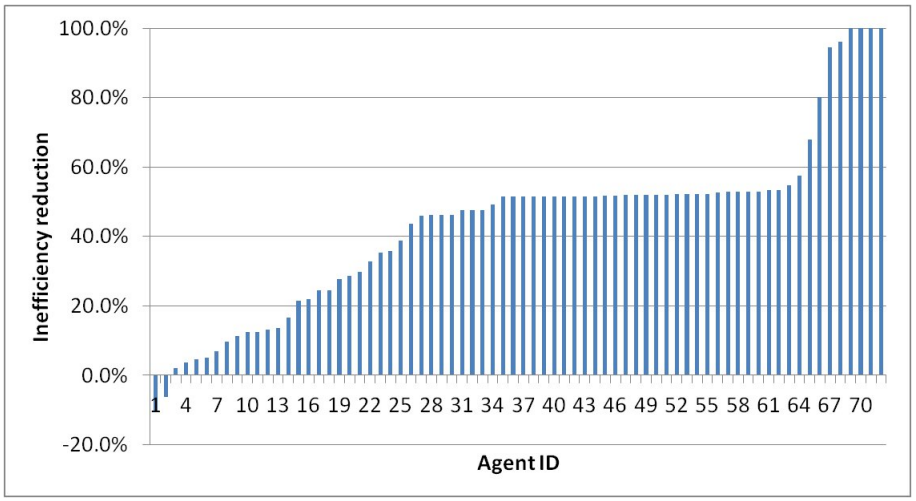


Fig. 11 Average reduction in the individual search inefficiency of Adaptive Learner (agent's ID represents the order rather than the specific agent's identification).

of the 72 agents slightly worsened their performance (maximum of 10.2% increase in search inefficiency (according to (6)), which is equivalent to a decrease of 2.4% in individual performance (according to (5))). Similar results are obtained when excluding agents that query and choose only a single opportunity. In this case the average reduction in individual search inefficiency is 37.6% and the average individual performance improvemant is 10%.

Figure 12 depicts the average accuracy of the adaptive learner in assigning the 72 agents to their appropriate classes. The accuracy is taken to be a binary

measure, receiving 1 if the agent is classified to the class in which the restructuring method yields the best performance among the four restructuring heuristics discussed in this paper and 0 otherwise. The figure presents the average accuracy cross-agents, per round number (i.e., the value for round x reflects the average accuracy resulting from a classification that takes as an input a set of x-1 problem instances). As observed from the figure, the adaptive learner achieves quite impressive classification accuracy even with a moderate number of previously observed problem instances. In fact, even with 10 records in its database, the adaptive learner manages to achieve 86% classification accuracy. The fact that the performance of the method remains steady around 92% as more observations accumulate suggests that the agents are consistent and the strategies they use are adequately identified based only on their results. The fact that the agents are consistent is not surprising, since most agents did not use randomization in their strategy. Even those agents that did use randomization in their strategy were mostly from the "random" class and could be identified based on the fact that they explored only a single opportunity. The fact that agents were adequately classified based on a small subset of observations is of greater importance. It implies that the strategies used by agents of different classes are substantially different, and the difference is reflected to a great extent, according to the performance measure used, enabling an accurate classification. The fact that the classification accuracy does not go beyond 92%, even with 5000 problem instances suggests that there is a small subset of agents which cannot be correctly classified by the adaptive learner, regardless of the extent of prior observations available.

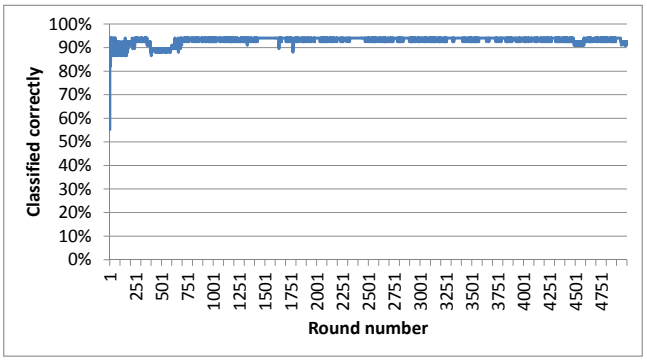


Fig. 12 Percentage of correct classification in 5000 problems (cross agents) performed by the adaptive learner heuristic. The values on the horizontal axis represent the number of prior problem instances available to the adaptive learner for the classification.

Figure 13 complements the analysis given based on Figure 12 by depicting the number of changes made in the classification decision of the adaptive learner for each agent. It is divided into three graphs, each referring to a different interval of iterations of the 5000 rounds. The horizontal axis represents agent IDs and the vertical axis is the number of changes made in that agent's classification in the relevant interval (agent's ID represents order rather than a specific agent's identification). The most interesting observation made based on Figure 13, is that for most agents (39) there was no change in agent classification after determin-

ing its initial classification based on its result in a single problem instance. The classification of 63 of the agents did not change after 100 observations . Accumulating more observations change the classification of the remaining agents almost insignificantly. Overall, the number of changes in the classification of agents of that latter group is small and such changes are rare in the interval $100 - 5000$.

