Consensus on the Average on Arbitrary Strongly Connected Digraphs Based on Broadcast Gossip Algorithms

Mauro Franceschelli, Alessandro Giua and Carla Seatzu

Abstract

In this paper we propose a new decentralized algorithm to solve the consensus on the average problem on arbitrary strongly connected digraphs through a gossip algorithm based on broadcasts. We directly extend previous results by not requiring that the digraph is balanced. Our algorithm is an improvement respect to known gossip algorithms based on broadcasts in that the average of the initial state is preserved after each broadcast. The nodes are assumed to know their out-degree anytime they transmit information. The algorithm convergence analysis is preliminary and performance is shown by simulations.

Published as:
M. Franceschelli, A. Giua, C. Seatzu " Consensus on the Average on Arbitrary Strongly Connected Digraphs Based on Broadcast Gossip Algorithms" *1st IFAC Workshop on Estimation and Control of Networked Systems*, (Venice, Italy), Sep. 2009.

M. Franceschelli, A. Giua and C. Seatzu are with the Dept. of Electrical and Electronic Engineering, University of Cagliari, Piazza D'Armi, 09123 Cagliari, Italy. Email: francmau83@hotmail.com, {giua,seatzu}@diee.unica.it.

I. INTRODUCTION

In recent years great effort has been directed to the study of the consensus problem, i.e., [4], [9], [7], [6], [3], [11], and its many applications. One such application, namely wireless sensor networks and in general peer-to-peer networks, is now the focus of a huge amount of research in many disciplines of information technology. The reason why the distributed average problem has received great attention is that it allows to achieve tasks with a minimum overhead of communication since it requires only local information exchange between nodes directly connected, i.e., no routing is needed and so no congestion due to network traffic is generated. One of the networks in which this is desirable is the internet in which the availability of information on the average of local quantities generated by users behavior is of great relevance for statistical analysis, marketing, security and so on. If such objectives can be achieved without unnecessarily overloading network nodes and user bandwidth the relevance of such algorithms becomes clear. A different kind of networks are wireless embedded sensor networks, intended to be composed of a huge number of cheap wireless sensors scattered around a target, be it a city, a forest, a war field or a polluted area. By definition if a wireless sensor is to be cheap it has to consume very little power for achieving its task and to this end the ability to retrieve the average of the measurements with only local packet exchange is of great relevance.

Many previous works on the consensus problem and gossip algorithms [11], [9], [6], [3], [10], [2], [12] are based on bidirectional communications and so represent the network through an undirected graph, possibly with a switching topology. The requirement of bidirectional communications requires synchronization between transmitter and receiver and some overhead required by the communication protocols like acknowledgments. Furthermore even if a set of nodes can communicate between each other, communications are inherently sequential and pairwise if they are not done in the form of broadcasts. An attempt to use broadcasts in the distributed average problem has been made with gossip algorithms, the tradeoff of this approach is that agreement is only reached in the form of a random variable whose expectation corresponds to the average of the initial measurements and whose shape is deeply affected by the sequence in which the nodes perform broadcasts.

A different approach to this issue is the use of distributed Kalman filtering based on consensus [8], [5]. A couple of years ago this problem was solved by adapting the optimal Kalman gain of such filter with respect to the outflow of each node [1] to achieve consensus on the average on arbitrary strongly connected digraphs. The proposed technique was time-variant and proposed as a decentralized iterative algorithm with synchronized updates. In this paper we propose an alternative approach based on gossip.

In [9] the study of consensus on digraphs was motivated by reduction in communication costs, unfortunately the conclusion of the authors was that consensus on the average of achievable only for balanced digraphs, i.e., graphs in which the in-degree and out-degree of each node are the same. Starting from this, we develop a new algorithm, with the same feature of Laplacian-based consensus, that can achieve the same objective for the wider class of arbitrary digraphs. This generalizes the consensus problem and allows a consistent reduction of complexity since it allows the use of only broadcasting as communication mean.

Furthermore wireless sensor networks are usually required to perform tasks more complex then just computing the average of some quantity. We argue that an algorithm that allows consensus on directed graphs can actually be implemented as simple and small "overhead" on normal communication between the sensors. For instance with the ZigBee protocol for wireless networks we have packets with a maximum payload of around 104 bytes, it is clearly too much to just send a scalar integer value of 16 bits. We argue that such consensus protocol could have a more meaningful and real application if thought as network overhead for distributed estimation purposes that does not actual "increase" the load in the network, i.e., since no specific acknowledge or response is required no dedicated communication is required but the usual communication due to data transfer between nodes for other purposes or simply for routing algorithms which periodically check network health or topology. With the previous assumption while the nodes use only mono-directional communications, they always know their out-degree.

