An asynchronous consensus-based algorithm for estimation from noisy relative measurements

Andrea Carron, Marco Todescato, Ruggero Carli, *Member, IEEE,* and Luca Schenato, *Member, IEEE*

***Abstract*—In this work we address the problem of optimal estimating the position of each agent in a network from relative noisy vectorial distances with its neighbors. Although the problem can be cast as a standard least-squares problem, the main challenge is to devise scalable algorithms that allow each agent to estimate its own position by means of only local communication and bounded complexity, independently of the network size and topology. We propose a consensus-based algorithm with the use of local memory variables which allows asynchronous implementation, has guaranteed exponential convergence to the optimal solution under mild deterministic and randomised communication protocols, and requires minimal packet transmission. In the randomized scenario we then study the rate of convergence in expectation of the estimation error and we argue that it can be used to obtain upper and lower bound for the rate of converge in mean square. In particular, we show that for regular graphs the convergence rate in expectation is reduced by a factor N, which is the number of nodes, which is the same asymptotic degradation of memory-less asynchronous consensus algorithms. Additionally, we show that the asynchronous implementation is also robust to delays and communication failures. We finally complement the analytical results with some numerical simulations comparing the proposed strategy with other algorithms which have been recently proposed in the literature.**

***Index Terms*—Wireless sensor networks, distributed localization algorithms, consensus algorithms**

I. Introduction

The proliferation of relatively inexpensive devices capable of communicating, computing, sensing, interacting with the environment and storing information is promising an unprecedented number of novel applications throughout the cooperation of these devices toward a common goal. These applications include swarm robotics, wireless sensor networks, smart energy grids, smart traffic networks, and smart camera networks. These applications also pose new challenges, of which *scalability* is one of the major ones. Scalability is intended as the ability for an application to continue functioning without any dramatic performance degradation even if the number of devices involved keep increasing. In particular, an application is scalable if it is not necessary to increase HW resources or to adopt a more complex SW algorithms in each device even if the total number of devices increases.

In this work we address the problem of designing algorithms that are capable to reconstruct the optimal estimate of the location of a device from noisy relative measurements from its neighbors in a connected network. In particular, we want to design distributed algorithms that allow each device to reconstruct its own position only from exchanging information with its neighbors, regardless of the size of the network. Moreover, these algorithms must be scalable, i.e. their computational complexity, bandwidth and memory requirements should be independent of the network size. Finally, the estimate provided should asymptotically converge to solution of a centralized optimization problem.

Distributed optimization has being attracting ever growing attention in the past years since many problems in large scale network have been cast as convex optimization problem. In particular, the problem at hand in this work can be cast as the following unconstrained optimization problem:

$$\min_{x_1,\dots,x_N} \sum_{(i,j)\in\mathcal{E}}^{|\mathcal{E}|} f_{ij}(x_i - x_j) \tag{1}$$

where $x_i \in \mathbb{R}^\ell$, $\mathcal{E}$ represents all the pair of nodes for which are available relative measurements and f_{ij} are convex functions. Many problems can be written in this framework such as sensor localization [1], [2], sensor calibration [3], clock synchronization [4] and camera localization [5], [6]. For example, in the context of localization from vectorial relative distances in a plane, the cost function f_{ij} are given by:

$$f_{ij}(x_i - x_j) = \|x_i - x_j - z_{ij}\|^2$$

where $z_{ij} \in \mathbb{R}^\ell$ is the noisy measurement of the relative (vector) distance of node i from node j. As a consequence, the optimization problem in Eqn. (1) becomes a distributed least-square problem. Several scalable distributed solutions to this problem are already available in the literature. In [1], [2] the authors propose a distributed Jacobi solution based on a synchronous implementation, which was later extended to account for asynchronous communication and packet losses [7]. Differently, in [3] a broadcast consensus-based algorithm, which is suitable for asynchronous implementation, is proposed but the local estimates do not converge and exhibits an oscillatory behaviour around the optimal value. A similar approach has been proposed in [8], [9] where the local ergodic average of the gossip asynchronous algorithm is proved to converge to the optimal value as $1/k$, where k is the number of iterations. An alternative approach based on the Kaczmarz method for the solution of general linear systems has been suggested in [10], however a practical asynchronous implementation for distributed localization from relative measurements which satisfies the specific edge and

This work is supported by the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n. 257462 HYCON2 Network of excellence.

A. Carron, M. Todescato, R. Carli and L. Schenato are with the Department of Information Engineering, University of Padova, Via Gradenigo 6/a, 35131 Padova, Italy {carronan|todescat|carlirug|schenato}@dei.unipd.it.

node activation probabilities dictated by the algorithm, is not given, moreover, no robustness analysis in terms of delays is provided. Finally, in [11] the authors propose an asynchronous strategy based on local greedy minimization which requires a substantial coordination and a large number of exchanged messages.

The main contribution of this work is to propose a novel asynchronous algorithm whose main idea consists in casting the estimation problem as a consensus problem under some suitable changes of coordinates, and then to add some extra memory variables at each node to keep track of the estimated location of its neighbors, i.e. the nodes from which they collected the relative distance measurements. Estimates of these local variables eventually converge to the estimates of the neighbors, thus guaranteeing the convergence of the whole algorithm, at the price of some delay. This strategy has several relevant advantages, namely (i) is scalable, (ii) has proven exponential rate of convergence under mild assumptions, (iii) is robust to packet losses and delays, and (iv) requires the transmission of a single communication packet per each iteration. This last feature is particularly relevant for Wireless Sensor Networks (WSNs) applications since agents have limited energy budget and communication is more expensive than computation from an energy point of view. We also study the performance of the proposed algorithm in terms of the convergence rate. This task is particularly challenging since the proposed algorithm turns out to be a higher order consensus algorithm, for which little analytic tools are available. In fact, the few works available in the literature which address the rate of convergence of randomized higher order consensus algorithms are limited to the convergence in expectation [12]. In this work, we exactly compute the rate of convergence in expectation of our algorithm for regular graphs, and through extensive numerical simulations we conjecture that it also provides upper bound for rate of convergence in mean square. Moreover, we show that, asymptotically, such rate of convergence in expectation is reduced by a factor N, where N is the number of nodes, which is the same of standard memoryless asynchronous consensus algorithms, thus implying that asymptotically in N the reduction of rate of convergence due to memory is negligible.

We also prove the convergence of the proposed algorithm when bounded delays and packet losses are present, thus making it particularly suitable for applications using lossy wireless communication. We finally complement the theoretical results with some numerical simulations which show that the proposed algorithm has a performance in terms of rate of convergence per iteration which is slightly slower than the fastest algorithms available in the literature. However, it greatly outperforms them if rate of convergence is computed in terms of number of exchanged messages, i.e, the estimation error obtained by sending a fixed number of packets is much lower for our proposed algorithm than the other algorithms available in the literature.

The paper is organized as follows. In Section II we formulate the problem we aim at solving. In Section III we introduce the synchronous consensus-based algorithm (denoted as s-CL). In Section IV we propose a more realistic asynchronous implementation of the s-CL algorithm (denoted as a-CL). In Section V we establish the convergence of the a-CL algorithm and we provide some bounds on the rate of convergence in mean-square. In Section VI we show that the a-CL algorithm is robust to delays and communication failures. In Section VII we provide some numerical results comparing the a-CL algorithm to other strategies recently proposed in the literature. Finally in Section VIII we gather our conclusions.

A. Mathematical preliminaries

Before proceeding, we collect some useful definitions and notations. In this paper, $\mathcal{G} = (V, \mathcal{E})$ denotes an directed graph where $V = \{1, ..., N\}$ is the set of vertices and $\mathcal{E}$ is the set of directed edges, i.e., a subset of $V \times V$. More precisely the edge (i, j) is incident on node i and node j and is assumed to be directed away from i and directed toward j. The graph $\mathcal{G}$ is said to be bidirected if $(i, j) \in \mathcal{E}$ implies $(j, i) \in \mathcal{E}$.

Given a directed graph $\mathcal{G} = (V, \mathcal{E})$, a directed path in $\mathcal{G}$ consists of a sequence of vertices $(i_1, i_2, \dots, i_r)$ such that $(i_j, i_j + 1) \in \mathcal{E}$ for every $j \in \{1, \dots, r-1\}$. An undirected path in $\mathcal{G}$ consists of a sequence of vertices $(i_1, i_2, \dots, i_r)$ such that either $(i_j, i_{j+1}) \in \mathcal{E}$ or $(i_{j+1}, i_j) \in \mathcal{E}$ for every $j \in \{1, \dots, r-1\}$[1]. The directed graph $\mathcal{G}$ is said to be *strongly connected* (resp. *weakly connected*) if for any pair of vertices (i, j) there exists a directed path (resp. undirected path) connecting i to j. Given the directed graph $\mathcal{G}$, the set of neighbors of node i, denoted by $\mathcal{N}_i$, is given by $\mathcal{N}_i = \{j \in V \,|\, (i, j) \in \mathcal{E}\}$. A directed graph is said to be regular if all the nodes have the same number of neighbors.

