A Market-Based Approach to Recommender Systems

YAN ZHENG WEI, LUC MOREAU, and NICHOLAS R. JENNINGS
University of Southampton

Recommender systems have been widely advocated as a way of coping with the problem of information overload for knowledge workers. Given this, multiple recommendation methods have been developed. However, it has been shown that no one technique is best for all users in all situations. Thus we believe that effective recommender systems should incorporate a wide variety of such techniques and that some form of overarching framework should be put in place to coordinate the various recommendations so that only the best of them (from whatever source) are presented to the user. To this end, we show that a marketplace, in which the various recommendation methods compete to offer their recommendations to the user, can be used in this role. Specifically, this article presents the *principled* design of such a marketplace (including the auction protocol, the reward mechanism, and the bidding strategies of the individual recommendation agents) and evaluates the market's capability to effectively coordinate multiple methods. Through analysis and simulation, we show that our market is capable of shortlisting recommendations in decreasing order of user perceived quality and of correlating the individual agent's internal quality rating to the user's perceived quality.

Categories and Subject Descriptors: H.1.m [**Models and Principles**]: Miscellaneous; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*

General Terms: Algorithms, Design, Economics

Additional Key Words and Phrases: Recommender systems, auctions, marketplace

1. INTRODUCTION

The World Wide Web (the Web) [Berners-Lee et al. 1992] presents us with a vast array of information. Also, regardless of the metric used (i.e., growth in

This research is funded in part by QinetiQ and the EPSRC Magnitude Project (reference GR/N35816).

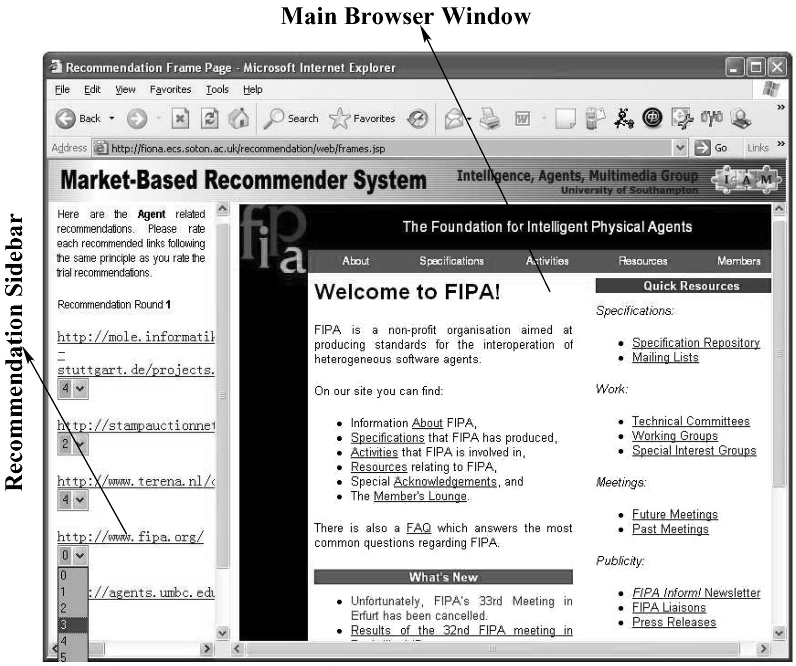
Authors' current addresses: Y. Z. Wei, British Telecom, Orion Building (MLB1 pp12), Adastral Park, Martlesham Health, Ipswich, U.K. IP5 3RE; email: yanzheng.wei@bt.com; L. Moreau, and N. R. Jennings (corresponding author), Intelligence Agents Multimedia Group, School of Electronics and Computer Science, University of Southampton, Southampton, U.K. SO17 1BJ; email: {L.Moreau, nrj}@ecs.soton.ac.uk.

the number of networks, hosts, users, or traffic), the Internet is growing at least 10 percent per month and the content of the Web grows by an estimated 170,000 pages daily [Turban et al. 2000, p. 495]. When taken together, these factors make it very hard to know what documents are out there, let alone find the ones that are most suitable for the task at hand. To address this information overload problem, a range of tools to assist with indexing, retrieving, searching, and filtering have been developed [Zamboni 1998; Pinkerton 2000; Howe and Dreilinger 1997; Yan and Garcia-Molina 1995]. However, while these tools can certainly assist in this endeavor, they are typically not personalized to individual users or their prevailing context [Sheth and Maes 1993]. Additionally, such tools still tend to have the weakness of either providing too much irrelevant information or missing relevant information [Goldberg et al. 1992].

To overcome these limitations, *recommender systems* have been advocated. Recommender systems help make choices among recommendations from all kinds of sources without the users needing to have sufficient personal experience of all these alternatives [Resnick and Varian 1997]. Thus, in this context a recommendation is viewed as a reference to an item that will be directed to the user who is looking for information. A typical recommender system aggregates and directs recommendations to appropriate recipients. Given this view, it can be seen that a recommender system's main value lies in information aggregation and its ability to match the recommendations with people seeking information. It differs from conventional filtering systems in that recommendations are based upon subjective values assigned by people, namely, the quality of items, rather than more objective properties (such as the text content of a document) of the items themselves [Resnick et al. 1994; Shardanand and Maes 1995]. Compared to a system that only has searching or other simple information filtering functionalities, recommender systems require less experience on the part of the user and less effort to specify and restrain their interests when querying and operating the system [Resnick and Varian 1997]. This is because recommender systems provide their users with recommendations that have been recognized as good (based on their previously expressed preferences or the preferences of other users with similar interests).

Against this background, this research is concerned with the problem of information overload on the Web and in how recommender systems can be used to help overcome this problem. In particular, it deals with the "where to go next" problem by presenting recommendations (represented as URLS) that are relevant to the users' current browsing context. This method is beneficial since users often ask questions such as "What else should I read?" and "Where do other people go from here?" By convention, such recommendations are usually displayed in a separate window without interrupting a user's current navigation (Figure 1 is an example of the system that we have built for this task).

To date, two typical kinds of filtering approaches have been used to produce recommendations: content-based and collaborative filtering (see Section 2 for more details). The former makes recommendations by analyzing the similarity between the contents of the items that are ready to be recommended and those


Main Browser Window
Recommendation Sidebar
Recommendation Frame Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://fiona.ecs.soton.ac.uk/recommendation/web/frames.jsp
Go Links
Market-Based Recommender System
Intelligence, Agents, Multimedia Group
University of Southampton
I A M
fipa
About Specifications Activities Resources Members
Quick Resources
What's New
0 1 2 3 4 5

Here are the **Agent** related recommendations. Please rate each recommended links following the same principle as you rate the trial recommendations.

Recommendation Round **1**

http://mole.informatik-stuttgart.de/projects.
4

http://stampauctionnet
2

http://www.terena.nl/c
4

http://www.fipa.org/
0

://agents.umbc.edu

The Foundation for Intelligent Physical Agents

Welcome to FIPA!

FIPA is a non-profit organisation aimed at producing standards for the interoperation of heterogeneous software agents.

On our site you can find:

- Information About FIPA,
- Specifications that FIPA has produced,
- Activities that FIPA is involved in,
- Resources relating to FIPA,
- Special Acknowledgements, and
- The Member's Lounge.

There is also a FAQ which answers the most common questions regarding FIPA.

What's New

- Unfortunately, FIPA's 33rd Meeting in Erfurt has been cancelled.
- Results of the 32nd FIPA meeting in

Quick Resources

Specifications:

- Specification Repository
- Mailing Lists

Work:

- Technical Committees
- Working Groups
- Special Interest Groups

Meetings:

- Future Meetings
- Past Meetings

Publicity:

- *FIPA Inform!* Newsletter
- FIPA Liaisons
- Press Releases

Fig. 1. Browser with recommendations. The main window displays the user's current context (the page being viewed). The side bar on the left is the output of the recommender system and displays a list of URLS in decreasing order of relevance to the user's current context.

that have previously been marked as liked by the user. The latter makes recommendations by putting forward items that have been deemed appropriate by people who have similar interests to the user. Based on these two techniques, a large number of recommendation filtering methods have been developed (again see Section 2 for more details). However, most conventional recommender systems share two major weaknesses:

(1) Each recommender system typically embeds some specific algorithm to compute correlations (the similarity of two relevant objects). However, there is no universally best way of doing this (and nor do we believe that there will ever be such a method). Rather, it is always the case that some methods are better under particular conditions and others are better under other conditions [Breese et al. 1998]. Given this, we believe the solution is to have a suite of recommendation methods available and to have the system automatically detect which one is the most appropriate in the prevailing context. However, such coordination is very difficult to attain, because the outputs from these diverse methods need to be compared.

(2) As ever more information is available on the Web, the pool from which recommendations can be made will continue to grow. However, users do not want correspondingly more recommendations to be presented (otherwise they will be overloaded). Thus, there is a need to be ever more selective and ensure that only the most appropriate recommendations are put forward.

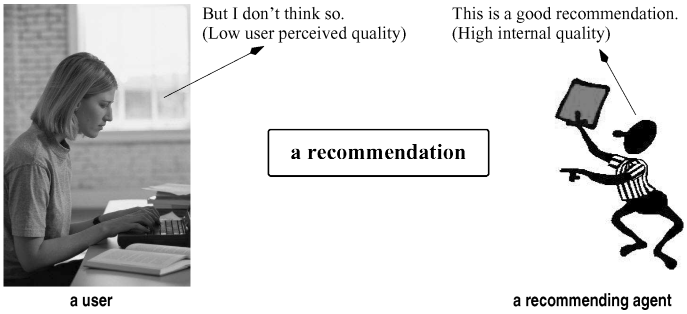


Fig. 2. Different valuations of quality.

Given these observations, we believe the best way forward in this area is to allow the multiple recommendation methods to coexist and to provide an overarching system that coordinates their outputs such that only the best recommendations (from whatever source or method) are presented to the user [Moreau et al. 2002]. To this end, a *market-based approach* is an efficient means of achieving such coordination because the problem of selecting appropriate recommendations to display in the sidebar space can be viewed as one of scarce resource allocation and markets are an efficient solution for this class of problems [Clearwater 1996; Wellman and Wurman 1998]. Moreover, the underlying economic theory provides an analytical framework for predicting aggregate behavior and designing individual information providers [Mullen and Wellman 1995]. Specifically, in this article, we report on the design and evaluation of a market-based system capable of recommending documents relevant to the users' current browsing context as a way of dealing with the problem of information overload.[1]

To deal with information overload, all recommender systems share the same objective of improving recommendation quality. However, most of the existing systems lack a means of (i) specifically defining the quality of recommendations from the viewpoint of the user and the various recommendation methods (since these may well differ); (ii) correlating these different qualities in a meaningful manner. In more detail, given a specific recommendation provided by a recommending agent with a specific recommendation method, a user's valuation of the recommendation may differ from that of the agent. For example, in Figure 2, a particular recommending agent might highly rate a recommendation and therefore wish to highlight it to the user. However, the user may see this as a poor-quality recommendation that is not very relevant. Given this situation, the quality of a recommendation can be viewed from two viewpoints. From the viewpoint of the user, how well a recommendation satisfies him (or her) is

[1] In this work, we are *not* concerned with developing new recommendations methods. Our aim is to efficiently coordinate existing methods so that the overall system produces the best information to the user (i.e., the performance of our system is reliant on the effectiveness of the constituent recommendation methods and what the market does is to allow the best recommendation to be highlighted). We do not compare the relative performance of the methods. Rather our concern lies with the fact that different methods make recommendations simultaneously and we let the user decide which recommendations are good (irrespective of the specific methods they are provided by).

termed the *user perceived quality* (UPQ). From the viewpoint of a recommending agent with a specific recommendation method, the relevance score it computes for a particular recommendation is termed its *internal quality* (INQ). Moreover, the INQ values produced by different methods can vary significantly from one another (even for the same document). Therefore, without a systematic means of relating the UPQ to the recommendation methods' INQs, it is very difficult to provide high-quality recommendations. In this research, the key challenge is the design of a reward mechanism (which reflects the user's satisfaction of the recommendations) so that the marketplace can effectively correlate these two values. In sum, the key role of our marketplace is to try to connect the INQ and UPQ values by imposing a reward regime that incentivizes different recommending agents to bid in a manner that establishes an appropriate correlation between these values and their bid prices. In this way, the marketplace can be viewed as a black box with recommendations provided by different recommending agents as the input and only a few best recommendations passed through to the user as the output.

In more detail, the work presented in this article advances the state of the art in the following ways. First, the marketplace that we designed for this task provides a method for coordinating the behavior of multiple recommendation methods with diverse measures of INQ. No other recommender system has attempted to incorporate multiple approaches in this way. Existing hybrid filtering systems do attempt to combine different techniques into one system. However, they do so in a rigid and predetermined way, rather than in a context-specific manner that depends on the user's feedback. Second, our market automatically optimizes the recommender system's performance so that it can shortlist recommendations in decreasing order of the user's preference. Specifically, the market works as a black box, with a large number of recommendations from various methods as the input and a small number of items as the output, and its performance in terms of presenting recommendations is always equivalent to that of the best method inside the black box. Third, the market design forces (incentivizes) the individual recommending methods to adapt their behavior so as to align their suggestions with the feedback received from the user. Thus, the market correlates the agents' internal valuation and the user's valuation of the recommendations by invoking a bidding and a rewarding regime. Fourth, the market is highly efficient and effective as an economic system. Specifically, it is Pareto-optimal, maximizes social welfare, and is stable and fair to all component methods integrated in the system (see Section 3 for details). In making these contributions, our aim is to establish, in principle, the viability of the market-based approach to recommender systems. Given this, the actual construction of a real system that operates with actual user inputs is beyond the scope of the current work and is left for the future.

The remainder of the article is structured in the following way. We first outline the related work in terms of diverse filtering approaches and market-based recommender systems in Section 2. Building on past work, we introduce the design of the marketplace from the perspectives of the protocol, the reward mechanism, and the bidding strategy in Section 3. We then evaluate the marketplace through simulation and demonstrate how it can correlate the component methods' INQs and the user's UPQs of the recommendations in Section 4. Finally, we conclude and point to future work in Section 5.

2. RELATED WORK

To date, a large number of recommendation techniques have been developed. These are, however, based mainly on *content-based* and *collaborative* filtering (although there is also some work on *hybrid* and *demographic* filtering techniques). Each of these categories will be examined in turn in the remainder of this section.

