

Designing Tools for Supporting User Decision-Making in e-Commerce

Alistair Sutcliffe and Faisal Al-Qaed

Manchester Business School
University of Manchester,
Booth Street West
Manchester M15 3 BP
UK

Abstract. The paper describes a set of tools designed to support a variety of user decision-making strategies. The tools are complemented by an online advisor so they can be adapted to different domains and users can be guided to adopt appropriate tools for different choices in e-commerce, e.g. purchasing high-value products, exploring product fit to users' needs, or selecting products which satisfy requirements. The tools range from simple recommenders to decision support by interactive querying and comparison matrices. They were evaluated in a scenario-based experiment which varied the users' task and motivation, with and without an advisor agent. The results show the tools and advisor were effective in supporting users and agreed with the predictions of ADM (adaptive decision making) theory, on which the design of the tools was based.

Keywords. Decision support, e-commerce, recommenders, advisor agents.

1 Introduction

Users face many different choices and tasks when shopping on the Internet, although support tools tend to fall into only two categories: ranked list matching tools as exemplified by the Google search tool, and recommenders as found in Amazon.com's website. Unfortunately few systems provide more than one tool so users' choice is restricted; furthermore, the available tools rarely match users' natural strategies. Since one of the principles of HCI design is to create interfaces which match the user's task and natural way of working [1, 2], design of e-commerce decision-support tools appears to be somewhat limited.

Decision-support tools range from recommenders [3] to comparison matrix tools and interactive visualisation-based tools for exploring choices [4]. Each tool assumes that a particular user strategy will be adopted in decision-making and search tasks; however, users employ multiple decision-making strategies which they mix in a flexible manner [5]. The conventional user-centred design approach is to specify a task model and then base the interface and dialogue design on the task model. When faced with multiple strategies the usual response is to design configurable or

adaptable interfaces [2, 6, 7]. However, when user strategies become nearly as complex as the task itself, the adaptation/adaptive design approach becomes difficult to sustain. An alternative response is to adopt a toolbox approach to design, as found in graphical drawing toolkits which have discrete tools for lines, shapes, textures, etc. that can be combined in a flexible manner. In previous work we developed an advisor agent to help users make decisions with a single tool based on Shneiderman's alpha slider design, and demonstrated that the advice was effective [8]. In this paper we explore the multiple tool paradigm with advisor agents for interactive decision support in e-commerce tasks.

A secondary motivation for this research is to explore the effectiveness of task-based advice. We investigate decision-making advice based on strategies from the Adaptive Decision Making (ADM) theory [9] with online advisors to help users not only to follow the appropriate cognitive strategy but also select effective tool support for their strategies. Since users tend to access help and explanation systems when they perceive the task to be difficult, we investigate the effectiveness of avatar agents for delivering advice.

The paper is organised as follows: the next section provides a brief background of related research, then the design of the decision-support toolkit and advisor agent is described. This is followed by explanation of the design, methods and results of the evaluation experiment which compared use of the toolkit with and without the advisor agent. The discussion reviews the experimental findings and concludes with plans for future research.

2 Related Work

Personalisation and recommender systems have tailored choices to users by modelling individual consumers' preferences [7, 10-12], one of the most popular being Amazon.com [13] which provides personalised recommendations according to consumers' profiles, ratings of particular products and purchase histories. The SETA toolkit supports creation of e-shops and recommends products best fitting users' needs as well as adapting the layout of store catalogues to user preferences and expertise [14-15], while SYSKILL and WEBERT [16] recommend web pages from users' feedback on viewed pages which the system uses as training data for learning user preferences. Recommenders have been a popular approach to help consumer choice [17], although Shneiderman has regarded decision making as an interactive search process supported by iterative querying and visualisation [4, 18]. Haubl and Trifts [19] argue that people follow a two-stage decision-making process, noting that while recommenders can support the initial screening of alternatives, comparison matrix tools are necessary for in-depth evaluation of options to help users make actual decisions. Pereira [20] investigated the interaction between the search strategy and users' product knowledge, using two strategies from ADM theory, Elimination By Aspects (EBA) for initial filtering and the Weighted ADDition method (WADD) for comparison and trade-off analysis, supported by personalised recommenders and hypertext for browsing alternatives. Subjects with little product knowledge preferred recommenders whereas subjects with high product knowledge preferred WADD and

EBA strategies. Although user needs and decision strategies have been considered, few adaptable decision support tools have been produced for e-commerce.

3 Toolkit and System Architecture

A two-tier client server architecture was adopted (see figure 1) with the client containing eight tools supporting different browsing and search functions, a configuration user interface which captures data for the tool recommender, and the advisor agent.