Given a directed graph $\mathcal{G} = (V, \mathcal{E})$ with $|\mathcal{E}| = M$ let the *incidence matrix* $A \in \mathbb{R}^{M \times N}$ of $\mathcal{G}$ be defined as $A = [a_{ei}]$, where $a_{ei} = 1, -1, 0$, if edge e is incident on node i and directed away from it, is incident on node i and directed toward it, or is not incident on node i, respectively.

Let $\mathbf{1}_N$ be the N-dimensional column vector with all components equal to one. If there is no risk of confusion we will drop the subscript N. Given a matrix B with $B^\dagger$ we denote its pseudo-inverse. Given a vector v with v^T we denote its transpose. A matrix $P \in \mathbb{R}^{N \times N}$ is said to be stochastic if all its elements are nonnegative and $P\mathbf{1} = \mathbf{1}$. Moreover it is said to be doubly stochastic if it is stochastic and, additionally, $\mathbf{1}^T P = \mathbf{1}^T$. A stocastic matrix P is primitive if it has only one eigenvalue equal to 1 and all the other eigenvalues are strictly inside the unitary circle. With the simbol $\rho_{ess}(P)$ we denote the essential spectral radius of P (see [13]), namely, the second largest, in absolute value eigenvalue of P.

The symbol $\mathbb{E}$ denotes the expectation operator. Given two functions $f, g : \mathbb{N} \longmapsto \mathbb{R}$, we say that $f \in o(g)$ if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$.

II. Problem Formulation

The problem we deal with is that of estimating N variables $x_1, \dots, x_N$ from noisy measurements of the form

$$z_{ij} := x_i - x_j + n_{ij}, \qquad i, j \in \{1, \dots, N\}, \tag{2}$$

[1]Basically, an *undirected path* is a path from a node to another node that does not respect the orientation of the edges.

where n_{ij} is zero-mean measurement noise. Though the variables are often vector-valued, for simplicity, in this paper we assume that $x_i \in \mathbb{R}$, $i \in \{1,\dots,N\}$. This estimation problem can be naturally associated with a *measurement graph* $\mathcal{G}_m = (V;\mathcal{E}_m)$. The vertex set V of the measurement graph consists of the set of nodes $V = \{1,\dots,N\}$ where N is the number of nodes, while its edge set $\mathcal{E}_m$ consists of all of the ordered pairs of nodes (i,j) such that a noisy measurement of the form (2) between i and j is available to node i. The measurement errors on distinct edges are assumed uncorrelated. The measurement graph $\mathcal{G}_m$ is a directed graph since $(i,j) \in \mathcal{E}_m$ implies the measurement z_{ij} is available to node i, while $(j,i) \in \mathcal{E}_m$ implies the measurement z_{ji} is available to node j, and these two are in general distinct.

Next we formally state the problem we aim at solving. Let $\mathbf{x} \in \mathbb{R}^N$ be the vector obtained stacking together all the variables $x_1,\dots,x_N$, i.e., $\mathbf{x} = [x_1,\dots,x_N]^T$, and let $\mathbf{z} \in \mathbb{R}^M$ and $\mathbf{n} \in \mathbb{R}^M$, where $M = |\mathcal{E}_m|$, be the vectors obtained stacking together all the measurements z_{ij} and the noises n_{ij}, respectively. Additionally, let $R_{ij} > 0$ denote the covariance of the zero mean error n_{ij}, i..e, $R_{ij} = \mathbb{E}[n_{ij}^2]$, and let $R \in \mathbb{R}^{M\times M}$ be the diagonal matrix collecting in its diagonal the covariances of the noises n_{ij}, $(i,j) \in \mathcal{E}$, i.e., $R = \mathbb{E}[\mathbf{n}\mathbf{n}^{\mathbf{T}}]$. Observe that Eqn. (2) can be rewritten in a vector form as

$$\mathbf{z} = A\mathbf{x} + \mathbf{n}$$

Now, define the set

$$\chi := \underset{\mathbf{x}\in\mathbb{R}^N}{\operatorname{argmin}} (\mathbf{z} - A\mathbf{x})^T R^{-1} (\mathbf{z} - A\mathbf{x}).$$

The goal is to construct an optimal estimate $\mathbf{x}_{\text{opt}}$ of $\mathbf{x}$ in a least square sense, namely, to compute

$$\mathbf{x}_{\text{opt}} \in \chi \tag{3}$$

Assume the measurement graph $\mathcal{G}_m$ to be *weakly connected*, then it is well known (see [2]) that

$$\chi = \left\{ \left(A^T R^{-1} A\right)^{\dagger} A^T R^{-1} \mathbf{z} + \alpha \mathbf{1} \right\}.$$

Moreover let

$$\mathbf{x}^*_{\text{opt}} = \left(A^T R^{-1} A\right)^{\dagger} A^T R^{-1} \mathbf{z},$$

then $\mathbf{x}^*_{\text{opt}}$ is the minimum norm solution of (3), i.e.,

$$\mathbf{x}^*_{\text{opt}} = \min_{\mathbf{x}_{\text{opt}}\in\chi} \| \mathbf{x}_{\text{opt}} \|$$

The matrix $A^T R^{-1} A$ is called in literature the *Weighted Generalized Grounded Laplacian* [2].

Remark II.1 Observe that, just with relative measurements, determining the x_i's is only possible up to an additive constant. This ambiguity might be avoided by assuming that a node (say node 1) is used as reference node, i.e., $x_1 = 0$.

III. A SYNCHRONOUS DISTRIBUTED CONSENSUS BASED SOLUTION

To compute an optimal estimate $\mathbf{x}_{\text{opt}}$ directly, one needs all the measurements and their covariances $(\mathbf{z}, R)$, and the topology of the measurement graph $\mathcal{G}_m$. In this section the goal is to compute the optimal solution in a distributed fashion, employing only local communications. In particular we assume that a node i and another node j can communicate with each other if either $(i,j) \in \mathcal{E}_m$ or $(j,i) \in \mathcal{E}_m$. Accordingly, we introduce the *communication graph* $\mathcal{G}_c(V,\mathcal{E}_c)$, where $(i,j) \in \mathcal{E}_c$ if either $(i,j) \in \mathcal{E}_m$ or $(j,i) \in \mathcal{E}_m$. Observe that, if $(i,j) \in \mathcal{E}_c$ then also $(j,i) \in \mathcal{E}_c$, namely, $\mathcal{G}_c$ is a bidirected graph. From now on, $\mathcal{N}_i$ denotes the set of neighbors of node i in the communication graph $\mathcal{G}_c(V,\mathcal{E}_c)$.

In what follows we introduce a distributed solution which is based on standard linear consensus algorithm. A discussion of the linear consensus algorithm can be found in the review papers [14], [15], hence we refrain from describing it here. Instead we make the presentation of the algorithm self-contained. Firstly, we assume that the communications among the nodes are synchronous, namely, all nodes perform their transmissions and updates at the same instant, and design the algorithm for this scenario. We refer to this algorithm as the *synchronous consensus-based localization* algorithm (denoted hereafter as s-CL algorithm). In section IV we modify the s-CL algorithm to make it suitable to asynchronous communications. We assume that before running the s-CL algorithm, the nodes exchange with their neighbors their relative measurements as well as the associated covariances. So every node has access to the measurements on the edges that are incident to it, whether the edge is directed to or away from it. Each node uses the measurements obtained initially for all future computations. The s-CL algorithm is formally described as follows.

Processor states: For $i \in \{1,\dots,N\}$, node i stores in memory the measurements $\{z_{ij}, (i,j) \in \mathcal{E}_m\}$, and $\{z_{ji}, (j,i) \in \mathcal{E}_m\}$, and the associated covariances $\{R_{ij}, (i,j) \in \mathcal{E}_m\}$ and $\{R_{ji}, (j,i) \in \mathcal{E}_m\}$. Moreover node i stores in memory an estimate $\hat{x}_i$ of x_i.

Initialization: For $i \in \{1,\dots,N\}$, node i initializes its estimate $\hat{x}_i(0)$ to any arbitrary value.

Transmission iteration: For $k \in \mathbb{N}$, at the start of the $(k+1)$- th iteration of the algorithm, node i transmits its estimate $\hat{x}_i(k)$ to all its neighbors. It also gathers the k-th estimates of its neighbors, $\hat{x}_j(k)$, $j \in \mathcal{N}_i$.

