A Computational Market for Information Filtering in Multi-Dimensional Spaces

Karakoulas, G.J.; Ferguson, I.A.

Publisher's version / Version de l'éditeur:

Proceedings of the AAAI Fall Symposium on AI Applications in Knowledge Navigation and Retrieval, 1995

A Computational Market for Information Filtering in Multi-Dimensional Spaces

Grigoris J. Karakoulas
Dept. of Systems and Computer Engineering
University of Carleton,
1125 Colonel By Drive,
Ottawa ON, Canada K1S 5B6
grigoris@sce.carleton.ca

Innes A. Ferguson
Interactive Information Group
Institute for Information Technology
National Research Council,
Ottawa ON, Canada K1A 0R6
innes@ai.iit.nrc.ca

Abstract

This paper presents the computational market of SIGMA (System of Information Gathering Market-based Agents) as a model of decentralized decision making for the task of information filtering in multi-dimensional spaces such as the Usenet netnews. Different learning and adaptation techniques are integrated within SIGMA for creating a robust network-based application which adapts to both changes in the characteristics of the information available on the network as well as to changes in individual users' information interests. The functionality of the system is discussed together with work underway for its evaluation.

Introduction

As the space of electronically stored information continues to expand across computer networks, the need to reason about users' interests within this multi-dimensional and largely unstructured space becomes imperative. The problem of designing an information filtering (IF) system is challenging because of the volatility of the space and the limited ability of users to specify global interests over an inherently uncertain area of this space.

When considering the design of an IF system the inherent distribution and dynamics of today's information networks would naturally appear to suggest some form of robust, decentralized, multiagent design approach (Ferguson 1995). In addition, since network users' information delivery needs may be changing over time, it would seem necessary to endow such agents with appropriate behaviors for adapting to relevant changes in the environment. The objective of our work is to develop a multiagent framework in which the various agents involved in the IF task are able to adapt to changes in both the information space and in users' interests for information.

Within the artificial intelligence community there has been research on building intelligent IF agents that augment and enhance current information services over the Internet. Example services include electronic mail (Lashkari et al. 1994; Mitchell et al. 1994); NetNews (Fischer & Stevens 1991; Jennings & Higuchi 1992; Baclace 1993; Sheth & Maes 1993; Yan & Molina 1994); and electronic bibliographies and catalogues of products (Foltz & Dumais 1992; Hammond et al. 1994). In these works two general approaches to the IF problem can be distinguished: (i) the *static* approach, according to which different types of knowledge are explicitly pre-specified and embedded into the agent for reasoning about users' requests and the information environment of the IF task; and (ii) the *dynamic* approach, according to which the agent improves its performance on the IF task over time by exploiting user feedback in order to learn associations between users' interests and information items.

To deal with issues of task complexity and scaleability most of the aforementioned work has followed the dynamic approach. Thus, various machine learning techniques have been employed for embedding into agents the capabilities of learning and adaptivity. These techniques include memory-based reasoning, neural networks, reinforcement learning and genetic algorithms. Although the latter technique provides a multiagent adaptation and coordination mechanism it does not support the inherent asynchrony of distributed information networks and it does not exploit sharing of IF agents. The latter feature is important especially when the system is confronted with new user requests that are similar to existing ones. Lashkari et al. (1994) have proposed a multiagent framework for supporting collaboration among agents. Each agent in this framework builds a model of trusting its peers and collaboration is initiated according to these models. On the other hand, the metaphor of economic markets has been of particular interest for developing multiagent systems since it provides well-defined mechanisms for adaptation and competition/collaboration among agents (e.g. Wellman's work on distributed configuration design (Wellman 1994)).

