Robust Average Consensus using Total Variation Gossip Algorithm

Walid Ben-Ameur
Télécom SudParis
Institut Mines-Télécom
SAMOVAR, CNRS (UMR 5157)
walid.benameur@it-sudparis.eu

Pascal Bianchi
Télécom ParisTech
Institut Mines-Télécom
LTCI, CNRS (UMR 5141)
pascal.bianchi@telecom-paristech.fr

Jérémie Jakubowicz
Télécom SudParis
Institut Mines-Télécom
SAMOVAR, CNRS (UMR 5157)
jeremie.jakubowicz@it-sudparis.eu

***Abstract*—Consider a connected network of N agents observing N arbitrary samples. We investigate distributed algorithms, also known as *gossip algorithms*, whose aim is to compute the sample average by means of local computations and nearby information sharing between agents. First, we analyze the convergence of some widespread gossip algorithms in the presence of misbehaving (stubborn) agents which permanently introduce some false value inside the distributed averaging process. We show that the network is driven to a state which exclusively depends on the stubborn agents. Second, we introduce a novel gossip algorithm called *Total Variation Gossip Algorithm*. We show that, provided that the sample vector satisfies some regularity condition, the final estimate of the network remains close to the sought consensus, and is unsensitive to large perturbations of stubborn agents. Numerical experiments complete our theoretical results.**

I. Introduction

Algorithms designed to estimate averages over a network in a distributed fashion have been the subject of intense research [1], [2], [3]. They are usually referred to as Gossip algorithms [4], even if the word *gossip* is sometimes used for a broader family of distributed algorithms [4]. One of the most widespread approach to achieve this distributed average computation is through iteration of linear operations mimicking the behaviour of heat equation [3]: at each round, nodes average the values in their neighborhood (including themselves). We will refer to this general strategy as *linear gossip*. Under mild hypotheses, linear gossip can be shown to converge to a state where each node in the network has the same value – this value being the sought network average.

In its most simple form, linear gossip is synchronous: time is slotted and at each slot, all agents wake up and perform their local averaging [1], [2]. But many variants have been studied as well: asynchronous gossip, where some random node wakes up, and then communicates with some neighbors; it can, for instance, broadcast its value to its neighborhood and let its neighborhood perform the averaging, or choose a single random neighbor and both adjust to their common average (a variant known as *random pairwise gossip* [3]). However, most of this works share a common view of the network: all agents show good will. They do not, for instance, deliberately introduce some false value inside the network, or refuse to update their value. There are a few recent work raising the problem of misbehaving agents in the gossip process [5], [6].

Contributions. The contributions of the paper are twofold. First, we study the impact on the network of the presence of stubborn agents, that do not change their minds and keep the same value instead of performing correct averaging with their neighborhood. We assess this impact in both quantitative and qualitative terms. In quantitative terms, we characterize the convergence of synchronous linear gossip in the presence of stubborn agents. We show in particular that a single stubborn agent flips the whole network, in the sense that the whole network end up taking his/her value. We also analyze the impact of several stubborn agents and show that the network is driven by their values. Second we propose a non-linear gossip algorithm which we shall refer to as *Total Variation Gossip Algorithm* (TVGA) which is meant to be more robust to misbehaving agents. Under some regularity conditions, we prove that the algorithm converges to the sought consensus in the ideal case where the network is free from misbehaving agents. We also study the behavior of TVGA in case a stubborn agent perturbates the network. We prove that, irrespective to the magnitude of the perturbation, the final estimate of the network remains close to the sought consensus. Finally, we provide numerical experiments showing the attractive behavior and robustness of TVGA.

The paper is organized as follows. In Section II, some well known fact about linear gossip are recalled. In Section III, standard linear gossip algorithm are studied in the presence of stubborn agents. In Section IV a new gossip algorithm is proposed. In Section V its performance is evaluated in the presence of stubborn agents. In Section VI numerical experiments are conducted to assess its behavior on simulated data and we conclude in Section VII, rejecting technical proofs to the appendix.

II. Framework

The common goal of average gossip algorithms [3], [7] is to estimate the average of values spread over a network. The goal of this section is to formalize precisely this notion and recall some well known results.

The agents of the network are represented by a set V, their ability to communicate with each other is represented by a set of *undirected edges* E: an undirected edge is a pair $\{v, w\}$ with

VALUETOOLS 2012, October 09-12, Cargèse, France

DOI 10.4108/valuetools.2012.250316

$(v,w) \in V^2$, $v \neq w$. The assertion $\{v,w\} \in E$ is sometimes denoted $v \sim w$, represents the fact that agent v and w are able to communicate (they may or may not use this ability), while $v \not\sim w$ means that v and w can never communicate directly with one another. The *undirected graph* $G = (V, E)$ sums up the communication infrastructure of the network.

The information held by the agents is represented by functions $\boldsymbol{x} : V \to \mathbb{R}$, meaning that agent v holds the scalar $\boldsymbol{x}(v)$. For instance if all agents share the same value, *e.g.* 1, the corresponding function: $v \in V \mapsto 1$ is denoted $\mathbf{1}$. By abuse of notations, we consider $\boldsymbol{x}$ as a vector in $\mathbb{R}^N$ with components $\boldsymbol{x}(v)$ in a given order, where $N = |V|$ is the total number of agents in the network. Now, each agent has its own initial information, encoded $\boldsymbol{x}^0$ and wants to share it with the other agents so that anyone can estimate $\frac{1}{N}\sum_{v \in V} \boldsymbol{x}^0(v)$. It is convenient to use the couple of orthogonal projectors $\boldsymbol{J} = \frac{1}{N}\mathbf{1}\mathbf{1}^T$ and $\boldsymbol{J}^{\perp} = \boldsymbol{I} - \boldsymbol{J}$. Using these notations, the goal of any average gossip algorithm is to go from $\boldsymbol{x}^0$ to $\boldsymbol{x}_\infty = \boldsymbol{J}\boldsymbol{x}^0$ using some specific communication protocols we now specify.

The communication protocol used by the agents is *not* encoded by G, instead it has to be *compliant* to G: two agents are allowed to communicate only if they are connected in G. Obviously, there are many ways to satisfy this constraint: agents can communicate *pairwise* or not, *synchronously* or not, etc. Gossip algorithms are iterative algorithms and can be cast in terms of the following update equations:

$$\forall n \geq 0, v \in V,\ \boldsymbol{x}_{n+1}(v) = f_{n+1,v}\left(\boldsymbol{x}_n(v), (\boldsymbol{x}_n(w))_{w \sim v}, \xi_{n+1}\right) \quad (1)$$

where, $\boldsymbol{x}_n(v)$ is seen as a state variable attached to agent v at time n and $(\xi_{n+1})_{n \geq 0}$ are independent and identically distributed random variables.

Remark 1: The fact that $\boldsymbol{x}_{n+1}(v)$ depends only on $\boldsymbol{x}_n(v)$, its neighbors $\boldsymbol{x}_n(w)$ and some independent randomness ξ_{n+1} ensures that the corresponding gossip algorithm is indeed *distributed*. Moreover, it also helps agents to use a limited amount of memory (only the previous state is needed). It is always possible to concatenate the T previous states in a larger state vector, hence this model also encompasses dependence on a fixed number of previous states. Random variables ξ_n usually represent two sources of randomness: noise and asynchronism. Noise alters transmitted values while asynchronism alters order in which operations are performed: which agent initiate a given communication, with which destination, etc.

