A Parameterized Approach to Spam-Resilient Link Analysis of the Web

James Caverlee, *Member*, *IEEE*, Steve Webb, *Member*, *IEEE*,
Ling Liu, *Senior Member*, *IEEE*, and William B. Rouse, *Fellow*, *IEEE*

Abstract—Link-based analysis of the Web provides the basis for many important applications—like Web search, Web-based data mining, and Web page categorization—that bring order to the massive amount of distributed Web content. Due to the overwhelming reliance on these important applications, there is a rise in efforts to manipulate (or spam) the link structure of the Web. In this manuscript, we present a parameterized framework for link analysis of the Web that promotes spam resilience through a source-centric view of the Web. We provide a rigorous study of the set of critical parameters that can impact source-centric link analysis and propose the novel notion of influence throttling for countering the influence of link-based manipulation. Through formal analysis and a large-scale experimental study, we show how different parameter settings may impact the time complexity, stability, and spam resilience of Web link analysis. Concretely, we find that the source-centric model supports more effective and robust rankings in comparison with existing Web algorithms such as PageRank.

Index Terms—Internet search, information search and retrieval, information storage and retrieval, information technology and systems, distributed systems, systems and software, Web search, general, Web-based services, online information services.

1 Introduction

The Web is arguably the most massive and successful distributed computing application today. Millions of Web servers support the autonomous sharing of billions of Web pages. From its earliest days, the Web has been the subject of intense focus for organizing, sorting, and understanding its massive amount of data. One of the most popular and effective Web analysis approaches is link-based analysis for considering the number and nature of hyperlink relationships among Web pages. Link analysis powers many critical Web applications, including Web crawling, Web search and ranking, Web-based data mining, and Web page categorization.

Since Web link analysis plays a central role in so many critical Web applications, Web spammers spend a considerable effort on manipulating (or spamming) the link structure of the Web to undermine the link-based algorithms that drive these applications (like the PageRank algorithm for Web page ranking). This manipulation is a serious problem, and more and more incidents of Web spam are observed, experienced, and reported [1], [2], [3]. In this manuscript, we focus on three prominent types of link-based vulnerabilities we have identified in Web ranking systems. These vulnerabilities corrupt link-based ranking algorithms like HITS [4] and PageRank [5] by making it appear that a reputable page is endorsing the Web spam target pages. The three types are the following:

- *Hijacking*. Spammers insert links into legitimate pages that point to a spammer-controlled page. There are a number of avenues for hijacking legitimate pages, including the insertion of spam links into public message boards, openly editable wikis, and Web logs.
- *Honeypots*. Spammers create quality sites to collect legitimate links that are then passed on to spammer-controlled pages. Rather than risking exposure by hijacking a link, a *honeypot* induces links so that it can pass its accumulated authority to a spam target page.
- *Collusion*. A spammer constructs specialized linking structures across one or more spammer-controlled pages. In a *link exchange*, multiple spammers trade links to pool their collective resources for mutual page promotion. Another example is a *link farm*, in which a large number of colluding pages point to a single target page.

To defend against these important types of link-based vulnerabilities, we promote a source-centric view of the Web and a novel notion of influence throttling for countering the influence of spammers to manipulate link-based algorithms. Most link-based algorithms to date have been based on the most basic Web element—Web pages. Page-based link analysis relies on a fundamentally flat view of the Web, in which all pages are treated as equal nodes in a Web graph. In contrast, a number of recent studies have noted a strong Web link structure, in which links display strong source-centric locality in terms of domains and hosts (e.g., [6] and [7]). This

- *J. Caverlee is with the Department of Computer Science, Texas A&M University, 3112 TAMU, College Station, TX 77843-3112. E-mail: caverlee@cs.tamu.edu.*
- *S. Webb is with the Purewire, 14 Piedmont Center NE, Suite 850, Atlanta, GA 30305. E-mail: swebb@purewire.com.*
- *L. Liu is with the College of Computing, Georgia Institute of Technology, 266 Ferst Dr., Atlanta, GA 30332-0765. E-mail: lingliu@cc.gatech.edu.*
- *W.B. Rouse is with the Tennenbaum Institute, Georgia Institute of Technology, 760 Spring Street, NW, Atlanta, GA 30332-0210. E-mail: bill.rouse@ti.gatech.edu.*

Manuscript received 20 Sept. 2008; revised 21 Aug. 2008; accepted 26 Sept. 2008; published online 7 Oct. 2008.
Recommended for acceptance by Y. Pan.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2007-09-0327.
Digital Object Identifier no. 10.1109/TPDS.2008.227.

link locality naturally suggests the importance of source-centric link analysis (SLA). Complementary to the page-based view, the source-centric view relies on a hierarchical abstraction of the flat page-level view.

Research on SLA has shown some initial success; however, most studies over the past years have focused exclusively on a single goal—improving the efficiency of page-based ranking algorithms (e.g., [8], [7], and [9]). All of the approaches have explored only a fraction of the parameter space, leaving many important questions unanswered. We argue that fully exploring SLA can have a profound impact on link-based algorithms and our general understanding of the Web.

In this manuscript, we introduce a parameterized framework to support the systematic study and evaluation of SLA of the Web with an emphasis on spam resilience. We address the following three important open questions:

- What are the most important parameters for guiding SLA?
- How should these parameters be set to achieve the specific objectives of the link analysis?
- What impact do the parameter settings have on the effectiveness of the analysis? Do certain parameter settings conflict or correlate with the objectives?

To this end, we identify a set of critical parameters that can impact the effectiveness of SLA, including source size, the presence of self-links, and different source-citation link weighting schemes (e.g., uniform, link count, and source consensus). We provide a rigorous study on the set of critical parameters, especially with respect to the above three open questions. We conduct a large-scale comparative study of different parameter settings of SLA over four large Web data sets against multiple and possibly competing objectives—spam resilience, time complexity, and stability—and we show how the parameters should be tuned to ensure efficient, stable, and robust Web ranking. Analytically, we provide a formal discussion on the effectiveness of SLA against link-based attacks. We show how source-centric analysis provides strong resistance to manipulation and raises the cost of rank manipulation to a Web spammer.

The rest of this manuscript is organized as follows: We present source-centric analysis in Section 2 and describe several critical parameters impacting the quality of source-centric analysis. In Section 3, we examine SLA in the context of Web ranking. In Section 4, we analyze the spam-resilience properties of source-centric link-based ranking. We evaluate the approach in Section 5, describe related work in Section 6, and wrap up in Section 7.

2 Source-Centric Link Analysis

To counter link-based vulnerabilities, we study the Web from a source-centric point of view. In this complementary hierarchical view to the traditional page graph, pages are grouped into logical collections of Web pages that we call sources. In this section, we identify important parameters for guiding **SLA**, including how sources are defined, and discuss how these parameters impact the effectiveness of link analysis. SLA relies on a source view of the Web. Just as the page graph $\mathcal{G}_\mathcal{P} = \langle \mathcal{P}, \mathcal{L}_\mathcal{P} \rangle$ models the Web as a directed graph where the nodes of the graph correspond to Web pages $\mathcal{P}$ and the set of directed edges $\mathcal{L}_\mathcal{P}$ correspond to hyperlinks between pages, the *source graph* has nodes that correspond to sources and edges that denote the linkage between sources. We use the term *source edge* to refer to the notion of source-centric citation. A source s_1 has a source edge to another source s_2 if one page in s_1 has a hyperlink to a page in s_2. We call s_1 the originating source and s_2 the target source.

A source graph can consist of multiple levels of source hierarchy; that is, a page may belong to a source that belongs to a larger source, and so on. In the rest of this manuscript, we shall require that each page in the page graph belong to one and only one source in the source graph, meaning that the hierarchical view of the Web consists of two levels: a page level and a source level. Hence, a Web source graph $\mathcal{G}_\mathcal{S} = \langle \mathcal{S}, \mathcal{L}_\mathcal{S} \rangle$ is a directed graph where the nodes of the graph correspond to Web sources in $\mathcal{S}$ and the set of directed edges $\mathcal{L}_\mathcal{S}$ corresponds to source edges, as described above.

2.1 Overview

Given the source view of the Web, we next discuss the choice of parameters for guiding SLA. The choice of parameters and their specific settings are greatly impacted by the particular application of the link analysis (e.g., ranking, categorization, and clustering). In this manuscript, we focus our parameter discussion primarily on the objective of **spam resilience**. Spam resilience may come at a price, however, and so, we also consider two additional objectives that are fundamental across link analysis applications: **time complexity** for understanding how to leverage the higher source-abstraction level to improve the time complexity relative to page-based approaches and **stability** for understanding SLA in the face of the Web's constant evolution.

We identify five key parameters that impact the objectives of SLA:

- *Source definition* (Γ). The first and most important parameter is the source definition. The determination of how sources are organized is at the heart of SLA, and all other parameter settings are entirely dependent on the source definition.
- *Source-centric citation* (Θ). The second parameter we consider is the nature of the citation-based association between sources. We study the presence and strength of the linkage arrangements from one source to another.
- *Source size* (Ξ). Since sources may vary greatly in the number of constituent pages, the third parameter we study is the source size and how this nonlinkage information may be directly incorporated into the analysis.
- *Influence throttling* (Λ). The fourth parameter considers the degree to which a source's influence in the underlying application should be limited or throttled. Determining the level of influence throttling may require information external to the link structure of the source graph.

TABLE 1
Summary of Web Data Sets (in Millions)

Dataset	Pages	Links
WB2001	118.1	992.8
UK2002	18.5	292.2
IT2004	41.3	1,135.7
UK2006	80.6	2465.8

TABLE 2
Fraction of Page Links

Dataset	Intra-TLD	Intra-Domain	Intra-Host	Intra-Directory
WB2001	97.9%	95.5%	94.1%	62.7%
UK2002	100.0%	94.6%	92.3%	66.9%
IT2004	100.0%	91.0%	90.8%	67.9%
UK2006	100.0%	98.0%	96.0%	72.4%

- *Application-specific parameters* (Υ). Finally, there may be some additional application-specific parameters that are necessary, e.g., the number of iterations to run a ranking algorithm until sufficient convergence.

We describe SLA in terms of an application and a specific objective and as a combination of these five parameters: $SLA_{<app,obj>}(\Gamma;\Theta;\Xi;\Lambda;\Upsilon)$. In the following sections, we discuss the first four of these important parameters, present some of their possible settings, and provide insight into how best these parameters may be assigned based on the ultimate objectives of the link analysis. We examine the fifth parameter in the context of Web ranking in Section 3. We find that a careful approach to these parameters is necessary to ensure high-quality results across objectives, especially with respect to spam resilience.

2.2 Parameter 1: Source Definition

How does the source definition impact the quality of SLA with respect to the three objectives? Clearly, the determination of how sources are organized should have a profound impact on the quality and value of SLA. Previous studies have noted the importance of source definition, including some definitions based on functional properties and others based on link-based properties of the sources (see, e.g., [10] and [11]).

To understand the importance of source definition, we consider five different approaches—in the first, we treat each page as a unique source, meaning that the source view of the Web corresponds directly to the page view; in the second, we disregard all page relationships and randomly assign pages to sources. The other approaches rely on the link locality of the Web and assign pages based on their administrative organization—by domain, host, or directory.

To illustrate the locality-based linking phenomenon on the Web, we consider four large real-world Web data sets (see Table 1). The first data set—**WB2001**—was collected by the Stanford WebBase project and includes pages from a wide variety of top-level domains (TLDs). The second data set—**UK2002**—is derived from a 2002 crawl of the `.uk` TLD by UbiCrawler [12]. The third data set—**IT2004**—is derived from a 2004 crawl of the `.it` TLD, again by UbiCrawler. The fourth data set—**UK2006**—is derived from a 2006 crawl of `.uk`; note that this more recent data set is significantly larger than the UK2002 Web graph.[1]

In Table 2, we report four classes of links over these four data sets. We report the fraction of all links that point from pages in one domain to pages in the *same* domain (intradomain links), the fraction that point from pages in one host to pages in the *same* host (intrahost links), and the fraction that point from pages in one directory to pages in the *same* directory or lower in the directory hierarchy (intradirectory links). Since the WB2001 data set includes pages from many domains, we also report the fraction of pages that point from pages in one TLD to pages in the *same* TLD.