Update iteration: For $k \in \mathbb{N}$, node i, $i \in \{1,\dots,N\}$, based on the information received from its neighbors, updates its estimate as follows

$$\hat{x}_i(k+1) := p_{ii}\,\hat{x}_i(k) + \sum_{j\in\mathcal{N}_i} p_{ij}\,\hat{x}_j(k) + b_i$$

where

$$b_i = \epsilon \sum_{(i,j)\in\mathcal{E}_m} R_{ij}^{-1} z_{ij} - \epsilon \sum_{(j,i)\in\mathcal{E}_m} R_{ji}^{-1} z_{ji}$$

and where

$$p_{ij} = \begin{cases} \epsilon(R_{ij}^{-1} + R_{ji}^{-1}) & \text{if } (i,j)\in\mathcal{E}_m \text{ and } (j,i)\in\mathcal{E}_m \\ \epsilon R_{ij}^{-1} & \text{if } (i,j)\in\mathcal{E}_m \text{ and } (j,i)\notin\mathcal{E}_m \\ \epsilon R_{ji}^{-1} & \text{if } (j,i)\in\mathcal{E}_m \text{ and } (i,j)\notin\mathcal{E}_m \end{cases}$$

and

$$p_{ii} = 1 - \sum_{j\in\mathcal{N}_i} p_{ij}$$

being ϵ a positive constant *a-priori* assigned to the nodes.

Now, let $P \in R^{N\times N}$ be the matrix defined by the weights p_{ij} above introduced. One can see that such matrix P is equal to

$$P = I - \epsilon A^T R^{-1} A.$$

Moreover let

$$b = \epsilon A^T R^{-1} \mathbf{z},$$

and let

$$\hat{\mathbf{x}}(k) = [\hat{x}_1(k), \ldots, \hat{x}_N(k)]^T.$$

Then the s-CL algorithm can be written in a compact form as

$$\hat{\mathbf{x}}(k+1) = P\hat{\mathbf{x}}(k) + b$$

To characterize the convergence properties of the s-CL algorithm, we next introduce two definitions and a crucial property of the matrix P. First, let $d_{max} = \max\{|\mathcal{N}_i|, i \in \{1,\ldots,N\}\}$. Second, let $R_{min} = \min\{R_{ij}, (i,j) \in \mathcal{E}_m\}$. Observe that, if $0 < \epsilon < 1/(2d_{\max}R_{min}^{-1})$, then the matrix P is stochastic and, if, in addition, the measurement graph $\mathcal{G}_m$ is weakly connected, and, in turn, the communication graph $\mathcal{G}_c$ is strongly connected, then the matrix P is primitive. We have the following Proposition.

Proposition III.1 *Consider the s-CL algorithm running over a* weakly connected *measurement graph $\mathcal{G}_m$. Let ϵ be such that $0 < \epsilon < 1/(2d_{max}R_{min}^{-1})$. Moreover let $\hat{x}_i$, $i \in \{1,\ldots,N\}$, be initialized to any real number. Then the following two facts hold true*

(i) *the evolution $k \to \hat{\mathbf{x}}(k)$ asymptotically converges to an optimal estimate $\mathbf{x}_{opt} \in \chi$, i.e., there exists $\alpha \in \mathbb{R}$, such that*

$$\lim_{k\to\infty} \hat{\mathbf{x}}(k) = \mathbf{x}^*_{opt} + \alpha\mathbf{1};$$

where α linearly depends on $\hat{\mathbf{x}}(0)$.

(ii) *the convergence is exponential, namely, there exists $C > 0, \rho_{ess} < 1$ such that*

$$\|\hat{\mathbf{x}}(k) - \left(\mathbf{x}^*_{opt} + \alpha\mathbf{1}\right)\| \leq C\rho^k_{ess}(P)\,\|\hat{\mathbf{x}}(0) - \left(\mathbf{x}^*_{opt} + \alpha\mathbf{1}\right)\|.$$

Proof: We start by proving item (i). Let us define the change of variable $\xi = \hat{\mathbf{x}} - \mathbf{x}^*_{\text{opt}}$. Since $\mathbf{x}^*_{\text{opt}} = P\mathbf{x}^*_{\text{opt}} + b$, it is possible to write

$$\begin{aligned}\hat{\mathbf{x}}(k+1) - x^*_{\text{opt}} &= P\hat{\mathbf{x}}(k) + b - \mathbf{x}^*_{\text{opt}} \\ &= P\hat{\mathbf{x}}(k) + b - (P\mathbf{x}^*_{\text{opt}} + b) \\ &= P(\hat{\mathbf{x}}(k) - \mathbf{x}^*_{\text{opt}})\end{aligned}$$

and, in turn, $\xi(k+1) = P\xi(k)$. This equation describes the iteration of the classical consensus algorithm. Since P is a primitive doubly stochastic matrix, we have that

$$\xi(k) \to \frac{\mathbf{1}\mathbf{1}^T}{N}\xi(0)$$

where $\xi(0) = \hat{\mathbf{x}}(0) - \mathbf{x}^*_{\text{opt}}$. This implies that

$$\hat{\mathbf{x}}(k) \to \mathbf{x}^*_{\text{opt}} + \frac{\mathbf{1}\mathbf{1}^T}{N}\hat{\mathbf{x}}(0) - \frac{\mathbf{1}\mathbf{1}^T}{N}\mathbf{x}^*_{\text{opt}}$$

The fact that $\frac{\mathbf{1}\mathbf{1}^T}{N}\mathbf{x}^*_{\text{opt}} = 0$ concludes the proof of item (i).

Concerning item (ii) it is well known ([13]) that the convergence rate of a consensus algorithm ruled by a primitive matrix P, is exponential and is upper bounded by the essential spectral radius $\rho_{ess}(P)$. ■

Remark III.2 The s-CL algorithm is similar to the algorithm proposed in [8]. However in [8], the measurement graph is assumed to be undirected, namely, both measurements z_{ij} and z_{ji} are available to node i and j under the additional assumption that $z_{ij} = -z_{ji}$.

Remark III.3 The authors in [16] solved the problem formulated in (3) proposing a synchronous algorithm that implements the Jacobi iterative method. The performance of this algorithm, in terms of rate of convergence to the optimal solution, is similar, for many families of measurement graphs, to the performance of the synchronous consensus-based algorithm introduced in this section.

IV. An asynchronous implementation of distributed consensus based solution

The distributed algorithm illustrated in the previous section, has an important limitation: it is applicable only to sensor networks with synchronized and reliable communication. Indeed, the s-CL algorithm requires that there exists a predetermined common communication schedule for all nodes and, at each communication round, each node must simultaneously and reliably communicate its information. The aim of this section is to reduce the communication requirements of the s-CL algorithm, in particular in terms of synchronization. To do so, we next introduce the *asynchronous Consensus-based Localization* algorithm (denoted as a-CL hereafter). This algorithm is based on an asymmetric broadcast communication protocol. Differently from the s-CL, at each iteration of the a-CL there is only one node transmitting information to all its neighbors. Since the actual value of neighboring estimates are not available at each iteration, we assume that each node stores in its local memory a copy of the neighbors' variables recorded from the last communication received. For $j \in \mathcal{N}_i$, we denote by $\hat{x}_j^{(i)}(k)$ the estimate of x_j kept in i's local memory at the end of the k-th iteration. If node j performed its last transmission to node i during h-th iteration, $h \leq k$, then $\hat{x}_j^{(i)}(k) = \hat{x}_j(h)$.

The a-CL algorithm is formally described as follows.

Processor states: For $i \in \{1,\ldots,N\}$, node i stores in memory the measurements z_{ij}, z_{ji} and the covariances R_{ij}, R_{ji} for all $j \in \mathcal{N}_i$. Moreover node i stores in memory also the estimate $\hat{x}_i$ of x_i and, for $j \in \mathcal{N}_i$ an estimate $\hat{x}_j^{(i)}$ of $\hat{x}_j$.

Initialization: Every node i initializes its estimate $\hat{x}_i$ and the variables $\hat{x}_j^{(i)}$, $j \in \mathcal{N}_i$, to arbitrary values.

Transmission iteration: For $k \in \mathbb{N}$, at the start of the $(k+1)$-th iteration of the algorithm, there is only one node, say i, which transmits information to its neighbors; precisely, node i sends the value of its estimate $\hat{x}_i(k)$ to node j, $j \in \mathcal{N}_i$.

