2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

A New Approach for Aggregated PageRank Computation via Distributed Randomized Algorithms

Hideaki Ishii, Roberto Tempo, and Er-Wei Bai

Abstract— **At Google, the PageRank algorithm helps rankings in search results by providing measures of web page importance. This paper builds upon the distributed randomized approach for this algorithm proposed in our recent works. To reduce computation and communication, we develop a method to systematically aggregate web pages into groups by exploiting the sparsity inherent in the web. Each group computes an aggregated PageRank, which can be distributed among group members. We provide a decentralized scheme for its computation and analyze convergence properties.**

Index Terms— **Distributed computation, Multi-agent consensus, PageRank algorithm, Randomization, Search engines**

I. Introduction

The search results when using the search engine Google take account of various aspects of web pages, but it has been acknowledged that the so-called PageRank algorithm provides crucial information. This algorithm assigns to each web page a measure of its importance or popularity based on the link structure of the web (see, e.g., [16]).

One of the main challenges in the implementation of this algorithm is the size of the web. Numerical methods for PageRank have been a subject of recent research. In the adaptive scheme of [14], computational resources are allocated to pages whose convergence to the values is slow. The work of [1] employs techniques based on Monte Carlo simulation. Numerical analysis methods known as asynchronous iterations are applied in [15].

In our recent paper [10], we developed a distributed randomized approach for PageRank computation. From the control theoretic viewpoint, a key observation is that the PageRank computation shares several features with multi-agent consensus problems, which have recently gained much attention. Thus, we view the web as a network of agents having computation and communication capabilities and let each web page, or the server that hosts it, compute its own PageRank value by communicating with neighboring pages. To realize asynchronous communication, it employs the so-called gossip protocol, where the pages randomly determine when information should be transmitted. Such a randomization-based method is motivated by the probabilistic methods in systems and control [18] and has also been adopted for multi-agent consensus (e.g., [3], [5], [6]). In [11], we have also considered the effects of communication failures under this approach.

In this paper, we generalize and improve the distributed algorithms in [10] by reducing the amount of computation and communication. In doing so, the computation of the true PageRank values may become difficult. Consequently, we provide an alternative method for finding a good approximate with an estimate on the possible errors.

The proposed approach is based on a novel aggregation method of the original web to reduce the problem size. The pages are first divided into a number of groups, e.g., based on the hosts or the domains of the pages. It is known that most links in the web are intra-host ones [16], and thus the underlying graph has certain sparsity properties. We further aggregate the graph so that each group either (i) has more internal links than those going outside or (ii) consists of just one page. The aggregation procedure employs a simple criterion and can be applied to graphs with any link structures. Then, each group computes only one value in a decentralized manner via an enhanced version of the algorithms in [10]. This value represents the total value of the group members and can be distributed to determine the individual values. Aggregation can significantly reduce the computational cost while maintaining the accuracy and the convergence rate at a level similar to the non-aggregated full-order case.

The aggregation-based technique in this paper is a generalization of those studied in our works [11], [13], which are limited to computation of only the group values. In this paper, we provide a more systematic view on the problem. It is particularly motivated by the singular perturbation analyses for large-scale systems in Markov chains [17] and multi-agent consensus type problems [2], [7]. A common point in these works is that the interaction among groups is assumed to be weak, in which case the update scheme can be approximated by a lower-order one. By contrast, however, such a strict sparsity assumption does not hold for the web graph. Hence, in our grouping procedure, which is applicable to any graphs, such pages are treated as exceptions, being put into groups of their own. Aggregation for PageRank has also been explored to compute acceptable approximation in [16] by classical methods in the Markov chain literature and in [4] through extensive simulation.

This paper is organized as follows: Section II gives an overview of the PageRank problem. In Section III, we

This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology in Japan under Grant-in-Aid for Scientific Research, No. 21760323.

Hideaki Ishii is with the Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, 4259, Nagatsuta-cho, Midori-ku, Yokohama 226-8502, Japan. (e-mail: ishii@dis.titech.ac.jp).

Roberto Tempo is with IEIIT-CNR, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy. (e-mail: roberto.tempo@polito.it).

Er-Wei Bai is with the Department of Electrical and Computer Engineering, The University of Iowa, 4316 Seamans Center for the Engineering Arts and Sciences, Iowa City, IA 52242-1527, U.S.A. (e-mail: er-wei-bai@uiowa.edu).

formulate the problem of the aggregation-based algorithm, followed by the main results in Section IV. The paper is concluded in Section V. More details of the results can be found in the full version [12] of the paper.

Notation: For vectors and matrices, inequalities are used to denote entry-wise inequalities: For $X, Y \in \mathbb{R}^{n\times m}$, $X \leq Y$ implies $x_{ij} \leq y_{ij}$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m$; we say that the matrix X is nonnegative if $X \geq 0$ and positive if $X > 0$. A probability vector is a nonnegative vector $v \in \mathbb{R}^n$ such that $\sum_{i=1}^n v_i = 1$. A matrix $X \in \mathbb{R}^{n\times n}$ is said to be (column) stochastic if it is nonnegative and each column sum equals 1. Let $\mathbf{1} \in \mathbb{R}^n$ be the vector whose entries are all 1 as $\mathbf{1} := [1 \cdots 1]^T$. Similarly, $S \in \mathbb{R}^{n\times n}$ is the matrix with all entries being 1.