Content-based filtering approaches recommend items for the user based on the descriptions of previously evaluated items. Such approaches are widely used in making recommendations of information items. For example, Syskill recommends Web documents based on users' binary ratings ("hot" and "cold") of their interests [Pazzani et al. 1996] and Newsweeder helps users filter Usenet news articles by learning the user's profile based on his or her ratings [Lang 1995]. Generally speaking, however, content-based filtering approaches have a number of weaknesses in recommending good items. First, a user's selection is often based on the subjective attributes of the item [Goldberg et al. 1992], whereas content-based approaches are based on objective information about the items and do not take the user's perceived valuation of such subjective attributes into account [Montaner et al. 2003]. This makes it impossible to compute the relevance of items with no machine-parsable format (such as sound and video files). To this end, our market-based recommender system, by integrating both collaborative and content-based filtering methods, can meet the user's subjective requirements in the way that when subjective attributes are the user's interests, recommendations from collaborative (to be discussed in the next paragraph) component recommenders will be at the top of the shortlist (mutatis mutandis for objective attributes). Second, content-based filtering techniques do not have an inherent method for generating serendipitous finds [Shardanand and Maes 1995]. They tend to recommend more of what the user has already seen. In comparison, again, our market-based approach overcomes this by having different types of recommenders present.

Collaborative filtering techniques, on the other hand, match people with similar interests and then recommend one person's highly evaluated items to the others [Goldberg et al. 1992; Resnick et al. 1994]. Thus, rather than computing the similarity of items (which relies on machine analysis of content [Herlocker et al. 2000]), collaborative filtering computes the similarity of user's interests. This means that subjective data about items can be incorporated into recommendations (of the content-based approach). This, in turn, facilitates serendipitous new finds. In addition, collaborative filtering techniques can be used to recommend both machine-parsable items (such as textual articles [Terveen et al. 1997]) and non-machine-parsable ones (such as audio and video files [Shardanand and Maes 1995; Hill et al. 1995]). Indeed, perhaps the greatest strength of collaborative techniques is that they are completely independent of any machine-readable representations of the objects being recommended. Thus, they work well for complex objects such as music and movies where variations in taste are responsible for much of the variation in preference [Burke 2002].

Given these benefits, collaborative recommenders have been developed for many applications. For example, Ringo recommends music album and artists based on the word-of-mouth recommendations by weighting users' votes [Shardanand and Maes 1995], GroupLens helps people find Usenet news articles on a collaborative basis [Konstan et al. 1997], and MEMOIR assists people in finding other people (rather than documents) with similar interests [DeRoure et al. 2001]. However, collaborative filtering approaches also have a number of shortcomings. First, large numbers of people must participate so as to increase the likelihood that any one person will find other users with similar interests [Terveen and Hill 2001] (the sparsity problem). The difficulty of achieving a critical mass of participants makes collaborative filtering experiments expensive. Second, a user whose interests share little with others' will receive poor recommendations on a collaborative basis. An extreme case of this phenomenon happens when new users start off with nothing in their profiles of interests and must train a profile from scratch (the "cold start problem" [Resnick and Varian 1997]). Even with a start profile, there is still a training period before the profile accurately reflects the user's preferences [Maltz and Ehrlich 1995]. Third, these systems suffer from the "early-rater problem" [Montaner et al. 2003]: when a new item appears in the database, there is no way it can be recommended to a user until more information is obtained through another user either rating it or specifying which other items it is similar to.

As can be seen, both content-based and collaborative filtering have weaknesses. Moreover, these weakness tend to complement one another [Montaner et al. 2003]. Thus, hybrid filtering systems that integrate the two approaches have been advocated [Herlocker et al. 2000]. In a hybrid system, both objective and subjective properties of an item are taken into account in predicting its quality when making recommendations. For example, filterbots integrate content-based filtering techniques to build virtual users in the GroupLens collaborative system [Sarwar et al. 1998], the Fab collaborative system maintains user profiles by using content-based analysis [Balabanovic and Shoham 1997], Pazzani's system involves user collaborations to determine the ratings of predicted items and a content-based profile to compute similarity among users [Pazzani 1999], Popescul's system uses secondary data (e.g., document contents) to predict users' preferences in collaborative recommendations when there is a lack of user ratings [Popescul et al. 2001], and Claypoole's system employs separate collaborative and content-based recommenders and uses an adaptive weighted average of the two in making its selections (as the number of users accessing an item increases, the weight of the collaborative component tends to increase [Claypool et al. 1999]).

While hybrid systems can sometimes overcome the shortcomings of pure content-based and pure collaborative systems, with respect to the objective and subjective properties of recommendations, they do so in a rigid and predetermined manner. Specifically, such systems try to use one of the recommendation properties (either objectiveness or subjectiveness) to complement the weaknesses of the other when the latter does not work effectively. However, there is no automated way of determining in what circumstances which kind of properties (objective, subjective, or both) are relevant to a particular user in their current context. In contrast, by using the market to reward effective recommenders (irrespective of whether they use subjective or objective methods, or a combination of the two), our system dynamically tunes the relative importance of the methods according to the feedback received from the users.

The final type of filtering technique that has been used in recommender systems is demographic filtering. This approach uses descriptions of people (such as occupation, age, and gender) to learn the relationship between a single item and the type of people who like it [Krulwich 1997]. For example, a mature, sophisticated woman is likely to prefer an expensive leather jacket, whereas a teenage school girl may prefer a cheap denim one. However, this method has two principle shortcomings. First, it creates profiles by classifying users using stereotypical descriptions [Rich 1979]. Thus, the same items are recommended to people with similar demographic profiles. However, in many cases, the stereotypes are too general to generate good quality recommendations [Montaner et al. 2003]. Second, if the user's interests shift over time, demographic filtering does not adapt their profile [Koychev 2000]. For these reasons, demographic filtering is rarely used independently of the other filtering techniques.

In terms of combining different recommendation methods using a marketplace, the most related work to our own is that of Bohte et al. [2001, 2004]. Essentially, the main purpose of that work was to provide a mechanism for retail businesses to advertise their products. However, this was less concerned with the information retrieval and filtering. Specifically, they used a market to competitively allocate consumers' attention space in the domain of retailing. Here, the scarce resource is the consumer's ability to focus on a set of banners or products. This work developed an adaptive bidding strategy that the agents can use to learn the consumer's preferences. However, this work and our own use the market mechanisms in different ways to solve the resource allocation problem in recommender systems. The market in Bohte et al. [2001] was used only to coordinate agents' bidding. However, our market is used not only for this purpose, but also to coordinate the objectiveness and subjectiveness of recommendations and to correlate various recommendation methods' internal valuation of qualities to the user's actual interests.

In a somewhat related fashion, a number of Web portals and search engines, such as Overture (`www.overture.com`) and Google (`www.google.com`), now implement market-based mechanisms to provide an advertisement service for small businesses to meet their potential online customers. However, these mechanisms are different from ours in a number of important ways. Specifically, the marketplaces in these systems are established primarily to coordinate currency transactions (between the advertisers and the Web site owners). However, our marketplace is a means for coordinating different recommendation algorithms. Thus, our system is an information filtering system rather than an e-commerce system. Additionally, these mechanisms are weak in personalizing their offerings to the user or in responding to user feedback. Their advertisements are typically selected based on the very general keywords of products, such as "medical books" or "comedy movies." But no attempt is made to identify which medical book or which comedy movie is appropriate for a specific user. Even more importantly, the ranking of recommendations is not oriented to the users. Such systems simply rank recommendations by the price that advertisers are willing to pay. In contrast, our system (through its reward mechanism) incentivizes advertisers to modify their behavior in order to make recommendations that better fit with the user's preferences.

3. DESIGNING THE MARKETPLACE

In this section, we first present an overview of the marketplace architecture (see Section 3.1). Then, we introduce the evaluation metrics that we will use to examine the marketplace's properties in Section 3.2. We present a detailed market mechanism design (including the auction protocol, the reward mechanism, and the bidding strategy) in Section 3.3. Finally, we analyze how the market performs from an economic viewpoint in Section 3.4.

3.1 The Marketplace Architecture

At an abstract level, the problem of populating the limited space of the sidebar from the large number of potential recommendations can be viewed as a scarce resource allocation problem. Moreover, one of the best ways of allocating scarce resources is to sell them using free market techniques [Samuelson and Nordhaus 2001; Varian 2003]. Given this, we decided to view our recommender system as a computational economy [Tesfatsion 2002].

More specifically, auctions are an excellent method of distributing resources to those who value them most highly [Reynolds 1996]. Here an auction is simply a market institution with an explicit set of rules determining resource allocation and prices on the basis of bids from the market participants [McAfee and McMillan 1987]. In a typical auction, there is an auctioneer, a seller, and potential bidders. The auctioneer, acting on behalf of the seller, wants to sell the item and get the highest possible price, while the bidders, employing some bidding strategies to place bids, want to buy the item at the lowest possible price [Vickrey 1961; Milgrom 1989; Klemperer 1999]. However, there is no universal auction design that is applicable to every context [Roth 2002; Jennings et al. 2001]. Auctions vary from one another and these variations make the auctions more or less efficient in particular types of application. In our case, the marketplace operates according to the following metaphor (see Figure 3). A user browses the Web in a particular information domain and requests recommendations from the marketplace. We assume *the user does not change his or her browsing context and his or her interests during the course of this browsing activity*. The auctioneer agent[2] acting on a user's behalf sells

[2] Agents are clearly identifiable problem-solving entities with well-defined boundaries and interfaces. They are situated (embedded) in a particular environment over which they have partial control and observability (receive inputs related to the state of their environment through sensors and act on the environment through effectors) and they are designed to fulfill a specific role [Jennings 2001]. In this article, the term *agent* is used specifically to represent the software agents, not the human agents in the traditional economic sense. Thus, for example, a recommending agent is a software entity that encapsulates a particular type of recommendation algorithm.

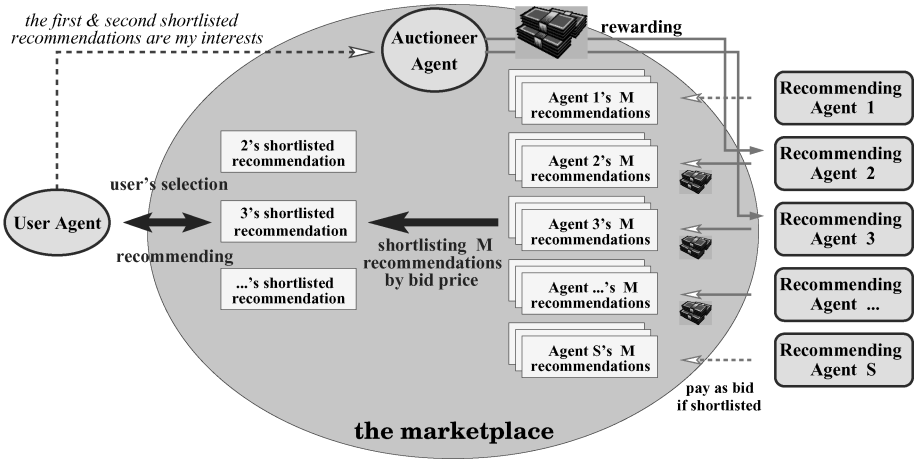

the first & second shortlisted recommendations are my interests
Auctioneer Agent
rewarding
Agent 1's M recommendations
Agent 2's M recommendations
Agent 3's M recommendations
Agent ...'s M recommendations
Agent S's M recommendations
Recommending Agent 1
Recommending Agent 2
Recommending Agent 3
Recommending Agent ...
Recommending Agent S
2's shortlisted recommendation
user's selection
User Agent
3's shortlisted recommendation
recommending
shortlisting M recommendations by bid price
...'s shortlisted recommendation
pay as bid if shortlisted
the marketplace

Fig. 3. The marketplace architecture.

sidebar space where information may be displayed (in our case the sidebar has M slots). Information providers are keen to get their recommendations advertised in the user's browser, and compete in the marketplace, ready to pay for such advertisements. Such information providers act as bidders. Each recommendation with a bidding price acts as one bid. The marketplace acts as the auctioneer, ranking and selecting the most valuable items and recommending them to the user. The user will then choose some of them according to his or her interests as the next documents to be viewed. Those agents who provided such recommendations are the winners in this auction and will receive some reward in return (since such documents are deemed useful). Those documents not chosen by the user are deemed to have no relevance to the current document and will therefore receive no reward.[3] Thus, over the longer term, those agents that make good recommendations become richer and so are able to get their recommendations advertised more frequently than the methods whose recommendations are infrequently chosen by the user.

There are millions of different types of auctions [Wurman et al. 1998]; however, as is often the case in designing computational economies [Dash et al. 2003], none of them are exactly suited to our scenario. Specifically, while standard auctions could probably deal with the shortlisting phase, they do not consider the subsequent reward phase. This means we need to design a bespoke auction (as detailed in Section 3.3).

3.2 The Market Evaluation Metrics

Designing market mechanisms is an *engineering design task*, in which the rules should be developed in order to meet particular objectives, either for certain

[3] The credits paid by recommender agents for advertising their recommendations and the rewards awarded to agents to encourage them to put forward good suggestions are not a real currency. Thus, there is no business model concerned with these credits and rewards; they are used only for the coordination of the recommender agents in our system.

participants or for society as a whole [Roth 2002]. In seeking to design the market mechanism for our recommender system, therefore, our first step is to identify the properties that we would like our auction to exhibit. This then gives us the requirement against which we can evaluate our design. In particular we would like to design a market that has the following standard properties [Sandholm 1999; Varian 2003; Dash et al. 2003]:

(1) *Pareto efficiency.* A solution x is Pareto efficient if there is no other solution x' such that at least one agent is better off in x' than in x and no agent is worse off in x' than in x. Pareto efficiency provides us with a way of comparing alternative mechanisms, and good mechanisms should be designed to maximize allocation efficiency [Roth 2002; Sandholm 1999; Varian 2003]. This is important from the point of view of the individual agents because if a non-Pareto efficient mechanism is chosen then the design could be improved upon (for at least one agent) without making any of the other agents worse off.

(2) *Social welfare maximization.* In our context, social welfare is a numeric measure of the sum of all agents' utilities. In contrast to Pareto efficiency, social welfare provides a way to rank different social preferences over the various solutions and to indicate which is best for the group of agents as a whole [Kagel and Roth 1995]. This is a supplement to the Pareto efficient criterion. From the viewpoint of individual agents, there may exist many Pareto efficient solutions to the given problem that cannot be distinguished between. In such cases, social welfare maximization provides a way of differentiating between them by determining which is the best from the social point of view [Sandholm 1999; Varian 2003].

(3) *Individually rationality.* Participation in an auction is individually rational for an agent if its payoff in the auction is no less than what it would get by not participating. A mechanism is individually rational if participation is individually rational for all agents [Sandholm 1999]. Individually rational protocols are essential in our context because all agents (representing the various recommendation methods) need a clear incentive to participate in the market so that the best possible recommendations can be picked by the market. Indeed, if the protocol is not individually rational for some agents, they would simply not participate in the auction and their recommendations would be lost.