II. Problem statement

We model the network of agents as a directed graph $\mathcal{G}(t) = \{V, E(t)\}$, with $V = \{1, \dots, n\}$ the set of vertices that represent the nodes, $E(t) \subseteq \{V \times V\}$ the time varying edge set that encodes the network topology, $(i, j) \in E(t)$ if and only if agent i receives information from agent j at time t. In the following directed edges from j to i are considered to have their "tail" in j and the "head" in i.

The graph can be encoded through its $n \times n$ *adjacency matrix*

$$A(t) = \{a_{i,j}(t)\} \text{ with } a_{i,j}(t) = \begin{cases} 1, & \text{if } (i,j) \in E(t); \\ 0, & \text{otherwise.} \end{cases}$$

The in-degree of a vertex corresponds to the number of "heads" incident in such vertex while the out-degree is the number of "tails" incident on it.

We define the two $n \times n$ matrices

$$\Delta_{in}(t) = \text{diag}\left(\delta_{in,1}(t), \dots, \delta_{in,n}(t)\right)$$

and

$$\Delta_{out}(t) = \text{diag}\left(\delta_{out,1}(t), \dots, \delta_{out,n}(t)\right)$$

where $\delta_{in,i}$ and $\delta_{out,i}$, for $i = 1, \dots, n$, are respectively the in-degree and out-degree of agent i.

The *Laplacian* of a time-varying digraph is defined as

$$\mathcal{L}(t) = \Delta_{in}(t) - A(t). \tag{1}$$

It is a weak diagonal row dominant positive semi-definite matrix. Defining $\mathbf{0}$ and $\mathbf{1}$ column vectors whose n elements are all, respectively, zeros and ones, we have that $\mathcal{L}(t)\mathbf{1} = \mathbf{0}$ by construction.

To each vertex i for $i = 1, \dots, n$ is associated a scalar $x_i(t)$ with an arbitrary initial value $x_i(0) = x_0$.

Furthermore we define the set of neighbors of agent i as $\mathcal{N}_i(t) = \{j : (i, j) \in E\}$ and with $|\mathcal{N}_i(t)|$ its cardinality. We point out that since the graph is directed, node i may be neighbor of node j while node j is not neighbor of node i.

Our objective is to find a decentralized control law that satisfies the network topology constraints given by $\mathcal{G}(t)$ and achieves consensus on the average on the initial states.

III. Consensus on the average on arbitrary digraphs

In our approach we associate to each vertex i for $i = 1, \ldots, n$, other than $x_i(t)$ a companion variable $z_i(t)$ with initial value $z_i(0) = 0$. In the following we study the following gossip algorithm based on mono-directional communications. Each node at each instant of time is then either transmitting information, receiving information or in an idle state.

Transmitter state update, node i

$$\begin{cases} x_i(t+1) = & x_i(t), \\ z_i(t+1) = & 0. \end{cases} \tag{2}$$

Receiver state update, node $j \in \mathcal{N}_i(t)$

$$\begin{cases} x_j(t+1) = & \frac{x_j(t)+x_i(t)}{2} + 0.5 z_j(t) + \frac{z_i(t)}{2\delta_{out,i}(t)}, \\ z_j(t+1) = & \frac{x_j(t)-x_i(t)}{2} + 0.5 z_j(t) + \frac{z_i(t)}{2\delta_{out,i}(t)}. \end{cases} \tag{3}$$

Idle nodes, $k \neq i, k \notin \mathcal{N}_i(t)$

$$\begin{cases} x_k(t+1) = & x_k(t), \\ z_k(t+1) = & z_k(t). \end{cases} \tag{4}$$

This behavior can be explained in simple words.

- *The transmitter node* i broadcasts its state value x_i to all nodes $j \in \mathcal{N}_i$. In doing so, it knows its out-degree and it also broadcasts the value $z_i(t)/\delta_{out,i}(t)$ by dividing the value of the companion variable by the number of nodes that receive the information. The transmitter node i does not change its value of $x_i(t)$ while it resets to 0 the companion variable.
- *The receiver nodes* update their $x_j(t)$ variable by computing the average between their and the received state value. Furthermore they correct their update by a fraction of their companion variable $z_j(t)$. The receiver nodes update their companion variable by adding up several terms, the first term is a compensation to keep constant the average of the network.