II. The PageRank problem

We briefly introduce the PageRank problem [16]. Consider the directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ representing a network of n web pages. Here, $\mathcal{V} := \{1, 2, \ldots, n\}$ is the set of nodes corresponding to the web page indices while $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges for the links among pages. If page i has an outgoing link to page j, then we have $(i, j) \in \mathcal{E}$.

The PageRank algorithm assigns some measure of importance to each web page. The PageRank value of page $i \in \mathcal{V}$ is given by $x_i^* \in [0, 1]$, where $x_i^* > x_j^*$ implies that page i has higher importance than page j. The pages are ranked such that a page having more links, especially those from important pages, becomes more important. This is done in such a way that the value of one page equals the sum of the contributions from all pages that have links to it. Let the values be in the vector form as $x^* \in [0, 1]^n$. Then, the PageRank vector x^* is defined by

$$x^* = Ax^*, \quad x^* \in [0, 1]^n, \quad \mathbf{1}^T x^* = 1, \tag{1}$$

where the link matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ is given by $a_{ij} = 1/n_j$ if $(j, i) \in \mathcal{E}$ and 0 otherwise, and n_j is the number of outgoing links of page j. Hence, x^* is a nonnegative unit eigenvector corresponding to the eigenvalue 1 of A.

For this eigenvector to be well defined, the convention is to modify the problem. First, for simplification, we redefine the graph by bringing in artificial links for nodes with no outgoing links such as PDF files. This can be done by adding links back to the pages having links to such pages. As a result, the link matrix A becomes a stochastic matrix. This implies that there exists at least one eigenvalue equal to 1. To guarantee the uniqueness, let m be a parameter such that $m \in (0, 1)$, and let the modified link matrix $M \in \mathbb{R}^{n\times n}$ be

$$M := (1 - m)A + \frac{m}{n}S.$$

Notice that M is a positive stochastic matrix. By Perron's theorem [9], the eigenvalue 1 is of multiplicity 1 and is the unique eigenvalue with maximum magnitude. Further, the corresponding eigenvector is positive. Hence, we redefine the value vector x^* by using M as follows.

Definition 2.1: The PageRank value vector x^* is given by

$$x^* = Mx^*, \quad x^* \in [0, 1]^n, \quad \mathbf{1}^T x^* = 1. \tag{2}$$

Due to the large dimension of the link matrix M, the computation of x^* is difficult. The solution employed in practice is based on the power method given by the recursion

$$x(k + 1) = Mx(k) = (1 - m)Ax(k) + \frac{m}{n}\mathbf{1}, \tag{3}$$

where $x(k) \in \mathbb{R}^n$ and the initial vector $x(0) \in \mathbb{R}^n$ is a probability vector. The second equality above follows from the fact $Sx(k) = \mathbf{1}$, $k \in \mathbb{Z}_+$. For implementation, the form on the far right-hand side is important, using only the sparse matrix A. This method asymptotically finds the value vector as shown below [9].

Lemma 2.2: In the update scheme (3), for any $x(0)$ that is a probability vector, it holds that $x(k) \to x^*$ as $k \to \infty$.

III. Problem formulation

In this section, we introduce the problem setting for the distributed computation of the aggregated PageRank. Following the randomized distributed approach of [10], we view the web as a network of agents having computation and communication capabilities. The focus here is to compute approximate values of the exact PageRank with reduced computation and communication. In what follows, we present the aggregation procedure and then the communication protocol.

A. Web aggregation

The original web is aggregated by assigning each page into a number of groups and then each group computes one value, which is the sum of the values of the group members. We aggregate pages sharing the following three properties: (i) The pages are placed under the same host/server so that their values can be computed together. (ii) Each group has a sufficiently large number of internal links. More specifically, pages have more links within their own groups than those pointing at pages that belong to other groups having multiple members. (iii) Group members are expected to take similar values in PageRank, which may be known from past computations and/or the link structure; see [13] for results related to this property. The process of grouping can be done at each host locally.

We develop a novel aggregation approach by exploiting sparsity properties of the web, as stated by (ii) above. The approach is closely related to the singular perturbation analysis for large-scale systems with network structures [2], [7], [17]. See [12] for further discussion.

First, partition the original graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and construct the *aggregated* graph $\widetilde{\mathcal{G}} = (\widetilde{\mathcal{V}}, \widetilde{\mathcal{E}})$ as follows:

(i) The node set is given by $\widetilde{\mathcal{V}} := \{1, 2, \ldots, r\}$, and each node i represents a partition set $\mathcal{U}_i$ of $\mathcal{V}$, that is, $\bigcup_i \mathcal{U}_i = \mathcal{V}$ and $\mathcal{U}_i \cap \mathcal{U}_j = \emptyset$, $\forall i \neq j$. We call the set $\mathcal{U}_i$ a group of pages. Let r be the number of groups, and let $\widetilde{n}_i$ be the number of pages in group $\mathcal{U}_i$. Thus, $\sum_{i=1}^r \widetilde{n}_i = n$.

(ii) The edge set $\widetilde{\mathcal{E}} = \widetilde{\mathcal{V}} \times \widetilde{\mathcal{V}}$ satisfies that if $(i_1, i_2) \in \mathcal{E}$, then $(h(i_1), h(i_2)) \in \widetilde{\mathcal{E}}$, where $h : \mathcal{V} \to \widetilde{\mathcal{V}}$ is the function indicating the group j that the web page i belongs to such that $h(i) = j$, or $i \in \mathcal{U}_j$.

