Facilitating Effective User Navigation through Website Structure Improvement

Min Chen and Young U. Ryu

Abstract—Designing well-structured websites to facilitate effective user navigation has long been a challenge. A primary reason is that the web developers' understanding of how a website should be structured can be considerably different from that of the users. While various methods have been proposed to relink webpages to improve navigability using user navigation data, the completely reorganized new structure can be highly unpredictable, and the cost of disorienting users after the changes remains unanalyzed. This paper addresses how to improve a website without introducing substantial changes. Specifically, we propose a mathematical programming model to improve the user navigation on a website while minimizing alterations to its current structure. Results from extensive tests conducted on a publicly available real data set indicate that our model not only significantly improves the user navigation with very few changes, but also can be effectively solved. We have also tested the model on large synthetic data sets to demonstrate that it scales up very well. In addition, we define two evaluation metrics and use them to assess the performance of the improved website using the real data set. Evaluation results confirm that the user navigation on the improved structure is indeed greatly enhanced. More interestingly, we find that heavily disoriented users are more likely to benefit from the improved structure than the less disoriented users.

Index Terms—Website design, user navigation, web mining, mathematical programming

1 Introduction

The advent of the Internet has provided an unprecedented platform for people to acquire knowledge and explore information. There are 1.73 billion Internet users worldwide as of September 2009, an increase of 18 percent since 2008 [1]. The fast-growing number of Internet users also presents huge business opportunities to firms. According to Grau [2], the US retail e-commerce sales (excluding travel) totaled $127.7 billion in 2007 and will reach $218.4 billion by 2012. In order to satisfy the increasing demands from online customers, firms are heavily investing in the development and maintenance of their websites. InternetRetailer [3] reports that the overall website operations spending increased in 2007, with one-third of site operators hiking spending by at least 11 percent, compared to that in 2006.

Despite the heavy and increasing investments in website design, it is still revealed, however, that finding desired information in a website is not easy [4] and designing effective websites is not a trivial task [5], [6]. Galletta et al. [7] indicate that online sales lag far behind those of brick-and-mortar stores and at least part of the gap might be explained by a major difficulty users encounter when browsing online stores. Palmer [8] highlights that poor website design has been a key element in a number of high profile site failures. McKinney et al. [9] also find that users having difficulty in locating the targets are very likely to leave a website even if its information is of high quality.

A primary cause of poor website design is that the web developers' understanding of how a website should be structured can be considerably different from those of the users [10], [11]. Such differences result in cases where users cannot easily locate the desired information in a website. This problem is difficult to avoid because when creating a website, web developers may not have a clear understanding of users' preferences and can only organize pages based on their own judgments. However, the measure of website effectiveness should be the satisfaction of the users rather than that of the developers. Thus, Webpages should be organized in a way that generally matches the user's model of how pages should be organized [12].

Previous studies on website has focused on a variety of issues, such as understanding web structures [13], finding relevant pages of a given page [14], mining informative structure of a news website [15], [16], and extracting template from webpages [17]. Our work, on the other hand, is closely related to the literature that examines how to *improve website navigability* through the use of user navigation data. Various works have made an effort to address this question and they can be generally classified into two categories [11]: to facilitate a particular user by dynamically reconstituting pages based on his profile and traversal paths, often referred as *personalization*, and to modify the site structure to ease the navigation for all users, often referred as *transformation*.

In this paper, we are concerned primarily with transformation approaches. The literature considering transformations approaches mainly focuses on developing methods to completely reorganize the link structure of a website. Although there are advocates for website *reorganization*

- *M. Chen is with the School of Management, George Mason University, Fairfax, VA 22030. E-mail: mchen15@gmu.edu.*
- *Y.U. Ryu is with the Naveen Jindal School of Management, The University of Texas at Dallas, SM33, 800 West Campbell Road, Richardson, Texas 75080-3021. E-mail: ryoung@utdallas.edu.*

Manuscript received 18 Nov. 2010; revised 12 Oct. 2011; accepted 23 Oct. 2011; published online 16 Nov. 2011.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2010-11-0606. Digital Object Identifier no. 10.1109/TKDE.2011.238.

approaches, their drawbacks are obvious. First, since a complete reorganization could radically change the location of familiar items, the new website may disorient users [18]. Second, the reorganized website structure is highly unpredictable, and the cost of disorienting users after the changes remains unanalyzed. This is because a website's structure is typically designed by experts and bears business or organizational logic, but this logic may no longer exist in the new structure when the website is completely reorganized. Besides, no prior studies have assessed the usability of a completely reorganized website, leading to doubts on the applicability of the reorganization approaches. Finally, since website reorganization approaches could dramatically change the current structure, they cannot be frequently performed to improve the navigability.

Recognizing the drawbacks of website reorganization approaches, we address the question of how to *improve* the structure of a website rather than *reorganize* it substantially. Specifically, we develop a mathematical programming (MP) model that facilitates user navigation on a website with *minimal* changes to its current structure. Our model is particularly appropriate for *informational websites* whose contents are static and relatively stable over time. Examples of organizations that have informational websites are universities, tourist attractions, hospitals, federal agencies, and sports organizations. Our model, however, may not be appropriate for websites that purely use dynamic pages or have volatile contents. This is because a steady state might never be reached in user access patterns in such websites, so it may not be possible to use the weblog data to improve the site structure [19].

The number of outward links in a page, i.e., the out-degree, is an important factor in modeling web structure. Prior studies typically model it as hard constraints so that pages in the new structure cannot have more links than a specified out-degree threshold, because having too many links in a page can cause information overload to users and is considered undesirable. For instance, Lin [20] uses 6, 8, and 10 as the out-degree threshold in experiments. This modeling approach, however, enforces severe restrictions on the new structure, as it prohibits pages from having more links than a specified threshold, even if adding these links may greatly facilitate user navigation. Our model formulates the out-degree as a cost term in the objective function to penalize pages that have more links than the threshold, so a page's out-degree may exceed the threshold if the cost of adding such links can be justified.

We perform extensive experiments on a data set collected from a real website. The results indicate that our model can significantly improve the site structure with only few changes. Besides, the optimal solutions of the MP model are effectively obtained, suggesting that our model is practical to real-world websites. We also test our model with synthetic data sets that are considerably larger than the real data set and other data sets tested in previous studies addressing website reorganization problem. The solution times are remarkably low for all cases tested, ranging from a fraction of second to up to 34 seconds. Moreover, the solution times are shown to increase reasonably with the size of the website, indicating that the proposed MP model can be easily scaled to a large extent.

To assess the user navigation on the improved website, we partition the entire real data set into training and testing sets. We use the training data to generate improved structures which are evaluated on the testing data using simulations to approximate the real usage. We define two metrics and use them to assess whether user navigation is indeed enhanced on the improved structure. Particularly, the first metric measures whether the *average* user navigation is facilitated in the improved website, and the second metric measures how many users can benefit from the improved structure. Evaluation results confirm that user navigation on the improved website is greatly enhanced.

In summary, this paper makes the following contributions. First, we explore the problem of improving user navigation on a website with minimal changes to the current structure, an important question that has never been examined in the literature. We show that our MP model not only successfully accomplishes the task but also generates the optimal solutions surprisingly fast. The experiments on synthetic data indicate that our model also scales up very well. Second, we model the out-degree as a cost term in the objective function instead of as hard constraints. This allows a page to have more links than the out-degree threshold if the cost is reasonable and hence offers a good balance between minimizing changes to a website and reducing information overload to users. Third, we propose two evaluation metrics and use them to assess the improved structure to confirm the validity of our model. The evaluation procedure developed in this paper provides a framework for evaluating website structures in similar studies.

The rest of the paper is organized as follows: Section 2 reviews related literature. Section 3 introduces the metric for evaluating user navigation and describes the problem. Section 4 presents the model formulation with several illustrative examples. Section 5 describes the data set, analyzes the results of computational experiments, and reports the evaluation results. Section 6 discusses issues related to this research, and Section 7 concludes the paper.

2 Related Work

The growth of the Internet has led to numerous studies on improving user navigations with the knowledge mined from webserver logs and they can be generally categorized in to web personalization and web transformation approaches [11].

Web personalization is the process of "tailoring" webpages to the needs of specific users using the information of the users' navigational behavior and profile data [21]. Perkowitz and Etzioni [11] describe an approach that automatically synthesizes index pages which contain links to pages pertaining to particular topics based on the co-occurrence frequency of pages in user traversals, to facilitate user navigation. The methods proposed by Mobasher et al. [22], [23], [24] and Yan et al. [25] create clusters of users profiles from weblogs and then dynamically generate links for users who are classified into different categories based on their access patterns.

Nakagawa and Mobasher [26] develop a hybrid personalization system that can dynamically switch between recommendation models based on degree of connectivity and the user's position in the site. For reviews on web personalization approaches, see [21] and [27].

Web transformation, on the other hand, involves changing the structure of a website to facilitate the navigation for a large set of users [28] instead of personalizing pages for individual users. Fu et al. [29] describe an approach to reorganize webpages so as to provide users with their desired information in fewer clicks. However, this approach considers only local structures in a website rather than the site as a whole, so the new structure may not be necessarily optimal. Gupta et al. [19] propose a heuristic method based on simulated annealing to relink webpages to improve navigability. This method makes use of the aggregate user preference data and can be used to improve the link structure in websites for both wired and wireless devices. However, this approach does not yield optimal solutions and takes relatively a long time (10 to 15 hours) to run even for a small website. Lin [20] develops integer programming models to reorganize a website based on the cohesion between pages to reduce information overload and search depth for users. In addition, a two-stage heuristic involving two integer-programming models is developed to reduce the computation time. However, this heuristic still requires very long computation times to solve for the optimal solution, especially when the website contains many links. Besides, the models were tested on randomly generated websites only, so its applicability on real websites remains questionable. To resolve the efficiency problem in [20], Lin and Tseng [28] propose an ant colony system to reorganize website structures. Although their approach is shown to provide solutions in a relatively short computation time, the sizes of the synthetic websites and real website tested in [28] are still relatively small, posing questions on its scalability to large-sized websites.

There are several remarkable differences between web transformation and personalization approaches. First, transformation approaches create or modify the structure of a website used for all users, while personalization approaches dynamically reconstitute pages for individual users. Hence, there is no predefined/built-in web structure for personalization approaches. Second, in order to understand the preference of individual users, personalization approaches need to collect information associated with these users (known as user profiles). This computationally intensive and time-consuming process is not required for transformation approaches. Third, transformation approaches make use of aggregate usage data from weblog files and do not require tracking the past usage for each user while dynamic pages are typically generated based on the users' traversal path. Thus, personalization approaches are more suitable for dynamic websites whose contents are more volatile and transformation approaches are more appropriate for websites that have a built-in structure and store relatively static and stable contents.

This paper examines the questions of how to improve user navigation in a website with *minimal* changes to its structure. It complements the literature of transformation approaches that focus on reconstructing the link structure of a website. As a result, our model is suitable for website maintenance and can be applied in a regular manner.

3 Metric for Evaluating Navigation Effectiveness

3.1 The Metric

Our objective is to improve the navigation effectiveness of a website with minimal changes. Therefore, the first question is, given a website, how to evaluate its *navigation effectiveness*. Marsico and Levialdi [30] point out that information becomes useful only when it is presented in a way consistent with the target users' expectation. Palmer [31] indicates that an easy-navigated website should allow users to access desired data without getting lost or having to backtrack. We follow these ideas and evaluate a website's navigation effectiveness based on *how consistently the information is organized with respect to the user's expectations*. Thus, a well-structured website should be organized in such a way that the discrepancy between its structure and users' expectation of the structure is minimized. Since users of informational websites typically have some information targets [19], [32], i.e., some specific information they are seeking, we measure this discrepancy by the number of times a user has attempted before locating the target.

