



HAL
open science

A distributed algorithm with consistency for PageRank-like linear algebraic systems

Constantino Lagoa, Luca Zaccarian, Fabrizio Dabbene

► **To cite this version:**

Constantino Lagoa, Luca Zaccarian, Fabrizio Dabbene. A distributed algorithm with consistency for PageRank-like linear algebraic systems. IFAC World Congress 2017, Jul 2017, Toulouse, France. pp.5172 - 5177, 10.1016/j.ifacol.2017.08.441 . hal-01700152

HAL Id: hal-01700152

<https://laas.hal.science/hal-01700152>

Submitted on 17 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A distributed algorithm with consistency for PageRank-like linear algebraic systems[★]

Constantino M. Lagoa^{*} Luca Zaccarian^{**,***}
Fabrizio Dabbene^{****}

^{*} *EE Department, The Pennsylvania State University, State College, PA 16802, USA (e-mail: szh199@psu.edu , e-mail: lagoa@psu.edu).*

^{**} *LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France, (email: zaccarian@laas.fr)*

^{***} *Department of Industrial Engineering, University of Trento, Italy.*

^{****} *CNR-IEIT, Politecnico di Torino, 10129 Torino, Italy (e-mail: fabrizio.dabbene@ieit.cnr.it).*

Abstract: We present a novel solution algorithm for a specific set of linear equations arising in large scale sparse interconnections, such as the PageRank problem. The algorithm is distributed, exploiting the underlying graph structure, and completely asynchronous. The main feature of the proposed algorithm is that it ensures that the consistency constraint (the sum of the solution components summing to one) is satisfied at every step, and not only when convergence is reached, as in the case of the different algorithms available in the literature. This represents an important feature, since in practice this kind of algorithms are stopped after a fixed number of steps. The algorithm is based on two projection steps, and represents a variation of the classical Kaczmarz method. In this paper, we present a completely deterministic version, and prove its convergence under mild assumptions on the node selection rule. Numerical examples testify for the goodness of the proposed methodology.

Keywords: Distributed linear equations, PageRank

1. INTRODUCTION

The problem of computing in a distributed manner the solution of linear equation has a very long history, dating back to the work of Gauss and Jacobi (Saad, 2003). In recent years, the advent of large-scale networks, such as the world-wide web, or large networks of wireless devices, has largely renewed the interest in such techniques, see e.g. the recent work of Mou et al. (2015) and the references therein.

A paradigmatic example of this type of problems is the computation of the Google PageRank (Brin and Page, 1998), see also Langville and Meyer (2006). In this case, the size of the network corresponds to the size of the entire web, which is said to be composed of over 8 billion pages. For this reason, centralized computation is proving excessively cumbersome (it is reported that the computation, based on power method, can take more than a week), and distributed computation techniques have been presented, see e.g. Ishii and Tempo (2010); Nazin et al. (2011); Fercoq et al. (2013). These techniques move the computational

load to the webpages, which autonomously compute their PageRank value based on exchanged information with the neighboring pages. The reader is referred to Ishii and Tempo (2014) for a nice survey and to You et al. (2017) for some recent developments.

These methods have been recently improved, and the algorithms recently proposed enjoy important features, such as exponential convergence and the possibility of asynchronous updates, see You et al. (2017). Moreover, special attention is devoted to algorithms able to work on time-varying graphs.

Mathematically, the PageRank x can be seen as the solution of a (large) linear equality $Hx = g$, subject to an equality constraint (namely the components of the vector x should sum up to one), which we refer to as *consistency constraint*. The present paper is motivated by the simple observation that all distributed methods with exponential convergence currently share a common characteristic: *the consistency constraint is only guaranteed when the algorithm has reached convergence*. Since, in practice, the algorithm is stopped after a finite number of iterations, it is of interest to satisfy the consistency constraint at every step. Hence, we propose a distributed asynchronous algorithm that, starting from an initial condition that satisfies the above mentioned consistency constraint, preserves global

[★] Research partially supported by grant OptHySYS funded by the University of Trento, and by grant PowerLyap funded by CaRiTRO and National Science Foundation (NSF) Grants CNS-1329422 and ECCS-1201973.

consistency at every step and exponentially converges to the optimal solution.

2. PROBLEM STATEMENT AND DEFINITIONS

In this paper, we propose an algorithm to find the (unique) solution to the following linear equation:

$$Hx^* = g \quad H := \begin{bmatrix} h_1^T \\ \vdots \\ h_n^T \end{bmatrix}, \quad g := \begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix}, \quad (1)$$

where $H \in \mathbb{R}^{n \times n}$ and $g \in \mathbb{R}^n$ satisfy the next assumption.