(4) *Convergence.* If the prices for the goods being allocated converge after a number of rounds of auctions, the market is said to be *convergent*. This is important from the viewpoint of the bidding agents since it enables them to learn to bid rationally at a certain level for a given type of good (characterized by UPQ level in this case) in order to maximize their revenue [Roth 2002]. Without convergence, an agent will never know how much to bid with respect to a given recommendation and therefore the marketplace behavior would be chaotic.

(5) *Effective shortlist in decreasing order of UPQ.* This is the common aim of all recommender systems [Herlocker et al. 2004]. The marketplace should

be capable of shortlisting the recommendations in decreasing order of the UPQ after a number of auction iterations. This is important from the point of view of the users since they only want a small number of the best recommendations.

(6) *Clear incentives.* A good mechanism design should give agents incentives to act in particular way, such that the system's global goals is attained despite the individual goals of the self-interested agents [Dash et al. 2003; Sandholm 1999]. In our context, the protocol should be able to incentivize the recommending agents about the user's interests so that they can bid differently for different INQ levels. This is important because a recommender agent needs to relate its bids to the internal quality of the recommendations through the feedback from the marketplace which reflects the user's preferences.

(7) *Stability.* A protocol is stable if it provides all agents with an incentive to behave in a particular way over time. The marketplace should be designed to be stable because, if a self-interested agent is better off behaving in some other manner than desired, it will do so [Sandholm 1999]. Thus, an unstable protocol allows agents to behave with intentions that make the system deviate from the its best potential outcome [Roth 2002]. Therefore, stability is important because without it the system behavior is unpredictable.

(8) *Fairness.* A good market mechanism should be fair to all participants [Roth 2002; Dash et al. 2003]. In our context, a protocol is fair if it gives all recommendations equal opportunity of being shortlisted (irrespective of the agent or method that generates them). This is important because we want the system to shortlist the best recommendations in an unbiased manner.

The first three points of the above criteria are set from a pure economic point of view and, therefore, the marketplace can be evaluated against these metrics at design time. The remaining five items relate to the quality of the system's output and can only be evaluated by experiments. Hence, the evaluation against the former metrics are discussed when analyzing the market equilibrium in Section 3.4, whereas the evaluations against the latter metrics will be discussed in Section 4.

3.3 The Market Mechanism Design

With the evaluation metrics in place, Sections 3.3.1, 3.3.2, and 3.3.3, respectively, detail the auction protocol we designed, the reward mechanism we established, and the bidding strategies of the individual agents. Section 3.4 then analyzes how the market performs with such a market mechanism and the corresponding bidding strategies in place.

3.3.1 *The Auction Protocol.* This section defines the auction protocol for managing the multiple recommending agents (as per Figure 4). To ensure recommendations are provided in a timely and computationally efficient manner,

The Variables:

- S: the number of recommending agents ($S \gg 1$[4]);
- $A_{b1}, A_{b2}, ..., A_{bS}$: S bidding agents;
- A_B: complete set of bidding agents, that is, $A_{b1}, A_{b2}, ..., A_{bS}$;
- A_a: auctioneer agent;
- A_u: user agent;
- T_b: duration of the auction;
- M: number of recommendations that A_u requests from A_a;
- $b_{ij} = \langle A_{bi}, rec_j, price_j \rangle$: bid provided by A_{bi}, containing the jth recommendation with bidding price $price_j$ ($i \in [1 \cdots S], j \in [1 \cdots M]$);
- B^{ALL}: a set of bids which represents all bids submitted to A_a;
- B^M: a set of bids which represents the shortlisted bids that will be recommended to A_u;
- B^R: a set of bids which represents those selected by the user (and will be rewarded by A_a);
- SU: a set of recommendations displayed in the user's sidebar (i.e., B^M ignoring the prices);
- SU^R: a set of recommendations that are selected by the user (i.e., B^R ignoring the prices);
- N: number of user-selected recommendations;
- b_l, b_h: two bids for temporary use ($l, h \in [1 \cdots M]$);
- R_h: reward to hth user-selected recommendation.

The Algorithm:

```
$B^{ALL} = \phi$;  $B^M = \phi$;  $B^R = \phi$;                        // system initialization
$CallForBids(A_B, M, T_b)$;                                           // system calls for bids
repeat during the duration of auction $T_b$
{
    $b_{ij} = GenerateBid(A_{bi}, url_j, price_j)$;
    $B^{ALL} = B^{ALL} \cup \{b_{ij}\}$;
}
for $l = 1$ to $M$ do                                                 // shortlist M highest bids
{
    $b_l = FindBidWithLthTopPrice(B^{ALL}, l)$;
    $B^M = B^M \cup \{b_l\}$;
}
$SU = \{ \langle A_{bi}, url_j \rangle \mid \langle A_{bi}, url_j, price_j \rangle \in B^M \}$;   // the set of shortlisted URLs
$SU^R = UserSelectsURLs(SU)$;                                         // user makes selection ($SU^R \subseteq SU$)
$B^R = \{ \langle A_{bi}, url_j, price_j \rangle \mid \langle A_{bi}, url_j \rangle \in SU^R$ and $\langle A_{bi}, url_j, price_j \rangle \in B^M \}$;
$N = |B^R|$;                                                          // the number of user selected items
for $h = 1$ to $N$ do
{                                                                     // reward the user selected items
    $b_h = FindHthBid(B^R, h)$;
    $R_h = ComputeReward(b_h)$;
}
```

Fig. 4. The auction protocol.

we choose a *generalized first-price sealed-bid auction* in which all agents whose recommendations are shortlisted pay an amount equal to their valuation of the advertisement (meaning we have price differentiation[5]). We choose a sealed-bid

[4] We assume the number of recommenders makes the number of recommendations sufficiently large with respect to the number of sidebar slots such that there is sufficient competition to make the marketplace operate efficiently.

[5] If there is more than one item to be sold, the items can all be sold at the same price (called *price uniformity*) or they may be sold at different prices (called *price differentiation*). In this work, we

auction (in which agents will typically make a single bid) to minimize the time for running the auction and the amount of communication generated. We choose a first-price auction with price differentiation because the relative ordering of the recommendations effects the likelihood of them being selected by the user. In particular, in the market, each information provider agent is keen to get its recommendations advertised to the user. Each agent has a valuation of the recommendation (which will be different for the different agents) and is willing to pay up to this amount to display its recommendations. When an agent gets its recommendations shortlisted, and therefore advertised to the user's browser, it has consumed the advertisement service provided by the recommender system. In return, it needs to pay an amount of credit (at the bidding price) to the system for each of its shortlisted items.

In more detail, the market operates in the following manner. Each time the user browses a new page, the auction is activated. In each such activation, the auctioneer agent calls for a number of bids (M which equals to the number of recommendations being sought). Then each bidding agent submits M bids. After a fixed time, the auctioneer agent ranks all the bids it received by their bidding price, and directs the M bids with the highest prices to the user's browser sidebar (as shortlisted recommendations). Those bidding agents whose recommendations are shortlisted pay the auctioneer agent according to how much they bid. Those bidding agents whose recommendations are not shortlisted do not pay anything. The user may then follow up a number of the shortlisted recommendations, in which case the agent that supplied them is rewarded. In the case where multiple shortlisted recommendations use the same document and only one of them is selected by the user, all of them will be rewarded the same amount.

More formally, the protocol for each auction round is defined in Figure 4. It should be noted that: (i) function $GenerateBid$ $(A_{bi}, rec_j, price_j)$ relates to the bidding strategy and will be discussed in Section 3.3.3; (ii) function $UserSelectsRecs(SU)$ concerns the user's behavior of making choices among the shortlisted recommendations and will be discussed in Section 4.1.3; and (iii) function $ComputeReward(b_h)$ concerns the reward mechanism and will be discussed in Section 3.3.2.

3.3.2 *The Reward Mechanism.* With the auction protocol in place, we now turn to the reward mechanism. According to our protocol, the user may select multiple recommendations from the shortlist. For each such user-selected recommendation, the suggesting agent is given a reward. In defining the $ComputeReward$ function, our aim is to ensure that it is both Pareto efficient and social welfare maximizing (as per Section 3.2).

Since *the global objective is to shortlist the most valuable recommendations in decreasing order of user perceived quality*, we decided to reward the user-selected recommendations purely based on the UPQ (*not* INQ). The UPQs can be defined as Q_h ($h \in [1 \cdots N]$ and Q_h is a positive natural number that represents

exploit price differentiation because it differentiates recommendations so as to display them at different advertisement slots and it allows a seller to obtain the maximum possible profit [Varian 2003, pp.439–441.], a user's ratings or preferences of the interesting recommendations). In practice, however, all user-selected recommendations are ordered in decreasing rank of UPQ such that $Q_1 \geqslant Q_2 \geqslant \cdots \geqslant Q_N$. Thus, Q_h denotes *the* hth *rewarded recommendation* (user-selected recommendation with the hth highest UPQ). To ensure different quality recommendations' bidding prices converge to different levels (so that our marketplace is able to differentiate recommendation qualities), we involved two other variables: P_h ($h \in [1 \cdots N]$) and P^*_m ($m \in [1 \cdots M]$). The former is the bidding price of the hth rewarded recommendation. The latter is the historical average bidding price of the mth shortlisted recommendation during the system's lifetime (note the bidding agents do not actually know this value). By this definition, P^*_m indicates the price for the mth advertisement displayed in the user's browser sidebar which is decided by the "invisible hand" (namely the market). With this information, we can define the reward to the hth rewarded recommendation as

$$R_h = \delta \cdot Q_h \cdot P_{M+1} - \ \alpha \cdot |P^*_h - P_h|, \tag{1}$$

where δ and α are two system coefficients ($\delta > 0$ and $\alpha > 1$) and P_{M+1} is the highest not shortlisted bid price (the detailed justification for this particular choice is given in Wei et al. [2003b]). The values of δ and α will depend upon the specifics of the application (see Section 4.1.1 for details), but they need to be set at suitable values to ensure $R_h > P_h$ so that the rewarded agents can make profits. We based the reward on P_{M+1} (whose value is not known by the bidding agents) so that the market cannot easily be manipulated by the participants [Varian 2003, page 289]. This approach also reduces the possibility of bidding collusions because the reward is based on something that the rewarded agents are unaware of and cannot control.

3.3.3 *Designing the Agents' Bidding Strategies.* In our marketplace, three kinds of information are revealed to a bidder with regards to a specific recommendation: (i) this recommendation's INQ, (ii) this bidder's last bid price (P^{last}), and (iii) the previous rewards to this recommendation (a bidder actually knows the second piece of information). With this information, a rational bidder seeks to maximize its revenue by bidding sensibly for recommendations based on its knowledge of previous outcomes. Such bids can result in one of the following outcomes occurring: the bid is not shortlisted, it is shortlisted but not rewarded, or it is rewarded. With respect to a given INQ level, a bidder's strategy depends on the last outcome in the following way (again see Wei et al. [2003b] for a justification for these choices):

—*Bid not shortlisted.* The only way to increase revenue is to get the recommendation shortlisted. Therefore, the agent will increase its bidding price:

$$P^{next} = Y \cdot P^{last} \qquad (Y > 1).$$

—*Bid shortlisted but not rewarded.* This means the agent overrated its INQ with respect to the UPQ and so the agent should decrease its price in subsequent rounds so as to lose less:

$$P^{next} = Z \cdot P^{last} \qquad (0 < Z < 1).$$

Table I. Price Adjustment and Results

Current price	Adjustment	$\lvert P_h^* - P_h \rvert$	$\Delta\xi$
$P_h < P_h^*$	$+\Delta P$	↘	> 0
	$-\Delta P$	↗	< 0
$P_h > P_h^*$	$+\Delta P$	↗	< 0
	$-\Delta P$	↘	> 0

—*Bid rewarded.* These agents have a good correlation between their INQ for a recommendation and that of the UPQ. Therefore, these agents have a chance of increasing their revenue. The profit made by the hth rewarded recommendation is

$$\xi_h = \delta \cdot Q_h \cdot P_{M+1} - \ \alpha \cdot |P_h^* - P_h| - P_h.$$

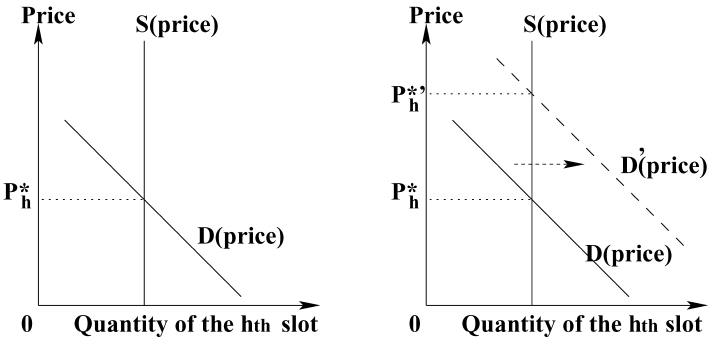
However, the agent is unaware of P_h^* (as per Section 3.3.2), so in practice it does not know whether ξ_h has been maximized. Hence, it must minimize $(\alpha \cdot |P_h^* - P_h| + P_h)$ so as to maximize ξ_h. Furthermore, the agent does not know whether P_h is higher or lower than P_h^*. In either case, however, the agent will definitely make a loss if P_h is not close to P_h^*. Therefore, we find that the hth rewarded agent can always be aware of whether its price is closer to or farther from the hth historical average market price, P_h^*, by adjusting its bidding prices (see Wei et al. [2003b] for the formal proof).

We have previously proved that a rational rewarded bidder will adjust its price in order to the corresponding average market price to maximize its profit [Wei et al. 2003b]. Therefore, a rewarded agent's practical strategy with respect to certain rewarded recommendations is to bid in the following manner: whatever its current price is with respect to the historical average, when adjusting the bid price, if the adjustment results in making less profit, it indicates the action is wrong and $(P_h \pm \Delta P)$ is farther from P_h^*; if it results in making more profit, it indicates the action is right and $(P_h \pm \Delta P)$ is closer to P_h^* (see Wei et al. [2003b] for more details). This phenomenon is listed in Table I ($\Delta\xi$ represents the possible profit of the next bid compared to that of the current bid). In fact, Table I specifies the strategy for the rewarded agents: chasing the corresponding historical average market price. The actual value of ΔP will be defined in an application-specific manner.

3.4 The Market Equilibrium and Economic Justifications

According to the strategy for rewarded bidders (Section 3.3.3), such bidders must bid in a manner that aligns their internal view of quality with that of the user. Thus, over time, each individual recommending agent improves its correspondence between its bid price and the user's preferences for recommendations. Only by achieving this can an agent maximize its profit. However, how quickly this convergence occurs depends on the adjustment of price ΔP.

