# Multidamping simulation framework for link-based ranking

Giorgos Kollias[1], Efstratios Gallopoulos[2]

[1] University of Patras, Computer Engineering and Informatics Department
26500, Patras, Greece
gdk@hpclab.ceid.upatras.gr

[2] University of Patras, Computer Engineering and Informatics Department
26500, Patras, Greece
stratis@ceid.upatras.gr

**Abstract.** We outline an approach for re-interpreting methods for ranking web pages based on interesting recent work on matrix power series representations for PageRank and its variants, such as Linear Rank, HyperRank and TotalRank. This is based on some novel properties of Google type matrices. Multidamping can be generalized and could help in the exploration of new approximations of PageRank.

## 1 Introduction

Web Information Retrieval has emerged into an extremely dynamic topic of research, providing the ground for contributions from and synergies across an extremely broad spectrum of areas of Informatics and Computational Science and Engineering; see e.g. [5,8,15,18,29]. The PageRank algorithm [35] introduces content-neutral ranking over Web pages[3]. This ranking is applied on matrices derived from the link adjacency matrix resulting from a crawl, or to the set of pages returned by the Google search engine in response to posting a search query. PageRank is based in part on two simple, common sense concepts: (i) A page is important if many important pages include links to it. (ii) A page containing many links has reduced impact on the importance of the pages it links to. Linear algebra, graph theory and stochastic modeling play a key role. Two viewpoints are to order the pages based on the nonnegative eigenvector (aptly described as the $\$25,000,000,000$ vector in [17]), or the nonnegative solution of a linear system. See [29] for a delightful mathematical account. A stochastic approach is to order based on the steady state distribution of a Markov chain; other approaches are also possible, see e.g. [1,12,21,32]. One especially interesting viewpoint is that of ranking based on a link-expressing infinite series [34,14]. A critical parameter is the damping factor, $\mu$, used to construct from the original link matrix the Google matrix, as it is often called. Boldi et al. (see [3]), proposed to introduce greater flexibility into such series by carefully selecting its coefficients. We will establish here a new interpretation of the series approach. Specifically, we will show that it is possible to represent the finite versions

[3] Stanford Univ. holds the relevant US Patent as “A method for node ranking in a linked database”.

Dagstuhl Seminar Proceedings 07071
Web Information Retrieval and Linear Algebra Algorithms
http://drops.dagstuhl.de/opus/volltexte/2007/1060

of this series as products of Google-type matrices, each built from the same link matrix but, typically, different damping factors. This representation that we term multidamping, provides an alternative framework for the simulation and interpretation of ranking vectors originally defined by means of finite series.

### 1.1 Notation

We follow Householder's notational conventions, and use capital letters to denote matrices, lower-case Roman letters for vectors except when designating dimensionality indices, and lower-case Greek letters for scalars. Whenever possible, if a scalar belongs to a matrix or vector, we will try to use the Greek letter best corresponding to the Roman one. Similarly, vectors belonging to a matrix will be named according to the latter. Furthermore, vectors will be column vectors. We denote by $\mathrm{e}$ the vector of all 1's and dimension commensurate with the context. As usual, for any square matrix $A$ and vector $\mathrm{b}$ of commensurate size, we let $K_m(A, \mathrm{b}) := \mathrm{span}\langle \mathrm{b}, A\mathrm{b}, ..., A^{m-1}\mathrm{b}\rangle$ denote the Krylov subspace of dimension $m$.

Assuming that $n$ pages are modeled, we denote the adjacency matrix by $A$ (possibly obtained by means of a web crawl, or synthetically generated using statistical results, e.g. [16]). Therefore, $\alpha_{ij} = 1$ if and only if page $i$ points to page $j$, otherwise $\alpha_{ij} = 0$. The *transition matrix* $P$ has elements $\pi_{ij} = \alpha_{ij}/\deg(i)$ when $\deg(i) \neq 0$, and zero otherwise (dangling pages); here $\deg(i) = \sum_j \alpha_{ij}$ is the outdegree of page $i$. From these, we define the (column) *stochastic matrix* $S$, as $S := P^\top + \mathrm{w}\ \mathrm{d}^\top$; $\mathrm{w} = \frac{1}{n}\mathrm{e}$, where $\mathrm{d}$ is the *dangling indicator vector* whose nonzero elements are $\delta_i = 1$ *iff* $\deg(i) = 0$. The *Google matrix* $G$ is then[4] $G := \mu\ S + (1-\mu)\ \mathrm{v}\ \mathrm{e}^\top$. For a random web surfer about to visit his next page, the damping parameter $\mu \in [0, 1]$ is the probability of choosing a link-accessible page, otherwise, i.e., with probability $1-\mu$, a path from the complete Web page set is selected based on the conditional probabilities in $\mathrm{v}$. Vector $\mathrm{v}$ is referred as *personalization* or *teleportation* vector, and matrix $H := \mathrm{v}\ \mathrm{e}^\top$ as *teleportation matrix*. Note that because $\mathrm{e}^\top v = 1$, $H$ is an oblique projection. Typically, $v = w$, while the choice of $\mu$ is an element of debate and research. It is said that Google initially used $\mu = 0.85$. The value of $\mu$ has a probabilistic interpretation, however it also affects the convergence of iterative methods for computing PageRank; e.g. see [36]. Recent studies indicate that a value close to $0.5$ might be more appropriate [2]. Note that $G$ is nonnegative, column stochastic and irreducible, therefore has a unique maximal eigenvalue, $\lambda_1 = 1$. By Perron-Frobenius theory, there exist corresponding positive right and left eigenvectors $(x, y)$ such that $Gx = x, y^\top G = y^\top$. Furthermore, if we assume that any of them is normalized, then it is also unique.

[4] For the sake of consistency with the Householder convention outlined above, we opt to use $\mu$ rather than $\alpha$ for the damping parameter, since the latter could be mistaken as an element of $A$.