Without loss of generality, we assume that in the PageRank vector x^*, the first $\widetilde{n}_1$ entries correspond to the pages

belonging to group $\mathcal{U}_1$, and the following $\widetilde{n}_2$ entries are for those in group $\mathcal{U}_2$, and so on. We also make the following assumption. It says that each group should have a sufficiently small number of external links compared to internal ones. Recall that n_i denotes the number of outgoing links of page i, and let $n_{\text{ext},i}$ be the number of links from page i to groups having more than one page. As in [7], we define the *node parameter* δ_i of page i by

$$\delta_i := \frac{n_{\text{ext},i}}{n_i}, \quad i = 1, \ldots, n. \tag{4}$$

Assumption 3.1: Given the bound $\delta \in (0,1)$ on node parameters, each group j satisfies one of the following conditions:

(i) For each page i in group j, it holds that $\delta_i \leq \delta$.

(ii) Group j consists of only one page.

In view of (ii) above, groups with one member are called *single* groups; denote by r_1 the number of such groups. These groups represent exceptional pages having high ratios of external links.

The update scheme employs the coordinate transformation $\widetilde{x}(k) := Vx(k)$ via the matrix $V = \begin{bmatrix} V_1^T & V_2^T \end{bmatrix}^T$, where $V_1 \in \mathbb{R}^{r \times n}$ and $V_2 \in \mathbb{R}^{(n-r)\times n}$ are given by

$$V_1 := \text{bdiag}(\mathbf{1}_{\widetilde{n}_i}^T), \quad V_2 := \text{bdiag}\Big([I_{\widetilde{n}_i - 1} \ 0] - \frac{1}{\widetilde{n}_i}\mathbf{1}_{\widetilde{n}_i - 1}\mathbf{1}_{\widetilde{n}_i}^T\Big), \tag{5}$$

where $\text{bdiag}(X_i)$ denotes a block-diagonal matrix whose ith diagonal block is X_i. Note that V_1 and V_2 are block-diagonal matrices containing r and $r - r_1$ blocks, respectively. They have simple structures, depending only on the sizes $\tilde{n}_i$ of the groups. Also, in V_2, the columns corresponding to the pages that form groups of their own are zero. Moreover, V_1 and V_2 are orthogonal: $V_1 V_2^T = 0$.

The PageRank vector $\widetilde{x}^*$ and the state $\widetilde{x}(k)$ after the transformation are partitioned as

$$\widetilde{x}^* = \begin{bmatrix} \widetilde{x}_1^* \\ \widetilde{x}_2^* \end{bmatrix} := \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} x^*, \quad \widetilde{x}(k) = \begin{bmatrix} \widetilde{x}_1(k) \\ \widetilde{x}_2(k) \end{bmatrix} := \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} x(k). \tag{6}$$

In the first part $\widetilde{x}_1^*$ the ith entry is the total value of the members in group i; this $\widetilde{x}_1^*$ is called the *aggregated PageRank*. In the second part $\widetilde{x}_2^*$, each entry represents the difference between a page value and the average value of the group members. In the distributed algorithm developed in Section IV, the objective is to compute $\widetilde{x}_1^*$ via information exchange only among groups. After this is completed, the second part $\widetilde{x}_2^*$ should be obtained. It will be shown that in this stage, transmissions among pages in different groups is necessary, but only once during the algorithm. Hence, reduced communication load can be expected for small r.

Remark 3.2: A simple grouping procedure for Assumption 3.1 to hold can be described as follows. The pages are initially grouped based on their hosts, so the computation of the node parameters δ_i in (4) can be done locally. Any page i whose δ_i does not satisfy the condition (i) is taken out from the group; such pages are treated as single groups, for which the condition (ii) applies. Other pages still belong to the same group, and thus their parameters δ_i are updated to check whether (i) holds for this new group. These steps are repeated until all pages under one host satisfy the assumption. This procedure terminates for any given δ. $\triangledown$

B. Communication protocol via random gossiping

For the computation of $\widetilde{x}_1(k)$, the groups send their values to linked groups. Here, we employ a gossip-type asynchronous protocol, where the groups decide to communicate with their linked neighbors at random times.

In the aggregated graph $\widetilde{\mathcal{G}} = (\widetilde{\mathcal{V}}, \widetilde{\mathcal{E}})$, the nodes exchange their values over their outgoing links. Denote by $\widetilde{\mathcal{V}}_i$ the set of indices of the groups having links from node i as

$$\widetilde{\mathcal{V}}_i := \big\{ j \in \widetilde{\mathcal{V}} : \ (i,j) \in \widetilde{\mathcal{E}}, \ j \neq i \big\}.$$

Here, we allow node i to communicate with a subset of $\widetilde{\mathcal{V}}_i$ at a time. This helps to reduce the instantaneous communication load especially for nodes having many links. For this purpose, we partition $\widetilde{\mathcal{V}}_i$ into the sets $\widetilde{\mathcal{V}}_{i,1}, \ldots, \widetilde{\mathcal{V}}_{i,g_i}$, where g_i is the number of partition sets, i.e., it holds that

$$\widetilde{\mathcal{V}}_i = \bigcup_{\ell=1}^{g_i} \widetilde{\mathcal{V}}_{i,\ell}, \qquad \widetilde{\mathcal{V}}_{i,\ell} \cap \widetilde{\mathcal{V}}_{i,j} = \emptyset, \quad \forall \ell \neq j.$$