In the short term, assuming that the set of recommending agents remains unchanged between successive auctions and they produce recommendations of the same quality level, we can show that the market reaches an equilibrium. The hth historical average market price reflects the market equilibrium price:



(a) Market equilibrium

(b) Equilibrium changed

Fig. 5. Market Equilibrium and Its Change. (**(a)** The supply curve **S** is vertical, indicating that whatever the deal is, the supply of the *h*th advertisement slot is constant. The demand curve **D** has a slope indicating that more agents are willing to pay a low price and few agents are willing to pay a high price for the same slot. The cross indicates that at a certain price level the quantity of demand equals that of supply. This cross point represents the market equilibrium. **(b)** At each price level, more recommendations become available and the demand curve shifts to the right.)

thus, at a certain price, the quantity of demand of the *h*th advertisement slot equals the quantity of the supply (see Figure 5(a)[6]).

In the long run, however, these assumptions will not hold and the equilibrium will tend to be broken. However, this new market situation will gradually tend toward another equilibrium and will reach it as long as the changes in the recommending agents are not too frequent with respect to convergence times (see Figure 5(b)). If, for example, there is more demand in the system, the demand curve will shift right compared to Figure 5(a). This means that, at each price level, there are more bidders willing to pay for the same advertisement slot (because, for example, more better recommendations are being produced).

At equilibrium, since the bidding prices are aligned with the UPQs, the system will produce a shortlist of recommendations in decreasing order of UPQ (see Wei et al. [2003b] for the proof), which is precisely the objective of the recommender system. Moreover at this point, our reward mechanism (see Formula (1)) exhibits Pareto efficiency and social welfare maximization for the recommending agents. Here, we briefly sketch how the mechanism achieves these two properties (see Wei et al. [2003b] for full details and proofs). With our reward mechanism, the historical average market price, P_h^*, reflects how the majority of bidders value a given advertisement slot and this price becomes the expected equilibrium price. This incentivizes each bidder to iterate itself to the corresponding expected equilibrium price. Therefore, the market has a tendency to converge to equilibrium. With the market tending to equilibrium, the second term in Formula (1) tends to zero. Therefore, this mechanism is equivalent to rewarding agents proportionally to their user-selected recommendations' UPQs, which is precisely what we want to achieve.

[6] Strictly speaking, the demand curve should be discrete in this case. And the quantity of supply is 1 since we differentiate between each of the M slots and there is only one hth slot. To simplify the discussion, however, we use a continuous demand curve in this context.

Additionally, with the market tending to equilibrium, the bidding strategies outlined in Section 3.3.3 are dominant (the best thing to do, irrespective of the actions of any other agents) and so will be adopted by the designers of all rational recommending agents.

4. EVALUATING THE MARKETPLACE

This section reports on the simulation experiments to evaluate the market mechanisms designed for our recommender system in Section 3 with respect to the last five criteria described in Section 3.2. The experimental settings are discussed in Section 4.1. The evaluations of the marketplace are then presented in Section 4.2. Section 4.3 evaluates the market properties and the correlation between the UPQ and the INQ in more general cases when multiple features of recommendations are considered. Section 4.4 evaluates the system's ability to seek out the recommendation with the highest UPQ value from all bids and recommend it to the user.

4.1 Experimental Settings

Our system is composed of three kinds of agents: the auctioneer agent, the recommending agents, and the user agent (as per Figure 3) (discussed, respectively, in Sections 4.1.1, 4.1.2, and 4.1.3). Before we discuss these agents, however, an important system variable, the number of bids called for, M (defined in Section 3.3.1), needs to be decided. Here we use the value of 10 (because our previous study showed this is the number of items that can be managed efficiently in the browser's sidebar [Moreau et al. 2002]).

4.1.1 *Configuring the Auctioneer Agent.* The auctioneer agent determines the reward paid to the agents who make recommendations selected by the user. Given that the rewarded mechanism is defined in Formula (1), two system variables control the auctioneer agent: δ and α (defined in Section 3.3.2). From the reward mechanism, we can see that δ affects the volume of the credit paid to a particular user-selected recommendation. The bigger δ is, the more the recommendation is paid. We can also see that α affects the sensitivity of the incentives the marketplace delivers to the recommending agents to make them aware of the equilibrium (because the recommending agents need large alterations to chase the equilibrium price if α is big). In our experiment, we set $\delta = 1.5$ and $\alpha = 1.5$ (based on our experience that these values enable the recommending agents to both increase their revenue by making good recommendations over the long term and chase the equilibrium quickly [Wei et al. 2003a]).

4.1.2 *Configuring the Recommending Agents.* In this subsection, we discuss how a recommending agent generates a bid and how it relates the bidding price to its INQ for a recommendation. Before delving into this discussion, however, the number of recommending agents contained in our system needs to be defined. We assign this system variable (see S defined in Figure 4) a value of 9 (to ensure there is a sufficient number of input recommendations and sufficient competition in the marketplace). This value is not chosen for experimental expediency and, in practice, it would depend on how many actual bidding agents participate in the marketplace.

Each agent has a set of recommendations available to suggest (typically ordered according to their INQs). Each such agent needs to compute the relation between its local perception of relevance and the user's preference. Having done this, it can then bid an appropriate price to maximize its revenue. Thus, the agent will relate its bidding price to its knowledge about the UPQ (reflected by the rewards it has received) with respect to different INQ levels. We term this relationship between the bidding price and the INQ an agent's *strategy profile*. This profile is on a per agent basis. It records an agent's bidding price for different INQ levels and indicates how an agent should relate its bid to its INQ.

4.1.2.1 *Simulating Recommendation Methods.* To assess the broad feasibility of our market-based approach, we want our representation of the INQs to be capable of corresponding to as many recommendation techniques as possible. Moreover, we do not want our results to be skewed by any innate bias in the recommendation methods themselves. Therefore, we take an abstract view on the recommender methods and view them simply as being able to learn a user's interests based on their internal belief about certain recommendation properties (features or attributes) that the user's context focuses on. We believe this is a reasonable abstraction because a recommendation method's ability to adaptively match certain recommendation properties to the user's actual preferences has been shown to be crucial to making high-quality recommendations [Claypool et al. 1999]. Given these observations, we define the INQ of a specific recommendation method to be the sum of the weighted evaluation scores made of different techniques on different properties of a recommendation (see Equation (2)). This is consistent with the observation that effective recommendation methods need to combine filtering techniques based on different recommendation properties to achieve peak performance [Burke 2002]. To this end, we simulate the recommendation methods' INQs on a linear basis.[7] More formally,

$$q(Rec) = k_1 \cdot \Phi_1(Rec) + k_2 \cdot \Phi_2(Rec) + \cdots + k_I \cdot \Phi_I(Rec) \qquad (I > 0), \tag{2}$$

where $q(Rec)$ represents the INQ of item Rec based on a specific method. This method evaluates an item from I perspectives (i.e., properties, features, or attributes). The value of I is on a per method basis because different methods evaluate different numbers of properties of an item. Here, each $\Phi_i(Rec)$ $(i \in [1 \cdots I])$ represents the evaluation score based on a specific property of

[7] This linear combination is used by several hybrid recommender systems in combining results from different recommendation methods [Claypool et al. 1999; Pazzani 1999; Littlestone and Warmuth 1994]. Through combining different weighted properties or features, it is believed that a recommendation method can improve its precision in predicting the user's preference and improve its quality of recommendations [Pazzani 1999; Yu et al. 2003].

Rec ($\Phi_i(Rec) \in [0, 1.0]$).[8] Such properties can be either objective (such as the TFIDF [Salton 1989] of a document), subjective (such as customers' opinions of the tastes of the foods in a restaurant) or a mixture of the two (such as users' opinions of the textual and graphical descriptions of the products of a store). Variable k_i ($k_i > 0$) specifies the weight of $\Phi_i(Rec)$ and $k_1 + k_2 + \cdots + k_I = 1.0$ in order to ensure $0 \leqslant q \leqslant 1.0$.

For example, consider the case where the user's browsing context is local restaurants. In this situation, an individual recommendation method might base its INQ on the TFIDF of an online restaurant menu with a value between 0 to 1.0, other people's opinions of the food on the restaurant's Web Site with an integer voting value of $1 \cdots 5$ (normalization will be used), whether the user has ever consumed the service of the current restaurant with a binary value of 0 or 1, or any other possible properties of the item. In our case, each of these corresponds to a specific $\Phi_i(Rec)$ and if a particular method uses a combination of these base terms then appropriate values for the respective k_i's would be set.

The next step is to determine how to simulate Φ. Based on our previous studies in this area, by randomly collecting 400 different Web pages on the subject of "world news," we found that the keyword similarity [Moreau et al. 2002] of the 400 documents compared to CNN's frontpage (`www.cnn.com`) follows a Gaussian normal distribution (see the contour of the distribution in Figure 6(a)). Hence, we decided to use some Gaussian normal distributions to model the properties (Φ) of recommendations in predicting user's preferences on a probabilistic basis [Popescul et al. 2001; Sharma and Poole 2001]. Specifically, in our experiments, we simulated different document properties of one method by different random variables that follow different normal distributions. The probability density function of the normal distribution is defined as[9]:

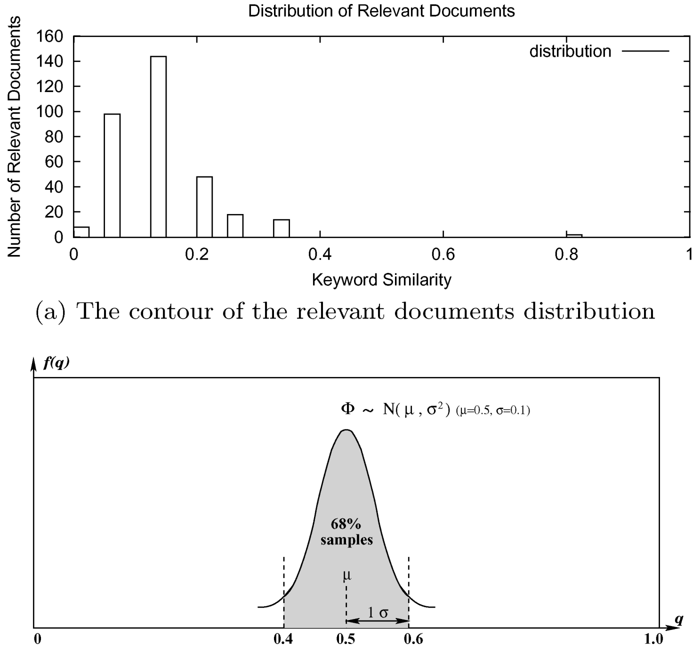
$$N(\mu, \sigma^2): \quad f(q) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(q-\mu)^2}{2\sigma^2}}, \qquad q \in [0, 1.0], \tag{3}$$

where μ and σ are the mean value and the standard deviation of the random samples (see Figure 6(b)). The mean of the distribution represents the average value of the INQs of all samples generated by the corresponding method. The middle range (between one unit of standard deviation on both sides of the mean) of the distribution contains the majority of the samples (about 68% of its total).

One of the key objectives of the recommending agents is to build up their strategy profiles so that they can relate their bidding price to their INQs based on their knowledge about the reward (which, in turn, reflects the UPQ of the recommendations). In order to learn such characteristics for all INQ levels, each agent divides its strategy profile into 20 continuous segments. In each auction,

[8] When evaluating different recommendation methods, we perform a normalization on the results to fix them into a range of [0, 1.0]. This is because different recommendation methods have different quality (or rating) ranges [Pennock et al. 2000]. This can be achieved in practice by adaptively matching a method's min and max INQ value onto 0 and 1.0, respectively. This makes the values from different methods meaningful in our market based recommender system in terms of INQ and UPQ.

[9] We fix the sample into the range [0, 1.0] (rather than $(-\infty, +\infty)$) since we have manipulated the INQ into this range (see Equation (2)).



(a) The contour of the relevant documents distribution

(b) Distribution of a hypothetical property of a set of recommendations

Fig. 6. Simulating evaluation technique.

a recommending agent needs to compute the INQs of ten recommendations and make 10 corresponding bids. In the early auction rounds, all the agents' strategy profiles are empty. With an empty strategy profile, an agent will bid proportionally (because it can only expect a high INQ recommendation to receive a high UPQ and, consequently, more reward than a low INQ recommendation) to the INQ of 10 (value of M defined in Section 3.3.1) recommendations based on an initial seeding price. We set different initial seeding price values (randomly generated from the range [128, 256][10]) for different recommending agents (because different agents value their recommendations differently with their empty strategy profiles). After each auction, all strategy profile segments record and update information about the last bid status (not shortlisted, shortlisted but not rewarded, or rewarded), the last bid price, the last rewarded price, and the last rewarded profit. Based on such information about each segment, and using the appropriate bidding strategy, an agent can compute its bids in subsequent auctions if there are recommendations that belong to this segment. After a number of iterations, those segments that cover the majority of samples will have sufficient information to reach the equilibrium price and form a stable strategy profile.

4.1.3 *Configuring the User Agent.* Again in seeking to evaluate the principle of a market-based approach to recommendation, we want to work in a

[10] The exact values of the boundary of the range are not important. What matters is whether such a randomly given range can make the market converge and exhibit the other properties specified in Section 3.2.

Table II. User's Decision of Different Models[a]

Shortlisted recommendations	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	Q_8	Q_9	Q_{10}
User perceived quality	70	50	75	30	60	82	90	85	65	55
Decision of independent selection	1	0	1	0	1	1	1	1	1	0
Decision of search-till-satisfied	1	0	1	0	1	1	0	0	0	0

[a] Both models have the same *AT* of 60. Search-till-satisfied model has a ST of 80. "1" means the recommendation is selected to be rewarded, while "0" means not selected.

well-controlled environment. Thus we simulate the users of our recommender system (as others have done when seeking to validate the principle of a new method [Billsus and Pazzani 2000; Bohte et al. 2004; Gonzalez et al. 2004]). Specifically, when a user is faced with a set of shortlisted recommendations, she or he will visit some of the recommendations and will then have a valuation of each visited item. Thus, a user assigns a number, Q_i ($Q_i \in [0 \cdots 100]$, $i \in [1 \cdots M]$), to each visited item according to her or his valuation of the recommendation. This number Q_i is the UPQ value. To simulate the choices of a user in selecting recommendations, we deployed a user model inside the user agent. Building on the user simulation of [Bohte et al. 2001], we adopted the following models:

—*Independent selection.* The selection of one recommendation is independent of the others. Once the UPQ of a recommendation is greater than or equal to a particular *acceptance threshold* (*AT*), the recommendation is accepted and rewarded. Those recommendations with UPQ less than *AT* will not be selected and therefore receive no reward.

—*Search-till-satisfied Behavior.* The selection of one recommendation is dependent on other recommendations that are ranked above it in the list. In this case, the user stops searching once he or she discovers a recommendation that has a UPQ greater than or equal to a particular *satisfaction threshold* (ST).