The most widespread family of average gossip algorithm is the following:

$$\boldsymbol{x}_0(v) = \boldsymbol{x}^0(v) \quad (2)$$

$$\boldsymbol{x}_{n+1}(v) = w_{n+1,v,v}\boldsymbol{x}_n(v) + \sum_{w \sim v} w_{n+1,v,w}\boldsymbol{x}_n(w) \quad (3)$$

where $w_{n,v,w}$ is the (v,w) entry of an $N \times N$ matrix $\boldsymbol{W}_n$.

Assumption 1: Gossip matrices $\boldsymbol{W}_n$ are independent and identically distributed (iid) and such that:

(a) $\boldsymbol{W}_n$ has its entries in $[0,1]$ almost surely.
(b) $\boldsymbol{W}_n\mathbf{1} = \mathbf{1}$ almost surely.
(c) $\boldsymbol{W}_n^T\mathbf{1} = \mathbf{1}$ almost surely.
(d) $\rho(\mathbb{E}[\boldsymbol{W}_n] - \boldsymbol{J}) < 1$.

Remark 2: Even if some authors consider negative weights [3], the first assumption allows to easily interpret W as a weight matrix. Assumption $\boldsymbol{W}_n\mathbf{1} = \mathbf{1}$ is a sanity check: if all agents agree on the value 1 at time $n = 0$ the network should remain unchanged at each iteration. Assumption $\boldsymbol{W}_n^T\mathbf{1} = \mathbf{1}$ asks for a good deal of cooperation in the network: agents have to coordinate their weights. Condition $\rho(\mathbb{E}[\boldsymbol{W}_n] - \boldsymbol{J}) < 1$ ensures connectivity on average, see [3] for details.

Remark 3: As an interesting example of random matrices fulfilling this set of conditions, let us mention Random Pairwise Gossip. At time n, a node v_n is chosen uniformly at random in the network. This node chooses uniformly at random one of its neighbor w_n, then $\boldsymbol{W}_n = \boldsymbol{I} - (\boldsymbol{e}_{v_n} - \boldsymbol{e}_{w_n})(\boldsymbol{e}_{v_n} - \boldsymbol{e}_{w_n})^T/2$. As a byproduct, $\mathbb{E}[\boldsymbol{W}_n]$ is a deterministic matrix the same assumption.

Remark 4: In the particular case where $\boldsymbol{W}_n$ is constantly equal to some matrix $\boldsymbol{W}$, note that this scheme needs a synchronous network since at each time n, all agents update their state simultaneously. We will refer to this case as *synchnronous gossip*.

The following result from [8] addresses convergence.

Theorem 1 ([8]): Under Assumption 1, iterations $\boldsymbol{x}_{n+1} = \boldsymbol{W}_{n+1}\boldsymbol{x}_n$ converge almost surely to the consensus state $\boldsymbol{J}\boldsymbol{x}_0$.

III. Gossip with Stubborn Agents

In this section, we study the usual linear gossip algorithm of the previous section under the assumption that some agents misbehave and show how things can go wrong in that case.

More precisely we assume that some agents, called *stubborn*, following the terminology from [6], never change their state. The rationale behind this model is twofold: either these agents are malfunctioning, or they might want to deliberately pollute or influence the network. What is going to appear from the following analysis, is that a single *stubborn agent* is enough to drive the network to a given prescribed state instead of the sought average state when using standard gossip algorithms.

The algorithm under study is thence the same: $\boldsymbol{x}_{n+1} = \boldsymbol{W}_{n+1}\boldsymbol{x}_n$, with the following assumptions.

Assumption 2 (Network Structure): (a) Vertices set $V = R \cup S$ is the disjoint union of *regular* agents R and *stubborn* agents set S. Both sets are assumed non-empty.

(b) State vector $\boldsymbol{x}_n$ is written in bloc form $\boldsymbol{x}_n = \begin{pmatrix}\boldsymbol{x}_n^R \\ \boldsymbol{x}^S\end{pmatrix}$.

(c) Edge set E contains no edges between stubborn agents: $\forall (s,s') \in S^2, s \neq s' \Rightarrow \{s,s'\} \notin E$.

Note that, by assumption, the state vector of stubborn agents is constant over time, which justify their name. The iterations cannot follow the very same assumptions than in the previous section. Indeed, symmetry or sum preservation asks for network cooperation. And stubborn agents have no reason to cooperate; on the opposite, it is obvious that, in order to remain constant over time, $w_n(s,v) = 0$ for each $v \neq s$. Hence symmetry of $\boldsymbol{W}_n$ cannot be assumed. Instead, we make the following assumptions, distinguishing, as in the previous section, deterministic gossip from random gossip:

Assumption 3 (Random Gossip Structure):
(a) $(\boldsymbol{W}_n)_{n\geq 1}$ forms an independent and identically distributed random matrices sequence.
(b) Matrix $\boldsymbol{W}_n$ has its entries in $[0,1]$ almost surely.
(c) Matrix $\boldsymbol{W}_n$ is written in bloc form:

$$\boldsymbol{W}_n = \begin{pmatrix} \boldsymbol{W}_n^R & \boldsymbol{W}_n^S \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix} \tag{4}$$

(d) $\boldsymbol{W}_n \boldsymbol{1} = \boldsymbol{1}$
(e) Considering the directed edge structure E' defined by: $(v,w) \in E' \Leftrightarrow \mathbb{P}[\boldsymbol{W}_n(v,w) > 0] > 0$, there exists a directed path from each regular node r to some sutbborn node s.

Remark 5: The bloc structure of matrix $\boldsymbol{W}_n$ follows from the constraint that stubborn agents do not change their state over time. Requiring $\boldsymbol{W}_n\boldsymbol{1} = \boldsymbol{1}$ is still a sanity check: if all the agents, *including the stubborn*, agree on some value, it would be inconsistant not to keep it. The last requirement means that the weighted graph induced by $\mathbb{E}[\boldsymbol{W}_n]$ is connected. The last requirement is also very natural, if it were to be not fulfilled, there would exist regular agents that communicate in complete autarky. The last requirement is also very natural. If it were to be not fulfilled, there would exist regular agents that communicate in autarky and cannot be aware of stubborn agents' opinions.

We now address the issue of convergence of such gossip algorithms in the presence of stubborn agents. Let us begin with a technical Lemma.

Lemma 1: Under Assumption 3, for each complex number z with $|z| \geq 1$, matrix $z\boldsymbol{I} - \mathbb{E}[\boldsymbol{W}_n^R]$ is invertible. In particular, in the deterministic case, matrix $\boldsymbol{I} - \boldsymbol{W}^R$ is invertible.

Theorem 2: Under Assumptions 2 and 3 and $\boldsymbol{W}_n = \boldsymbol{W}$, algorithm $\boldsymbol{x}_{n+1} = \boldsymbol{W}\boldsymbol{x}_n$ converges to

$$\boldsymbol{x}_\infty = (\boldsymbol{I} - \boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\boldsymbol{x}^S \tag{5}$$

Consensus is not necessarily reached since $\boldsymbol{x}_\infty$ is not proportional to $\boldsymbol{1}$ as long as stubborn agents disagree with each other. In addition, the limit state does not depend on the initial state, it only depends on the stubborn agents state. In other terms, the stubborn agents solely drive the network, initial opinions of regular agents is lost, even with one single stubborn agent.