For each node $i \in \widetilde{\mathcal{V}}$, let $\eta_i(k) \in \{0, 1, \ldots, g_i\}$ be the i.i.d. random process that specifies the set of nodes to which it sends the value $(\widetilde{x}_1(k))_i$ at time k. That is,

$$\eta_i(k) = \begin{cases} \ell & \text{if node } i \text{ sends its value to nodes in } \widetilde{\mathcal{V}}_{i,\ell}, \\ 0 & \text{if node } i \text{ does not communicate.} \end{cases} \tag{7}$$

The probability distribution of this process is given as

$$\alpha_{i,\ell} = \text{Prob}\{\eta_i(k) = \ell\}, \quad \ell = 0, 1, \ldots, g_i, \quad k \in \mathbb{Z}_+. \tag{8}$$

The update probabilities $\alpha_{i,\ell} \in (0,1)$ are chosen so as to satisfy the condition $\sum_{\ell=0}^{g_i} \alpha_{i,\ell} = 1$, $i \in \widetilde{\mathcal{V}}$.

The problem of this paper can be roughly stated as follows: Design a distributed randomized algorithm for computing approximated PageRank values such that (i) the groups compute $\widetilde{x}_1(k)$, the total values of their member pages, following the gossip protocol for communication and then (ii) from $\widetilde{x}_1(k)$, the PageRank vector $x(k)$ and, in particular, the values for individual pages are obtained.

IV. Aggregation-based PageRank computation

In this section, we present the approach for aggregating the web graph and propose an approximated version of the PageRank. Then, the distributed randomized algorithm for computing the group values $\widetilde{x}_1(k)$ is discussed.

A. Definition of aggregated PageRank

We begin by analyzing the centralized update scheme of (3) in Section II when the state is transformed as $\widetilde{x}(k) = Vx(k)$ by (6). Let $\widetilde{A} := VAV^{-1}$ and partition it in accordance with the dimensions of $\widetilde{x}_1(k)$ and $\widetilde{x}_2(k)$ as

$$\widetilde{A} = \begin{bmatrix} \widetilde{A}_{11} & \widetilde{A}_{12} \\ \widetilde{A}_{21} & \widetilde{A}_{22} \end{bmatrix}$$

with $\widetilde{A}_{11} \in \mathbb{R}^{r\times r}$. The update scheme is then expressed as

$$\widetilde{x}_1(k+1) = (1-m)\widetilde{A}_{11}\widetilde{x}_1(k) + (1-m)\widetilde{A}_{12}\widetilde{x}_2(k) + \frac{m}{n}u, \quad (9)$$

$$\widetilde{x}_2(k+1) = (1-m)\widetilde{A}_{21}\widetilde{x}_1(k) + (1-m)\widetilde{A}_{22}\widetilde{x}_2(k), \quad (10)$$

where $u := V_1\mathbf{1} = [\widetilde{n}_1 \ \cdots \widetilde{n}_r]^T$; we also used the fact $V_2\mathbf{1} = 0$. The initial states are such that $\widetilde{x}_1(0) \geq 0$ and $\mathbf{1}_r^T\widetilde{x}_1(0) = 1$. The steady state is $\widetilde{x}^*$ given in (6).

Now, to derive an approximated version of the update scheme above, we focus on the characteristics of the submatrices $\widetilde{A}_{ij}$. The inverse of the transformation matrix V in (5) can be found in an explicit form, which will be useful in our analysis. Denote it by $W := V^{-1}$ and partition it as $W = \begin{bmatrix} W_1 & W_2 \end{bmatrix}$, where $W_1 \in \mathbb{R}^{n\times r}$ and $W_2 \in \mathbb{R}^{n\times(n-r)}$ are given by

$$W_1 := \mathrm{bdiag}\Big(\frac{1}{\widetilde{n}_i}\mathbf{1}_{\widetilde{n}_i}\Big), \quad W_2 := \mathrm{bdiag}\left(\begin{bmatrix} I_{\widetilde{n}_i-1} \\ -\mathbf{1}_{\widetilde{n}_i-1}^T \end{bmatrix}\right).$$

Again, W_1 and W_2 are block-diagonal matrices with r and $r-r_1$ blocks, respectively. Moreover, the rows in W_2 that correspond to single groups are zero. It is obvious that $V_1W_1 = I$, $V_1W_2 = 0$, $V_2W_1 = 0$, and $V_2W_2 = I$.