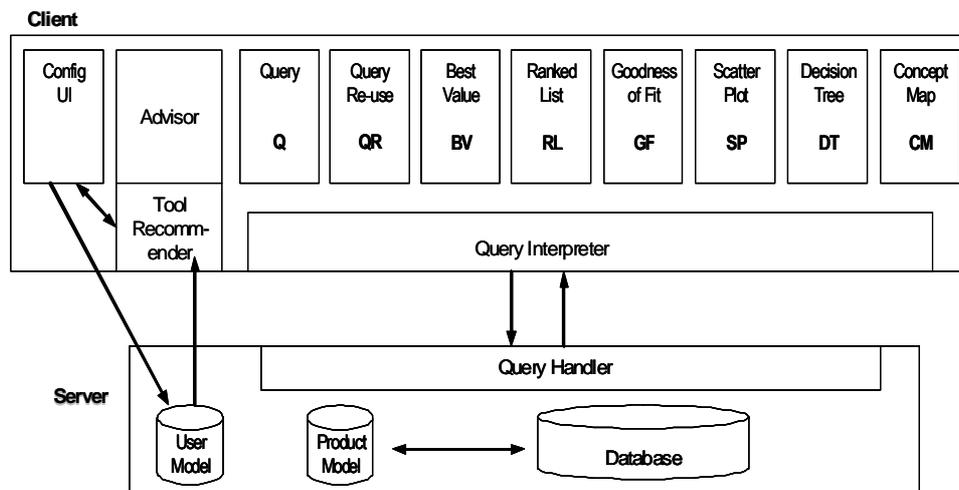


Fig. 1. System architecture of the Adaptive Decision Support System (ADSS)

The system design was informed by Adaptive Decision Making Theory [5] which considers the decision-making strategies that people should adopt according to the decision-making task, their knowledge of the domain, and contextual constraints such as time and motivation. In more complex tasks and domains compensatory (trade-off) strategies are predicted, while in less complex domains non-compensatory strategies are adopted, e.g. filter on single attribute, rely on memory of previous choices, etc. Three tools support non-compensatory search strategies with minimal user interfaces so these are not described in further detail: a template-based keyword search for products, a query reuse tool which allows previous queries to be saved and reused by picking from a menu list, and a recommender which selects the best value options from a range of pre-set queries. Non-compensatory strategies focus decisions on a single attribute choice (e.g. cost) without any trade-off decisions. Three tools support compensatory strategies with slider controls so users can interactively assess different range-value setting in queries with different feedback displays (ranked list, goodness-of-fit and scatterplot); and the final two tools support browsing-style exploration (decision tree and concept maps). Examples of compensatory strategies in ADM theory are filtering choices (Elimination by Aspects) and WADD weighted addition

decisions which trade off different attribute values. We implemented the eight tools to cover the diversity of decision-making strategies described in ADM theory and with the application scenario that ADSS would act as an Advisor plug that could be configured for several different e-commerce sites, so it had to adapt to different product domains as well as user constraints and knowledge. All the tools, the query interpreter and the tool recommender were implemented as JavaAplets and HTML. The advisor agent was developed using the Guile 3D studio agent tool to script the advice dialogue.

The server side houses the application database, the user model populated by the configuration dialogue and the product model which contains meta-data on the application database. A query handler communicates with the client-side query interpreters, calling database search functions implemented as a MySQL server. Server-side functions were implemented as Java Servlets hosted on Tomcat Server 3.5.

3.1 Tool Descriptions

Three tools share the slider-based query controls to support more compensatory, value-range exploration of choices, based on the alpha slider paradigm [4]. The ranked list tool illustrated in figure 2 implements slider-based querying with user controls for ranking the results for each variable.

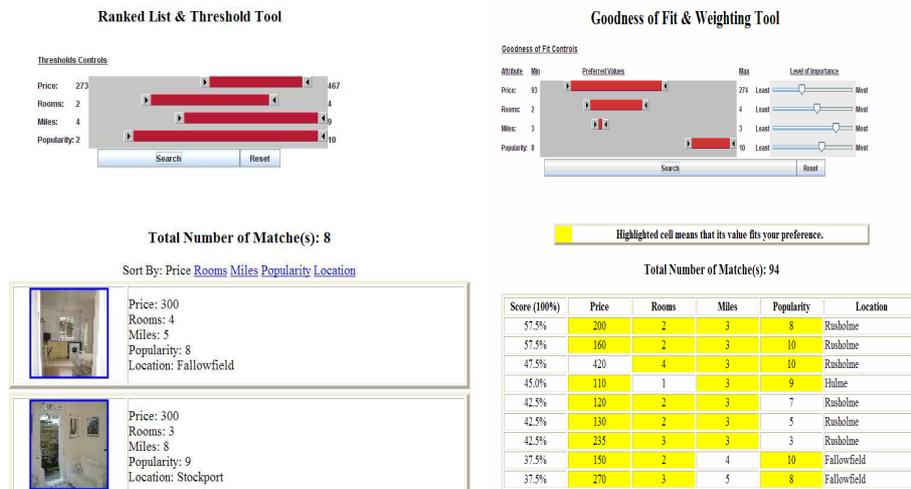


Fig. 2. Ranked list tool (left) and Goodness-of-fit tool (right)

The tool can be configured with sliders for any numeric attributes in the dataset with a simple palette picking list.