In this paper we briefly present the market model of SIGMA (System of Information Gathering Market-based Agents) and its application to the task of personalized filtering of Usenet netnews. This is a typical class of IF tasks characterized by streams of incoming textual information broadcast from remote sources and by partial descriptions of users' information interests. Besides our current application, other IF examples include: digital library tasks such as filtering and retrieval of journal articles and technical reports; and news service tasks such as personalized filtering of newswire feeds. The remainder of the paper deals first with how different learning and adaptation techniques – i.e. reinforcement learning, relevance feedback and bidding – are integrated within SIGMA for creating a robust network-based application which adapts to both changes in the characteristics of the information available on the network, as well as to changes in individual users' information delivery requirements; and secondly with how the computational economy of SIGMA is implemented using multiple types of agents that are defined upon the CALVIN (Communicating Agents Living Vicariously In Networks) framework.

The SIGMA Computational Market

Like in most other large-scale distributed systems, the activities in an IF system are characterized by competition for resources. On the one hand, the users that arrive in the system are endowed with limited resources such as time and money, and different interests. They pose queries to the IF system and they aim to spend their resources as efficiently as possible in order that their interests be matched as satisfactorily as possible against dynamic information streams. On the other hand, the system contains agents which represent automated indexing and filtering procedures, informational resources and human-computer interaction procedures. These procedures need to be allocated computation time in order to serve as efficiently as possible the various demands for information within the system.

In the field of economics, the problem of allocating limited or scarce resources among competing agents has been extensively studied in the context of markets. The latter are models of decentralized decision making where each agent processes only local information in order to evaluate and make decisions regarding goods and services. The backbone of this decentralized decision making process is the price mechanism. Prices are inherently local to each market. They impose low information requirements since they effectively reflect the efficiency of resource allocation within a market. Because of their decentralized and local nature, markets can spontaneously be developed for the exchange of goods according to local needs; they can also adjust to unforeseeable changes.

Due to such advantages over centralized decision making, economic theory has provided the basis for the development of computational markets for job-scheduling problems in distributed computing (Miller & Drexler 1988; Huberman & Hogg 1993). Most of these computational frameworks mix market and other heuristic mechanisms. Within artificial intelligence, Wellman has proposed the WALRAS computational economy for distributed planning (Wellman 1994). The design of WALRAS is purely couched in terms of ideas about markets from theoretical economics. It is based on an auction mechanism among producers and consumers for determining market equilibrium prices and allocating resources for the production and consumption of goods. Wellman has applied the WALRAS architecture to the task of distributed configuration design. Compared with the IF task this is a relatively well structured, static task that fits well into the WALRAS framework. The design assumptions of WALRAS, however, incur four main limitations to applying this computational economy to our IF task.

First, WALRAS computes equilibrium prices without accounting for any ongoing market activity. Second, it makes the assumption of "perfect competition", namely that each agent is small enough so that its actions cannot have any effect on prices. For this reason prices are taken as given. In our case prices should depend on the performance history of each agent's actions since an increase in market competition may enhance the adaptivity of our system for coping with changes in both the dynamics of the information environment (e.g. new information streams created) and the arrival of new users. Third, in the WALRAS computational economy the consumers are assumed to have utility functions that reflect the global preferences of the user over the entire search space of the task. This imposes a restriction on adding new goods to the system while it is running. Moreover, in our task it is rather unrealistic to assume that the user's global preferences for information items (e.g. news articles, technical reports etc.) can be cast into a pre-defined utility function, especially since users' interests change. There is therefore a need for a learning procedure that can adapt to the user's preferences by using local feedback from the user. Fourth, the allocation that results from bidding in WALRAS is only for a single good. An approach with multi-good allocation can enhance the scaleability of the computational economy to more complex tasks such as the IF task.

Similarly to economic markets, the computational economy of SIGMA can be defined in terms of goods and agents that produce and consume those goods. The *goods* traded in SIGMA are information items (e.g. news articles) in different representation forms depending on the stage of processing. The agents are of two general categories: (i) the *consumer* agents and (ii) the *producer* agents. A consumer represents a user's query for a filtering activity over a time period. At each stage of this activity the consumer is endowed with a budget M which is a constraint on how much the consumer can spend on buying goods from producers at prices that depend on the bidding process and on the relevance of each good. The budget should reflect the limited resources of the user (e.g. time and money) in the respective task. Given this constraint the decision task of a consumer at each time t amounts to buying from the set of the K best producers at time t, denoted by x_t, a vector of goods $g_t = [g_{t1}, \ldots, g_{tn}]$ such that the user's cumulative benefit from buying goods in the long term is maximized, i.e.