## 2 Formulations of PageRank

Most frequently, Google's PageRank is described in terms of either of the following, equivalent, characterizations: $i$) As the eigenvector satisfying

$$\mathrm{x}^{\mathrm{PR}} := \arg\{G\,\mathrm{x} = \mathrm{x}\}. \tag{1}$$

$ii$) As solution of the linear system

$$\mathrm{x}^{\mathrm{PR}} := \arg\{(I - R)\,\mathrm{x} = (1-\mu)\mathrm{v}\mathrm{e}^{\top}\mathrm{x} = \mathrm{b}\}, \tag{2}$$

where $\mathrm{b} = (1-\mu)\ \mathrm{v}$ and $R = \mu\ S$ is the *relaxed stochastic matrix* in the sense that a scalar multiple is stochastic. In exact arithmetic, the resulting vectors are the same. Nevertheless, because of the size of the problem, the choice of definition and subsequent specific algorithm selection pose interesting problems to researchers. Problem size, for example, would typically prevent the use of direct methods for the linear system. A casual glance at the literature appears to indicate that most published experiments have been utilizing some of the simplest, iterative, algorithms: The power method for $(i)$ and simple relaxation methods for $(ii)$. In fact, the deployment of more sophisticated numerical machinery appears to still be an exception rather than rule; for some of these efforts, see e.g. [6,19,20,25,26].

### 2.1 Series representations

From (2) it follows that

$$\mathrm{x}^{\mathrm{PR}} := (I - \mu S)^{-1}\mathrm{v}(1-\mu). \tag{3}$$

We next note that the spectral radius $\rho(\mu S) < 1$, therefore the sought vector can be equivalently expressed by means of the convergent Neumann series,

$$\mathrm{x}^{\mathrm{PR}} = (1-\mu)\sum_{i=0}^{\infty}\mu^i S^i \mathrm{v}. \tag{4}$$

It is somewhat instructive to verify that stochasticity is preserved:

$$e^{\top}\mathrm{x}^{\mathrm{PR}} = (1-\mu)\sum_{i=0}^{\infty}\mu^i \mathrm{e}^{\top} S^i \mathrm{v} = 1,$$

Using normal forms for the Google matrix, it is possible to derive rational expressions for the PageRank vector [36]. See [13] for an interesting survey of several algebraic expressions for PageRank, including polynomial, series, rational and continued fraction representations. There is something particularly interesting regarding infinite and finite series expressions for the PageRank vector. This is related to graph theory, which is another tributary area for the ranking problem. There is a very old, intimate connection between graph path problems and matrix multiplication [23,31]. Specifically, if $A$ is a digraph's adjacency matrix, then each element in position $(i,j)$ of $A^k$ counts the number of paths of length $k$ connecting nodes $i$ and $j$. Therefore, term $(i, j)$ of the matrix series $\sum_{k=1}^{\kappa} A^k$ counts the total number of paths of length 1 up to $\kappa$ connecting these nodes. In view of this, characterization (4) is not only an algebraic relation but provides information related to paths. This idea did not remain unnoticed [3,14,33,34]. As noted in [14]:

> "... the representation of PageRank as power series provides deeper insight into the nature and properties of this ranking. The effects of the parameters, i.e. the graph, the personalization vector and the damping factor, are clearly separated from each other, and their influence on the resulting scores becomes clear".

### 2.2 Functional ranking

The above ideas led researchers to alternative ranking vectors. Of greatest interest is the idea, proposed and analyzed in [3], to generalize the series representation in (4) by letting the coefficients be functions of $\mu$:

$$\mathrm{x}^{df} := \sum_{j=0}^{\infty} \psi(j) S^j v \tag{5}$$

Function $\psi$ is referred as *damping function* and the resulting ranking as *functional*. Even though, as we will show, damping functions cannot be arbitrary, their presence makes for some interesting choices for page ranking. In fact, relation (5) can be considered to be template for a general ranking vector based on $\psi$. The above template allows the representation of series with a finite number of terms, say $\kappa + 1$, e.g. letting $\psi(j) = 0$ for $j > \kappa$. This permits to use damping functions to represent, within roundoff, any implementation of PageRank, since, in practice, any iterative method computing PageRank based on either (1) or (2) has to terminate, essentially returning a finite series for its computed value. Moreover, since a functional damping as defined by (5) amounts to the application of some function of matrix $S$ times a vector, it is well known that it is completely determined completely by its values on the spectrum of $S$ and can be written exactly as a (finite degree) polynomial times a vector [24]. To occasionally highlight the subtle difference between $i)$ a series that is finite because the damping function $\psi$ is selected to be 0 over a certain index value $\kappa$, and $ii)$ a finite implementation of an infinite series, so that $\psi$ is forced to be 0, we will specifically refer to the latter function and series as *truncated*. Several damping functions and functional rankings were introduced and analyzed in [3,9]. *Linear Rank*, for example, is

$$\mathrm{x}^{\mathrm{LR}(\kappa)} = \sum_{j=0}^{\kappa} \frac{2(\kappa + 1 - j)}{(\kappa + 1)(\kappa + 2)} S^j \mathrm{v} \tag{6}$$

We describe several more in Section 4. Note that Linear Rank is represented by means of a finite sum.

It is worth noting that it is not necessary for functional rankings to be provided directly in the series form of template (5). For instance, the actual PageRank vector can be written in terms of the minimal polynomial of $S$ with respect to $v$, say that it is of degree $m$ and denoted by $q_m(\lambda)$ [13]:

$$x^{\mathrm{PR}} = \hat{q}_{m-1}(\lambda)v, \text{ where } q_m(\lambda) = (\lambda - 1)\hat{q}_{m-1}(\lambda) \tag{7}$$

In fact, as will be established below, in analogy with polynomials, we can express the series representation of functional ranking in a special product form that we will call multidamping.