Assumption 1. Matrix H is invertible and has a real eigenvalue λ_H corresponding to a left eigenvector $\mathbf{1}$, namely $\mathbf{1}^T H = \lambda_H \mathbf{1}^T$.

Clearly, under Assumption 1 there exists a unique $x^* \in \mathbb{R}^n$ solving (1). Another consequence of this assumption is that the solution x^* to (1) satisfies:

$$\mathbf{1}^T x^* = \lambda_H^{-1} \mathbf{1}^T g, \quad (2)$$

namely the sum of its elements is equal to the sum of the elements of g multiplied by λ_H^{-1} .

We are interested in problems with the following characteristics: i) the size n is very large, ii) the matrix H is sparse. To best represent the sparsity requirement, for each row h_k of H , let us introduce the following selection vector:

$$s_k := \begin{bmatrix} s_{k,1} \\ \vdots \\ s_{k,n} \end{bmatrix}, \quad s_{k,i} := \begin{cases} 1, & \text{if } h_{k,i} \neq 0, \\ 0, & \text{if } h_{k,i} = 0. \end{cases} \quad (3)$$

Such a vector satisfies a few useful properties, such as:

$$\text{diag}(s_k)h_k = h_k, \quad \mathbf{1}^T h_k = s_k^T h_k, \quad \mathbf{1}^T s_k = s_k^T s_k. \quad (4)$$

As mentioned in the Introduction, an example where this type of large scale sparse problems arise is the PageRank computation, which is briefly reviewed next.

Example 1. (PageRank computation). Consider a set of indexed pages with labels $1, 2, \dots, n$. Let n_j be the number of outgoing links of page j . Define, the matrix A whose (i, j) entry is

$$a_{i,j} = \begin{cases} 1/n_j & \text{if page } j \text{ links to page } i \\ 0 & \text{otherwise.} \end{cases}$$

Let $m \in (0, 1)$ be a constant whose value is usually taken to be 0.15. Also, let $\mathbf{1}$ be the vector of dimension n having all entries equal to 1. The PageRank problem is defined as finding the unique solution of

$$x = (1 - m)Ax + \frac{m}{n}\mathbf{1}.$$

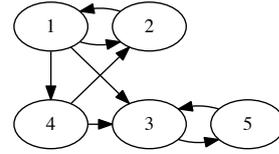
Equivalently, we want to solve the set of equations

$$Hx^* = g$$

where

$$H = I - (1 - m)A \quad \text{and} \quad g = \frac{m}{n}\mathbf{1}.$$

The solution of such a set of equations always sums to one; i.e., $\mathbf{1}^T x^* = 1$, thereby providing $\lambda_H = m$.



As an illustrative example, consider the following network.

Matrix A is given by

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/2 & 1 \\ 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (5)$$

Then, we have

$$H = \begin{bmatrix} 1.0 & -0.85 & 0 & 0 & 0 \\ -0.283 & 1.0 & 0 & -0.425 & 0 \\ -0.283 & 0 & 1.0 & -0.425 & -0.85 \\ -0.283 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & -0.85 & 0 & 1.0 \end{bmatrix}$$

◇

We formalize next the exact problem at stake.

Problem 1. Find an iterative algorithm whose output z corresponds to an asymptotic estimate of x^* , enjoying the following properties:

- (1) *exponential convergence:* the estimate z converges exponentially to x^* in some deterministic or stochastic sense.
- (2) *distributedness:* at each iteration, the algorithm, only depends on one row h_k, g_k of H, g and only on the elements of the estimate z corresponding to nonzero entries in h_k (namely on z_k and $\text{diag}(s_k)z$) and similarly for possible additional variables;
- (3) *sparsity:* at each iteration, the algorithm only updates the elements of the estimate z corresponding to nonzero entries in h_k (namely $(I - \text{diag}(s_k))(z^+ - z) = 0$) and similarly for possible additional variables;
- (4) *consistency:* at each iteration, the estimate z satisfies the consistency property in (2) (namely $\mathbf{1}^T z = \lambda_H^{-1} \mathbf{1}^T g$).

Remark 1. It should be noted that properties (1)–(3) are shared by other algorithms recently presented in the literature. For instance, the randomized algorithm presented in You et al. (2017) enjoys almost sure exponential convergence. On the other hand, we are not aware of any algorithm which guarantees consistency. As previously discussed, this represents a very important feature. Indeed, in practice this kind of algorithms are usually stopped after a fixed number of steps, and one may not be sure that convergence has been reached completely. This becomes even more crucial in the case of time varying graphs (as the world-wide-web surely is). In this case, the PageRank

will be slowly changing, and hence one would like to be sure that at each time instant the current PageRank value is consistent. To this end, we remark that the proposed solution methodology can be extended to the case of time-varying graphs.