It is possible to gain further insight on the limit state since it is basically the solution of Laplace equation with Dirichlet type boundary condition. In particular the following maximum principle holds:

Theorem 3: Under the assumptions of Theorem 2, one has

$$\min_{s\in S} \boldsymbol{x}^S(s) \leq \min_{r\in R} \boldsymbol{x}_\infty(r) \leq \max_{r\in R} \boldsymbol{x}_\infty(r) \leq \max_{s\in S} \boldsymbol{x}^S(s) \tag{6}$$

Let us now study the case of random gossip, *i.e.* $\boldsymbol{W}_n$ is an iid sequence not necessarily constant.

Theorem 4: Under Assumptions 2 and 3, and the assumption that there exists a connected component for E' with at least two nodes in S, algorithm $\boldsymbol{x}_{n+1} = \boldsymbol{W}_{n+1}\boldsymbol{x}_n$, with $\boldsymbol{W}_n$ corresponding to random pairwise matrices with node from R only waking up almost surely does *not* converge.

This situation is very different from the one of the previous section for which all sound gossip algorithms, random or not, lead to the correct consensus. See also a related result in [6].

It appears clearly from the previous results that using standard linear gossip algorithms in the presence of stubborn agents leads to non-desirable results.

IV. Total Variation Gossip

A. Notations

For notational convenience, we equip each edge $\{v,w\} \in E$ with an arbitrary orientation. We denote by $\mathcal{G} = (V,\mathcal{E})$ the corresponding directed graph, where $\mathcal{E} \subset V^2$ is composed of couples of vertices of the form (v,w) (notice the curved letters $\mathcal{G} = (V,\mathcal{E})$ as opposed to the initial undirected graph $G = (V,E)$). The gradient on $\mathcal{G}$ is the linear operator $\nabla : \mathbb{R}^V \to \mathbb{R}^{\mathcal{E}}$ defined for any $\boldsymbol{x} \in \mathbb{R}^V$ by $\nabla\boldsymbol{x} : (v,w) \mapsto \boldsymbol{x}(v) - \boldsymbol{x}(w)$.

We set $\text{sign}\,(x) = 1$ if $x > 0$, $\text{sign}\,(x) = -1$ if $x < 0$ and define $\text{sign}\,(0)$ as an arbitrary value in $[-1,1]$, say 0. If F is a real function on a given Euclidean space and $\boldsymbol{x}$ is a point in that space, we denote by $\partial F(\boldsymbol{x})$ the set of subgradients of F at $\boldsymbol{x}$. Finally, $\|\,.\,\|_1$ and $\|\,.\,\|_2$ respectively stand for the ℓ_1-norm and ℓ_2-norm of vectors.

B. A Distributed Optimization Problem

The above analysis of classical gossip strategies shows that the estimate $\boldsymbol{x}_n$ is not robust to the presence of stubborn agents: the estimate $\boldsymbol{x}_n$ not only fails to converge to the sought consensus $\boldsymbol{J}\boldsymbol{x}^0$ as n tends to infinity, but its asymptotic behavior is exclusively governed by the stubborn agents regardless to the initial value $\boldsymbol{x}^0$. For instance, large values of the stubborn agents can drive the network arbitrarily far away from the sought consensus. The cause of this misbehavior is intuitively related to the fact that stubborn agents permanently inject their value in the network, whereas regular agents tend to forget their initial value as time goes on. In order to design robust gossip algorithms, it is thus legitimate to reintroduce the initial data at each step of the algorithm, hopping this way to balance the overwhelming effect of stubborn agents. Motivated by this remark, the proposed approach seeks to distributively solve the following optimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^V} \frac{\|\boldsymbol{x} - \boldsymbol{x}^0\|_2^2}{2} + \lambda \|\nabla\boldsymbol{x}\|_1 \ . \tag{7}$$

We refer to the second term $\|\nabla\boldsymbol{x}\|_1$ in (7) as the *total variation* of $\boldsymbol{x}$. We recall that:

$$\|\nabla\boldsymbol{x}\|_1 = \sum_{\{v,w\}\in E} |\boldsymbol{x}(v) - \boldsymbol{x}(w)| \ .$$

Hence, $\|\nabla\boldsymbol{x}\|_1$ is a regularization term which penalizes the estimates $\boldsymbol{x}$ that are away from consensus: $\|\nabla\boldsymbol{x}\|_1 \geq 0$ with equality if and only if $\boldsymbol{x} = \boldsymbol{J}\boldsymbol{x}$, as long as G is connected. The second term $\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^0\|_2^2$ in (7) measures the goodness of fit *i.e.*, penalizes the estimates $\boldsymbol{x}$ that are away from the initial value $\boldsymbol{x}_0$. Finally, $\lambda \geq 0$ is an ad-hoc parameter allowing to set the tradeoff between regularization and goodness of fit.

In the sequel, we provide a new gossip algorithm called Total Variation Gossip Algorithm (TVGA) for solving the optimization problem (7) in a distributed fashion. Both synchronous and asynchronous variants of the algorithm are proposed. In order to validate the proposed algorithm, we first analyze its behavior in the *absence* of stubborn agents. We prove that TVGA asymptotically achieves the sought consensus provided that λ is chosen large enough. The study of the behavior of TVGA in the presence of stubborn agents is postponed to the end of the paper.

C. Synchronous TVGA

We now introduce a distributed programming technique for solving (7). Note that the objective function is nondifferentiable but convex, and subgradients are straightforward to compute. For any $\boldsymbol{x} \in \mathbb{R}^V$, we define $\phi(\boldsymbol{x}) \in \mathbb{R}^V$ as the vector whose vth component is given by:

$$\phi_v(\boldsymbol{x}) := \boldsymbol{x}(v) - \boldsymbol{x}^0(v) + \lambda \sum_{w \sim v} \operatorname{sign}(\boldsymbol{x}(v) - \boldsymbol{x}(w)) \ .$$

Define $F(\boldsymbol{x}) := \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^0\|_2^2 + \lambda \|\nabla \boldsymbol{x}\|_1$. The proof of the following lemma is left to the reader.

Lemma 2: For any $\boldsymbol{x} \in \mathbb{R}^V$, $\phi(\boldsymbol{x}) \in \partial F(\boldsymbol{x})$.

It is worth noting that $\phi_v(\boldsymbol{x})$ depends on $\boldsymbol{x}$ only through the elements $\boldsymbol{x}(v)$ and $\boldsymbol{x}(w)$ for w in the neighborhood of v. Therefore, if $\boldsymbol{x}$ represents a vector of local agents' values, a given agent $v \in V$ can compute $\phi_v(\boldsymbol{x})$ by merely collecting the values $\boldsymbol{x}(w)$ of its neighbors w.