## 3 Multidamping

Many published papers on computing PageRank ([]) are based on characterization (1) and apply the simple power method [4] on $G$. In order to facilitate notation, in the sequel we will make explicit the dependence of $G$ on $\mu$ and write $G(\mu) := \mu S + (1-\mu)\mathrm{v}\mathrm{e}^\top$. Therefore, remembering the fact that $v$ is a probability vector so that $\|v\|_1 = 1$, the application of $\kappa$ steps of the power method amounts to using

$$\mathrm{x} := G(\mu)^\kappa \mathrm{v} \tag{8}$$

as approximation to PageRank. In general, of course, the exact value is obtained letting $\kappa \to \infty$. The following lemma is of interest ([7]):

**Lemma 1.** *Given $A \in \mathbb{R}^{n\times n}$, $\mathrm{u}, \mathrm{v} \in \mathbb{R}^n$ and $j > 0$, then*

$$(B + \mathrm{g}\mathrm{h}^\top)^j = B^j + K_j E_j L_j^\top$$

*where $K_j = [\mathrm{g}, B\mathrm{g}, ..., B^{j-1}\mathrm{g}]$, $L_j = [\mathrm{h}, B^\top\mathrm{h}, ..., (B^\top)^{j-1}\mathrm{h}]$, and $E_j = \mathtt{eye}(:, j : -1 : 1)$.*

Note that Matlab notation is explicitly used.

**Corollary 1.** *Using the above on $B := \mu S$, $\mathrm{g} := (1-\mu)\mathrm{v}$, $\mathrm{h} := \mathrm{e}$ from stochasticity follows that*

$$(G(\mu))^\kappa = \mu^\kappa S^\kappa + (1-\mu)(p_{\kappa-1}(S)\mathrm{v})\mathrm{e}^\top$$

*where $p_{\kappa-1}(S) = 1 + \mu S + \cdots \mu^{\kappa-1}S^{\kappa-1}$.*

The use of the same $\mu$ in every term of the product can be considered as a stationary or homogeneous process. We now extend (10) and introduce the following definition:

**Definition 1.** *Let $S$ and $v$ be as above, and the sequence of scalars $\{\mu_1, \mu_2, ...,\} \in [0,1]$. Then we call the transitions described by*

$$\mathrm{v} \to G(\mu_1)\mathrm{v} \to G(\mu_2)G(\mu_1)\mathrm{v} \to \cdots \to G(\mu_i)\cdots G(\mu_1)\mathrm{v} \rightsquigarrow \tag{9}$$

*as a multidamping surfing process modeled by damping parameters $\{\mu_1, \mu_2, ...\} \in [0,1]$.*

If the sequence of nonzero damping values is finite, then, starting from vector $v$, and applying the transitions, we obtain

$$\mathrm{x} := G(\mu_\kappa)\cdots G(\mu_1)\mathrm{v}. \tag{10}$$

When the $\mu_j$'s are not all the same, the above corresponds to a time inhomogeneous or non-stationary Markov process [28].

The following four results can be easily shown:

**Lemma 2.** *Let $G(\mu)$ as above. Then $i)$ All $G(\mu_j)$ are stochastic and $\Rightarrow$ $\rho(G(\mu_j)) = 1$. $ii)$ All products $G(\mu_{j_1})G(\mu_{j_2})\cdots G(\mu_{j_i})$ are stochastic. $iii)$ If $\mathrm{e}^\top\mathrm{v}$, then $\mathrm{e}^\top(\prod_{k=1}^{i} G(\mu_{j_k})\mathrm{v}) = 1$.*

**Lemma 3.** *Let $S$, $G(\mu)$, $H$ and $\mathrm{v}$ be as above. Then the following properties hold: $i)$ $H\mathrm{v} = \mathrm{v}$; $ii)$ $H^2 = H$; $iii)$ $HS = H$; $iv)$ $HG(\mu) = H$.*

**Corollary 2.** *From the previous lemma it follows that any product of the form $P_1P_2\cdots P_j\mathrm{v}$, where each term $P_i$ is either $H$ or $S$, can be simplified as follows: $i)$ Anything to the right of the first term $H$ drops; $ii)$ terms that end with $H\mathrm{v}$ become $\mathrm{v}$.*

**Theorem 1.**

$$\prod_{j=1}^{\kappa} G(\mu_j) = \left(\prod_{j=1}^{\kappa} \mu_j\right) S^\kappa + E$$

*where $E := \left(\sum_{j=0}^{\kappa-1} \zeta_j S^j \mathrm{v}\right)\mathrm{e}^\top$ so that $\mathrm{rank}(E) = 1$ and*

$$\begin{aligned}
\zeta_\kappa &= \mu_\kappa \cdots \mu_2\mu_1, \\
\zeta_{\kappa-1} &= \mu_\kappa \cdots \mu_2(1-\mu_1) \\
\cdots&\cdots\cdots \\
\zeta_1 &= \mu_\kappa(1-\mu_{\kappa-1}), \\
\zeta_0 &= 1-\mu_\kappa
\end{aligned}$$

### 3.1 Tracing a path with multidamping

Assuming that $H$ is the teleportation matrix corresponding to the personalization vector $\mathrm{v}$, then $H\mathrm{v} = \mathrm{v}$. It is instructive to consider the expansion resulting from two steps of multidamping defined by means of parameters $\{\mu_1, \mu_2\}$. The following derivations depend critically on Lemma 3 and Corollary 2.