Update iteration: For $j \in \mathcal{N}_i$, node j performs the following actions in order

(i) it sets $\hat{x}_i^{(j)}(k+1) = \hat{x}_i(k)$, while for $s \in \mathcal{N}_j \setminus \{i\}$, $\hat{x}_s^{(j)}$ is left unchanged, i.e., $\hat{x}_s^{(j)}(k+1) = \hat{x}_s^{(j)}(k)$;

(ii) it updates $\hat{x}_j$ as

$$\hat{x}_j(k+1) := p_{jj}\hat{x}_j(k) + \sum_{h\in\mathcal{N}_j} p_{jh}\hat{x}_h^{(j)}(k+1) + b_j. \quad (4)$$

Clearly for $s \notin \mathcal{N}_i$, $\hat{x}_s$ is left unchanged during the $(k+1)$-th iteration of the algorithm, i.e, $\hat{x}_s(k+1) = \hat{x}_s(k)$.

Remark IV.1 Observe that the above algorithm has been described assuming that the communication channels are reliable, i.e, no packet losses occur, and that the communication delays are negligible, i.e., when node i perform a transmission, the estimate $\hat{x}_i$ is instantaneously used by its neighbors. We will come back on these non-idealities on Section VI.

Next, we rewrite the updating step of the a-CL in a more compact way. Observe preliminarily that, under the assumption of reliable communications over the network, the broadcast protocol lets only two information about the estimate of x_i, $i \in V$, to flow through the network: specifically, $\hat{x}_i(k)$, that is the actual value of the estimate $\hat{x}_i$ at iteration k, and $\hat{x}_i(t_i'(k))$, being $t_i'(k)$ the iteration during which node i has performed its last transmission up to iteration k of a-CL (that is, $\hat{x}_i(t_i'(k))$ is the value of $\hat{x}_i$ at its last communication round). Notice that, for $j \in \mathcal{N}_i$, $\hat{x}_i^{(j)}(t'') = \hat{x}_i(t_i'(k))$ for all t'' such that $t_i'(k) < t'' \leq k$.

Now let us define $x_i'(k) = \hat{x}_i(k)$ and $x_i''(k) = \hat{x}_i(t_i'(k))$ and, accordingly, let $\mathbf{x}'(k) = [x_1'(k), \ldots, x_N'(k)]^T$ and $\mathbf{x}''(k) = [x_1''(k), \ldots, x_N''(k)]^T$. Moreover let $Q_i \in \mathbb{R}^{2N\times 2N}$ be defined as

$$Q_i = \begin{bmatrix} Q_{11}^{(i)} & Q_{12}^{(i)} \\ Q_{21}^{(i)} & Q_{22}^{(i)} \end{bmatrix} \quad (5)$$

where

$$\begin{aligned} Q_{11}^{(i)} &= \sum_{h\notin\mathcal{N}_i} e_h e_h^T + \sum_{j\in\mathcal{N}_i} \left(p_{jj} e_j e_j^T + p_{ji} e_j e_i^T\right) \\ Q_{12}^{(i)} &= \sum_{j\in\mathcal{N}_i} e_j \left(\sum_{h\in\mathcal{N}_j/i} p_{jh} e_h^T\right) \\ Q_{21}^{(i)} &= e_i e_i^T \\ Q_{22}^{(i)} &= I - e_i e_i^T \end{aligned}$$

being e_ℓ, $\ell \in \{1,\ldots,N\}$, the N-dimensional vector having all the components equal to zero except the ℓ-th component which is equal to 1. Observe that, for $i \in \{1,\ldots,N\}$, Q_i is a $2N$-dimensional stochastic matrix. Finally let

$$B_i = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} e_j^T b \\ 0_N \end{bmatrix}$$

Assume, w.l.o.g., that node i is the node performing the transmission during the $(k+1)$-th iteration of the a-CL. Hence the updating step of a-CL can be written in vector form as

$$\begin{bmatrix} \mathbf{x}'(k+1) \\ \mathbf{x}''(k+1) \end{bmatrix} = Q_i \begin{bmatrix} \mathbf{x}'(k) \\ \mathbf{x}''(k) \end{bmatrix} + B_i. \quad (6)$$

Now let us introduce the auxiliary variable

$$\xi(k) = \begin{bmatrix} \mathbf{x}'(k) \\ \mathbf{x}''(k) \end{bmatrix} - \begin{bmatrix} \mathbf{x}^*_{\text{opt}} \\ \mathbf{x}^*_{\text{opt}} \end{bmatrix}.$$

By exploiting the fact that, for $i \in \{1,\ldots,N\}$,

$$\begin{bmatrix} \mathbf{x}^*_{\text{opt}} \\ \mathbf{x}^*_{\text{opt}} \end{bmatrix} = Q_i \begin{bmatrix} \mathbf{x}^*_{\text{opt}} \\ \mathbf{x}^*_{\text{opt}} \end{bmatrix} + B_i \quad (7)$$

we have that the variable ξ satisfies the following $2N$-dimensional recursive equation

$$\xi(k+1) = Q_i \xi(k). \quad (8)$$

Observe that $\hat{\mathbf{x}}(k) \to \mathbf{x}^*_{\text{opt}} + \alpha\mathbf{1}$ if and only if $\xi(k) \to \alpha\mathbf{1}$. Moreover, since Q_i is a stochastic matrix for any $i \in \{1,\ldots,N\}$, we have that (8) represents a $2N$-dimensional time-varying consensus algorithm.

In next sections, we analyze the convergence properties and the robustness to delays and packet losses of the a-CL algorithm by studying system (8) resorting to the mathematical tools developed in the literature of the consensus algorithms. In particular we will provide our results considering two different scenarios which are formally described in the following definitions.

Definition IV.2 (Randomly persistent communicating network) *A network of N nodes is said to be a* randomly persistent communicating network *if there exists a N-upla $(\beta_1,\ldots,\beta_N)$ such that $\beta_i > 0$, for all $i \in \{1,\ldots,N\}$, and $\sum_{i=1}^N \beta_i = 1$, and such that, for all $k \in \mathbb{N}$,*

$$\mathbb{P}\left[\textit{the transmitting node at iteration } k \textit{ is node } i\right] = \beta_i.$$

Definition IV.3 (Uniformly persistent communicating network) *A network of N nodes is said to be a* uniformly persistent communicating network *if there exists a positive integer number τ such that, for all $k \in \mathbb{N}$, each node transmits the value of its estimate to its neighbors at least once within the time interval $[k, k+\tau)$.*

V. Performance analysis of a-CL algorithm under randomly persistent communications

The following result characterizes the convergence properties of the a-CL when the network is randomly persistent communicating.

Proposition V.1 *Consider a* randomly persistent communicating network *of N nodes running the a-CL algorithm over a weakly connected measurement graph $\mathcal{G}_m$. Let ϵ be such that $0 < \epsilon < 1/(2d_{max}R_{min}^{-1})$. Moreover let $\hat{x}_i$, $i \in \{1,\ldots,N\}$, $\hat{x}_j^{(i)}$, $j \in \mathcal{N}_i$, be initialized to any real number. Then the following facts hold true*

(i) *the evolution $k \to \hat{\mathbf{x}}(k)$ converges almost surely to an optimal solution $\mathbf{x}_{opt} \in \chi$, i.e., there exists $\alpha \in \mathbb{R}$ such that*

$$\mathbb{P}\left[\lim_{k\to\infty} \hat{\mathbf{x}}(k) = \mathbf{x}^*_{opt} + \alpha\mathbf{1}\right] = 1.$$

(ii) *the evolution $k \to \hat{\mathbf{x}}(k)$ is exponentially convergent in mean-square sense, i.e., there exist $C > 0$ and $0 \leq \rho < 1$ such that*

$$\begin{aligned} \lim_{k\to\infty} \mathbb{E}\left[\|\hat{\mathbf{x}}(k) - (\mathbf{x}^*_{opt} + \alpha\mathbf{1})\|^2\right] \\ \leq C\rho^k \mathbb{E}\left[\|\hat{\mathbf{x}}(0) - (\mathbf{x}^*_{opt} + \alpha\mathbf{1})\|^2\right]. \end{aligned}$$

Proof: The proof of Proposition V.1 is based on proving the convergence to consensus of (8) using the mathematical tools developed in [17]. Let σ be the random process such that $\sigma(k)$ denotes the node performing the transmission action at the beginning of the $k+1$-th iteration. Clearly, in the randomized scenario we are considering, we have that, for $i \in \{1,\dots,N\}$, $\mathbb{P}[\sigma(k)=i]=\beta_i$ for all k. Let

$$S(k)=\prod_{h=0}^{k} Q_{\sigma(h)}.$$

Observe that $S(k)$ inherits the same block structure of the matrices $\{Q_i\}_{i=1}^N$, namely we can write

$$S(k)=\left[\begin{array}{cc} S_{11}(k) & S_{12}(k) \\ S_{21}(k) & S_{22}(k) \end{array}\right]$$

As consequence of Theorem 3.1 in [17] the a-CL reaches almost surely consensus if and only if, for every i and j in V

$$\mathbb{P}\left[\mathcal{E}_{ij}\right]=1, \tag{9}$$

where

$$\mathcal{E}_{ij}=\left\{\exists \ell, \exists k \,|\, S_{i\ell}(k)S_{j\ell}(k)>0\right\}.$$