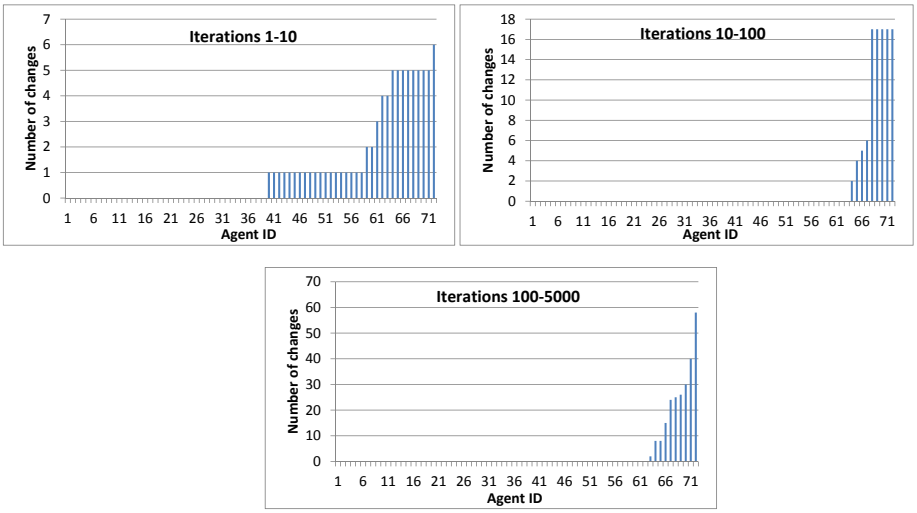


Fig. 13 The number of changes made by the adaptive learner in the classification decision for each agent over three intervals: (0 - 10), (10 - 100) and (100 - 5000).

7.2 Evaluation with Different Problem Sets

Table 4 presents the results obtained for the second and the third problem sets with the 72 agent searchers (as discussed in Section 5.3). As can be seen from the table, the improvement obtained using the adaptive heuristic over the second and the third sets of problems is consistent with the one obtained using the first set.

	Original	Inc. Var	Inc. Quer.
Non-Restructured	446.8	3895.3	559.6
Adaptive	344.3	3004.16	428.9
Optimal	223.1	1349.7	332.3
Average reduction in search inefficiency	43.7%	25.7%	49.3%
Average individual performance improvement	20.3%	18.2%	19.7%
Maximum individual increase in search inefficiency	10.2%	16.7%	26.6%
Maximum individual performance degradation	2.4%	0.5%	4.4%

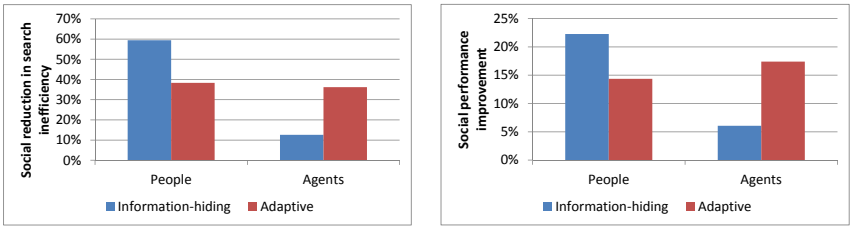
Table 4 Performance for different classes of problems

7.3 Results Obtained with Human Searchers

Since a "between subjects" design was used for the experiments with people as decision makers, the methodology used for the analysis of the results in this section is different from the one used for agents (as discussed in 5.6 in more detail). The primary evaluation measures used for evaluating the restructuring heuristic in this section is the social performance improvement (according to (3)) and the social reduction in search inefficiency (according to (4)), and the problems used for this experiment are from the fourth set of problems. The average overall performance of people (in terms of overall expense in the "repairman problem") without applying any restructuring technique was 369.2. When using the Information hiding manipulation, the average was 287, and with the adaptive learner the average was 316.2. The average performance of the optimal strategy when used with this set is 230.8. These results reflect a social reduction in search inefficiency of 59.4% (according to (4)), and 22.3% improvement in social performance (according to (3)) when the information-hiding heuristic is used. With the adaptive learner, a reduction in search inefficiency (according to (4)) of 38.3% and a 14.4% improvement in social performance (according to (3)) are obtained.