STEP 1

$$\begin{aligned}
G(\mu_1)\mathrm{v} &= \mu_1 S\mathrm{v} + (1-\mu_1)H\mathrm{v} \\
&= \mu_1 S\mathrm{v} + (1-\mu_1)\mathrm{v} \in \mathrm{span}\langle \mathrm{v}, S\mathrm{v}\rangle
\end{aligned}$$

STEP 2

$$\begin{aligned} G(\mu_2)G(\mu_1)\mathrm{v} &= \mu_2\mu_1 S^2\mathrm{v} + \mu_1(1-\mu_2)HS\mathrm{v} + (1-\mu_1)\mu_2 SH\mathrm{v} \\ &\quad +(1-\mu_1)(1-\mu_2)H^2\mathrm{v} \\ &= \mu_2\mu_1 S^2\mathrm{v} + (1-\mu_1)\mu_2 S\mathrm{v} + (1-\mu_2)\mathrm{v} \\ &\in \mathrm{span}\langle \mathrm{v}, S\mathrm{v}, S^2\mathrm{v}\rangle \end{aligned}$$

In general, expanding the product form, the following pattern is obtained:

$$\begin{aligned} G(\mu_\kappa)\cdots G(\mu_1)\mathrm{v} &= \prod_{j=\kappa:-1:1} (\mu_j S + (1-\mu_j)\mathrm{v}\mathrm{e}^\top)\mathrm{v} \\ &= \sum_{j=0}^{\kappa} \zeta_j S^j \mathrm{v} \in K_{\kappa+1}(S, \mathrm{v}), \end{aligned}$$

for coefficients $\{\zeta_0, ..., \zeta_\kappa\}$ so that

$$\zeta_\kappa = \prod_{j=1}^{\kappa} \mu_j, \quad \sum_{j=0}^{\kappa} \zeta_j = 1.$$

The first relation follows directly from the expansion of the product form while the latter from the stochasticity of each term $G(\mu_j)$ and the fact that $\mathrm{e}^\top \mathrm{v} = 1$. What is especially interesting is that we can use the above to derive formulas for the damping parameters $\{\mu_1, \cdots, \mu_\kappa\}$ in terms of the coefficients $\{\zeta_0, \cdots, \zeta_\kappa\}$ and vice-versa. This is not too difficult, as the following example illustrates:

*Example 1.* Let $\kappa = 3$, $\eta$ be a normalizing scalar and define $\rho_j := \frac{\zeta_j}{\zeta_{j-1}}$. Then

$$\begin{array}{ll} \mu_1\mu_2\mu_3 = \eta\zeta_3 & \\ (1-\mu_1)\mu_2\mu_3 = \eta\zeta_2 & \\ (1-\mu_2)\mu_3 \quad = \eta\zeta_1 & \\ 1-\mu_3 \qquad\quad = \eta\zeta_0 & \end{array} \Rightarrow \begin{array}{l} \frac{\mu_1}{1-\mu_1} = \quad \rho_3 \Rightarrow \mu_1 = 1 - \frac{1}{1+\rho_3} \\ \frac{1-\mu_1}{1-\mu_2}\mu_2 = \rho_2 \Rightarrow \mu_2 = 1 - \frac{1}{1+\frac{\rho_2}{1-\mu_1}} \\ \frac{1-\mu_2}{1-\mu_3}\mu_3 = \rho_1 \Rightarrow \mu_3 = 1 - \frac{1}{1+\frac{\rho_1}{1-\mu_2}} \end{array}$$

We can thus compute the damping parameters corresponding to the series coefficients as follows:

$$\begin{aligned} \mu_i &= 1 - \frac{1}{1 + \frac{\rho_{\kappa-i+1}}{1-\mu_{i-1}}}, i = 1, ..., \kappa, \\ \text{where} \quad & \mu_0 = 0 \end{aligned}$$

The above construction leads to the next Theorem.

**Theorem 2.** *Let* $p_\kappa(\theta) := \sum_{j=0}^{\kappa} \zeta_j \theta^j$ *so that* $p_\kappa(1) = 1$, *and all* $\zeta_j > 0$. *Then there exist* $\mu_j \in [0, 1], j = 1, ..., \kappa$, *not necessarily distinct, such that*

$$\prod_{j=1}^{\kappa} G(\mu_{\kappa-j+1}) = \zeta_\kappa S^\kappa + p_{\kappa-1}(S)\mathrm{v}\mathrm{e}^\top,$$

*where* $G(\mu_j) = \mu_j S + (1-\mu_j)\mathrm{v}\mathrm{e}^\top$.

**Corollary 3.** *Let $e^\top w = 1$. Then*

$$\prod_{j=1}^{\kappa} G(\mu_j) w = \left( \prod_{j=1}^{\kappa} \mu_j \right) S^{\kappa} v + \sum_{j=0}^{\kappa-1} \zeta_j S^j w.$$

*Specifically, if $w := v$ then $\prod_{j=1}^{\kappa} G(\mu_j) v = p_\kappa(S) v$.*

Theorem 2 is central to our work. It says that it is possible, to express the series form template for pagerank-type vectors as a multidamping process, that is in product form, where each term is a Google-type matrix. Note that this differs from the usual product form of a polynomial (cf. next Subsection). Because of this special type of factorization, we can simulate functional rankings by means of multidamping processes:

**Encode:** From coefficients $\mathcal{Z} := \{\zeta_0, ..., \zeta_\kappa\}$ of power form representation of any damping function compute values of damping parameters $\phi(\mathcal{Z}) = \{\mu_1, ...., \mu_\kappa\}$.
**Simulate:** Compute $p_\kappa(S) v$ as application of multidamping $G(\mu_\kappa) \cdots G(\mu_1) v$

In addition to providing an alternative computational mechanism for computing the ranking vector, the multidamping representation could provide further insight into the underlying functional ranking. For example, because of the nature of the damping parameters, sequence $\{\mu_1, \mu_2, ..., \mu_i\}$ can be viewed as describing a set of intentions or pattern of behavior, possibly attributed to the "psychology" of the surfer: A value $\mu = 1$ implies that the surfer follows with certainty one of the preexisting links, while $\mu = 0$ means that the surfer does not care about existing links, but jumps anywhere, based on the probabilities in $v$. Specific sequence pattern could be the result of a specific behavior, e.g. a monotonically increasing sequence might imply a surfer whose attention is increasingly focused towards the preexisting structure, whereas a monotonically decreasing sequence, a surfer who is bored, and decides to "jump around". The previous theorem also implies that we could choose a specific sequence for the damping parameters, convert it into its series representation and obtain the corresponding functional ranking as in the constructs of [3], i.e. effectively consider $\mu$'s as parameters for the $\psi$ damping functions.