These statistics consistently show that the Web exhibits a strong locality-based link structure. Given this phenomenon, it is natural to assign pages to sources based on one of these administrative organizations. Hence, we study five different settings for the source definition parameter Γ—by domain, by host, and by directory, as well as the extremes of by page and by random assignment.

As we shall see in Section 5, the analysis quality depends heavily on the presence of link locality and the source definition. We find that a lack of locality results in poor time complexity but that even moderate locality ($\sim$65 percent) leads to good time complexity and stability results that are comparable with source definitions with extremely high locality.

The source definition provides a first step toward mitigating the influence of a Web spammer. In the ideal scenario, all of the pages under the control of a Web spammer would be mapped to a single source (and all legitimate pages would be mapped to their appropriate source as well), meaning that collusion among Web spammers could be muted entirely by discounting the links within each source. In practice, spammers can never be perfectly identified, and they can still rely on hijacking and honeypots to collect links from legitimate pages. Hence, the next parameter—source-centric citation—can provide another layer of defense against link-based manipulation.

2.3 Parameter 2: Source-Centric Citation

Unlike the straightforward notion of linkage in the page graph, source edges are derived from the page edges in the underlying page graph. Different page edges often carry different significance with respect to the sources involved. Careful design that takes these factors into account is critical, and so, the second parameter we study is the nature and strength of source-centric citation. Several previous studies have identified this parameter as an important one, but there has been little systematic study on how the choice of source-centric citation impacts link analysis (see, e.g., [13], [7], and [14]).

Given the directed source graph $\mathcal{G}_\mathcal{S} = \langle \mathcal{S}, \mathcal{L}_\mathcal{S} \rangle$, our goal is to understand the source-centric citation in terms of the appropriate edge weights for the set of directed edges $\mathcal{L}_\mathcal{S}$. Let $w(s_i, s_j)$ denote the weight assigned to the source edge $(s_i, s_j) \in \mathcal{L}_\mathcal{S}$. We consider source-centric citation as a scalar value in the range [0, 1], where the outgoing edge

1. All four data sets are available at http://webgraph-data.dsi.unimi.it/.

weights for any source sum to 1. In cases where the normalization is not explicit, we will require the normalization of the raw edge weights. We consider six edge weighting schemes:

1. **Uniform**. This is the simplest case where all source edges pointing out from an originating source are treated equally. This *uniform (u)* weighting is defined as

$$w_u(s_i, s_j) = \frac{1}{\sum_{s_k \in \mathcal{S}} \mathcal{I}[(s_i, s_k) \in \mathcal{L}_\mathcal{S}]},$$

where the indicator function $\mathcal{I}(\cdot)$ is 1 if the argument to the function is true and 0 otherwise.

Since each node in the source graph is an aggregation of one or more pages, treating each source edge equally may not properly capture the citation strength between two sources. With this in mind, we next introduce three source edge weighting schemes that are based on the hyperlink information encoded in the page graph $\mathcal{G}_\mathcal{P} = \langle \mathcal{P}, \mathcal{L}_\mathcal{P} \rangle$.

2. **Link count**. The link count scheme assigns edge weights based on the count of *page links* between pages that belong to sources. Such an edge weighting is effective when we would like to reward sources that have strong linkage at the page level. The *link count (lc)* weighting is

$$w_{lc}(s_l, s_j) = \sum_{p_i | s(p_i) = s_i} \left(\sum_{p_j | s(p_j) = s_j} \mathcal{I}\big[(p_i, p_j) \in \mathcal{L}_\mathcal{P}\big] \right),$$

where the source to which page p_i belongs is denoted by $s(p_i)$.

3. **Source consensus**. This edge weighting scheme counts the number of *unique pages* within an originating source that point to a target source. We may wish to differentiate between the case where a single page within the originating source is contributing all n links to the target and the case where there are n pages in the originating source and each has a single link to the target. We capture this notion of *source consensus (sc)* in the following edge weighting definition:

$$w_{sc}(s_i, s_j) = \sum_{p_i | s(p_i) = s_i} \left(\bigvee_{p_j | s(p_j) = s_j} \mathcal{I}\big[(p_i, p_j) \in \mathcal{L}_\mathcal{P}\big] \right).$$

4. **Target diffusion**. In contrast to how many pages in the originating source are responsible for the page links between sources, another factor that is of interest when evaluating the source-citation strength is the number of different target pages that are pointed to by the originating source. The *target diffusion (td)* weighting is defined as

$$w_{td}(s_i, s_j) = \sum_{p_j | s(p_j) = s_j} \left(\bigvee_{p_i | s(p_i) = s_i} \mathcal{I}\big[(p_i, p_j) \in \mathcal{L}_\mathcal{P}\big] \right).$$

In addition to these purely link-based approaches, we also consider two approaches that rely on both the page links and the *quality* of the pages that provide the linking, where we denote page p_i's quality score by $q(p_i)$. Quality could be measured using the PageRank score for the page or a simple heuristic like the page's relative depth in the directory tree.

5. **Quality-weighted link count**. This edge weighting scheme directly integrates the page quality score into the *link count* weighting scheme. Let (q) denote the use of a page quality metric. We define the quality-weighted link count scheme as follows:

$$w_{lc(q)}(s_i, s_j) = \sum_{p_i | s(p_i) = s_i} \left(\sum_{p_j | s(p_j) = s_j} q(p_i) \cdot \mathcal{I}\big[(p_i, p_j) \in \mathcal{L}_\mathcal{P}\big] \right).$$

6. **Quality-weighted source consensus.** Similarly, we can integrate the page quality score into the *source consensus* edge weighting scheme to produce the quality-weighted source consensus edge weighting scheme:

$$w_{sc(q)}(s_i, s_j) = \sum_{p_i | s(p_i) = s_i} q(p_i) \cdot \left(\bigvee_{p_j | s(p_j) = s_j} \mathcal{I}\big[(p_i, p_j) \in \mathcal{L}_P\big] \right).$$

There is not a natural quality-weighted extension to the *target diffusion* edge weighting scheme since it is not focused on which page in the source is providing the forward linkage.

From a spam-resilience point of view, the source consensus edge weighting schemes place the burden on the hijacker (or honeypot) to capture *many* pages within a legitimate source to exert any influence over the spam target pages. Hijacking a few pages in source i will have little impact over the source-level influence flow to a spammer source j; that is, $w(s_i, s_j)$ is less subject to manipulation in the presence of many other pages within a source, since it is aggregated over the link characteristics of all pages in the source.

Another factor that can influence source-centric citation is whether we take into account self-edges. Given a particular edge weighting scheme, there may be some applications that require self-edges, while others do not. For example, in a ranking context, a self-edge may be interpreted as a self-vote by the source, meaning that the source could manipulate its own rank. In the case where self-edges are eliminated, we will require the edge weight $w(s_i, s_i) = 0$ for all $s_i \in \mathcal{S}$. On the other hand, it may be reasonable to include self-edges since the locality-based structure of Web links indicates a strong degree of association between a source and itself.

Hence, we shall consider 12 different settings for the source citation parameter Θ—the looped and loopless versions of the six association strength edge weighting schemes. We find that some edge weighting schemes are extremely vulnerable to spam manipulation, while others

are much less vulnerable. In terms of stability, we find that self-edges have a very strong impact.

2.4 Parameter 3: Source Size

Since sources may vary greatly in size, from a source of a single page to a source encompassing millions of pages, what is the impact of source size on the underlying objectives of SLA? For many applications, it may be reasonable to distinguish between sources based on the per-source size discrepancy. The importance of source size has been noted previously (see, e.g., [13] and [7]), but there has been little systematic study of its impact on SLA.

The source size is one example of nonlinkage information that can be incorporated into the link analysis. Of course, there could be other nonlink information of interest (like source topic or source trustworthiness), but in this manuscript, we shall restrict our examination to the source size. The parameter Ξ considers two options—the size in pages of each source s_i (denoted by $|s_i|$) and no size information. As we shall see in Section 5, the source size is a very important parameter for the stability of the algorithm but results in the least satisfactory spam resilience. In our experiments, we further explore this fundamental tension.

2.5 Parameter 4: Influence Throttling

The fourth parameter is concerned with selectively limiting or throttling the influence of certain sources based on external knowledge. The source view of the Web and the careful selection the other source-centric parameters can provide a foundation toward mitigating the influence of link-based manipulation, but there are still open vulnerabilities. For example, a spammer may control pages in multiple colluding sources, meaning that the spammer can construct a linking arrangement to ensure any arbitrary edge weight between colluding sources.

As a result, we next consider the final parameter of source-centric analysis for managing the impact of spammer-controlled links—*influence throttling*—so that a spammer cannot take unfair advantage of the underlying application, even in the presence of large-scale link manipulation. For each source $s_i \in \mathcal{S}$, we associate a throttling factor $\kappa_i \in [0,1]$. We refer to this $|\mathcal{S}|$-length vector $\boldsymbol{\kappa}$ as the *throttling vector*. Many factors may impact the specific choice of $\boldsymbol{\kappa}$, including the size of the Web data set, the number of pages considered, the link density, and other link characteristics. In the rest of this manuscript, we focus on one alternative for determining $\boldsymbol{\kappa}$ using the notion of *spam proximity*. We consider two settings for the influence throttling parameter Λ—in one case, we extract influence throttling factors based on spam proximity; in the other, we apply no influence throttling at all. Our goal in this manuscript is to provide an initial understanding of influence throttling; we anticipate further study of the optimal setting in future work.

3 Applying SLA to Web Ranking

The parameters introduced in the previous section can be combined in a number of ways to achieve a particular objective with respect to a link-based application (e.g., ranking and clustering). To more fully examine SLA, we select one application area—Web ranking—and examine the parameter settings with respect to the three objectives—spam resilience, time complexity, and stability. Source-centric ranking has intuitive appeal since many users may be interested in identifying highly ranked sources of information (e.g., CNN or ESPN) rather than specific pages.

Here, we adopt a ranking approach that is similar in spirit to the "random surfer" model often used to describe PageRank but adapted to SLA. Just as PageRank provides a single global authority score to each page on the Web based on a random walk over the linkage structure of the entire Web, the source-centric ranking approach (SLA_{Rank}) can be used to rank all sources. In general, a source will be ranked highly if many other high-ranking sources point to it. We denote source s_i's authority score as σ_i, where $\sigma_i > \sigma_j$ indicates that the ith source is more important than the jth source. We write the authority score for all sources using the vector notation $\boldsymbol{\sigma}$, where all $|\mathcal{S}|$ sources are assigned a score.

The random walk over the source graph proceeds as follows: for each source $s \in \mathcal{S}$

- with probability α, the random source walker follows one of the source edges of source s, and
- with probability $1-\alpha$, the random source walker teleports to a randomly selected source.

We refer to the first option as the *edge following factor* and the second option as the *teleportation factor*. Associated with the *edge following factor* is an $|\mathcal{S}| \times |\mathcal{S}|$ transition matrix $\mathbf{T}$, where the ijth entry indicates the probability that the random source walker will navigate from source s_i to source s_j. Associated with the *teleportation factor* is an $|\mathcal{S}|$-length teleportation probability distribution $\mathbf{c}$, where c_i indicates the probability that the random walker will teleport to source s_i. Such a random walk may be written as

$$\boldsymbol{\sigma}^T = \alpha \cdot \boldsymbol{\sigma}^T \cdot \mathbf{T} + (1-\alpha) \cdot \mathbf{c}^T. \quad (1)$$

Given the source-centric ranking model (SLA_{Rank}), we next address two questions: 1) How do the SLA parameters map to the Web ranking context? 2) How do we evaluate the objectives of link analysis in the context of Web ranking?

3.1 Mapping Parameters

All five parameters—source definition (Γ), source-centric citation (Θ), source size (Ξ), influence throttling (Λ), and the application-specific parameters (Υ)—impact Web ranking. Clearly, the source definition is critically important since it determines the fundamental unit of ranking. The source-centric citation is necessary to construct the transition matrix $\mathbf{T}$ according to the edge weights determined by Θ, that is, $T_{ij} = w(s_i, s_j)$. The source size parameter can be used to guide the teleportation factor—that is, $c_i = |s_i| / \sum_{j=1}^{|\mathcal{S}|} |s_j|$—which intuitively captures the behavior of a random surfer being more likely to jump to large sources. Alternatively, the source size can be disregarded, so the teleportation factors defaults to a uniform distribution: $c_i = 1/|\mathcal{S}|$.