We are now in position to state a subgradient descent algorithm for solving (7). In accordance with the generic gossip scheme formalized by (1), the proposed algorithm is an iterative one, for which each node $v \in V$ maintains an estimate $\boldsymbol{x}_n(v)$ of the sought average at each iteration n of the algorithm. Each node v receives the current estimates $\boldsymbol{x}_n(w)$ of its neighbors $w \sim v$ and performs the update:

$$\boldsymbol{x}_{n+1}(v) = \boldsymbol{x}_n(v) - \gamma_n \phi_v(\boldsymbol{x}_n) \tag{8}$$

where $\gamma_n > 0$ is a step size such that the following holds:

Assumption 4: $\sum_n \gamma_n = +\infty$ and $\sum_n \gamma_n^2 < \infty$.

We set the initial value to $\boldsymbol{x}_0 = \boldsymbol{x}^0$ (although $\boldsymbol{x}_0$ could in fact be chosen arbitrarily without changing our results).

Theorem 5: Under Assumption 4, the sequence $(\boldsymbol{x}_n)_{n\geq 1}$ defined by (8) converges to the unique minimizer of (7).

Uniqueness of the minimizer follows from the strict convexity of (7). Next, Theorem 5 follows from Lemma 1 and standard results on subgradient methods (see [9] or references therein). Theorem 5 can also be seen a special case of Theorem 6 proved in the sequel.

Of course, the result stated in Theorem 5 should still be completed by an analysis of the solutions to (7). However, before addressing this point, we first introduce an asynchronous variant of TVGA.

D. Asynchronous TVGA

The algorithm introduced in the previous section is synchronous in the sense that for any iteration n, all vertices of the graph G must be able to simultaneously exchange their values. Here, we extend our algorithm to a less stringent context.

The following asynchronous model is inspired from [3]. We assume that all agents have of independent random clocks driven by a Poisson process (see [3] for details). At the nth time instant, assume that the clock of agent v_n is ticking. Agent v_n becomes active and contacts some other agent w_n randomly selected amongst its neighbors. We shall say that an edge $e \in E$ is *active* at time n if $e = \{v_n, w_n\}$. We shall say that a vertex/agent $u \in V$ is *active* if u belongs to the active edge. Let $p : V \times V \to \mathbb{R}_+$ be a function such that $p_{v,w} = 0$ whenever $v \nsim w$ and $\sum_w p_{v,w} = 1$ for any v. As formally stated by the Assumption below, $p_{v,w}$ represents the probability that a node v, when awake, asks for the value of node w.

Assumption 5: Random variables $(v_n, w_n)_{n\geq 1}$ form an i.i.d. sequence. Random variable v_1 follows the uniform distribution on V. For any $(v, w) \in V \times V$,

$$\mathbb{P}(w_1 = w \,|\, v_1 = v) = p_{v,w} \ .$$

As a remark, one can easily check that a given edge $\{v, w\} \in E$ is active at time n with probability:

$$q_{\{v,w\}} := \frac{p_{v,w} + p_{w,v}}{N} \ ,$$

while a given vertex v is active with probability:

$$\alpha_v := \frac{1 + \sum_w p_{w,v}}{N} \ .$$

For any $(v, w) \in V \times V$, we set:

$$\psi_{v,w}(\boldsymbol{x}) = \boldsymbol{x}(v) - \boldsymbol{x}^0(v) + \lambda \operatorname{sign}(\boldsymbol{x}(v) - \boldsymbol{x}(w)) \ .$$

The asynchronous TVGA is summarized in **Algorithm 1**.

Algorithm 1 Asynchronous TVGA

Initialize: Set $\boldsymbol{x}_0(v) = \boldsymbol{x}^0(v)$ for any $v \in V$.

Iterate: At each time $n = 1, 2, \cdots$

The clock of some agent $v \in V$ is ticking.

Agent v selects an agent w according to the probability measure $(p_{v,w} : w \in V)$

Agents v, w share their current estimate and update:

$$\boldsymbol{x}_{n+1}(v) = \boldsymbol{x}_n(v) - \gamma_n \, \psi_{v,w}(\boldsymbol{x}_n) \, ,$$
$$\boldsymbol{x}_{n+1}(w) = \boldsymbol{x}_n(w) - \gamma_n \, \psi_{w,v}(\boldsymbol{x}_n) \, .$$

For any $u \notin \{v, w\}$, set $\boldsymbol{x}_{n+1}(u) = \boldsymbol{x}_n(u)$.

In order to analyze the convergence of the above algorithm, we need further notations. To any edge $(v, w) \in \mathcal{E}$ of the directed graph $\mathcal{G}$, associate the weight $q_{\{v,w\}}$. Denote by $\overline{\mathcal{G}} = (V, \mathcal{E}, \mathcal{W})$ the corresponding directed weighted graph. Denote by $\overline{\nabla} : \mathbb{R}^V \to \mathbb{R}^{\mathcal{E}}$ the gradient operator on $\overline{\mathcal{G}}$ defined for any

$\boldsymbol{x} \in \mathbb{R}^V$ by $\overline{\nabla}\boldsymbol{x} : (v,w) \mapsto q_{\{v,w\}}(\boldsymbol{x}(v) - \boldsymbol{x}(w))$. Finally, we set $\|\boldsymbol{x}\|_{2,\alpha}^2 := \sum_v \alpha_v \boldsymbol{x}(v)^2$ for any $\boldsymbol{x} \in \mathbb{R}^V$.

Theorem 6: Under Assumptions 4 and 5, the sequence $(\boldsymbol{x}_n)_{n\geq 1}$ generated by **Algorithm 1** converges almost surely to the unique minimizer of the problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^V} \frac{\|\boldsymbol{x}-\boldsymbol{x}^0\|_{2,\alpha}^2}{2} + \lambda \left\|\overline{\nabla}\boldsymbol{x}\right\|_1 \ . \tag{9}$$

Proof: Define $\overline{F}$ as the objective function in (9). It can be shown that the vector $\overline{\phi}(\boldsymbol{x})$ whose vth element is equal to $\overline{\phi}_v(\boldsymbol{x}) := \alpha_v(\boldsymbol{x}(v) - \boldsymbol{x}^0(v)) + \lambda \sum_{w\sim v} q_{\{v,w\}}\mathrm{sign}\,(\boldsymbol{x}(v) - \boldsymbol{x}(w))$ is a subgradient of $\overline{F}$ at point $\boldsymbol{x}$. On the otherhand, it can be shown after some algebra that **Algorithm 1** can be quivalently written as $\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \gamma_n\overline{\phi}(\boldsymbol{x}_n) + \gamma_n\boldsymbol{\xi}_{n+1}$ where $\boldsymbol{\xi}_{n+1}$ is a martingale increment noise. Thus, sequence $\boldsymbol{x}_n$ satisfies a stochastic subgradient recursion whose convergence can be analyzed: due to the lack of space, we only provide a brief sketch of the analysis. Upon noting that:

$$\frac{\boldsymbol{x}_{n+1}-\boldsymbol{x}_n}{\gamma_n} \in -\partial\overline{F}(\boldsymbol{x}_n) + \boldsymbol{\xi}_{n+1} \ ,$$

the results of [10] imply that the continuous-time interpolated process associated with $\boldsymbol{x}_{n+1}$ is *perturbed* solution to the differential inclusion (DI) $\mathrm{d}\boldsymbol{x}(t)/\mathrm{d}t \in -\partial\overline{F}(\boldsymbol{x}(t))$. As $\overline{F}$ is a Lyapunov function for the DI, the limit set of any solution to the latter DI coincides with the minimizer of $\overline{F}$, which concludes the proof by Theorem 5 of [10]. ∎

E. Analysis of the Minimizers

So far, we introduced a novel gossip algorithm in a synchronous (resp. asynchronous) setting, and proved its convergence to the minimizer of (7) and (9) respectively. The next step is to analyze these minimizers. Unfortunately, an exact characterization of the minimizers is a notoriously difficult problem. Nevertheless, we provide a sufficient condition on the initial values $\boldsymbol{x}^0$ in order that the proposed algorithm converges to the sought consensus $\boldsymbol{J}\boldsymbol{x}^0$.