The goodness-of-fit tool (figure 2) augments the ranked list design by post-processing exact-match and near-miss query results according to how closely they fitted the search criteria. The search criteria can be biased by user-configurable

weights so different rankings can be produced reflecting search-variable importance as well as closeness-of-fit to search parameters.

The scatterplot tool adds a dynamically updated display to the functionality of the basic ranked list tool (see figure 3) so the location of the search results can be inspected on a map. Results are also displayed in table format and can be sorted by clicking on the column headings. As with the other tools, the scatterplot is configurable, so any attributes can be selected for the sliders. If non-spatial attributes are chosen a scatterplot graph display is used to show the results within value coordinate space, following Shneiderman's starfield design.

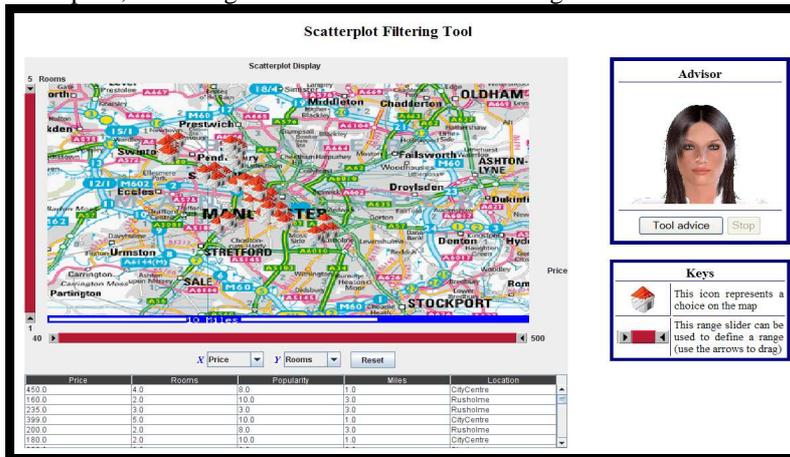


Fig. 3. Scatterplot tool showing the map configuration

Two tools support exploratory search: first, the decision tree, illustrated in figure 4, displays queries in tree format with pre-set value-range splits, allowing the user to walk the tree to reach different queries and their results in terminal branches. The tree can be reconfigured to choose query attributes, range splits and the query order.

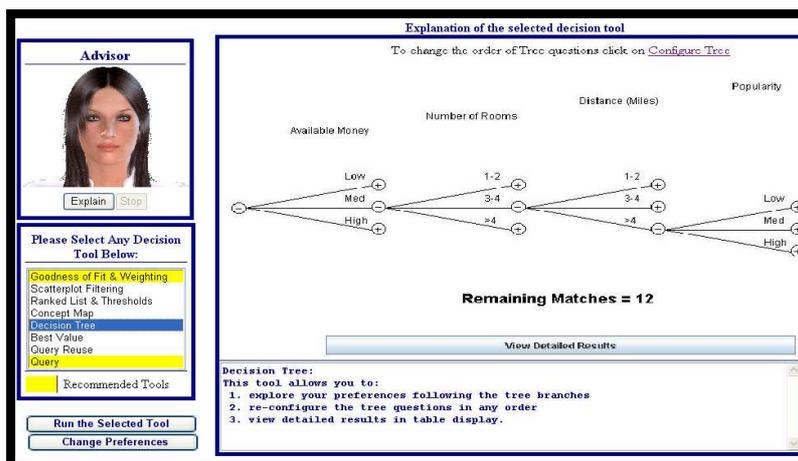


Fig. 4. Decision tree tool, also showing the tool recommender and advice

The second exploration tool is a conceptual map (see figure 5), which displays a flattened decision tree with range-split leaves. Queries can be formed by picking any combination of leaf nodes and selecting areas on the map. Results are displayed on the map as well as in a table.