For Web ranking, there are two application-specific parameters—the mixing parameter α and the convergence criterion for terminating the algorithm. For the influence

throttling parameter Λ, we augment the original source graph $\mathcal{G}_{\mathcal{S}} = \langle \mathcal{S}, \mathcal{L}_{\mathcal{S}} \rangle$ to require that all sources have a self-edge, regardless of the characteristics of the underlying page graph, i.e., $\forall s_i \in \mathcal{S}, (s_i, s_i) \in \mathcal{L}_{\mathcal{S}}$ holds. Including self-edges in the source graph is a sharp departure from the classic PageRank perspective and may initially seem counterintuitive—since it allows a source to have a direct influence over its own rank—but we will see how it is a critical feature of adding spam resilience to SLA.

For each source $s_i \in \mathcal{S}$, we associate the throttling factor $\kappa_i \in [0, 1]$ such that the self-edge weight $w(s_i, s_i) \geq \kappa_i$. By requiring a source to direct some minimum amount of influence (κ_i) on itself, we throttle the influence it can pass along to other sources. In the extreme, a source's influence is completely throttled when $\kappa_i = 1$, meaning that all edges to other sources are completely ignored (and hence, the throttled source's influence on other sources is diminished). Conversely, a source's influence is not throttled at all when $\kappa_i = 0$. Based on the throttling vector $\boldsymbol{\kappa}$, we can construct a new influence-throttled transition matrix $\mathbf{T}'$ where the transition probabilities are

$$T'_{ij} = \begin{cases} \kappa_i, & \text{if } T_{ij} < \kappa_i \text{ and } i = j, \\ \frac{T_{ij}}{\sum_{i \neq k} T_{ik}} \cdot (1 - \kappa_i), & \text{if } T_{ij} < \kappa_i \text{ and } i \neq j, \\ T_{ij}, & \text{otherwise.} \end{cases}$$

For a source that does not meet its minimum throttling threshold (i.e., $T_{ii} < \kappa_i$), the self-edge weight in the transformed transition matrix is tuned upward (i.e., $T'_{ii} = \kappa_i$), and the remaining edge weights are rescaled such that $\sum_{i \neq j} T'_{ij} = 1 - \kappa_i$.

Unlike the original PageRank-style random source walker, the influence-throttled random source walker can be interpreted as a *selective random walk*, whereby a random walker arrives at a source and flips a source-specific biased coin. The random walk proceeds as follows: For source $s_i \in \mathcal{S}$

- with probability $\alpha\kappa_i$, the random walker follows source s_i's self-edge,
- with probability $\alpha(1 - \kappa_i)$, the random walker follows one of source s_i's out edges, and
- with probability $1 - \alpha$, the random walker teleports to a randomly selected source.

Note that this influence-throttled random walk could also be applied to the original PageRank formulation. Indeed, similar approaches for page-based ranking have been considered elsewhere, including [15] and [16]. However, the observed link locality phenomenon naturally couples with the notion of influence throttling by revising the weight of the self-directed links in the source graph, and so, we believe that the source-based version is intuitively more appealing.

3.2 Spam-Proximity Throttling

Determining the level of influence throttling for each source is an important component. In this section, we discuss one alternative for determining $\boldsymbol{\kappa}$ using the notion of *spam proximity*. The key insight is to tune κ_i higher for known spam sources and those sources that link to known spam sources (e.g., through hijacking, honeypots, or collusion). Spam proximity is intended to reflect the "closeness" of a source to other spam sources in the source graph. A source is "close" to spam sources if it is a spam source itself, if it directly links to a spam source, or if the sources it directly links to a link to spam sources, and so on (recursively).

Given a small seed of known spam sources, we adopt a propagation approach that relies on an inverse PageRank-style model to assign a spam proximity value to *every* source in the Web graph, similar to the BadRank [17] approach for assigning in essence a "negative" PageRank value to spam. First, we reverse the links in the original source graph $\mathcal{G}_{\mathcal{S}} = \langle \mathcal{S}, \mathcal{L}_{\mathcal{S}} \rangle$ so that we have a new *inverted* source graph $\mathcal{G}'_{\mathcal{S}} = \langle \mathcal{S}, \mathcal{L}'_{\mathcal{S}} \rangle$, where the source edge $(s_i, s_j) \in \mathcal{L}_{\mathcal{S}} \Rightarrow (s_j, s_i) \in \mathcal{L}'_{\mathcal{S}}$. A source that is *pointed to* by many other sources in the original graph will now itself point to those sources in the inverted graph. The spam-proximity vector σ_s is the solution to the linear system:

$$\boldsymbol{\sigma_s}^T = \beta \cdot \boldsymbol{\sigma_s}^T \cdot \mathbf{U} + (1 - \beta) \cdot \mathbf{d}^T, \tag{2}$$

where $\mathbf{U}$ is the transition matrix associated with the reversed source graph $\mathcal{G}'_{\mathcal{S}}$, β is a mixing factor, and $\mathbf{d}$ is a static score vector derived from the set of prelabeled spam sources. An element in $\mathbf{d}$ is 1 if the corresponding source has been labeled as spam and 0 otherwise. By including the prelabeled spam sources, the stationary distribution σ_s is a spam-proximity vector biased toward spam and sources "close" to spam.

Based on the stationary distribution, we can assign a throttling value to each source such that sources that are "closer" to spam sources are throttled more than more distant sources. Naturally, there are a number of possible ways to assign these throttling values. In this manuscript, we choose a simple heuristic such that sources with a spam-proximity score in the top-k are throttled completely (i.e., $\kappa_i = 1$ for all s_i in the top-k) and all other sources are not throttled at all.

3.3 Evaluating Objectives

In addition to the spam-resilience, time complexity, and stability objectives, we also consider a fourth objective that is specific to Web ranking—approximating PageRank.

- **Spam resilience**. To evaluate the spam-resilience properties, we measure the impact of several spam scenarios in terms of the ranking impact on a target source: $SLA_{Rank;Spam}(\Gamma; \Theta; \Xi; \Lambda; \Upsilon)$.
- **Time complexity**. To measure time complexity, we examine the calculation efficiency of the source-centric ranking approach in terms of the time it takes to calculate each ranking vector:

$$SLA_{Rank;Time}(\Gamma; \Theta; \Xi; \Lambda; \Upsilon).$$

- **Stability**. We consider two flavors of stability. First, we evaluate the stability of the ranking algorithm as the Web graph evolves and new pages and sources are discovered. Second, we investigate the stability in terms of the similarity of

rankings induced by the various parameter settings: $SLA_{Rank;Stab}(\Gamma;\Theta;\Xi;\Lambda;\Upsilon)$.

- **Approximating PageRank**. Finally, we consider the ranking-specific objective of approximating the traditional global PageRank vector by combining the source-level ranking information with the per-source ranking information. Such approximation promises to speed the PageRank calculation considerably: $SLA_{Rank;Approx}(\Gamma;\Theta;\Xi;\Lambda;\Upsilon)$.

Several previous research efforts have considered a source-centric ranking calculation over groups of pages, including [18] and [11]. These approaches have had different ultimate objectives, and each approach has focused exclusively on a handful of parameter settings with respect to a single objective. The first approach sought to bootstrap the calculation of PageRank with an initial starting "guess" derived from a decomposition of the Web into a higher level block layer and a local level [7]. The second approach has focused on replacing the traditional PageRank vector with an alternative ranking approach by determining a page's authority as a combination of multiple disjoint levels of rank authority (e.g., [8], [19], [9], [20], and [21]); the traditional PageRank vector is never computed. The third approach decentralizes the computation for use in peer-to-peer networks (e.g., [14] and [22]).

Each of these previous approaches relies on only a few parameter settings in the context of a single objective and can be seen as a fairly limited exploration of the parameter space of SLA_{Rank}. For example, the BlockRank [7] and ServerRank [14] algorithms both consider host-level sources and a quality-weighted link count citation weight with self-edges and disregard source size. By considering the five source definition parameter settings, the 12 source-citation settings, and the two teleportation vectors, we examine 120 different parameter settings for source-centric ranking (SLA_{Rank}), which we evaluate over four distinct objectives. To the best of our knowledge, ours is the first study to consider such a large parameter space and in the context of multiple possibly competing objectives.

4 Spam-Resilience Analysis

In this section, we analyze the spam-resilience properties of the ranking model and compare it to PageRank. We consider a Web spammer whose goal is to maximize its influence over a single *target source* through the manipulation of links (both from within the source and from other sources), which corresponds to the vulnerabilities identified in the Introduction.[2]

We focus on two important spam techniques. These two techniques—link manipulation within a source (in Section 4.1) and cross-source link manipulation (in Section 4.2)—are fundamental building blocks that spammers can combine in sophisticated ways to achieve their goals (see, e.g., [26]). Effectively countering these fundamental techniques is important and informative for developing more customized antispam defenses.

2. The analysis in this section builds on previous studies of PageRank and its variations, e.g., [23], [24], and [25].

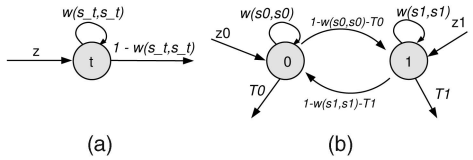

Fig. 1. What is the optimal source configuration?

4.1 Link Manipulation within a Source

We begin by studying link manipulation that is confined to a single source, which would correspond to collusive arrangements among spammer-controlled Web pages, like link farms and link exchanges. In the source view of the Web, all intrasource page links are reflected in a single self-edge to the source, and all page links to others sources are reflected in source edges to external sources.

How should the Web spammer configure the target source s_t to maximize its SLA_{Rank} score, which in turn will have the greatest impact on the target source's rank relative to all other sources? In Fig. 1a, we consider a generic source configuration for s_t. The target has a self-edge weight of $w(s_t, s_t)$, leaving $1 - w(s_t, s_t)$ for all source edges to external sources. Let z denote the aggregate incoming score to the target source from sources beyond the control of the Web spammer. Here, the Web spammer has direct influence over its own links (reflected in $w(s_t, s_t)$) but no influence over the incoming links from other sources. Recall (1); we can write the target source's score:

$$\sigma_t = \alpha z + \alpha \cdot w(s_t, s_t) \cdot \sigma_t + \frac{1-\alpha}{|\mathcal{S}|},$$

$$\sigma_t = \frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1 - \alpha \cdot w(s_t, s_t)},$$

which is maximized when $w(s_t, s_t) = 1$. The optimal configuration is for the source to *eliminate all out edges* and retain only a self-edge. Hence, the optimal σ_t^* is

$$\sigma_t^* = \frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha}. \tag{3}$$

Given that the target source has an initial throttling factor $\kappa < 1$ and that $w(s_t, s_t) = \kappa$, the next question to consider is by how much may a source improve its score by adopting a self-edge weight even higher than its throttling factor (i.e., by increasing $w(s_t, s_t)$ beyond the minimum κ throttling value). Examining the relative SLA_{Rank} score for s_t, we have

$$\frac{\sigma_t^*}{\sigma_t} = \frac{\frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha}}{\frac{\alpha z + \frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha\kappa}} = \frac{1-\alpha\kappa}{1-\alpha}.$$

For a source with an initial baseline throttling value of $\kappa = 0$, a source may increase its SLA_{Rank} score by $\frac{1}{1-\alpha}$ by increasing its $w(s_t, s_t)$ to 1. For typical values of α—from 0.80 to 0.90—this means that a source may increase its score from 5 to 10 times. For sources that are more throttled, there is less room for manipulation. In Fig. 2, we show, for increasing values of a baseline κ, the maximum factor

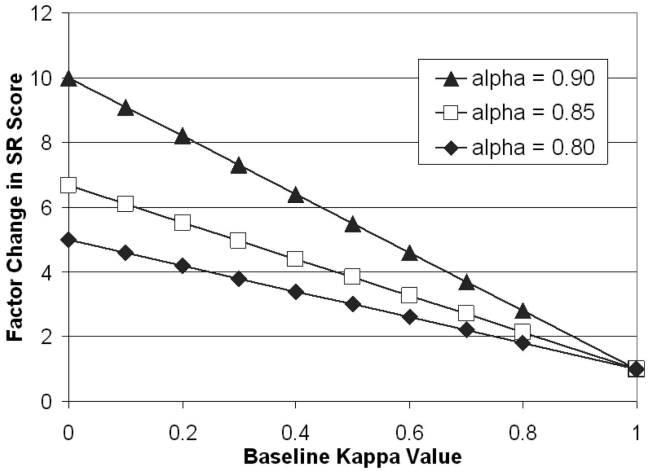

Fig. 2. Change in the spam-resilient SR score by tuning κ from a baseline value to 1.

change in the SLA_{Rank} score by tuning the κ value closer to 1. A highly throttled source may tune its SLA_{Rank} score upward by a factor of two for an initial $\kappa = 0.80$, a factor of 1.57 for $\kappa = 0.90$, and not at all for a fully throttled source.