The key observation in our aggregated approach is that the matrix A can be decomposed into three parts as [17]

$$A = I + A_{\text{int}} + A_{\text{ext}}. \quad (11)$$

Here, the *internal* link matrix A_{int} is block diagonal; its ith block is of the size $\widetilde{n}_i \times \widetilde{n}_i$, whose nondiagonal entries are the same as those of A, but its diagonal entries are chosen so that the column sums are zero. This implies that $I + A_{\text{int}}$ is a block-diagonal stochastic matrix. Hence, we have

$$V_1A_{\text{int}} = 0. \quad (12)$$

On the other hand, the *external* link matrix A_{ext} contains all elements in A which are not in the block-diagonal A_{int} while its diagonal entries are chosen so that each column sum equals zero. Let A_{ext0} be an $n \times n$ matrix whose jth column is the same as that of A_{ext} if page j belongs to a non-single group and zero otherwise for $j = 1, \ldots, n$. By the definition of W_2, we can check that

$$A_{\text{ext}}W_2 = A_{\text{ext0}}W_2. \quad (13)$$

By using the facts $W = V^{-1}$, (11), (12), and (13), the submatrices in $\widetilde{A}$ can be expressed as

$$\begin{bmatrix} \widetilde{A}_{11} & \widetilde{A}_{12} \\ \widetilde{A}_{21} & \widetilde{A}_{22} \end{bmatrix} = \begin{bmatrix} V_1AW_1 & V_1AW_2 \\ V_2AW_1 & V_2AW_2 \end{bmatrix} = \begin{bmatrix} I + V_1A_{\text{ext}}W_1 & V_1A_{\text{ext0}}W_2 \\ V_2(A_{\text{int}} + A_{\text{ext}})W_1 & I + V_2(A_{\text{int}} + A_{\text{ext0}})W_2 \end{bmatrix}. \quad (14)$$

For later use, from $\widetilde{A}_{22}$, we construct the block-diagonal matrix $\widetilde{A}'_{22}$ by removing A_{ext0} as

$$\widetilde{A}'_{22} := I + V_2A_{\text{int}}W_2. \quad (15)$$

The following results will become helpful later.

Lemma 4.1: (i) The matrix $\widetilde{A}_{11}$ is stochastic.

(ii) The matrix $\widetilde{A}'_{22}$ in (15) has spectral radius smaller than or equal to 1.

(iii) Under Assumption 3.1, it holds that $\|A_{\text{ext0}}\|_1 \leq 2\delta$.

An important implication of (ii) and (iii) of this lemma is that if the node parameter δ is sufficiently small, the matrix $(1-m)\widetilde{A}_{22}$ is stable; this is because from (14), we have $\widetilde{A}_{22} = I + V_2(A_{\text{int}} + A_{\text{ext0}})W_2 = \widetilde{A}'_{22} + V_2A_{\text{ext0}}W_2$, where A_{ext0} is proportional to δ. This fact leads us to the idea of how to approximate the scheme (9) and (10). First, express (10) for $\widetilde{x}_2(k)$ using its steady state ($\widetilde{x}_2(k+1) = \widetilde{x}_2(k)$) as

$$\widetilde{x}_2(k) = (1-m)\big[I - (1-m)\widetilde{A}_{22}\big]^{-1}\widetilde{A}_{21}\widetilde{x}_1(k), \quad (16)$$

where the matrix $I - (1-m)\widetilde{A}_{22}$ is nonsingular. This expression is motivated by the time-scale separation in singular perturbation based approaches of [2], [7], [17]. Substituting this into the recursion (9) for $\widetilde{x}_1(k)$ yields

$$\widetilde{x}_1(k+1) = (1-m)\Big\{\widetilde{A}_{11} + (1-m)\widetilde{A}_{12} \times \big[I - (1-m)\widetilde{A}_{22}\big]^{-1}\widetilde{A}_{21}\Big\}\widetilde{x}_1(k) + \frac{m}{n}u. \quad (17)$$

Note that if in addition this recursion is stable, then the steady states of the scheme above with (16) and (17) become $\widetilde{x}^*$ in (6), the same as those of (9) and (10).

In this approximate form (16) and (17), the scheme requires the recursive computation of only $\widetilde{x}_1(k)$, whose dimension equals the number r of groups. It thus appears that information should be exchanged only among groups. However, notice that the term $\widetilde{A}_{12}[I - (1-m)\widetilde{A}_{22}]^{-1}\widetilde{A}_{21}\widetilde{x}_1(k)$ involves the product of vectors of dimension $n-r$ and consequently may not be suitable for distributed computation.

We further simplify the scheme by relaxing the objective to that of computing the approximated version of the state $\widetilde{x}(k)$. Specifically, we modify the scheme (16) and (17) above under the assumption that δ is small enough. The scheme consisting of three steps is summarized as follows.

Algorithm 4.2: 1. Take the initial state $\widetilde{x}_1(0) \in \mathbb{R}^r$ as a probability vector. At each time k, compute the first state $\widetilde{x}_1(k) \in \mathbb{R}^r$ via the reduced-order recursion

$$\widetilde{x}_1(k+1) = (1-m)\widetilde{A}_{11}\widetilde{x}_1(k) + \frac{m}{n}u. \quad (18)$$

2. Compute the second state $\widetilde{x}_2(k) \in \mathbb{R}^{n-r}$ by

$$\widetilde{x}_2(k) = (1-m)\big[I - (1-m)\widetilde{A}'_{22}\big]^{-1}\widetilde{A}_{21}\widetilde{x}_1(k). \quad (19)$$

3. Transform the state back in the original coordinate by

$$x(k) = W\widetilde{x}(k) = W_1\widetilde{x}_1(k) + W_2\widetilde{x}_2(k). \quad (20)$$