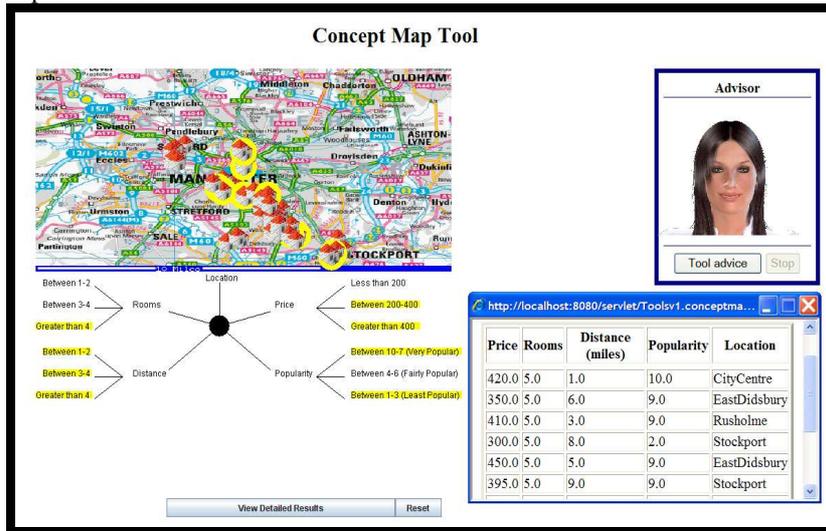


Fig. 5. Conceptual map tool

3.2 Advisor Agent and Tool Recommender

The advisor agent illustrated in figures 2-5 provides advice on operating the appropriate tool, general search strategy advice and recommendations for selecting tools, tailored according to the user's answers in the configuration dialogue. The configuration dialogue asks the user to rate their motivation on a three-point scale and select either direct search or browsing exploration preference for their task. This information is combined with the product model which categorises the configured database on a 3-point scale for product value (unit cost), complexity and volume of products in the database. The tool recommender highlights one or more tools determined by a set of rules, based on ADM theory. An example rule to illustrate the format is:

IF User <Motive = High, Task = Explore> AND Product <Value = High, Complexity = Medium, Volume = High> THEN Recommend <Concept-Map, Decision Tree, Scatterplot>

In this case compensatory exploration-support tools are recommended; for similar products with search tasks, ranked list-related tools are suggested; when user motivation is low, non-compensatory tools are preferred (best value, query, query reuse), etc. Speech advice is generated from text scripts which can be replayed or cancelled by the user.

4 Experimental Evaluation

The multi-tool environment was experimentally evaluated to assess the effect of advice and user tool/strategy choice. 24 subjects (14 males and 10 females, age range 20-39) from the University of Manchester participated in the experiment. Most subjects were postgraduate students with considerable Internet search experience. The subjects were paid £10 for their participation. A between-groups design was used to assess the effect of task and tool-use advice; the Advisor group were provided with tool recommendations, task-related advice and tool-use help, while the Control group had no advice or help. The experimental procedure was (i) brief training on using the eight decision tools and the advisor (in the Advisor group); (ii) experimental task with four scenario variations; (iii) post-test questionnaire; and (iv) debriefing interview. The task was to find apartments near the University of Manchester with the following scenarios, counterbalanced between subjects in both conditions:

Scenario 1: Low Motivation/Time and General Exploration

Assume that you do not have clear preferences, you have little time and you need to quickly find an apartment taking into account that the deadline for signing the apartment contract is closing soon. Find any apartment that would be acceptable.

Scenario 2: High Motivation/Time and General Exploration

Assume that you do not have clear preferences; you have plenty of time and you need to make sure that you find the most suitable apartment. Find an apartment that you really like.

Scenario 3: Low Motivation/Time and Search

Assume that you have little time and you need to quickly find an apartment taking into account that the deadline for signing the apartment contract is closing soon. Your preferences are for an apartment that has at least 2 bedrooms, its weekly price is less than £200 and is as close as possible to the university.

Scenario 4: High Motivation/Time and Search

Assume that you have plenty of time and you need to make sure that you find the best apartment you can. Find an apartment that has at least 3 bedrooms, its weekly price is less than £200 and is as close as possible to the university. Also, the most popular apartment is preferred. You should do more search evaluations and make sure the apartment is really suitable by checking the results.

No time limit was set for task completion and all subjects produced queries to complete the task. In the Advisor group, the subjects had to answer two configuration questions on their perceived motivation level (high/low) and the expected task (search/explore) after viewing the test scenario. Based on the configuration answers, the Advisor recommended one or more tools using rules was based on ADM theory [5]. In the Control group, the subjects had to select tools without any recommendations. The independent variables, manipulated in the four scenarios, were perceived motivation based on available time (high/low) and task type (explore/search). The dependent variables were number of tools used, performance times, tools selected, queries generated, usability errors and tool usage patterns.