The solution we propose is a modification of the projection algorithm proposed by Kaczmaz (1937) whose distributed version has become rather popular in the last years with the introduction of randomized distributed version, see for instance Strohmer and Vershynin (2008); Liu and Wright (2016) and references therein.

3. PROPOSED ALGORITHM AND ITS PROPERTIES

To solve Problem 1 we introduce two variables $z \in \mathbb{R}^n$, and $b \in \mathbb{R}^n$, which are updated at each iteration of the algorithm according to the following discrete-time dynamics:

$$\begin{aligned} z^+ &= z - L_\kappa \left(b + \frac{h_\kappa}{|h_\kappa|^2} y_\kappa \right) \\ b^+ &= b - L_\kappa b + \frac{s_\kappa s_\kappa^T h_\kappa}{|s_\kappa|^2 |h_\kappa|^2} y_\kappa, \end{aligned} \quad z(0) = \lambda_H^{-1} g, \quad b(0) = 0, \quad (6)$$

with

$$L_\kappa := \text{diag}(s_\kappa) - \frac{s_\kappa s_\kappa^T}{|s_\kappa|^2}, \quad y_\kappa := h_\kappa^T (z - b) - g_\kappa, \quad (7)$$

and where the sequence $j \in \mathbb{Z}_{\geq 0} \mapsto \kappa(j) \in \{1, \dots, n\}$ characterizes the selection of rows of H sequentially considered by the algorithm. Note that the initialization $z(0)$ requires knowledge of λ_H , which is indeed globally known, e.g., in the PageRank problem of Example 1, where $\lambda_H = m$.

We first establish below good properties of the proposed algorithm in terms of the last three items of Problem 1.

Proposition 1. Algorithm 6 satisfies items 2 to 4 of Problem 1.

Proof. Proof of item 2. First note that scalar y_κ in (7) only depends on the elements of z and b corresponding to nonzero entries of h_κ (due to the scalar product), and on g_κ . Similarly, $L_\kappa b$ only depends on those elements due to the effect of the selection matrix s_κ . The remaining elements of the algorithm only depend on s_κ (which is a function of h_κ) and on h_κ itself.

Proof of item 3. First notice that $(I - \text{diag}(s_\kappa))L_\kappa = 0$ and $(I - \text{diag}(s_\kappa))s_\kappa = 0$ because $\text{diag}(s_\kappa) \text{diag}(s_\kappa) = \text{diag}(s_\kappa)$ and $\text{diag}(s_\kappa)s_\kappa = s_\kappa$. Then sparsity follows from the next derivations:

$$\begin{aligned} (I - \text{diag}(s_\kappa))(z^+ - z) &= \\ - (I - \text{diag}(s_\kappa))L_\kappa \left(b + \frac{h_\kappa}{|h_\kappa|^2} y_\kappa \right) &= 0 \end{aligned}$$

Proof of item 4. Due to the specific selection of the initial condition in (6), we have $\mathbf{1}^T z(0) = \lambda_H^{-1} \mathbf{1}^T g$. Moreover, using $\mathbf{1}^T L_\kappa = 0$, we have for all $j \geq 0$,

$$\begin{aligned} \mathbf{1}^T z(j+1) &= \mathbf{1}^T z(j) - \mathbf{1}^T L_{\kappa(j)} \left(b(j) + \frac{h_{\kappa(j)}}{|h_{\kappa(j)}|^2} y_{\kappa(j)} \right), \\ &= \mathbf{1}^T z(j) \end{aligned}$$

which establishes the result by induction. \diamond

Proving the first item of Problem 1 requires additional assumptions on persistence of excitation from the selection signal $\kappa(\cdot)$ appearing in (6).

4. CONVERGENCE ANALYSIS

To suitably study the exponential convergence of (6), it is convenient to use the following error coordinates:

$$(e_1, e_2) := (z - b - x^*, z - x^*), \quad (8)$$

where the definition of e_1 is motivated by the fact that y_κ in (7) can be expressed as $y_\kappa = h_\kappa^T (z - b) - h_\kappa^T x^* = h_\kappa^T e_1$.