Each time node i transmits, we model the interaction topology through a graph $\mathcal{G}_i(t)$ at time t, obtained by graph $\mathcal{G}(t)$ removing all arcs whose tail is not node i. We let $A_i(t)$, $\Delta_{in,i}(t)$ and $\mathcal{L}_i(t)$ denote, respectively, the incidence matrix, the in-degree matrix and the Laplacian of this graph.

Let us define

$$P_i(t) = I - 0.5\mathcal{L}_i(t),$$

$$\hat{\Gamma}_i(t) = \frac{A_i(t)}{2\delta_{out,i}(t)} + 0.5\Delta_{in}(t)$$

and

$$\Gamma_i(t) = \frac{A_i(t)}{2\delta_{out,i}(t)} - 0.5\Delta_{in}(t) + (I - e_i e_i^T),$$

where I si the identity matrix and e_i is the i-th canonical basis vector of dimension n.

If we denote

$$C_i(t) = \begin{bmatrix} P_i(t) & \hat{\Gamma}_i(t) \\ I - P_i(t) & \Gamma_i(t) \end{bmatrix} \tag{5}$$

one can readily observe that under the decentralized state update rule (2), (3) and (4), the system dynamics at time t, is:

$$\begin{bmatrix} x(t+1) \\ z(t+1) \end{bmatrix} = C_i(t) \begin{bmatrix} x(t) \\ z(t) \end{bmatrix}. \tag{6}$$

***Remark** 1:* In this paper it is assumed that at each instant of time each node has a strictly positive probability of broadcasting its state to the neighboring sensors. This assumption is to model the inherent asynchrony of wireless communications between sensor nodes. The results on the convergence properties of algorithms developed with this assumption hold for any deterministic scheduling of the communications between the nodes because the order in which the updates are performed is not relevant to the stability of the equilibrium point of the algorithm.

Furthermore given that the sensor network is distributed in space, any pair of nodes sufficiently far apart can perform a broadcast while not interfering with each other. The inherent parallelism of the network is fully exploited and is expected to greatly improve the convergence time of the proposed algorithm, nonetheless in this paper we focus our attention in studying the stability of the equilibrium point of the algorithm leaving the study of its convergence time to future work. ■

In the following, the dependence of C_i from t will be omitted.

IV. Algorithm Convergence Properties

In this section we study the converge properties of the algorithm. We first characterize the eigenstructure of matrices C_i and then we present a conjecture on the convergence to the consensus.

***Proposition** 1:* C_i is idempotent for any $i = 1, \ldots, n$.

Proof: Using the general identities $A_i^2 = A_i \Delta_{in,i} = 0$ and $\Delta_{in,i} A_i = A_i$, one can readily verify that for all $i = 1, \ldots, n$ it holds $C_i^2 = C_i$. □

Since C_i is idempotent, its eigenvalues are always either 0 or 1. Unfortunately since it is not symmetric, it represents an *oblique* projection which does not result in a contractive matrix in general.

We observe, however, that the system is conservative.

***Proposition** 2:* System (7) evolves on the hyperplane

$$\mathbf{1}^T x(t) + \mathbf{1}^T z(t) = \mathbf{1}^T x(0) + \mathbf{1}^T z(0).$$

Proof: For all $i = 1, \ldots, n$, the row vector $[\mathbf{1}^T \ \mathbf{1}^T]$ is a left eigenvector for matrix C_i associated to eigenvalue 1, because it holds

$$[\mathbf{1}^T \ \mathbf{1}^T] C_i(t) = [\mathbf{1}^T (P_i + I - P_i) \ \mathbf{1}^T (\hat{\Gamma}_i + \Gamma_i)] = [\mathbf{1}^T \ \mathbf{1}^T].$$

□

Since the system is autonomous and the companion initial state can be arbitrary chosen, we select $z(0) = \mathbf{0}$. With such assumption we obtain that the information about the average of the initial state is preserved despite communications are mono-directional and asynchronous. In particular if at any instant of time t we have that $z(t) = \mathbf{0}$, we have

$$\mathbf{1}^T x(t) = \mathbf{1}^T x(0).$$

***Remark** 2:* We point out that Proposition 2 suggests that if noise or some disturbance is added to the system, it results in a shift of the hyperplane on which the system is evolving. In particular it can be shown by simple manipulations that if an external input (noise, disturbance, bias or else) is added to the system model such that:

$$\left[\begin{array}{c} x(t+1) \\ z(t+1) \end{array}\right] = C_i(t) \left[\begin{array}{c} x(t) \\ z(t) \end{array}\right] + \ d(t) \ , \tag{7}$$

where $d(t)$ is a column vector with $2n$ time varying elements, then

$$\mathbf{1}^T x(t) + \mathbf{1}^T z(t) = \mathbf{1}^T x(0) + \mathbf{1}^T z(0) + \mathbf{1}^T \sum_{k=0}^{t} d(k).$$

■

Let us now consider the equilibrium points.

