에이전트 모형 및 메타 휴리스틱을 이용한 인터넷 상점 사용자 편의 기능 평가

안 형 준
hjahn@hongik.ac.kr
College of Business Administration, Hong Ik University,
72-1, Sangsu-Dong, Mapo-Gu,
121-791, Seoul, Korea

ABSTRACT

많은 인터넷 상점들이 다양한 사용자 편의 기능을 제공하고 있다. 이 논문에서는 그러한 편의 기능을 평가하기 위한 새로운 분석 기법을 제시하다. 제시된 기법은 에이전트 기반 모형과 메타 휴리스틱인 evolution strategy를 이용하여 고객들의 행태를 모형화하고 최적화한 후 여러 가지 다양한 사용자 편의 기능을 평가해 본다. 이때 평가의 초점은 개인화된 추천 페이지에 두고 이를 인기상품 추천, 카테고리 정렬 등 여러 가지 다른 기능들과 비교해 본다. 이를 위해 가상 인터넷 상점이 구현 되며 데이터셋을 활용하여 시뮬레이션 실험 및 분석이 수행된다. 분석 결과 개인화된 서비스 기능들이 항상 고객들의 쇼핑 효율 및 효과를 항상 높여주지는 않는 것으로 나타났다.

1. INTRODUCTION

With e-commerce maturing across industries and countries, being able to sell products or services online does not give much competitive advantage to businesses anymore. Any companies are now able to quickly build an e-commerce infrastructure and launch services with readily available technology solutions and development experts at low costs. One of the efforts that many companies have made in order to survive in this competitive environment is to develop various customer aid functions that can provide more satisfactory shopping experience to customers over rival Internet stores.

There are many such customer aid functions that range from simply assorting products in various ways to providing intelligent personalized services. For example, many stores like Amazon.com recommend products in several different ways according to individual customers' activity or transaction history. Many stores also have personalized pages that provide a list of recommended products, information, or advertisements that are prepared according to the estimated preference of each customer based on customer activities or transaction records. There are also simpler and non-personalized recommendations provided in many stores that show lists of best selling products, top-rated products, most-clicked products, etc. Simple customer aid functions are relatively easy to build, but sophisticated methods such as personalization often requires applying advanced techniques based on statistics or artificial intelligence.

The main goal of this study is to develop a novel method of analyzing and evaluating the usefulness of such customer aid functions by developing an agent-based model of customer behavior and applying evolution strategies, an optimization technique, to the behavior model. There are several advantages in this approach. First, the approach enables experimental analysis of the usefulness of customer aid functions without implementing customer aid functions or conducting expensive empirical studies. Second, with agent-based modeling, individual customer's behavior is modeled naturally with parameters that can represent different behavioral characteristics of individuals that determine the way individual customers shop online. With evolution strategies, we can investigate how useful a customer aid function is by optimizing the agent-based model for each customer and observing the simulated behavior of each customer. The approach is applied to a virtual online DVD rental store constructed with a publicly available dataset to evaluate some customer aid functions, especially focusing on personalized pages in comparison with others.

The paper is organized as follows. Section 2 gives a brief overview of related literature. Section 3 presents the agent-based model of customer behavior and the application of evolution strategies to the model. Section 4 gives the results of the experiments. Section 5 discusses the results and section 6 concludes.

2. OVERVIEW OF RELATED LITERATURE

2.1. Customer behavior models

There have been many studies that empirically test the factors that affect the effectiveness of personalization or personalized recommendation using experiments with human participants on fully or partly working Internet stores (Klopping & McKinney, 2004; Komiak & Benbasat, 2006; Stern, Royne, Stafford, & Bienstock, 2008). However, not much attention has been paid to developing and using customer behavior models for simulation-based analysis of a broad range of customer aid functions. Some of the studies on online customer behavior models show how agent-based modeling can be useful for analyzing various aspects of markets, stores, or individual customers (Schwaiger & Stahmer, 2003; Twomey & Cadman, 2002). In literature, agents are usually defined as autonomous software entities that exhibit intelligent behavior which resembles that of customers and business organizations (H. Ahn, Lee, & Park, 2003; Barbuceanu, Teigen, & Fox, 1997; Chen, et al., 2004; Nissen, 2001; Wooldridge & Jennings, 1995). Because agent-based models can capture the characteristics of independent entities naturally and realistically, it is increasingly being regarded as a powerful tool for analyzing complex dynamics of customers and businesses in various market environments (Chaturvedi, Mehta, Dolk, & Ayer, 2005; F & Genoese, 2006; Leombruni, Richiardi, Saam, & Sonnessa, 2006; Marks, 2006; Said, Bouron, & Drogoul, 2002; Schwaiger & Stahmer, 2003; Sueyoshi & Tadiparthi, 2008; Twomey & Cadman, 2002). There are other types of studies that apply traditional techniques of operations research to modeling and analyzing customer behavior. For example, the study by Zohar et al. analyzes the behavior of customers for telecommunication services using a queuing model (Zohar, Mandelbaum, & Shimkin, 2002). Moe and Fader's study uses a probabilistic model for modeling and predicting users' conversion behavior at Internet stores (W. W. Moe & Fader, 2004). The study by Lewis (Lewis, 2004) evaluates the effects of a loyalty program of an online merchant using dynamic programming formulation.

The above studies provide significant insights into this study. Especially, this study takes advantage of agent-based models as found in the literature for building the customer behavior model. Also, the work by Lewis (Lewis, 2004) has some similarity to what this study aims to achieve, but the focus of the studies are different in that this study focuses on the usefulness of customer aid functions for browsing and shopping, while Lewis' work evaluates how useful loyalty programs are in retaining customers over long periods.

2.2. Personalization at Internet stores and the evaluation of personalization methods

Among many customer aid functions, this study targets personalized pages for applying and illustrating the use of the customer behavior model. Personalization has received considerable attention with the proliferation of online services. Personalization usually refers to the presentation of products, services, or information to customers tailored according to individual user's preference that is estimated by various methods usually using data collected through online transactions. One of the most popular applications of personalization at online stores is personalized recommendation of products. For example, online stores such as Amazon.com recommend products to customers whenever customers click on some product within the store. The recommendation is based on the similarity between the recommended products and other products that have been previously clicked or purchased by the user, or the similarity between products that are often purchased together by customers. This mechanism is called collaborative filtering (CF) and is one of the most widely used methods of recommendation (H. J. Ahn, 2006; Burke, 2002; Jonathan L. Herlocker, Konstan, & Riedl, 2000; Jonathan L. Herlocker, Konstan, Terveen, & Riedl, 2004; Schein, Popescul, & Ungar, 2005). Many stores also provide a personal page, often called 'My Page', that contains a list of products or information that are relevant to each user. The list is again created using various methods including CF that predict a user's preference based on the user's past browsing history or transactions.