### 3.2 Properties of multidamping

We next provide some further analysis for multidamping. The following theorem can be used to obtain the eigenvalues of the Google matrix[5].

**Theorem 3.** *[11] Let $A \in \mathbb{R}^{n \times n}$ be an arbitrary matrix with eigenvalues $\lambda_1, ..., \lambda_n$, $v$ be an eigenvector associated with eigenvalue $\lambda_k$ and $q$ any $n$-dimensional vector. Then the matrix $A + vq^\top$ has eigenvalues $\lambda_1, ..., \lambda_{k-1}, \lambda_k + v^\top q, \lambda_{k+1}, ..., \lambda_n$.*

Alternative proofs can be found in [22,27,30].

**Corollary 4.** *The eigenvalues of $\prod_{j=1}^{\kappa} G(\mu_j)$ are $\{1, \gamma\lambda_2^\kappa, ..., \gamma\lambda_n^\kappa\}$, where $\gamma := \prod_{j=1}^{\kappa} \mu_j$ (ordering does not matter).*

[5] We thank Stefano Serra-Capizzano for bringing Brauer's paper to our attention.

*Proof.* From stochasticity of $S$, $e$ is a left eigenvector corresponding to the largest unit eigenvalue, therefore the maximum eigenvalue of $\zeta_\kappa S^\kappa + p_{\kappa-1}(S)ve^\top$ is

$$\zeta_\kappa + e^\top p_{\kappa-1}(S)v = \sum_{j=0}^{\kappa} \zeta_j = 1$$

from results about the $\{\zeta_j\}$. The remaining eigenvalues are unaffected by the rank-1 perturbation, so they are $\{\gamma\lambda_2^\kappa, ..., \gamma\lambda_n^\kappa\}$

It is also instructive to compare the factorization implied by Theorem 2 with ordinary factorization. In particular, the expressions for functional ranking are $\kappa$ degree polynomials in $S$, therefore, there exist $\omega_j \in \mathbb{C}, j = 1, ..., \kappa$, so that

$$p_\kappa(S)v = \zeta_\kappa \prod_{j=1}^{\kappa}(S - \omega_j I)v.$$

The difference from our approach is that each of the factors $S - \omega_j I$ is relaxed stochastic since satisfies $e^\top(S - \omega_j I) = (1 - \omega_j)e^\top$ instead of stochastic. Furthermore, unlike the product form of Theorem 2, it is not Google-type. Critical to this is that when the sequence of coefficients, $\zeta_j$, is positive decreasing, then the damping coefficients $\mu_j \in [0, 1]$. On the other hand, the shifts $\omega_j$ could be complex. It can be shown that they are lie outside the closed unit disk by making use of the following theorem:

*Property 1.* Let $p(z) = \zeta_\kappa z^\kappa + \cdots + \zeta_0$ so that $\zeta_j > 0$ for all $j$. Then the following properties, referred to as Eneström-Kakeya, hold [10]: If $\zeta_0 \geq \cdots \geq \zeta_\kappa > 0$ then all the roots of $p$ lie in the annulus

$$\min_{j=0,\ldots,\kappa-1} \frac{\zeta_j}{\zeta_{j+1}} \leq |z| \leq \max_{j=0,\ldots,\kappa-1} \frac{\zeta_j}{\zeta_{j+1}}.$$

If the coefficients are monotonically strictly decreasing, then all roots lie outside the closed unit disk.

## 4 Simulating functional rankings with multidamping

In this section we illustrate, by means of selected examples, how multidamping can provide an alternative viewpoint of generalized PageRank formulas based on functional rankings.

### LinearRank [3]

This corresponds, by construction, to a finite procedure: The surfer basically follows the existing link structure (with random jumps only followed at dangling pages), shorter paths are weighted most in a linear fashion, in such a way that paths longer than $\kappa$ steps are not taken into account in ranking ($\kappa$ is effectively a “cut-off” parameter)

$$x^{\mathrm{LR}} = \sum_{j=0}^{\kappa} \frac{2(\kappa + 1 - j)}{(\kappa + 1)(\kappa + 2)} S^j v$$

Then if $\kappa = 2\lambda + 1$, then

$$\phi(\mathcal{Z}) = (\frac{1}{3}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \cdots \frac{\lambda}{\lambda+1}, \frac{2\lambda+1}{2\lambda+3}).$$

or if $\kappa = 2\lambda + 2$,

$$\phi(\mathcal{Z}) = (\frac{1}{3}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \cdots \frac{2\lambda+1}{2\lambda+3}, \frac{\lambda+1}{\lambda+2}).$$

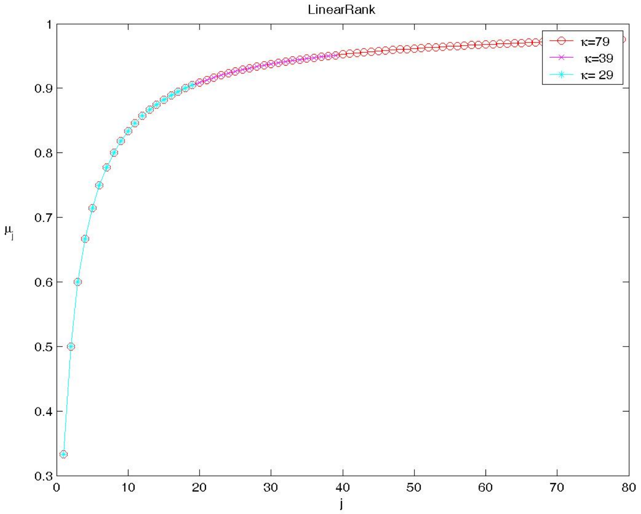

**Fig. 1.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of "cut-off" parameter $\kappa$. It can be seen that the respective Google-type surfer progressively focuses on the link structure in order to simulate LinearRank.