Now observe that, since the measurement graph is weakly connected, then the communication graph is a connected undirected graph. This fact together with the fact the diagonal elements of $Q_{11}^{(i)}$ are all positive for any $i \in \{1,\dots,N\}$ implies that there exists almost surely $\bar{k}$ such that, for all $k' \geq \bar{k}$, all the elements of the matrix $S_{11}(k')$ are strictly greater than 0. Assume now, without loss of generality, that $\sigma(k')=i$, for $k' \geq k$. Then, since the i-th row of $S_{21}(k'+1)$ is equal to $e_i e_i^T S_{11}(k')$, it turns out that, all the elements of the i-th row of $S_{21}(k'+1)$ are strictly greater than 0. Moreover, it is easy to see that they will remain strictly greater than 0 also for any $k'' \geq k'$. Hence we can argue that, there exists almost surely, also a $\bar{k}'$ such that for all $k' \geq \bar{k}'$, all the elements of the matrix $S_{21}(k')$ are strictly greater than 0. It follows that the property stated in (9) is satisfied for any $k \geq \bar{k}'$ and for any $\ell \in \{1,\dots,N\}$. This concludes the proof of item (i).

Concerning item (ii), we again resort to the results in [17]. Let $\Omega = I - 1/2N\mathbf{1}\mathbf{1}^T$ where in this expression we assume that I is the $2N$-dimensional identity matrix and the vector $\mathbf{1}$ is $2N$-dimensional. From the results in [17], it follows that to study the rate of convergence of $\mathbb{E}\left[\|\xi(k)-\alpha\mathbf{1}\|^2\right]$ is equivalent to study the convergence rate of $\mathbb{E}\|\Omega\xi(k)\|^2$ and in particular of the linear recursive system

$$\Delta(t+1)=\mathbb{E}\left[Q_{\sigma(0)}^T \Delta(t) Q_{\sigma(0)}\right]$$

where $\Delta(0)=\Omega$. Observe that $\Delta(t)$ is the evolution of a linear dynamical system which can be written in the form

$$\Delta(t+1)=\mathcal{L}(\Delta(t))$$

where $\mathcal{L}:\mathbb{R}^{2N\times 2N}\to\mathbb{R}^{2N\times 2N}$ is given by

$$\mathcal{L}(M)=\mathbb{E}\left[Q_{\sigma(0)}^T M Q_{\sigma(0)}\right].$$

As highlighted in [17], the linear operator $\mathcal{L}$ can be represented by the matrix $\mathbf{L}=\mathbb{E}[\mathbf{Q}_{\sigma(\mathbf{0})}\otimes\mathbf{Q}_{\sigma(\mathbf{0})}]^{\mathbf{T}}$ where $\otimes$ denotes the Kronecker product of matrices. Following the proof of Proposition 4.3 of [17], one can see that $\mathbf{L^T}$ is a primitive stochastic matrix which, therefore, has the eigenvalue 1 with algebraic multiplicity 1. Moreover, $\mathbf{L^T}(\mathbf{1}\otimes\mathbf{1})=(\mathbf{1}\otimes\mathbf{1})$ and $(\mathbf{1}\otimes\mathbf{1})(\Omega\otimes\Omega)=0$, from which it follows that $\mathbb{E}\|\Omega\xi(k)\|^2 \leq C\rho_{ess}(\mathbf{L^T})\mathbb{E}\|\mathbf{\Omega}\xi(\mathbf{0})\|^{\mathbf{2}}$ where $\rho_{ess}(\mathbf{L^T})$ denotes the essential spectral radius of $\mathbf{L^T}$. ■

A. Bounds on the convergence rate of the a-CL algorithm

In this section we provide some insights on the convergence rate of the a-CL algorithm in the randomly persistent communicating scenario. To do so, we consider the algorithm (8) whose performance in terms of rate of convergence to the consensus can be analyzed following again the treatment in [17]. Typically, one would like to study the convergence rate of a randomized consensus algorithm by providing a mean-square analysis of the behavior of the distance between the state and the asymptotic consensus point, namely, by analyzing the speed convergence of the quantity $\mathbb{E}\left[\|\xi-\alpha\mathbf{1}\|^2\right]$. Unfortunately, this is not a trivial task in general.

To overcome this difficulty we introduce the auxiliary variable $\Omega\,\xi$ where $\Omega = I-\frac{1}{2N}\mathbf{1}\mathbf{1}^T$ [2]. The first consequence of the results obtained in [17] is that the quantities $\mathbb{E}\left[\|\xi-\alpha\mathbf{1}\|^2\right]$ and $\mathbb{E}\left[\|\Omega\xi\|^2\right]$ have the same exponential convergence rate to zero, or, more formally, given any initial condition $\xi(0)$,

$$\limsup_{k\to\infty}\mathbb{E}\left[\|\xi(k)-\alpha\mathbf{1}\|^2\right]^{1/k}=\limsup_{k\to\infty}\mathbb{E}\left[\|\Omega\xi(k)\|^2\right]^{1/k}.$$

For this reason, in what follows we study the right-hand expression, which turns out to be simpler to analyze. In order to have a single figure not dependent on the initial condition, we focus on this worst case exponential rate of convergence

$$R=\sup_{\xi(0)}\limsup_{k\to\infty}\mathbb{E}\left[\|\Omega\xi(k)\|^2\right]^{1/k}$$

It has been proved in Proposition 4.4 of [17] that

$$\left[\rho_{\text{ess}}(\bar{Q})\right]^2 \leq R \leq sr(\mathbb{E}(Q_i^T\Omega Q_i)). \tag{10}$$

where $\bar{Q}$ is the average consensus matrix, namely, $\bar{Q}=\mathbb{E}[Q_i]=\sum_{i=1}^N \beta_i Q_i$, and where $sr(\mathbb{E}(Q_i^T\Omega Q_i))$ denotes the spectral radius of the semidefinite positive matrix $\mathbb{E}(Q_i^T\Omega Q_i)$, i.e., its largest eigenvalue. Unfortunately, it turns out from a numerical inspection over significant families of graphs, like Cayley graphs (see [13]), random geometric graphs, that the upper bound $sr(\mathbb{E}(Q_i^T\Omega Q_i))$ is greater than 1, that is, it is not informative for our analysis. However we have run a number of MonteCarlo simulations randomized over graphs of different topology and size and over different initial conditions, and it always resulted that $\limsup_{k\to\infty}\mathbb{E}\left[\|\Omega\xi(k)\|^2\right]^{1/k} \leq \rho_{\text{ess}}(\bar{Q})$. Based on this experimental evidence we formulate the following conjecture.

Conjecture V.2 *The quantity $\rho_{ess}(\bar{Q})$ is an upper bound for the exponential convergence rate R, i.e.,*

$$R \leq \rho_{ess}(\bar{Q})$$

[2] In this case the identity matrix I is $(2N\times 2N)$-dimensional and the vector $\mathbf{1}$ is $2N$-dimensional.

The above conjecture and the fact that $\left[\rho_{\text{ess}}(\bar{Q})\right]^2 \leq R$ motivates to study $\rho_{ess}(\bar{Q})$.

Remark V.3 Notice that equation (8) describes a higher order consensus algorithm, for which little analytic tools are available. In fact, the few works available in the literature which address the rate of convergence of randomized higher order consensus algorithms are limited to the convergence in expectation [12].

B. Rate Analysis of a-CL algorithm for regular graphs

In this section we assume that the measurements graph $\mathcal{G}_m = (V, \mathcal{E}_m)$ is a strongly connected bidirected regular graph such that, for $i \in \{1, \dots, N\}$, $|\mathcal{N}_i| = \nu$. Moreover we assume the following properties.

Assumption V.4 *We have that*

(i) *the error measurements covariances are all identical, i.e., $R_{ij} = R$ for all $(i,j) \in \mathcal{E}_m$;*
(ii) $\epsilon = R/(2(\nu+1))$*;*
(iii) *the probabilities $\{\beta_1, \dots, \beta_N\}$ are uniform, i.e., $\beta_1 = \dots = \beta_N = 1/N$.*

Observe that, from properties (i) and (ii) of Assumption V.4, it turns out that the matrix $P = I - \epsilon A^T R^{-1} A$, associated to the s-CL algorithm, is symmetric and such that $P_{ij} = 1/(\nu+1)$ for $j \in \mathcal{N}_i \cup \{i\}$. Let $\lambda_1(P) = 1 > \lambda_2(P) \geq \dots \geq \lambda_N(P)$ be the eigenvalues of P. Then $\rho_{\text{ess}}(P) = \max\{|\lambda_2(P)|, |\lambda_N(P)|\}$. The following Lemma illustrates how the $2N$ eigenvalues of $\bar{Q}$ are related to those of P.