The arising error dynamics can be conveniently computed from (6) and corresponds to:

$$\begin{aligned} e_1^+ &= z - b - x^* - L_\kappa \frac{h_\kappa}{|h_\kappa|^2} y_\kappa - \frac{s_\kappa s_\kappa^T h_\kappa}{|s_\kappa|^2 |h_\kappa|^2} y_\kappa \\ &\stackrel{(7)}{=} e_1 - \text{diag}(s_\kappa) \frac{h_\kappa}{|h_\kappa|^2} y_\kappa \\ &\stackrel{(4)}{=} \left(I - \frac{h_\kappa h_\kappa^T}{|h_\kappa|^2} \right) e_1 \end{aligned} \quad (9)$$

$$\begin{aligned} e_2^+ &= z - x^* - L_\kappa \left(e_2 - e_1 + \frac{h_\kappa}{|h_\kappa|^2} h_\kappa^T e_1 \right) \\ &= (I - L_\kappa) e_2 + L_\kappa \left(I - \frac{h_\kappa h_\kappa^T}{|h_\kappa|^2} \right) e_1, \end{aligned} \quad (10)$$

which reveals a convenient cascaded structure.

It is evident that exponential convergence to zero of $e_2 = z - x^*$ requires convergence to zero of e_1 . To this end, and due to the relatively simple time-varying dynamics in (9), it is quite evident that the selection signal $\kappa(\cdot)$ must be “rich” enough in terms of persistence of excitation, to be able to drive e_1 to zero. In particular, note that matrix $\left(I - \frac{h_\kappa h_\kappa^T}{|h_\kappa|^2} \right)$ is a projection matrix with one eigenvalue equal to 0 and $n - 1$ eigenvalues equal to 1. The key to convergence to zero of e_1 is that κ persistently spans all possible directions in the invertible matrix H (see Assumption 1). This requirement is formalized in the next assumption that implies the follow-up full rank property.

Assumption 2. There exists a scalar $N \in \mathbb{Z}_{>0}$ such that, for each $k \in \{1, \dots, n\}$,

- (1) there exists $j \in \{0, \dots, N\}$ such that $\kappa(j) = k$;
- (2) for each $j_1 \in \mathbb{Z}_{\geq 0}$ satisfying $\kappa(j_1) = k$, there exists $j_2 \in \{1, \dots, N\}$ such that $\kappa(j_1 + j_2) = k$.

Note that Assumption 2 corresponds to some kind of reverse dwell time condition about the recurrence of each one of the n rows of H within the selection performed by signal κ . With this assumption in place we can prove

$$\begin{aligned}
\Delta V_1 &:= V_1(j+1, e_1^+) - V_1(e_1) = (e_1^+)^T (\lambda_1 I - M(j+1)) e_1^+ - e_1^T (\lambda_1 I - M(j)) e_1 \\
&= \lambda_1 \left(\underbrace{(I - \Gamma_\kappa) e_1}_{=e_1^+} \right)^T (I - \Gamma_\kappa) e_1 - e_1^T e_1 + e_1^T M(j+1) e_1 - (e_1^+)^T M(j+1) e_1^+ - e_1^T \Delta M(j) e_1 \\
&= e_1^T (-I + \Pi_\infty + \Gamma_\kappa + \lambda_1 \Gamma_\kappa^2 - 2\lambda_1 \Gamma_\kappa) e_1 + e_1^T (\Gamma_\kappa M(j+1) \Gamma_\kappa - \Gamma_\kappa M(j+1) - M(j+1) \Gamma_\kappa) e_1 \\
&= -e_1^T \underbrace{(I - \Pi_\infty + (\lambda_1 - 1) \Gamma_\kappa^2 - \Gamma_\kappa M(j+1) \Gamma_\kappa + \Gamma_\kappa M(j+1) + M(j+1) \Gamma_\kappa)}_{=Q(j)} e_1.
\end{aligned}$$

the following rank condition, which is a key tool for our Lyapunov construction.

Given the decentralized nature of the algorithm developed, we also need assumptions on the ‘‘connectivity’’ of the distributed update scheme.

Assumption 3. Construct an undirected graph \mathcal{G} where node i is connected to node j whenever there exists a k such that $h_{k,i} \neq 0$ and $h_{k,j} \neq 0$. We assume \mathcal{G} to be connected.

Lemma 1. Under Assumptions 2 and 3, there exists a symmetric matrix $\Sigma > 0$ such that, for all $j \in \mathbb{Z}_{\geq 0}$,

$$0 < \Sigma \leq \frac{1}{N} \sum_{k=j}^{j+N-1} \frac{h_{\kappa(k)} h_{\kappa(k)}^T}{|h_{\kappa(k)}|^2}, \quad (11)$$

$$0 < \Sigma \leq \mathbf{1}\mathbf{1}^T + \frac{1}{N} \sum_{k=j}^{j+N-1} L_{\kappa(k)}, \quad (12)$$

where $\mathbf{1}$ denotes the vector whose components are all equal to 1.