### TotalRank [9]

TotalRank computes an average of classic PageRank vectors for teleportation parameters $\mu$ chosen uniformly over $[0, 1]$

$$x^{\mathrm{TR}} = \int_0^1 x^{\mathrm{PR}}(\mu) d\mu = \sum_{j=0}^{\infty} \frac{1}{(j+1)(j+2)} S^j v$$

We have two possibilities in introducing a multidamping surfer. We could either forcibly *truncate* the TotalRank (TR) expansion or arrange for a very similar *finite* version with coefficients summing up to one.

Truncated TR

$$\mathcal{Z} = \{\frac{1}{2}, \frac{1}{6}, ..., \frac{1}{(\kappa+1)(\kappa+2)}\}$$

then

$$\phi(\mathcal{Z}) = 1 - \left[\frac{\kappa+2}{2(\kappa+1)}, \ldots, \frac{\kappa+2}{(j+2)(\kappa-j+1)}, ..., \frac{\kappa+2}{(\kappa+1)2}\right]$$

Finite TR

$$\mathcal{Z} = [\frac{1}{2}, \frac{1}{6}, ..., \frac{1}{\kappa(\kappa+1)}, \frac{1}{\kappa+1}]$$

$$\phi(\mathcal{Z}) = \left[\frac{\kappa}{\kappa+1}, \ldots, \frac{j+1}{j+2}, ..., \frac{1}{2}\right]$$

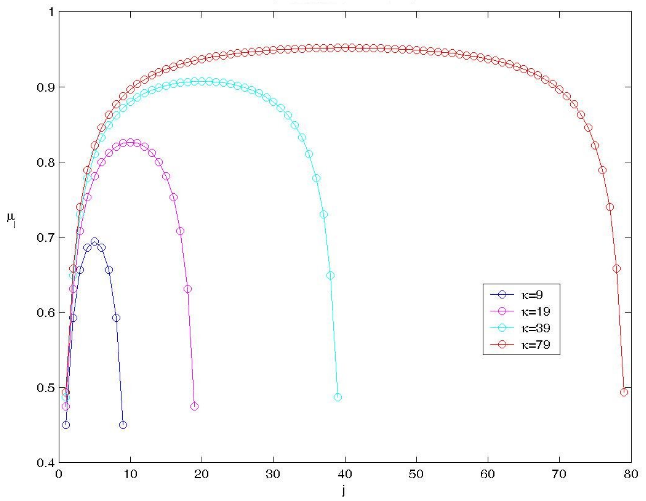

**Fig. 2.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of largest expansion order $\kappa$ for truncated TotalRank case. It can be seen that the respective Google-type surfer would have to spend most of his time focusing on the existing link structure; however note the abrupt increase (decrease) phases during his first (last) "clicks"; these phases can be the dominant feature for small $\kappa$.

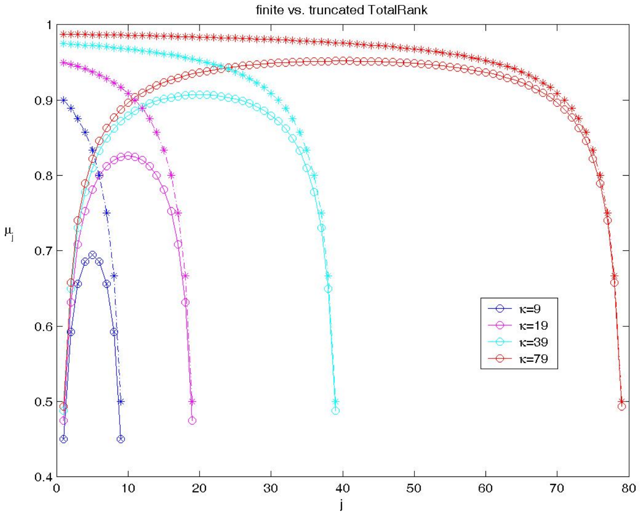

**Fig. 3.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of largest expansion order $\kappa$ for fiinite TotalRank case: The surfer gets bored following the existing link structure.

### General hyperbolic rank [3]

General hyperbolic ranking (GHR) in a way generalizes TR by using an exponent parameter $\beta$ in describing the effect of longer paths in ranking (with small $\beta$'s actually favoring contribution of longer paths)

$$x^{\mathrm{GHR}} = \frac{1}{\zeta(\beta)} \sum_{j=0}^{\infty} \frac{1}{(j+1)^{\beta}} S^j v$$

where

$$\zeta(\beta) := \sum_{j=0}^{\infty} \frac{1}{(j+1)^{\beta}}, \quad \beta > 1.$$

i.e. Riemann's zeta function. Figures 4 and 5 contain $mu$ evolution for truncated versions of GHR expansion.

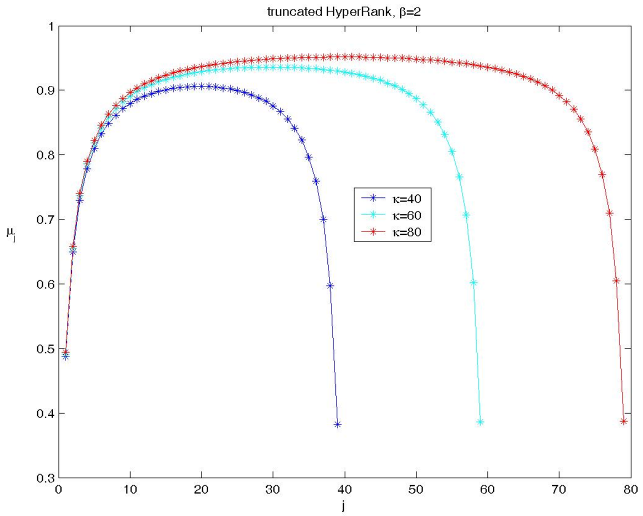

**Fig. 4.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of largest expansion order $\kappa$ for GHR case: Symmetry and behavior are similar to that of TR as expected by the similarity in the corresponding expansions.