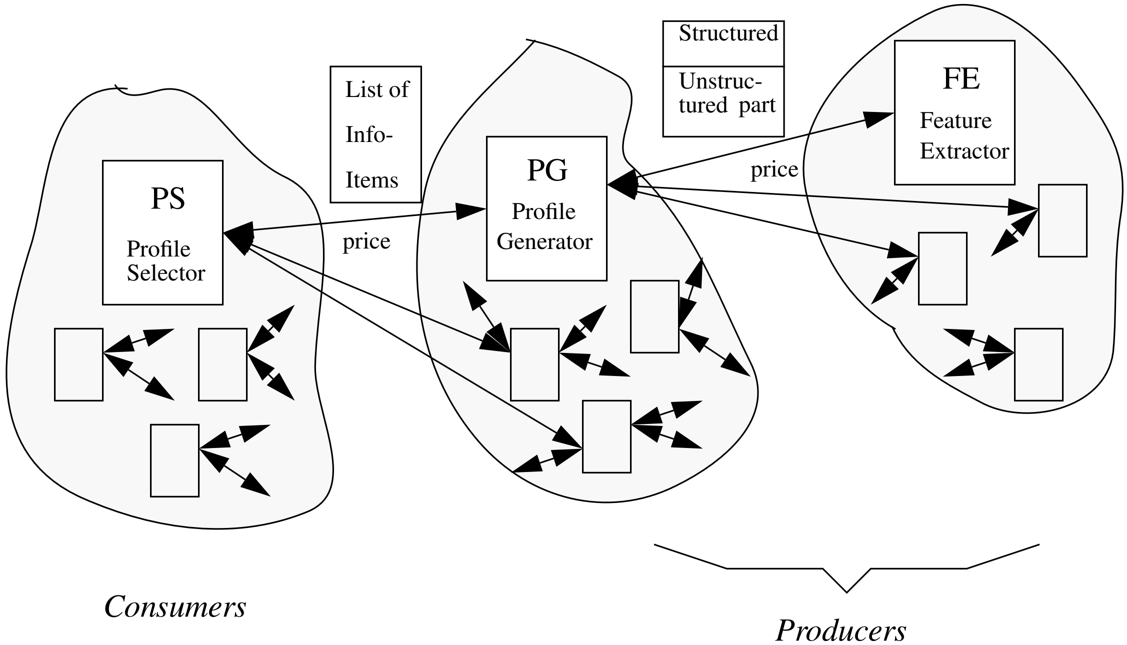

Figure 1: The flow of goods in the IF application of SIGMA

$$max_{g_t} \sum_{t=0}^{\infty} \gamma^t r(x_t, g_t) \quad (1)$$

where γ, $0 < \gamma < 1$, is the discount factor and $r(x_t, g_t)$ is a vector whose elements are the rewards received from the user as feedback when buying each of the goods in g_t. These rewards reflect the contribution of the goods in g_t towards achieving the goal of the user when submitting a request for a task, e.g. a filtering activity. Through this feedback the consumer should learn how to allocate its budget among the producers as satisfactorily as possible. It should be noted that, at each time, a consumer allocates its budget among the *K* best producers and buys the best goods from these producers instead of spending its budget on a single producer. This diversification in purchase behavior is dictated by the need for robustness in order to cope with the uncertainty of the task. We describe the allocation policy in more detail in the context of the IF task later on.