5 Results

5.1 Number of Tools Used

Overall, more tools were used in the Control than in the Advisor condition (Mann-Whitney $U=18$, $df=1$, $p=0.001$); and this difference was also found when High-Motivation (2,4; $U=19$, $df=1$, $p=0.001$) and Low-Motivation (1,3; $U=33$, $df=1$, $p<0.05$) were compared with data for explore and search tasks. When analysed by task (motivations merged) more tools were used in the control condition for both the explore (1,2) - ($U=34$, $df=1$, $p<0.05$), and search (3,4) - ($U=24.5$, $df=1$, $p=0.005$), tasks. When differences were compared between groups for each scenario, more tools were used in the Control condition for both high motivation scenarios (4 Search, Mann-Whitney $U=35$, $df=1$, $p<0.05$) and (2 Explore, Mann-Whitney $U=35$, $df=1$, $p=0.052$), but not in the Low-Motivation scenarios (1 and 3); see figure 6. Within both Control and Advisor groups, more tools were used in the High-Motivation (2,4) than in the Low-Motivation scenarios (1,3; Wilcoxon test, $p<0.005$). However, no task (explore/search scenarios) differences were apparent in either condition, so it appears that system recommendation narrows the choice of tools, while motivation increases the diversity of tools used in both conditions.

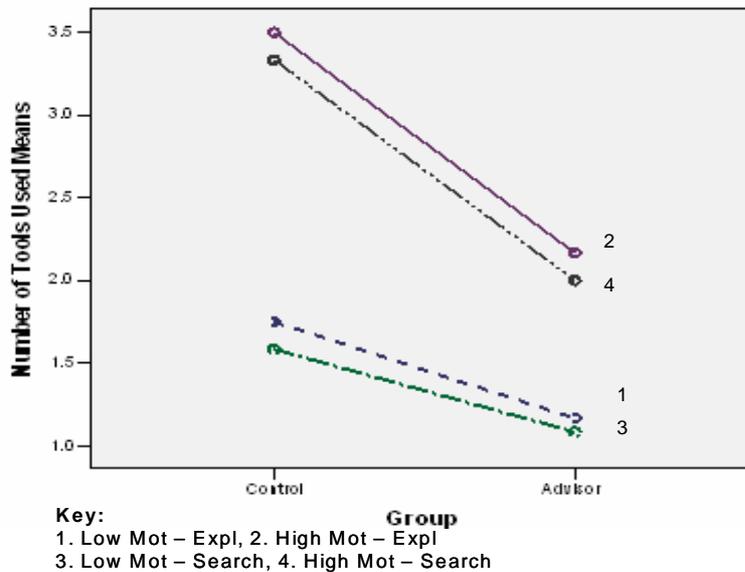
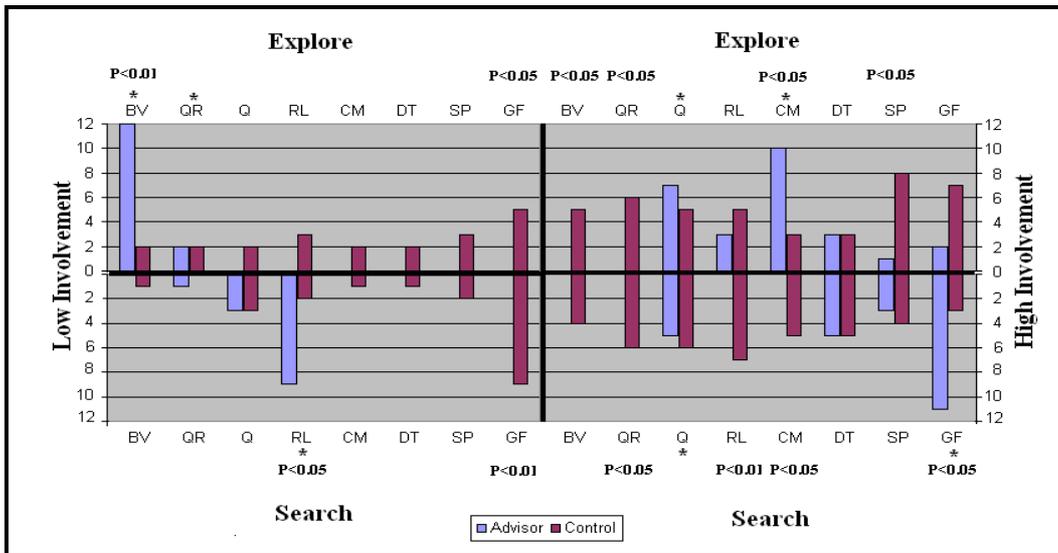


Fig. 6. Number of tools used

When individual scenarios were compared within the Control group, more tools were used with higher motivations in both tasks (1 & 3) v. (2 & 4) $p<0.01$, Wilcoxon matched pairs test, and the same differences were found in the Advisor condition ($p<0.01$, Wilcoxon). However, no differences were apparent between scenarios with different tasks (explore/search) at the same motivation level in both groups, so users

appear to follow similar strategies and tool use for search and explore oriented decisions.

Figure 7 illustrates the subjects' tool selections in the four scenarios. Dark bars in the figure represent subjects' tool usage in the Control group (no advice) and light bars represent subjects' tool usage in the Advisor group (with advice). The left/right quadrants map to the subjects' level of motivation (low and high) while the upper/lower quadrants map to the subjects' task (explore and search). The (*) indicates the recommended tools in each scenario.