Our metric is related to the notion of *information scent* developed in the context of *information foraging theory* [33], [34], [35]. Information foraging theory models the cost structure of human information gathering using the analogy of animals foraging for food and is a widely accepted theory for addressing the information seeking process on the web [32], [36], [37], [38], [39]. Information scent refers to proximal cues (e.g., the snippets of text and graphics of links) that allow users to estimate the location of the "distal" target information and determine an appropriate path [34]. Users are faced with a decision point at each page; they use information scent to evaluate the likely effort and the probability of reaching their targets via each link and make navigation decisions accordingly [7]. Consequently, a user is assumed to follow the path that appears most likely to lead him to the target. This suggests that a user may backtrack to an already visited page to traverse a new path if he could not locate the target page in the current path. Therefore, we use *the number of paths a user has traversed to reach the target* as a proximate measure to the number of times the user has attempted to locate one target.

We use backtracks to identify the paths that a user has traversed, where a *backtrack* is defined as a user's revisit to a previously browsed page. The intuition is that users will backtrack if they do not find the page where they expect it [40]. Thus, a *path* is defined as a sequence of pages visited by a user without backtracking, a concept that is similar to the *maximal forward reference* defined in Chen et al. [41]. Essentially, each backtracking point is the end of a path. Hence, the more paths a user has traversed to reach the target, the more discrepant the site structure is from the user's expectation.

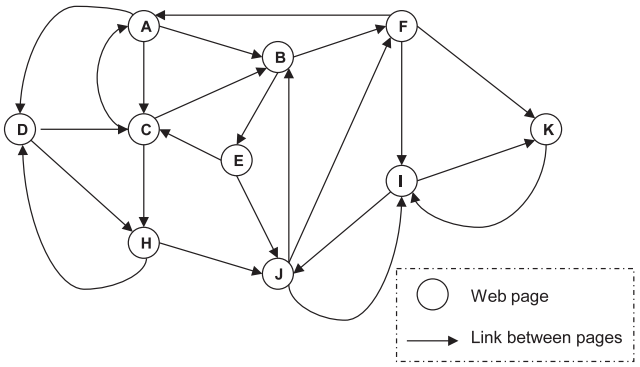


Fig. 1. A website with 10 pages.

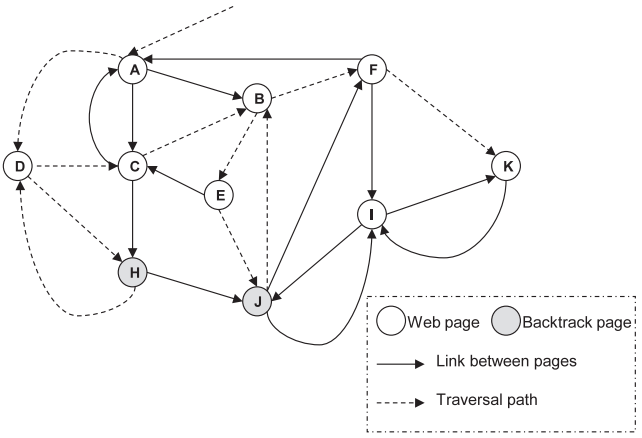


Fig. 2. Example of a mini session.

3.2 An Example

We use an example to illustrate the aforementioned concepts and how to extract the metric from weblog files. To analyze the interaction between users and a website, the log files must be broken up into user *sessions*. Cooley et al. [42] define a session as a group of activities performed by a user during his visit to a site and propose timeout methods to demarcate sessions from raw log files. In this definition, a session may include one or more target pages, as a user may visit several targets during a single session. Since the metric used in our analysis is the number of paths traversed to find one target, we use a different term *mini session* to refer to a group of pages visited by a user for only *one* target. Thus, a session may contain one or more mini sessions, each of which comprises a set of paths traversed to reach the target.

We use the page-stay timeout heuristic described in [40], [43] to demarcate mini sessions. Specifically, we identify whether a page is the target page by evaluating if the time spent on that page is greater than a timeout threshold. The intuition is that a user generally spends more time reading on the documents that they find relevant than those they do not [44]. Though it is impossible to identify user sessions unerringly from weblog files [19], we find the page-stay heuristic an appropriate technique for the context of our problem and we provide a detailed discussion on this heuristic in Section 6.

We depict in Fig. 1 a hypothetical website that has 10 pages. Fig. 2 illustrates a mini session, where a user starts from A, browses D and H, and backtracks to D, from where he visits C, B, E, J, and backtracks to B. Then, this user goes from B to F and finally reaches the target K. We formally denote the mini session by $S = \{\{A, D, H\}, \{C, B, E, J\}, \{F, K\}\}$, where an element in S represents a path traversed by the user. In this example, mini session S has three paths as the user backtracks at H and J before reaching the target K. Note that D and B only appear once in S because of caching.

3.3 Problem Description

Difficulty in navigation is reported as the problem that triggers most consumers to abandon a website and switch to a competitor [45]. Generally, having traversed several paths to locate a target indicates that this user is likely to have experienced navigation difficulty. Therefore, Webmasters can ensure effective user navigation by improving the site structure to help users reach targets faster. This is especially vital to commercial websites, because easy-navigated websites can create a positive attitude toward the firm, and stimulate online purchases [46], whereas websites with low usability are unlikely to attract and retain customers [47].

Our model allows Webmasters to specify *a goal for user navigation* that the improved structure should meet. This goal is associated with individual target pages and is defined as the *maximum number of paths allowed to reach the target page* in a mini session. We term this goal the *path threshold* for short in this paper. In other words, in order to achieve the user navigation goal, the website structure must be altered in a way such that the number of paths needed to locate the targets in the improved structure is not larger than the path threshold.

In the example shown in Fig. 2, the user has traversed three paths before reaching the target. An intuitive solution to help this user reach the target faster is to introduce more links [10], [19], [48]. There are many ways to add extra links. If a link is added from D to K, the user can directly reach K via D, and hence reach the target in the first path. Thus, adding this link "saves" the user two paths. Similarly, establishing a link from B to K enables the user to reach the target in the second path. Hence, this saves him one path. We could also insert a link from E to K, and this is considered the same as linking B to K. This is because both B and E are pages visited in the second path, so linking either one to K saves only one path. Yet, another possibility is to link C to F, a nontarget page. In this case, we assume that the user does not follow the new link, because it does not directly connect a page to the target.

While many links can be added to improve navigability, our objective is to achieve the specified goal for user navigation with minimal changes to a website. We measure the changes by the number of new links added to the current site structure. There are several reasons that we should insert minimal links. First, minimizing changes to the current structure can avoid disorienting familiar users. Second, adding unnecessary links can lead to pages having too many links, which increases users' cognitive loads and makes it difficult for them to read and comprehend [49]. Third, since our model improves site structures on a regular basis, the number of new links should be kept at minimum

such that the links in a website in the whole course of maintenance do not expand in a chaotic manner.

There are cases where users could have reached the targets through existing links, but failed to do so in practice. One reason could be that these links are placed in inconspicuous locations and hence are not easily noticeable. Another reason might be that the labels of these links are misleading or confusing, causing difficulty to users in predicting the content at the target page [50]. As a result, Webmasters should focus on enhancing the design of these existing links before adding new links. Our model considers this issue by placing a preference on the selection of such existing links.

4 Problem Formulation

4.1 The Model

Our problem can be regarded as a special graph optimization problem. We model a website as a directed graph, with nodes representing pages and arcs representing links. Let N be the set of all webpages and λ_{ij}, where $i, j \in N$, denote page connectivity in the current structure, with $\lambda_{ij} = 1$ indicating page i has a link to page j, and $\lambda_{ij} = 0$ otherwise. The current out-degree for page i is denoted by $W_i = \sum_{j \in N} \lambda_{ij}$.

From the log files, we obtain the set T of all mini sessions. For a mini session $S \in T$, we denote $tgt(S)$ the target page of S. Let $L_m(S)$ be the length of S, i.e., the number of paths in S, and $L_p(k, S)$, for $1 \leq k \leq L_m(S)$, be the length of the kth path in S, i.e., the number of pages in the kth path of S. We further define $docno(r, k, S)$, for $1 \leq k \leq L_m(S)$ and $1 \leq r \leq L_p(k, S)$, as the rth page visited in the kth path in S. Take the mini session S in Fig. 2 for example, it follows that $L_m(S) = 3, L_p(1, S) = 3$, and $docno(1, 1, S) = A$, as this mini session has three paths and the first path has three pages (A, D, and H) in which page A is the first page. We define $E = \{(i, j) : i, j \in N$ and $\exists S \in T$ such that $i \in S$ and $j = tgt\ (S)\}$ and $N_E = \{i : (i, j) \in E\}$. In essence, E is the set of *candidate links* that can be selected to improve the site structure to help users reach their targets faster. Our problem is to determine whether to establish a link from i to j for $(i, j) \in E$. Let $x_{ij} \in \{0, 1\}$ denote the decision variable such that $x_{ij} = 1$ indicates establishing the link.

As explained earlier, Webmasters can set a goal for user navigation for each target page, which is denoted by b_j and is termed the *path threshold* for page j. Given a mini session S with target page j and a path threshold b_j, we can determine whether the user navigation goal is achieved in S by comparing the length of S, i.e., $L_m(S)$, with path threshold (b_j) for the target page of S. If the length of S is larger than b_j, it indicates the user navigation in S is "below" the goal. Then, we need to alter the site structure to improve the user navigation in S to meet the goal. Otherwise, no improvement is needed for S.

Intuitively, given path thresholds, we can determine which mini sessions need to be improved and hence are relevant to our decision (termed *relevant mini sessions*). The *irrelevant* mini session are not considered in our model. We denote the set of relevant mini sessions by $T^R \subseteq T$. For a mini session $S \in T^R$, it is said to be *improved* if the website is altered in a way such that the user could reach the target within the associated path threshold after changes are made to the site structure.

We define parameters a^S_{ijkr} to be 1 if $docno(r, k, S) = i$ and $tgt(S) = j$, and 0 otherwise. In other words, $a^S_{ijkr} = 1$ if and only if page i is the rth visited page in the kth path in S and page j is the target of S. Further, we define variable c^S_{kr} which will be set to one if the solution indicates establishing a link from the rth page in the kth path in S to the target page of S, i.e., $tgt(S)$, and 0 otherwise. As will be explained later, the use of a^S_{ijkr} is to build connections between variables x_{ij} and c^S_{kr}, where the first variable uses global indices and the latter is defined using local indices.

Similar to prior studies, appropriate out-degree thresholds can be specified for webpages. We denote C_i the *out-degree threshold* for page i. Nevertheless, our model "penalizes" a page if its out-degree is larger than the threshold instead of modeling the threshold as a hard constraint. In effect, out-degree C_i indicates the maximum number of links that page i can have without being penalized. Let p_i be the number of links exceeding the out-degree threshold C_i for page i in the improved structure. Depending on the application of our model, different weights of penalties can be imposed on pages whose out-degree exceeds the respective out-degree threshold. We denote the weight by m and term it the *multiplier for the penalty term*. Table 1 provides a summary of the notations used in this paper.