Personalized product recommendation methods are often classified into two types: content-based filtering (CBF) and CF. CBF uses content information such as keywords of products purchased by a customer to build customer profiles, and recommend products that match the profiles to the customers (Adomavicius & Tuzhilin, 2005; Kim, Yum, Song, & Kim, 2005; Li, Lu, & Xuefeng, 2005; Melville, Mooney, & Nagarajan, 2002; Mirzadeh, Ricci, & Bansal, 2005). On the other hand, as introduced above, CF uses similarity between products or customers using non-content data such customers' ratings or purchasing records (H. Ahn, 2008; H. J. Ahn, 2006; Linden, Smith, & York, 2003; Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994). CF usually provides superior performance over CBF, although the ratings data needed for CF might not always be available for use.

There have been lots of studies that have developed and evaluated many different methods of personalized recommendation. Most studies use public datasets for evaluating the proposed methods (Burke, 2002; Jonathan L. Herlocker, et al., 2004). As discussed in Section 2.1, there have been also many empirical studies that use experiments involving human participants that assess the impact of different factors on the effectiveness or adoption of personalized recommendation.

Compared with many studies that have developed and/or evaluated various methods of personalization, this study is different in a couple of aspects. First, many studies evaluate personalization functions based on how accurately customer preferences are estimated with the personalization methods. However, this type of evaluation cannot show whether the personalization methods are actually useful for helping customers shop more effectively and efficiently. This is because no matter how accurate the estimations could be, the personalized services might still not be used by customers because customers can find other methods or routes of shopping more useful, and choose not to use such services in actual shopping. Second, because empirical experiments using fully or semi-fully operating online stores with human participants are usually very expensive and often infeasible, the approach of this study can be a practical alternative for more realistic analysis.

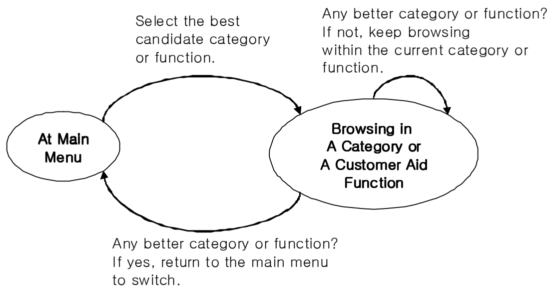
3. AGENT-BASED MODELING OF CUSTOMER BEHAVIOR AND APPLICATION OF EVOLUTION STRATEGIES

3.1. Overview of the models

Now this section presents the model of customer behavior that is used for capturing and representing the characteristics of customers' online browsing activities. The model is based on several broad assumptions:

- First, according to customer decision making theories (Kotler & Armstrong, 1991; Simon, 1959), customers collect information on products first before making purchasing decisions. It is assumed that the amount of information collected and the time taken during the information gathering phase of the purchasing process is directly related to shopping efficiency and effectiveness.
- Second, for simplicity of the analysis, it is assumed that an internet store consists of a main menu, product categories under the main menu, and customer aid functions. Customer aid functions can be regarded as special categories that list products chosen by certain criteria in a certain order to aid customers' shopping. Products may belong to multiple categories and functions.
- Third, it is assumed that customers are rational and try to improve their decision process gradually. In other words, customers are assumed to try to maximize the amount of information they acquire in a given time, or minimize the time taken to collect a given amount of information needed. It is also assumed that the amount of information sought or the time available for shopping may vary for different customers or different shopping purposes.
- Fourth, customers are assumed to have their own estimation on which categories or functions are more useful for their shopping goals, and hence, have their own plan or strategy for browsing in the store. This estimation and plan are updated as customers browse through and collect information on more products.

Figure 1 shows the illustration of the overall customer behavior model that is based on the above assumptions. As can be seen, customers begin shopping at the main menu of the store with their own estimations of which categories or functions are likely to yield better results in terms of collecting information for decision making. And then customers select a category or a function based on the estimations and keep browsing within the category, clicking on products, collecting information, and updating their estimates, until the customers find and switch to a better category or a function based on the updated estimation.



[Figure 1] Overview of the customer behavior model

3.2. Formal modeling of agents and Internet store

Now, based on the overall ideas of customer behavior just presented, formal models of users and Internet stores are defined as follows. The models were classified into static and dynamic ones, where the static models define attributes or features of users and Internet stores, and the dynamic model defines how users behave in a store according to the changes in the variables of the static models.

A. Static models of customer behavior and the Internet store

A user vector $\boldsymbol{u}$ for each customer is defined as $\boldsymbol{u} = <p_{c1}, p_{c2}, \ldots, p_{cN}, p_{f1}, p_{f2}, \ldots, p_{fM}, g, w>$ where p_{cj} and p_{fk} are the user's estimates of the success probabilities of category c_j and customer-aid function f_k respectively where the estimates represent how successful the user will be in collecting information browsing in a category or a function. That is, for example, if the estimated probability is 0.7 for a certain category or a function, the user expects that he or she will find satisfactory products among 70 percent of the remaining items to browse in the category or function. N is the number of categories in the Internet store and M is the number of shopping-aid functions in the store. g and w are the *gap* and the *window* parameters that affect the user's behavior of updating estimates and switching to other categories of functions. The gap and the window parameters and how they are used will be explained later when the dynamic model is presented.

Similarly, an Internet store is defined as $\boldsymbol{S}$ = <$\boldsymbol{A}$, $\boldsymbol{Cat}$, $\boldsymbol{Fun}$>, where $\boldsymbol{A} = \{a_i \mid a_i$ is a product sold in $\boldsymbol{S}\}$ is the set of products sold in the store, and $\boldsymbol{Cat}$ is the relationship between products and categories. Because we assume that a product may belong to multiple categories, $\boldsymbol{Cat}$ is defined as a relation on $\boldsymbol{S} \times \boldsymbol{C}$, where $\boldsymbol{C} = \{c_1, c_2, ..., c_N \mid c_j$ is a category in $\boldsymbol{S}\}$. Similarly, $\boldsymbol{Fun}$ represents the relationship between customer-aid functions and products for each user, that is, $\boldsymbol{Fun}$ is defined as a relation on $\boldsymbol{A} \times \boldsymbol{F} \times \boldsymbol{U}$ where $\boldsymbol{F} = \{f_1, f_2, ..., f_M \mid f_k$ is a customer-aid function in $\boldsymbol{S}\}$, and $\boldsymbol{U}$ is the set of user vectors. Using $\boldsymbol{Fun}$, we can identify which products are presented to a given user by a specific customer aid function such as a personalized page.