Lemma V.5 *Consider the a-CL algorithm running over a bidirected regular graph $\mathcal{G}_m = (V, \mathcal{E}_m)$ such that, for $i \in \{1, \dots, N\}$, $|\mathcal{N}_i| = \nu$. Assume Assumption V.4 holds true. Then the $2N$ eigenvalues of $\bar{Q}$ are the solutions of the following N second-order equations*

$$f(s, \lambda_i, N, \nu) = s^2 + (a+b)s + (ab+c) \tag{11}$$

where

$$\begin{aligned} a &= -\left[\frac{N-\nu}{N} + \frac{\lambda_i}{N} + \frac{\nu-1}{N(\nu+1)}\right] \\ b &= -\frac{N-1}{N} \\ c &= -\frac{\nu-1}{N^2}(\lambda_i - \frac{1}{\nu+1}) \end{aligned}$$

Now let $s_1^{(i)}$ and $s_2^{(i)}$ denote the two solutions of $f(s, \lambda_i, N, \nu)$. It easy to see that $s_1^{(1)} = 1$ and $s_2^{(1)} = 1 - \frac{\nu^2+1}{N(\nu+1)}$. The following result restricts the search of $\rho_{\text{ess}}(\bar{Q})$ among the values $|s_1^{(2)}|$, $|s_2^{(2)}|$ and $1 - \frac{\nu^2+1}{N(\nu+1)}$.

Theorem V.6 *Consider the a-CL algorithm running on an bidirected regular graph $\mathcal{G}_m = (V, \mathcal{E}_m)$ such that, for $i \in \{1, \dots, N\}$, $|\mathcal{N}_i| = \nu$. Assume Assumption V.4 holds true. Moreover let*
$\gamma^* = \frac{\nu - 1 + N(\nu+1) - \sqrt{N^2(\nu+1)^2 - 2N(\nu^3+\nu+2) + (\nu-1)^2 + (\nu^2+1)^2}}{\nu+1}$
then

(i) *if $1 - \rho_{ess}(P) \leq \gamma^* \Longrightarrow \rho_{ess}(\bar{Q}) = \max(|s_1^{(2)}|, |s_2^{(2)}|)$;*
(ii) *if $1 - \rho_{ess}(P) > \gamma^* \Longrightarrow \rho_{ess}(\bar{Q}) = s_2^{(1)} = 1 - \frac{\nu^2+1}{N(\nu+1)}$.*

The proofs of Lemma V.5 and Theorem V.6 follows from standard algebraic manipulations. Due to space constraints we do not include them here, but we refer the interested reader to the document [18].

We provide now an asymptotic result on $\rho_{\text{ess}}(\bar{Q})$. To do so, consider a sequence of connected undirected regular graphs $\mathcal{G}_N$ of increasing size N, and fixed degree ν. Assume Assumption V.4 holds true for any $\mathcal{G}_N$. Then to any $\mathcal{G}_N$ we can associate a stochastic matrix P_N such that $(P_N)_{ij} = 1/(\nu+1)$ for all $j \in \mathcal{N}_i \cup \{i\}$. Let us assume the following property.

Assumption V.7 *Consider the sequence of matrices P_N associated to the sequence of graphs $\mathcal{G}_N$ above described, and assume that*

$$\rho_{ess}\left(P_N\right) = 1 - \varepsilon(N) + o(\varepsilon(N)), \tag{12}$$

where $\varepsilon : \mathbb{N} \to \mathbb{R}$ is a positive function such that $\varepsilon(N) \to 0$ as $N \to \infty$.

Important families of matrices satisfying the above assumption are given by the matrices built over the N-dimensional toruses and the Cayley graphs (see [13]).

Now, let the matrix $\bar{Q}_N$ represent the average matrix associated to the a-CL algorithm running over $\mathcal{G}_N$. The following result characterizes the asymptotic behavior of $\rho_{\text{ess}}\left(\bar{Q}_N\right)$, with respect to $\rho_{\text{ess}}\left(P_N\right)$.

Proposition V.8 *Consider the sequence of graphs $\mathcal{G}_N$ described above. Consider the a-CL algorithm running over $\mathcal{G}_N$. Assume Assumption V.4 and Assumption V.7 hold true. Then*

$$\rho_{ess}\left(\bar{Q}_N\right) = 1 - \frac{\nu(\nu+1)}{N(\nu^2+1)}\varepsilon(N) + o\left(\frac{\varepsilon(N)}{N}\right)$$

Proof: Let $\gamma_i = 1 - \lambda_i$, then we can rewrite Eqn. (11) as:

$$f(s, \lambda_i, N, \nu) = d(s, N, \nu) + \gamma_i n(s, N, \nu) \triangleq g(s, \gamma_i, N, \nu)$$

so that g is an explicit function of γ_i, where

$$\begin{aligned} d(s, N, \nu) &= s^2 - \frac{2N(\nu+1) - (\nu^2+1)}{N(\nu+1)}s + 1 - \frac{\nu^2+1}{N(\nu+1)} \\ n(s, N, \nu) &= \frac{s}{N} + \frac{\nu - N}{N^2} \end{aligned}$$

Note that

$$\lim_{N\to\infty} \gamma^*(\nu, N) = \frac{\nu^2+1}{(\nu+1)^2}$$

therefore, according to Theorem V.6 and assumption V.7, for N sufficiently large, $\rho_{ess}(\bar{Q}_N)$ is given by (i). Since $\gamma_2 = 1 - \lambda_2 + \epsilon(N) + o(\epsilon(N))$, then $s_1^{(2)} = \bar{s}_1^{(2)} + \alpha\varepsilon(N) + o(\varepsilon(N))$ and $s_2^{(2)} = \bar{s}_2^{(2)} + \beta\varepsilon(N) + o(\varepsilon(N))$ for some scalar α, β, where $\lambda_2 = \rho_{ess}(P_N)$ and $\bar{s}_1^{(2)}, \bar{s}_2^{(2)}$ are the solutions of second order equation $g(s, 0, N, \nu) = 0$. It is easy to verify that $\bar{s}_1^{(2)} = 1$ and $\bar{s}_2^{(2)} = 1 - \frac{\nu^2+1}{N(\nu+1)}$. Since $|\bar{s}_1^{(2)}| > |\bar{s}_2^{(2)}|$, then for N sufficiently large and by continuity we have $\rho_{ess}(\bar{Q}_N) = |s_1^{(2)}|$. We are therefore interested in explicitly computing the scalar α. Since

$$\begin{cases} g(1, 0, N, \nu) = 0 \\ \frac{\partial g}{\partial s}\Big|_{(1,0,N,\nu)} \neq 0 \end{cases}$$

it is possible to exploit the implicit function theorem that allows us to write:

$$\begin{aligned} s_1^{(2)} &= 1 - \frac{\partial g}{\partial \gamma_i}\left(\frac{\partial g}{\partial s}\right)^{-1}\Bigg|_{(1,0,N,\nu)} (\epsilon(N) + o(\epsilon(N))) \\ &= 1 - \frac{\nu(\nu+1)}{N(\nu^2+1)}\epsilon(N) + o\left(\frac{\epsilon(N)}{N}\right) \end{aligned}$$

which means that $\rho_{ess}(\bar{Q})$ can be expressed as

$$\rho_{\text{ess}}(\bar{Q}) = 1 - \frac{\nu(\nu+1)}{N(\nu^2+1)}\epsilon(N) + o\left(\frac{\epsilon(N)}{N}\right) \qquad (13)$$

■

Thank to [17], we know that the rate of convergence is lower bounded by

$$\rho_{\text{ess}}(\bar{Q}_N)^2 = 1 - 2\frac{\nu(\nu+1)}{N(\nu^2+1)}\varepsilon(N) + o\left(\frac{\varepsilon(N)}{N}\right)$$

while we recall we conjecture that $R \leq \rho_{\text{ess}}(\bar{Q}_N)$. Observe that, for $N \gg 1$,

$$\frac{1-\rho_{\text{ess}}(\bar{Q}_N)}{1-\rho_{\text{ess}}(P_N)} \simeq \frac{\nu(\nu+1)}{N(\nu^2+1)}$$

and so, we can conclude that the a-CL algorithm slows down of a factor $1/N$ with respect to the synchronous implementation which is the same of standard memoryless asynchronous consensus algorithms. However the fact that the rate of convergence in expectation is reduced by a factor N, is not surprising because in the a-CL there is only one node transmitting information at each iteration.

Remark V.9 Following Remark III.3, it is worth stressing that also the Jacobi-like strategy introduced in [2] is amenable of asynchronous implementation, see [7]. However, to the best of our knowledge, no theoretical analysis of the rate of convergence of the asynchronous version, introduced in [7], has been proposed in the literature.