These results with people, while encouraging, do not correspond with the dominance of the adaptive learner over the other approaches that was found with agents. Figure 14 depicts the overall average social reduction in search inefficiency and the overall average social performance improvement of people and agents over the fourth problem set with information-hiding and adaptive learner heuristics. For computer agents the best is the adaptive learner. As this figure shows, the opposite is true for people. This is reflected in both measures and to a similar extent. The dominance of the information-hiding over adaptive learner heuristic when used with people may be explained by people's inconsistency in decision-making (hence preventing the adaptive learner from efficiently classifying people) or by a difference between the strategies used by people and those strategies upon which the restructuring heuristics are based. Support for the first possible explanation can be found in Figure 15, depicting the percentage of people that the adaptive learner assigned to a class other than the default one, in each of the 20 rounds. As the figure shows, the classification to an actual (non-default) class converges to 40% of the people. For the remaining 60%, the information-hiding heuristic was used as a default. While the use of information-hiding with a large portion of the population turned out to be beneficial (based on the overall performance of the method with people), the fact that the adaptive learner resulted in a worse performance measure suggests that even those 40% that were classified according to non-default classes were used with a restructuring heuristic that actually made them perform substantially worse.

The main conclusion from the comparison of agents' and people's performance with the different heuristics is that for situations where the searchers are people a simple restructuring heuristic like the information hiding heuristic is more suitable than a complex one, which attempts to classify the searcher's strategy online, such as the adaptive technique. When facing a consistent searcher (i.e., agent), on the other hand, an adaptive heuristic can substantially improve the searcher's performance.



(a) (b)

Fig. 14 A comparison of the performance achieved with information-hiding and adaptive learner heuristics when applied over people and agents: (a) social reduction in search inefficiency; and (b) social performance improvement.

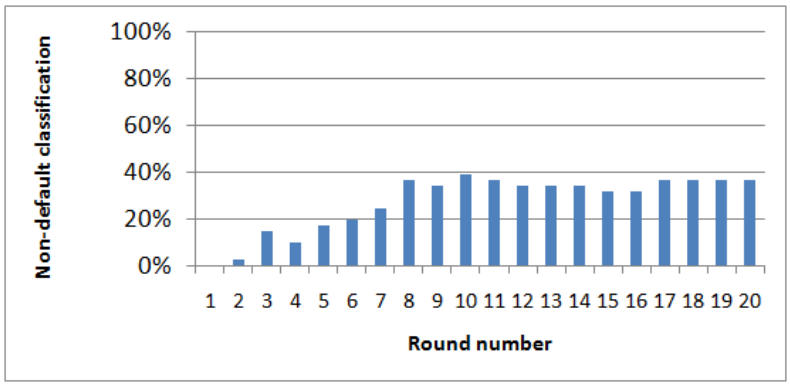


Fig. 15 Percentage of people that the adaptive learner assigned to class other than the default one, on each of the 20 rounds of the experiment.

8 Related Work

Research in psychology and behavioral economics has established that people are bounded rational [79,64,38] and argued that their decision-making may be influenced by such factors as the tendency to gather facts that support certain conclusions and disregard those that do not [9,21]. Other research attributes people's difficulty in decision-making to the conflict between a "reflective system" and an "automatic system" [85,17]. As described in the body of this paper, other research has investigated the usefulness of restructuring for decisions simpliciter [3,76,85, 11,86].

Research in economics on search theory has focused largely on theoretical aspects of the optimal search strategy, with a smaller body of work showing the extent to which people's strategies diverge from the theoretical optimal one [33, 58]. The major finding of this work, which focuses on a "stopping rule" (which assumes all opportunities share the same distributions of values, i.e., there is no heterogeneity of distributions), is that people tend to terminate their search before the theoretic-optimal strategy would [81,74].

A number of approaches have been pursued to the design of decision-support systems [62, 92, 8, interalia] which assist people in identifying, reasoning about, and assessing a (more) full set of possible outcomes. Research in artificial intelligence has proposed a variety of approaches such as "persuasion" [31, 60, 39, 27, 48] and negotiation [29, 47, 77], techniques that aim to convince people through argument or iterative discussion, respectively, to adopt different strategies or choices. The techniques for persuasion require models of the potential for different types of arguments to influence a person to change strategy. Negotiation methods require models of people's negotiation strategies and typically require several iterations to reach an agreement. The restructuring approach we present in this paper requires neither these complex models of people nor repeated interactions.

Research in multi-agent systems has produced a variety of types of agents to represent and act on behalf of people, including agents that act as proxies in trading (as in the Trading Agent Competition [91]) and buy and sell in virtual marketplaces ([15, interalia]). Agents have also been designed for the Colored-Trails game [26], which enables different negotiation and decision-making strategies to be tested. Chalamish et al. [13] use Peer-designed agents (PDAs) for simulating parking lots, giving evidence for some similarity between the behaviors exhibited by people and the agents they program. Empirical investigation of the level of similarity observed between PDAs and people in general is not conclusive: some work suggests a relatively strong correlation between the behaviors of the two [14, 52, 51], while other work reports that PDAs behave differently [26, 69].