The convergence of this scheme is outlined below. Let $\widetilde{x}'_1$ be the eigenvector of the stochastic matrix $(1-m)\widetilde{A}_{11} + (m/n)u\mathbf{1}^T$ corresponding to eigenvalue 1. This exists and is unique because $\widetilde{A}_{11}$ is stochastic by Lemma 4.1 (i) and moreover, u/n is a positive probability vector by definition; hence, the matrix $(1-m)\widetilde{A}_{11} + (m/n)u\mathbf{1}^T$ is positive stochastic and Perron's theorem [9] applies. Then, let

$$\widetilde{x}' := \begin{bmatrix} \widetilde{x}'_1 \\ \widetilde{x}'_2 \end{bmatrix}, \quad (21)$$

TABLE I

COMPARISON OF THE OPERATION COSTS

Algorithms	Bounds on numbers of operations
Original (3)	$O((f_0(A)+n)\overline{k})$
Aggregation-based (18)–(20)	$O((f_0(\widetilde{A}_{11})+r)\overline{k}+f_0(\widetilde{A}_{21})$ $+\sum_i(n_i-1)^2+2n-r)$

$f_0(\cdot)$: The number of nonzero entries of a matrix

where $\widetilde{x}'_2 := (1-m)\big[I-(1-m)\widetilde{A}'_{22}\big]^{-1}\widetilde{A}_{21}\widetilde{x}'_1$. The first part $\widetilde{x}'_1$ is the approximate of the aggregated PageRank $\widetilde{x}^*_1$; with some abuse of terminology, it will also be called the aggregated PageRank. Finally, let $x' := V^{-1}\widetilde{x}'$.

The update scheme above converges to x'. We state this fact as a proposition, which follows from Lemma 2.2.

Proposition 4.3: In the update scheme (18)–(20) of Algorithm 4.2, for any initial vector $\widetilde{x}_1(0)$ that is a probability vector, it holds that $x(k) \to x'$ as $k \to \infty$.

A few remarks are in order. In the first step (18), the r-dimensional state $\widetilde{x}_1(k)$ represents the groups. This step requires exchange of states only among groups and not among individual pages and is suitable for distributed computation. Once it reaches the steady state, the other two steps should be carried out. The second step (19) requires transmission over most links in the web for communicating the $(n-r)$-dimensional vector $\widetilde{A}_{21}\widetilde{x}_1(k)$. Nevertheless, the subsequent computation in this step as well as the third step (20) can be done locally within each group. This is because the matrices $I-(1-m)\widetilde{A}'_{22}$, W_1, and W_2 are all block diagonal.

Remark 4.4: The advantage of the aggregation-based approach can be highlighted in terms of its operation cost [8]. Table I summarizes the numbers of operations for the original scheme (3) and the proposed scheme (18)–(20). In both cases, $\overline{k}$ is the number of steps required for the convergence of the recursions; termination criteria have been introduced in, e.g., [14] for the centralized case and [10] for the distributed case. Also, $f_0(A)$ denotes the number of nonzero entries in the link matrix A. For a sparse matrix, its product with a vector requires operations of order $f_0(A)$. Note that for the proposed scheme, in the second step (19), the matrix $\widetilde{A}'_{22}$ is block diagonal, but the blocks may be fairly dense; in such a case, we have $f_0\big([I-(1-m)\widetilde{A}'_{22}]^{-1}\big) \approx \sum_i (n_i-1)^2$. Also, in the third step (20), the transformation matrices W_1 and W_2 satisfy $f_0(W_1) = n$ and $f_0(W_2) = n-r$, respectively. $\triangledown$

B. Aggregated PageRank and its approximation error

Here, we present a result to establish an error bound for the update scheme (18)–(20). The following theorem is based on the sparsity property in the graph $\mathcal{G}$, represented by the node parameter δ in Assumption 3.1. Let $\epsilon \in (0,1)$ be a parameter that determines the desired level of approximation. Aggregate the web so that δ is sufficiently small that

$$\delta \le \frac{m\epsilon}{4(1-m)(1+\epsilon)}. \tag{22}$$

Theorem 4.5: Under Assumption 3.1 with the parameter δ satisfying (22), the error between the steady state x' of the update scheme (18)–(20) of Algorithm 4.2 and the PageRank vector x^* is bounded as $\|x^* - x'\|_1 \le \epsilon$.

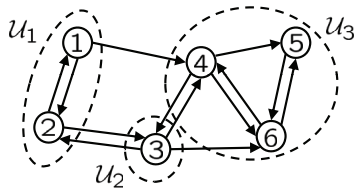

Fig. 1. The example web with groups indicated by the dashed lines

For the proof, it is useful to consider the following scheme:

$$\begin{bmatrix}\widetilde{x}_1(k+1)\\ \widetilde{x}_2(k+1)\end{bmatrix} = (1-m)\widetilde{A}'\begin{bmatrix}\widetilde{x}_1(k)\\ \widetilde{x}_2(k)\end{bmatrix} + \frac{m}{n}\begin{bmatrix}u\\ 0\end{bmatrix}, \tag{23}$$

where the matrix $\widetilde{A}'$ is given by

$$\widetilde{A}' := \begin{bmatrix}\widetilde{A}_{11} & 0\\ \widetilde{A}_{21} & \widetilde{A}'_{22}\end{bmatrix}. \tag{24}$$

It is a modified version of $\widetilde{A}$ by replacing $\widetilde{A}_{12}$ and $\widetilde{A}_{22}$ with 0 and $\widetilde{A}'_{22}$, respectively. Note that the matrix $(1-m)\widetilde{A}'$ is stable because by Lemma 4.1, $\widetilde{A}_{11}$ is stochastic and $(1-m)\widetilde{A}'_{22}$ is stable. It is straightforward to show that in this scheme (23), the state converges to $\widetilde{x}'$ in (21). Then, the vector $x' = V^{-1}\widetilde{x}'$ must be such that

$$x' = (1-m)A'x' + \frac{m}{n}\mathbf{1}, \text{ where } A' := V^{-1}\widetilde{A}'V. \tag{25}$$

The following is a key lemma for the theorem.