By including self-edges in the source graph and the throttling factor κ, we allow a Web spammer some room for manipulating the score of its sources; however, the manipulation is for a *one-time* increase only, and it may be limited by tuning the κ throttling factor higher. No such limit is provided under PageRank, meaning that a Web spammer may arbitrarily increase the score of a series of target pages by a factor even larger than what we see for SLA_{Rank}.

4.2 Cross-Source Link Manipulation

We now study link manipulation across two or more sources, which corresponds to hijacking and honeypot scenarios, as well as collusive arrangements that span multiple sources. For this analysis, the spammer wishes to maximize the score for the single target source by manipulating the links available in one or more *colluding* sources.

In Fig. 1b, we show a generic source configuration for a single target source s_0 and a single colluding source s_1. We let θ_0 and θ_1 denote the edge weighting for each source to sources outside the sphere of influence of the Web spammer. Hence, source s_0 has $1 - w(s_0, s_0) - \theta_0$ edge weighting available for the edge to source s_1. The corresponding edge weight holds for the edge from s_1 to s_0. Let z_0 and z_1 denote the incoming score to each source, respectively, from other sources beyond the control of the Web spammer. Hence, we may describe the SLA_{Rank} for the two sources with a system of equations, where the Web spammer may manipulate $w(s_0, s_0)$, $w(s_1, s_1)$, θ_0, and θ_1:

$$\sigma_0 = \alpha z_0 + \alpha w(s_0, s_0)\sigma_0 + \frac{1-\alpha}{|\mathcal{S}|} + \alpha(1 - w(s_1, s_1) - \theta_1)\sigma_1,$$

$$\sigma_1 = \alpha z_1 + \alpha w(s_1, s_1)\sigma_1 + \frac{1-\alpha}{|\mathcal{S}|} + \alpha(1 - w(s_0, s_0) - \theta_0)\sigma_0.$$

Solving and taking the partial derivative with respect to the four parameters, we find that the optimal scenario for a Web spammer who wishes to maximize the SLA_{Rank} score for source s_0 is to set $\theta_0 = \theta_1 = 0$, meaning that there are no source edges to sources outside of the Web spammer's sphere of influence; $w(s_0, s_0) = 1$, meaning that the target source points only to itself and not at all to the colluding source; and $w(s_1, s_1) = 0$, meaning that the colluding source points only to the target source. With the κ_1 throttling factor requirement, this means that the best the colluding source can do is meet the minimum requirement $w(s_1, s_1) = \kappa_1$ and direct the rest $(1 - \kappa_1)$ to the target.

If we extend this analysis to consider x colluding sources (labeled $s_1, \ldots, s_x$) all in service to a single target source, then the system of equations is

$$\sigma_0 = \alpha z_0 + \alpha w(s_0, s_0)\sigma_0 + \frac{1-\alpha}{|\mathcal{S}|} + \alpha \sum_{i=1}^{x} (1 - w(s_i, s_i)) \frac{\alpha z_i + \frac{1-\alpha}{|\mathcal{S}|}}{1 - \alpha w(s_i, s_i)},$$

$$\sigma_i = \alpha z_i + \alpha w(s_i, s_i)\sigma_i + \frac{1-\alpha}{|\mathcal{S}|} + \alpha(1 - w(s_0, s_0) - \theta_0)\sigma_0.$$

The optimal configuration is for all colluding sources to set $\theta_i = 0$, meaning that there are no source edges from colluding sources to sources outside of the Web spammer's sphere of influence; $w(s_0, s_0) = 1$, meaning that the target source points only to itself and not at all to the colluding source; and $w(s_i, s_i) = \kappa_i$, meaning that the colluding source directs the minimum edge weight to itself and the remainder $(1 - \kappa_i)$ to the target source. Hence, each colluding source s_i contributes some $SLA_{Rank}\, \Delta_{\sigma_i}(\sigma_0)$ to the target s_0:

$$\Delta_{\sigma_i}(\sigma_0) = \frac{\alpha}{1-\alpha} \sum_{i=1}^{x} (1 - \kappa_i) \frac{\alpha z_i + \frac{1-\alpha}{|\mathcal{S}|}}{1 - \alpha\kappa_i}. \quad (4)$$

Clearly, tuning the κ throttling factor for each source closer to 1 (meaning that the majority of the colluding source's edge weight is directed to itself) results in a smaller change to the score of the target source. Hence, the introduction of the self-edge and the use of the throttling factor limit the impact of intersource link manipulation.

To further understand the importance of the κ throttling factor on muting the impact of a Web spammer across sources, we consider a scenario in which a Web spammer controls x colluding sources, each source has the same throttling factor of κ, and the sources are configured optimally (as described above). Now, suppose the throttling factor is raised to κ' for each source, meaning that each colluding source has less influence on the target source. How many sources x' are needed to achieve the same score as in the original case? That is, what impact does raising the throttling factor have on the Web spammer?

If we let $z_i = 0$, we may write the original Spam-Resilient SLA_{Rank} score with x colluding sources and an initial throttling factor κ, as well as the SLA_{Rank} score under the higher throttling factor (κ') scenario:

$$\sigma_0(x, \kappa) = \frac{\left(\frac{\alpha(1-\kappa)x}{1-\alpha\kappa} + 1\right)\frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha},$$

$$\sigma_0(x', \kappa') = \frac{\left(\frac{\alpha(1-\kappa')x'}{1-\alpha\kappa'} + 1\right)\frac{1-\alpha}{|\mathcal{S}|}}{1-\alpha}.$$

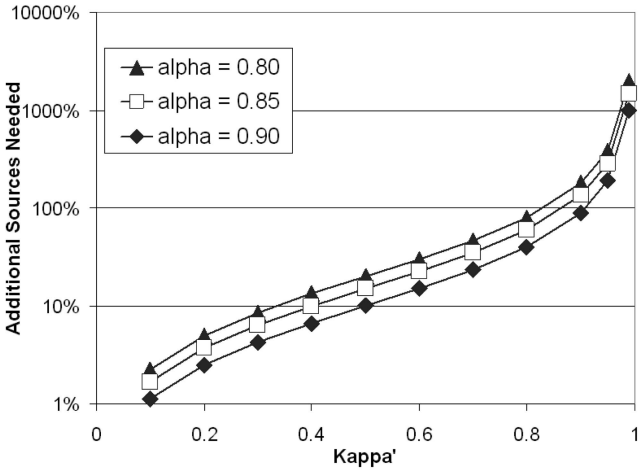

Fig. 3. Additional sources needed under κ' to equal the impact when $\kappa = 0$.

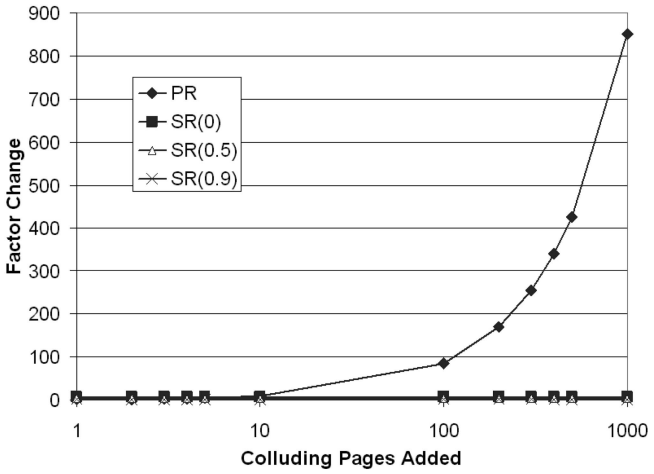

Fig. 4. Comparison with PageRank: Scenario 1.

Letting $\sigma_0(x, \kappa) = \sigma_0(x', \kappa')$, we may find a relationship between the number of original colluding sources x and the number of colluding sources x' necessary under the higher throttling factor:

$$\frac{x'}{x} = \frac{1 - \alpha\kappa'}{1 - \alpha\kappa} \cdot \frac{1 - \kappa}{1 - \kappa'}.$$

In Fig. 3, we plot the percentage of additional sources $(\frac{x'}{x} - 1)$ needed for a choice of κ' to equal the same influence on the score of the target page as that under an initial choice $\kappa = 0$. For example, when $\alpha = 0.85$ and $\kappa' = 0.6$, there are 23 percent more sources necessary to achieve the same score as in the case when $\kappa = 0$. When $\kappa' = 0.8$, the Web spammer needs to add 60 percent more sources to achieve the same influence; for $\kappa' = 0.9$, he needs 135 percent more sources; and for $\kappa' = 0.99$, he needs 1,485 percent more sources. Tuning the throttling factor higher considerably increases the cost of intersource manipulation.

4.3 Comparison with PageRank

Now that we have studied source-centric ranking and seen how influence throttling can be used to significantly increase the cost of manipulation to a Web spammer, we next compare SLA_{Rank} to PageRank. Since PageRank provides page-level rankings, we consider a Web spammer whose goal is to maximize its influence over a single *target page* within a target source. Extending the framework from the previous section, we consider three scenarios:

1. The target page and all colluding pages belong to the same source.
2. The target page belongs to one source, and all colluding pages belong to one colluding source.
3. The target page belongs to one source, and the colluding pages are distributed across many colluding sources.

For each scenario, the colluding pages are structured with a single link to the target page. We consider the impact of an increasing number of colluding pages (τ). Adopting a linear formulation of PageRank that is similar in spirit to (1), we may denote the PageRank score π_0 for the target page in terms of the PageRank due to pages outside of the sphere of influence of the Web spammer, the PageRank due to the teleportation component, and the PageRank due to the τ colluding pages:

$$\pi_0 = z + \frac{1 - \alpha}{|\mathcal{P}|} + \tau\alpha\frac{1 - \alpha}{|\mathcal{P}|},$$

where α refers to the teleportation probability, and $|\mathcal{P}|$ refers to the total number of pages in the page graph. The contribution of the τ colluding pages (where $\tau << |\mathcal{P}|$) to the overall PageRank score of the target page is

$$\Delta_\tau(\pi_0) = \tau\alpha\frac{1 - \alpha}{|\mathcal{P}|}.$$

For Scenario 1, the Web spammer configures the target source optimally (as we presented in (3)), meaning that the colluding pages' intrasource links to the target page have no impact on the SLA_{Rank} score (other than perhaps a one-time increase due to tuning the self-edge weight up from κ to 1). PageRank is extremely susceptible, as illustrated in Fig. 4, where the PageRank score (PR) of the target page jumps by a factor of nearly 100 with only 100 colluding pages.

For Scenario 2, the Web spammer adopts the optimal (worst case) two-source configuration discussed in the previous section. In this configuration, the target source points only to itself, and the colluding source that contains the colluding pages directs κ edge weight to itself and the rest to the target source. In Fig. 5, we see how PageRank is again extremely susceptible to such collusion, whereas the maximum influence over SLA_{Rank} is capped at ~2 times the original score for several values of κ. Since PageRank has no notion of a source, makes no effort to regulate the addition of new pages to the Web graph, and has no notion of influence throttling, all three spam scenarios under consideration will have the same extreme impact on the PageRank score of the target page.