From now on, in order to avoid technical details, we restrict our analysis to the complete graph.

Assumption 6: (a) G is the complete graph.
(b) $p_{v,w} = 1/(N-1)$ for any v, w such that $v \neq w$.
Note that Assumption 6(b) is meaningful only as far as the asynchronous setting is concerned. It means that whenever an agent v wakes up, this agent activates a neighbor w according to the uniform distribution amongst its neighbors.

Definition 1: We say that a vector $\boldsymbol{u} \in \mathbb{R}^V$ is *regular* if there exists a bijection $\sigma : \{1, \cdots, N\} \to V$ such that for any $1 \leq i \leq N-1$:

$$\left|\sum_{j=i+1}^{N} \boldsymbol{u}(\sigma(i)) - \boldsymbol{u}(\sigma(j))\right| \leq (N-i)(N-i+1) \ . \tag{10}$$

We denote by $\mathcal{R}_N \subset \mathbb{R}^V$ the set of regular vectors. We set $\tilde{\lambda} = \lambda/(N-\frac{1}{2})$.

Theorem 7: Let Assumption 5 hold.
(a) If $\boldsymbol{x}^0 \in \lambda\mathcal{R}_N$ then $\boldsymbol{J}\boldsymbol{x}^0$ is the unique minimizer of (7).
(b) If $\boldsymbol{x}^0 \in \tilde{\lambda}\mathcal{R}_N$ then $\boldsymbol{J}\boldsymbol{x}^0$ is the unique minimizer of (9).

Proof: Under Assumption 5(b), it is straightforward to show that $\overline{\nabla}\boldsymbol{x} = 2/(N(N-1))\nabla\boldsymbol{x}$ and $\|\boldsymbol{x}-\boldsymbol{x}^0\|_{2,\alpha}^2 = (N^{-1}+(N-1)^{-1})\|\boldsymbol{x}-\boldsymbol{x}^0\|_2^2$. Thus, Problem (9) is equivalent to Problem (7) only replacing λ with $\tilde{\lambda}$ in the latter. It is thus sufficient to prove the first statement (a) of the Theorem. Recall notation $F(\boldsymbol{x}) := \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{x}^0\|_2^2 + \lambda\, g(\boldsymbol{x})$ where we set $g(\boldsymbol{x}) := \|\nabla\boldsymbol{x}\|_1$. A vector $\boldsymbol{x}$ is a minimizer of (7) if and only if (iff) $0 \in \partial F(\boldsymbol{x})$ which is equivalent to $\boldsymbol{x}^0 - \boldsymbol{x} \in \lambda\partial g(\boldsymbol{x})$. Thus, $\boldsymbol{J}\boldsymbol{x}^0$ is a minimizer iff $\boldsymbol{b} \in \partial g(\boldsymbol{J}\boldsymbol{x}^0)$, where $\boldsymbol{b} = \boldsymbol{J}^\perp\boldsymbol{x}^0/\lambda$. This means that for any $\boldsymbol{h} \in \mathbb{R}^V$, $\langle\boldsymbol{b},\boldsymbol{h}\rangle \leq g(\boldsymbol{J}\boldsymbol{x}^0+\boldsymbol{h}) - g(\boldsymbol{J}\boldsymbol{x}^0)$. The latter inequality simply reads $\langle\boldsymbol{b},\boldsymbol{h}\rangle \leq \|\nabla\boldsymbol{h}\|_1$. Let $\sigma : \{1,\cdots,N\} \to V$ be a function such that (10) holds. For any $i = 1,\cdots,N-1$, define $\boldsymbol{c}_i \in \mathbb{R}^N$ by $\boldsymbol{c}_i(k) = 0$ for $j < i$, $\boldsymbol{c}_i(j) = -1$ for $k > i$ and $\boldsymbol{c}_i(i) = N-i$. The functions $\boldsymbol{d}_i := \boldsymbol{c}_i \circ \sigma^{-1}$ for $i = 1,\cdots,N-1$ form an orthogonal basis of the hyperplane orthogonal to $\mathbf{1}$. As $\boldsymbol{b}$ belongs to this hyperplane, we may thus write the decomposition $\boldsymbol{b} = \sum_i \beta_i\boldsymbol{d}_i$ where coefficients $\beta_i := \langle\boldsymbol{b},\boldsymbol{d}_i\rangle/\left\|\boldsymbol{d}_i\right\|_2^2$. After some algebra, we obtain for all $i = 1,\cdots,N-1$:

$$\beta_i = \frac{1}{(N-i)(N-i+1)}\sum_{j=i+1}^{N}\boldsymbol{b}(\sigma(i)) - \boldsymbol{b}(\sigma(j))$$

Upon noting that $\boldsymbol{b}(\sigma(i)) - \boldsymbol{b}(\sigma(j)) = (\boldsymbol{x}^0(\sigma(i)) - \boldsymbol{x}^0(\sigma(j)))/\lambda$, the hypothesis $(\boldsymbol{x}^0/\lambda) \in \mathcal{R}_N$ along with the choice of σ imply that $|\beta_i| \leq 1$ for any i. We conclude the proof by noting that $\langle\boldsymbol{b},\boldsymbol{h}\rangle = \sum_i\beta_i\langle\boldsymbol{d}_i,\boldsymbol{h}\rangle \leq \sum_i|\langle\boldsymbol{d}_i,\boldsymbol{h}\rangle|$ and using the fact that $\langle\boldsymbol{d}_i,\boldsymbol{h}\rangle = \sum_{j>i}\boldsymbol{h}(\sigma(i)) - \boldsymbol{h}(\sigma(j))$ we finally obtain:

$$\langle\boldsymbol{b},\boldsymbol{h}\rangle \leq \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}|\boldsymbol{h}(\sigma(i)) - \boldsymbol{h}(\sigma(j))| = \|\nabla\boldsymbol{h}\|_1 \ .$$

Thus, the result is proved. ∎

The above Theorem provides a sufficient condition in order that the TVGA converges to the sought value. For a given value of λ, it can be shown that there always exists data $\boldsymbol{x}^0$ for which consensus is not achieved. This can be seen as a price to pay for robustness. Fortunately, since $\lambda\mathcal{R}_N$ tends to $\mathbb{R}^V$ as $\lambda \to \infty$ and since λ can be chosen as large as needed, the set of values $\boldsymbol{x}^0$ successfully handled by our algorithm can be made arbitrarily large. Otherwise stated, whatever the value of $\boldsymbol{x}^0$, TVGA always converges to the sought consensus $\boldsymbol{J}\boldsymbol{x}^0$ provided that λ is large enough.