B. The dynamic model of customer behavior

Figure 2 shows the dynamic model of customer behavior in a pseudo-code format, which is in fact a formal refinement of Figure 1. As can be seen, a user continues browsing in the store until the user's shopping goal is satisfied. If the user is located at the main menu level, the user will select and go to the best category or function based on the current values of $p_{c1}, p_{c2}, ..., p_{cN}$ and $p_{f1}, p_{f2}, ..., p_{fM}$ (see lines 3 ~ 5 of the figure).

```
1.  Function shop() {
2.    Repeat while (NOT isShoppingGoalSatisfied()) {
3.      If (locationLevel == MAIN_MENU) {
4.         currentCategoryFunction = selectNextBestCategoryOrFunction();
5.         locationLevel = CATEGORY_OR_FUNCTION;
6.      }
7.      Else If (locationLevel == CATEGORY_OR_FUNCTION) {
8.         bestEstimate = getEstimateOfBestCategoryOrFunction();
9.         If (bestEstimate > (getEstimate(currentCategoryFunction) + g)) {
10.          locationLevel = MAIN_MENU;
11.        } Else {
12.          selectNextProduct (currentCategoryFunction);
13.          updateEstimate(currentCategoryFunction, w);
14.        }
15.     }
16.     increaseClickCount();
17.   }
18. }

19. Function selectNextProduct (currentCategoryFunction) {
20.   If (isSatisfactory (getNextProduct())
21.      addCategoryHistorySuccess(currentCategoryFunction);
22.   Else
23.      addCategoryHistoryFailure(currentCategoryFunction);
24. }
```

[Figure 2] The dynamic model of customer behavior represented in a pseudo-code format

If the user is located at a category or a function, user checks if switching to a different category or a function is needed based on the estimates of success probabilities for all the categories and functions (lines 7~10). The model assumes that the user switches to a category/function only when there is a category/function whose estimates are bigger than that of the current one plus g, the gap parameter. This is because switching can be costly in terms of clicks and time if switching occurs too frequently even when there is only a small difference in the estimates between categories or functions, and hence, it would be reasonable to switch only when the gap is larger than a certain threshold.

If switching is not needed, then the user will continue to browse in the same category/function by clicking on the next product (lines 11 ~ 13 and lines 19~24). Here, the user determines if the product is satisfactory and if it can be regarded as one of the possible options for purchase. After this, the user updates the estimate for the current category/function (line 13), which is again used in the next iteration of the loop to see if switching is required. Finally, the click count is increased at the end of the loop (line 16).

Note that when updating the estimate of the current category/function, w, the window parameter, is used (line 13). The window parameter represents the length of past browsing history considered when calculating the estimates. That is, it is assumed that only a certain portion of recent history for each category/function is used for updating the estimates. Figure 3 shows how the estimates are updated more formally.

```
Function updateEstimate(currentCategoryFunction, w) {
  initEstimate = currentCategoryFunction.intialEstimate;
  hLength = currentCategoryFunction.historyLength;
  If (hLength >= w) {
       newEstimate = successCountForLastWTries / w;
  } Else {
       plainEstimate = totalSuccess / hLength;
       newEstimate = (initEstimate * (w - hLength) + plainEstimate) / w
  }
  currentCategoryFunction.estimate = newEstimate;
}
```

[Figure 3] Updating an estimate of category/function

As for the shopping goal, in line with the assumptions presented earlier in this section, this study uses two different goals for all experiments, and accordingly, two different measures respectively: (1) the first type of goal is to collect as much information as possible (or finding as many satisfactory products as possible) within a given number of clicks (denoted as *MaxClick* constraints), and (2) the second type of goal is to minimize the number of clicks taken to collect a given amount of successful information, or to find a given number of satisfactory products (denoted as *MaxSuccess* constraints). The former corresponds to the effectiveness of a given customer's browsing, while the latter the efficiency of shopping. When the first goal is used, it will be measured how much of the information collected is satisfactory during MaxClick clicks. For the second goal, it will be counted how many clicks were used for collecting information on MaxSuccess number of products. The former measure will be denoted as *SuccessRate*, and the later as *ClicksTaken*. Some of the experiments will be performed varying the values of MaxClick and MaxSuccess to see the effect of changing these values on the usefulness of customer aid functions.

3.3. Optimization of customer behavior

Based on the assumption of rationality of customers, this study tries to optimize each customer's behavior using evolution strategy (ES), a heuristic optimization method (Back, Hoffmeister, & Schwefel, 1991; Beyer & Schwefel, 2002; Fogel, Inc, & La Jolla, 1994). ES is one of the optimization techniques that belong to the broad field of evolutionary computation. The basic idea of ES is based on Darwinian natural selection which explains the evolution of species as a process where individuals compete among a population, winning individuals survive, and the survivors evolve through reproducing themselves that involves mutation and crossover (Back, et al., 1991; Beyer & Schwefel, 2002; Fogel, et al., 1994). There are several advantages of using ES for this research. First, ES can naturally model the complex nature of customers with many variables as individuals in the population and apply the rules of evolution to the population. Second, the problem of this research needs to optimize many variables in a large search space simultaneously, for which ES is good as well. Third, the research problem does not require strict optimal values for all users, but near-optimal customer behaviors for some users can be acceptable for the purpose of comparing the effectiveness of Internet store functionalities. Fourth, compared to genetic algorithm (GA) that also belongs to the evolutionary computation field, ES can model individual traits as real values as well, which is required for representing the variables of the customer behavior model, while GA encodes genes of individuals discretely with only 0s and 1s.