VI. Robustness properties of the a-CL algorithm with respect to packet losses and delays

In section IV we have introduced the a-CL algorithm assuming that the communication channels are reliable, i.e., no packet losses occur, and that the transmission delays are negligible. In this section we relax these assumptions and we show that the a-CL algorithm still converges provided that the network is uniformly persistent communicating and the transmission delays and the frequencies of communication failures satisfy mild conditions which we formally describe next.

Assumption VI.1 (Bounded packet losses) *There exists a positive integer L such that the number of consecutive communication failures between every pair of neighboring nodes in the communication graph $\mathcal{G}_c$ is less than L.*

Assumption VI.2 (Bounded delay) *Assume node i broadcast its estimate to its neighbors at the beginning of iteration k, and, assume that, the communication link (i,j) does not fail. Then, there exists a positive integer D such that the information $\hat{x}_i(k)$ is used by node j to perform its local update not later than iteration $k+D$.*

Loosely speaking Assumption VI.1 implies that there can be no more than L consecutive packet losses between any pair of nodes i, j belonging to the communication graph. Differently, Assumption VI.2 consider the scenario where the received packets are not used instantaneously, but are subject to some delay no greater than D iterations.

Clearly, in this more realistic scenario, it turns out that the implementation of the a-CL is slightly different from the description provided in Section IV. Specifically, consider the k-th iteration of the a-CL algorithm and, without loss of generality, assume node i is the transmitting node during this iteration. Due to the presence of packet losses and delays, it might happen that the set of updating nodes is, in general, different from the set $\mathcal{N}_i$. In fact, for $j \in \mathcal{N}_i$, node j does not perform any update since the packet $\hat{x}_i(k)$ from node i is lost or simply because the update is delayed. Moreover there might be a node $h \notin \mathcal{N}_i$ which, during iteration k, decides to perform an update since it received a packet $\hat{x}_s$, $s \in \mathcal{N}_h$, within the last B iterations. This scenario can be formally represented by the set of nodes $V'(k) \subseteq V$ which decide to perform an update at iteration k. Then, Eqn. (4) can be rewritten as

$$\hat{x}_j(k+1) := p_{jj}\hat{x}_j(k) + \sum_{h \in \mathcal{N}_j} p_{jh}\hat{x}_h(k'_h) + b_j, \qquad (14)$$

for all $j \in V'(k)$, where $k - (\tau L + D) \leq k'_h \leq k$, i.e. loosely speaking when an update is performed, the local estimate of the neighbouring nodes cannot be older than $\tau L + D$ iterations[3]. Indeed, if $L = D = 0$, then we recover the standard a-CL algorithm where $V'(k) = \mathcal{N}_i$.

The following result characterizes the convergence properties of the a-CL in presence of delays, packet losses and when the network is uniformly persistent communicating.

Proposition VI.3 *Consider a* uniformly persistent communicating *of N nodes running the a-CL algorithm over a weakly connected measurement graph $\mathcal{G}_m$. Let Assumptions VI.1 and VI.2 be satisfied. Let ϵ be such that $0 < \epsilon < 1/(2d_{max}R_{min}^{-1})$. Moreover let $\hat{x}_i$, $i \in \{1,\dots,N\}$, $\hat{x}_j^{(i)}$, $j \in \mathcal{N}_i$, be initialized to any real number. Then the following facts hold true*

(i) *the evolution $k \to \hat{\mathbf{x}}(k)$ asymptotically converges to an optimal estimate $\mathbf{x}_{opt} \in \chi$, i.e., there exists $\alpha \in \mathbb{R}$ such that*

$$\lim_{k\to\infty} \hat{\mathbf{x}}(k) = \mathbf{x}^*_{opt} + \alpha\mathbf{1};$$

(ii) *the convergence is exponential, namely, there exists $C > 0$ and $0 \leq \rho < 1$ such that*

$$\|\hat{\mathbf{x}}(k) - \left(\mathbf{x}^*_{opt} + \alpha\mathbf{1}\right)\| \leq C\rho^k \|\hat{\mathbf{x}}(0) - \left(\mathbf{x}^*_{opt} + \alpha\mathbf{1}\right)\|.$$

Proof: The proof follows from the statement of Proposition 1 in [19]. In [19], the authors consider the following

[3]Recall we are assuming the network is uniformly persistent communicating, namely, for all $k \in \mathbb{N}$, each node performs at least one transmission within the time interval $[k, k+\tau)$.

consensus algorithm with delays[4]

$$x^i(k+1) = \sum_{j=1}^{m} a_j^i(k) x^j(k - t_j^i(k)) \tag{15}$$

where x^i denotes the state of node i, $i \in \{1, \dots, M\}$, the scalar $t_j^i(k)$ is nonnegative and it represents the delay of a message from agent j to agent i, while the scalar $a_j^i(k)$ is a nonnegative weight that agent i assigns to a delayed estimate $x^j(s)$ arriving from agent j at time k. It is assumed that the weights $a_j^i(k)$ satisfy the following assumption

Assumption i. There exists a scalar η, $0 < \eta < 1$ such that

(i) $a_i^i(k) \geq \eta$ for all $k \geq 0$;
(ii) $a_j^i(k) \geq \eta$ for all $k \geq 0$, and all agents j whose (potentially delayed) information $x^j(s)$ reaches agent i during the k-th iteration;
(iii) $a_j^i(k) = 0$ for all $k \geq 0$ and j otherwise.
(iv) $\sum_{j=1}^{m} a_j^i(k) = 1$ for all i and k.

For any k let the information exchange among the agents may be represented by a directed graph (V, E_k), where $V = \{1, \dots, m\}$ with the set E_k of directed edges given by $E_k = \{(j,i) | a_j^i(k) > 0\}$. The authors impose a connectivity assumption on the agent system, which is stated as follows.

Assumption ii. The graph (V, E_∞) is connected, where E_∞ is the set of edges (j,i) representing agent pairs communicating directly infinitely many times, i.e., $E_\infty = \{(j,i) | (j,i) \in E_k \text{ for infinitely many indices } k\}$.

Additionally it is assumed that the intercommunication intervals are bounded for those agents that communicate directly. Specifically,

Assumption iii. There exists an integer $B \geq 1$ such that for every $(j,i) \in E_\infty$, agent j sends information to its neighbor i at least once every B consecutive iterations.

Finally, it is assumed that the delays $t_j^i(k)$ in delivering a message from an agent j to any neighboring agent i is uniformly bounded at all times. Formally

Assumption iv. Let the following hold:

(i) $t_i^i(k) = 0$ for all agents i and all $k \geq 0$.
(ii) $t_j^i(k) = 0$ for all agents j communicating with agent i directly and whose estimates x^j are not available to agent i during the k-th iteration.
(iii) There is an integer B_1 such that $0 \leq t_j^i(k) \leq B_1 - 1$ for all agents i, j, and all k.

Under Assumptions i,ii,iii,iv, it is shown, in Proposition 1 of [19], that equation (15) converge exponentially to a consensus.

Consider now the a-CL algorithm in presence of delays and packet losses. Let $\delta_j(k) = \hat{x}_j(k) - [\mathbf{x}^*_{\text{opt}}]_\mathbf{j}$ where $[\mathbf{x}^*_{\text{opt}}]_j$ denotes the j-th component of the vector $\mathbf{x}^*_{\text{opt}}$. Recalling that $\mathbf{x}^*_{\text{opt}} = P\mathbf{x}^*_{\text{opt}} + b$ and, according to (14) we have that, if $j \in V'(k)$

$$\delta_j(k+1) := p_{jj}\delta_j(k) + \sum_{h \in \mathcal{N}_j} p_{jh}\delta_h(k'_h), \tag{16}$$

otherwise

$$\delta_j(k+1) = \delta_j(k).$$

[4]We adopt the notations of paper [19].

The above equations describe a consensus algorithm on the variables $\delta_1, \dots, \delta_N$ which satisfies the above assumptions i,ii,iii,iv. Indeed assumption (i) on the weights is trivially satisfied. Assumption (ii) follows from the facts that the communication graph $\mathcal{G}_c$ is connected, the network is uniformly persistent communicating and from Assumptions VI.1 and VI.2. Assumption iii is a consequence of the fact that the network is uniformly persistent communicating and Assumption VI.1; in our setup we have $B = L\tau$. Finally Assumption iv follows from Assumption VI.2 and equation (16). Hence the variables $\delta_1, \dots, \delta_N$ converge exponentially to a consensus value α which, in turn, implies that $\hat{\mathbf{x}}$ converge exponentially to $\mathbf{x}^*_{\text{opt}} + \alpha\mathbf{1}$. ■

Remark VI.4 We believe that the analysis of the robustness to packet losses of the a-CL algorithm might be performed also in the randomized scenario considered in Section V assuming that each transmitted packet might be lost with a certain probability. We leave this analysis as object of our future research. However in the numerical section, specifically in Example VII.1, we show the effectiveness of the a-CL algorithm also in presence of random communication failures when the network is randomly persistent communicating.