In Scenario 3, the Web spammer adopts the optimal configuration for x colluding sources (as we established in the previous section). Fig. 6 plots the extreme impact on PageRank. As the influence throttling factor is tuned higher, the SLA_{Rank} score of the target source is less easily manipulated.

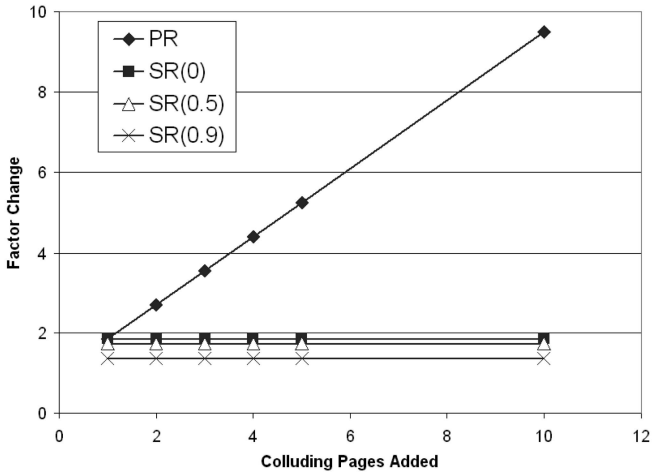

Fig. 5. Comparison with PageRank: Scenario 2.

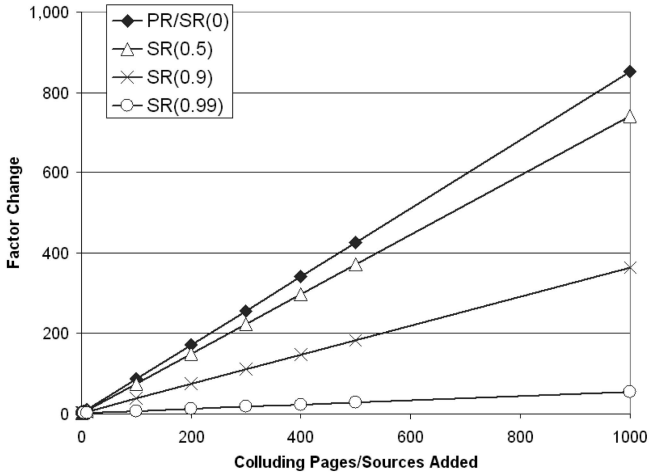

Fig. 6. Comparison with PageRank: Scenario 3.

5 Experimental Evaluation

In this section, we experimentally evaluate SLA in the context of Web ranking with respect to four objectives—spam resilience, time complexity, ranking stability, and approximating PageRank. Our spam-resilience evaluation is intended to confirm the analysis of the previous section over real Web data. Careful tuning of parameters is vital to ensure success over each objective. Some objectives cannot be maximized without negatively impacting other objectives. Note that the experimental validation focuses on several fundamental spamming scenarios over static snapshots of the Web; we anticipate revisiting dynamic Web models and more sophisticated scenarios in our future work.

5.1 Experimental Setup

All of our experimental evaluation is over the four Web data sets described in Section 2.2. For each data set, we extracted the domain, host, and directory information for each page URL and assigned pages to sources based on these characteristics. We also consider the extreme case when each page belongs to its own source (equivalent to the page graph described in Table 1). For the random source definition, we set the number of nodes in the graph to be the same as the number of hosts. In Table 3, we present summary information for each of the source graphs.

All of the ranking code was written in Java. The data management component was based on the WebGraph compression framework described in [27]. All experiments were run on a dual-processor Intel XEON at 2.8 GHz with 8-Gbyte of memory. We measured the convergence rate for all ranking calculations using the L2 distance of successive iterations of the Power Method. We terminated the ranking calculations once the L2 distance dropped below a threshold of 10e-9. As a baseline, we computed the PageRank vector (π) over each page graph using the parameters typically used in the literature (e.g., [5]), including a mixing parameter of 0.85, a uniform teleportation vector, and a uniform link following probability. For the quality-weighted edge weighting, we measure the quality of each page $q(p_i)$ using the page's PageRank score π_i. Although in practice, it may not be reasonable to use the PageRank score as a measure of quality since it is so expensive to calculate, we include these PageRank-weighted options here to understand their impact relative to the edge weighting schemes that do not require PageRank.

For compactness, we shall write a particular SLA_{Rank} parameter combination like $\mathcal{SR}(\mathbf{T}^*_U, \mathbf{c}_u)$, where the transition matrix $\mathbf{T}$ is appended with a subscript to indicate which source edge weighting scheme we use: $\mathbf{T}_U$, $\mathbf{T}_{LC}$, and so on. We shall append an asterisk to the transition matrix to indicate the inclusion of self-edges: $\mathbf{T}^*$. For the choice of teleportation vector $\mathbf{c}$, we consider the standard uniform vector $(\mathbf{c}_u)$ and the source-size-based vector $(\mathbf{c}_s)$.

5.2 Measures of Ranking Distance

We rely on two distance metrics for comparing ranking vectors. The Kendall Tau Distance Metric [28] is based solely on the relative ordering of the sources in two ranking vectors. In contrast, the Jensen-Shannon Divergence (JS-Divergence) [29] measures the distributional similarity of two vectors, meaning that it considers the magnitude of each source's authority score and not just the relative ordering.

Kendall Tau distance metric. This metric measures the relative ordering of two lists of ranked objects [28]. It is based on the original Kendall Tau correlation described in

TABLE 3
Source Graph Summary—By Source Definition

	Domain		Host		Dir		Rand		Page	
Dataset	Nodes	Links	Nodes	Links	Nodes	Links	Nodes	Links	Nodes	Links
WB2001	620k	10.5m	739k	12.4m	3,315k	24.7m	739k	955.4m	118.1m	992.8m
UK2002	81k	1.2m	98k	1.6m	360k	3.5m	98k	286.0m	18.5m	292.2m
IT2004	136k	2.7m	141k	2.8m	505k	8.6m	141k	1,069.3m	41.3m	1,135.7m
UK2006	81k	1.5m	95k	1.8m	638k	8.1m	95k	2,047.6m	80.6m	2,465.8m

TABLE 4
Wallclock Time (in Minutes) per Iteration

Dataset	Source Definition				
	Domain	Host	Dir	Rand	Page
WB2001	0.21	0.25	0.46	11.32	12.28
UK2002	0.02	0.03	0.07	2.45	2.76
IT2004	0.05	0.05	0.13	9.33	9.44
UK2006	0.02	0.03	0.10	16.63	19.18

[30] and provides a notion of how closely two lists rank the same set of objects (or Web sources in our case). The Kendall Tau Distance Metric takes values in the range [0, 1], where two rankings that are exactly the same have a distance of 0, and two rankings in the reverse order have a distance of 1. We rely on a variation of an $O(n \log n)$ version described in [31].

JS-Divergence. The JS-Divergence is a measure of the distributional similarity between two probability distributions [29]. It is based on the relative entropy measure (or KL-divergence), which measures the difference between two probability distributions p and q over an event space X: $KL(p,q) = \sum_{x \in X} p(x) \cdot \log(p(x)/q(x))$. If we let p be one of the ranking vectors $\boldsymbol{\sigma}$ and q be the other ranking vector $\boldsymbol{\sigma}'$, then we have $KL(\boldsymbol{\sigma}, \boldsymbol{\sigma}') = \sum_{i \in \mathcal{S}} \sigma_i \cdot \log(\sigma_i/\sigma_i')$. Intuitively, the KL-divergence indicates the inefficiency (in terms of wasted bits) of using the q distribution to encode the p distribution. Since the KL-divergence is not a true distance metric, the JS-Divergence has been developed to overcome this shortcoming, where

$$JS(\boldsymbol{\sigma}, \boldsymbol{\sigma}') = \phi_1 KL(\boldsymbol{\sigma}, \phi_1 \boldsymbol{\sigma} + \phi_2 \boldsymbol{\sigma}') + \phi_2 KL(\boldsymbol{\sigma}', \phi_1 \boldsymbol{\sigma} + \phi_2 \boldsymbol{\sigma}'),$$

where ϕ_1, $\phi_2 > 0$, and $\phi_1 + \phi_2 = 1$. In these experiments, we consider $\phi_1 = \phi_2 = 0.5$.

5.3 Objective-Driven Evaluation

We report the most significant results from a total of 480 different ranking vectors we computed by combining the five source definitions, the 12 source-citation edge weights, the two teleportation vectors, and the four data sets. For the 480 ranking vectors we analyze, we fix the mixing parameter α at the commonly adopted value of 0.85 used for PageRank (e.g., [5]).

5.4 Time Complexity

We begin by examining the ranking efficiency of the source-centric ranking approach in terms of the time it takes to calculate each ranking vector. The PageRank-style calculation scans the link file for each source graph multiple times until convergence.

Table 4 shows the average time per iteration to calculate the ranking vector over the five different source graphs for each of the data sets. We report the results for a ranking based on the uniform edge weight and a uniform teleportation vector: $\mathcal{SR}(\mathbf{T}_U, \mathbf{c}_u)$. These general per-iteration results also hold for the total time to reach the L2 stopping criterion. In our examination of the 480 different ranking vectors, we find that the source definition has the most significant impact on the calculation time, since the source definition directly impacts the size of the source graph. The choice of edge weights and teleportation vector has little discernable impact.

TABLE 5
Ranking Similarity: Parameter Settings

Shorthand	Version	Edge Weight	Self-Edges?	Telep. Factor
Baseline	$\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_u)$	LC	Yes	Uniform
NoL	$\mathcal{SR}(\mathbf{T}_{LC}, \mathbf{c}_u)$	LC	No	Uniform
Size	$\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_s)$	LC	Yes	Size
Uni	$\mathcal{SR}(\mathbf{T}^*_{U}, \mathbf{c}_u)$	U	Yes	Uniform
SC	$\mathcal{SR}(\mathbf{T}^*_{SC}, \mathbf{c}_u)$	SC	Yes	Uniform
TD	$\mathcal{SR}(\mathbf{T}^*_{TD}, \mathbf{c}_u)$	TD	Yes	Uniform
LC(q)	$\mathcal{SR}(\mathbf{T}^*_{LC(q)}, \mathbf{c}_u)$	LC(q)	Yes	Uniform
SC(q)	$\mathcal{SR}(\mathbf{T}^*_{SC(q)}, \mathbf{c}_u)$	SC(q)	Yes	Uniform

As we can see, the directory, host, and domain source definitions result in ranking computation that is one to two orders of magnitude faster than the page-based graph. Since PageRank over a Web graph of billions of nodes takes days, this improvement is important for source-centric ranking to compensate for PageRank's slow time to update. The random source definition performs poorly, even though there is the same number of nodes in the random graph and in the host graph. The key difference is that the random graph has no link locality structure and hence consists of nearly as many links as in the page graph. We conclude that link locality strongly impacts the degree of source graph size reduction and, hence, the ranking calculation time. Due to its poor performance, we shall drop the random source definition from the rest of our reported experimental results.

5.5 Stability—Ranking Similarity

We next explore the parameter space to investigate the stability in terms of the similarity of rankings induced by the various parameter settings. Due to its popularity in other works (e.g., [7] and [22]), we adopt a baseline ranking based on the link count edge weight with self-edges and a uniform teleportation vector, $\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_u)$, and report seven alternative ranking vectors computed by tweaking these baseline parameter settings. We consider a version without self-edges ($\mathcal{SR}(\mathbf{T}_{LC}, \mathbf{c}_u)$), a version including self-edges and the size-based teleportation component ($\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_s)$), and five additional versions using the other edge weighting schemes (e.g., $\mathcal{SR}(\mathbf{T}^*_{U}, \mathbf{c}_u)$), as shown in Table 5. We report the results for the host-based graph in this section; we see similar results across the directory and domain source definition settings.[3]

In Figs. 7 and 8, we compare the ranking vector resulting from the baseline parameter settings with the ranking vector resulting from each of these seven alternative parameter settings. The y-axis measures the distance between these alternative ranking vectors and the baseline configuration via the Kendall Tau Distance Metric and the JS-Divergence.