By means of an illustration, Table II is an example of a user's decision under the two different models. All recommendations with UPQ above the *AT* (60) are selected to be rewarded in the case of independent selection. However, Q_7, Q_8, and Q_9 are not selected to be rewarded by the search-till-satisfied behavior though their UPQs are above the *AT*. Indeed, the user stops searching since a document with a quality of 82 (Q_6) has been found above the ST (80).

We simulate the user by a user agent which knows its valuation for each recommendation and assigns the UPQ based on this valuation correspondingly. Thus, when a real user considers I' ($I' > 0$) properties of a recommendation (*Rec*), the UPQ of *Rec* is defined as

$$Q(Rec) = k'_1 \cdot \Phi'_1(Rec) + k'_2 \cdot \Phi'_2(Rec) + \cdots + k'_{I'} \cdot \Phi'_{I'}(Rec), \tag{4}$$

where $\Phi'_i(Rec)$ (whose definition is equivalent to that of $\Phi_i(Rec)$ in Equation (2), $\Phi'_i(Rec) \in [0, 1.0]$, $i \in [1 \cdots I']$) is the valuation of one of the properties of *Rec*. k'_i ($k'_i > 0$, $i \in [1 \cdots I']$) is the weight of $\Phi'_i(Rec)$ contributing to $Q(Rec)$. We set $k'_1 + k'_2 + \cdots + k'_{I'} = 100$ to ensure $0 \leqslant Q \leqslant 100$.

4.1.4 *Correlating the UPQ to the INQ.* From the formal specifications of the UPQ and the INQ of a given item, as given in Equations (2) and (4), it can be seen whether the properties of the document that the user considers overlap with those that a recommendation method considers. Here, we define the set of properties $\{\Phi'_1, \Phi'_2, \ldots, \Phi'_{I'}\}$ that the user evaluates as φ_Q. Likewise, we define the set of properties $\{\Phi_1, \Phi_2, \ldots, \Phi_I\}$ that a recommendation method evaluates as φ_q. We define $\varphi = \varphi_Q \cap \varphi_q$ as the recommendation method's *effective factors* in terms of the UPQ. We define $\overline{\varphi} = \varphi_Q - \varphi_q$ as the recommendation method's *ineffective factors*. The variable φ is important, because if $\varphi \neq \emptyset$ ($\emptyset$ stands for "empty set") the method will have some correlation with the UPQ since their evaluations of the recommendation items share some of the same properties. Otherwise, if $\varphi = \emptyset$, a recommendation method cannot correlate its INQ to the UPQ since they evaluate the items from totally different perspectives.[11] These issues will be discussed in detail in Sections 4.2 and 4.3.

By abstracting all recommendation methods as independent learners that predict user's preferences, all predictions can be seen as composed of effective data and noisy data on a probabilistic basis [Popescul et al. 2001; Sharma and Poole 2001]. This, in turn, simplifies modeling the market-based component recommenders on a high abstraction level. Specifically, by defining a recommendation method's effective and ineffective factors, given a recommendation item *Rec*, its UPQ can be represented in terms of a method's INQ as follows:

$$Q(Rec) = \Gamma(\varphi(Rec)) + \overline{\Gamma}(\overline{\varphi}(Rec)), \tag{5}$$

where Γ and $\overline{\Gamma}$ are two mapping functions that align the coefficients of the elements of φ and $\overline{\varphi}$ with the weightings (k'_i) of the properties (Φ'_i) of Q (see Equation (4)). For example, assume a user evaluates an item *Rec* from perspectives of Φ_a, Φ_b, and Φ_c and the importance of these properties are k'_a, k'_b, and k'_c, respectively ($k'_a + k'_b + k'_c = 100$ and $k'_a, k'_b, k'_c > 0$), the UPQ will be $Q = k'_a\Phi_a + k'_b\Phi_b + k'_c\Phi_c$. Assuming a recommendation method evaluates the item from perspectives of Φ_a, Φ_b, and Φ_d and their relative importance is k_a, k_b, and k_d, respectively ($k_a, k_b, k_d > 0$ and $k_a + k_b + k_d = 1.0$), its INQ is $q = k_a\Phi_a + k_b\Phi_b + k_d\Phi_d$. Thus, the INQ's effective factors are $\varphi = \{\Phi_a, \Phi_b\}$ and its ineffective factor is $\overline{\varphi} = \{\Phi_c\}$. Therefore, $\Gamma(\Phi_a, \Phi_b) = (\frac{k'_a}{k_a}\ \frac{k'_b}{k_b}) \times (k\ )_a\ \Phi_a k_b \Phi_b$ and $\overline{\Gamma}(\Phi_c) = (k'_c) \times (\Phi_c)$. We find that when a recommendation method's effective factors form a major weighting of both its INQ and the UPQ (e.g., in the above example, Φ_a and Φ_b contribute $\frac{k_a+k_b}{k_a+k_b+k_d}$ of the weighting of the INQ and $\frac{k'_a+k'_b}{k'_a+k'_b+k'_c}$ of the weighting of the UPQ), this method can easily correlate its INQ to the UPQ (see Section 4.3 for more details), and can continuously produce good recommendations and make profits. Otherwise, if a method has only ineffective

[11] We assume that one property (Φ_i) is totally independent of another (Φ_j) if $i \neq j$. This means any two different properties of a recommendation do not have a relationship to one another. However, this is not limiting because we have defined the UPQ and the INQ as a linear combination of some property values. Thus, in cases where the two properties do depend on each other, one of them can be decomposed into two subproperties, with one subproperty the same as the other property and the other subproperty independent of the former. However, we will not discuss this case here since it is not our main concern.

factors, the method cannot correlate its INQ to the UPQ and therefore makes poor recommendations most of the time and will go bankrupt. These properties will be discussed in more detail in Section 4.3.

4.2 Evaluation of the Marketplace

Having outlined the setup of the three kinds of agents specified in Section 4.1, this section will focus on evaluating the system properties. In our case, the market is the key to coordinating the various recommendation methods. If it does not work effectively, the system will not be able to make good recommendations. Among the five properties we want our market to exhibit, convergence is the most important because it forms the basis of the other four. Therefore, we will start with experiments on market convergence.

4.2.1 *Market Convergence.*[12] We endow our system with 100 documents ready to be recommended to the user. Every time the user visits a specific recommendation, the UPQ of this recommendation is assigned by the user and this value is independent of the various methods' INQs. To simplify the experiments on evaluating the properties (Φ_i) of a recommendation item, we assume each recommendation method evaluates items from only one property (but two different methods may use different properties). The more general case with more than one Φ_i involved in each method is dealt with in Section 4.3. We further assume the user considers two different properties of the recommendations (Φ_0 and Φ_1). Thus, the effective and ineffective factors of the recommendation methods can be easily controlled.[13] Assuming the weighting of the two properties are k_0 and k_1, respectively, the UPQ can be represented formally as

$$Q(Rec) = k_0 \cdot \Phi_0(Rec) + k_1 \cdot \Phi_1(Rec). \tag{6}$$

To generalize the experiments, nine component recommender agents are placed in our marketplace and each of them is based on one of three different properties (Φ_1, Φ_2, Φ_3) of recommendations (note here Φ_1 is the same as in Equation (6)), meaning that some of the recommendation methods contain the effective factors in terms of the UPQ and some of them do not. We will use three Gaussian normal distribution functions (see Equation (3)) to simulate the valuations of the three properties. Each property relates to one of the three distributions: $N(0.35, 0.1^2)$, $N(0.5, 0.1^2)$, and $N(0.65, 0.1^2)$ (see Figure 7). We set the standard deviation to a value of 0.1, meaning the three different properties share only a small intersection (so as to easily differentiate the different methods). Thus, those

[12] The working scenario and the configurations of the UPQ and the INQ in this section will be used for all experiments in Section 4.2

[13] We can exemplify this case in a scenario where the user is browsing the local restaurants on the Web. We assume the user evaluates the recommended restaurant Web sites from two perspectives: whether the restaurant sells some specific foods (Φ_0) and other customers' opinions of the foods in the restaurant (Φ_1). If a recommendation method also computes Φ_1, then Φ_1 is its effective factor and Φ_0 is its ineffective factor in terms of the UPQ (mutatis mutandis if the method computes Φ_0). If a recommendation method evaluates the recommendations by Φ_x (which is different from Φ_1 and Φ_0), then it has no effective factors.

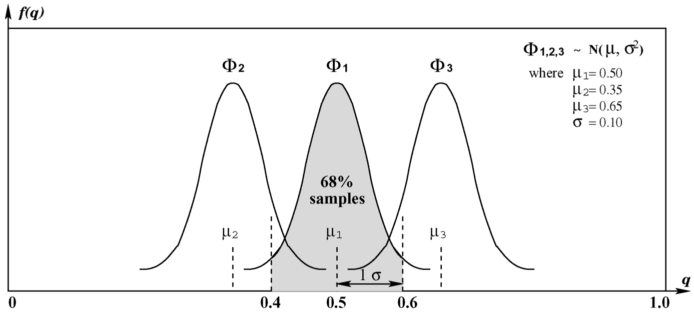


Fig. 7. Distributions of three properties of a set of recommendations.

Table III. Configurations of the Three Groups of Experiments

Experiments	Configurations
Experiment 1	$q_i(Rec) = \Phi_1(Rec)$ $(i \in [1\cdots3])$ and $q_j(Rec) = \Phi_2(Rec)$ $(j \in [4..6])$ and $q_k(Rec) = \Phi_3(Rec)$ $(k \in [7\cdots9])$
Experiment 2	$q_i(Rec) = \Phi_1(Rec)$ $(i \in [1\cdots9])$
Experiment 3	$q_1(Rec) = \Phi_1(Rec)$ and and $q_j(Rec) = \Phi_2(Rec)$ $(j \in [2..5])$ and $q_k(Rec) = \Phi_3(Rec)$ $(k \in [6..9])$

methods' INQs based on Φ_1 can be presented formally as

$$q_i(Rec) = \Phi_1(Rec) \qquad (i \in [1..3]). \tag{7}$$

In this case, the UPQ (Equation (6)) can be represented in terms of the INQ which contains the effective factors

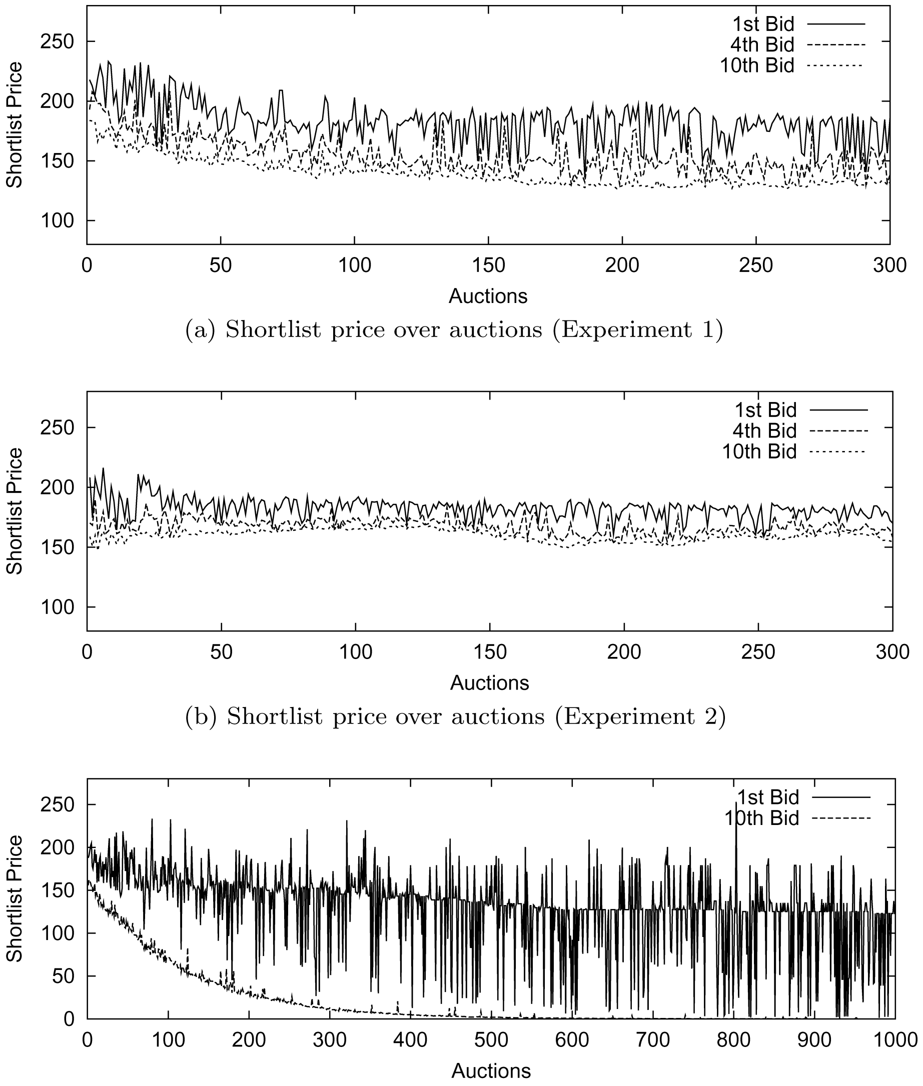
$$Q(Rec) = k_1 \cdot q_i(Rec) + k_0 \cdot \Phi_0(Rec) \qquad (i \in [1\cdots3]). \tag{8}$$

Having further configured the experimental settings, we are going to examine the system property from the perspective of market convergence. In Section 3.4, we showed that the marketplace can reach an equilibrium such that the shortlist prices converge at different levels with respect to different UPQ levels. To evaluate this, we arranged 300 auctions with 10 shortlisted recommendations using the independent selection user model ($AT = 66$) and ($k_1 = 75$, $k_0 = 25$)[14] for Equations (6) and (8) to see if the marketplace does indeed have such a convergence property. We organized three groups of experiments, each of which contains a different number of agents having the effective factors, to see whether the market converges in various cases. The configurations are shown in Table III.

In the first experiment, each of the three properties is shared by three agents; thus only the first three agents contain the effective factor, whereas the other six do not.[15] From Figure 8(a), we can see that the shortlisted prices converge

[14] k_1 and k_0 can be set to any other combinations in these experiments. 75 and 25 are chosen to exhibit the higher importance of Φ_1 over Φ_0.

[15] In this case, the first three agents can relate their bidding price to their INQs, since their INQs have a relationship with the UPQ (contributing 75% of its total weighting; See Equation (8)). Also the rewards they received reflect the UPQ with respect to a specific recommendation. The remaining six agents cannot relate their bids to their INQs because their INQs have no relationship with the UPQ and their rewards. This subject will be discussed further in Section 4.2.3.



(a) Shortlist price over auctions (Experiment 1)

(b) Shortlist price over auctions (Experiment 2)

(c) Shortlist price over auctions (Experiment 3)

Fig. 8. Convergence of shortlist prices.