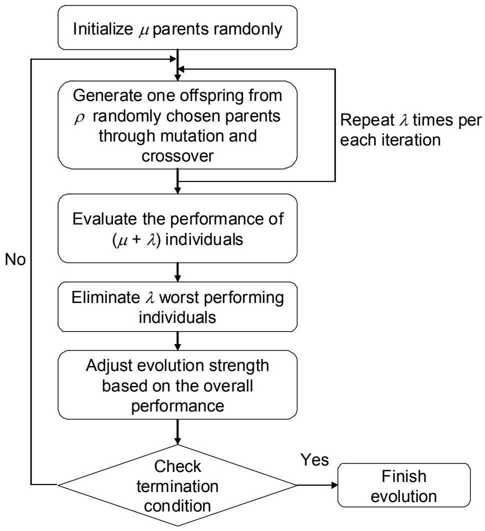
Readers may argue that customers do not always act rationally or optimally when shopping. However, the rationality assumption has two significant implications. First, although customers may not always behave optimally when shopping, they may actually evolve their pattern of behavior over a long time to improve their purchasing process in a given store, which is very similar to the optimization mechanism of evolution strategies. Second, because the goal of this study is to evaluate the effectiveness of different customer-aid functions of Internet stores, it is meaningful to compare the functions based on optimal behavior of customers. That is, a specific function that shows better performance than another for optimized customer behaviors can generally be accepted as a superior one, whereas a function that shows better performance for non-optimized or inefficient customer behavior may not be necessarily regarded as being superior. Third, if a function can indeed lead customers to better shopping experience, then Internet stores can actively guide customers who can benefit from the function to use the function with various links or help pages, which can in fact have a similar effect to evolving customer skills.

Because two measures are used in this study, there are two different optimizations goals, one for each measure. The first goal is to maximize the SuccessRate measure subject to the customer behavior models with given parameters within a fixed number of clicks. In the same way, the second goal is to minimize the ClicksTaken measure in finding a fixed number of satisfactory products. Because an Internet store and the customer behavior models are assumed to be fixed, the optimization problem translates to finding the best values of the parameters. Equally in the two optimization problems, the following parameters are optimized:

- Initial estimates of probability of categories and functions: $p_{c1}, p_{c2}, \ldots, p_{cN}, p_{f1}, p_{f2}, \ldots, p_{fM}$
- Gap and window parameters: g and w

That is, the optimization problems aim to find optimal values of all the dimensions in user vector **u** for each user.

Figure 4 shows the details of the actual ES used for this research. Note that $(\mu/\rho+\lambda)$-ES is used in this research. That is, there are μ parents, ρ parents are used for creating one offspring, and in total λ individuals are regenerated from the parents at each iteration. In order to maintain a fixed population size, among $\mu+\lambda$ individuals, λ worst performers are discarded at the end of each iteration. As is common when using ES, the strength of evolution is adjusted at the end of each iteration so that, if the performance is improving, the strength can be reduced so that more search can be focused on the current area of best performance, and on the other hand, if the performance is stalling, the strength can be increased to search wider areas and possibly escape from a local maximum. The evolution finishes if no improvement is observed for a predefined number of iterations. The detail of adjusting the evolution strength is presented at the appendix.



[Figure 4] Optimizing customer behavior using evolution strategies

4. EXPERIMENTS

4.1. Overview

In order to illustrate how the customer model can be used, an artificial Internet DVD rental store was created where users can browse movies and collect information on movies. The store is based on data on movies and movie ratings provided by Netflix (Netflix, 2007) and Internet Movie Database (IMDB) (IMDB, 2008). Note that the store is an abstract one used only for the purpose of simulating users' shopping and evaluating the performance.

Users are assumed to exhibit the optimal behavior found by ES. The main focus of the analysis will be on personalized recommendation pages (*MyPage* in Table 1) that are provided by many Internet stores such as Amazon.com. It will be compared with other general Internet store functions such as a page with top products (*TopN* in Table 1) and sorting products in each category by average rating (*Sorting* in Table 1). Also included in the comparison are when none of the functions is used (*None* in Table 1) and all of the functions are used (*All* in Table 1). Readers should note that the Sorting function is treated differently from MyPage or TopN in the simulations because it does not require an additional page for the function, but it only changes the order of products in the categories. However, for readability and simplicity, it will be denoted as a function as well.

[Table 1] Summary of the settings for experiments

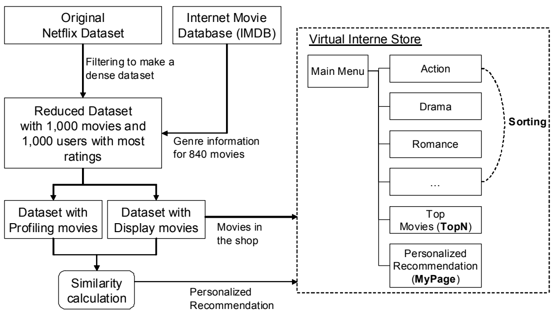
Experiment Options	Description
A personalized recommendation page (denoted as *MyPage*)	For each user, a page is provided where a list of products are recommended using collaborative filtering. The recommended movies are sorted by the estimated rating by the user.
A page listing most popular products (dented as *TopN*)	This page provides a list of most popular products sorted by average rating of the products by all users. The same page with the same products is presented to all users.
Sorting each category by average product ratings (denoted as *Sorting*)	With this enabled, products are sorted by average rating in each category rather than being presented in an arbitrary order.
All of the above are used (dented as *All*)	With this option, it is assumed that TopN, MyPage, and Sorting are all used together in the store.
None of the above is used (dented as *None*)	With this option, none of the above is used. Only basic browsing through unsorted categories is assumed.

4.2. Dataset

The primary dataset used for the research is the Netflix contest dataset which provides a huge number of user ratings on movies. There are about 17,000 movies and 480,000 users in the dataset and each rating is in a scale of 1 to 5. However, the dataset is sparse and only provides about 3.6 percent of all possible ratings for the entire (user × movie) matrix. Because we need to be able to evaluate how successful a user's browsing is in collecting information on movies in the virtual store when a user clicks on each movie, we need a denser dataset where most user ratings are available. At the same time, the dataset needs to be still large enough so that we can calculate significant similarity between products which is needed for building the personalized recommendation function based on the widely-used collaborative filtering recommendation method (H. Ahn, 2008; Burke, 2002; Jonathan L. Herlocker, et al., 2004) (see the Appendix for the details of the recommendation method used in this research). Therefore, 1,000 movies and 1,000 users with most ratings were selected from the dataset.

The new subset now has 817,585 ratings, that is, about 82 percent of all possible ratings are present in the selected set. Among 1,000 movies, 500 were used only for the purpose of profiling users and calculating similarity for personalized recommendation (*profiling movies*), while the other 500 were assumed to be displayed in the Internet store for customers' shopping (which can be denoted as *display movies*). In other words, 500 profiling movies are used as reference movies based on which the ratings of each display movie by each user is estimated for personalized recommendation.

Another type of data needed for the experiments is the genre information for each display movie. This is because the experiment assumes that movies are categorized by genres in the Internet store, that is, each genre is regarded as a product category. Since the Netflix dataset does not provide genre information, the IMDB database was used for finding genres for each of the 1,000 movies. In the IMDB database, there are 24 genres, and each movie can belong to multiple genres. Through matching of movie titles between the two data sources by a computer program and manual processing, genre information for 840 movies out of the 1,000 was collected. Note that the display movies were selected only among these 840 movies, because profiling movies does not need the genre information.