(Note: (*) indicates the recommended decision tools; only significant differences are reported, using Binomial test)

Fig. 7. Tool usage frequencies in different scenarios, for tool codes see figure 1 or Table 2.

The left-hand (low motivation) quadrants illustrate little effect of the task with a similar tool use distribution in the Control condition; however, users followed the system advice and selected the recommended tools (best value (BV) for search and ranked list (RL) for explore), with no use of goodness-of-fit (GF), in spite of their preference in the Control condition.. However, the query reuse tool recommendation was partially ignored, possibly because the BV tool appeared first in the recommendation list. The high-motivation scenarios in the right-hand quadrants also show the impact of system recommendations, although a wider selection of tools were used even in the Advisor condition. No particular differences were apparent in the Control condition apart from a slight preference for scatterplot (SP) and goodness-of-fit in explore. When users were not guided they tended to use more compensatory and exploratory support tools (GF, RL, SP, CM), especially in the high-motivation scenarios. Decision tree and scatterplot tools were used in spite of the system recommendations. In debriefing interviews subjects reported they found the scatterplot easy to use for querying and the decision tree gave a good overview for both tasks.

5.2 Queries Generated

More queries were submitted by the Control group than by the Advisor group ($p < 0.05$, Binomial test on totals), and following the pattern of previous results, high-motivation scenarios produced more queries than the low-motivation scenarios ($p < 0.05$), in both groups; see table 1. The Control group created more queries than the Advisor group, especially in Scenario 1 ($p < 0.0001$) and a few more in Scenario 2 ($p < 0.05$), but differences were not significant in the other two scenarios. There were no differences in query totals between the tasks for either group.

Table 1. Queries Submitted by Scenario in both conditions
L, H = Low, High Motivation; E, S = Explore, Search

	Scenario 1: L-E	Scenario 2: H-E	Scenario 3: L-S	Scenario 4: H-S	Total
Control	28	86	22	63	199
Advisor	2	68	16	54	152

Advice focused tool usage and encouraged users to match tools to the tasks more effectively. Although lower query volumes in the Advisor group might be taken as evidence of more efficient usage in the search scenarios (2, 4), the totals were lower, and inter-group differences just failed to reach significance: $p = 0.52$ in scenario 4, ns in scenario 3. The lower query volumes in the exploration scenarios (1 and 2) might be seen as less productive; however, the CM and DT tools afforded inspection-based exploration in both conditions, so fewer queries may have been necessary.

Table 2. Queries submitted by tool type in each condition
*Sig = Binomial test on totals * $p < 0.05$, ** < 0.01*

	Control	Advisor	Sig
Best Value	12	12	ns
Query Reuse	25	4	**
Query	22	26	ns
Ranked List	29	18	*
Concept Map	26	30	ns
Decision Tree	16	16	ns
Scatter Plot	43	22	*
Goodness of Fit	38	24	*
Total	199	152	*

In the Control group more queries were generated with the SP tools, possibly reflecting the ease of slider-based query formulation (see table 2). The Advisor group focused tool choice in both tasks, generally following system recommendations, although the subsequent rank-order query generation frequency was GF>RL>CM>QR. In the Control condition, queries were evenly distributed among the tools apart from decision tree (DT), which may have been used more for inspection in both conditions. Within the Advisor group, more frequent queries were created with the concept map (CM) tool, followed by Q>GF>SP, with more marked inter-task differences in use. The effect of advice appears to be to reduce query volumes for query reuse (QR), RL, SP and GF tools, reflecting more focused tool use ($p<0.05$ Binomial test). Thus, when the users were guided they showed more focused patterns of tool use, closely matching the scenario tasks (e.g. CM in scenario 2 explore, GF in scenario 4, search).

The main effect of the advice was to shift use from the SP, GF, RL and Q tools, which supported compensatory decision making, to the CM for an exploration overview.

5.3 Search Times and Performance

All subjects completed the search tasks correctly and there were no significant differences in the subjects' search times between the groups in all scenarios (ANOVA), even though subjects in the Advisor group had an additional task in configuring the system and listening to/reading advice for tool recommendations and tool-use help.