**PageRank**

Assume that we are given the first $\kappa + 1$ terms of power series for (asymptotic) PageRank,

$$\mathcal{Z} = \{1 - \mu, (1 - \mu)\mu, (1 - \mu)\mu^2, ..., (1 - \mu)\mu^{\kappa-1}, (1 - \mu)\mu^{\kappa}$$

then

$$\phi(\mathcal{Z}) = 1 - \left[\frac{1}{1+\mu}, \frac{1}{1+\mu+\mu^2}, ..., \frac{1}{1+\mu+\mu^2+\cdots+\mu^{\kappa}}\right]$$

This is actually a *truncated* version of classic PageRank, satisfactorily approximating it in the “long run”, i.e. $1 - \frac{1}{\sum_{i=0}^{\kappa} \mu^i} \to \mu$, for large values of $\kappa$.

# 5 Remarks

## 5.1 Some properties of $\mu$

- It is natural to impose the condition that coefficients of powers in $S$ in the expansion will be given by a decreasing function in the degree of the corresponding term. This is due to the fact that larger powers in $S$ denote multiple-hop paths within

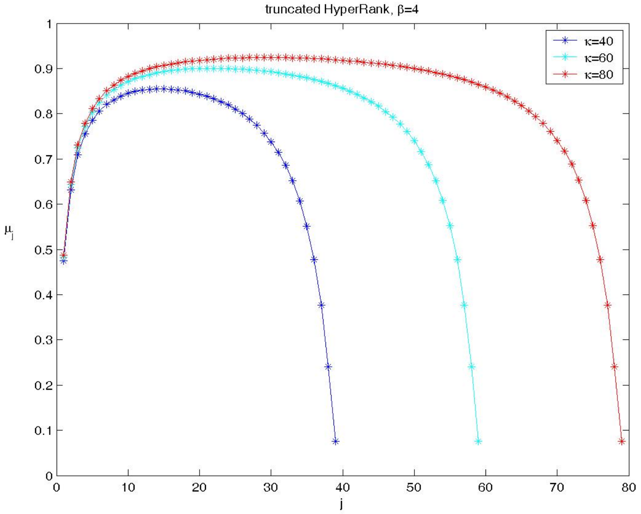

**Fig. 5.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of largest expansion order $\kappa$ for GHR case, $\beta = 4$: The symmetry of the previous figure is lost, but practically the behavior is the same.

the Web graph; more hops increase the number of choices in paving a path and so the corresponding probability of following a particular one is accordingly reduced. This condition sets the following upper bound in choosing the next $\mu$ in succession

$$\mu_{i+1} < \frac{1}{2 - \mu_i}$$

So $\mu_{i\max} = \{\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \dots\}$ for $i = 1, 2, 3, \dots$

- This condition permits generating successively smaller $\mu$'s. However if we choose the next $\mu$ to be larger that the one preceding it, it follows that the maximum permissible difference between successive $\mu$'s is

$$\delta\mu_{\max} = \frac{1 - 2\mu + \mu^2}{2 - \mu}$$

$\delta\mu_{\max}$ decreases with $\mu$ and almost vanishes as $\mu \to 1$. This explains plateaus of nearly constant $\mu$ for large $i$ for some typical cases.

## 6 Conclusions

We showed that it is possible to convert polynomials of the form $p(S)v$ that correspond to many published functional rankings, into products of Google-type matrices, where each depends on a valid damping value; not all values are necessarily equal. This

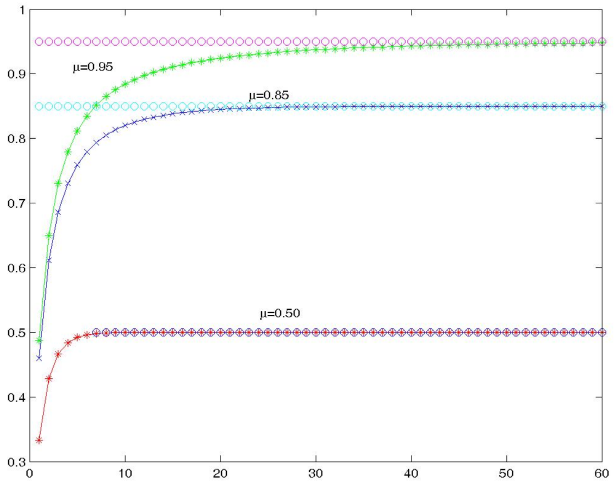

**Fig. 6.** Multidamping parameter $\mu_i$ with respect to iteration step $i$ for various values of largest expansion order $\kappa$ for truncated PageRank case: our simulation surfer soon after his first clicks faithfully adopts the classic Google-type random surfer (constant $\mu$).

generalizes the homogeneous, discrete-time, finite-state Markov process whose stationary distribution is the PageRank vector. Therefore, our approach is a tool that can be used towards the better understanding of link based ranking schemes. We note that the process can be further generalized to one also involving multiple personalization vectors. It also enables (and this is subject of ongoing work) the definition of alternative PageRank-like orderings by suitable choice of the damping coefficients.

### Acknowledgments

This work was presented during the Feb. 2007 Schloss Dagstuhl seminar *Linear Algebra and Web Information Retrieval*. We thank the organizers (A. Frommer, M. Mahoney and D. Szyld) as well as several of the participants for their valuable input. We also thank Stefano Serra-Capizzano for bringing to our attention reference [11]. We finally thank Tugrul Dayar, David Gleich, Gene Golub, Udo Krieger, Daniel Szyld and Sebastiano Vigna for specific insightful comments.