The other category of agents, producers, transform goods from an input form into an output form according to their technologies. In response to a consumer's demand for goods they enter the local market and compete with each other to serve as efficiently as possible the demands for goods from other agents — consumers or other producers — within the market. When a producer is selected at bidding its decision task amounts to choosing which articles to produce so that its profit is maximized. The profit is defined as the difference between revenues from selling the output and costs from producing it. The selling price for each output unit of a producer *j* is defined as a function of the producer's performance history, namely

$$p_t^j = f(E_{t-1}^j, B_{t-1}^j) \quad (2)$$

where E_{t-1}^j is the efficiency of the producer's output with respect to the task up to time *t-1*. B_{t-1}^j is the ratio of the number of times the producer *j* has been selected in a bid over the total number of bids made up to *t-1*. Thus, given an initial population of similar producers the price mechanism of (2) will lead to increasingly specialized producers that bid up their prices depending on the consumer's budget. This budget may depend on the priority and complexity of the user's task.

The Netnews IF Producers and Consumers

Figure 1 depicts the SIGMA computational market for netnews IF. There are two types of producers: (i) the Feature Extractors (FE) which transform each news article (a generic input good) into a document indexing representation; and (ii) the Profile Generators (PG) which are mid-producers since each of them takes as input a subset of the output of the FE producers and transforms it into a profile (a compact representation of documents) which the PG agent expects will satisfy the respective consumer's interests. We next define the behaviors of these two types of producers as well as the behavior of the Profile Selector (PS) which is the consumer agent in the SIGMA market.

FE Agent. Its purpose is to transform raw information contained in an article into a feature-based knowledge representation formalism that can be used to reason about the article. The vector space model (VSM) is used for implementing this formalism (Belkin & Croft 1992). Like any other type of document an article contains structured information in its header part (author, location, newsgroup, etc.) as well as unstructured information in its text part. The agent uses a stop list together with the WordNet thesaurus in order to discard noisy words and construct a subset of keywords. Weights are then calculated for each keyword by using the term frequency and the inverse document frequency factors. The agent stores the VSM representation of each processed article into a repository.

PG Agent. Upon creation it is initialized with a profile that contains the keywords specified by the user. When selected at bidding the agent explores the newsgroups and selects the articles for which its marginal profit is greater than a given threshold. The filtering technology that is employed for producing the selection of articles is based upon similarity metrics of the VSM representation. To ensure proactivity in the agent's behavior we allow the agent to follow this greedy policy with probability p and select any other article with probability $1-p$. The marginal profit is defined as a function of the similarity of the article and the profile of the agent. The articles are stored in a catalogue with the most similar article at the top. Given the articles that have been accepted by the user the agent applies relevance feedback to update its profile. This is done by adjusting the weights of existing keywords and by including new terms. To calculate its bidding price the agent uses a particular form of (2) where the performance efficiency is defined in terms of the agent's precision measure.

PS Agent. The behavior of this agent is based on reinforcement learning. The agent initially allocates its budget uniformly to the PG agents that have won the bidding and selects from each of them the respective top-ranked articles. For each article that the user accepts or rejects the PS agent receives a positive or negative reinforcement. This reinforcement is a function of the price that the agent has paid for the respective article. Such reinforcement signals enable the agent to learn over time a policy of buying the best articles from the most efficient producers. We have developed a Q-learning algorithm for implementing the learning behavior of the PS agent based on earlier work in (Karakoulas 1995).

It is worth noting that the SIGMA architecture exhibits two levels of learning and adaptation. At the individual agent level the PS and the PG agents learn and adapt their behavior via Q-learning and relevance feedback using the users' local evaluation for articles. At the inter-agent level the price mechanism and the purchase of articles from more than one agent enhance the robustness of the system with respect to the uncertainties of the environment.

Information Flow in Netnews IF

A query submitted by a user to SIGMA consists of fields that correspond to the structured part of a news article — namely author(s), subject(s), newsgroup(s) and organization(s) — and a field that corresponds to the unstructured part — namely text keyword(s). These fields may be partially filled in by the user according to his/her prior knowledge. In addition, the user can specify the values of global parameters such as the maximum number of articles desired, the earliest date of an article and maximum and minimum number of lines that an article may have. When a new query is submitted to SIGMA a PS agent is generated for representing within the computational economy the particular user's demand for news articles as specified in the query. The PS agent is endowed with a budget that is sufficient for buying the maximum number of articles desired.