(for example, the 4th and 10th bids oscillate around 150 and 130, respectively, which indicate P_4^* and P_{10}^*, respectively) after about 100 auctions. We find that, with the search-till-satisfied user model (with $ST = 60$ and $AT = 45$), the market also converges (for which we do not provide a figure), but only after a longer time (more auction rounds) compared to the independent selection. This takes longer because fewer agents are rewarded in this case and they need more bids to chase the equilibrium price.

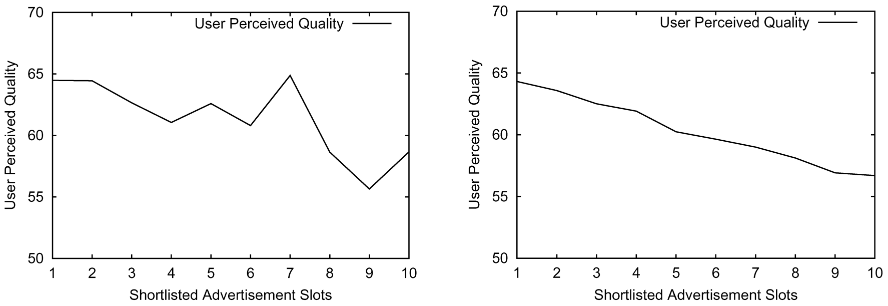
In the second experiment, all nine agents evaluate recommendations by property Φ_1. In this case, the market converges very quickly (after about 30 auctions;

see Figure 8(b)), because all agents' INQs are actually the effective factors in terms of the UPQ. Thus they have a good correlation with the user's valuation of the recommendations. Therefore, more recommendations at each quality level can be related to the UPQ and the agents receive more signals of the user's interests. This, in turn, means agents get sufficient chances to alter their price effectively to chase the equilibrium price with respect to each UPQ level. This results in a market that converges quickly.

In the third experiment, only the first agent evaluates Φ_1 and the other eight agents evaluate Φ_2 or Φ_3. The market still converges but very slowly (after about 600 auctions; see Figure 8(c)), with the first bid price oscillating around 125. This slow speed can be accounted for by the fact that only one agent can relate its good recommendations' bidding price to its INQ with respect to each UPQ level and there are insufficient good recommendations. Therefore, the agent needs a longer time to get a sufficient number of high-quality recommendations to be rewarded and to chase the equilibrium price. In this experiment with very few agents taking the effective factor in terms of UPQ, it is interesting to see that the 10th bid price decreases till it reaches zero (see Figure 8(c)).[16] The explanation is that most of the recommendations, from the eight agents with only ineffective factors as their INQs, cannot relate their bidding prices to their INQs. Thus, these agents cannot reason about the relationship between the rewards and the INQs of the rewarded recommendations (since the rewards are based on the UPQ, *not* on the INQ). Therefore, the equilibrium price for such bids (if it exists) has no relationship with the INQ. Such a recommending agent cannot chase the equilibrium price based on the INQ. Such shortlisted (both rewarded and not rewarded) recommendations will make negative profit most of the time. Hence, most of the recommendations will bid as low as possible to reduce their loss (this phenomenon continues till the bid prices reach zero, meaning paying nothing). The exception to this is the small number of bids from the only agent with effective factors. Overall, this experiment demonstrates that the marketplace deters bad recommendations and only good recommendations can pass through.

When all the experiments are taken together, we find that the shortlisted prices always converge after a number of iterations as long as there is at least one agent that has effective factors. The speed of the convergence depends on the setting of the parameters α, AT, ST, Y, and Z. Since these variables are not our main concern here, we only supply an overview of their effects. Broadly speaking, AT and ST affect the number of recommendations being rewarded (because more agents are rewarded if their values are small). By being rewarded more times, an agent receives more information and therefore can chase the equilibrium faster. The variables Y and Z also affect the speed with which the agent can chase the equilibrium. Specifically, with high values of these variables, an agent alters its price quickly to reach the equilibrium price.

[16] Actually only the first and second bid prices converge in this experiment. The second bid is not plotted in Figure 8(c) because it is close to the first bid and we want to clearly display the convergence. The other eight bids, the 3rd $\sim$ 10th, do not converge and decrease continuously till reaching zero (for the same reason only the 10th bid is plotted).



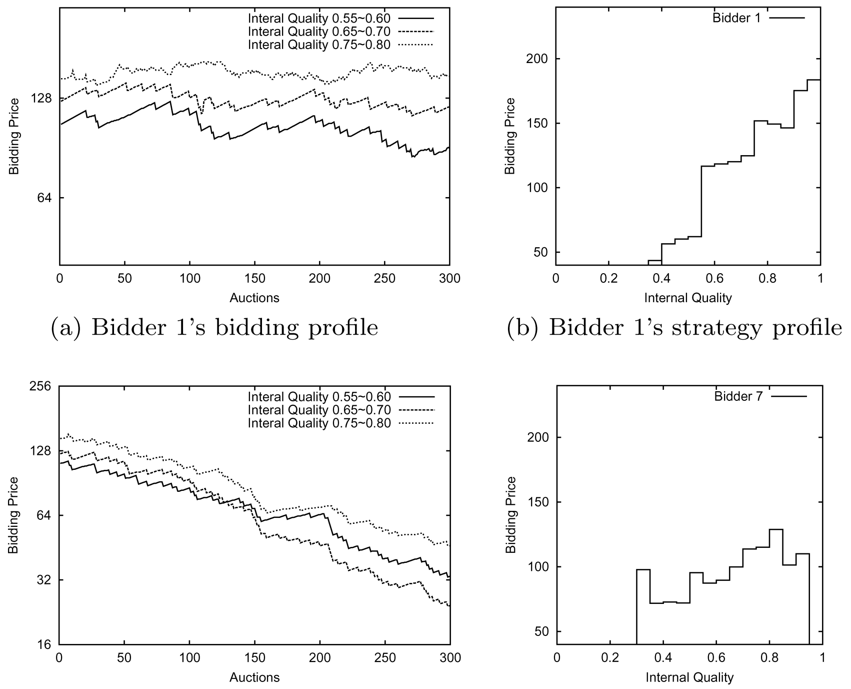
(a) User perceived quality in one auction

(b) Average user perceived quality of 15 continuous auctions

Fig. 9. The UPQ of shortlisted recommendations (Experiment 1).

4.2.2 *Efficient Shortlists.* The most important feature of our system is its capability of shortlisting the best recommendations in decreasing order of UPQ when the market converges. To this end, Figure 9(a) shows the UPQ of the shortlisted recommendations during the 100th auction (which is after convergence) in the first experiment introduced in Section 4.2.1. Here, we can see that the quality of the 10 shortlisted recommendations had an overall tendency to decrease in most cases (although there were some exceptions). Figure 9(b) shows the average UPQ of 15 continuous auctions after convergence (from the 101st to the 115th auction). By averaging over these auctions, we can see that the UPQ decreases monotonically. Thus, Figure 9 tells us that our market mechanism is indeed capable of shortlisting the best recommendations in decreasing order of UPQ. Through various experiments stated in Section 4.2.1, we find that our market can always do so and our results hold more broadly than just for this specific experiment.

4.2.3 *Clear Incentives.* The next step is to see if the recommending agents can relate their bids to the INQs of their recommendations (meaning an agent can generate a steady strategy profile). In this case, each recommending agent builds up its strategy profile from its knowledge about the bids with respect to its 20 INQ segments. Specifically, Figure 10(a) shows the bidding prices for different segments of the first recommending agent with effective factors Φ_1 as its INQ. From Figure 10(a), we can see that this agent's bidding prices for different INQ segments oscillate around certain levels after the market reaches equilibrium (after about 100 auctions). Figure 10(b) shows the agent's strategy profile (equilibrium bidding price versus the INQ segments) and that higher INQ does indeed relate to higher bidding price. Indeed, this agent evaluates its INQ on the effective factors, in particular, on those that have a high weighting in the UPQ (see Equations (6) and (8)). Thus, the agent can relate its bidding price to its INQ in such a way that the higher the INQ, the higher the corresponding UPQ, and the higher the bidding price. In this way, the agent maximizes its revenue. Figure 10(c) shows the bidding prices for different segments of the seventh agent with the ineffective factor Φ_3 as its INQ, and Figure 10(d) depicts this agent's strategy profile (which shows there is no relationship between the



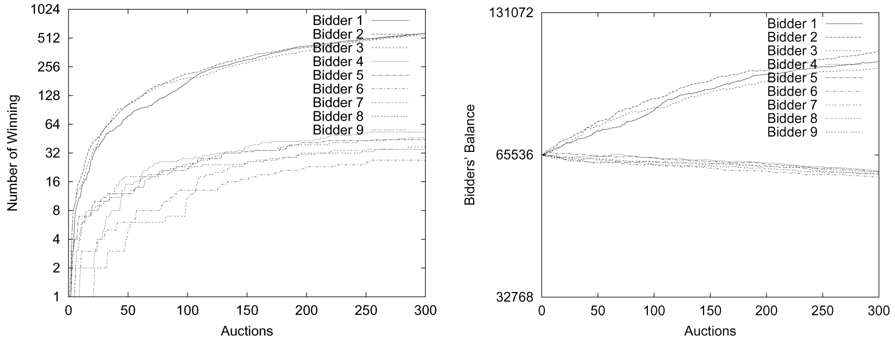
(a) Bidder 1's bidding profile (b) Bidder 1's strategy profile

(c) Bidder 7's bidding profile (d) Bidder 7's strategy profile

Fig. 10. Bidding profile and strategy profile of bidders with effective and ineffective factors (Experiment 1).

bidding price and the INQ). From Figures 10(c) and 10(d), we can see that this agent's bidding prices do not reach equilibrium (because the agent has only ineffective factors as its INQ). Therefore, it cannot relate its bids to its INQ, because it cannot reason about the relationship between the occasional rewards and the INQs of the rewarded recommendations. Since high INQ does not indicate high UPQ in this case, the UPQ with respect to a specific INQ segment can vary dramatically. Therefore, based on the UPQ, the rewards with respect to a specific INQ level do not converge (meaning that the agent can learn nothing from the marketplace). Hence, based on the rewards (see the relationship between the reward and the bidding price in Equation (1)), the bidding prices with respect to this INQ level do not converge. Thus, the agent cannot build up a practical strategy profile after the market converges. Agents with ineffective factor Φ_2 exhibit the same properties as those agents with Φ_3 and we do not comment further on them.

In addition to the bidding strategy profile, we also examined the revenue and the number of times these agents won in the auctions. From Figure 11(a), we can see that the first three agents, with the effective factors, win more times than the remaining six agents (which have ineffective factors). Figure 11(b) shows that the first three agents can make profits whereas the other six make losses over time. Indeed, the agents with ineffective factors always bid high enough to be shortlisted (see Section 4.2.5 for more information about equal



(a) Number of winning over auctions (b) Bidders' balance over auctions

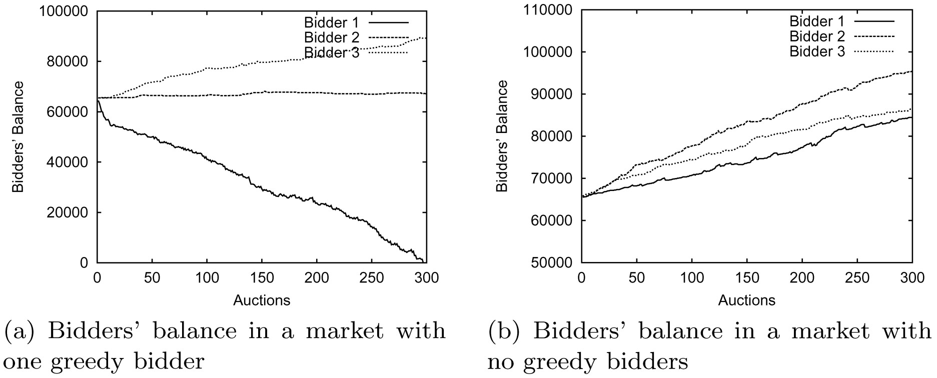
Fig. 11. Number of winning and balance of bidders with effective and ineffective factors (Experiment 1).

opportunities of being shortlisted), but they are not able to learn anything from the few occasional rewards that they receive. Thus, these agents pay more when shortlisted than they earn when rewarded and will eventually go bankrupt.[17]

When taken together, Figures 10 and 11 indicate that the agents with effective factors in terms of UPQ are capable of "learning" from the marketplace to alter their bids to certain levels in order to chase the equilibrium price. This, in turn, results in a maximization of their revenue. In contrast, agents with ineffective factors are not capable of learning from the market. From our observation of the various simulations, with good correlations to the UPQ, a recommending agent's strategy profile changes quickly before the market converges and then becomes relatively stable after convergence.

4.2.4 *Stability.* To evaluate the stability of the market with respect to bidding strategies, we now consider what happens if some of the agents no longer follow the dominant strategies of Section 3.3.3. Here we assume the agents adopt a greedy strategy, meaning they bid as much as possible on every round to outbid others. To this end, we use the setting of the second experiment introduced in Section 4.2.1 with all nine agents taking the effective factors as their INQs. However, we select one recommending agent (say the first one) as the greedy bidder with the other agents still taking the dominant strategy. Here, all recommending agents are endowed with an initial credit of 65535. The greedy bidder always bids much higher than the others to get its recommendations shortlisted with the hope of making profit. However, this greedy bidder does not receive any more rewards from its recommendations when compared with the rewarded recommendations provided by the other, nongreedy, bidders. Indeed, the reward is not based on the bid price, but rather on the UPQ (for exactly this reason). With the same amount of reward with respect to the same level of UPQ, however, the greedy bidder pays much more for each of its

[17] The rational bidding strategy for those agents who cannot learn anything from the market is to bid as low as possible to lose less money; see Figure 8(c) and its explanation.



(a) Bidders' balance in a market with one greedy bidder

(b) Bidders' balance in a market with no greedy bidders

Fig. 12. Balance of bidders with effective and ineffective factors.

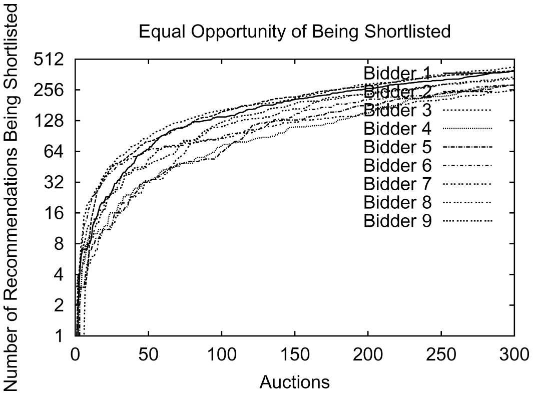


Fig. 13. Opportunity of being shortlisted (Experiment 1).

shortlisted recommendations. Therefore, the greedy bidder goes bankrupt over time, as shown in Figure 12(a), while the nongreedy bidders keep increasing their balance steadily. In comparison, when no greedy bidders participate, all recommending agents keep increasing their balance, as shown in Figure 12(b).