***Proposition** 3:* The consensus state in which $x(t) = \alpha \mathbf{1}$ for some scalar α and $z(t) = \mathbf{0}$ is an equilibrium state for system (7).

Proof: For all $i = 1, \ldots, n$, the column vector $[\mathbf{1}^T \ \mathbf{0}^T]^T$ is a right eigenvector for matrix C_i associated to eigenvalue 1, because it holds

$$C_i(t) \left[\begin{array}{c} \mathbf{1} \\ \mathbf{0} \end{array}\right] = \left[\begin{array}{c} P_i \mathbf{1} \\ (I - P_i)\mathbf{1} \end{array}\right] = \left[\begin{array}{c} \mathbf{1} \\ \mathbf{0} \end{array}\right],$$

due to the Laplacian property $\mathcal{L}\mathbf{1} = \mathbf{0}$. □

We now consider the null space of the consensus matrices.

***Proposition** 4:* For all $i = 1, \ldots, n$, the kernel of C_i has dimension $dim(Ker(C_i)) = |\mathcal{N}_i| + 1$.

Proof: One can readily verify that in matrix C_i the multiplicity of eigenvalue 0 is $|\mathcal{N}_i| + 1$. A set of linearly independent eigenvectors that form a basis of the null space are:

- for $j \in \mathcal{N}_i$ a vector $v_j = [e_j^T \ \ -e_j^T]^T$;
- a vector $\hat{v}_i = [\hat{x}_i^T \ \hat{z}_i^T]^T$ with

$$\hat{x}_i(j) = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise} \end{cases}$$

and

$$\hat{z}_i(j) = \begin{cases} -2\delta_{out,i} & \text{if } j = i, \\ 1 & \text{if } j \in \mathcal{N}_i, \\ 0 & \text{otherwise} \end{cases} .$$

□

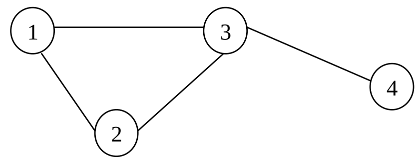

Fig. 1. Network considered in Example 1.

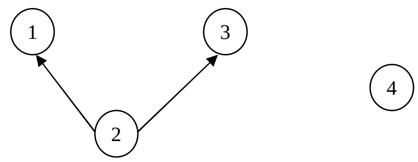

Fig. 2. Interaction topology when node 2 performs a broadcast in Example 1.

***Example** 1:* Let us consider the network in Figure 1. When node 2 performs a broadcast, the interaction topology is represented by a directed graph, shown in Figure 2. The adjacency matrix for the resulting graph is:

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Following our previous definitions, $\delta_{out,2} = 2$, and we have:

$$P_2 = \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\Gamma_2 = \begin{bmatrix} 1/2 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1/4 & 1/2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\hat{\Gamma}_2 = \begin{bmatrix} 1/2 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1/4 & 1/2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

finally

$$C_2 = \left[\begin{array}{cccc|cccc} 1/2 & 1/2 & 0 & 0 & 1/2 & 1/4 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 & 0 & 1/4 & 1/2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1/2 & -1/2 & 0 & 0 & 1/2 & 1/4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1/2 & 1/2 & 0 & 0 & 1/4 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right].$$

By Proposition 4 the following is a basis of linearly independent eigenvectors for the null space:

$$\left[\begin{array}{ccc} v_1 & v_3 & \hat{v}_2 \end{array}\right] = \left[\begin{array}{ccc} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & -4 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{array}\right].$$

■

Now we consider a property that holds for strongly connected graphs.

***Proposition** 5:* If

$$\hat{\mathcal{G}}(t,T) = \bigcup_{k=1}^{T} \mathcal{G}(t+k)$$

is strongly connected, then

$$dim\left(\bigvee_{i=1}^{n} ker(C_i(t))\right) = 2n-1,$$

where $\vee$ denotes the linear combination of vector spaces.