The problem of improving the user navigation on a website while minimizing the changes to its current structure can then be formulated as the mathematical programming model below:

$$\text{Minimize} \sum_{(i,j) \in E} x_{ij}[1 - \lambda_{ij}(1 - \varepsilon)] + m \sum_{i \in N_E} p_i$$

subject to

$$c^S_{kr} = \sum_{(i,j) \in E} a^S_{ijkr} x_{ij}; r = 1, 2, \ldots, L_p(k, S), \quad k = 1, 2, \ldots, L_m(S), \forall S \in T^R \tag{1}$$

$$\sum_{k=1}^{b_j} \sum_{r=1}^{L_p(k,S)} c^S_{kr} \geq 1; \forall S \in T^R, j = tgt(S) \tag{2}$$

$$\sum_{j:(i,j) \in E} x_{ij}(1 - \lambda_{ij}) + W_i - p_i \leq C_i; \forall i \in N_E \tag{3}$$

$$x_{ij} \in \{0, 1\}, p_i \in \{0\} \cup \mathrm{Z}^+, \forall (i, j) \in E, i \in N_E. \tag{4}$$

The objective function minimizes the cost needed to improve the website structure, where the cost consists of two components: 1) the number of new links to be established (the first summation), and 2) the penalties on pages containing excessive links, i.e., more links than the out-degree threshold (C_i), in the improved structure (the second summation).

We have noted that some existing links may often be neglected by users due to poor design or ambiguous labels.

TABLE 1
Summary of Notations

Notation	Definition
S	A mini session that contains the set of paths traversed by a user to locate one target page.
T	The set of all identified mini sessions.
T^R	The set of *relevant mini sessions*, i.e., mini sessions that need to be facilitated for given path thresholds.
N	The set of all Web pages.
λ_{ij}	1 if page i has a link to page j in the current structure; 0 otherwise.
E	The set of candidate links which can be selected for improving user navigation.
E^R	The set of *relevant candidate links*, i.e., candidate links that can help improve user navigation to meet the goal.
N_E	The set of the source nodes of links in set E.
W_i	The current out-degree of page i.
C_i	The out-degree threshold for page i.
p_i	The number of links that exceed the out-degree threshold C_i in page i.
m	Multiplier for the penalty term in the objective function.
b_j	The path threshold for mini sessions in which page j is the target page.
a^S_{ijkr}	1 if i is the rth page in the kth path and j is the target page in mini session S; 0 otherwise.
x_{ij}	1 if the link from page i to j is selected; 0 otherwise.
c^S_{kr}	1 if in mini session S, a link from rth page in the kth path to the target is selected; 0 otherwise.
$tgt\ (S)$	The target page of mini session S

Such links should be improved first before any new links are established. Therefore, we introduce $[1-\lambda_{ij}(1-\varepsilon)]$, where ε is a very small number, in the objective function to let the model select existing links whenever possible. Note that if $(1-\varepsilon)$ is not present, then there is no cost in choosing an existing link, and this could lead to a number of optima. As an extreme example, if $(1-\varepsilon)$ is removed and the penalty term is not included, the costs of establishing new links, i.e., $\sum_{(i,j)\in E} x_{ij}(1-\lambda_{ij})$ when selecting all existing links are the same as the costs when none of them is selected. This occurs because there is no cost in selecting an existing link, i.e., $(1-\lambda_{ij})=0$, when $\lambda_{ij}=1$. Thus, we add $(1-\varepsilon)$ to impose a very small cost on improving an existing link such that the model will select the minimal number of existing links for improvement.

Constraint (1) defines variable c^S_{kr}, which is set to 1 if and only if $a^S_{ijkr}=1$ and $x_{ij}=1$, for some $(i,j)\in E$, and 0 otherwise. It uses parameter a^S_{ijkr} to build connections between variables x_{ij} and c^S_{kr}, because the objective function is defined by variable x_{ij} which uses global indices (i and j) to label webpages, whereas the constraint (2) is defined by variable c^S_{kr} which uses local indices (S, k, and r) to identify a page's position in a mini session. The values of a^S_{ijkr} can be easily obtained after mini sessions are identified from weblog files, as demonstrated in an example later.

Constraint (2) requires that the goal for user navigation be achieved for all relevant mini sessions, where the goal is defined as path threshold (b_j). Particularly, for a mini session $S\in T^R$ in which the user navigation is below the specified goal, i.e., $L_m(S)>b_j$ for $j=tgt(S)$, at least one link from pages visited on or before the b_jth path to the target j is either established or improved so that the user can reach the page j within the path threshold set by the Webmaster.

Constraint (3) uses p_i to capture the number of links exceeding the out-degree threshold C_i for page $i\in N_E$. This value (p_i) is then used to compute penalties in the objective function. The degree of penalty can be controlled by the multiplier for the penalty term (m). Constraint (4) imposes that decision variables are binary and p_i are nonnegative integers.

In practice, parameter values such as b_j and C_i are context dependent and can vary across webpages. However, for simplicity, we used a single value denoted by b as the path threshold and denoted by C as the out-degree threshold for all pages in our examples and experiments. We discuss how to select appropriate parameter values in detail in Section 6.

Note that a special case of our MP model when $m=0$ and $\lambda_{ij}=0$ for $i,j\in N$ can be viewed as the hitting set problem. That is, when pages are not penalized for having too many links and no preference is placed on selecting existing links, our formulation reduces to a hitting set problem. The hitting set problem is stated as follows: Given a ground set X and a collection of subset F, the objective is to find the smallest subset $H\subseteq X$ of elements that "hits" every set of F, i.e., $H\cap A\neq\emptyset$ for every $A\in F$. It is equivalent to the set-covering problem which is known to be NP-complete [51].

In the context of our problem, E is the ground set containing all candidate links that can be used for improving navigability. For a relevant mini session $S\in T^R$ with path threshold b_j, denote the set of candidate links that can be selected to help achieve the user navigation goal in S by $B=\{(i,j):(i,j)\in E$ and $a^S_{ijkr}=1$, for $j=tgt(S)$, $1\le k\le b_j$, and $1\le r\le L_p(k,S)\}$. Further, we denote the collection of the sets of such candidate links for all mini sessions in T^R by D. The objective of our formulation when $m=0$ and $\lambda_{ij}=0$ is to find the smallest subset $H\subseteq E$ of elements that hits every set of D, i.e., $H\cap B\neq\emptyset$ for every $B\in D$.

4.2 Observations on Reduction of Problem Size

The formulation has $|E|$ binary variables corresponding to the number of candidate links and $|T^R|$ constraints

corresponding to the number of relevant mini sessions. While in practice the size of a website and the number of mini sessions obtained from server logs can be very large, it turns out that, in the context of our problem, the formulation can be reduced to a significantly smaller one that can be quickly solved. We make several observations related to the problem size. These observations together provide insights into why the problem size in our formulation can be considerably reduced and help explain the fast solution times in our experiments in the later sections. In fact, as will be shown later, our formulation has already taken steps to reduce the problem size.

4.2.1 Relevant Mini Sessions

Recall that a mini session is relevant only if its length is larger than the corresponding path threshold. Consequently, only relevant mini sessions need to be considered for improvement and this leads to a large number of *irrelevant* mini sessions (denoted as T^I) being eliminated from consideration in our MP model. In other words, define $T^I = T \backslash T^R$, any mini session $S \in T^I$ will not be considered in our formulation as the user navigation in S already meets the goal (set as path threshold).

As will be shown later, the choice of path threshold can have significant impacts on relevant mini sessions. Generally, increasing the path threshold leads to a smaller number of relevant mini sessions while decreasing it has the opposite effect. For example, for the real data set used in the experiments (details are provided in Section 5.1), when the path threshold (b) increases from 3 to 5, the number of relevant mini sessions reduces from several thousand to only a few hundred. Even for the case when $b = 1$, a large number of irrelevant mini sessions can be eliminated from consideration.

4.2.2 Relevant Candidate Links

Define $E' = \{(i, j) : i, j \in N\}$ as the possible links between all pages in a website with node set N. Theoretically, any link from E' can be considered in our decision problem without a preprocessing step, leading to a total number of $|N| \times |N|$ links (variables). This number can be very large even for a small website. Intuitively, not every link in E' can be used to improve user navigation. Recall that we term the links that can be selected to help user navigate as *candidate* links (denoted by E), which can be easily obtained from mini sessions. Thus, it follows that $x_{ij} = 0$ in the optimal solution $\forall (i, j) \in E' \backslash E$. In other words, noncandidate links do not help improve user navigation and hence need not enter the formulation.

It turns out that many candidate links can also be eliminated from consideration because they are not relevant to the decision for two reasons. First, given path thresholds, denote the set of candidate links for relevant mini sessions by E^{RM} and the set of candidate links for irrelevant mini sessions by E^{IM}. It follows that the candidate links in $E^{IM} \backslash E^{RM}$ are only for irrelevant mini sessions that need no improvement and hence can be eliminated from consideration. Second, not all candidate links in E^{RM} might be relevant to the decision. Particularly, for a mini session $S \in T^R$ with path threshold b, a link is said to be *relevant* to S if adding/improving it can help the user in S reach the target in no more than b paths, i.e., achieve the user navigation goal in S. Thus, the candidate links relevant to the decision are those originating from the pages visited on the bth path or before. The other candidate links for S can be eliminated from consideration because selecting them cannot improve S to achieve the specified goal for user navigation. We term the set of candidate links relevant to the decision the *relevant candidate links*, and we denote them by $E^R = \{(i, j) : (i, j) \in E \text{ and } \exists S \in T^R \text{ such that } i \in S,\ j = tgt(S), a^S_{ijkr} = 1 \text{ for } 1 \leq k \leq b_j, \text{ and } 1 \leq r \leq L_p(k, S)\}$. This leads to $x_{ij} = 0$ in the optimal solution $\forall (i, j) \in E^{RM} \backslash E^R$. Thus, the cardinality of the set of relevant candidate links ($|E^R|$) could be relatively small even for a large website.

	n_1	n_2	n_3	n_4	n_5	n_6
n_1	0	0	0	1	1	0
n_2	1	0	0	1	1	1
n_3	1	1	0	0	0	0
n_4	1	1	1	0	1	0
n_5	1	1	1	1	0	1
n_6	0	1	1	1	0	0

Fig. 3. The Connectivity matrix for illustrative examples.

4.2.3 Dominated Mini Sessions

Another reason for the problem size reduction is that many relevant mini sessions "dominate" others with respect to relevant candidate links. Mini session S_p *dominates* mini session S_q if the set of relevant candidate links for S_q contains (at least) all relevant candidate links for S_p. This dominance is strict if there exists at least one candidate link that is relevant to S_q but is irrelevant to S_p. Therefore, when a mini session is improved in the new structure, the mini sessions that are dominated by this one are also improved. Consequently, the constraints corresponding to dominated mini sessions are redundant and can be eliminated from consideration in the MP model.

4.3 Illustrative Examples

In this section, we use examples to illustrate our model and how the choices of different parameter values could affect the problem size of the model, the solution spaces, and the solutions. Let the matrix in Fig. 3 represent the current connectivity of a website that has six pages. An entry value indicates if a row node has a link to a column node, with 1 indicating a link exists. The out-degree of page i can be obtained by summing the entry values in row n_i. For example, the matrix shows page 1 has a link to pages 4 and 5, i.e., $\lambda_{14} = \lambda_{15} = 1$, and hence its out-degree is $W_1 = 2$.

Table 2 shows a set T of six mini sessions, each of which has two or more paths. For example, mini session S_1 has three paths and the target is page 6. The user starts from page 2 and backtracks twice at pages 1 and 4, respectively, because neither page has a link to 6. In the third path, the user finally locates the target after visiting page 5. We provide in Table 3 the set of candidate links that can be added or improved to help users reach the targets faster for each of the six mini sessions.

Now, consider a case where the Webmaster sets the path threshold to $b = 1$, which means that the site structure

TABLE 2
An Example of Mini Sessions

ID	Mini sessions
S_1	{{2, 1}, {4}, {5, 6}}
S_2	{{4, 3},{5, 1}, {2, 6}}
S_3	{{1, 5, 2}, {6, 4}}
S_4	{{6, 3}, {2, 1}, {5, 4}}
S_5	{{4, 1}, {5}, {3}, {2, 6}}
S_6	{{5, 3, 1}, {2, 4}}

should be improved such that users can reach their targets in one path. In this case, all six mini session are relevant as their lengths are all larger than the path threshold. This requires that for each mini session, at least one link from the pages visited in the first path to the target be established or improved. In other words, only the candidate links originating from the pages visited in the first path are relevant to our decision. Table 4 lists the set of relevant candidate links for relevant mini sessions.