[Figure 5] Summary of the data processing

With the final dataset created through the above process, we have the data as shown in Figure 5 ready for experiments. Figure 5 also shows the summary of the whole process of data preparation. As can be seen, the dataset now can be used as a virtual Internet movie store with 24 movie categories, a personalized recommendation page for each user, and a page listing most popular movies.

4.3. Experiments

Four experiments were performed as summarized in Table 2. All the experiments are performed twice using both measures, SuccessRate and ClicksTaken.

[Table 2] Summary of the four experiments

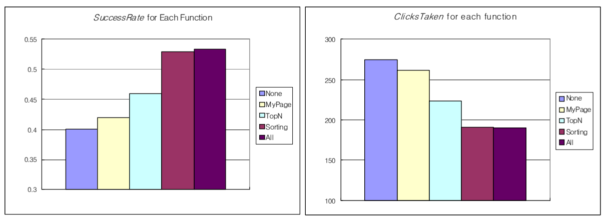
No	Description	Independent variable
1	Basic comparison of the performance of using different web site functions	Use of different functions
2	Comparison of different web site functions by changing shopping goals, MaxClick and MaxSuccess	Shopping goals: MaxClick and MaxSuccess
3	Performance change of the MyPage function by changing the number of recommended movies	Number of movies recommended to users at the MyPage function
4	Relative performance of the MyPage function compared to category sorting as the shopping target broadens (or narrows)	The scope of shopping target (or the specificity of shopping intention)

The first experiment compares the performance of the experiment options: MyPage, TopN, and Sorting. It compares how using or not using each function affects the performance of user's browsing in the store. The second experiment sees how the performance of the functions changes as the two shopping constraints, MaxClick and MaxSuccess, varies. The third experiment tests how the performance is affected by changing the number of movies recommended by the MyPage function. The fourth experiment sees the relative improvement of the MyPage function against Sorting as the specificity of shopping goals changes. Unless otherwise stated, the following parameters were used as defaults in all the experiments:

- MaxClick = 300 when using the SuccessRate measure,
- MaxSuccess = 50 when using the ClicksTaken measure (a shopping also ends when the click count reaches 800),
- The number of movies listed at the TopN page = 100,
- The number of movies listed at MyPage = 50,
- All the experiments are repeated for randomly chosen 50 users for statistical reliability.

A. Basic comparison of the Internet store functions

Figure 6 shows the comparison of the performance when each of the experiment options is taken. The graphs show an interesting result where we can see that users exhibit much better performance using Sorting only compared with MyPage or TopN. The MyPage and TopN functions also provide benefits compared to the baseline option None although the advantage is smaller than that of using Sorting. This result implies that the advantage of providing a page of personalized recommendation may not be as effective as people might believe generally, compared with simple sorting of products in each category. Combining all the functions together yields a slight improvement over the Sorting option.



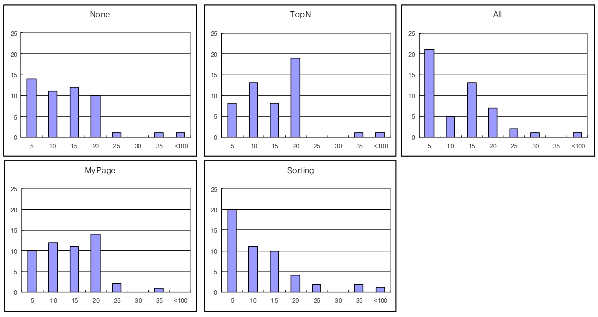
[Figure 6] Comparison of the functions

Although the average performance of Sorting is superior to MyPage, it is not clear whether it is true for all users. Table 3 shows an answer to this. It presents the result of optimizing p_{cj} and p_{fk} for categories and functions as priorities among MyPage, TopN, and general categories. That is, the Table shows which of the MyPage, TopN, or general categories had higher values of preference for users and were browsed first as the result of optimizing the respective parameters. We can see that when using the MyPage option under the MaxSuccess shopping constraint, MyPage was first browsed by 27 of the 50 users showing higher priority for the function. With TopN option, 39 users browsed the function first. However, when Sorting was used together with MyPage and TopN in the All option experiment, category browsing was given a much higher priority and 34 users browsed some categories first before exploring MyPage or TopN. A similar result can be seen when the shopping constraint by MaxClick is used as well. This overall result is consistent with the comparison in Figure 6.

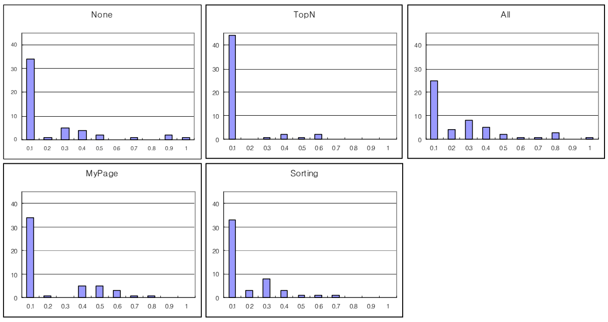
[Table 3] Priorities of functions when each experiment option was used

Priorities / Exp. Option	Shopping Goal by MaxSuccess		
	MyPage first	TopN first	Category first
MyPage	27	-	23
TopN	-	39	11
All (with Sorting)	7	9	34
Priorities / Exp. Option	Shopping Goal by MaxClick		
	MyPage first	TopN first	Category first
MyPage	25	-	25
TopN	-	21	29
All (with Sorting)	15	12	23

Next, Figure 7 shows the distribution of the two parameters in the user vector, w and g, at the end of optimization runs with ES among 50 test users. It shows that the window parameter is quite evenly distributed between 0 ~ 20, while many of the gap parameters are located near low values. From these, we can say that most users take into account last 0 ~ 20 results when estimating the success ratio of a product category, and more users switch to other categories frequently even when the gap of estimates is small between categories and functions.