Lemma 4.6: Under Assumption 3.1, it holds that $\|A - A'\|_1 \le 4\delta$.

Example 4.7: Consider the web consisting of six pages shown in Fig. 1. Based on (2), the PageRank vector can be found as $x^* = \big[0.0614\ 0.0857\ 0.122\ 0.214\ 0.214\ 0.302\big]^T$. Pages 4 and 6 have the largest number of incoming links, resulting in large PageRank values.

We aggregate the nodes into three groups as $\mathcal{U}_1 = \{1,2\}$, $\mathcal{U}_2 = \{3\}$, and $\mathcal{U}_3 = \{4,5,6\}$; these are indicated by the dashed lines in Fig. 1. In this case, the node parameters are $\delta_1 = \delta_2 = 1/2$, $\delta_3 = 1$, $\delta_4 = 1/3$, and $\delta_5 = \delta_6 = 0$. Thus, with $\delta = 0.5$, all the pages satisfy Assumption 3.1. The PageRank after the coordinate transformation can be found as $\widetilde{x}^* = \big[(\widetilde{x}^*_1)^T\ (\widetilde{x}^*_2)^T\big]^T = \big[0.147\ 0.122\ 0.731 \mid -0.0121\ -0.0294\ -0.0294\big]^T$. Notice that $\widetilde{x}^*_1$ is a probability vector.

The matrices in the proposed scheme (18) and (19) are

$$\widetilde{A}_{11} = \begin{bmatrix}0.5 & 0.333 & 0\\ 0.25 & 0 & 0.111\\ 0.25 & 0.667 & 0.889\end{bmatrix},$$

$$\big[I-(1-m)\widetilde{A}'_{22}\big]^{-1}\widetilde{A}_{21} = \begin{bmatrix}0 & -0.167 & 0\\ 0.174 & 0.161 & -0.113\\ -0.0758 & -0.172 & -0.00177\end{bmatrix}.$$

For this scheme, the steady state in the original coordinate is $x' = W\widetilde{x}' = \big[0.0566\ 0.0920\ 0.125\ 0.212\ 0.213\ 0.302\big]^T$. Comparing this with the true value x^* is indeed small as $\|x' - x^*\|_1 = 0.0188$. $\triangledown$

C. Distributed randomized algorithm

We provide a distributed randomized scheme for aggregated PageRank by applying the approach of [13]. To simplify the notation, rewrite the aggregated PageRank in (21) as $\xi' := \widetilde{x}'_1$ and the recursion in the first step (18) as

$$\xi(k+1) = (1-m)\Phi\xi(k) + \frac{m}{n}u, \tag{26}$$

where the link matrix is denoted by $\Phi = (\phi_{ij}) := \widetilde{A}_{11}$ and the state by $\xi(k) := \widetilde{x}_1(k)$.

The objective is to compute the aggregated PageRank ξ' via the distributed update scheme of the form

$$\xi(k+1) = (1-\hat{m})\Phi_{\eta(k)}\xi(k) + \frac{\hat{m}}{n}u, \tag{27}$$

where $\xi(k) \in \mathbb{R}^r$ is the state whose initial condition $\xi(0)$ is a probability vector, and $\hat{m} \in (0,1)$; the process $\eta(k) := [\eta_1(k) \ \cdots \ \eta_r(k)]$ defined in (7) determines the communication pattern at time k. In this scheme, each group i also computes the time average of its own state ξ_i. Let $\psi(k)$ be the average of $\xi(0), \ldots, \xi(k)$ as

$$\psi(k) = \frac{1}{k+1}\sum_{\ell=0}^{k}\xi(\ell) = \frac{1}{k+1}\big(k\psi(k-1) + \xi(k)\big). \tag{28}$$

Let $\alpha \in (0,1]$ be the *base probability*. The update probability $\alpha_{i,\ell}$ in (8) is the probability that group i transmits to its neighbors in $\widetilde{\mathcal{V}}_{i,\ell}$ for $\ell \neq 0$. We take them as

$$\alpha_{i,\ell} = \begin{cases} 1-\alpha & \text{if } \ell = 0, \\ \alpha\dfrac{\sum_{j\in\widetilde{\mathcal{V}}_{i,\ell}}\phi_{ji}}{\sum_{j\in\widetilde{\mathcal{V}}_i}\phi_{ji}} & \text{if } \ell = 1,\ldots,g_i, \end{cases} \quad \text{for } i \in \widetilde{\mathcal{V}}_i. \tag{29}$$

Note that the probability for group i to transmit information to some neighbor is $\sum_{\ell=1}^{g_i}\alpha_{i,\ell} = \alpha$. Also, the frequency of communication among groups with more links is higher. Also, let $\hat{m} = m\alpha/[1-(1-\alpha)m]$.