For the purpose of illustration, we do not consider the penalty term for now. The problem is formulated as

$$\text{Minimize} \sum_{(i,j)\in E} x_{ij}[1-\lambda_{ij}(1-\varepsilon)]$$

subject to

$$c_{kr}^S = \sum_{(i,j)\in E} a_{ijkr}^S x_{ij}; r = 1,2,\ldots,L_p(k,S), \quad k = 1,2,\ldots,L_m(S), \forall S \in T^R \tag{1}$$

$$c_{11}^{S_1} + c_{12}^{S_1} \geq 1 \tag{S1}$$

$$c_{11}^{S_2} + c_{12}^{S_2} \geq 1 \tag{S2}$$

$$c_{11}^{S_3} + c_{12}^{S_3} + c_{13}^{S_3} \geq 1 \tag{S3}$$

$$c_{11}^{S_4} + c_{12}^{S_4} \geq 1 \tag{S4}$$

$$c_{11}^{S_5} + c_{12}^{S_5} \geq 1 \tag{S5}$$

$$c_{11}^{S_6} + c_{12}^{S_6} + c_{13}^{S_6} \geq 1 \tag{S6}$$

TABLE 3
The Set of All Candidate Links

ID	Candidate links
S_1	{(2, 6), (1, 6), (4, 6)}
S_2	{(4, 6), (3, 6), (5, 6), (1, 6)}
S_3	{(1, 4), (5, 4), (2, 4)}
S_4	{(6, 4), (3, 4), (2, 4), (1, 4)}
S_5	{(4, 6), (1, 6), (5, 6), (3, 6)}
S_6	{(5, 4), (3, 4), (1, 4)}

TABLE 4
Relevant Candidate Links for $b = 1$

ID	Relevant candidate links
S_1	{(2, 6), (1, 6)}
S_2	{(4, 6), (3, 6)}
S_3	{(1, 4), (5, 4), (2, 4)}
S_4	{(6, 4), (3, 4)}
S_5	{(4, 6), (1, 6)}
S_6	{(5, 4), (3, 4), (1, 4)}

$$x_{ij} \in \{0,1\}, \forall (i,j) \in E.$$

The first constraint connects x_{ij} with c_{kr}^S, such that when $x_{ij} = 1$ and $a_{ijkr}^S = 1$, the corresponding c_{kr}^S is also set to 1. The a_{ijkr}^S values can be easily obtained after mini sessions are extracted from log data. For example, it follows from Table 2 that $a_{2611}^{S_1} = 1$, because the first page in the first path of S_1 is page 2 and the target is page 6. As a result, a solution that links pages 2 to 6, i.e., $x_{26} = 1$, will also set $c_{11}^{S_1} = 1$, and hence satisfy constraint (S_1). The other values of a_{ijkr}^S can be obtained similarly. We set $\varepsilon = 1.0\text{E}-8$ in the example. Solving this math program results in the optimal solution $x_{26} = x_{46} = x_{14} = x_{64} = 1$, with the other variables being 0. The only new link needed is (4, 6), with the others being existing links, i.e., $\lambda_{26} = \lambda_{14} = \lambda_{64} = 1$.

The solution will change accordingly when we incorporate the penalty term into the objective function and change the path threshold. For instance, if the out-degree threshold and the multiplier for the penalty term are set as $C = 3$ and $m = 5$, respectively, and the path threshold increases to $b = 2$, then the new problem is formulated as follows:

$$\text{Minimize} \sum_{(i,j)\in E} x_{ij}[1-\lambda_{ij}(1-\varepsilon)] + 5\sum_{i\in N_E} p_i$$

subject to

$$c_{kr}^S = \sum_{(i,j)\in E} a_{ijkr}^S x_{ij}; r = 1,2,\ldots,L_p(k,S), \quad k = 1,2,\ldots,L_m(S), \forall S \in T^R \tag{1}$$

$$c_{11}^{S_1} + c_{12}^{S_1} + c_{21}^{S_1} \geq 1 \tag{S1}$$

$$c_{11}^{S_2} + c_{12}^{S_2} + c_{21}^{S_2} + c_{22}^{S_2} \geq 1 \tag{S2}$$

$$c_{11}^{S_4} + c_{12}^{S_4} + c_{21}^{S_4} + c_{22}^{S_4} \geq 1 \tag{S4}$$

$$c_{11}^{S_5} + c_{12}^{S_5} + c_{21}^{S_5} \geq 1 \tag{S5}$$

$$\sum_{j=1} x_{ij}(1-\lambda_{ij}) + W_i - p_i \leq 3; \forall i \in N_E \tag{P}$$

$$x_{ij} \in \{0,1\}, p_i \in \{0\} \cup \mathrm{Z}^+, \forall (i,j) \in E, i \in N_E.$$

The new formulation has several noticeable changes. First, the new objective function has an extra term which penalizes the pages that have excessive links, i.e., more

TABLE 5
Relevant Candidate Links for $b = 2$

ID	Relevant candidate links
S_1	{(2, 6), (1, 6), (4, 6)}
S_2	{(4, 6), (3, 6), (5, 6), (1, 6)}
S_4	{(6, 4), (3, 4), (2, 4), (1, 4)}
S_5	{(4, 6), (1, 6), (5, 6)}

than three links $(C = 3)$ in this example. Specifically, adding a new link into pages having three links or more will result in a penalty which is five times $(m = 5)$ as large as the cost of adding a link into pages containing less than three links. Second, constraint (P) is added to compute exactly how many links exceed the out-degree threshold for each page. Third, instead of having six constraints, the new formulation has only four constraints corresponding to the four relevant mini sessions. Two mini sessions $(S_3 \text{ and } S_6)$ no longer need to be facilitated when b increases to 2, because they have only two paths and hence are not considered in this example.

Increasing the path threshold also leads to more relevant candidate links and subsequently a larger solution space for each relevant mini session. As shown in Table 5, some candidate links that were irrelevant for $b = 1$ will become relevant to the decision when path threshold increases to $b = 2$. For example, S_2 has two more relevant candidate links for $b = 2$, i.e., (5, 6) and (1, 6), which can be used to improve S_2 to achieve the user navigation goal for $b = 2$ but not for $b = 1$. Note that S_5 strictly dominates S_2, so if S_5 is improved, S_2 will also be improved. Thus, the constraint with respect to S_2 is redundant.

Solving the math program yields the optimal solution $x_{26} = x_{14} = x_{56} = 1$, with the other variables being 0. Since all three links already exist according to the connectivity matrix, no new link is needed at this time. It is worth mentioning that when the multiplier for the penalty term is set to $m = 5$ with the path threshold remaining at $b = 1$, the optimal solution would change to $x_{16} = x_{36} = x_{64} = x_{54} = 1$. Compared to the first case where $m = 0$ and $b = 1$, this case adds two new links, i.e., (1, 6) and (3, 6), as opposed to only one link, i.e., (4, 6). The reason is that after adding the penalty term in the objective function, the MP model would add (1, 6) and (3, 6) for a total cost of 2 (pages 1 and 3 have only two links and are not penalized for having one more link) instead of adding (4, 6) to incur a penalty of 5 (page 4 already has four links). Nevertheless, when the path threshold increases to $b = 2$, no new link is needed because fewer mini sessions are relevant to the decision and the solution space for each relevant mini session increases.

5 Computational Experiments and Performance Evaluations

Extensive experiments were conducted, both on a data set collected from a real website and on synthetic data sets. We first tested the model with varying parameters values on all data sets. Then, we partitioned the real data into training and testing data. We used the training data to generate improved site structures which were evaluated on the testing data using two metrics that are discussed in detail later. Moreover, we compared the results of our model with that of a heuristic.

TABLE 6
Out-Degree Statistics

Out-degree	Number of pages
>100	1
81–100	2
61–80	10
41–60	21
21–40	166
11–20	538
0–10	178
Total	916

5.1 Real Data Set

5.1.1 Description of the Real Data Set

The real data set was collected from the Music Machines website (http://machines.hyperreal.org) and contained about four million requests that were recorded in a span of four months. This data set is publicly available and has been widely used in the literature [19], [29]. Table 6 shows the number of pages in the website that had out-degrees within a specified range. This website has in total 916 pages, of which 716 have an out-degree of 20 or less, with the majority (83 percent) of the remaining pages having 40 links or less.

Before analysis, we followed the log preprocessing steps described in [29] to filter irrelevant information from raw log files. These steps include: 1) filter out requests to pages generated by Common Gateway Interface (CGI) or other server-side scripts as we only consider static pages that are designed as part of a website structure, 2) ignore unsuccessful requests (returned HTTP status code not 200), and 3) remove requests to image files (.gif, .jpg, etc.), as images are in general automatically downloaded due to the HTML tags rather than explicitly requested by users [33].

We utilized the page-stay time to identify target pages and to demarcate mini sessions from the processed log files. Three time thresholds (i.e., 1, 2, and 5 minutes) were used in the tests to examine how results changes with respect to different parameter values. Furthermore, we adapted the algorithm proposed in [40] to identify the backtracking pages in mini sessions, which are then used to demarcate the paths traversed to reach the targets. Table 7 lists the number of mini sessions comprising a given number of paths (> 1) for different time thresholds.

5.1.2 Results and Analysis—Real Data Set

We set $\varepsilon = 1.0\text{E} - 8$ and vary the out-degree threshold (C), the path threshold (b), and the multiplier for the penalty term (m) to examine how results change with respect to these parameters. Table 8 reports the experiment results. The math programs were coded in AMPL and solved using CPLEX/AMPL 8.1 on a PC running Windows XP on an Intel Core 2 Duo E6300 processor. The times for generating optimal

TABLE 7
Path Characteristics of Mini Sessions

Number of paths	Number of mini sessions		
	t=1 min.	t=2 min.	t=5 min.
2	27,140	23,485	20,964
3	4,457	4,242	4,075
4	1,340	1,469	1,427
5	477	590	652
6–10	395	498	525
>10	3	8	7
Total	33,812	30,292	27,650

solutions varied from 0.109 to 0.938 seconds, indicating that our model is very effective and practical for real-world websites. We have reported the times taken to solve the math programs only; the times taken for preprocessing steps and obtaining values of a^S_{ijkr} are not included, as they can be done very quickly in practice. Note that the size of the real website considered in our paper is significantly larger than the average website size [52] as well as those used in related papers addressing the website reorganization problem. For example, Gupta et al. [19] and Lin and Tseng [28] report results based on websites with only 427 and 146 pages, respectively.