(A) Distribution of the optimized window parameter (w)

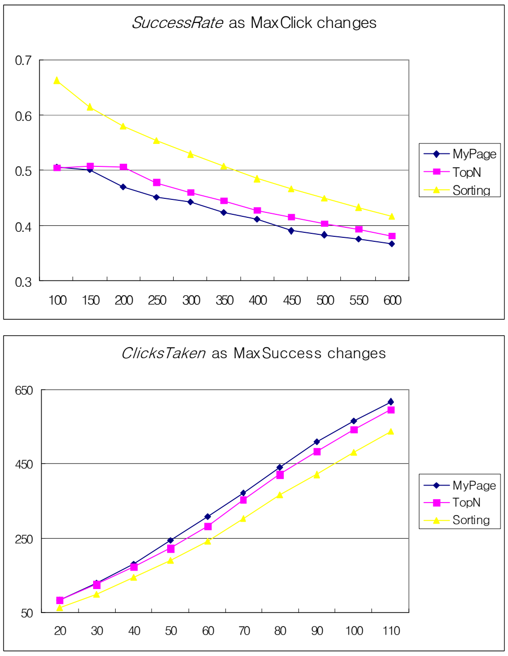


(B) Distribution of the optimized gap parameter (g)

[Figure 7] Distribution of the parameters w and g after running ES for users

B. Performance of the functions as MaxClick or MaxSuccess changes

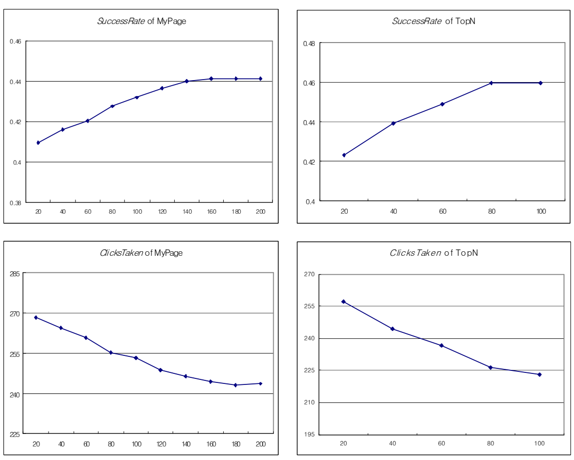
The aim of the next experiment is to see the effect of changing the levels of the MaxClick and MaxSuccess on the performance of the three tested functions, MyPage, TopN, and Sorting. The results are summarized in Figure 8, where the upper chart shows the change in the SuccessRate measure as MaxClick changes, while the lower shows the change of ClicksTaken as MaxSuccess changes. In both of the results, we can observe that the dominance of Sorting over the other two functions remain the same at all the tested levels. Also, TopN shows better or at least equal performance compared with MyPage. One notable trend is that the gap between Sorting and the other two is bigger when the MaxClick level is low or the MaxSuccess level is high. In other words, the advantage of having sorted categories is even bigger when users have to collect information in a smaller timeframe (= small MaxClick), or when users have to find a larger amount of information on products (=large MaxSuccess). Conversely, the gap becomes smaller when MaxClick is large or MaxSuccess is small.



[Figure 8] Changes in the performance of each function at different levels of shopping goals

C. Effect of the number of movies listed at MyPage and TopN

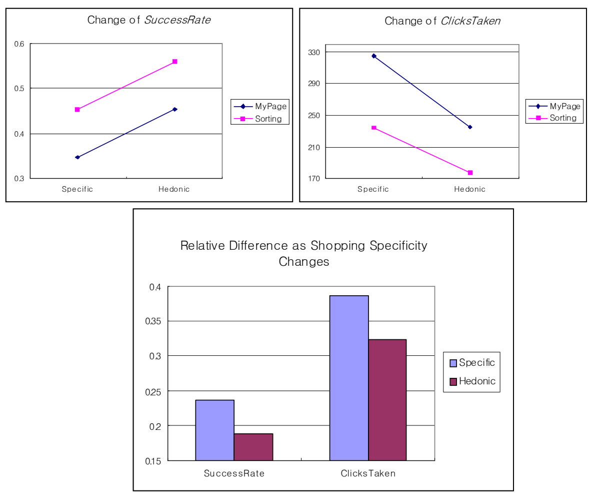
Now, the next experiment sees how the number of movies listed at MyPage or TopN affects the performance of the respective function. In Figure 9, the left two graphs show the performance changes of MyPage and the right two shows those of TopN. As can be seen, all the performance measures for the two functions appear to reach their maximum or minimum at some points. For MyPage, listing more than about 160 movies does bring much improvement; for Top N, similarly, listing more than 80 moves does not give much more advantage. We can note that for different functions, we may have to list different number of products in order to gain maximum benefits of the functions.



[Figure 9] Change of the performance of using MyPage or TopN as the number of movies listed at the functions change. X-axis represents the number of listed movies

D. The effect of the scope of shopping

In contrast to general expectations, MyPage showed overall lower performance than Sorting in the first experiment. However, we have to consider that the MyPage function lists many different types of products that belong to different categories, which may be the reason why it is less effective than Sorting when the shopping goal is to find products that belong to only certain categories. Therefore, in this experiment, the scope of shopping goal is manipulated such that two groups of shoppers are examined, one group with a very specific shopping goal, and the other with less-specific, or hedonic browsing purpose (Wendy Moe, 2003; WW Moe & Fader, 2001). For the specific group, the given shopping goal was to find information on movies that belong to C_s = {Drama, Musical}; for the hedonic-group, the scope was C_h = {Drama, Musical, Romance, Action, Family, Comedy}. Figure 10 shows the results of the experiment. As can be seen, at both specificity levels, Sorting still outperforms MyPage (see the top two graphs of Figure 10). However, the relative difference is smaller for both of the measures when shopping specificity is low (see the bottom graph of Figure 10), which is consistent with the previous expectation that shopping purpose might have impact upon the usefulness of MyPage.



[Figure 11] The performance of MyPage and Sorting as shopping specificity changes. Relative difference is defined as:

$$\frac{|Difference\_in\_performance|}{Performance\_of\_Sorting}$$

5. DISCUSSION

Table 4 summarizes the findings from the four experiments. Overall, the results of the experiments exhibit an interesting finding that the personalized shopping aid function might not always be helpful to all users. It was found that on average users can shop more effectively and efficiently with each product category sorted according to average product ratings than using a personalized page. This result remained consistent at many different levels of shopping constraints (experiment B). However, it was also found that the result is not the same for all users because it was seen that some shoppers are better off choosing MyPage first (Table 4, experiment A). In the third experiment, it was also found that there is a limit in the increase of shopping performance as more movies are listed at the functions. The limits can be different for different functions. The top product page (TopN) reached its maximum performance with fewer movies listed at the page than the personalized page (MyPage). Next, although the personalized page is showing overall lower performance compared with category sorting, it was found that the relative performance of it increases as the shopping specificity of users decreases. In other words, for users with more hedonic shopping goals, the relative performance of personalized pages may be bigger. We can postulate that this is because personalized pages list products of various categories that best match individual users' preferences, which is sufficient to satisfy the needs of hedonic browsers, whereas for specific shopping goals, many products at MyPage might not be relevant.