As we can see, the exclusion of self-edges (`NoL`) and the choice of teleportation vector (`Size`) are the two factors with the most significant impact on the resulting ranking vector in terms of ranking distance from the baseline setting. Hence, we

3. We also considered a version that used a teleportation component based on the sum of the PageRank scores of each source's constituent pages, but we find that such a version qualitatively behaves much like the size-based version.

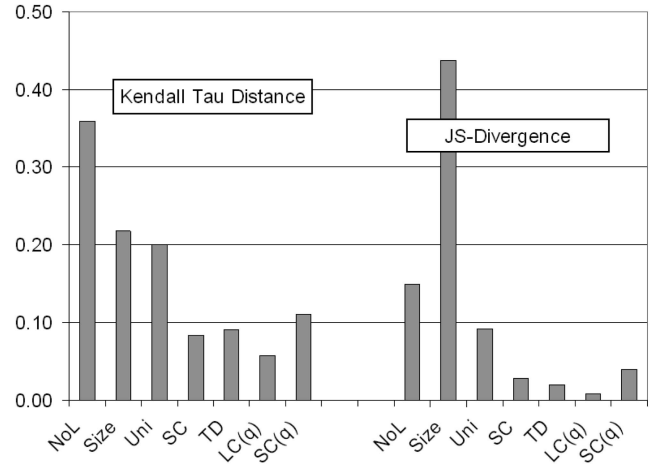

Fig. 7. Parameter tuning: ranking distance versus baseline configuration, IT2004.

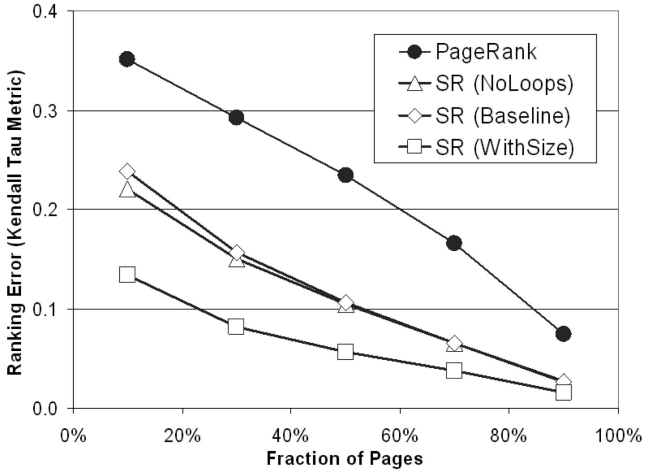

Fig. 9. Ranking stability, WB2001.

must be careful when setting these two critical parameters, since the resulting ranking vectors depend so heavily on them. The choice of edge weights has less impact, though we observe that the uniform edge weighting results in the most dissimilar ranking vector of all the edge weighting schemes. The uniform edge weighting scheme is a less intuitively satisfactory edge weighting scheme, and these results confirm this view. What is interesting here is that the source consensus, target diffusion, quality-weighted link count, and quality-weighted source consensus edge weights have a relatively minor impact on the resulting ranking vector versus the baseline link count version. We note that the quality-weighted link count deviates very little from the link count version, in spite of the incorporation of the expensive PageRank scores.

5.6 Stability—Link Evolution

We next evaluate the stability of SLA_{Rank} as the Web graph evolves for each of the four Web data sets. Since the source view of the Web provides an aggregate view over Web pages, we anticipate that domain, host, and directory-based rankings should be less subject to changes in the underlying page graph than page-based rankings. Our goal is to emulate the gradual discovery of Web pages, similar to how a Web crawler may incrementally discover new pages for ranking.

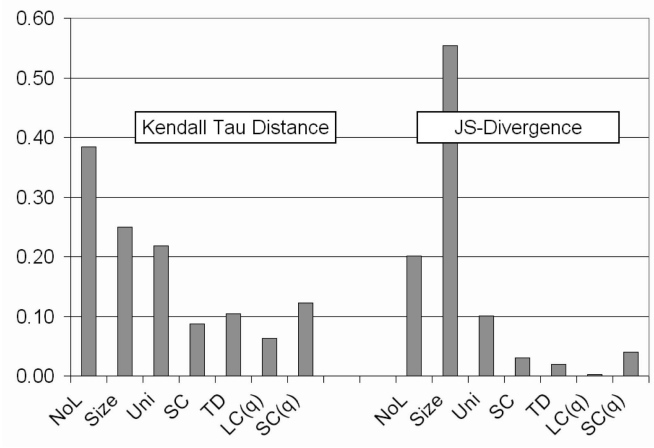

Fig. 8. Parameter tuning: ranking distance versus baseline configuration, UK2006.

For each data set, we randomly selected a fraction of the pages (10 percent, 30 percent, ...) and computed the standard PageRank vector over just this fraction of pages, yielding $\pi_{10\text{ percent}}$, $\pi_{30\text{ percent}}$, and so on. Additionally, we computed the ranking vector for the domain-, host-, and directory-based source graphs derived from the same fraction of all pages, yielding $\sigma_{10\text{ percent}}$, $\sigma_{30\text{ percent}}$, and so on. We then compared the relative page rankings for the pages in $\pi_{10\text{ percent}}, \pi_{30\text{ percent}}, \ldots$ to the relative rankings of the *exact same pages* in the PageRank vector for the full Web page graph. Similarly, we compared the relative source rankings for the sources in $\sigma_{10\text{ percent}}, \sigma_{30\text{ percent}}, \ldots$ to the relative rankings of the *exact same sources* in the ranking vector for the full Web source graph. We have also considered a third model, in which we first randomly sample sources (and then randomly sample pages from each source). Qualitatively, such an alternative Web page discovery model behaves similarly to the results reported here. To evaluate the stability, we rely on the Kendall Tau Distance Metric as a measure of ranking error.

In Fig. 9, we show the ranking error for the WB2001 data set for PageRank and for three representative parameter settings over the host-based source graph—the baseline version $\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_u)$, the loopless version $\mathcal{SR}(\mathbf{T}_{LC}, \mathbf{c}_u)$, and the size-based teleportation version $\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_s)$. Note that these are the three settings that resulted in the most different ranking vectors in our previous experiment. In all cases, the source-centric rankings display significantly less error relative to the rankings over the full Web graph than the PageRank rankings do, meaning that we can rely on source-centric rankings computed over an incomplete Web crawl with substantial confidence. Also, note that the size-based version is the most stable, and we find that this stability generally improves as the source definition becomes more inclusive (from page to directory to host to domain). Since the page and source ranking vectors are of different lengths, we additionally considered a similar stability analysis over just the top-100 and top-1,000 page

and source rankings. We relied on a variation of the Kendall Tau Distance Metric known as the Kendall Min Metric [28] for evaluating top-k ranked lists. These results further validate the source stability.

5.7 Approximating PageRank

As we have mentioned, one of the important goals of source-centric ranking is to approximate the traditional global PageRank vector by combining the source-level ranking information with per-source ranking information (the local PageRank scores). Such approximation promises to speed the PageRank calculation considerably. In this experiment, we aim to understand under what conditions source-centric ranking may be used to reasonably approximate PageRank. We decompose the global PageRank of a page into source and local components:

$$\pi(p_i) = \sigma(s_j) \cdot \pi(p_i|s_j), \tag{5}$$

where we denote the local PageRank score for page i in source j as $\pi(p_i|s_j)$. The local PageRank score is calculated based only on local knowledge (e.g., based on the linkage information of pages within the source), takes comparably little time relative to the full PageRank calculation, and forms a probability distribution (i.e., $\sum_{p_k \in s_j} \pi(p_k|s_j) = 1$).

For the PageRank decomposition to hold over all pages, ideally, we would have that the local PageRank scores $\pi(p_i|s_j)$ would exactly match the relative global distribution:

$$\pi(p_i|s_j) = \frac{\pi(p_i)}{\sum_{p_k \in s_j} \pi(p_k)}. \tag{6}$$

By replacing $\pi(p_i|s_j)$ in (5) with the right-hand side of (6), we find that the source-centric component $\sigma(s_j)$ should equal the sum of the global PageRanks of the constituent pages: $\sigma(s_j) = \sum_{p_k \in s_j} \pi(p_k)$.

To test how well the source-centric rankings may be used to approximate PageRank, we compare the rankings induced from various parameter settings with the rankings induced from ranking the sources by the sum of the global PageRanks of their constituent pages. In Fig. 10, we report the Kendall Tau Distance Metric and the JS-Divergence for three representative parameter settings—the baseline version $\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_u)$, the loopless version $\mathcal{SR}(\mathbf{T}_{LC}, \mathbf{c}_u)$, and the size-based teleportation version $\mathcal{SR}(\mathbf{T}^*_{LC}, \mathbf{c}_s)$—over the domain, host, and directory-based source definitions for the IT2004 data set. Similar results hold for the other two data sets.

The baseline parameter setting (used elsewhere, e.g., [7] and [22]) performs poorly and is not appropriate for approximating PageRank. Similarly, the loopless version, which disregards the strong evidence of link locality for setting edge weights, also performs poorly. Only the size-based version is highly correlated with the sum of the actual PageRank values for each source, meaning that source size and the presence of self-edges are critical for approximating PageRank.

5.8 Spam Resilience

Finally, we study the spam-resilience properties of SLA through two popular Web spam scenarios.

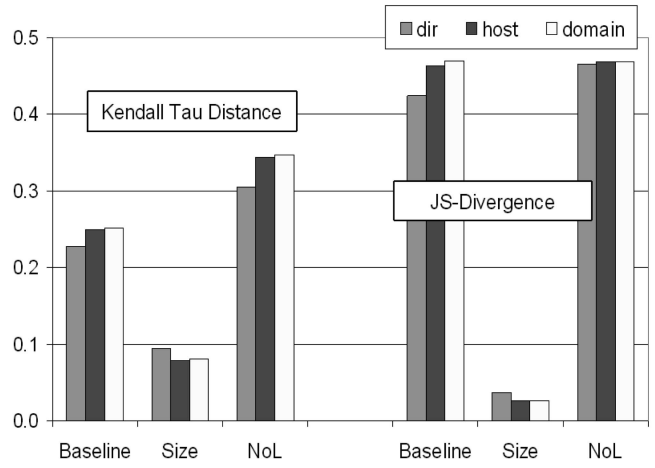

Fig. 10. Approximating PageRank, IT2004.

We first aim to validate the analysis in Section 4.1 by considering the impact of page-level manipulation *within* a single source. For this *intrasource manipulation*, we study the impact of a spammer who manipulates the pages internal to a target source for increasing the rank of a target page within the target source. We again consider the three representative parameter settings—the baseline, loopless, and size-based teleportation versions. For each version, we randomly selected five sources from the bottom 50 percent of all sources on the host graph (averaging in the 26th percentile of all sources for WB2001, the 26th percentile for UK2002, 32nd percentile for IT2004, and 24th percentile for UK2006). For each source, we randomly selected a target page within the source and created a link farm consisting of a single new spam page within the same source with a link to the target page. This is case A. For case B, we added 10 spam pages to the link farm within the source, each with a link to the target page. We repeated this setup for 100 pages (case C) and 1,000 pages (case D). For each case, we then constructed the new spammed page graph and host graph for each of the four Web data sets. We ran PageRank and the three source-centric versions for each of the four cases. In Figs. 11 and 12, we show the influence of the Web spammer in manipulating the rank of the target page and the rank of the target source through the average ranking percentile increase. For example in the WB2001 case, the PageRank of the target page jumped 80 percentile points under case C (from an average rank in the 19th percentile to the 99th), whereas the score of the target source jumped only 4 percentile points for the baseline version (from the 27th percentile to the 31st).