9 Discussion and Conclusions

This paper presents novel methods for improving computer agent performance within sequentially dependent decisions. It defines a set of restructuring heuristics that enable platforms such as websites connecting producers and consumers of various types, to improve the search of autonomous agents acting as proxies for people. This kind of restructuring is useful for settings in which a platform is not able to influence directly the design of agent strategies but can only control the information they obtain. The use of restructuring is completely transparent to the decision maker, and does not require any direct interaction with her. The method also has many benefits even in situations where direct interaction with the decision maker is possible: it saves the need to explain the optimal strategy to the decision maker or persuade non-experts in decision making in the correctness and optimality of that strategy. Within the scope of economic search, there are many platforms in which users need to spend time (usually a scarce resource) in order to reduce uncertainty about the benefits (or characteristics) of different possible choices or opportunities. The improvement in users' welfare (e.g., in time saved or better decisions made) that results from problem restructuring should make a website more appealing to users, thus increasing traffic and revenues.

The research described in this paper is, to the best of our knowledge, the first to define restructuring for sequential decision making. Restructuring sequentially dependent decision problems has significant challenges not present in restructuring for decisions simpliciter. In particular, the restructuring heuristics cannot eliminate alternative opportunities simply on the basis of a single value or rules of thumb based on certainty and variance alone. The temporal nature of sequential decision

making combined with uncertainty about outcomes requires some consideration of possible futures in which the prior removal of some alternative could worsen a decision-maker's performance. The research is novel also in focusing on improving agent performance from the "outside", which is required because websites cannot control the design of the agent proxies that interact with them.

The results reported in Section 7 provide a proof of concept for the possibility of substantially improving agent performance in recurring sequential decision making by restructuring the problem space. An extensive evaluation of the heuristics in four different classes of search environments, and with both agents (for whom they were designed) and people (to test generalization). The empirical investigations involved a large number of agents, each designed by a different person, and a large number of problem instances within each class. Even though it has no information about an agent's strategy, the information hiding heuristic substantially improves average performance. Although it can degrade some individual performance, the extent to which it does so is limited both in the level of negative impact and the extent of agents involved. Given even a small amount of information about an agent's prior search behavior, the adaptive heuristic is able to increase overall average performance and lower the possible negative impact on individual agent performance. The improvement in people's performance resulting from use of the heuristics supports the hypothesis that people's divergence from the theoretical-optimal strategy is caused primarily by their limited ability to extract it. Examination of the code and documentation for the agents used in our experiments revealed that a large number of programmers used a strategy based in some way on the means (rather than the distributions) of the opportunities, supporting both prior work that shows people have preference for mean over distribution and that the mean-manipulation heuristic has the potential to be broadly applicable. Interestingly, none of the agents used the optimal strategy, providing additional, albeit weaker, evidence that this strategy is not an intuitively obvious one.

The information hiding heuristic substantially improved the performance of both agents and people. This result indicates that both people and agents have difficulty identifying opportunities that have a low probability of contributing to the search process according to the theoretic-optimal solution. This result is yet another piece of evidence that observed differences between people's behavior and the theoretic-optimal strategy is rooted in people's limited understanding of the search problem rather than limitations of the payoff-maximizing search model.

As reported in Section 7, the adaptive learner heuristic led to the best performance for agents, but was less successful when used with people. These results raise several issues. First, the identification of additional agent strategies for which new restructuring heuristics are useful will increase the adaptive heuristics potential for enhancing performance, because the adaptive heuristic's architecture is modular, allowing easy incorporation of new restructuring heuristics. Second, the lesser improvement provided by this heuristic for people raises the question of whether the failure to classify people's strategies results from their being less consistent than agents or from their use of strategies for which the particular set of heuristics we explored is less applicable. Finally, we note that a natural extension the approach taken in this paper is to develop heuristics that will choose the restructuring method to be applied based not only on agent classification but also on problem instance characteristics. To do so, will require a more refined analysis

at the agent level. In that spirit we also suggest future work in developing effective restructuring heuristics for people. Indeed the benefit achieved with the heuristics proposed for agents was found to be substantial also when tested with people. Still, it is possible that specific restructuring-heuristics, ones that are designed to take advantage of human behaviors observed in sequentially dependent decisions, will be found to be even more effective.

Acknowledgments

Preliminary results of this research appear in a conference paper [71]. This research was partially supported by ISF/BSF grants 1401/09 and 2008-404. We are grateful to Moti Geva for his help with developing the agent-based experimental infrastructure and the proxy program.