[Table 4 Summary of the Findings

No.	Key Findings
1	Sorting products in each category resulted in better performance in achieving users' shopping goals than providing a list of movies at TopN or MyPage functions. When all the three functions were provided together, more users visited category pages first.
2	The Sorting function showed superiority at all tested levels of the two shopping constraints, MaxSuccess and MaxClick. The gap between Sorting and TopN or MyPage was bigger when users had less time (smaller MaxClick) or had to find more information (bigger MaxSuccess)
3	Listing more than a certain number of movies at MyPage or TopN does not appear to improve the performance anymore. The performance of TopN reaches its maximum with fewer movies than that of MyPage.
4	The relative performance of using MyPage compared with Sorting increases as the specificity of users' shopping goal decreases.

There are many interesting practical implications for businesses that can be drawn from the results. First, it was found that a simple function such as sorting products can be much more effective than relatively complex and expensive approaches such as personalized pages. Hence, many Internet stores with limited resources can possibly improve customer satisfaction more easily by focusing first on basic and inexpensive functions. Second, because there are limits on the number of products on personalized pages or top product pages that will increase customer satisfaction, Internet stores can find such limits and list only up to that number of products, and still gain the most improvement of user's satisfaction. Third, despite the overall results, we cannot conclude that the personalized pages of Internet stores are not helpful to users because it was also found that some users choose the personalized pages first, and the usefulness might be different for customers with different shopping purposes. Rather, businesses should identify the differences in customers or shopping intentions and provide different methods or paths of browsing so that customers can learn and choose the best ways of shopping for themselves.

There are also limitations of this research as well that require more attention paid to interpreting the results, and that open up issues for further research. First, the main focus of the research was on illustrating the use of modeling and optimizing customer behavior using ES for evaluating the customer-aid functions of Internet stores, and thus, care needs to be taken in generalizing the results to other types of products or Internet stores. Although the same approach can be applied to other domains by modifying or adapting it, the results can be different. Second, the results are based on the assumption that users will learn from experience and exhibit optimal browsing behavior over time, which can be sometimes different from reality. Hence, a further study is needed on investigating how different actual user behavior is from the modeled behavior, and whether different analysis results can be found. Third, the user behavior model is a simplified one that does not take into account all possible factors that may affect the browsing patterns. Therefore, further studies can be performed using models with more factors and diverse classification of users and products. Also, more customer aid functions and other design aspects of online stores can be included in further studies. Finally, empirical investigation of actual data might be considered. Although it is usually very hard and expensive to collect data for empirical investigation, it would be meaningful to verify whether the results using the model and ES are consistent with empirical investigation.

6. CONCLUSION

The contributions of this paper can be summarized as follows: First, to the author's best knowledge, this is the first study that showed how agent-based modeling and evolution strategy can be used together for evaluating customer aid functions at Internet stores. Second, the suggested approach has advantages over previous attempts at evaluating personalization, because it can test the usefulness of customer aid functions at Internet stores in general, not only personalization, in a realistic and practical way compared with many studies that only evaluate how accurate the estimation of customer preferences is for products, or the studies that involve expensive empirical experiments involving human participants and fully or semi-fully operating Internet stores. Third, the application of the approach to the example online DVD store also showed a very interesting result that the sophisticated functions such as personalized pages might not always be helpful for customers in making their purchasing process more effective and efficient.

Although this study showed the model of customer behavior applied to a single product type of a virtual Internet store, the method can be modified and extended for analyzing businesses in other industry sectors as well. However, care should be taken in generalizing the findings of the analysis to other areas because customers or products of different types may exhibit disparate patterns of usage leading to very different conclusions. Although there are some limitations of the study and further research issues remaining, the author believes that the method, if applied properly, can facilitate the development and evaluation of many creative new customer aid functions for a wide variety of Internet stores for improving customer satisfaction.

REFERENCES

Adomavicius, G., & Tuzhilin, A. (2005). Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Transactions on Knowledge & Data Engineering, 17*(6), 734-749.

Ahn, H. (2008). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences, 178*(1), 37-51.

Ahn, H., Lee, H., & Park, S. (2003). A flexible agent system for change adaptation in supply chains. *Expert Systems With Applications, 25*(4), 603-618.

Ahn, H. J. (2006). Utilizing Popularity Characteristics for Product Recommendation. *International Journal of Electronic Commerce, 11*(2), 57-78.

Back, T., Hoffmeister, F., & Schwefel, H. (1991). A survey of evolution strategies. *Proceedings of the Fourth International Conference on Genetic Algorithms, 2*(9).

Barbuceanu, M., Teigen, R., & Fox, M. (1997). Agent Based Design and Simulation of Supply Chain Systems. *Proceedings of the 6th Workshop on Enabling Technologies Infrastructure for Collaborative Enterprises (WET-ICE'97)*, 18?10.

Beyer, H., & Schwefel, H. (2002). Evolution strategies뺸 comprehensive introduction. *Natural Computing, 1*(1), 3-52.

Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction, 12*(4), 331-370.

Chaturvedi, A., Mehta, S., Dolk, D., & Ayer, R. (2005). Agent-based simulation for computational experimentation: Developing an artificial labor market. *European Journal of Operational Research, 166*(3), 694-716.

Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F., & Chakraborty, D. (2004). Intelligent Agents Meet the Semantic Web in Smart Spaces. *IEEE INTERNET COMPUTING*, 69-79.

F, S., & Genoese, M. (2006). Agent-based simulation of the German electricity markets-An analysis of the German spot market prices in the year 2001. *Proceedings of the, 9.*

Fogel, D., Inc, N., & La Jolla, C. (1994). An introduction to simulated evolutionary optimization. *Neural Networks, IEEE Transactions on, 5*(1), 3-14.

Herlocker, J. L., Konstan, J. A., & Riedl, J. (2000, December 2000). *Explaining collaborative filtering recommendations.* Paper presented at the 2000 ACM conference on Computer supported cooperative work, Philadelphia, Pennsylvania, United States.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems, 22*(1), 5-53.

IMDB (2008). Internet Movie DataBase (IMDB) Retrieved September 2008, from http://imdb.com

Kim, Y. S., Yum, B.-J., Song, J., & Kim, S. M. (2005). Development of a recommender system based on navigational and behavioral patterns of customers in e-commerce sites. *Expert Systems with Applications, 28*(2), 381.