Proof: To show this, let us take the union of all basis vectors for the null spaces of all matrices $C_i(t)$, as defined in the proof of Proposition 4. Since the graph is strongly connected (sufficient condition), each node is at least once a transmitter and at least once a receiver. Thus combining all vectors we obtain the following matrix

$$V = [v_1 \ \cdots \ v_n \ \hat{v}_1 \ \cdots \ \hat{v}_n] = \left[\begin{array}{cc} I & A(t) \\ -I & A(t) - 2\Delta_{out}(t) \end{array}\right].$$

By elementary row operations we show this matrix to be equivalent to

$$\left[\begin{array}{cc} I & A(t) \\ 0 & 2A(t) - 2\Delta_{out}(t) \end{array}\right] = \left[\begin{array}{cc} I & A(t) \\ 0 & -2\mathcal{L}_{out}(t) \end{array}\right]$$

where $\mathcal{L}_{out} = \Delta_{out} - A$ denotes the out-degree Laplacian, whose rank is $n-1$ if the graph is strongly connected.

Thus matrix V has rank $2n-1$ and this proves the result.

$\square$

Thanks to the above propositions the following important result can be proved.

***Proposition** 6:* For a connected graph $\mathcal{G}(t)$ column vector $[\mathbf{1}^T\ \mathbf{0}^T]^T$ is the single equilibrium state for system (7).

Proof: The fact that vector $[\mathbf{1}^T\ \mathbf{0}^T]^T$ is an equilibrium state was shown Proposition 3. Its uniqueness follows from Proposition 5. $\square$

Now we state the main result of this paper for which a formal proof is missing but whose relevance is shown by simulations.

***Conjecture** 1:* If there exists a periodic interval of time T in which the union of the time-varying digraphs

$$\hat{\mathcal{G}}(t,T) = \bigcup_{k=1}^{T} \mathcal{G}(t+k)$$

is strongly connected, if the network evolves according to the state update rule described by (7) with $z(0) = \mathbf{0}$, then:

$$\lim_{t\to\infty} x(t) = \frac{\mathbf{1}\mathbf{1}^T}{n} x(0).$$

$\blacksquare$

The above conjecture is validated by the experiments in the simulations section.

***Remark** 3:* The proposed conjecture, while intuitive and validated by simulation results poses great difficulties in its proof. First, the description of the stability of a switching linear system as system (7) has been treated only for simple cases in which at each instant of time t the system matrix is contractive. Others have used Markov chain theory and applied it to the study of consensus theory. Some other result instead use the Common Lyapunov function approach to study the convergence properties of such system. In our case the system matrix C is not symmetric, it is not weakly diagonal dominant (as the Laplacian matrix, which simplifies considerably the study of its spectrum), nor is with non-negative elements thus invalidating almost all the results about Markov Chain theory. Unfortunately, the problem of deciding wether the random product of a finite set of matrices converges is still an open problem in matrix theory and all the results are either not applicable or relate to classes of matrices not suitable for our purposes. $\blacksquare$

V. SIMULATIONS

In this section we provide some simulations in order to corroborate the algorithm analysis. We consider a network consisting in 20 nodes scattered at random on a terrain of $10\ km^2$. Each node is assumed to be able to communicate through a wireless link with any other node within a range of $40\ m$, the resulting network topology is shown in Figure 3. Each node performs a measurement, for instance the sensed quantity of a polluting chemical in the air. Then each node using only broadcast communications wants to estimate the average quantity of the polluted

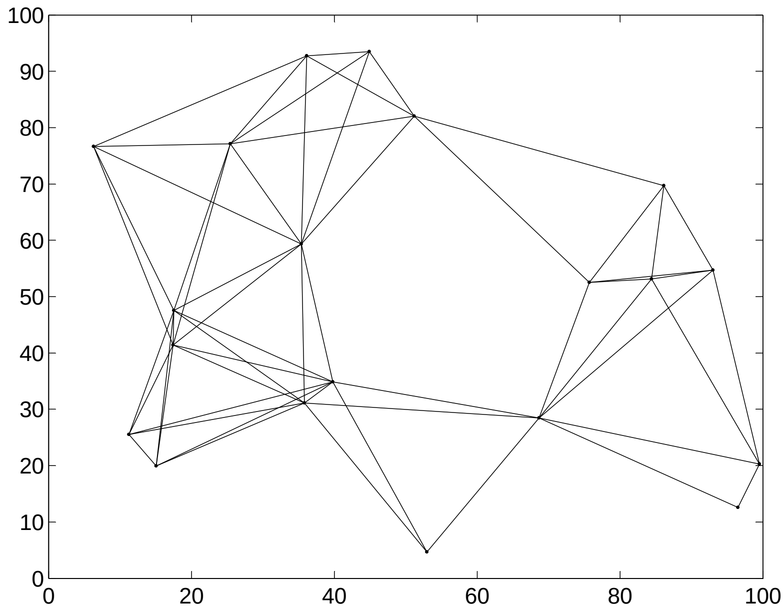

Fig. 3. Topology of the network used for the simulations consisting of 20 nodes with random positions and connected through a proximity graph.