References

1. Ma. Allais. The foundations of a positive theory of choice involving risk and a criticism of the postulates and axioms of the american school (1952). In *Expected Utility Hypotheses and the Allais Paradox*, volume 21 of *Theory and Decision Library*, pages 27–145. Springer Netherlands, 1979.
2. AMT. *http://www.mturk.com/*.
3. D. Ariely. *Predictably Irrational, Revised and Expanded Edition: The Hidden Forces That Shape Our Decisions*. HarperCollins, 2010.
4. Y. Bakos. Reducing buyer search costs: Implications for electronic marketplaces. *Management Science*, 42:1676–92, 1997.
5. G. Barron and I. Erev. Small feedback-based decisions and their limited correspondence to description-based decisions. *Journal of Behavioral Decision Making*, 16(3):215–233, 2003.
6. T. Bench-Capon, K. Atkinson, and P. McBurney. Using argumentation to model agent decision making in economic experiments. *Autonomous Agents and Multi-Agent Systems*, 25(1):183–208, 2012.
7. M. Bertrand and S. Mullainathan. Do people mean what they say? implications for subjective survey data. *American Economic Review*, 91(2):67–72, 2001.
8. P. Bharati and A. Chaudhury. An empirical investigation of decision-making satisfaction in web-based decision support systems. *Decision support systems*, 37(2):187–197, 2004.
9. G. C. Blackhart and J. P. Kline. Individual differences in anterior EEG asymmetry between high and low defensive individuals during a rumination/distraction task. *Personality and Individual Differences*, 39(2):427–437, 2005.
10. M. Brown, C. J. Flinn, and A. Schotter. Real-time search in the laboratory and the market. *The American Economic Review*, 101(2):948–974, 2011.
11. A. Burgess. Nudginghealthy lifestyles: The uk experiments with the behavioural alternative to regulation and the market. *European Journal of Risk Regulation*, 1:3–16, 2012.
12. D. Carmel and S. Markovitch. Exploration and adaptation in multiagent systems: A model-based approach. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI 97)*, pages 606–611, 1997.
13. M. Chalamish, D. Sarne, and S. Kraus. Mass programmed agents for simulating human strategies in large scale systems. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems (AAMAS-07)*, pages 1–3, 2007.
14. M. Chalamish, D. Sarne, and S. Kraus. Programming agents as a means of capturing self-strategy. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems (AAMAS-2008)*, pages 1161–1168, 2008.
15. A. Chavez and P. Maes. Kasbah: An agent marketplace for buying and selling goods. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, pages 75–90, 1996.
16. S. P. Choi. Optimal time-constrained trading strategies for autonomous agents. In *Proceedings of MAMA2000*, 2000.

17. Kahneman D. *Thinking, fast and slow*. Macmillan, 2011.
18. V. I. Danilov and A. Lambert-Mogiliansky. Expected utility theory under non-classical uncertainty. *Theory and decision*, 68(1-2):25–47, 2010.
19. M. L. Dekay and T. G. Kim. When Things Don't Add Up: The Role of Perceived Fungibility in Repeated-Play Decisions. *Psychological Science*, 16(9):667–672, 2005.
20. W. P. Dommermuth. The shopping matrix and marketing strategy. *Journal of Marketing Research*, 2:128–132, 1965.
21. R. A. Drake. Processing persuasive arguments: Discounting of truth and relevance as a function of agreement and manipulated activation asymmetry. *Journal of Research in Personality*, 27(2):184–196, 1993.
22. T. Dudey and P. M. Todd. Making good decisions with minimal information: Simultaneous and sequential choice. *Journal of Bioeconomics*, 3(2-3):195–215, 2001.
23. G. Dushnitsky and T. Klueter. Is there an ebay for ideas? insights from online knowledge marketplaces. *European Management Review*, 8:17–32, 2011.
24. H. J. Einhorn and R. M. Hogarth. Behavioral decision theory: Processes of judgment and choice. *Journal of Accounting Research*, 19(1):1–31, 1981.
25. T. Ferguson. Who solved the secretary problem? *Statistical Science*, 4(3):282–289, 1989.
26. B. J. Grosz, S. Kraus, S. Talman, B. Stossel, and M. Havlin. The influence of social dependencies on decision-making: Initial investigations with a new game. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2004)*, pages 780–787, 2004.
27. M. Guerini and O. Stock. Toward ethical persuasive agents. In *Proceedings of the International Joint Conference of Artificial Itelligence Workshop on Computational Models of Natural Argument*, 2005.
28. V.A. Ha and P. Haddawy. Toward case-based preference elicitation: Similarity measures on preference structures. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI '98)*, pages 193–201, 1998.
29. G. Haim, Y.K. Gal, M. Gelfand, and S. Kraus. A cultural sensitive agent for human-computer negotiation. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2012)*, pages 451–458, 2012.
30. C. Harries, J. S. Evans, and I. Dennis. Measuring doctors' self-insight into their treatment decisions. *Applied Cognitive Psychology*, 14:455–477, 2000.
31. N. Hazon, R. Lin, and S. Kraus. How to change a groups collective decision? *To appear in The 23rd International Joint Conference on Artificial Intelligence (IJCAI-13)*, 2013.
32. J. D. Hey. Search for rules for search. *Journal of Economic Behavior & Organization*, 3(1):65–81, 1982.
33. J. D. Hey. Still searching. *Journal of Economic Behavior and Organization*, 8(1):137 – 144, 1987.
34. T. Hills and R. Hertwig. Information search in decisions from experience : do our patterns of sampling foreshadow our decisions? *Psychological Science*, 21(12):1787–1792, 2010.
35. L. Hogg and N. R. Jennings. Socially intelligent reasoning for autonomous agents. *IEEE Trans on Systems, Man and Cybernetics - Part A*, 31(5):381–399, 2001.
36. D. Kahneman and D. Lovallo. Timid choices and bold forecasts: A cognitive perspective on risk taking. *Management science*, 39(1):17–31, 1993.
37. D. Kahneman and A. Tversky. Prospect theory: An analysis of decision under risk. *Econometrica: Journal of the Econometric Society*, pages 263–291, 1979.
38. D. Kahneman and A. Tversky. *Choices, values, and frames*. Cambridge University Press, New York, 2000.
39. M. Kaptein, S. Duplinsky, and P. Markopoulos. Means based adaptive persuasive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, (CHI '11), pages 335–344, 2011.
40. G. I. Kempen, M. J. Van Heuvelen, R. H. Van den Brink, A. C. Kooijman, M. Klein, P. J. Houx, and J. Ormel. Factors affecting contrasting results between self-reported and performance-based levels of physical limitations. *Age and Ageing*, 25(6):458–464, 1996.
41. J. O. Kephart and A. Greenwald. Shopbot economics. *Autonomous Agents and Multi-Agent Systems*, 5(3):255–287, 2002.
42. G. Keren. Additional tests of utility theory under unique and repeated conditions. *Journal of Behavioral Decision Making*, 4(4):297–304, 1991.
43. G. Keren and W. A. Wagenaar. Violation of utility theory in unique and repeated gambles. *Journal of experimental psychology. Learning, memory, and cognition*, 13(3):387–391, 1987.