In Table 8, the column "No. of new links" indicates how many new links need to be added into the current structure in order to achieve the users navigation goal specified in the column "Path threshold (b)" for all mini sessions. The column "No. of excessive links" reports the number of new links added to pages that have excessive links, i.e., have more than C links. For example, if we set time threshold to $t = 2$, out-degree threshold to $C = 20$, path threshold to $b = 1$, and do not penalize pages for having excessive links

TABLE 8
Results from the Real Data Set

Time threshold (t)	Multiplier for penalty term (m)	Path threshold (b)	Out-degree threshold (C)=20				Out-degree threshold (C)=40			
			No. of new links	No. of links to be improved	No. of excessive links	Time (sec)	No. of new links	No. of links to be improved	No. of excessive links	Time (sec)
1 min.	0	1	9,180	1,806	7,354	0.109	9,180	1,806	3,482	0.109
		2	1,643	648	1,256	0.484	1,643	648	654	0.484
		3	590	375	441	0.391	590	375	215	0.406
	1	1	9,180	1,798	7,310	0.140	9,191	1,801	3,160	0.125
		2	1,659	662	940	0.625	1,708	679	190	0.625
		3	613	384	192	0.422	620	398	36	0.422
	5	1	9,180	1,798	7,310	0.140	9,213	1,803	3,147	0.125
		2	1,681	671	926	0.641	1,937	700	84	0.719
		3	647	383	170	0.563	686	408	8	0.469
2 min.	0	1	7,895	1,711	6,332	0.125	7,895	1,711	3,147	0.125
		2	1,523	666	1,192	0.484	1,523	666	685	0.469
		3	633	365	485	0.438	633	365	257	0.406
	1	1	7,897	1,702	6,261	0.156	7,909	1,706	2,808	0.140
		2	1,539	679	873	0.672	1,601	694	205	0.625
		3	648	367	231	0.532	670	376	44	0.532
	5	1	7,897	1,704	6,261	0.157	7,937	1,708	2,793	0.140
		2	1,563	677	860	0.657	1,849	723	91	0.938
		3	689	371	201	0.860	739	391	10	0.515
5 min.	0	1	7,158	1,626	5,733	0.110	7,158	1,626	2,907	0.125
		2	1,375	645	1,107	0.438	1,375	645	668	0.438
		3	562	408	455	0.407	562	408	267	0.406
	1	1	7,158	1,624	5,659	0.141	7,177	1,629	2,569	0.156
		2	1,392	681	804	0.594	1,450	698	212	0.609
		3	585	416	219	0.704	609	426	45	0.485
	5	1	7,159	1,624	5,658	0.141	7,195	1,629	2,556	0.141
		2	1,409	684	792	0.594	1,711	716	94	0.609
		3	619	419	198	0.704	679	430	9	0.454

$(m = 0)$, then we need to establish 7,895 new links and improve 1,711 existing links. Among all new links, 6,332 are added to pages having 20 links or more. The numbers are small considering the number of mini sessions needed for improvement (30,292): approximately one new link is needed per four mini sessions.

The number of new links needed decreases significantly as the path threshold (b) increases. As discussed in Section 4.2, a primary reason for this is that a large number of mini sessions become irrelevant to the decision as path threshold increases. In our test, when $t = 2$ and $b = 1$ we have $30{,}292$ mini sessions that are relevant and will be considered in the MP model. This number decreases to 6,807 as the path threshold increases to $b = 2$: the other 23,485 (= 30,292-6,807) mini sessions containing two paths already meet the user navigation goal and hence are no longer considered in the MP model for $b = 2$.

The results show that when there is no penalty term $(m = 0)$, the number of new links and the number of existing links to be improved are the same across different out-degree thresholds (C). This is because the out-degree threshold plays no role in the MP model if the penalty term is removed from the objective function. When the penalty term is imposed, i.e., $m \neq 0$, we find that while a larger multiplier for the penalty term (m) leads to more new links, it also adds fewer links to nodes having excessive links. This is anticipated because as m increases, the MP model would prefer to establish more links to pages with small out-degrees in order to prevent large penalties.

Table 8 also indicates that, when the penalty term is used, the use of a larger out-degree threshold (C) leads to more links established. This occurs because, the larger the out-degree threshold, the fewer the pages violating the out-degree threshold. Thus, a larger out-degree threshold provides the MP model with a larger "space" to add new links without being penalized, and this results in more links being established to pages having less than C links so that heavy costs can be avoided when a penalty term is imposed in the objective function.

5.1.3 Some Insights into Efficiency

The solution times taken to solve the math programs are very low, consistently remaining below 1 second. Several reasons contribute to the low solution times, as explained in Section 4.2. The first reason is that a large number of irrelevant mini sessions (those that need not to be improved for a given path threshold) can be eliminated from consideration. As a result, the number of relevant mini sessions that will be improved and considered by the MP model is not too large. Generally, increasing path thresholds reduces the number of relevant mini sessions. Thus, the size of relevant mini sessions is further reduced with increased path thresholds. In our experiment, the number of relevant mini sessions for $t = 1$ is 33,812 when path threshold is set to $b = 1$, and this number decreases to 6,672 when the path threshold increases to $b = 2$.

The second reason that can explain low solution times is that our model formulation helps reduce the number of relevant candidate links (the links can be selected to improve user navigation to meet the goal). Particularly, we identify the set of candidate links from mini sessions instead of considering all possible links in a website, and this significantly reduces the search space for our model. Besides, as explained earlier, not all candidate links can help relevant mini sessions meet the specified goal for user navigation. Such candidate links are irrelevant to our decision for a given path threshold and hence can be eliminated from consideration. In the experiments, the total number of candidate links for $t = 1$ is 23,032, and among these links, the number of relevant candidate links for $b = 1, b = 2$, and $b = 3$ are 18,373, 11,676, and 7,209, respectively. Further, the number of relevant candidate links for each relevant mini session was observed to be not very large (generally consists of five links or less) and this leads to a relatively small solution space for individual mini sessions.

The third reason for low solution times is that many mini sessions "dominate" others with respect to relevant candidate links. Consequently, the dominated mini sessions need not to be considered in the constraints and can be eliminated from our model. While the problem size can be considerably reduced in our model formulation, the remaining formulation continues to have a nontrivial number of variables and constraints. We note that the data set is from a real website and spans a four-month period, which is long enough to cover most realistic situations.

5.2 Synthetic Data Sets

In addition to the real data set, synthetic/artificial data sets were generated and considered for computational experiments to evaluate the scalability of our model with respect to the size of the website and the number of mini sessions. For this reason, the artificial website structures and mini sessions were generated to have similar statistical characteristics as the real data set. For instance, the average out-degree for pages in the real website is 15, so the link structure for the artificial website was generated in a way such that each page contained 15 links on average.

Three websites consisting of 1,000, 2,000, and 5,000 webpages were constructed. Our approach for generating the link structure is similar to that described in [28]. Particularly, to generate the link structure that contains 1,000 pages, we first constructed a complete graph of 1,000 nodes (pages) and each directed edge was assigned a random value between 0 and 1. Then, we selected the edges with the smallest 15,000 values to form the link structure, resulting in an average out-degree of 15 for this website. In a similar manner, we generated the link structures for other artificial websites. The mini sessions were generated in a slightly different manner. Specifically, we directly generated the set of relevant candidate links for each mini session instead of creating the user's traversal path. As a result, this allows us to directly apply the model on synthetic data sets. The sets of relevant candidate links in synthetic data sets has similar characteristics with those from the real one, comprising one to five relevant candidate links per each relevant mini session, with each link being randomly selected.

Each of the three artificial websites was tested with 10,000, 50,000, 100,000, and 300,000 mini sessions, resulting in 12 different categories. In each category, three data sets were generated and the results were averaged over the three sets. We note that the synthetic data sets considered in

TABLE 9
Evaluation Results on Improved Website Using Number of Paths Per Mini Session for $T = 5$ Min

Multiplier for penalty term (m)	Avg. no. of paths in improved Web site and no. of new links needed (in parenthesis)					
	Out-degree threshold C=20			Out-degree threshold C=40		
	b=1	b=2	b=3	b=1	b=2	b=3
0	1.335 (5,794)	1.589 (1,145)	1.785 (467)	1.335 (5,794)	1.589 (1,145)	1.785 (467)
1	1.346 (5,794)	1.632 (1,166)	1.815 (482)	1.349 (5,813)	1.650 (1,214)	1.827 (502)
5	1.346 (5,794)	1.639 (1,182)	1.855 (514)	1.351 (5,839)	1.680 (1,399)	1.840 (555)

this paper are significantly larger than those used in related papers, which report results based on synthetic data sets with at most 200 pages and 3,200 links [28].

The math programs for the synthetic data were coded in AMPL and solved using CPLEX/AMPL 11.1.1 on a PC running Windows 7 on a 3.4 GHz processor. We experimented the model with two out-degree thresholds, i.e., $C = 20$ and $C = 40$, and two multipliers for the penalty term, i.e., $m = 0$ and $m = 5$, on each synthetic data set. Noticeably, the times for generating optimal solutions are low for all cases and parameter values tested, ranging from 0.05 to 24.727 seconds. This indicates that the MP model is very robust to a wide range of problem sizes and parameter values. Particularly, the average solution times for website with 1,000, 2,000, and 5,000 pages are 0.231, 1.352, and 3.148 seconds. While the solution times do go up with the number of webpages, they seem to increase within a reasonable range.

Besides these data sets, two large websites with 10,000 and 30,000 pages were generated and experimented with 300,000, 600,000, and 1.2 million mini sessions to emphasize the fact that the model presented here is scalable to an even larger extent. The solution times are also remarkably low even in this case, varying from 1.734 to 33.967 seconds. In particularly, the average solution times for websites with 10,000 and 30,000 pages are 3.727 and 6.086 seconds, respectively. While the solution times also increase with the size of the website, they seem to increase linearly or slower.

5.3 Evaluation of the Improved Website

In addition to the extensive computational experiments on both real and synthetic data sets, we also perform evaluations on the improved structure to assess whether its navigation effectiveness is indeed enhanced by approximating its real usage. Specifically, we partition the real data set into a training set (first three months) and a testing set (last month). We generate the improved structure using the training data, and then evaluate it on the testing data using two metrics: *the average number of paths per mini session* and *the percentage of mini sessions enhanced to a specified threshold*. The first metric measures whether the improved structure can facilitate users to reach their targets faster than the current one on average, and the second metric measures how likely users suffering navigation difficulty can benefit from the improvements made to the site structure. The evaluation procedure using the first metric consists of three steps and is described as follows:

1. Apply the MP model on the training data to obtain the set of new links and links to be improved.
2. Acquire from the testing data the mini sessions that can be improved, i.e., having two or more paths, their length, i.e., number of paths, and the set of candidate links that can be used to improve them.
3. For each mini session acquired in step 2, check whether any candidate link matches one of the links obtained in step 1, that is, the results from the training data. If yes, with the assumption that users will traverse the new link or the enhanced link in the improved structure, remove all pages (excluding the target page) visited after the source node of the matching candidate link to obtain the new mini session for the improved website, and get its *updated length* information.

We illustrate the evaluation process with an example. Let $S_1 = \{\{2,1\},\{4\},\{5\},\{3,6\}\}$ and $S_2 = \{\{6,3\},\{2,5\},\{1,4\}\}$ be two mini sessions in testing data. Their *current lengths*, i.e., their lengths in the current structure, are $L_m(S_1) = 4$ and $L_m(S_2) = 3$. The set of candidate links for S_1 is $CL_1 = \{(2,6),(1,6),(4,6),(5,6)\}$ and for S_2 is $CL_2 = \{(6,4),(3,4),(2,4),(5,4)\}$. Let $x_{16} = x_{46} = x_{24} = 1$ be the results obtained from the MP model on training data. Clearly, links (1, 6) and (4, 6) match two candidate links for S_1 and link (2, 4) matches one candidate link for S_2, so we remove all nontarget pages after pages 1 and 4 in S_1 and page 2 in S_2. In this example, since page 1 is visited prior to page 4 in S_1, we remove all nontarget pages after page 1 instead of page 4. The new mini sessions for the improved structure are denoted by $S_1' = \{\{2,1,6\}\}$ and $S_2' = \{\{6,3\},\{2,4\}\}$ and their updated lengths are $L_m(S_1') = 1$ and $L_m(S_2') = 2$, respectively.

The evaluation results using the first metric, *the average number of paths per mini session*, are reported in Table 9. We only report the results obtained from setting the threshold $t = 5$, and the results obtained from other time thresholds are similar. Note that the average of number of paths needed to reach the targets before the improvements for $t = 5$ is 2.383.