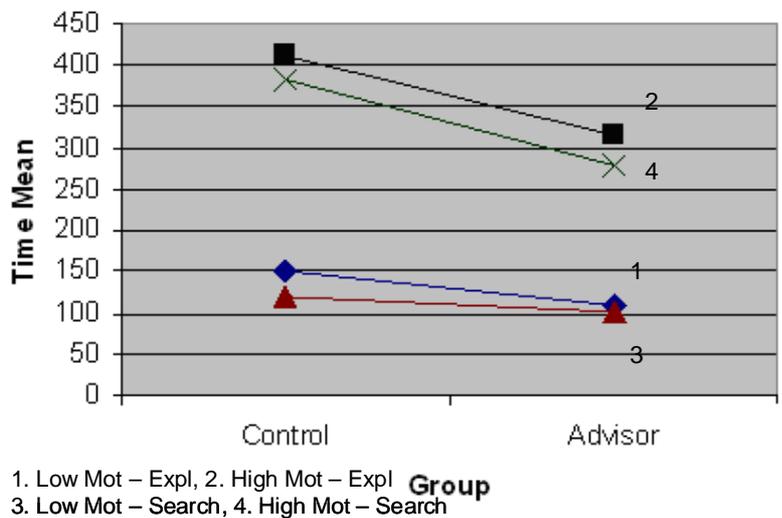


Fig. 8. Search performance (Advisor vs. Control) – time in seconds, excluding configuration and advice accesses time

When configuration and advice access times were subtracted from total completion times, the Advisor condition was significantly faster than the Control group ($F=14.73$,

df=1,23, p=0.001) and for each scenario (p<0.05); see figure 8. Following the pattern of tool usage, the motivation manipulation also produced significant differences in completion times between scenarios 1, 3 and 2, 4 for both conditions (t tests, p<0.5). This shows that task-related and operational help was effective in improving the subjects' search performance.

5.4 Usability Problems

There were few usability problems overall, although the Control group did experience 14 errors compared with 8 in the Advisor group. The errors were either problems in using the slider controls or failing to find the sort/rank functions on the table headings. All errors were eventually overcome in both groups. In the Advisor group 15 critical incidents were reported when users complained of difficulty in mapping the scenarios to the answers in the configuration dialogue.

5.5 Debriefing Interviews

In debriefing interviews, positive comments were made about the system providing different ways of searching that matched different situations (Advisor group: 83.3%, Control group: 75%), being better than the current e-commerce tools (A: 75%, C: 66.7%). Most subjects reported that the tools were easy to use (100%, 83.3%) and that the tools complemented each other (75%, 58.3%). The Advisor group reported that the highlighted (recommended) choices helped them to decide which tool to use in different scenarios (83.3%). Use of sliders for queries rather than entering values into text boxes also received many favourable comments (75%, 58.3%).

Negative comments came mainly from the Control group, highlighting the need for tool recommenders; for example, too many tools, confused over which one to select (91.7%); some tools require help (e.g. DT, CM, SP filtering tools: 75%); and advice on which tool is appropriate for each task would be helpful (91.7%). A few subjects in the Advisor group complained that the configuration questions were not clear (33.3%); however, they liked the agent appearance and noted that it drew their attention to the advice (91.7%), and preferred speech to text in the configuration advice (58.3%), even though the speech sounded artificial (66.7%).

6 Discussion

The environment described in this paper illustrates theory-led design, in that the design of the tools and the recommender was based on Adaptive Decision Making, sound cognitive theory describing user strategies for making choices. Our design approach follows in the toolkit tradition of HCI design in which tool support is targeted at user strategies within a complex task, hence building on research into adaptable and adaptive systems [6]. The toolkit can be exploited in two ways, first using the recommender as illustrated in this paper, in which the assumed implementation environment is adaptable decision support which could be portable across domains. This would require configuration of product models and database

querying modules for several domains, which could be provided with modest development effort. The second route follows a toolbox paradigm in which the recommender indicates appropriate tools for a specific application and a sub-set of the tools are configured for that domain.

The experimental evaluation demonstrated that the system advice and tool recommendations were effective in shaping user behaviour and improving the efficiency of decision making. Post-experiment questionnaires and interviews also demonstrated that the recommender and toolkit design was well received. While evidence for improved performance is limited, we can point to the reduced number of queries submitted for task completion and the shorter completion times, once allowance was made for the configuration dialogue. Since the configuration setup dialogue is a one-off investment for each application domain, over time the performance improvement payoff should be considerable. The tool recommendations influenced user tool choice following the strategies in ADM theory which were embedded in the recommender rules. The scenario manipulations also demonstrated that users follow the advice according to their motivation and the task context. Observed tool choice matched the system recommendations; however, occasionally users seemed to prefer using the scatterplot and decision tree in the high motivation-search tasks in spite of the system advice. This may reflect a sequential ordering in users' strategies, in which they explore the search space to check alternatives even when undertaking well specified searches. In our future work we will investigate recommendations for sequential and integrated use of the tools in both exploration and search tasks. Active training in trade-off strategies improves user exploration of options and satisfaction [3, 21], so another direction is to augment our advisor with more directed tuition for compensatory tools.

The limitations of the study lie in the possible suggestibility of our participants who may have followed the system recommendations from an experimental task conformance bias; however, we point to the positive comments and questionnaire rating which the participants made about the recommenders and their reports that, while provision of several tools was useful, they were more effective with a recommender. The same criticism can be levelled at the scenario manipulations; although in this case we point to the clear effect of the manipulations in tool choice, and the credibility checks we carried out in de-briefing interviews. The subjects' choice of decision-making strategies elicited from interviews not connected to the experimental task demonstrated that the selected strategies agreed with the observations in the experimental tasks.