44. L. R. Klein and G. T. Ford. Consumer search for information in the digital age: An empirical study of prepurchase search for automobiles. *Journal of Interactive Marketing*, 17(3):29–49, 2003.
45. D. Kleinmuntz and J. Thomas. The value of action and inference in dynamic decision making. *Organizational Behavior and Human Decision Processes*, 39(3):341–364, 1987.
46. A. Klos, E. U. Weber, and M. Weber. Investment decisions and time horizon: Risk perception and risk behavior in repeated gambles. *Management Science*, 51(12):1777–1790, 2005.
47. S. Kraus, P. Hoz-Weiss, J. Wilkenfeld, D. R. Andersen, and A. Pate. Resolving crises through automated bilateral negotiations. *Artificial Intelligence*, 172(1):1–18, 2008.
48. S. Kraus, K. Sycara, and A. Evenchik. Reaching agreements through argumentation: a logical model and implementation. *Artificial Intelligence*, 104(1):1–69, 1998.
49. T. Langer and M. Weber. Prospect theory, mental accounting, and differences in aggregated and segregated evaluation of lottery portfolios. *Management Science*, 47(5):716–733, 2001.
50. M. D. Lee. A hierarchical bayesian model of human decision-making on an optimal stopping problem. *Cognitive Science*, 30(3):555–580, 2006.
51. R. Lin, S. Kraus, N. Agmon, S. Barrett, and P. Stone. Comparing agents: Success against people in security domains. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI-2011)*, pages 809–814, 2011.
52. R. Lin, S. Kraus, Y. Oshrat, and Y. Gal. Facilitating the evaluation of automated negotiators using peer designed agents. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-2010)*, pages 817–822, 2010.
53. M. L. Littman. *Algorithms for sequential decision making*. PhD thesis, Brown University, 1996.
54. E. M. Liu. Time to change what to sow: Risk preferences and technology adoption decisions of cotton farmers in china. *Forthcoming Review of Economics and Statistics*, 2013.
55. H. Liu and A. Colman. Ambiguity aversion in the long run: Repeated decisions under risk and uncertainty. *Journal of Economic Psychology*, 30(3):277–284, 2009.
56. H. Markowitz. Mean variance approximations to expected utility. *Forthcoming European Journal of Operational Research*, 2013.
57. H. Montgomery and T. Adelbratt. Gambling decisions and information about expected value. *Organizational Behavior and Human Performance*, 29(1):39 – 57, 1982.
58. P. Moon and A. Martin. Better heuristics for economic search experimental and simulation evidence. *Journal of Behavioral Decision Making*, 3(3):175–193, 1990.
59. K. Natarajan, M. Sim, and J. Uichanco. Tractable robust expected utility and risk models for portfolio optimization. *Mathematical Finance*, 20(4):695–731, 2010.
60. H. Oinas-Kukkonen. Behavior change support systems: A research model and agenda. In T. Ploug, P Hasle, and H. Oinas-Kukkonen, editors, *Persuasive Technology*, volume 6137 of *Lecture Notes in Computer Science*, pages 4–14. Springer Berlin Heidelberg, 2010.
61. G. Paolacci, J. Chandler, and P. Ipeirotis. Running experiments on amazon mechanical turk. *Judgment and Decision Making*, 5(5):411–419, 2010.
62. D. J. Power and R. Sharda. Decision support systems. *Springer Handbook of Automation*, pages 1539–1548, 2009.
63. J. Quiggin. A theory of anticipated utility. *Journal of Economic Behavior & Organization*, 3(4):323–343, 1982.
64. M. Rabin. Psychology and economics. *Journal of Economic Literature*, pages 11–46, 1998.
65. A. Rapoport and T. S Wallsten. Individual decision behavior. *Annual Review of Psychology*, 23(1):131–176, 1972.
66. D. A. Redelmeier and A. Tversky. Discrepancy between medical decisions for individual patients and for groups. *The New England journal of medicine*, 322(16):1162, 1990.
67. J. H. Roberts and J. M. Lattin. Development and testing of a model of consideration set composition. *Journal of Marketing Research*, pages 429–440, 1991.
68. J. H. Roberts and G. L. Lilien. Explanatory and predictive models of consumer behavior. *Handbooks in Operations Research and Management Science*, 5:27–82, 1993.
69. A. Rosenfeld and S. Kraus. Modeling agents based on aspiration adaptation theory. *Autonomous Agents and Multi-Agent Systems*, 24(2):221–254, 2012.
70. P. A. Samuelson. Risk and uncertainty: A fallacy of large numbers. *Scientia 6th Series*, pages 1–6, 1963.
71. D. Sarne, A. Elmalech, B. J. Grosz, and M. Geva. Less is more: restructuring decisions to improve agent search. In *The 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS-2011)*, pages 431–438, 2011.