Given this demand the PS agent enters a local market of PG agents by posting the query to the bulletin board of the market. This bulletin board serves as the communication medium for the PS-PG transactions. If such a local market does not exist a new one is created that consists of a pre-specified number of PG agents. The profile of a new PG agent is initiated either with the contents of the query or with the contents of an existing PG agent in another local market that has a profile similar to the one contained in the query. Once a particular demand for articles is advertised each of the PG agents calculates its bidding price by using historical data regarding the demand of its articles by the PS agent. Initially, when such data do not exist the bidding price of each of the PG agents is set to a price threshold which is equal to the ratio of the budget M of PS over the maximum number of articles N that PS wants to buy. Each PG agent posts its price to the bulletin board of the market. The PS agent selects the PG agents whose bidding prices are above the aforementioned price threshold and informs them for this selection. These PG agents in turn post their demand for articles to the market of the FE agents.

Each PG agent is assumed to pay a flat cost c for buying an output unit from an FE agent. In general, this cost can vary depending on the complexity of the information item and the information source (i.e. access fees). The population of FE agents is assumed to be allocated to newsgroups according to a static scheme. This scheme can become dynamic by implementing a market between the population of FE agents and the set of newsgroups. Since the information load over newsgroups varies this market can give rise to an efficient allocation of resources depending on the load of newsgroups. The FE agents sell articles transformed into the VSM representation. Each PG agent buys the articles that maximize its profits and stores them in a list that is sorted according to the similar-