V. Robustness to Stubborn Agents

In this section, we let Assumption 2 hold true. Some subset $S \subset V$ of the vertices is composed of stubborn agents which do not change their value as time goes on. On the opposite, we assume that regular agents apply the TVGA update described in the previous section. Let us focus on a synchronous setting, and consider the following recursion:

$$\begin{cases} \boldsymbol{x}_{n+1}(v) = \boldsymbol{x}_n(v) - \gamma_n \phi_v(\boldsymbol{x}_n)\,, & \forall v \in R \\ \boldsymbol{x}_{n+1}(v) = \boldsymbol{x}^0(v)\,, & \forall v \in S\ . \end{cases} \tag{11}$$

Let $\mathcal{G}^R$ be the restriction of the graph $\mathcal{G}$ to the regular agents R. We denote by ∇^R the gradient operator on $\mathcal{G}^R$. We introduce $\boldsymbol{x}^{0,R} := (\boldsymbol{x}^0(v)\,:\, v \in R)$ the restriction of $\boldsymbol{x}^0$ to R. Similarly, we define $\boldsymbol{x}_n^R$ as the restriction of $\boldsymbol{x}_n$ to R.

Theorem 8: Consider the algorithm (11). Under Assumption 4, the sequence $(\boldsymbol{x}_n^R)_{n\geq 1}$ converges to the unique minimizer of the problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^R} \frac{\|\boldsymbol{x}-\boldsymbol{x}^{0,R}\|_2^2}{2} + \lambda \left\|\nabla^R \boldsymbol{x}\right\|_1 + \lambda \sum_{\substack{(v,w)\in R\times S \\ v\sim w}} \left|\boldsymbol{x}(v) - \boldsymbol{x}^0(w)\right| \ . \tag{12}$$

Theorem 8 shows that stubborn agents introduce a perturbation in the objective function minimized by our algorithm. The remaining task is therefore to analyze the effect of this perturbation on the minimizer and, hopefully, to show that the final estimate cannot escape too far from the sought consensus, even if stubborn agent contaminate the network with very large values. As already mentioned, a compact characterization of the minimizers of (12) is difficult. We thus restrict our analysis to a simpler scenario, assuming that G is the complete graph and that S is a singleton.

Assumption 7 (Single stubborn agent): The set S is reduced to a singleton $\{s\}$. We define $a := \boldsymbol{x}^0(s)$.

Let $\mathbf{1}_R \in \mathbb{R}^R$ be the constant function equal to 1.

Theorem 9: Assume that $\boldsymbol{x}^{0,R} \in \lambda\mathcal{R}_{N-1}$ and let Assumptions 6(a) and 7 hold true. Define

$$\overline{\boldsymbol{x}}^{0,R} = \frac{1}{N-1}\sum_{v\in R} \boldsymbol{x}^{0,R}(v)$$

and

$$\overline{x} = \begin{cases} a & \text{if } \left|\overline{\boldsymbol{x}}^{0,R} - a\right| \leq \lambda \\ \overline{\boldsymbol{x}}^{0,R} + \lambda & \text{if } \overline{\boldsymbol{x}}^{0,R} < a - \lambda \\ \overline{\boldsymbol{x}}^{0,R} - \lambda & \text{if } \overline{\boldsymbol{x}}^{0,R} > a + \lambda\,. \end{cases}$$

Then, $\overline{x}\mathbf{1}_R$ is the unique minimizer of Problem (12).

Proof: Due to the lack of space, we provide a sketch of proof in the case $\overline{\boldsymbol{x}}^{0,R} < a - \lambda$. Under the stated assumptions, Problem (12) is equivalent to the minimization of the function defined on $\mathbb{R}^R$ by $F(\boldsymbol{x}) := \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{x}^{0,R}\|_2^2 + \lambda\, g(\boldsymbol{x})$ where we set $g(\boldsymbol{x}) := \left\|\nabla^R\boldsymbol{x}\right\|_1 + \sum_{v\in R}|\boldsymbol{x}(v)-a|$. Note that $0 \in \partial F(\boldsymbol{x})$ is equivalent to $\lambda^{-1}(\boldsymbol{x}^{0,R} - \boldsymbol{x}) \in \partial g(\boldsymbol{x})$. Setting $\boldsymbol{b} := \boldsymbol{J}^{\perp}\boldsymbol{x}^0/\lambda$, one can write $\lambda^{-1}(\boldsymbol{x}^{0,R} - \overline{x}\mathbf{1}_R) = \boldsymbol{b} - \mathbf{1}_R$. Thus, $0 \in \partial F(\overline{x}\mathbf{1}_R)$ iff for any $\boldsymbol{h} \in \mathbb{R}^R$, $\langle \boldsymbol{b} - \mathbf{1}_R, \boldsymbol{h}\rangle \leq g(\overline{x}\mathbf{1}_R + \boldsymbol{h}) - g(\overline{x}\mathbf{1}_R)$ which can be restated as:

$$\langle \boldsymbol{b} - \mathbf{1}_R, \boldsymbol{h}\rangle \leq \|\nabla^R\boldsymbol{h}\|_1 + \sum_{v\in R}|\boldsymbol{h}(v) + \overline{x} - a| - \sum_{v\in R}|\overline{x} - a|$$

We have already established in the proof of Theorem 7 that $\langle \boldsymbol{b}, \boldsymbol{h}\rangle \leq \|\nabla\boldsymbol{h}\|_1$. Thus, it is sufficient to prove that for any $\boldsymbol{h}$,

$$-\langle \mathbf{1}_R, \boldsymbol{h}\rangle \leq \sum_{v\in R}|\boldsymbol{h}(v) + \overline{x} - a| - |\overline{x} - a|\ . \tag{13}$$

As $\overline{x} - a < 0$, (13) is equivalent to

$$-\sum_{v\in R}\boldsymbol{h}(v) \leq \sum_{v\in R}|\boldsymbol{h}(v) + \overline{x} - a| + \overline{x} - a\,.$$

The latter inequality is indeed satisfied, so that (13) holds true. The result is proved in the case where $\overline{\boldsymbol{x}}^{0,R} < a - \lambda$. Other cases follow the same steps. ■

We conclude this section with some comments regarding Theorem 9. Provided that the initial data $\boldsymbol{x}^{0,R}$ is regular enough, Theorem 9 implies that, even in the worst case scenario where the stubborn value is overwhelming, the final estimate deviates from at most λ from the sought consensus $\overline{\boldsymbol{x}}^{0,R}$. This is of course unlike the classical gossip algorithms studied in Section III which can be driven arbitrarily far away from the sought consensus.

Theorem 9 also provides insights on the way to select parameter λ. The selection of a small λ allows to reduce the residual error to consensus whereas the selection of a large λ allows to enlarge the set of vectors $\boldsymbol{x}^{0,R}$ which can be successfully handled by our algorithm.

VI. Experiments

In this section, we perform some numerical experiments on the proposed Total Variation Gossip Algorithm.

The underlying network is the complete graph with $N = 20$ agents throughout this section, and the initial data is represented in Figure 1. There is one stubborn agent corresponding to index 1 in all experiments, though it is only for graphical convenience since the complete graph is homogeneous. Figure 1 also represents the true average value over the regular agents. Let us recall that regular agents are all but the first. We focus on synchronous TVGA for simplicity reasons.