chemical in the region. In the simulation each node has the following initial conditions that for sake of simplicity were chosen randomly between 0 and 1:

$$x(0) = \begin{bmatrix} 0.4552 \\ 0.5036 \\ 0.5704 \\ 0.6412 \\ 0.7854 \\ 0.8220 \\ 0.4822 \\ 0.5844 \\ 0.0130 \\ 0.4338 \\ 0.2396 \\ 0.5397 \\ 0.7003 \\ 0.6065 \\ 0.3457 \\ 0.2594 \\ 0.3905 \\ 0.4151 \\ 0.0542 \\ 0.9566 \end{bmatrix}, \quad z(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

In Figure 4 is shown the evolution of the state trajectory $x_i(t)$ for each node in the network: it can be seen that the algorithm converges toward the average of the initial measurements which

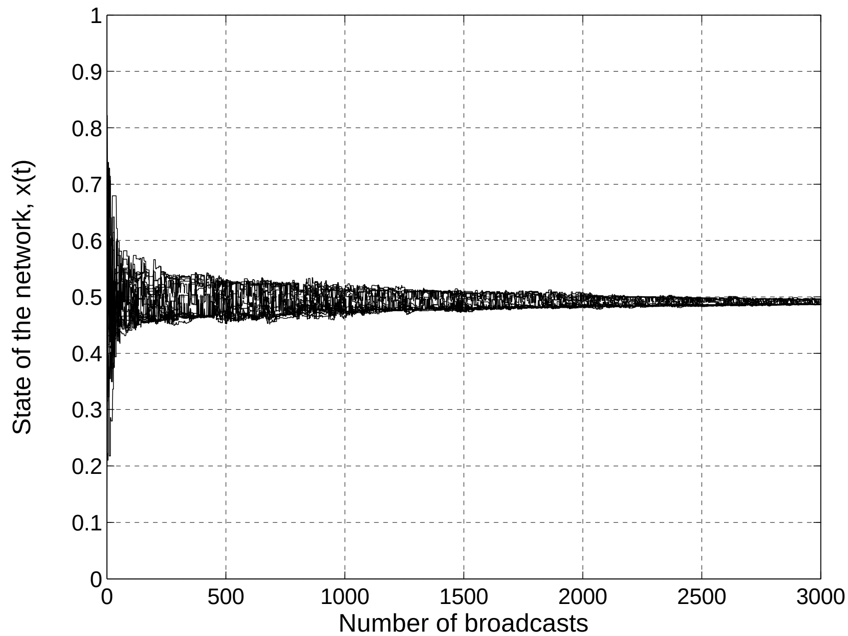

Fig. 4. Trajectory of $x(t)$ for each network node.

is 0.4899.

In Figure 5 is shown the evolution of the companion variables $z_i(t)$ for each node in the network, it can been seen that the initial and final state of such variables is 0. This puts in evidence the fact that such companion variables are used only to make the network converge exactly toward the average of the initial measurement through a gossip algorithm despite the communication graph is directed and the update matrix is not symmetric. The final state of the network after 3000 broadcasts (an average of 150 broadcasts for each node) is:

$$x(3000) = \begin{bmatrix} 0.4959 \\ 0.4959 \\ 0.4962 \\ 0.4868 \\ 0.4867 \\ 0.4955 \\ 0.4869 \\ 0.4968 \\ 0.4939 \\ 0.4864 \\ 0.4868 \\ 0.4877 \\ 0.4873 \\ 0.4962 \\ 0.4865 \\ 0.4870 \\ 0.4867 \\ 0.4870 \\ 0.4870 \\ 0.4869 \end{bmatrix}, \quad z(3000) = \begin{bmatrix} 0.0000 \\ 0 \\ -0.0000 \\ 0 \\ -0.0001 \\ -0.0007 \\ -0.0001 \\ 0.0005 \\ -0.0020 \\ -0.0003 \\ 0.0000 \\ 0.0010 \\ 0.0003 \\ 0 \\ -0.0002 \\ 0.0000 \\ -0.0001 \\ 0.0002 \\ 0.0002 \\ 0.0001 \end{bmatrix}.$$

In particular taking $\bar{x}$ as the average of the initial state, we have that

$$\frac{\|x(3000) - \bar{x}\|}{\|x(0) - \bar{x}\|} = 1.8\%.$$

The algorithm convergence time clearly depends on how well the graph is connected.