Klopping, I. M., & McKinney, E. (Writer) (2004). Extending the Technology Acceptance Model and the Task-Technology Fit Model to Consumer E-Commerce [Article], *Information Technology, Learning & Performance Journal.*

Komiak, S. Y. X., & Benbasat, I. (2006). The Effects of Personalization And Familiarity on Trust and Adoption of Recommendation Agents. *MIS Quarterly, 30*(4), 941-960.

Kotler, P., & Armstrong, G. (1991). *Principles of Marketing* (5 ed.): Prentice Hall International.

Leombruni, R., Richiardi, M., Saam, N., & Sonnessa, M. (2006). A common protocol for agent-based social simulation. *Journal of Artificial Societies and Social Simulation, 9*(1).
Lewis, M. (Writer) (2004). The Influence of Loyalty Programs and Short-Term Promotions on Customer Retention [Article], *Journal of Marketing Research (JMR)*: American Marketing Association.
Li, Y., Lu, L., & Xuefeng, L. (2005). A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce. *Expert Systems with Applications, 28*(1), 67-77.
Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Computing, 7*(1), 76-80.
Marks, R. (2006). Market design using agent-based models. *Tesfatsion & Judd (2006)*, 1339-1380.
Melville, P., Mooney, R. J., & Nagarajan, R. (2002, July 2002). *Content-Boosted Collaborative Filtering for Improved Recommendations.* Paper presented at the Eighteenth National Conference on Artificial Intelligence(AAAI-2002), Edmonton, Canada.
Mirzadeh, N., Ricci, F., & Bansal, M. (2005, March 2005). *Feature Selection Methods for Conversational Recommender Systems.* Paper presented at the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service, Hong Kong, China.
Moe, W. (2003). Buying, Searching, or Browsing: Differentiating Between Online Shoppers Using In-Store Navigational Clickstream. *Journal of Consumer Psychology, 13*(1&2), 29-39.
Moe, W., & Fader, P. (2001). Uncovering Patterns in Cybershopping. *California Management Review, 43*(4), 106-117.
Moe, W. W., & Fader, P. S. (Writer) (2004). Dynamic Conversion Behavior at E-Commerce Sites [Article], *Management Science*: INFORMS: Institute for Operations Research.
Netflix (2007, 28/Nov/2005). Netflix movie dataset Retrieved November 2006, from http://www.netflixprize.com/
Nissen, M. (2001). Agent-Based Supply Chain Integration. *Information Technology and Management, 2*(3), 289-312.
Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994, October 1994). *GroupLens: An Open Architecture for Collaborative Filtering of Netnews.* Paper presented at the ACM 1994 Conference on Computer Supported Cooperative Work, Chapel Hill, North Carolina.
Said, L., Bouron, T., & Drogoul, A. (2002). Agent-based interaction analysis of consumer behavior. *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, 184-190.
Schein, A. I., Popescul, A., & Ungar, L. H. (2005). CROC: A New Evaluation Criterion for Recommender Systems. *Electronic Commerce Research, 5*(1), 51-74.
Schwaiger, A., & Stahmer, B. (2003). SimMarket: Multiagent-Based Customer Simulation and Decision Support for Category Management. *Multiagent System Technologies: First German Conference, MATES 2003, Erfurt, Germany, September 2003: Proceedings*.
Simon, H. (1959). Theories of decision-making in economics and behavioral science. *American Economic Review, 49*(3), 253-283.
Stern, B. B., Royne, M. B., Stafford, T. F., & Bienstock, C. C. (Writer) (2008). Consumer acceptance of online auctions: An extension and revision of the TAM [Article], *Psychology & Marketing*.
Sueyoshi, T., & Tadiparthi, G. (2008). Wholesale Power Price Dynamics Under Transmission Line Limits: A Use of an Agent-Based Intelligent Simulator. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 38*(2), 229-241.
Twomey, P., & Cadman, R. (2002). Agent-based modelling of customer behaviour in the telecoms and media markets. *Info, 4*(1), 56-63.
Wooldridge, M., & Jennings, N. (1995). Agent Theories, Architectures, and Languages: A Survey. *Intelligent Agents, 22*.
Zohar, E., Mandelbaum, A., & Shimkin, N. (2002). Adaptive Behavior of Impatient Customers in Tele-Queues: Theory and Empirical Support. *Management Science, 48*(4), 566.

APPENDIX

A. Adjustment of evolution strengths

Each mutant $\boldsymbol{u'}$ of a user vector $\boldsymbol{u}$ is created with

$$\boldsymbol{u'} = \boldsymbol{u} + \mathbf{z},$$

where $z := (\sigma_1 N_1(0, 1), \ldots, \sigma_l N_l(0, 1))$, l is the number of dimensions in $\boldsymbol{u}$, and $N_i(0, 1)$ is a random sample from the standard normal probability distribution. Each σ_i represents the strength of mutation. When there is improvement after the current iteration, the strength decreases so that the next search can focus more on the nearby areas of the current best solutions. If there is no improvement for a while, (i.e. for a pre-defined number of iterations), the strength increases so that the search can possibly escape from points of local maximum. For real-valued dimensions of the user vector that range between 0 and 1, the following nine levels of strengths were used: {0.04, 0.12, 0.2, 0.28, 0.4, 0.52, 0.8, 1.2, 2}. For integer valued dimensions that range between 1 ~ ∞, the following levels were used: {0.2, 0.6, 1, 1.4, 2, 2.6, 4, 6, 10}. Although many variations were attempted in adjusting mutation strengths in the experiments, not much improvement was observed over the present method.

B. Constructing personalized pages based on product similarities

The personalized pages for the virtual Internet store are constructed with a widely-used recommendation method that utilizes the similarity between products (Burke, 2002; Jonathan L. Herlocker, et al., 2004). The basic idea is to list products that are most likely to be preferred by each user on the personalized pages, where the preference is estimated for each user based on the user's ratings of other products and the similarity between the rated products and other products. That is, products that are similar to the highly rated products by each user are listed in the personalized page of the user ordered by the estimated preference. Formally, the estimation of the preference for product *i* of user *u* is calculated as:

$$p'_{u,i} = \bar{r}_u + \frac{\sum_j \left(r_{u,j} - \bar{r}_u\right) \cdot sim(i,j)}{\sum_j \left|sim(i,j)\right|}$$

, where $\bar{r}_u$ is the average rating of user *u* for all products, $r_{u,j}$ is the rating of product *j* by user *u*, and *sim (i, j)* is the similarity between product *i* and *j*. *sim (i, j)* is often calculated with Pearson's correlation or cosine between the ratings vector of *i* and *j* (Burke, 2002; Jonathan L. Herlocker, et al., 2004). This study used correlation.