Proof. *Proof of (11)* The immediate implication of Assumption 1 is that every N consecutive steps, the algorithm selects at least once each row of H . As a consequence, since matrices $\frac{h_{\kappa(k)} h_{\kappa(k)}^T}{|h_{\kappa(k)}|^2}$ are all positive semidefinite, denoting by S the right hand side of (11), for each $j \in \mathbb{Z}_{\geq 0}$, we have:

$$\begin{aligned}
NS &:= \frac{N}{N} \sum_{k=j}^{j+N-1} \frac{h_{\kappa(k)} h_{\kappa(k)}^T}{|h_{\kappa(k)}|^2} \\
&\geq \sum_{i=1}^n \frac{h_i h_i^T}{|h_i|^2}.
\end{aligned}$$

Since matrix H is full rank by assumption, then there exists a scalar $\bar{h} > 0$ satisfying $\bar{h} \geq |h_i|$ for all i . Then we obtain:

$$\bar{h}^2 NS \geq \sum_{i=1}^n h_i h_i^T =: \bar{S},$$

where the sum \bar{S} at the right hand side is necessarily full rank because non-singularity of H implies that for each vector $\bar{x} \neq 0$ there exists at least one index $j \in \{1, \dots, n\}$ such that $h_j^T \bar{x} \neq 0$ (otherwise we would have $H\bar{x} = 0$ and H could not be full rank). Then, for that same generic vector \bar{x} we would get

$$\bar{x}^T \bar{S} \bar{x} = \bar{x}^T \sum_{i=1}^n h_i h_i^T \bar{x} \geq |h_i^T \bar{x}|^2 > 0,$$

which implies that \bar{S} is positive definite. As a consequence, we may select $\Sigma = \frac{1}{\bar{h}^2 N} \bar{S}$ and prove the claim.

Proof of (12). Following similar steps to the ones above, to prove (12) it is sufficient to show that the following matrix

$$\frac{1}{n} \sum_{k=1}^n L_k,$$

has rank $n - 1$ whose zero eigenvalue corresponds to the eigenvector $\mathbf{1}$.

To this end define the matrix $T_k := I - L_k$. Given the definition of L_k , T_k is symmetric, doubly stochastic, all entries are nonnegative and

$$T_k(i, j) > 0 \Leftrightarrow i = j \text{ or } (h_{k,i} \neq 0 \text{ and } h_{k,j} \neq 0)$$

Further define

$$T := \frac{1}{n} \sum_{k=1}^n T_k = I - \frac{1}{n} \sum_{k=1}^n L_k.$$

The matrix T is symmetric, doubly stochastic and $T(i, j) > 0$ if and only if (i, j) is an edge of the graph \mathcal{G} in Assumption 3. Given the fact that \mathcal{G} is assumed to be strongly connected and the diagonal entries of T are strictly positive, this implies that T is a double stochastic primitive (and hence irreducible) matrix. Therefore i) the spectral radius $\rho(T) = 1$, ii) T has an eigenvalue equal to one and all other eigenvalues have magnitude strictly less than one and iii) the eigenvector associated with eigenvalue one is $\mathbf{1}$. This immediately implies that

$$\frac{1}{n} \sum_{k=1}^n L_k = I - T$$

is a positive semidefinite matrix with exactly one eigenvalue equal to zero with corresponding eigenvector $\mathbf{1}$. \diamond

Based on Lemma 1, we can prove the following result.

Theorem 1. Consider algorithm (6) under Assumptions 1 and 3. There exist scalars $K > 0$ and $\mu \in [0, 1)$ such that, for any selection function κ satisfying Assumption 2, the sequence generated from (1) satisfies:

$$|z(j) - x^*| \leq K \mu^j |z(0) - x^*| \quad (13)$$

Proof. We prove the theorem by focusing on the error dynamics already derived in (9)–(10), which has a convenient cascaded structure.

Exponential convergence of e_1 . Let us first consider the upper subsystem (9) and prove the exponential conver-

gence to zero of e_1 . To this end, define the following matrix function of the algorithm iteration $j \in \mathbb{Z}_{\geq 0}$:

$$\Pi(j) := \frac{1}{j} \sum_{i=0}^{j-1} \left(I - \frac{h_{\kappa(i)} h_{\kappa(i)}^T}{|h_{\kappa(i)}|^2} \right) = \frac{1}{j} \sum_{i=0}^{j-1} (I - \Gamma_{\kappa(i)}), \quad (14)$$

and note that, by Lemma 1 (see equation (11)), we have that there exist a symmetric matrix $\Sigma > 0$ and a scalar $\sigma_1 > 0$ such that:

$$\Pi_{\infty} := \lim_{j \rightarrow \infty} \Pi(j) \leq I - \Sigma < I, \quad (15)$$

$$- \sigma_1 I \leq M(j) := j(\Pi(j) - \Pi_{\infty}) \leq \sigma_1 I. \quad (16)$$