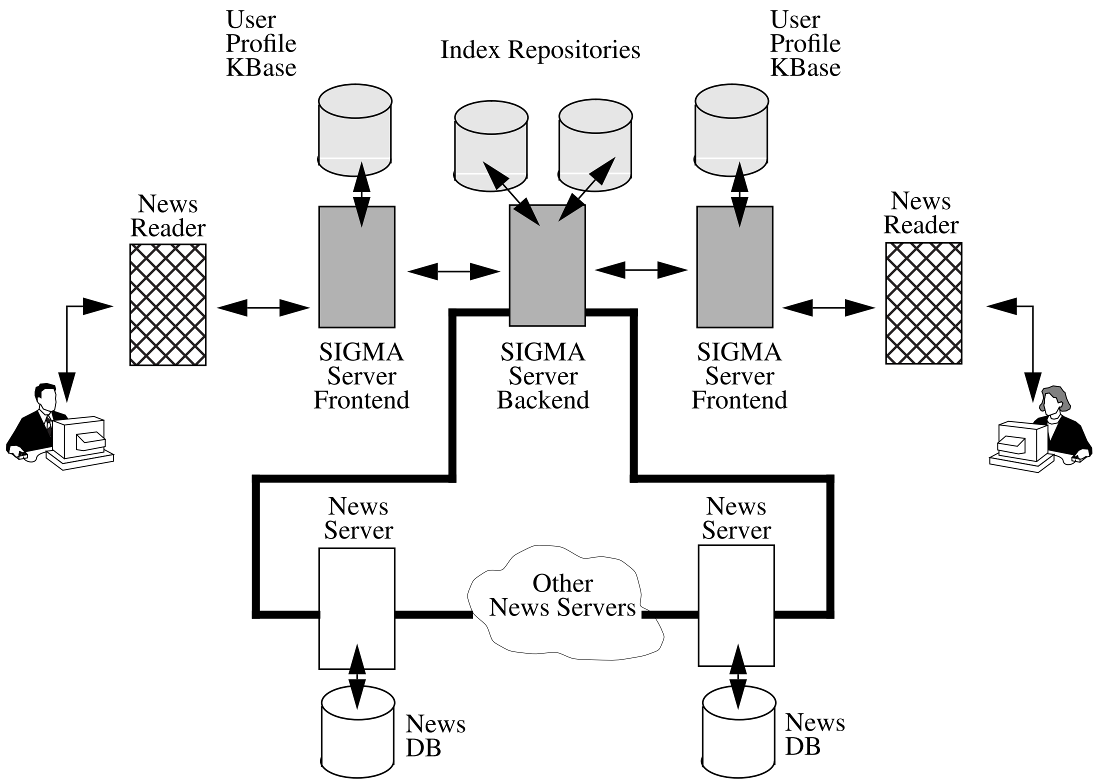

Figure 2: The system architecture

ity of each article with the profile of the agent. It then posts the list to the bulletin board of its PS-PG local market.

When the foregoing production phase of the PG agents is finished the PS agent starts buying articles from the PG lists; it allocates its budget amongst the top-ranked articles of the PG lists by following the policy that maximizes its current estimate of the expected cumulative reward in (1). The PS agent sends the articles purchased to the user via an interface module agent. The latter is responsible for both the presentation of the articles to the user and the transmission of user feedback (i.e. reinforcement) to the PS agent. When the PS agent receives a reinforcement signal for an article it updates its evaluation estimate and propagates this signal to the PG agents that produced that article. The latter agents update their profile using the VSM representation of the particular article. This completes the cycle of inter-agent information flow between SIGMA and the user for the IF task. We next give a description of SIGMA for this task from the system point of view.

Discussion

The SIGMA market model described above is implemented as a collection of specialized CALVIN agents (Ferguson & Davlouros 1995). The CALVIN (Communicating Agents Living Vicariously In Networks) framework is an open architecture for facilitating the development of agent-oriented applications. As such, the framework provides application developers with a powerful set of agent programming tools including libraries of intra- and inter-agent protocols (e.g. KQML (Finin et al. 1992)), sensory and effectory apparatus, internal behavior APIs, persistent storage management, and (currently under consideration) CORBA compliance.

The IF system (see Figure 2) can be divided into two main components. The first, a per-user SIGMA Server Frontend, lies between the user's News Reader and the SIGMA Server Backend. The Frontend processes user queries, generates and maintains profile information reflecting the user's interests, and manages the delivery of articles to the user's News Reader. These articles are obtained from an NNTP News Server via the SIGMA Server Backend. The Frontend also processes feedback from the user regarding the relevance/quality of the articles read; this information is used to update the profile knowledge base reflecting the user's evolving interests. Profile Selector (PS) and Profile Generator (PG) agents, the main "protagonists" in the SIGMA computational economy (as described in section 2), reside in the Frontend component. The SIGMA Server Frontend, in addition, performs multi-user profile management; this entails two operations: (i) new users get similar matching profiles from existing users by cloning new PG agents from existing ones; and (ii) bankrupted PG agents are collected as garbage.

The second component, the SIGMA Server Backend, autonomously generates and maintains a user-independent index of all Usenet articles. The index, which is maintained by Feature Extractor (FE) agents, is implemented as a knowledge base which contains feature-based representations of each raw article retrieved. As described in section 2, use is made of the Wordnet lexical reference system, term stops lists, and term and inverse document frequency factors to generate a compact and efficient index of Usenet articles. The SIGMA Server Backend can be accessed by multiple simultaneous per-user Frontend components.

An advantage of the system is that it is built on top of conventional news reader applications (in our current application, Mosaic on the World-Wide Web) in order to augment their functionality while at the same time relieving the user from the burden of learning new system features. The user does not need to know about the operation of the SIGMA Server (Frontend or Backend). Moreover, he/she can always bypass this operation by requesting to read all the articles of a newsgroup.