In conclusion, the system proved to be an effective means of improving users' performance in complex decision making where multiple strategies will be necessary as rapid change in domains can influence task strategies. The toolkit recommender architecture has the potential for exploitation as both a configurable toolbox and a toolkit for flexible decision support across domains. From our results it appears that user-driven decision making with exploration as well as search is an important part of e-commerce functionality which needs to be balanced against the current trend for developing system initiative recommender systems.

7. References

- [1] Shneiderman, B. and C. Plaisant, *Designing the User Interface: Strategies for Effective Interaction*. 2004. Reading MA: Addison-Wesley.
- [2] Benyon, D., P. Turner and S. Turner, *Designing Interactive Systems: People, Activities, Contexts, Technologies*. 2004. Reading MA: Addison Wesley.
- [3] Pu, P. and P. Kumar. Evaluating Example-based Search Tools. In *Proceedings of the 5th ACM Conference on Electronic Commerce*. 2004. New York: ACM Press.
- [4] Ahlberg, C. and B. Shneiderman. Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In *Readings in Information Visualization: Using Vision to Think*. 1999. San Francisco: Morgan Kaufmann.
- [5] Payne, J.W., J.R. Bettman, and E.J. Johnson, *The Adaptive Decision Maker*. 1993. Cambridge: Cambridge University Press.
- [6] Fischer, G. User Modeling in Human-Computer Interaction, *User Modeling and User-Adapted Interaction*, 2001: vol. 11(1/2), pp. 65-86.
- [7] Flor, G.D.L., *User Modeling & Adaptive User Interfaces*. 2004. ILRT Research, University of Bristol, UK.
- [8] Al-Qaed, F. and A. G. Sutcliffe, Adaptive Decision Support System (ADSS) for B2C E-Commerce In *Proceedings 8th International Conference on Electronic Commerce*, 2006. 492-503, New York: ACM Press.
- [9] Corbin, R.M. Decisions that Might not get Made. In *Cognitive Processes in Choice and Decision Behavior*. 1980. Hillsdale, NJ: Lawrence Erlbaum Associates.
- [10] Hong, W., J.Y.L. Thong, and K.Y. Tam, The Effects of Information Format and Shopping Task on Consumers' Online Shopping Behavior: A Cognitive Fit Perspective. *Journal of Management Information Systems*, 2005. vol. 21(no. 3): pp. 149-184.
- [11] Hung, L.-P., A Personalized Recommendation System Based on Product Taxonomy for One-to-One Marketing Online. *Expert Systems with Applications*, 2005. vol. 29(no. 2): pp. 383-392.
- [12] Langley, P. User Modeling in Adaptive Interfaces. In *Proceedings of the Seventh International Conference on User Modeling*. 1999. Banff, Alberta: Springer.
- [13] Hildebrand, C., One to a Customer: Customer Relationship Management. 1999, *CIO Enterprise Magazine*.
- [14] Ardissono, L., A. Goy, G. Petrone, and M. Segnan, Personalization in Business-to-Customer Interaction. *Communications of the ACM*, 2002. vol. 45(no. 5): pp. 5253.
- [15] Ardissono, L., A. Goy, G. Petrone, and M. Segnan, A Multi-Agent Infrastructure for Developing Personalized Web-Based Systems. *ACM Transactions on Internet Technology*, 2005. vol. 5(no. 1): pp. 47-69.
- [16] Pazzani, M., J. Muramatsu, and D. Billsus. SYSKILL & WEBERT: Identifying Interesting Web Sites. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*. 1996. Portland: AAAI Press.
- [17] Herlocker, J.L., J.A. Konstan, L.G. Terveen, and J. Riedl, Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems*, 2004. vol. 22(no. 1): pp 5-53.
- [18] Shneiderman, B. Creating Creativity: User Interfaces for Supporting Innovation, *ACM Transactions on Computer-Human Interaction*, 2000: vol. 7(1), pp. 114-138.
- [19] Haubl, G. and V. Trifts, Consumer Decision Making in Online Shopping Environments: The Effects of Interactive Decision Aids. *Marketing Science*, 2000. vol. 19(no. 1): pp. 4-21.
- [20] Pereira, R.E., Optimizing Human-Computer Interaction for the Electronic Commerce Environment. *Journal of Electronic Commerce Research*, 2000. vol. 1(no. 1): pp. 23-44.
- [21] Pu, P. and L. Chen. Integrating Tradeoff Support in Product Search Tools for E-Commerce Sites. In *Proceedings of the 6th ACM conference on Electronic Commerce*. 2005. New York: ACM Press.