The results in Table 9 indicate that improved structures can facilitate users to reach their targets in fewer paths than

TABLE 10
Results of Evaluation on Improved Website Using Percentage of Mini Sessions Enhanced for $T = 5$ Min

Multiplier for penalty term (m)	Evaluation threshold (b_e)	Mini sessions enhanced (%)					
		C=20			C=40		
		b=1	b=2	b=3	b=1	b=2	b=3
0	1	71.54%	51.12%	36.45%	71.54%	51.12%	36.45%
	2	86.26%	72.04%	57.53%	86.26%	72.04%	57.53%
	3	87.95%	75.72%	65.01%	87.95%	75.72%	65.01%
1	1	70.80%	48.23%	34.89%	70.60%	47.29%	33.97%
	2	85.15%	69.11%	53.97%	85.01%	67.15%	53.56%
	3	87.57%	72.47%	61.38%	87.00%	70.17%	59.66%
5	1	70.80%	47.89%	31.17%	70.52%	45.34%	33.49%
	2	85.29%	68.48%	53.28%	84.87%	64.16%	51.88%
	3	87.38%	71.32%	59.66%	86.81%	67.88%	57.55%

the current one *on average*. As an example, when $m = 0, b = 2$, and $C = 40$, the number of paths per mini session in the improved structure decreases to 1.589 from 2.383, with the help of 1,145 new links. Note that while the number of paths per mini session decreases with the decreasing value of b for the MP model on the training data, the number of new links that need to be established also increases significantly. This is anticipated because, as shown earlier, a smaller path threshold leads to more links established, which results in fewer paths per mini session in the improved structure as a consequence.

We now proceed to report the evaluation results using the second metric, *the percentage of mini sessions enhanced to a specified threshold*. We devise this metric to assess that, for mini sessions whose current length are larger than a specified evaluation threshold, how many of them are enhanced such that their updated lengths in the improved structure are no longer larger than the evaluation threshold. In other words, we use this metric to measure how many mini sessions, in which the current user navigation is below the evaluation threshold, can benefit from the improved structure.

We denote the *evaluation threshold* by b_e where the subscript e is used to distinguish the evaluation threshold from the path threshold (b) that is used for the MP model on the training set. The evaluation process is similar to the steps for the first metric, but it needs one more step after step 3, i.e., after length information is updated. Particularly, for each mini session whose current length is larger than b_e, we check if its updated length decreases to b_e or less; if yes, then the mini session is considered *enhanced* by the improved structure to satisfy the evaluation threshold.

We continue with the earlier example, where $S_1 = \{\{2,1\},\{4\},\{5\},\{3,6\}\}$ and $S_2 = \{\{6,3\},\{2,5\},\{1,4\}\}$ are in the testing set. When the evaluation threshold is set to $b_e = 3$, only S_1 needs to be evaluated, because the current length of S_1 is larger than 3, i.e., $L_m(S_1) = 4 > 3$, whereas the current length of S_2, i.e., $L_m(S_2) = 3$, is not. Suppose the solution from the training set is $x_{46} = x_{64} = 1$. We then follow the three steps in the evaluation procedure and update S_1 to $S_1' = \{\{2,1\},\{4,6\}\}$. Since its updated length is $L_m(S_1') = 2 < b_e = 3$, S_1 is considered enhanced by the improved structure. If the evaluation threshold changes to $b_e = 1$, then both S_1 and S_2 need to be evaluated, as their current lengths are larger than 1. Similarly, we update S_1 and S_2 to $S_1' = \{\{2,1\},\{4,6\}\}$ and $S_2' = \{\{6,4\}\}$. Now, only S_2 is considered enhanced in the improved structure, because its updated length is 1, while the length of S_1' is still larger than 1. Three values of b_e, i.e., $b_e = 1, 2$, and 3, were used for evaluation. We only report the results for $t = 5$ in Table 10, and the results obtained using other time thresholds are similar.

In Table 10, the values of b, m, and C are the parameters used to generate the improved structures using training data. The column "Mini sessions enhanced (%)" lists the percentage of mini sessions that have more than b_e paths in the current structure but are enhanced to satisfy the evaluation threshold, i.e., have b_e or less paths, in the improved structure. For example, when using $m = 0$, $C = 40$, and $b = 1$ for the training data, 71.54 percent of the mini sessions that have two or more paths in the current structure are enhanced to have only one path $(b_e = 1)$ in the improved structure. This means that the improved structure helps more than 70 percent of users reach the targets in *at least* one less path than the current structure.

As shown in Table 10, the use of a larger evaluation threshold (b_e) leads to more mini sessions enhanced. For example, when using $m = 1, C = 20$, and $b = 1$ to generate the improved structure, 70.80 and 87.57 percent of the mini sessions are enhanced for $b_e = 1$ and 3, respectively. In other words, 87.57 percent of the users who traversed more than three paths are facilitated in the improved structure whereas only 70.80 percent of the users who traversed more than one path are facilitated. This suggests that severely disoriented users, i.e., those who need to exert more effort to locate their targets, are more likely to benefit from the improved site structure than the less disoriented users, i.e., those who need to exert relatively less effort to locate their targets. This is a very appealing result because in practice the severely disoriented users are more likely to abandon the website as compared to those who are not.

TABLE 11
Comparison of Our Model with a Heuristic

Time threshold (t)	Path threshold (b)	No. of new links	No. of improved links	Difference = (Heuristic−Our model)/Our model				
				No. of new links	Avg. no of path reduced	Mini sessions enhanced		
						b_e=1	b_e=2	b_e=3
1 min.	1	9,308	1,829	22.15%	3.18%	3.73%	1.84%	0.16%
	2	2,917	679	112.12%	11.42%	14.88%	4.38%	2.80%
	3	1,186	267	141.62%	16.02%	23.65%	-0.56%	1.51%
2 min.	1	8,175	1,751	26.54%	3.63%	4.31%	2.43%	0.52%
	2	2,872	703	125.05%	10.65%	15.20%	3.30%	0.08%
	3	1,324	313	152.13%	11.77%	33.91%	3.99%	-0.67%
5 min.	1	7,529	1,705	29.94%	3.85%	4.38%	2.02%	1.09%
	2	2,784	681	143.14%	8.94%	13.31%	4.84%	2.95%
	3	1,342	319	187.44%	12.49%	18.06%	8.73%	2.16%

We also observe that as the path threshold for the training data (b) increases, more mini sessions are enhanced in the testing data. This is because considerably more links are added with a smaller value of b, as shown in Table 9. However, adding more links may not necessarily always lead to significantly superior results. For example, when setting $m = 0, C = 20$, and $b_e = 1$, while the use of $b = 1$ adds 5,794 new links, a number that is considerably larger than the 1,145 new links added when setting $b = 2$, the percentage of mini sessions enhanced increases only by 20.42 percent $(= 71.54\% - 51.12\%)$.

5.4 Comparison with a Heuristic

As mentioned earlier, although several transformation approaches have been developed to *restructure* a website, the objective of *improving* the user navigation on a website with *minimal* changes to its structure has not yet been examined. For this reason, we could not compare our model with other approaches proposed in the previous studies. In order to provide some comparisons, we compare the performance of our model with that of a heuristic instead and report the results in Table 11. The heuristic essentially identifies the set of relevant candidate links for each mini session and then randomly selects one link. We provide only the comparison results for the case where $m = 0$ and $C = 20$ are set in our model, and the comparisons of the heuristic with our model using other parameters are similar.

The first two columns of Table 11 list the parameters used for both the heuristic and our model. The number of new links and the number of improved links for the heuristic are shown in the third and fourth columns. The results of heuristic are obtained by performing the heuristic three times and then taking the average. The five columns under "Difference = (Heuristic-Our model)/Our model" report the comparisons between the heuristic and our model. Specifically, the first column lists how many more new links are introduced by the heuristic than our model; the next four columns show the increase/decrease in performance by the heuristic as compared with our model. For example, Table 11 shows that when the time threshold and the path threshold are set to $t = 1, b = 2$ (second row), the heuristic adds 112.12 percent more links than our model. Meanwhile, it only reduces the average number of paths per mini session by 11.42 percent and enhances 4.38 percent more mini sessions for $b_e = 2$ than our model.

The comparison shows that, under the same path threshold, the heuristic adds far more links than our model, and the difference increases significantly as the path threshold increases. For example, the heuristic adds 22.15 percent more new links for $t = 1$ and $b = 1$, but it adds more than twice as many new links as our model does (112.12 percent more links) when b increases to 2. The result indicates that although the heuristic adds considerably more links, it leads to only marginal improvements in performance. Sometimes, our model outperforms the heuristic with significantly fewer new links. For instance, when $t = 2, b = 3$ (sixth row), while the heuristic enhances 3.99 percent more mini sessions when the evaluation threshold is $b_e = 2$, it adds 152.13 percent more new links to the current structure than our model. Interestingly, when $b_e = 3$, our model even enhances more mini sessions with much fewer new links. Overall, this comparison shows that our model dominates this heuristic with respect to the number of new links added while achieving comparable or better results in facilitating user navigation.

6 Discussion

6.1 Mini Session and Target Identification

We employed the page-stay timeout heuristic to identify users' targets and to demarcate mini sessions. The intuition is that users spend more time on the target pages. Page-stay time is a common implicit measurement found to be a good indicator of page/document relevance to the user in a number of studies [44], [53], [54]. In the context of web usage mining, the page-stay timeout heuristic as well as other time-oriented heuristics are widely used for session identification [29], [42], [43], [19], [55], and are shown to be quite robust with respect to variations of the threshold values [56].

The identification of target pages and mini sessions can be affected by the choice of page-stay timeout threshold. Because it is generally very difficult to unerringly identify

mini sessions from anonymous user access data [19], we ran our experiments for different threshold values. Generally, increasing the threshold will result in fewer mini sessions with proportionally more mini sessions having a large number of paths while decreasing the threshold will have the opposite effect (see Table 7). In other words, increasing time threshold would decrease the number of relevant mini session for small path thresholds but could increase the number of relevant mini sessions for large path thresholds. As a result, we observed that as the time threshold increase, the number of new links decreases for $b = 1$ and 2, but increases for $b = 3$. While the results did change slightly, we showed that our model succeeded in finding the minimal number of links that can be used to improve user navigation substantially.

The time thresholds need to be properly selected based on the amount of information displayed in the webpages. In general, a larger time threshold is needed for websites consisting of information-rich pages than those whose pages contain less information. Other approaches can also help accurately identify target pages. As an example, since web designers have a good understanding of web contents, they can identify a list of important pages with high probabilities of being users' targets. Such information could greatly help improve the accuracy of target page identification.

6.2 Searching Sessions versus Browsing Sessions

While the majority of users typically have one or more goals and search for particular pieces of information when navigating a websites [57] ("searching" sessions), some users might simply browse for general information ("browsing" sessions). Though exact distinction between these two web surfing behaviors is often impossible by only looking at the anonymous user access data from weblogs, certain characteristics can help differentiate the two types of sessions. For example, some visits are clearly purposeless and finish abruptly at pages that cannot be target pages, so these sessions are more likely to be browsing sessions. To the best of our knowledge, there is no algorithm developed to distinguish between the two types of sessions and further investigation on this question is needed. While we did not explicitly separate searching sessions from browsing sessions, the preprocessing steps can help eliminate many purposeless browsing sessions. As a result, the final improved website structure mainly "shortens" searching sessions and also reduces purposeful browsing sessions.

6.3 Implications of This Research

This research contributes to the literature on improving web user navigation by examining this issue from a new and important angle. We have performed extensive experiments on both real and synthetic data sets to show that the model can be effectively solved and is highly scalable. In addition, the evaluation results confirm that users can indeed benefit from the improved structure after suggested changes are applied. There are several important implications from this research.