4.2.5 *Fairness.* We expect the market to be fair to all recommending agents irrespective of the recommendation method they use. To see this, we use the first experiment configuration introduced in Section 4.2.1. From Figure 13, it can be seen that the curves that represent the number of recommendations being shortlisted (including both rewarded and not rewarded) for each agent are close to each other, meaning all agents have an equal opportunity of being shortlisted. Thus, the market is fair to all agents whatever methods they use.

However, different methods do not necessarily have an equal opportunity of being rewarded, as shown in Figure 11(a). This, in turn, highlights the fact that a fair market does not mean that all agents are equally likely to receive rewards. Rather, the opportunity of being rewarded depends on the UPQ. Therefore, fairness of the market means all agents are treated the same.

4.3 Dealing with Multiple Recommendation Properties

Having evaluated the system properties with respect to the metrics stated in Section 3.2, this section considers the case where more than one recommendation property (Φ, introduced in Section 4.1.2) is evaluated by both the user and the recommending agents. This is important because many real recommendation methods evaluate more than one property (or feature) of recommendations [Burke 2002; Littlestone and Warmuth 1994] and it is important that our market-based recommender system perform well in such cases.

However, first, we need to establish the configurations of the three kinds of agents in our marketplace. Since the auctioneer agent simply acts as the organizer of the marketplace, rewarding the user-selected recommendations based on the UPQ, this agent remains the same as in Section 4.1.1. We still use the independent selection user model with $AT = 66$. Since it is not practical to gather up every possible case that contains an arbitrary number of properties (Φ) in one formula (for either UPQ or INQ) and to exemplify the correlations between these two qualities in a simple set of experiments, we begin the analysis with two properties involved for each quality function (meaning for both the user and the recommending agents). The more general cases in which each quality function evaluates more than two properties can be analyzed in the same way. To this end, the configuration of the user agent also remains unchanged, $Q(Rec) = 75\Phi_1(Rec) + 25\Phi_0(Rec)$. The recommending agents are each configured to evaluate two properties: some agents share both properties, some share only one property, and some share no property with the user's valuation of the recommendations. In this section, we consider eight recommending agents and their INQs that are configured as follows:

$$\begin{aligned}
q_1(Rec) = q_5(Rec) = 0.75\Phi_1(Rec) + 0.25\Phi_0(Rec),\\
q_2(Rec) = q_6(Rec) = 0.75\Phi_1(Rec) + 0.25\Phi_3(Rec),\\
q_3(Rec) = q_7(Rec) = 0.75\Phi_3(Rec) + 0.25\Phi_0(Rec),\\
q_4(Rec) = q_8(Rec) = 0.75\Phi_2(Rec) + 0.25\Phi_3(Rec).
\end{aligned} \tag{9}$$

Φ_0, Φ_1, Φ_2, and Φ_3 are configured as per Section 4.2.1. With these settings, we can see that q_1 and q_5 fully contain the effective factors, and they match the user's valuation of recommendations accurately. Likewise, q_2, q_6, q_3, and q_7 partially match the user's valuation, whereas q_4 and q_8 have no match. More formally, using a transformation of the UPQ, $Q(Rec) = (75\Phi_1(Rec)+25\Phi_0(Rec))/100$, to subtract each item in Equation Array (9), we can expect the four methods to exhibit the following correlations to the UPQ (where"$\ncong$" stands for "has no relationship to"):

$$\begin{aligned}
q_1(Rec) &= q_5(Rec) = 0.01 \cdot Q(Rec),\\
q_2(Rec) &= q_6(Rec) = 0.01 \cdot Q(Rec) + 0.25 \cdot (\,\Phi_3(Rec) - \Phi_0(Rec)\,),\\
q_3(Rec) &= q_7(Rec) = 0.01 \cdot Q(Rec) + 0.75 \cdot (\,\Phi_3(Rec) - \Phi_1(Rec)),\\
q_4(Rec) &= q_8(Rec) \ncong Q(Rec).
\end{aligned} \tag{9'}$$

Having configured the three kinds of agents, we are going to evaluate the market properties and validate that the correlations in Equation Array (9′)

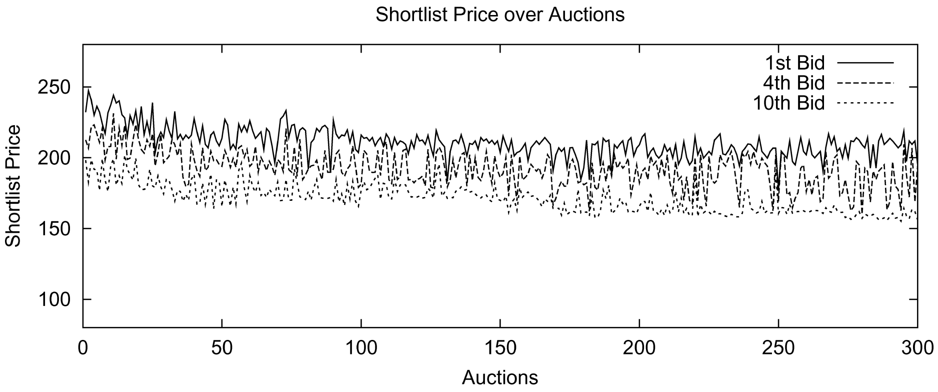


Fig. 14. Convergence of shortlisted prices.

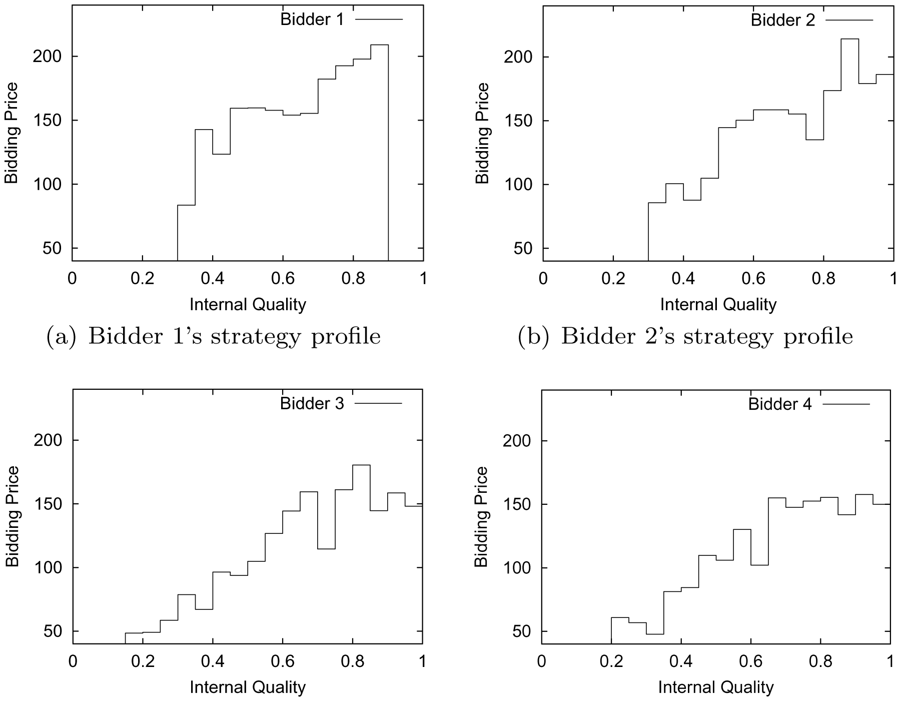
do effect the agents' bidding and learning behavior. Again, the evaluation begins with the most important system property—market convergence. Figure 14 again demonstrates that the market converges (after about 80 auctions), with at least one agent capable of relating its INQ to the UPQ (the first and the fifth agents in this experiment).

Using similar simulations to the ones of Section 4.2, we find that the market exhibits the same properties: namely, efficient shortlists, clear incentives for agents to bid, stability, and fairness. Thus, we do not further discuss these issues in this section. Instead, we will focus on how the different recommendation methods correlate their INQs to the UPQ. This problem can be decomposed into two subproblems:

(i) Can the agents relate their bids to their internal quality?
(ii) To what extent does each individual agent relate its INQ to the UPQ?

To this end, the strategy profiles for four agents (q_1, q_2, q_3, and q_4) at the point when the market reaches equilibrium are plotted in Figure 15. From Figure 15(a), we can see that the first agent bids its recommendations from INQ segments that are above the level of 0.65 at a level that is much higher than 160, which is actually the equilibrium price of the 10th bid (see Figure 14 after 80 auctions). Since the equilibrium price of the 10th bid represents the lowest price to be shortlisted, we refer to it as the *market access price*. For the first agent, both evaluation properties (Φ_1 and Φ_0) are the effective factors, and their weightings both match those in the UPQ. Thus, its INQ fully matches the UPQ. Being capable of relating its INQ to the UPQ, this agent can establish from which specific INQ segments its recommendations can be rewarded. From Figure 15(a), we can also see that bids from INQ segments that are below the level of 0.65 are lower than the market access price. Indeed, the first agent learns from the marketplace that these recommendations will not be rewarded, and so it decreases their price so as not to shortlist these items and avoid paying for them when they are unlikely to produce a return.

From Figure 15(b), we can see that the second agent bids its recommendations from very high INQ segments (higher than the level of 0.80) at a level that



(a) Bidder 1's strategy profile

(b) Bidder 2's strategy profile

(c) Bidder 3's strategy profile

(d) Bidder 4's strategy profile

Fig. 15. Strategy profiles of bidders with effective and ineffective factors.

is higher than the market access price. The second agent has one of its two evaluating properties (Φ_1) as the effective factor, and this contributes significantly to both the INQ and the UPQ (both with a weighting of 0.75). In this case, only a very high value of Φ_1 can give a high value of q_2 since Φ_1's weighting is much bigger than Φ_3's. Thus, very high INQs indicate high values of UPQ, and, therefore, such recommendations have good correlations with the user's preferences. Therefore, the agent only bids on very high INQ recommendations that are highly likely to be shortlisted. It does this to make profit without incurring a high risk of losing credits (i.e., being shortlisted but not rewarded).

From Figure 15(c), we can see that the third agent has few segments with bids higher than the market access price (compared to the first and second agents). The explanation is that, even though one of its two evaluating properties (Φ_0) is the effective factor, it contributes too little to its INQ (coefficient value 0.25). Therefore, its INQ cannot easily be related to the UPQ. With fewer concrete signals from the rewards received, it is difficult for the agent to relate its bids to its INQs. Thus the agent is not confident enough to bid for certain items at a very high price (since it has a high risk of losing credits without earning).

Figure 15(d) demonstrates that the fourth agent, having no effective factors, does not dare to bid high enough for any items from any segments to be shortlisted. It behaves in this way because it does not want to incur the risk of being shortlisted without receiving any reward. This uncertainty comes from the fact

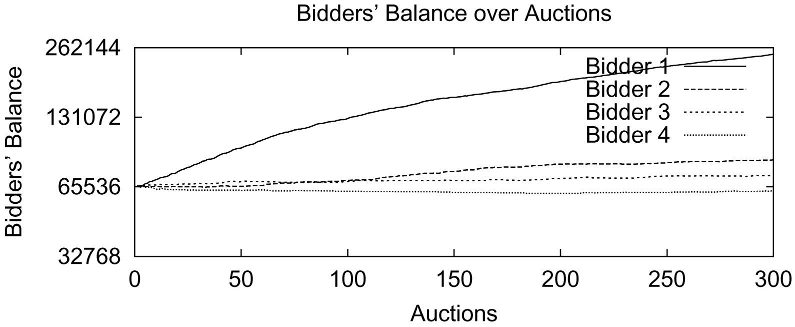


Fig. 16. Balance of bidders with effective and ineffective factors.

that the agent cannot effectively relate its INQ to the UPQ. Thus it does not know what items from which segments match the user's preference.

When taken together, these experiments show that the agents' confidence to relate their bids to the INQ decreases from the first agent to the fourth. Theoretically, this point can be shown in their INQ functions with respect to the UPQ (see Equations (9′)). Thus, the noise between the four agents' INQs and the UPQs is, respectively, 0, $0.25(\Phi_4(Rec) - \Phi_0(Rec))$, $0.75(\Phi_5(Rec) - \Phi_1(Rec))$, and full noise. Therefore, the agents' ability to relate their INQs to the UPQ is in decreasing order. On the other hand, since the agents' bids are based on rewards and rewards are based on the UPQ, the bids can be related to the UPQ. Thus, the agents' ability to relate their INQs to their bids is in decreasing order. This, in turn, affects their balance. Specifically, Figure 16 demonstrates that the more strongly an agent can relate its INQs to its bids, the more profit it will make.

4.4 Validating the System's Ability to Seek Out the Best Recommendation

Having evaluated the market with respect to the metrics listed in Section 3.2 and the correlation between the INQ and the UPQ of the recommendations, this section evaluates the system's ability to seek out the best item from all the source recommendations. This is clearly an important feature from the user's viewpoint, since if the system cannot recommend the best items, the user will not use it.

To evaluate this aspect of the system, we use the first experiment discussed in Section 4.2.1 and trace the bidding price of the recommendation with the highest UPQ value selected by the first agent (see Figure 17, in which the cross points represent the bidding price of this particular recommendation). From this, we can see that this recommendation's bidding price keeps increasing till it converges to the first bid price of the shortlisted items. This means that as long as the first agent chooses the highest UPQ recommendation to bid in an auction round (after the market converges), this item is always displayed in the first slot of the sidebar of the user's browser. Therefore, in case of either user model (independent selection or search-till-satisfaction), this recommendation will be selected by the user, since the first shortlisted recommendation has the highest UPQ. This result shows that the system is capable of seeking out the best recommendation and presenting it to the user.

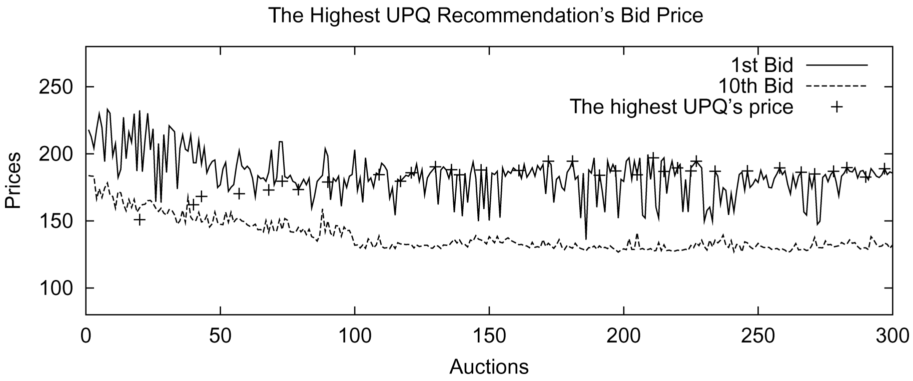


Fig. 17. The best recommendation's bidding price.