Consider now the following Lyapunov function candidate:

$$V_1(j, e_1) := e_1^T (\lambda_1 I - M(j)) e_1, \quad (17)$$

where $\lambda_1 > \sigma_1$, so that, also using (16), the following uniform quadratic bound holds:

$$\underline{c}_1 |e_1|^2 \leq V_1(j, e_1) \leq \bar{c} |e_1|^2, \quad \forall j, e_1, \quad (18)$$

which clearly implies that the Lyapunov function is uniformly positive definite. We perform the following preliminary calculation to compute the variation of V_1 at each step of the algorithm:

$$\begin{aligned} \Delta M(j) &:= M(j+1) - M(j) \\ &= I - \Gamma_{\kappa(j)} - (j+1-j)\Pi_{\infty} \\ &= I - \Pi_{\infty} - \Gamma_{\kappa(j)}. \end{aligned}$$

Then, using the property $\Gamma_{\kappa}^2 = \Gamma_{\kappa}$ and the fact that Γ_{κ} is symmetric, we may compute the bounds given at the top of the page. Consider now inequality (15), which clearly implies the existence of a (small) positive scalar $\varepsilon_1 > 0$ such that $I - \Pi_{\infty} > 2\varepsilon_1 I$ and then, also using (16), we may bound matrix $Q(j)$ given at the top of the previous page as follows:

$$\begin{aligned} e_1^T Q(j) e_1 &\geq 2\varepsilon_1 |e_1|^2 + (\lambda_1 - 1 - \sigma_1) |\Gamma_{\kappa} e_1|^2 - 2\sigma_1 |e_1| |\Gamma_{\kappa} e_1| \\ &= \begin{bmatrix} |e_1| \\ |\Gamma_{\kappa} e_1| \end{bmatrix}^T \begin{bmatrix} 2\varepsilon_1 & -\sigma_1 \\ -\sigma_1 & \lambda_1 - 1 - \sigma_1 \end{bmatrix} \begin{bmatrix} |e_1| \\ |\Gamma_{\kappa} e_1| \end{bmatrix}, \end{aligned}$$

which clearly reveals that picking $\lambda_1 > 0$ large enough, it is possible to obtain

$$\Delta V_1(j, e_1) \leq -\varepsilon_1 |e_1|^2. \quad (19)$$

Then, using standard Lyapunov theory, the bound above can be combined with (18) to obtain an exponential bound:

$$|e_1(j)| \leq K_1 \sqrt{(1 - \varepsilon_1)^j} |e_1(0)|. \quad (20)$$

for some positive constant K_1 .

Exponential convergence of e_2 . Let us now consider the evolution of the second error variable e_2 . To this end, consider the following coordinate change, where U_2 is any orthonormal basis of the orthogonal complement to the subspace generated by the vector $\mathbf{1}$ having all its elements equal to 1:

$$\begin{aligned} \eta &:= \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} := \begin{bmatrix} n^{-1/2} \mathbf{1}^T \\ U_2^T \end{bmatrix} e_2, \\ e_2 &= \begin{bmatrix} n^{-1/2} \mathbf{1} \\ U_2 \end{bmatrix} \eta. \end{aligned} \quad (21)$$

Let us first note that since s_{κ} has all its elements either equal to zero or to 1, then we have $\mathbf{1}^T s_{\kappa} = |s_{\kappa}|^2$, which

implies $\mathbf{1}^T L_{\kappa} = s_{\kappa}^T - |s_{\kappa}|^2 \frac{s_{\kappa}^T}{|s_{\kappa}|^2} = 0$. Then we obtain from expression (10),

$$\mathbf{1}^T e_2^+ = \mathbf{1}^T e_2 + \mathbf{1}^T L_{\kappa} (-e_2 + (I - \Gamma_{\kappa}) e_1) = \mathbf{1}^T e_2, \quad (22)$$

which implies, for all $j \in \mathbb{Z}_{\geq 0}$,

$$\begin{aligned} \eta_1(j) &= n^{-1/2} \mathbf{1}^T e_2(j) = n^{-1/2} \mathbf{1}^T e_2(0) \\ &= n^{-1/2} (\mathbf{1}^T z(0) - \mathbf{1}^T x^*) = 0, \end{aligned} \quad (23)$$

where we used property (2) and the initial condition in (6).