We first note the dramatic increase in PageRank for the target page across all four Web data sets, which confirms the analysis about the susceptibility of PageRank to rank manipulation. Although PageRank has typically been thought to provide fairly stable rankings (e.g., [32]), we can see how link-based manipulation has a profound impact, even in cases when the spammer expends very little effort (as in cases A and B). We note that the loopless source-centric version shows no change in rank value, since the addition of new intrasource links has no impact on the resulting source graph and ranking vector. The baseline version does increase

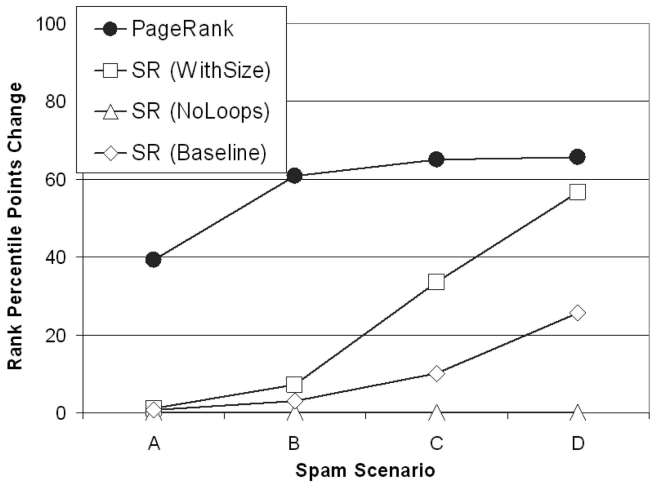

Fig. 11. Intrasource link farm, IT2004.

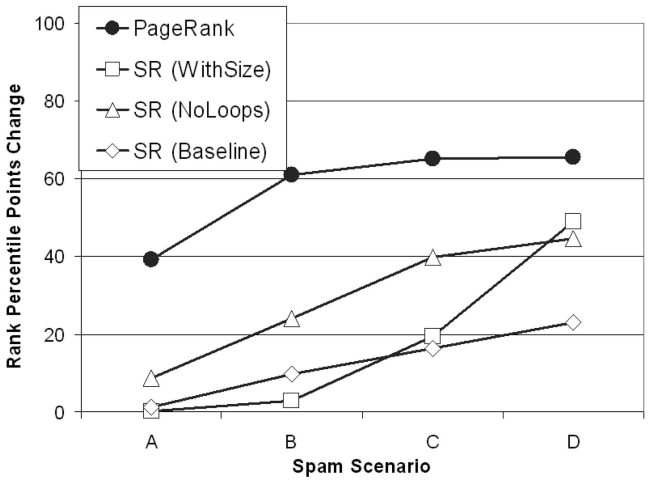

Fig. 13. Intersource link farm, IT2004.

some but not nearly as much as PageRank. Since the source is an aggregation of many pages, the weighting of the source edges is less susceptible to changes in the underlying page graph. In contrast, the size-based teleportation version is the most vulnerable to intrasource manipulation. In fact, under this scenario, a spammer need only add new pages (not links) to increase a source's score. The addition of so many new pages increases the size of the source, making it more attractive to the random walker who considers the size-based teleportation component. In fact, under this scenario, a spammer need only add new pages (not links) to increase the score of a source.

In the second Web spam scenario, we consider the impact of manipulation *across* sources, which corresponds to the analysis in Section 4.2. For this scenario, the spam links are added to pages in a colluding source that point to the target page in a different source. We paired the randomly selected target sources from the previous experiment with a randomly selected colluding source, again from the bottom 50 percent of all sources. For each pair, we added a single spam page to the colluding source with a single link to the randomly selected target page within the target source. This is case A. We repeated this setup for 10 pages (case B), 100 pages (case C), and 1,000 pages (case D). For each case, we then constructed the new spammed page graph and source graph for each of the four Web data sets. We ran PageRank and Spam-Resilient SourceRank for each of the four cases.

In Figs. 13 and 14, we show the influence of the Web spammer in manipulating the rank of the target page and the target source. Since the page-level view of the Web does not differentiate between intrasource and intersource page links, we again see that the PageRank score dramatically increases, whereas the source-centric scores are impacted less. We are encouraged to observe that all three source-centric versions perform better than PageRank. We witness this advantage using no additional influence throttling information for the sources under consideration, meaning that the source-centric advantage would be even greater with the addition of more throttling information. The baseline version does increase some, but not nearly as much as PageRank. Since the source is an aggregation of many pages, the weighting of the source edges is less susceptible to changes in the underlying page graph. Interestingly, the loopless version is the least resistant

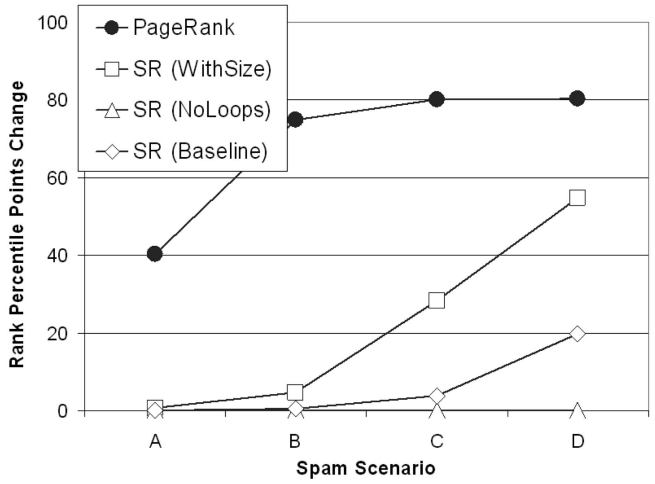

Fig. 12. Intrasource link farm, WB2001.

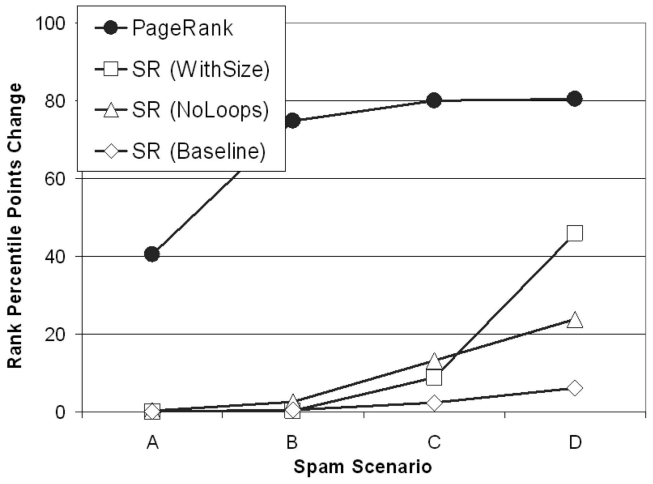

Fig. 14. Intersource link farm, WB2001.

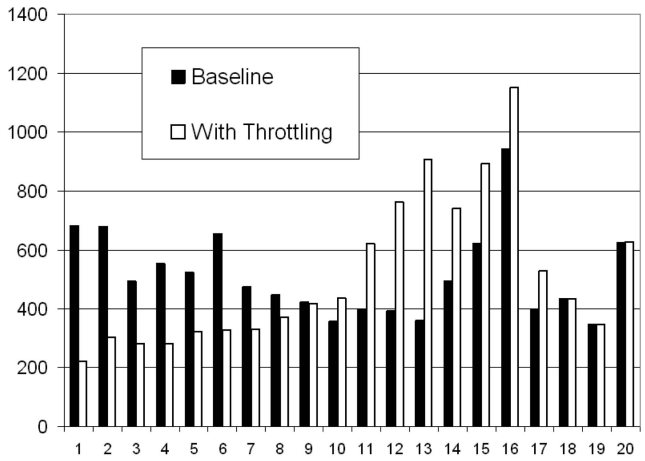

Fig. 15. Rank distribution of all spam sources.

to manipulation for cases A, B, and C. In the loopless version, external links are the sole determiner of a source's rank, meaning that intersource manipulation wields more influence here than for the looped versions. The size-based teleportation version is the most vulnerable for case D.

5.9 Influence Throttling Effectiveness

In the final experiment, we study the impact of influence throttling on the spam-resilience characteristics of source-centric ranking. For the WB2001 data set, we manually identified 10,315 pornography-related sources and labeled these as spam. It is unreasonable for a spam identification algorithm (whether manual or automated) to identify all spam sources with high precision. Hence, of these 10,315 spam sources, we randomly selected just 1,000 (fewer than 10 percent) to use as a seed set for the spam-proximity calculation. We calculated the spam-proximity score for each source using the approach described in Section 2.5.

Based on these scores, we assigned an appropriate throttling value to each source, such that sources that are "closer" to spam sources are throttled more than more distant sources. These spam proximity scores are propagated to all sources in the data set based only on the seed set of 1,000 identified spam sources. We assigned the top-20,000 spam-proximity sources a throttling value of $\kappa = 1$, meaning that their influence was completely throttled. For all other sources, we assigned a throttling value of $\kappa = 0$, meaning that these sources were throttled not at all. We then computed the source-centric ranking vector using these throttling values. As a point of comparison, we also computed the baseline ranking vector using no throttling information.

For each of the two ranking vectors, we sorted the sources in decreasing order of scores and divided the sources into 20 buckets of equal number of sources. Along the x-axis in Fig. 15, we consider these 20 buckets for the WB2001 data set, from the bucket of top-ranked sources (bucket 1) to the bucket of the bottom-ranked sources (bucket 20). Along the y-axis, we plot the number of actual spam sources (of the 10,315 total spam sources) in each bucket. The approach using influence throttling penalizes spam sources considerably more than the baseline approach, even when fewer than 10 percent of the spam sources have been explicitly marked as spam.

Note that this experiment provides evidence of the importance of influence throttling, but there are many alternative approaches worthy of consideration. We believe that further study of influence throttling would be valuable in work that builds on the results reported here.

5.10 Summary of Experiments

The evaluation has yielded interesting observations:

- SLA heavily depends on the source definition and the degree of link locality. We find that a lack of locality results in poor time complexity but that even moderate locality (e.g., $\sim$65 percent) leads to good time complexity and stability results that are comparable with source definitions that display extremely high locality.
- In terms of ranking vector stability, the most important parameters are self-edges and the source-size teleportation component. We also found that incorporating expensive quality information into the edge weighting schemes resulted in only a slight change to the resulting ranking vector.
- To best approximate PageRank and for the most stable rankings in the face of Web link evolution, we saw the critical need for using the size-based teleportation component.
- However, using the size-based teleportation component resulted in the most severe vulnerability to spam, although it has these two desirable properties.
- Finally, we saw how incorporating influence throttling information resulted in better spam-resilience properties than the baseline approach.

6 Related Work

In addition to the related work cited elsewhere, there have been some other efforts to understand higher level Web abstractions. In [33], the *hostgraph* was explored in terms of various graph properties like indegree and outdegree distribution and size of connected components. Crawling mechanisms based on the site paradigm, rather than the traditional page-based one, were enumerated in [34]. In [11], the potential spam properties of a HostRank algorithm were observed, and in [35], the ranking quality of several site-level-style PageRank variations was studied. In contrast to page aggregations, other researchers [36] have considered disaggregating Web pages into smaller units for providing ranking over individual components of Web pages. Several studies have identified large portions of the Web to be subject to malicious rank manipulation [1], [37], especially through the construction of specialized link structures for promoting certain Web pages. Several researchers have studied collusive linking arrangements with respect to PageRank, including [38] and [39]. Link farms have been studied in [40]. Separately, optimal link farms and the effectiveness of spam alliances have been studied in [24]. Davison [41] was the first to investigate the identification of so-called nepotistic links on the Web.

7 Summary

In this manuscript, we have introduced a parameterized framework for SLA, explored several critical parameters, and conducted the first large-scale comparative study of SLA over multiple large real-world Web data sets and multiple competing objectives. We find that careful tuning of these parameters is vital to ensure success over each objective and to balance the performance across all objectives. We have introduced the notion of influence throttling, studied analytically its impact, and provided experimental validation of the effectiveness and robustness of our spam-resilient ranking model in comparison with PageRank.

The struggle to combat spam is an evolutionary process with each side constantly shifting strategies and techniques. A natural and important extension to this work would be the dynamic analysis of the factors impacting SLA to provide adaptive spam resilience. In our continuing work, we are considering a number of dynamic models for capturing the dynamic and evolutionary nature of the Web graph, for example, to dynamically adapt the intrasource link weights based on the behavior of source nodes in previous time instances. We are also interested in dynamically updating the influence throttling values from period to period for providing some measure of adaptation to shifts in spammer strategies.