First, we demonstrate that it is possible to improve user navigation significantly with only few changes to the structure of a website using the proposed model. This is important because as time passes and the need for information changes, websites also need to be regularly maintained and improved. However, the current literature focuses on how to *restructure* a website and hence is not suitable for this purpose. Our research complements the literature by addressing this issue using a MP model. While the approaches proposed in previous studies are either heuristic-based or applicable only to small-sized website, we have shown that our model not only can be effectively solved for optimal solutions but also can scale up well, partly due to the fact that our model formulation can reduce the problem size considerably.

Second, we model the out-degree as a penalty (cost) term in the objective function, and this not only leads to more flexible website structures than modeling the out-degree as hard constraints, but also offers a good balance between minimizing changes to the website and reducing information overload to users (avoid clustering too many links in webpages). In particular, the experiment results indicate that when the penalty term is used, although more links are needed, the number of links inserted into pages with large out-degrees is also significantly reduced. This helps prevent further adding links to page with many links and helps the users locate the desired links in these pages more easily.

Third, we show that the use of a small path threshold will introduce far more changes to a website but may not always lead to significantly better outcomes for user navigation. This suggests that Webmasters need to carefully balance the tradeoff between desired improvements in the user navigation and the changes needed to accomplish the task when selecting appropriate path thresholds. This is particularly important when a website is improved on a regular basis.

Last, the proposed model can be exceptionally desirable to severely disoriented users. The evaluation results show that severely disoriented users, i.e., those who need to exert more effort to locate the targets, are more likely to benefit from the changes suggested by our model than the less disoriented users. This is an appealing result because the severely disoriented users are more likely to abandon their search for targets as compared to those who are not.

6.4 Choice of Parameter Values for the Model

6.4.1 Path Threshold

The path threshold represents the goal for user navigation that the improved structure should meet and can be obtained in several ways. First, it is possible to identify when visitors exit a website before reaching the targets from analysis of weblog files [40], [58]. Hence, examination of these sessions helps make a good estimation for the path thresholds. Second, surveying website visitors can help better understand users' expectations and make reasonable selections on the path threshold values. For example, if the majority of the surveyed visitors respond that they usually give up after traversing four paths, then the path threshold should be set to four or less. Third, firms like comScore and Nielsen have collected large amounts of client-side web usage data over a wide range of websites. Analyzing such data sets can also provide good insights into the selection of path threshold values for different types of websites.

Although using small path thresholds could result in more improvements in web user navigation in general, our experiments showed that the changes (costs) needed increase significantly as the path threshold decreases.

Sometimes, additional improvements in user navigation from using a small threshold are too little to justify the increased costs. Thus, Webmasters need to cautiously consider the tradeoff between desired improvements to user navigation and the changes needed when selecting appropriate values for path threshold. A cost benefit analysis that compares "benefits" and "costs" of using different path thresholds can be useful for this purpose. In the context of our problem, we can view the number of new links needed as the cost and the improvement on user navigation (this, for instance, can be measured as the average number of paths shortened by the improved structure) as the benefit. The benefit-cost ratio (BCR) that is used for the analysis of the cost effectiveness of different options can be expressed as (improvement on user navigation)/(number of new links). As a result, we can employ the evaluation procedures described in Section 5.3 to approximate the BCRs for different path thresholds and to find the path threshold value that is the most cost effective and provides a good tradeoff. In addition, our experiments showed that an extremely small path threshold (e.g., $b = 1$) could add many links and hence might not be a good choice if Webmasters plan to use our model to improve websites progressively.

6.4.2 Out-Degree Threshold

Webpages can be generally classified into two categories [29]: index pages and content pages. An index page is designed to help users better navigate and could include many links, while a content page contains information users are interested in and should not have many links. Thus, the out-degree threshold for a page is highly dependent on the purpose of the page and the website. Typically, the out-degree threshold for index pages should be larger than that for content pages. For instance, out-degree thresholds are set to 30 and 10 for index and content pages, respectively, in the experiments in [29]. Since out-degree thresholds are context dependent and organization dependent, behavioral and experimental studies that examine the optimal out-degree threshold for different settings are needed. In general, the out-degree threshold could be set at a small value when most webpages have relatively few links, and as new links are added, the threshold can be gradually increased. Note that since our model does not impose hard constraints on the out-degrees for pages in the improved structure, it is less affected by the choices of out-degree thresholds as compared to those in the literature.

6.4.3 Multiplier for the Penalty Term

As shown in Section 5, the use of the penalty term can prevent the model from adding new links to pages that already have many links. This helps keep the information load low for user at the cost of inserting more new links into other pages with small out-degrees. Generally, if a website have both pages with small out-degrees and pages with very large out-degrees, then it is reasonable to use a large multiplier (m) to avoid clustering too many links in a page. If the out-degrees are relatively small for all pages, then it could be more appropriate to use a relatively small multiplier to minimize the total number of new links added. When our model is used for website maintenance, a small multiplier could be used in the beginning when out-degrees are generally small for most pages, and as new links are inserted, a larger multiplier is needed to prevent adding extra links to pages that already have many links. In general, we suggest not using a very large multiplier for the penalty term, because the benefit from the reduced information load by using the penalty term was observed to improve only very slightly as the multiplier increases in the experiments. In particular, as shown in Table 8, increasing m from 0 to 1 would prevent far more links from adding to pages with large out-degrees (i.e., reduce far more "excessive" links) than increasing m from 1 to 5, while the numbers of new links needed were similar in the two cases.

6.5 Evaluation Procedure

We used simulations to approximate the real usage and to evaluate how the user navigation could be enhanced in the improved website structure. The use of simulation for website usability evaluation is very popular and has been widely used in modeling users' choices in web navigation and usability test [34], [36], [37], [59], [60]. However, simulation studies often have to make simplifying assumptions in order to simulate real-life scenarios, posing questions on the generalizability of the results. In the context of our simulation approach, we assume that users would find their target pages effectively through a new/improved link if it exists. In practice, certain criteria related to the visual design of web interfaces need to be followed in order to effectively apply the suggested changes to a website. We note that there exist an abundant literature on both webpage design [6], [8], [12], [31], [59], [61] and hyperlink design [34], [49], [50], [62], [63], [64]. Though we did not explicitly consider design issues in this paper, we do assume that Webmasters follow the guidelines and suggestions from such studies when creating and editing links and designing webpages. Consequently, in the simulation approach used for user navigation evaluation, we assume that new links are carefully designed and existing links are appropriately edited. In addition, they should also be placed in proper places for users to easily locate. Thus, these links should provide users with accurate knowledge on the contents on the other end of a link and help them make correct selections.

Because of the assumption made for the new and improved links, the claimed benefit can be interpreted as the upper bound and optimal benefit of our model. However, we would like to claim that improved and newly added links could guide users to find their target pages more efficiently to some extent. This is because: 1) our method establishes efficient paths to target pages that were not available in the website structure before optimization, and 2) our method suggests improving links that would lead to users' target pages efficiently but missed by users (since they did not know what these links would lead to), so that more efficient navigation can be facilitated.

Since our evaluation is simulation based, a usability study involving real users may help strengthen the results of our study and deserves further investigation. However, we note that such usability studies are generally more expensive and time consuming in the context of website evaluation [60], and hence are usually conducted on small-sized websites [5]. In contrast, simulation can be easily implemented, quickly performed for various parameter settings, and tested on a large scale. Thus, the simulation

studies in our paper complement usability studies by offering its own distinct advantages.

7 Conclusions

In this paper, we have proposed a mathematical programming model to improve the navigation effectiveness of a website while minimizing changes to its current structure, a critical issue that has not been examined in the literature. Our model is particularly appropriate for informational websites whose contents are relatively stable over time. It *improves* a website rather than *reorganizes* it and hence is suitable for website maintenance on a progressive basis. The tests on a real website showed that our model could provide significant improvements to user navigation by adding only few new links. Optimal solutions were quickly obtained, suggesting that the model is very effective to real-world websites. In addition, we have tested the MP model with a number of synthetic data sets that are much larger than the largest data set considered in related studies as well as the real data set. The MP model was observed to scale up very well, optimally solving large-sized problems in a few seconds in most cases on a desktop PC.

To validate the performance of our model, we have defined two metrics and used them to evaluate the improved website using simulations. Our results confirmed that the improved structures indeed greatly facilitated user navigation. In addition, we found an appealing result that heavily disoriented users, i.e., those with a higher probability to abandon the website, are more likely to benefit from the improved structure than the less disoriented users. Experiment results also revealed that while using small path thresholds could result in better outcomes, it would also add significantly more new links. Thus, Webmasters need to carefully balance the tradeoff between desired improvements to the user navigation and the number of new links needed to accomplish the task when selecting appropriate path thresholds. Since no prior study has examined the same objective as ours, we compared our model with a heuristic instead. The comparison showed that our model could achieve comparable or better improvements than the heuristic with considerably fewer new links.

The paper can be extended in several directions in addition to those mentioned in Section 6. For example, techniques that can accurately identify users' targets are critical to our model and future studies may focus on developing such techniques. As another example, our model has a constraint for out-degree threshold, which is motivated by cognitive reasons. The model could be further improved by incorporating additional constraints that can be identified using data mining methods [65]. For instance, if data mining methods find that most users access the finance and sports pages together, then this information can be used to construct an additional constraint.

Acknowledgments

The authors would like to thank the editors and the anonymous reviewers for their insightful comments and helpful suggestions, which have resulted in substantial improvements to this work.

References

[1] Pingdom, "Internet 2009 in Numbers," http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/, 2010.

[2] J. Grau, "US Retail e-Commerce: Slower But Still Steady Growth," http://www.emarketer.com/Report.aspx?code=emarketer_2000492, 2008.

[3] Internetretailer, "Web Tech Spending Static-But High-for the Busiest E-Commerce Sites," http://www.internetretailer.com/dailyNews.asp?id = 23440, 2007.

[4] D. Dhyani, W.K. Ng, and S.S. Bhowmick, "A Survey of Web Metrics," *ACM Computing Surveys,* vol. 34, no. 4, pp. 469-503, 2002.

[5] X. Fang and C. Holsapple, "An Empirical Study of Web Site Navigation Structures' Impacts on Web Site Usability," *Decision Support Systems,* vol. 43, no. 2, pp. 476-491, 2007.

[6] J. Lazar, *Web Usability: A User-Centered Design Approach.* Addison Wesley, 2006.

[7] D.F. Galletta, R. Henry, S. McCoy, and P. Polak, "When the Wait Isn't So Bad: The Interacting Effects of Website Delay, Familiarity, and Breadth," *Information Systems Research,* vol. 17, no. 1, pp. 20-37, 2006.

[8] J. Palmer, "Web Site Usability, Design, and Performance Metrics," *Information Systems Research,* vol. 13, no. 2, pp. 151-167, 2002.

[9] V. McKinney, K. Yoon, and F. Zahedi, "The Measurement of Web-Customer Satisfaction: An Expectation and Disconfirmation Approach," *Information Systems Research,* vol. 13, no. 3, pp. 296-315, 2002.

[10] T. Nakayama, H. Kato, and Y. Yamane, "Discovering the Gap between Web Site Designers' Expectations and Users' Behavior," *Computer Networks,* vol. 33, pp. 811-822, 2000.

[11] M. Perkowitz and O. Etzioni, "Towards Adaptive Web Sites: Conceptual Framework and Case Study," *Artificial Intelligence,* vol. 118, pp. 245-275, 2000.

[12] J. Lazar, *User-Centered Web Development.* Jones and Bartlett Publishers, 2001.