The improvement of the proposed algorithm respect to other gossip algorithms is that by using broadcasts the inherent parallelism in a distributed network is fully exploited between all the nodes and not only between nodes not directly connected. This feature is especially relevant in small world networks where few nodes have a very high out-degree.

***Remark** 4:* Further simulations to corroborate our conjecture on the stability of the algorithm have been repeated 1000 times for random initial values at the sensors with a uniform distribution between 0 and 10. Each simulation was initialized with 10 nodes forming a randomly connected network. For each simulation the convergence time was considered to be the number of random broadcasts needed such that the state of the network had reduced its distance from the equilibrium point starting from a random initial condition by a factor of 100, namely:

$$\tau : \forall t \geq \tau \quad \frac{\|x(t) - \frac{\mathbf{1}\mathbf{1}^T}{n} x(0))\|}{\|x(0) - \frac{\mathbf{1}\mathbf{1}^T}{n} x(0))\|} < 0.01.$$

All our simulations achieved convergence with $\tau < 190$ and on average $\bar{\tau} \simeq 93$. Such converge time obviously depend on the properties of the random network topologies with 10 nodes taken in consideration.

■

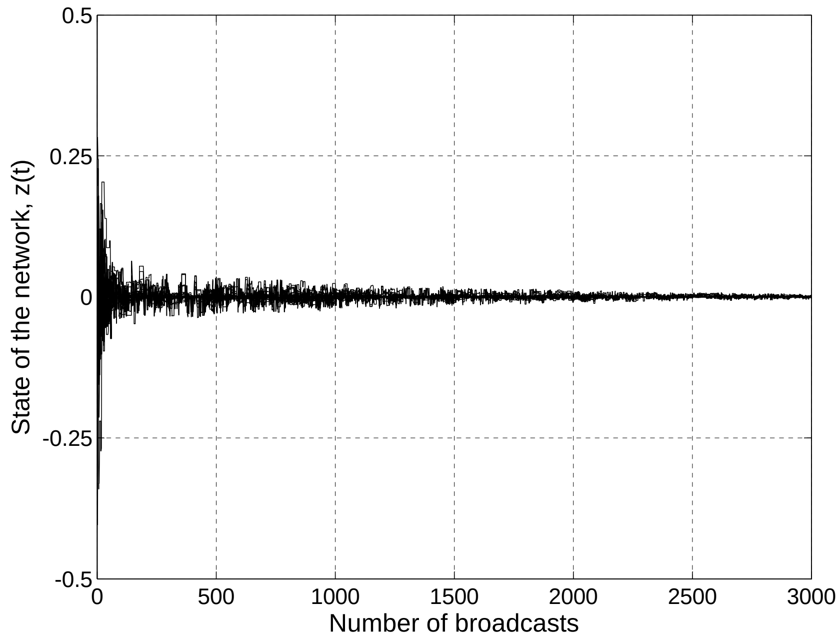

Fig. 5. Trajectory of $z(t)$ for each network node.

We now discuss a comparison between the proposed gossip scheme with a simpler but effective gossip scheme illustrated in [13], [14]. In this works is studied a simple gossip algorithm which can be summarized as follows:

Algorithm 1: **Standard gossip with broadcast**

- At each instant of time a node broadcasts its value to its neighbors.
- If at any time a node listens to a broadcast, it computes the average between its state and the broadcaster state. It then takes this new value as its state.
- Repeat until all the nodes have the same value.

In Figure 6 is shown the evolution of a network executing the standard gossip with broadcast algorithm on the network in Figure 3 with the same initial state as the previous simulations. It can be seen that despite the algorithm converges more quickly respect to our algorithm, the average of the initial state is not preserved as time goes by and so the precision of the final state, where all the nodes estimate 0.5919 instead of 0.4899, greatly depends on the sequence of the broadcasts and on the initial state. It has been shown that in this case the final network state is a random variable whose expectation is the true average of the measurements while its variance decreases as the number of nodes increases. This is a great drawback of the standard gossip with broadcast algorithm since it trades off speed for precision. Unfortunately increasing the number of nodes decreases speed and for a sufficient precision an unfeasible number of nodes might be required.