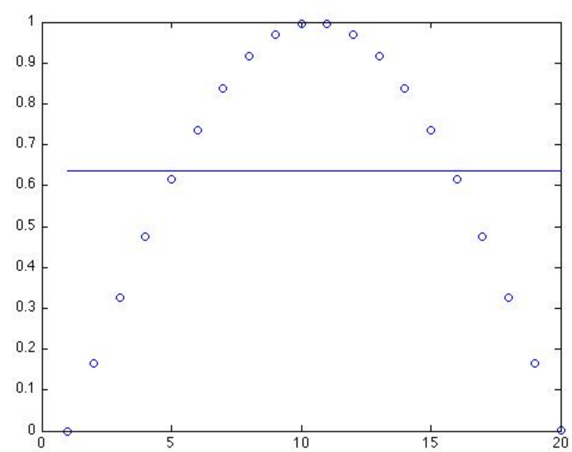

Fig. 1. Initial data (circles) and the corresponding average over regular agents (all but agent 1)

Next we represent iteration 1, 5 and 10 of TVGA on this data (see Figure 2). Observe how the stubborn agent does not evolve and how the disparity diminishes with iterations.

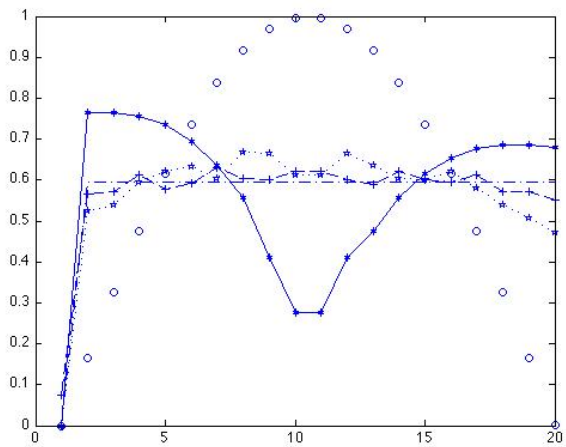

Fig. 2. TVGA first iterations. Iteration 1 in plain line, iteration 5 in dotted line, iteration 10 in dashed line, iteration 375 in dashdot.

Consensus seems already attained at iteration 375. In order to measure how far regular agents are from consensus, we represent the evolution of $\|\boldsymbol{J}^{\perp}\boldsymbol{x}_n^R\|$ with n (see Figure 3). One can see that convergence to consensus takes place fast. To illustrate the (in)sensitivity to outliers of TVGA table I

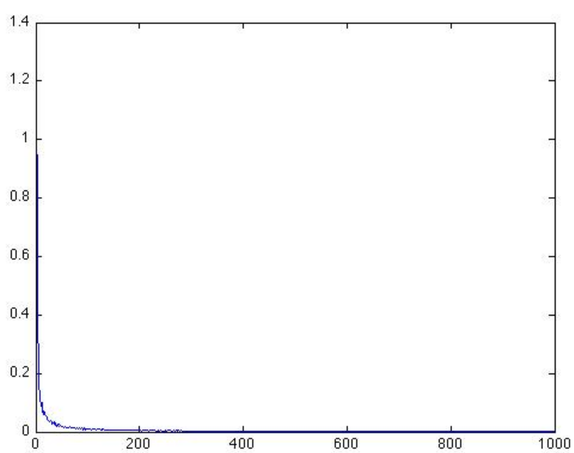

Fig. 3. $n \mapsto \|\boldsymbol{J}^{\perp}\boldsymbol{x}_n^R\|$

represents the average attained over regular agents for several value of the stubborn agent. First it is striking that TVGA is not perturbed by huge values of the stubborn agent. Second, as predicted by Theorem 9, consensus value is separated by $\lambda = .04$ in our experiment from the "true" average value for $\boldsymbol{x}^0$ over the regular agents.

VII. Conclusion

In this paper we analyzed the effect of stubborn agents on standard linear gossip algorithms. It appears that they behave

Stubborn Agent Value	0	-10	-10^9	10	10^9
Consensus Attained	0.60	0.60	0.60	0.68	0.68

TABLE I

TVGA. Compare to "true" average 0.64. Note that this is in perfect accordance with Theorem 9 since $\lambda = .04$ in this experiment.

badly in this scenario. They are affected by even a single stubborn agent, at the point of converging to a value that depends exclusively on stubborn agents. In the same time, we proposed a non-linear gossip algorithm based on total variation regularization that behaves in a very robust way in the presence of stubborn agents and even other forms of perturbation (which is not illustrated in this paper but checked in unreported experiments), while still performing precisely in the absence of perturbation. We provide several quantitative statements to back up these observations. This algorithm appears to be extremely promising and should be analyzed further to confirm these preliminary findings.

References

[1] J. Tsitsiklis, *Problems in Decentralized Decision Making and Computation*, Ph.D. thesis, Massachusetts Institute of Technology, 1984.

[2] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Foundations of Computer Science, Annual IEEE Symposium on*. 2003, IEEE Computer Society.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized Gossip Algorithms," *IEEE Transactions on Inform. Theory*, vol. 52, no. 6, pp. 2508–2530, 2006.

[4] D. Shah, *Gossip algorithms*, vol. 3 of *Foundations and Trends in Networking*, Now Publishers Inc., 2009.

[5] C. Eksin and A. Ribeiro, "Distributed network optimization with heuristic rational agents," *IEEE Trans. Signal Process,(submitted)*, 2011.

[6] D. Acemoglu, G. Como, F. Fagnani, and A. Ozdaglar, "Opinion fluctuations and persistent disagreement in social networks," in *IEEE CDC 2011*, 2011, pp. 2347–2352.

[7] T.C. Aysal, M.E. Yildiz, A.D. Sarwate, and A. Scaglione, "Broadcast Gossip Algorithms for Consensus," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2748–2761, 2009.

[8] A. Tahbaz-Salehi and A. Jadbabaie, "A necessary and sufficient condition for consensus over random networks," *Automatic Control, IEEE Transactions on*, vol. 53, no. 3, pp. 791–795, 2008.

[9] S. Boyd, L. Xiao, and A. Mutapcic, "Subgradient methods," Tech. Rep., October 2003, Lecture Notes.

[10] M. Benaim, J. Hofbauer, and S. Sorin, "Stochastic Approximations and Differential Inclusions," *SIAM Journal on Control and Optimization*, vol. 44, no. 1, pp. 328–348, 2005.

Appendix

Assume that $z\boldsymbol{I} - \boldsymbol{W}^R$ be not invertible, then there exists a vector $\boldsymbol{x} \neq 0$ such that $\boldsymbol{W}^R\boldsymbol{x} = z\boldsymbol{x}$. Let us denote by $v_0 \in R$ a node such that $|\boldsymbol{x}(v)|$ be maximum. We have,

$$z\boldsymbol{x}(v_0) = \sum_{v\in R} w^R(v_0, v)\boldsymbol{x}(v) \ .$$

Hence, $|z||\boldsymbol{x}(v_0)| \leq \sum_{v\in R} w^R(v_0, v)|\boldsymbol{x}(v)|$. By definition of v_0: $|z||\boldsymbol{x}(v_0)| \leq (\sum_{v\in R} w^R(v_0, v))|\boldsymbol{x}(v)|$. Since $\boldsymbol{W}_n\boldsymbol{1} = \boldsymbol{1}$, we have $\sum_{v\in R} w^R(v_0, v) \leq 1$. Thence, necessarily,

$$\sum_{v\in R} w^R(v_0, v) = 1 \ ,$$

which proves that all neighbors of v_0 for E' are in R, which in turn implies that $|z||\boldsymbol{x}(v_0)| = \sum_{v\in R} w^R(v_0,v)|\boldsymbol{x}(v)|$. Moreover, $w^R(v_0,v) > 0$ implies that $\boldsymbol{x}(v) = z\boldsymbol{x}(v_0)$, otherwise the equality $z\boldsymbol{x}(v_0) = \sum_{v\in R} w^R(v_0,v)\boldsymbol{x}(v)$ would be violated. So one can repeat the argument with all the neighbors of v_0: all their own neigbors are necessarily in R and eventually all the connected component containing v_0 lies in R, which completes the proof.