72. P.J. Schoemaker. The expected utility model: Its variants, purposes, evidence and limitations. *Journal of Economic Literature*, pages 529–563, 1982.
73. A. Schotter and Y.M. Braunstein. Economic search: an experimental study. *Economic Inquiry*, 19(1):1–25, 1981.
74. D. Schunk and J. Winter. The relationship between risk attitudes and heuristics in search tasks: A laboratory experiment. *Journal of Economic Behavior Organization*, 71(2):347 – 360, 2009.
75. G. Shackle. *Decision, Order and Time in Human Affairs*. Cambridge University Press, 1969.
76. I. Sheena. *The Art of Choosing*. Twelve, March 2010.
77. C. Sierra, N. R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In *Proceedings of the 4th International Workshop on Intelligent Agents IV, Agent Theories, Architectures, and Languages*, ATAL '97, pages 177–192. Springer-Verlag, 1998.
78. H. A. Simon. Rational choice and the structure of the environment. *Psychological Review*, 63(2):129–38, 1956.
79. H. A. Simon. Theories of bounded rationality. *Decision and organization: A volume in honor of Jacob Marschak*, pages 161–176, 1972.
80. E. H. Simpson. The interpretation of interaction in contingency tables. *Journal of the Royal Statistical Society*, 13(2):238–241, 1951.
81. J. Sonnemans. Strategies of search. *Journal of economic behavior & organization*, 35(3):309–332, 1998.
82. C. Starmer. Developments in non-expected utility theory: The hunt for a descriptive theory of choice under risk. *Journal of Economic Literature*, 38(2):332–382, June 2000.
83. T. Tanaka, C. Camerer, and Q. Nguyen. Risk and time preferences: Linking experimental and household survey data from vietnam. *American Economic Review*, 100(1):557–71, 2010.
84. R. H. Thaler and E. J. Johnson. Gambling with the house money and trying to break even: The effects of prior outcomes on risky choice. *Management science*, 36(6):643–660, 1990.
85. R. H. Thaler and C. R. Sunstein. *Nudge: Improving decisions about health, wealth, and happiness*. Yale University Press, 2008.
86. A. N. Thorndike, L. Sonnenberg, J. Riis, S. Barraclough, and D. E. Levy. A 2-phase labeling and choice architecture intervention to improve healthy food and beverage choices. *American Journal of Public Health*, 102(3):527–533, 2012.
87. A. Tversky and D. Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5(4):297–323, 1992.
88. D. H. Wedell. Evaluations of single-and repeated-play gambles. *Wiley Encyclopedia of Operations Research and Management Science*, 2011.
89. D. H. Wedell and U. Böckenholt. Contemplating single versus multiple encounters of a risky prospect. *The American Journal of Psychology*, pages 499–518, 1994.
90. M. L. Weitzman. Optimal search for the best alternative. *Econometrica*, 47(3):641–54, May 1979.
91. www.sics.se/tac.
92. E. S. Yu. Evolving and messaging decision-making agents. In *Proceedings of the fifth international conference on Autonomous agents*, pages 449–456, 2001.