[13] Y. Yang, Y. Cao, Z. Nie, J. Zhou, and J. Wen, "Closing the Loop in Webpage Understanding," *IEEE Trans. Knowledge and Data Eng.,* vol. 22, no. 5, pp. 639-650, May 2010.

[14] J. Hou and Y. Zhang, "Effectively Finding Relevant Web Pages from Linkage Information," *IEEE Trans. Knowledge and Data Eng.,* vol. 15, no. 4, pp. 940-951, July/Aug. 2003.

[15] H. Kao, J. Ho, and M. Chen, "WISDOM: Web Intrapage Informative Structure Mining Based on Document Object Model," *IEEE Trans. Knowledge and Data Eng.,* vol. 17, no. 5, pp. 614-627, May 2005.

[16] H. Kao, S. Lin, J. Ho, and M. Chen, "Mining Web Informative Structures and Contents Based on Entropy Analysis," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 1, pp. 41-55, Jan. 2004.

[17] C. Kim and K. Shim, "TEXT: Automatic Template Extraction from Heterogeneous Web Pages," *IEEE Trans. Knowledge and Data Eng.,* vol. 23, no. 4, pp. 612-626, Apr. 2011.

[18] M. Kilfoil et al., "Toward an Adaptive Web: The State of the Art and Science," *Proc. Comm. Network and Services Research Conf.,* pp. 119-130, 2003.

[19] R. Gupta, A. Bagchi, and S. Sarkar, "Improving Linkage of Web Pages," *INFORMS J. Computing,* vol. 19, no. 1, pp. 127-136, 2007.

[20] C.C. Lin, "Optimal Web Site Reorganization Considering Information Overload and Search Depth," *European J. Operational Research,* vol. 173, no. 3, pp. 839-848, 2006.

[21] M. Eirinaki and M. Vazirgiannis, "Web Mining for Web Personalization," *ACM Trans. Internet Technology,* vol. 3, no. 1, pp. 1-27, 2003.

[22] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization," *Data Mining and Knowledge Discovery,* vol. 6, no. 1, pp. 61-82, 2002.

[23] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic Personalization Based on Web Usage Mining," *Comm. ACM,* vol. 43, no. 8, pp. 142-151, 2000.

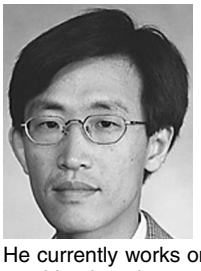
[24] B. Mobasher, R. Cooley, and J. Srivastava, "Creating Adaptive Web Sites through Usage-Based Clustering of URLs," *Proc. Workshop Knowledge and Data Eng. Exchange,* 1999.

[25] W. Yan, M. Jacobsen, H. Garcia-Molina, and U. Dayal, "From User Access Patterns to Dynamic Hypertext Linking," *Computer Networks and ISDN Systems,* vol. 28, nos. 7-11, pp. 1007-1014, May 1996.

[26] M. Nakagawa and B. Mobasher, "A Hybrid Web Personalization Model Based on Site Connectivity," *Proc. Web Knowledge Discovery Data Mining Workshop,* pp. 59-70, 2003.

[27] B. Mobasher, "Data Mining for Personalization," *The Adaptive Web: Methods and Strategies of Web Personalization,* A. Kobsa, W. Nejdl, P. Brusilovsky, eds., vol. 4321, pp. 90-135, Springer-Verlag, 2007.
[28] C.C. Lin and L. Tseng, "Website Reorganization Using an Ant Colony System," *Expert Systems with Applications,* vol. 37, no. 12, pp. 7598-7605, 2010.
[29] Y. Fu, M.Y. Shih, M. Creado, and C. Ju, "Reorganizing Web Sites Based on User Access Patterns," *Intelligent Systems in Accounting, Finance and Management,* vol. 11, no. 1, pp. 39-53, 2002.
[30] M.D. Marsico and S. Levialdi, "Evaluating Web Sites: Exploiting User's Expectations," *Int'l J. Human-Computer Studies,* vol. 60, no. 3, pp. 381-416, 2004.
[31] J. Palmer, "Designing for Web Site Usability," *Computer,* vol. 35, no. 7, pp. 102-103, June 2002.
[32] J. Liu, S. Zhang, and J. Yang, "Characterizing Web Usage Regularities with Information Foraging Agents," *IEEE Trans. Knowledge and Data Eng.,* vol. 16, no. 5, pp. 566-584, May 2004.
[33] P. Pirolli and S.K. Card, "Information Foraging," *Psychological Rev.,* vol. 106, no. 4, pp. 643-675, 1999.
[34] E.H. Chi, P. Pirolli, and J. Pitkow, "The Scent of a Site: A System for Analyzing and Predicting Information Scent, Usage, and Usability of a Web Site," *Proc. SIGCHI Conf. Human Factors in Computing Systems,* pp. 161-168, 2000.
[35] C. Olston and E.H. Chi, "ScentTrails: Integrating Browsing and Searching on the Web," *ACM Trans. Computer-Human Interaction,* vol. 10, no. 3, pp. 177-197, 2003.
[36] E.H. Chi, P. Pirolli, K. Chen, and J. Pitkow, "Using Information Scent to Model User Information Needs and Actions on the Web," *Proc. ACM Conf. Human Factors in Computing Systems,* pp. 490-497, 2001.
[37] W.T. Fu and P. Pirolli, "SNIF-ACT: A Cognitive Model of User Navigation on the World Wide Web," *Human-Computer Interaction,* vol. 22, pp. 355-412, 2007.
[38] W. Willett, J. Heer, and M. Agrawala, "Scented Widgets: Improving Navigation Cues with Embedded Visualizations," *IEEE Trans. Visualization and Computer Graphics,* vol. 13, no. 6, pp. 1129-1136, Nov. 2007.
[39] X. Xie, H. Liu, W. Ma, and H. Zhang, "Browsing Large Pictures under Limited Display Sizes," *IEEE Trans. Multimedia,* vol. 8, no. 4, pp. 707-715, Aug. 2006.
[40] R. Srikant and Y. Yang, "Mining Web Logs to Improve Web Site Organization," *Proc. 10th Int'l Conf. World Wide Web,* pp. 430-437, 2001.
[41] M.S. Chen, J.S. Park, and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns," *IEEE Trans. Knowledge and Data Eng.,* vol. 10, no. 2, pp. 209-221, Mar./Apr. 1998.
[42] R. Cooley, B. Mobasher, and J. Srivastava, "Data Preparation for Mining World Wide Web Browsing Patterns," *Knowledge and Information Systems,* vol. 1, pp. 1-27, 1999.
[43] M. Spiliopoulou, B. Mobasher, B. Berendt, and M. Nakagawa, "A Framework for the Evaluation of Session Reconstruction Heuristics in Web-Usage Analysis," *INFORMS J. Computing,* vol. 15, no. 2, pp. 171-190, 2003.
[44] M. Morita and Y. Shinoda, "Information Filtering Based on User Behavior Analysis and Best Match Text Retrieval," *Proc. 17th Ann. Int'l ACMSIGIR Conf. Research and Development in Information Retrieval,* pp. 272-281, 1994.
[45] Tealeaf, "The Two Waves of Online Abandonment: The 2007 Harris Interactive Survey of Online Customer Behavior," http://www.tealeaf.com/downloads/tealeaf-executivebrief_Harris2007.pdf, 2007.
[46] J. Song and F.M. Zahedi, "A Theoretical Approach to Web Design in E-Commerce: A Belief Reinforcement Model," *Management Science,* vol. 51, no. 8, pp. 1219-1235, 2006.
[47] V. Venkatesh and R. Agarwal, "From Visitors into Customers: A Usability-Centric Perspective on Purchase Behavior in Electronic Channels," *Management Science,* vol. 52, no. 3, pp. 367-382, 2006.
[48] W. Lin, S. Alvarez, and C. Ruiz, "Efficient Adaptive-Support Association Rule Mining for Recommender Systems," *Data Mining and Knowledge Discovery,* vol. 6, pp. 83-105, 2002.
[49] K. Larson and M. Czerwinski, "Web Page Design: Implications of Memory, Structure and Scent for Information Retrieval," *Proc. SIGCHI Conf. Human Factors in Computing Systems,* pp. 25-32, 1998.
[50] M. Otter and H. Johnson, "Lost in Hyperspace: Metrics and Mental Models," *Interacting with Computers,* vol. 13, pp. 1-40, 2000.
[51] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman, 1979.
[52] T. Boutell, "WWW FAQs: How Many Websites Are There?" http://www.boutell.com/newfaq/misc/sizeofweb.html, 2007.
[53] D.W. Oard and J. Kim, "Modeling Information Content Using Observable Behavior," *Proc. ASIST Ann. Meeting,* pp. 481-488, 2001.
[54] M. Claypool, P. Le, M. Waseda, and D. Brown, "Implicit Interest Indicators," *Proc. Sixth Int'l Conf. Intelligent User Interfaces,* pp. 33-40, 2001.
[55] H. Liu and V. Keselj, "Combined Mining of Web Server Logs and Web Contents for Classifying User Navigation Patterns and Predicting Users' Future Requests," *Data and Knowledge Eng.,* vol. 61, no. 2, pp. 304-330, 2007.
[56] B. Berendt, B. Mobasher, M. Spiliopoulou, and J. Wiltshire, "Measuring the Accuracy of Sessionizers for Web Usage Analysis," *Proc. Web Mining Workshop First SIAM Int'l Conf. Data Mining,* pp. 7-14, 2001.
[57] J. Morrison, P. Pirolli, and S.K. Card, "A Taxonomic Analysis of What World Wide Web Activities Significantly Impact People's Decisions and Actions," *Proc. ACM Conf. Human Factors in Computing Systems,* pp. 163-164, 2001.
[58] R.E. Bucklin and C. Sismeir, "A Model of Website Browsing Behavior Estimated on Clickstream Data," *J. Marketing Research,* vol. 40, no. 3, pp. 249-267, 2003.
[59] M. Ivory, R.R. Sinha, and M. Hearst, "Empirically Validated Web Page Design Metrics," *Proc. SIGCHI Conf. Human Factors in Computing Systems,* pp. 53-60, 2001.
[60] C.S. Miller and R.W. Remington, "Modeling Information Navigation: Implications for Information Architecture," *Human Computer Interaction,* vol. 19, pp. 225-271, 2004.
[61] M. Ivory and M. Hearst, "Improving Web Site Design," *IEEE Internet Computing,* vol. 6, no. 2, pp. 56-63, Mar. 2002.
[62] J. Nielsen, *Designing Web Usability: The Practice of Simplicity.* New Riders Publishing, 2000.
[63] J.M. Spool, T. Scanlon, W. Schroeder, C. Snyder, and T. DeAngelo, *Web Site Usability: A Designer's Guide.* Morgan Kaufman, 1999.
[64] J. Kim and B. Yoo, "Toward the Optimal Link Structure of the Cyber Shopping Mall," *Int'l J. Human-Computer Studies,* vol. 52, no. 3, pp. 531-551, 2000.
[65] B. Padmanabhan and A. Tuzhilin, "On the Use of Optimization for Data Mining: Theoretical Interactions and eCRM Opportunities," *Management Science,* vol. 49, no. 10, pp. 1327-1343, 2003.

Min Chen received the PhD degree in management science from the School of Management, The University of Texas at Dallas, in 2011. He is an assistant professor in the School of Management at George Mason University. His research interests include data mining, machine learning, optimization models, and economics of information systems.



Young U. Ryu received the PhD degree in management science and information systems from the McCombs Graduate School of Business, The University of Texas at Austin. He is an associate professor of information systems in the School of Management, The University of Texas at Dallas. He has studied applications of data mining and artificial intelligence technologies and decision science methods to the modeling and analysis of information systems. He currently works on data separation as information classification and machine learning, computer security, and social network analysis.