Equation $\boldsymbol{x}_{n+1} = \boldsymbol{W}\boldsymbol{x}_n$ writes $\boldsymbol{x}^R_{n+1} = \boldsymbol{W}^R\boldsymbol{x}^R_n + \boldsymbol{W}^S\boldsymbol{x}^S$. Repeating the argument, we get:

$$\boldsymbol{x}^R_n = (\boldsymbol{W}^R)^n\boldsymbol{x}^R_0 + \sum_{k=0}^{n-1}(\boldsymbol{W}^R)^k\boldsymbol{W}^S\boldsymbol{x}^S$$

Lemma 1 shows that $\rho(W^R) < 1$. From $\rho(W^R) < 1$ we deduce that $(\boldsymbol{W}^R)^n\boldsymbol{x}^R_0$ tends to 0 and $\sum_{k=0}^{n-1}(\boldsymbol{W}^R)^k\boldsymbol{W}^S\boldsymbol{x}^S$ tends to $(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\boldsymbol{x}^S$.

We only prove the inequality $\max_{r\in R}\boldsymbol{x}_\infty(r) \leq \max_{s\in S}\boldsymbol{x}^S(s)$. Matrix $(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}$ has nonnegative entries since it can be written $\sum_{k=0}^{+\infty}(\boldsymbol{W}^R)^k$ and $\boldsymbol{W}^R$ itself has nonnegative entries. Hence $(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S$ preserves coordinate-wise partial order on vectors. And $\boldsymbol{x}^S \preceq (\max_{s\in S}\boldsymbol{x}(s))\cdot\mathbf{1}$. We then deduce:

$$(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\boldsymbol{x}^S \preceq \max_{s\in S}(\boldsymbol{x}(s))\cdot(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\mathbf{1}$$

Now, $\boldsymbol{W}\mathbf{1} = \mathbf{1}$; so $\boldsymbol{W}^R\mathbf{1} + \boldsymbol{W}^S\mathbf{1} = \mathbf{1}$ and

$$(\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\mathbf{1} = (\boldsymbol{I}-\boldsymbol{W}^R)^{-1}(\mathbf{1}-\boldsymbol{W}^R\mathbf{1}) = \mathbf{1}\,,$$

which proves the result.

Let us denote by $\boldsymbol{W}^R = \mathbb{E}[\boldsymbol{W}^R_n]$ and $\boldsymbol{W}^S = \mathbb{E}[\boldsymbol{W}^S_n]$. Consider a random outcome ω that give rise to the matrix sequence $\boldsymbol{W}_n(\omega)$. The symbol ω is dropped for the sake of readability. Assume that the algorithm converge to a vector $\boldsymbol{x}_\infty(\omega)$. The following equality stems directly from $\boldsymbol{x}_{n+1} = \boldsymbol{W}_{n+1}\boldsymbol{x}_n$:

$$\boldsymbol{x}^R_{n+1} - \boldsymbol{x}^R_n = \boldsymbol{\eta}_n + \boldsymbol{\delta}_n + \boldsymbol{h}$$

where $\boldsymbol{\eta}_n = (\boldsymbol{W}^R_n - \boldsymbol{W}^R)\boldsymbol{x}^R_\infty + (\boldsymbol{W}^S_n - \boldsymbol{W}^S)\boldsymbol{x}^S$, $\boldsymbol{\delta}_n = (\boldsymbol{W}^R_n - \boldsymbol{W}^R)(\boldsymbol{x}^R_n - \boldsymbol{x}^R_\infty) + (\boldsymbol{W}^R - \boldsymbol{I})(\boldsymbol{x}^R_n - \boldsymbol{x}^R_\infty)$ and $\boldsymbol{h} = \boldsymbol{W}^S\boldsymbol{x}^S + (\boldsymbol{W}^R - \boldsymbol{I})\boldsymbol{x}^R_\infty$. Now, summing k consecutive terms and dividing by k leads to:

$$\boldsymbol{x}^R_{n+k} - \boldsymbol{x}^R_n = \boldsymbol{h} + \frac{1}{k}\sum_{m=n}^{n+k-1}(\boldsymbol{\eta}_m + \boldsymbol{\delta}_m)$$

Choose n such that $|\boldsymbol{\delta}_m| < \varepsilon$. The strong law of large numbers implies that, for almost all ω, $\frac{1}{k}\sum_{m=n}^{n+k-1}\boldsymbol{\eta}_m$ tends to 0 when k goes to infinity. Indeed, $\boldsymbol{W}_n$ have bounded entries and $\mathbb{E}[\boldsymbol{W}_n - \boldsymbol{W}] = 0$. Obviously $\boldsymbol{x}_{n+k} - \boldsymbol{x}_n$ tends to $\boldsymbol{x}_\infty - \boldsymbol{x}_n$ when k tends to ∞. Which gives,

$$|\boldsymbol{h}| < |\boldsymbol{x}^R_\infty - \boldsymbol{x}^R_n| + \varepsilon\ .$$

Letting n go to infinity gives $|\boldsymbol{h}| \leq \varepsilon$, and since it is true whatever the value of ε, it implies $\boldsymbol{h} = 0$. Which shows that, necessarily

$$\boldsymbol{x}^R_\infty = (\boldsymbol{I}-\boldsymbol{W}^R)^{-1}\boldsymbol{W}^S\boldsymbol{x}^S$$

In order to finish the proof, we study the case of random pairwise gossip on a connected component meeting R and two nodes in S with distinct values. Vector $\begin{pmatrix}\boldsymbol{x}^R_\infty\\ \boldsymbol{x}^S\end{pmatrix}$ is not collinear to $\mathbf{1}$ since $\boldsymbol{x}^S$ has at least two distinct entries. Hence, there exists two neighbors v and w in G such that $v \in R$ and $\boldsymbol{x}_\infty(v) \neq \boldsymbol{x}_\infty(w)$. The probability of selecting edge $\{v,w\}$ is positive. It thence happens for an infinite subset of integers A. For n in that subset A, $(\boldsymbol{W}_n - \boldsymbol{W})\boldsymbol{x}_\infty = \boldsymbol{\xi} \neq 0$. So, for $n \in A$,

$$\boldsymbol{x}^R_{n+1} - \boldsymbol{x}^R_n = \boldsymbol{\xi} + \boldsymbol{\delta}_n$$

A contradiction stems from the fact that A is infinite and both $\boldsymbol{x}^R_{n+1} - \boldsymbol{x}^R_n$ and $\boldsymbol{\delta}_n$ tend to $\mathbf{0}$.