Acknowledgments

This work is partially supported by faculty start-up funds from Texas A&M University and the Texas Engineering Experiment Station and by grants from the US National Science Foundation (NSF) CyberTrust program, an AFOSR grant, and an IBM SUR grant.

References

[1] D. Fetterly, M. Manasse, and M. Najork, "Spam, Damn Spam, and Statistics," *Proc. Seventh Int'l Workshop the Web and Databases (WebDB),* 2004.

[2] Z. Gyöngyi and H. Garcia-Molina, "Web Spam Taxonomy," *Proc. First Int'l Workshop Adversarial Information Retrieval on the Web (AIRWeb),* 2005.

[3] C. Mann, "Spam + Blogs = Trouble," *Wired,* 2006.

[4] J.M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," *J. ACM,* vol. 46, no. 5, 1999.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," technical report, Stanford Univ., 1998.

[6] K. Bharat, B.-W. Chang, M. Henzinger, and M. Ruhl, "Who Links to Whom: Mining Linkage between Web Sites," *Proc. IEEE Int'l Conf. Data Mining (ICDM),* 2001.

[7] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub, "Exploiting the Block Structure of the Web for Computing PageRank," technical report, Stanford Univ., 2003.

[8] A. Broder, R. Lempel, F. Maghoul, and J. Pedersen, "Efficient PageRank Approximation via Graph Aggregation," *Proc. 13th Int'l World Wide Web Conf. (WWW),* 2004.

[9] Y. Lu, B. Zhang, W. Xi, Z. Chen, Y. Liu, M.R. Lyu, and W.-Y. Ma, "The PowerRank Web Link Analysis Algorithm," *Proc. 13th Int'l World Wide Web Conf. (WWW),* 2004.

[10] E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. So, "The Connectivity Sonar: Detecting Site Functionality by Structural Patterns," *Proc. 14th ACM Conf. Hypertext and Hypermedia,* 2003.

[11] N. Eiron, K.S. McCurley, and J.A. Tomlin, "Ranking the Web Frontier," *Proc. 13th Int'l World Wide Web Conf. (WWW),* 2004.

[12] P. Boldi, B. Codenotti, M. Santini, and S. Vigna, "UbiCrawler: Scalability and Fault-Tolerance Issues," *Proc. 11th Int'l World Wide Web Conf. (WWW),* 2002.

[13] A.L. da Costa Carvalho, P.A. Chirita, E.S. de Moura, P. Calado, and W. Nejdl, "Site Level Noise Removal for Search Engines," *Proc. 15th Int'l World Wide Web Conf. (WWW),* 2006.

[14] Y. Wang and D.J. DeWitt, "Computing PageRank in a Distributed Internet Search Engine System," *Proc. 30th Int'l Conf. Very Large Data Bases (VLDB),* 2004.

[15] J. Caverlee and L. Liu, "Countering Web Spam with Credibility-Based Link Analysis," *Proc. 26th ACM SIGACT-SIGOPS Symp. Principles of Distributed Computing (PODC),* 2007.

[16] L. Nie, B. Wu, and B.D. Davison, "A Cautious Surfer for PageRank," *Proc. 16th Int'l World Wide Web Conf. (WWW),* 2007.

[17] B. Wu and B. Davison, "Identifying Link Farm Spam Pages," *Proc. 14th Int'l World Wide Web Conf. (WWW),* 2005.

[18] A. Arasu, J. Novak, A. Tomkins, and J. Tomlin, "PageRank Computation and the Structure of the Web," *Proc. 11th Int'l World Wide Web Conf. (WWW),* 2002.

[19] T.-Y. Liu and W.-Y. Ma, "Webpage Importance Analysis Using Conditional Markov Random Walk," *Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence (WI),* 2005.

[20] R. Song et al., "Microsoft Research Asia at Web Track and Terabyte Track," *Proc. 13th Text Retrieval Conf. (TREC),* 2004.

[21] G.-R. Xue, Q. Yang, H.-J. Zeng, Y. Yu, and Z. Chen, "Exploiting the Hierarchical Structure for Link Analysis," *Proc. ACM Special Interest Group on Information Retrieval (SIGIR),* 2005.

[22] J. Wu and K. Aberer, "Using SiteRank for P2P Web Retrieval," technical report, Swiss Fed. Inst. of Technology, 2004.

[23] M. Bianchini, M. Gori, and F. Scarselli, "Inside PageRank," *ACM Trans. Internet Technology,* vol. 5, 2005.

[24] Z. Gyöngyi and H. Garcia-Molina, "Link Spam Alliances," *Proc. 31st Int'l Conf. Very Large Data Bases (VLDB),* 2005.

[25] A.N. Langville and C.D. Meyer, "Deeper Inside PageRank," *Internet Math.,* vol. 1, no. 3, 2005.

[26] Y.-M. Wang, M. Ma, Y. Niu, and H. Chen, "Spam Double-Funnel: Connecting Web Spammers with Advertisers," *Proc. 16th Int'l World Wide Web Conf. (WWW),* 2007.

[27] P. Boldi and S. Vigna, "The WebGraph Framework I," *Proc. 13th Int'l World Wide Web Conf. (WWW),* 2004.

[28] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing Top k Lists," *SIAM J. Discrete Math.,* vol. 17, no. 1, 2003.

[29] J. Lin, "Divergence Measures Based on the Shannon Entropy," *IEEE Trans. Information Theory,* vol. 37, no. 1, 1991.

[30] M. Kendall and J.D. Gibbons, *Rank Correlation Methods.* Edward Arnold, 1990.

[31] P. Boldi, M. Santini, and S. Vigna, "Do Your Worst to Make the Best: Paradoxical Effects in PageRank Incremental Computations," *Proc. Third Int'l Workshop Algorithms and Models for the Web-Graph (WAW),* 2004.

[32] A.Y. Ng, A.X. Zheng, and M.I. Jordan, "Stable Algorithms for Link Analysis," *Proc. ACM Special Interest Group on Information Retrieval (SIGIR),* 2001.

[33] S. Dill, R. Kumar, K.S. Mccurley, S. Rajagopalan, D. Sivakumar, and A. Tomkins, "Self-Similarity in the Web," *ACM Trans. Internet Technology,* vol. 2, no. 3, 2002.

[34] M. Ester, H.-P. Kriegel, and M. Schubert, "Accurate and Efficient Crawling for Relevant Websites," *Proc. 30th Int'l Conf. Very Large Data Bases (VLDB),* 2004.

[35] M. Thelwall, "New Versions of PageRank Employing Alternative Web Document Models," *Proc. Assoc. for Information Management (ASLIB),* vol. 56, no. 1, 2004.

[36] D. Cai, X. He, J.-R. Wen, and W.-Y. Ma, "Block-Level Link Analysis," *Proc. ACM Special Interest Group on Information Retrieval (SIGIR),* 2004.

[37] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, "Combating Web Spam with TrustRank," *Proc. 30th Int'l Conf. Very Large Data Bases (VLDB),* 2004.

[38] H. Zhang, A. Goel, R. Govindan, and K. Mason, "Improving Eigenvector-Based Reputation Systems against Collusions," *Proc. Third Int'l Workshop Algorithms and Models for the Web-Graph (WAW),* 2004.

[39] R. Baeza-Yates, C. Castillo, and V. Lopez, "PageRank Increase under Different Collusion Topologies," *Proc. First Int'l Workshop Adversarial Information Retrieval on the Web (AIRWeb),* 2005.

[40] S. Adali, T. Liu, and M. Magdon-Ismail, "Optimal Link Bombs Are Uncoordinated," *Proc. First Int'l Workshop Adversarial Information Retrieval on the Web (AIRWeb)*, 2005.
[41] B. Davison, "Recognizing Nepotistic Links on the Web," *Proc. AAAI Workshop Artificial Intelligence for Web Search*, 2000.

James Caverlee received the BA degree in economics (magna cum laude) from Duke University in 1996, the MS degree in engineering-economic systems and operations research and the MS degree in computer science from Stanford University in 2000 and 2001, respectively, and the PhD degree from the Georgia Institute of Technology (Georgia Tech) in 2007 (advisor: Ling Liu; co-advisor: William B. Rouse). He is an assistant professor of computer science in the Department of Computer Science, Texas A&M University. He directs the Web and Distributed Information Management Lab at Texas A&M University and is also affiliated with the Center for the Study of Digital Libraries. At Texas A&M University, he is leading the following research projects: SocialTrust: Trusted Social Information Management; SpamGuard: Countering Spam and Deception on the Web; and Distributed Web Search, Retrieval, and Mining. He is a member of the IEEE.

Steve Webb received the BS degree in computer science from Baylor University in 2003 and the PhD degree in computer science from the Georgia Institute of Technology (Georgia Tech) in 2008. He is the chief research scientist at Purewire, Atlanta, a leading Internet security firm. Purewire secures business and social interactions on the Web. Founded by veteran security industry entrepreneurs, the company offers Web security as a service to increase ROI and lower the total cost of security for businesses. He is also affiliated with the Center for Experimental Research in Computer Systems (CERCS) and the Georgia Tech Information Security Center (GTISC). His primary research project is the Denial of Information (DoI) Project, and his research focuses on the removal of low-quality information from online information-rich environments (e.g., e-mail systems, the Web, social networking environments, etc.). He is a member of the IEEE.

Ling Liu is an associate professor in the College of Computing, Georgia Institute of Technology (Georgia Tech), Atlanta, where she directs the research programs in the Distributed Data Intensive Systems Lab (DiSL), examining research issues and technical challenges in building large-scale distributed computing systems that can grow without limits. She and the DiSL research group have been working on various aspects of distributed data-intensive systems, ranging from distributed computing systems and enterprise systems to business workflow management systems. Her research group has produced a number of software systems that are either open sources or directly accessible online, among which the most popular ones are WebCQ and XWRAPElite. Her current research is partly sponsored by grants from US National Science Foundation (NSF) CISE CSR, ITR, CyberTrust, AFOSR, and IBM. She has published more than 160 technical papers in the areas of Internet computing systems, Internet data management, distributed systems, and information security. She is currently on the editorial board of several international journals, including *IEEE Transactions on Knowledge and Data Engineering*, *VLDB Journal*, and *International Journal of Web Services Research*, and has served a number of conferences as the PC chair, the vice PC chair, or the general chair, including the IEEE International Conference on Data Engineering (ICDE 2004, 2006, and 2007), the IEEE International Conference on Distributed Computing (ICDCS 2006), the IEEE International Conference on Collaborative Computing (CollaborateCom 2005 and 2006), the IEEE International Conference on Web Services (ICWS 2004). She is the recipient of the Best Paper Award at WWW 2004 and the Best Paper Award at IEEE ICDCS 2003 and a recipient of the 2005 Pat Goldberg Memorial Best Paper Award. She received the IBM Faculty Award in 2003 and 2006. She is a senior member of the IEEE.

William B. Rouse is the executive director of the Tennenbaum Institute, Georgia Institute of Technology, Atlanta. He also is a professor in the College of Computing and School of Industrial and Systems Engineering. He has written hundreds of articles and book chapters and has authored many books, including, most recently, *People and Organizations: Explorations of Human-Centered Design* (John Wiley & Sons, 2007), *Essential Challenges of Strategic Management* (John Wiley & Sons, 2001), and the award-winning *Don't Jump to Solutions* (Jossey-Bass, 1998). He is the editor of *Enterprise Transformation: Understanding and Enabling Fundamental Change* (John Wiley & Sons, 2006), a coeditor of *Organizational Simulation: From Modeling & Simulation to Games & Entertainment* (John Wiley & Sons, 2005) and of the best-selling Handbook of Systems Engineering and Management (John Wiley & Sons, 1999), and the editor of the eight-volume series *Human/Technology Interaction in Complex Systems* (Elsevier). Among many advisory roles, he has served as the chair of the Committee on Human Factors of the National Research Council, a member of the US Air Force Scientific Advisory Board, and a member of the Department of Defense Senior Advisory Group on Modeling and Simulation. He is a member of the National Academy of Engineering, as well as a fellow of the IEEE, the International Council on Systems Engineering (INCOSE), the Institute for Operations Research and Management Science, and the Human Factors and Ergonomics Society.