The current focus of our work is on system evaluation. We propose to evaluate the IF application through two sets of experiments. The first of these concerns performance evaluation of the computational economy with respect to parameters such as initial number of PG agents assigned to each user query, number and initial weights of terms in the VSM representation, producers' marginal profit thresholds and consumers' budgets, declaration of bankruptcy status for PG agents, and the PG and PS agents' exploration factors. The second set of experiments involves measuring the (hopefully) improved access to Usenet that it provides to users. In addition to performing informal user surveys to gauge the level of user acceptance of the system, we have identified a number of quantitative measures to be applied to this task including, among others, the number and ratio of articles accepted/rejected by a user, the average length of time spent searching for articles to read in a newsgroup, the average length of time spent reading a given article, and the number of different newsgroups that articles are sought from[1]. On the assumption that it should become easier for users to find relevant articles and that they should have more time available for reading articles from other newsgroups, we hope to see an increase in the number of newsgroups that articles are selected from, as well as an increase in the number of articles the user actually reads.

[1]Several of these or related measures were employed by Maltz in his evaluation of a collaborative Usenet filtering application (Maltz 1994).

References

Baclace, P.E. 1993. Personal information intake filtering. In *Proceedings of the Bellcore Information Filtering Workshop*.

Belkin, N.J.; and Croft, W.B. 1992. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38.

Ferguson, I.A. 1995. Integrating models and behaviors in autonomous agents: Some lessons learned on action control. In *Proceedings AAAI Spring Symposium on Lessons Learned from Implemented Software Architectures for Physical Agents*, pp. 78-91, Palo Alto, CA. NRC 38340.

Ferguson, I.A.; and Davlouros, J.D. 1995. PeopleFinder: A multimodal multimedia communications tool for interconnecting office staff. In *Proceedings IJCAI Workshop on Intelligent Multimedia Information Retrieval*. pp. 175-182. NRC 38380.

Fischer, G.; and Stevens, C. 1991. Information access in complex, poorly structured information spaces. In *Proceedings of Human Factors in Computing Systems*, CHI'91.

Foltz, P.W.; and Dumais, T. 1992. Personalized information delivery: An analysis of information filtering methods. *Communications of the ACM*, 35(12):51-60.

Hammond, K.; Burke, R.; and Schmitt, K. 1994. A case-based approach to knowledge navigation. In *Proceedings of AAAI Workshop on Indexing and Reuse in Multimedia Systems*, pp. 46-57.

Jennings, A.; and Higuchi, H, 1992. A personal news service based on a user model neural network. *IEICE Transactions on Information Systems*, 75(2):198-209.

Huberman, B.A.; and Hogg, T. 1993. The emergence of computational ecologies. In L. Nadel and D. Stein (eds.), *Lectures in Complex Systems*, pp. 163–184 Addison-Wesley.

Karakoulas, G.J. 1995. Q-Learning for Learning-to-Control Agents. ERB-1044, Institute for Information Technology, National Research Council, Ottawa, ON, Canada.

Lashkari, Y.; Metral, M.; and Maes, P. 1994. Collaborative interface agents. In *Proceedings Conference of the American Association for Artificial Intelligence*, Seattle, WA, pp. 444–449.

Maltz, D.A. 1994. Distributing Information for Collaborative Filtering on Usenet Net News. Master's Thesis, Massachusetts Institute of Technology, Cambridge, MA.

Miller, M.S.; and Drexler, K.E. 1988. Markets and computation: Agoric open systems. In B.A. Huberman (ed.), *The Ecology of Computation*, Elsevier Science Publishers B.V. (North Holland), pp. 133–176.

Mitchell, T.; Caruana, R.; Freitag, D.; McDermott, J.; and Zabowski, D. 1994. Experience with a personal learning assistant. *Communications of the ACM*, 37(2):80-91.

Sheth, B; and Maes, P. 1993. Evolving agents for personalized information filtering. In *Proceedings Ninth Conference on Artificial Intelligence for Applications*, pp. 345-352. IEEE Computer Society.

Wellman, M.P. 1994. A computational market model for distributed configuration design. In *Proceedings Conference of the American Association for Artificial Intelligence*, Seattle, WA, pp. 401–407.

Yan, T.W.; and Garcia-Molina, H. 1994. SIFT - A tool for wide-area information dissemination. Technical Report, Department of Computer Science, Stanford University.