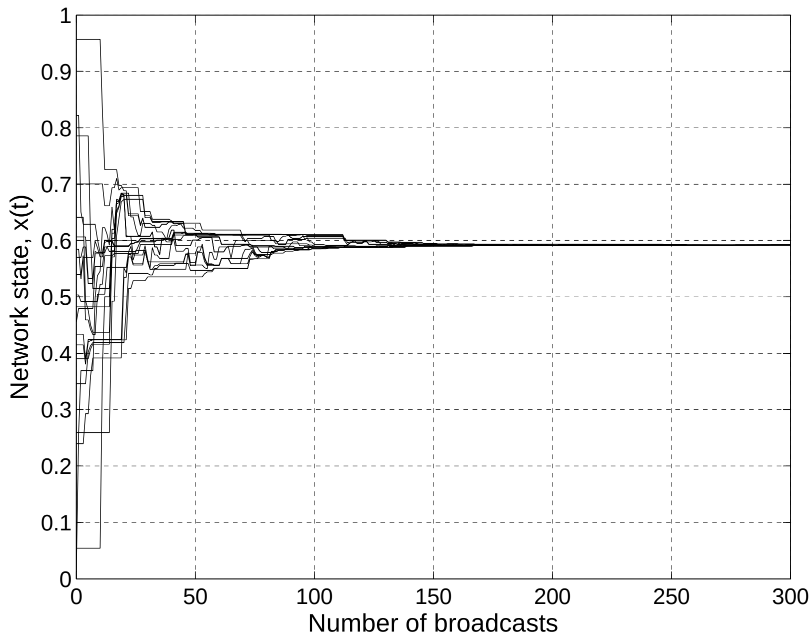

Fig. 6. Trajectory of $x(t)$ for each network node using standard gossip with broadcast.

VI. Conclusion

In this paper we have proposed a novel gossip algorithm based on broadcasts that achieves consensus on the average on arbitrary strongly connected digraphs. The study of the convergence properties is preliminary and convergence is shown by simulations. The proposed algorithm is the first gossip algorithm which preserves the information about the average of the initial state during its execution. A comparison between other gossip algorithms with broadcast has been made and simulations show that despite the convergence time seems greater than other known gossip algorithms, its average preserving properties and precision are undoubtedly superior.

References

[1] M. Alighanbari and J.P. How. An unbiased kalman consensus algorithm. In *American Control Conference*, Minneapolis, Minnesota, USA, 2006.

[2] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Gossip algorithms: design, analysis and applications. In *Proc. IEEE Infocom 2005*, Miami, USA, March 2005.

[3] F. Fagnani and S. Zampieri. Randomized consensus algorithms over large scale networks. *IEEE J. on Selected Areas of Communications*, 2008.

[4] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Autom. Control*, 48:988 –1001, 2003.

[5] Usman A. Khan and Jose M. F. Moura. Distributed kalman filters in sensor networks: Bipartite fusion graphs. In *IEEE/SP 14th Workshop on Statistical Signal Processing*, Madison, Wisconsin, USA, 2007.

[6] X. Lin and S. Boyd. Fast linear iterations for distributed averaging. *Systems and Control Letters*, 53:65–78, 2004.

[7] R. Olfati-Saber. Flocking for multi-agent dynamic system: Algorithms and theory. *IEEE Trans. on Automatic Control*, 51:401–420, 2006.

[8] R. Olfati-Saber. Distributed kalman filtering for sensor networks. In *46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, 2007.

[9] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *Automatic Control, IEEE Transactions on*, 49(9):1520–1533, 2004.

[10] G. Picci and T. Taylor. Almost sure convergence of random gossip algorithms. New Orleans, LA, USA, December 2007.

[11] M. Porfiri and D.J. Stilwell. Consensus seeking over random weighted directed graphs. *Automatic Control, IEEE Transactions on*, 52(9):1767–1773, Sept. 2007.

[12] M.G. Rabbat. On spatial gossip algorithms for distributed averaging. Madison, Wisconsins, US, 2007.

[13] A. Scaglione T.C. Aysal, M.E. Yildiz. Broadcast gossip algorithms. In *ITW '08. IEEE Information Theory Workshop*, 2008.

[14] A.D. Sarwate A. Scaglione T.C. Aysal, M.E. Yildiz. Broadcast gossip algorithms: Design and analysis for consensus. In *47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008.