Let us now consider the evolution of variable η_2 in (21), which corresponds to

$$\eta_2^+ = U_2^T (I - L_{\kappa}) e_2 + U_2^T L_{\kappa} (I - \Gamma_{\kappa}) e_1. \quad (24)$$

Using the following identities

$$\begin{aligned} U_2^T (I - L_{\kappa}) \underbrace{[n^{1/2} \mathbf{1} \ U_2]}_{=e_2} \eta \\ = U_2^T (I - L_{\kappa}) U_2 \eta_2 + n^{1/2} \underbrace{(U_2^T \mathbf{1} - U_2^T L_{\kappa} \mathbf{1})}_{=0} \eta_1, \end{aligned}$$

dynamics (24) can be written as:

$$\begin{aligned} \eta_2^+ &= U_2^T (I - L_{\kappa}) U_2 \eta_2 + d \\ &= (I - U_2^T L_{\kappa} U_2) \eta_2 + d, \end{aligned} \quad (25)$$

where we used $U_2^T U_2 = I$ and where $|d| \leq M_d |e_1|$ for some scalar M_d , because the input matrices multiplying e_1 at the right hand side of (24) are bounded.

Consider now inequality (12) established in Lemma 1. Pre- and post-multiplying both sides by U_2 and its transpose, we obtain:

$$0 < U_2^T \Sigma U_2 \leq \underbrace{U_2^T \mathbf{1} \mathbf{1}^T U_2}_{=0} + \frac{1}{N} \sum_{k=j}^{j+N-1} U_2^T L_{\kappa(k)} U_2, \quad (26)$$

which reveals that dynamics (25) and its property (26) share the same structure as the one of dynamics (9) and its property (11). As a consequence, we may follow exactly the steps of the first part of the proof and obtain a Lyapunov function V_2 satisfying,

$$\underline{c}_2 |\eta_2|^2 \leq V_2(j, \eta_2) \leq \bar{c}_2 |\eta_2|^2, \quad \forall j, \eta_2, \quad (27)$$

$$\Delta V_2(j, \eta_2) \leq -\varepsilon_2 |\eta_2|^2 + \zeta |\eta_2| |d| + \zeta |d|^2, \quad (28)$$

where $\zeta := \lambda_2 + \sigma_2$, and these constant come from similar constructions to the one used to analyze the dynamics of e_1 . Consider now the following overall Lyapunov function:

$$V(e_1, \eta_2) := \theta V_1(e_1) + V_2(\eta_2),$$

where θ will be selected large enough. Combining the bounds in (19) and (28), and recalling that $|d| \leq M_d |e_1|$, we obtain

$$\Delta V \leq -\varepsilon_1 \theta |e_1|^2 - \varepsilon_2 |\eta_2|^2 + \zeta M_d |\eta_2| |e_1| + \zeta M_d^2 |e_1|^2,$$

so that we may pick θ large enough to dominate the last bad quadratic term, and complete squares to dominate the mixed term, thereby obtaining:

$$\Delta V \leq -\frac{\varepsilon_2}{2} (|e_1|^2 + |\eta_2|^2), \quad (29)$$

which implies, by standard discrete-time Lyapunov theory, that there exist positive scalars K_V and $\mu_V < 1$ satisfying:

$$\begin{aligned} |\eta_2(j)|^2 &\leq |e_1(j)|^2 + |\eta_2(j)|^2 \\ &\leq K_V \mu_V^j (|e_1(0)|^2 + |\eta_2(0)|^2). \end{aligned}$$

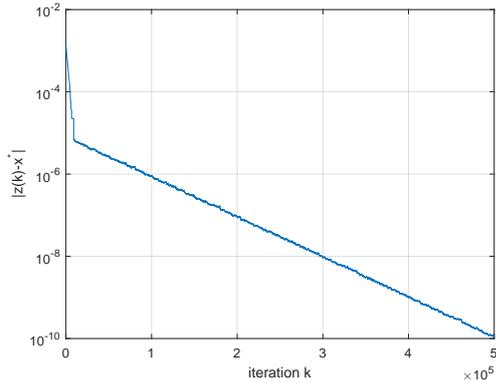


Fig. 1. Magnitude of error for a highly linked network.