Appendix

A - Optimal search strategy for the problem with multi-rectangular distribution functions

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Based on Weitzman's solution principles for the costly search problem [90] we constructed the optimal search strategy for the multi-rectangular distribution function that was used in our experiments (see Section 5). In multi-rectangular distribution functions, the interval is divided into n sub intervals $\{(x_0, x_1), (x_1, x_2), , .., (x_{n-1}, x_n)\}$ and the probability distribution is given by $f(x) = \frac{P_i}{x_i - x_{i-1}}$ for $x_{i-1} < x < x_i$ and $f(x) = 0$ otherwise, $(\sum_{i=1}^{n} P_i = 1)$. The reservation value of each opportunity is calculated according to:

$$c_i = \int_{y=0}^{r_i} (r_i - y) f(y) dy \tag{9}$$

Using integration by parts eventually we obtain:

$$c_i = \int_{y=0}^{r_i} F(y) \tag{10}$$

Now notice that for the multi-rectangular distribution function: $F(x) = \sum_{i=1}^{j-1} P_i + P_j(x - x_i)/(x_{i+1} - x_i)$ where j is the rectangle that contains x and each rectangle i is defined over the interval (x_{i-1}, x_i). Therefore we obtain:

$$\begin{aligned} c_i &= \int_{y=0}^{r_i} \Big(\sum_{i=0}^{j-1} P_i + \frac{P_j(y - x_i)}{(x_{i+1} - x_i)} \Big) dy = \qquad (11) \\ &\sum_{k=1}^{j-1} \int_{y=x_{k-1}}^{x_k} \Big(\sum_{i=1}^{k-1} P_i + \frac{P_k(y - x_{k-1})}{x_k - x_{k-1}} \Big) dy + \int_{y=x_{j-1}}^{r_i} \Big(\sum_{i=1}^{j-1} P_i + \frac{P_j(y - x_{j-1})}{x_j - x_{j-1}} \Big) dy = \\ &\sum_{k=1}^{j-1} \Big((x_k - x_{k-1}) \sum_{i=1}^{k-1} P_i + \frac{P_k((x_k)^2 - 2x_k x_{k-1} - (x_{k-1})^2 + 2(x_{k-1})^2)}{2(x_k - x_{k-1})} \Big) + \\ &(r_i - x_{j-1}) \sum_{i=1}^{j-1} P_i + \frac{P_j((r_i)^2 - 2r_i x_{j-1} - (x_{j-1})^2 + 2(x_{j-1})^2)}{2(x_j - x_{j-1})} = \\ &\sum_{k=1}^{j-1} \Big((x_k - x_{k-1}) \sum_{i=1}^{k-1} P_i + \frac{P_k(x_k - x_{k-1})}{2} \Big) + (r_i - x_{j-1}) \sum_{i=1}^{j-1} P_i + \frac{P_j(r_i - x_{j-1})^2}{2(x_j - x_{j-1})} \end{aligned}$$

From the above equation we can extract r_i which is the reservation value of opportunity i.

B - Interesting strategies

Among the more interesting strategies, in the set of agents received, one may find:

- Use a threshold for the sum of the costs incurred so far for deciding whether to query the next server associated with the lowest expected queue length or terminate search.
- Querying servers from the subset of servers in the 10-th percentile, according to server's variance, from highest to lowest, and terminating if these are all queried or if a value that is lower than the mean of all remaining servers in the set was obtained.
- Querying the server with the smallest expected waiting time. Then, if the value obtained is at least 20% higher than the second minimal expected query time then querying the latter, and otherwise terminating the exploration process and assigning the job to the first queried server (i.e., querying at most two servers).
- Sorting the servers by their highest probability mass rectangle and querying the server for which that rectangle is defined over the smallest interval. Than sequentially querying all the servers according to their sum of expected queue length and querying cost until none of the remaining servers are associated with a sum smaller than the best value found so far.
- Querying the first out of the subset of servers associated with the minimum sum of querying cost and expected value. If the value received for the first is greater than its expected value then querying the second; otherwise terminating.

- Sorting the servers by their expected value and querying cost sum. Querying according to the sum, from the lowest to highest, and terminating unless the probability that the next to be queried will yield a value lower than the best found so far is at least 60%.
- Sorting the servers by their expected value and querying cost sum. Querying according to the sum, from the lowest to highest, and terminating unless the difference between the sum of the next to be queried and the first that was queried is less than a pre-set threshold.
- Querying the server that its variance is the lowest among the group of 30% of the servers that are associated with the lowest expected value.