## References

1. K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova, *Monte Carlo methods in PageRank computation: When one iteration is sufficient*, Tech. report, University of Twente Research Report, Feb. 2005.
2. K. Avrachenkov, N. Litvak, and K.-S. Pham, *A singular perturbation approach for choosing pagerank damping factor*, Tech. report, Dec. 2006, in `arXiv:math.PR/0612079`.

3. R. Baeza-Yates, P. Boldi, and C. Castillo, *Generalizing PageRank: Damping functions for link-based ranking algorithms*, Proc. ACM SIGIR (Seattle, Washington, USA), ACM Press, 2006, pp. 308–315.
4. Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.
5. P. Baldi, P. Frasconi, and P. Smyth, *Modeling the internet and the web*, Wiley, 2003.
6. P. Berkhin, *A survey on pagerank computing*, Internet Math. **2** (2005), no. 1, 73120.
7. D.S. Bernstein and C.F. Van Loan, *Rational matrix functions and rank-1 updates*, SIAM J. Matrix Anal. Appl. **22** (2000), no. 1, 145–154.
8. M. Berry and M. Brown, *Understanding Search Engines: Mathematical Modeling and Text Retrieval*, second ed., SIAM, Philadelphia, 2005.
9. P. Boldi, *Total Rank*, Proc. WWW Conf., Chiba, Japan, ACM Press, 2005, pp. 898–899.
10. P. Borwein and T. Erdélyi, *Polynomials and polynomial inequalities*, Springer-Verlag, New York, 1995.
11. A. Brauer, *Limits for the characteristic roots of a matrix. IV: Applications to stochastic matrices*, Duke Math. J. **19** (1952), 75–91.
12. L.A. Breyer, *Markovian page ranking distributions: Some theory and simulations*, Available from `www.lbreyer.com/preprints.html`, 2002.
13. C. Brezinski and M. Redivo-Zaglia, *The pagerank vector: Properties, computation, approximation, and acceleration*, SIAM J. Matrix Anal. Appl. **28** (2006), no. 2, 551 – 575.
14. M. Brinkmeier, *Pagerank revisited*, ACM Trans. Inter. Tech. **6** (2006), no. 3, 282–301.
15. A. Broder and M.R. Henzinger, *Algorithmic Aspects of Information Retrieval on the Web*, Handbook of Massive Data Sets (J. Abello, P.M. Pardalos, and M.G.C. Resende, eds.), Kluwer, 2001.
16. A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A.Tomkins, and J. Wiener, *Graph structure in the web*, Proc. 9th Int'l. WWW Conf. on Computer Networks : The Int'l. J. Comput. Telecom. Netw. (Amsterdam, The Netherlands, The Netherlands), North-Holland Publishing Co., 2000, pp. 309–320.
17. K. Bryan and T. Leise, *The $25,000,000,000 eigenvector: The linear algebra behind google*, SIAM Rev. **48** (2006), no. 3, 569–581.
18. S. Chakrabarti, *Mining the Web: Discovering Knowledge from Hypertext Data*, Morgan Kaufmann Pub., 2003.
19. G.M. Del Corso, A. Gulli, and F. Romani, *Fast PageRank computation via a sparse linear system*, LNCS v. 3243, Springer, 2004, pp. 118–130.
20. ———, *Comparison of Krylov Subspace Methods on the PageRank*, Tech. Rept. TR-05-20, Università di Pisa Dipartimento di Informatica, July 2005.
21. S. Dominich and A. Skrop, *Pagerank and interaction information retrieval*, J. American Soc. Inform. Sci. Technol. **56** (2000), no. 1, 63–69.
22. L. Eldén, *A note on the eigenvalues of the Google matrix*, Tech. Report 0401177v1 [math.RA], arXiv:math, 2004.
23. L. Festinger, *The analysis of sociograms using matrix algebra*, Human Relations **2** (1949), 153 – 158.
24. F.R. Gantmacher, *The theory of matrices*, 2nd ed., vol. 1, Chelsea Pub. Co., New York, N.Y., 1959.
25. D. Gleich, L. Zhukov, and P. Berkhin., *Fast parallel pagerank: A linear system approach*, Technical report, Yahoo Corp., 2004.
26. G.H. Golub and C. Greif, *An Arnoldi-type algorithm for computing page rank*, BIT Numerical Mathematics **46** (2006), 759–771.
27. T.H. Haveliwala and S.D. Kamvar, *The Second Eigenvalue of the Google matrix*, Tech. report, Stanford Univ., March 2003.

28. D.L. Isaacson and R.W. Madsen, *Markov chains. theory and applications*, Wiley, 1976.
29. A.N. Langville and C. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton Univ. Press, 2006.
30. A.N. Langville and C.D. Meyer, *Deeper Inside PageRank*, Internet Mathematics **1** (2005), no. 3, 335–380.
31. R.D. Luce and A.D. Perry, *A method of matrix analysis of group structure*, Psychometrika **14** (1949), no. 2, 95–116.
32. F. McSherry, *A uniform approach to accelerated pagerank computation*, WWW '05: Proc. 14th Int'l. Conf. World Wide Web (New York, NY, USA), ACM Press, 2005, pp. 575–582.
33. J.C. Miller, G. Rae, F. Schaefer, L.A. Ward, A. Farahat, and T. LoFaro, *Modifications of Kleinberg's HITS algorithm using matrix exponentiation and web log records*, Proc. 24th Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval, 2001, pp. 444–445.
34. M.E.J. Newman, S.H. Strogatz, and D.J. Watts.01, *Random graphs with arbitrary degree distributions and their applications*, Physical Rev. E **64** (2001), no. 2, 1–17.
35. L. Page, S. Brin, R. Montwani, and T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*, Tech. report, Stanford Univ., 1998.
36. Stefano Serra-Capizzano, *Jordan Canonical Form of the Google Matrix: A Potential Contribution to the PageRank Computation*, SIAM J. Matrix Anal. Appl. **27** (2005), no. 2, 305–312.