Using (21) we obtain $|\eta_2(0)| \leq |e_2(0)|$, because $[n^{-1/2} \mathbf{1} \ U_2]$ is a unitary matrix. Then we may use the initial conditions in (6) and the definitions in (8) to obtain

$$\begin{aligned} |\eta_2(j)|^2 &\leq K_V \mu_V^j (|e_1(0)|^2 + |e_2(0)|^2) \\ &\leq K_V \mu_V^j (|z(0) - b(0) - x^*|^2 + |z(0) - x^*|^2) \\ &\leq 2K_V \mu_V^j |z(0) - x^*|^2. \end{aligned} \quad (30)$$

Let us now consider (21) and (23), which imply

$$|e_2(j)|^2 = |\eta_1(j)|^2 + |\eta_2(j)|^2 = |\eta_2(j)|^2, \quad (31)$$

where we used again the fact that $[n^{-1/2} \mathbf{1} \ U_2]$ is unitary. Then we may concatenate bounds (30) and (31), together with $e_2 = z - x^*$, to prove (13) with $K = \sqrt{2K_V}$ and $\mu = \sqrt{\mu_V}$. \diamond

5. NUMERICAL EXAMPLES

In this section, we apply the proposed algorithm to PageRank problems. Note that in the case of PageRank, the proposed communication scheme requires each page to collect the page rank values of incoming links. This is exactly the same setup discussed in Ishii and Tempo (2010). In our first numerical example, a network of $n = 1000$ pages is considered where each page has between 260 and 347 pages linking to it; i.e., we have a highly linked set of pages. The evolution of the distance to the optimum is depicted in Fig. 1. As expected, one has exponential convergence to the solution x^* and all iterations are consistent; i.e., $\mathbf{1}^T z(k) = 1$ for all k . Next, we consider a network with much less connectivity. More precisely, we again have $n = 1000$ pages but now each page has only between 12 and 41 pages linking to it. The evolution of the magnitude of the error for this network is shown in Fig. 2. Again, as expected, one has exponential convergence to the optimum and all the iterations are consistent. However, the fact that the network now has a much lower level of connectivity leads to a slower convergence rate.

6. CONCLUDING REMARKS

In this paper we considered the problem of solving a set of decentralized linear equalities together with a global consistency constraint. An algorithm is proposed that

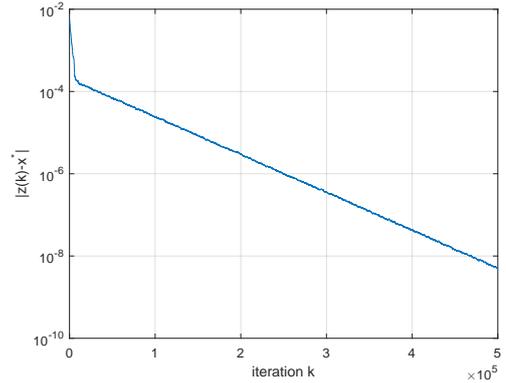


Fig. 2. Magnitude of error for a lowly linked network

i) converges exponentially to the feasible set and ii) at every iteration, the estimate satisfies the global consistency constraint. The proposed algorithm is decentralized and allows for “massive” parallel implementation. More precisely, as long as none of the equalities has common variables they can be updated in parallel. Numerical performance was demonstrated using PageRank examples. Effort is now being put in extended the type of global constraints that can be handled.

ACKNOWLEDGEMENTS

We would like to thank Dr. Roberto Tempo for the very helpful discussions and insights on the PageRank problem.

REFERENCES

- Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks & ISDN Systems*, 30, 107–117.
- Fercoq, O., Akian, M., Bouhtou, M., and Gaubert, S. (2013). Ergodic control and polyhedral approaches to PageRank optimization. *IEEE Transactions on Automatic Control*, 58(1), 134–148.
- Ishii, H. and Tempo, R. (2010). Distributed randomized algorithms for the PageRank computation. *IEEE Transactions on Automatic Control*, 55, 1987–2002.
- Ishii, H. and Tempo, R. (2014). The pagerank problem, multiagent consensus, and web aggregation: A systems and control viewpoint. *IEEE Control Systems*, 34(3), 34–53.
- Kaczmarz, S. (1937). Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin International de l'Academie Polonaise des Sciences*, 355–357.
- Langville, A. and Meyer, C. (2006). *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, Princeton.
- Liu, J. and Wright, S. (2016). An accelerated randomized Kaczmarz algorithm. *Mathematics of Computations*, 85, 153–178.
- Mou, S., Liu, J., and Morse, A.S. (2015). A distributed algorithm for solving a linear algebraic equation. *IEEE Transactions on Automatic Control*, 60(11), 2863–2878.

- Nazin, A., , and Polyak, B. (2011). Randomized algorithm to determine the eigenvector of a stochastic matrix with application to the PageRank problem. *Automation and Remote Control*, 72(2), 342–352.
- Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems (2 ed.)*. SIAM.
- Strohmer, T. and Vershynin, R. (2008). A randomized kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2), 262.
- You, K., Tempo, R., and Qiu, L. (2017). Distributed algorithms for computation of centrality measures in complex networks. *IEEE Transactions on Automatic Control*, To appear, arXiv:1507.01694.