In (27), the distributed link matrices $\Phi_{q_1,\ldots,q_r}$ for $q_i \in \{0,1,\ldots,g_i\}$, $i \in \widetilde{\mathcal{V}}$, are given by

$$(\Phi_{q_1,\ldots,q_r})_{pi} := \begin{cases} \frac{\alpha}{\alpha_{i,\ell}}\phi_{pi} & \text{if } q_i = \ell \neq 0,\ p \in \widetilde{\mathcal{V}}_{i,\ell}, \\ 1 - \frac{\alpha}{\alpha_{i,\ell}}\sum_{j\in\widetilde{\mathcal{V}}_{i,\ell}}\phi_{ji} & \text{if } q_i = \ell \neq 0,\ p = i, \\ 1 & \text{if } q_i = 0,\ p = i, \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

for $p, i \in \widetilde{\mathcal{V}}$. These link matrices are in accordance with the communication pattern specified by $\eta(k)$, i.e., $(\Phi_{\eta(k)})_{pi} > 0$ if group i sends its value to group p at time k.

The convergence result for the distributed scheme (27) is as follows [10], [13].

Theorem 4.8: Consider the distributed update scheme in (27) and (28). With update probabilities $\alpha_{i,\ell} \in (0,1]$, $i \in \widetilde{\mathcal{V}}$, $\ell \in \{0,1,\ldots,g_i\}$, in (29), the aggregated PageRank ξ' can be obtained from the time average $\psi(k)$ of the states $\xi(k)$ in the mean-square sense as $E\big[\big\|\psi(k) - \xi'\big\|^2\big] \to 0$, $k \to \infty$.

This distributed update scheme has the following features: (i) The computation performed at each group i includes the updates in the state ξ_i in (27) and the time average ψ_i in (28). (ii) The communication among the groups is local in that each group communicates only over direct outgoing links, as seen from the link matrices in (30). (iii) The communication load is determined by the process η, specifying the interaction among pages. (iv) At any group, the update probabilities $\alpha_{i,\ell}$ can be allocated to link groups locally within the group though α is a global parameter.

V. Conclusion

We have developed a distributed randomized algorithm for PageRank based on a novel aggregation technique. The approach utilizes a simple grouping procedure. We have introduced the notion of aggregated PageRank, from which approximates of the true values can be computed. Advantages of the approach in terms of computation as well as convergence properties have been demonstrated.

Acknowledgement: The authors would like to thank Athanasios C. Antoulas, Fabrizio Dabbene, Soura Dasgupta, Shinji Hara, and Jun-ichi Imura for helpful discussions.

References

[1] K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova. Monte Carlo methods in PageRank computation: When one iteration is sufficient. *SIAM J. Numer. Anal.*, 45:890–904, 2007.

[2] E. Bıyık and M. Arcak. Area aggregation and time-scale modeling for sparse nonlinear networks. *Syst. & Contr. Lett.*, 57:142–149, 2008.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. Inform. Theory*, 52:2508–2530, 2006.

[4] A. Z. Broder, R. Lempel, F. Maghoul, and J. Pedersen. Efficient PageRank approximation via graph aggregation. *Inform. Retrieval*, 9:123–138, 2006.

[5] K. Cai and H. Ishii. Quantized consensus and averaging on gossip digraphs. *IEEE Trans. Autom. Control*, 56:2087–2100, 2011.

[6] R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri. Communication constraints in the average consensus problem. *Automatica*, 44:671–684, 2008.

[7] J. H. Chow and P. V. Kokotovic. Time scale modeling of sparse dynamic networks. *IEEE Trans. Autom. Control*, 30:714–722, 1985.

[8] G. H. Golub and C. F. Van Loan. *Matrix Computations,* 3rd edition. Johns Hopkins University Press, 1996.

[9] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge Univ. Press, 1985.

[10] H. Ishii and R. Tempo. Distributed randomized algorithms for the PageRank computation. *IEEE Trans. Autom. Control*, 55:1987–2002, 2010.

[11] H. Ishii, R. Tempo, and E.-W. Bai. Distributed randomized PageRank algorithms based on web aggregation over unreliable channels. In *Proc.* 49*th IEEE Conf. Decision and Control*, pages 6602–6607, 2010.

[12] H. Ishii, R. Tempo, and E.-W. Bai. Distributed randomized PageRank algorithms based on web aggregation. Technical Report, 2011.

[13] H. Ishii, R. Tempo, E.-W. Bai, and F. Dabbene. Distributed randomized PageRank computation based on web aggregation. In *Proc.* 48*th IEEE Conf. on Decision and Control*, pages 3026–3031, 2009.

[14] S. Kamvar, T. Haveliwala, and G. Golub. Adaptive methods for the computation of PageRank. *Linear Algebra Appl.*, 386:51–65, 2004.

[15] G. Kollias, E. Gallopoulos, and D. B. Szyld. Asynchronous iterative computations with web information retrieval structures: The PageRank case. In G. R. Joubert *et al.*, editors, *Parallel Computing: Current and Future Issues of High-End Computing*, vol. 33 of *NIC Series*, pages 309–316. J. von Neumann-Institut for Computing, Germany, 2006.

[16] A. N. Langville and C. D. Meyer. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton Univ. Press, 2006.

[17] R. G. Phillips and P. V. Kokotovic. A singular perturbation approach to modeling and control of Markov chains. *IEEE Trans. Autom. Control*, 26:1087–1094, 1981.

[18] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Springer, London, 2005.