5. CONCLUSIONS AND FUTURE WORK

This article has investigated the feasibility of building a recommender system as a computational economy in which the various recommending agents (embodying different methods and having different qualities) compete to get their recommendations displayed to users. Through the development, analysis, and evaluation of our marketplace design, we have demonstrated that the system should be able to make good recommendations to users. In more detail:

(1) The market works as a means of coordinating various recommendation methods in an overarching system. Specifically, as there is no universal best recommendation method for all situations, there is a need to incorporate multiple methods into a single system so that each such system can contribute the best recommendations in the various circumstances that might arise. This ensures the peak performance of the overall system.

(2) Our marketplace successfully incentivizes the recommending agents to bid in a manner that is consistent with the user's preferences. Specifically, the market mechanism uses the reward regime to reflect the user's satisfaction of the recommendations. This ensures the agents receiving rewards frequently become aware of the types of recommendations that best satisfy the user.

(3) By analysis, our market is shown to be capable of shortlisting recommendations in decreasing order of UPQ. By defining a proportional reward mechanism, the market relates the bidding prices to the user's sidebar slots and to the UPQ levels. After market convergence, the higher a recommendation's UPQ, the higher its price is, and, thus, the higher its shortlisted position.

(4) By simulation, our market mechanism is shown to be capable of successfully correlating the two perspectives of recommendation quality (internal and user perceived). As discussed in Section 2, none of the previous systems correlate them together in an integrated manner. Specifically, our market uses the reward and price regime to quantify the UPQ and the various INQ measures of the individual recommenders. In this system, the bidding price represents the cost of advertising a recommendation with a specific INQ level

and the reward reflects the actual value of a recommendation with a specific UPQ level. Over time, the agent can align its INQ to its bidding price, and its bidding price to its corresponding reward, and its reward to the UPQ. This connection enables an agent to relate its INQ to the UPQ.

(5) By decomposing the INQ into linear combinations of evaluation scores on different properties of recommendations, we find that the more the effective factors influence the recommending agents' INQ, the stronger is their ability to relate their INQ to the UPQ and to make high-quality recommendations.

Having demonstrated the viability of this approach, the next step is to undertake a field trial of our recommender system, in which we replace our simulated users and recommendations with real ones. This will enable us to fully demonstrate the power and applicability of the approach and to ensure that the results we have produced through simulations actually hold in practice. Additionally, there are several other issues that come to the fore in turning our proof-of-concept system into a fully functioning and operational recommender system: (i) there is a need to endow the recommender agents with the ability to effectively learn user's interests so that they are able to quickly and frequently identify the best items while still maximizing their revenue (see Wei et al. [2005] for our initial work on this); (ii) while there is much scope for different recommender agents to share information about recommendations and user interactions in order to improve the computational efficiency of the system, this sharing needs to be balanced against issues related to maintaining trust and privacy for the users of our system; (iii) the issue of scalability of our approach as large numbers of documents are incorporated needs further investigation; (iv) communication costs between user agents, recommending agents, and the auctioneer agent may also need to be factored into the system when large numbers of users and recommendation methods participate.

REFERENCES

BALABANOVIC, M. AND SHOHAM, Y. 1997. Fab: Content-based, collaborative recommendation. *Commun. ACM 40*, 3, 66–72.

BERNERS-LEE, T., CAILLIAU, R., GROFF, T.-F., AND POLLERMANN, B. 1992. World-Wide Web: The information universe. *Electron. Netw. Res. Appl. Policy 1*, 2, 52–58.

BILLSUS, D. AND PAZZANI, M. J. 2000. User modeling for adaptive news access. *User Model. User-Adapt. Interact. 10*, 2–3, 147–180.

BOHTE, S. M., GERDING, E., AND POUTRÉ, H. L. 2001. Competitive market-based allocation of consumer attention space. In *Proceedings of the 3rd ACM Conference on Electronic Commerce* (Tampa, FL). 202–205.

BOHTE, S. M., GERDING, E., AND POUTRÉ, H. L. 2004. Market-based recommendation: Agents that compete for consumer attention. *ACM Trans. Internet Tech. 4*, 4.

BREESE, J. S., HECKERMAN, D., AND KADIE, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence*. 43–52.

BURKE, R. 2002. Hybrid recommender systems: Survey and experiments. *User Model. User-Adapt. Interact. 12*, 4, 331–370.

CLAYPOOL, M., GOKHALE, A., MIRANDA, T., MURNIKOV, P., NETES, D., AND SARTIN, M. 1999. Combining content-based and collaborative filters in an online newspaper. In *Proceedings of the ACM SIGIR Workshop on Recommender Systems* (Berkeley, CA).

CLEARWATER, S. H., Ed. 1996. *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific, Singapore.

DASH, R. K., PARKES, D. C., AND JENNINGS, N. R. 2003. Computational mechanism design: A call to arms. *IEEE Intell. Syst. 18*, 6, 40–47.

DEROURE, D. C., HALL, W., REICH, S., PIKRAKIS, A., HILL, G. J., AND STAIRMAND, M. 2001. Memoir—an open framework for enhanced navigation of distributed information. *Inform. Process. Manage. 37*, 53–74.

GOLDBERG, D., NICHOLS, D., OKI, B., AND TERRY, D. 1992. Using collaborative filtering to weave an information tapestry. *Commun. ACM 35*, 12, 61–70.

GONZALEZ, G., LOPEZ, B., AND DE LA ROSA, J. L. 2004. Managing emotions in smart user models for recommender systems. In *Proceedings of the 6th International Conference on Enterprise Information Systems*. 187–194.

HERLOCKER, J. L., KONSTAN, J. A., AND RIEDL, J. 2000. Explaining collaborative filtering recommendations. In *Proceedings of ACM Conference on Computer Supported Cooperative Work* (Philadelphia, PA). 241–250.

HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G., AND RIEDL, J. T. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inform. Syst. 22*, 1, 5–53.

HILL, W., STEAD, L., ROSENSTEIN, M., AND FURNAS, G. 1995. Recommending and evaluating choices in a virtual community of use. In *Proceedings of the ACM Conference on Human Factors in Computing Systems, (CHI'95)*. 194–201.

HOWE, A. E. AND DREILINGER, D. 1997. Savvysearch: A meta-search engine that learns which search engines to query. *AI Mag. 18*, 2, 19–25.

JENNINGS, N. R. 2001. An agent-based approach for building complex software systems. *Commun. ACM 44*, 4, 35–41.

JENNINGS, N. R., FARATIN, P., LOMUSCIO, A. R., PARSONS, S., SIERRA, C., AND WOOLDRIDGE, M. 2001. Automated negotiation: Prospects, methods and challenges. *J. Group Decis. Negotiat. 10*, 2, 199–215.

KAGEL, J. H. AND ROTH, A. E., Eds. 1995. *The Hand Book of Experimental Economics*. Princeton University Press, Princeton, NJ.

KLEMPERER, P. 1999. Auction theory: A guide to literature. *J. Econom. Surv. 13*, 3, 227–286.

KONSTAN, J. A., MILLER, B. N., MALTZ, D., HERLOCKER, J. L., GORDON, L. R., AND RIEDL, J. 1997. Grouplens: Applying collaborative filtering to usenet news. *Commun. ACM 40*, 3, 77–87.

KOYCHEV, I. 2000. Gradual forgetting for adaptation to concept drift. In *Proceedings of ECAI2000 Workshop Current Issues in Spatio-Temporal Reasoning* (Berlin, Germany). 101–106.

KRULWICH, B. 1997. Lifestyle finder: Intelligent user profiling using large-scale demographic data. *AI Mag. 18*, 2, 37–45.

LANG, K. 1995. NewsWeeder: Learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*. Morgan Kaufmann, San Mateo, CA, 331–339.

LITTLESTONE, N. AND WARMUTH, M. 1994. The weighted majority algorithm. *Inform. Comput. 108*, 2, 212–261.

MALTZ, D. AND EHRLICH, K. 1995. Pointing the way: Active collaborative filtering. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'95)*, (Denver, CO). 202–209.

MCAFEE, R. P. AND MCMILLAN, J. 1987. Auctions and bidding. *J. Econom. Lit. 25*, 2 (June), 699–738.

MILGROM, P. 1989. Auctions and bidding: A primer. *J. Econom. Perspect. 3*, 3 (Summer), 3–22.

MONTANER, M., LOPEZ, B., AND DELA, J. L. 2003. A taxonomy of recommender agents on the internet. *Artific. Intell. Rev. 19*, 285–330.

MOREAU, L., ZAINI, N., ZHOU, J., JENNINGS, N. R., WEI, Y. Z., HALL, W., ROURE, D. D., GILCHRIST, I., O'DELL, M., REICH, S., BERKA, T., AND NAPOLI, C. D. 2002. A market-based recommender system. In *Proceedings of the 4th International Workshop on Agent-Oriented Information Systems* (AOIS-2002, Bologna, Italy). 50–67.

MULLEN, T. AND WELLMAN, M. P. 1995. A simple computational market for network information services. In *Proceedings of the 1st International Conference on Multiagent Systems*. AAAI Press, Menlo Park, CA/MIT Press, Cambridge, MA, 283–289.

PAZZANI, M. 1999. A framework for collaborative, content-based and demographic filtering. *Artific. Intell. Rev. 13*, 5–6, 393–408.

PAZZANI, M., MURAMATSU, J., AND BILLSUS, D. 1996. Syskill & Webert: Indentifying interesting Web sites. In *Proceedings of the 13th National Conference on Artificial Intelligence*. 54–61.

PENNOCK, D. M., HORVITZ, E., AND GILES, C. L. 2000. Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*. AAAI Press, Menlo Park, CA/MIT Press, Cambridge, MA, 729–734.

PINKERTON, B. 2000. Webcrawler: Finding what people want. Ph.D. dissertation, University of Washington, Seattle, WA.

POPESCUL, A., UNGAR, L. H., PENNOCK, D. M., AND LAWRENCE, S. 2001. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence* (UAI-2001, Seattle, WA), 437–444.

RESNICK, P., IACOVOU, N., SUCHAK, M., BERGSTORM, P., AND RIEDL, J. 1994. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work* (Chapel Hill, NC), 175–186.

RESNICK, P. AND VARIAN, H. R. 1997. Recommender Systems. *Commun. ACM 40*, 3, 56–58.

REYNOLDS, K. 1996. Agorics, Inc. Available online at `http://www.agorics.com/library/auctions.html`.

RICH, E. 1979. User modeling via stereotyps. *Cogn. Sci. 3*, 329–354.

ROTH, A. E. 2002. The economist as engineer: Game theory, experimental economics and computation as tools of design economics. *Econometrica 70*, 4, 1341–1378.

SALTON, G. 1989. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, Reading, MA.

SAMUELSON, P. A. AND NORDHAUS, W. D. 2001. *Economics*, 17th ed. McGraw-Hill/Irwin, New York, NY.

SANDHOLM, T. W. 1999. Distributed rational decision making. In *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, G. Weiss, Ed. MIT Press, Cambridge, MA, 201–258.

SARWAR, B. M., KONSTAN, J. A., BORCHERS, A., HERLOCKER, J., MILLER, B., AND RIEDL, J. 1998. Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work* (Seattle, WA). 345–354.

SHARDANAND, U. AND MAES, P. 1995. Social information filtering: Algorithms for automating "word of mouth." In *Proceedings of the Conference on Human factors in Computing Systems*. ACM Press, New York, NY, 210–217.

SHARMA, R. AND POOLE, D. 2001. Symmetric collaborative filtering using the noisy sensor model. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence* (UAI-2001, Seattle, WA). 488–495.

SHETH, B. AND MAES, P. 1993. Evolving agents for personalized information filtering. In *Proceedings of the 9th Conference on Artificial Intelligence for Applications* (CAIA'93, Orlando, FL). 345–352.

TERVEEN, L. AND HILL, W. 2001. Beyond recommender systems: Helping people help each other. In *HCI in the New Millennium*, J. Carroll, Ed. Addison-Wesley, Reading, MA.

TERVEEN, L., HILL, W., AMENTO, B., MCDONALD, D., AND CRETER, J. 1997. PHOAKS: A system for sharing recommendations. *Commun. ACM 40*, 3, 59–62.

TESFATSION, L. 2002. Agent-based computational economics: Growing economies from the bottom up. *Artific. Life 8*, 1, 55–82.

TURBAN, E., LEE, J., KING, D., AND CHUNG, H. M. 2000. *Electronic Commerce: A Managerial Perspective*. Prentice Hall, Englewood, Cliffs, NJ.

VARIAN, H. R. 2003. *Intermediate Microeconomics: A Modern Approach*, 6th ed. W. W. Norton, New York, NY.

VICKREY, W. 1961. Counterspeculation, auctions, and competitive sealed tenders. *J. Finance 16*, 1 (Mar), 8–37.

WEI, Y. Z., MOREAU, L., AND JENNINGS, N. R. 2003a. Market-based recommendations: Design, simulation and evaluation. In *Proceedings of the 5th International Bi-Conf*

Workshop on Agent-Oriented Information Systems (AOIS-2003, Melbourne, Australia). 22–29.

WEI, Y. Z., MOREAU, L., AND JENNINGS, N. R. 2003b. Recommender systems: A market-based design. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multi Agent Systems* (AAMAS03, Melbourne, Australia). 600–607.

WEI, Y. Z., MOREAU, L., AND JENNINGS, N. R. 2005. Learning users' interests by quality classification in market-based recommender systems. *IEEE Trans. Knowl. Data Eng*. To appear.

WELLMAN, M. P. AND WURMAN, P. R. 1998. Market-aware agents for a multiagent world. *Robot. Auton. Syst. 24*, 115–125.

WURMAN, P. R., WELLMAN, M. P., AND WALSH, W. E. 1998. The Michigan Internet Auctionbot: A configurable auction server for human and software agents. In *Proceedings of the 2nd International Conference on Autonomous agents*. ACM Press, New York, NY, 301–308.

YAN, T. AND GARCIA-MOLINA, H. 1995. SIFT—a tool for wide-area information dissemination. In *Proceedings 1995 USENIX Technical Conference* (New Orleans, LA). 177–186.

YU, K., XU, X., ESTER, M., AND KRIEGEL, H.-P. 2003. Feature weighting and instance selection for collaborative filtering: An information-theoretic approach. *Knowl. Inform. Syst. 5*, 2 (Apr.), 201–224.

ZAMBONI, G. 1998. Search tools. Technical rep. University of Cordoba, Cordoba, Spain.

Received December 2003; accepted April 2005