Remark VI.5 Also the Jacobi-like strategy has been shown to be robust to packet losses, see [7]. Instead, concerning the other algorithms recently proposed in the literature, see [11], [10], to the best of our knowledge, no analysis considering the non-idealities introduced in this section has been proposed in the literature.

VII. Numerical Results

In this Section we provide some simulations implementing the localization consensus-based strategy introduced in this paper.

Example VII.1 In this example we consider a random geometric graph generated by choosing $N = 100$ points randomly placed in the interval $[0, 1]$. Two nodes are connected and take measurements if they are sufficiently close, i.e more specifically, both measurements z_{ij} and z_{ji} are available provided that $|x_i - x_j| \leq 0.15$. This choice resented in networks with an average number of neighbours per node of about 7. Every measurement was corrupted by Gaussian noise with covariance $\sigma^2 = 10^{-4}$. In this example we assumed that the network is randomly persistent communicating with uniform communication probabilities $(\beta_1, \dots, \beta_N)$, namely, $\beta_1 = \dots = \beta_N = 1/N$. Moreover the possibility of communication failure is taken into account. Specifically, supposing node i is transmitting, each node $j \in \mathcal{N}_i$ with a certain probability i.e., p_f, can not receive the sent packet.

In Figure 1 we plotted the behavior of the error

$$J(k) = \log\left(\|A(\hat{x}(k) - x^*)\|\right)$$

for different values of the failure probability p_f.
The plot reported is the result of the average over 1000 Monte Carlo runs, randomized with respect to both the measurement

graph[5] and the initial conditions. Observe that the trajectory of J converges to 0 exponentially.

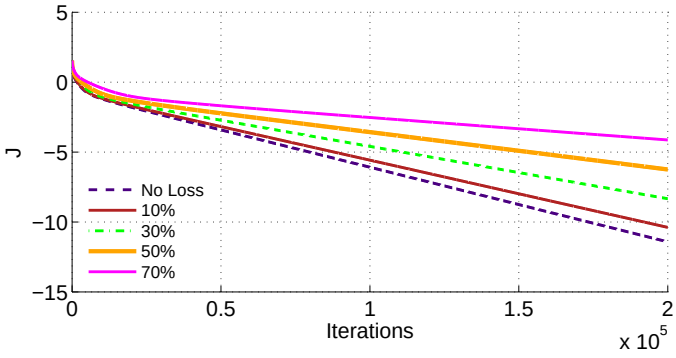

Fig. 1: Behavior of J for a randomly persistent communicating network in a random geometric graph, for different values of the probability failure p_f.

Example VII.2 In this example we tested the validity of conjecture V.2. We consider a single connected geometric graph generated as in the previous example with $N = 200$. The values of $\rho_{\text{ess}}(\bar{Q})$, $\rho_{\text{ess}}(\bar{Q})^2$ and R are reported in the table below.

$\rho_{\text{ess}}(Q)^2$	R	$\rho_{\text{ess}}(Q)$
0.999885	0.999940	0.999942

Example VII.3 In this example we provide a numerical comparison with some well known algorithms proposed in literature which, for the sake of the completeness, we briefly recall.

The first algorithm considered, hereafter called **a-GL algorithm**, is proposed in [3]. Similarly to the a-CL algorithm, during its k-th iteration one node, say h, transmits its variable $\hat{x}_h$ to all its neighbors. For $l \in \mathcal{N}_h$, node l, based on the information received from node h, performs the following update

$$\begin{aligned}\hat{x}_l(k+1) &= 1/2\,(\hat{x}_l(k) + \hat{x}_h(k) + 1/2(z_{lh} - z_{hl})) \\ &= \hat{x}_l(k) + 1/2\,(\hat{x}_l(k) - \hat{x}_h(k) + 1/2(z_{lh} - z_{hl}))\end{aligned}$$

while for $l \notin \mathcal{N}_h$ the state remains unchanged, i.e., $\hat{x}_l(k+1) = \hat{x}_l(k)$. Note that just one packet is transmitted at each iteration. Moreover, since this algorithm is known to reach mean square convergence [8], then its ergodic mean has been proposed as a possible estimator of the state.

The second algorithm, denoted hereafter as **BC algorithm**, is proposed in [11]. It requires a coordinated broadcast communication protocol meaning that, during k-th iteration one node, say h, asks the variable $\hat{x}_l$ to all its neighbors $l \in \mathcal{N}_h$. When it receives the current state values, it performs the following greedy local optimization based on the current status of the network:

$$\begin{aligned}\hat{x}_h(k+1) &:= \operatorname{argmin}_{\hat{x}_h} \textstyle\sum_{(i,j)\in\mathcal{E}}^{|\mathcal{E}|} \|\hat{x}_i(k) - \hat{x}_j(k) - z_{ij}\|^2 \\ &= \tfrac{1}{2|\mathcal{N}_h|} \textstyle\sum_{l\in\mathcal{N}_h} (2\hat{x}_l(k) - z_{lh} + z_{hl})\end{aligned}$$

Note that the number of communications performed during one iteration are $|\mathcal{N}_h| + 1$, since there is a broadcast packet sent by node h, and $|\mathcal{N}_h|$ packets sent by all its neigbours.

The last algorithm that we considered is the Randomized Extended Kaczmarz, hereafter called **REK algorithm**, presented in [10], consisting of two different update steps. The first step is an orthogonal projection of the noisy measurements onto the column space of the incidence matrix A in order to bound the measurements error. The second step is similar to the standard Kaczmarz update. Since a distributed implementation is not formally presented in [10], we propose the following. More specifically, let $s \in \mathbb{R}^M$ the current projection of the noisy measurements onto the column space of A. Similarly as above, we denote with a little abuse of notation the e-th entry of s with the corresponding edge, i.e. $s_e = s_{ij}$. Then, the REK algorithm proposed in [10] for general least-squares problems, reduces in our setting to randomly and independently selecting a node h and an edge (i,j) at each iteration k according to the following probabilities:

$$p_h = \frac{|\mathcal{N}_h| + 1}{2M}; \quad p_{ij} = \frac{1}{M}$$

and then to performing the following local updates:

$$s_{\ell h}(k+1) = s_{\ell h}(k) + \frac{\sum_{m\in\mathcal{N}_h}(s_{hm}(k) - s_{mh}(k))}{|\mathcal{N}_h|+1}, \quad \forall \ell \in \mathcal{N}_h$$

$$s_{h\ell}(k+1) = s_{h\ell}(k) - \frac{\sum_{m\in\mathcal{N}_h}(s_{hm}(k) - s_{mh}(k))}{|\mathcal{N}_h|+1}, \quad \forall \ell \in \mathcal{N}_h$$

$$\hat{x}_i(k+1) = \hat{x}_i(k) + \frac{z_{ij} - s_{ij}(k) - (\hat{x}_i(k) - \hat{x}_j(k))}{2}$$

$$\hat{x}_j(k+1) = \hat{x}_j(k) - \frac{z_{ij} - s_{ij}(k) - (\hat{x}_i(k) - \hat{x}_j(k))}{2}$$

We point out that, since in the updating step only local informations are required, the algorithm is implemented in a distributed fashion and it exactly requires $N_j + 5$ communication rounds to perform an iteration. Specifically the first $N_j + 2$ are due to the update of the variable s and the last 3 are needed to update $\hat{x}$.

Algorithm	**Sent packets per iteration**
a-CL	1
a-GL	1
BC	$N_i + 1$
REK	$N_j + 5$

Number of sent packets per iteration for each algorithm.

In this example we consider a random geometric measurement graph $\mathcal{G}$ built as in the previous example. In Figure 2 we plot the behavior of J respect to the number of iterations and sent packets. From these simulations, we observe that from an energy point of view the a-CL algorithm is the most convenient since the effective number of sent packets to achieve a certain estimation error, is lower. On the other hand if no energy

[5]In performing our average we kept only the random geometric graphs which resulted to be connected.

constraint are imposed, then REK is the fastest algorithm, although not majorly faster than BC and a-CL.

As observed in [3] the local estimates of a-GL algorithm do not converge to the optimal solution, but they oscillate around it. However, a-GL exhibits the fastest transient among all algorithms and it is also energetically efficient. In our recent work, we thus proposed to combine the a-CL algorithm with the a-GL algorithm in order to have fast transient as well as guaranteed exponential asymptotic convergence by using suitable switching strategies [20].

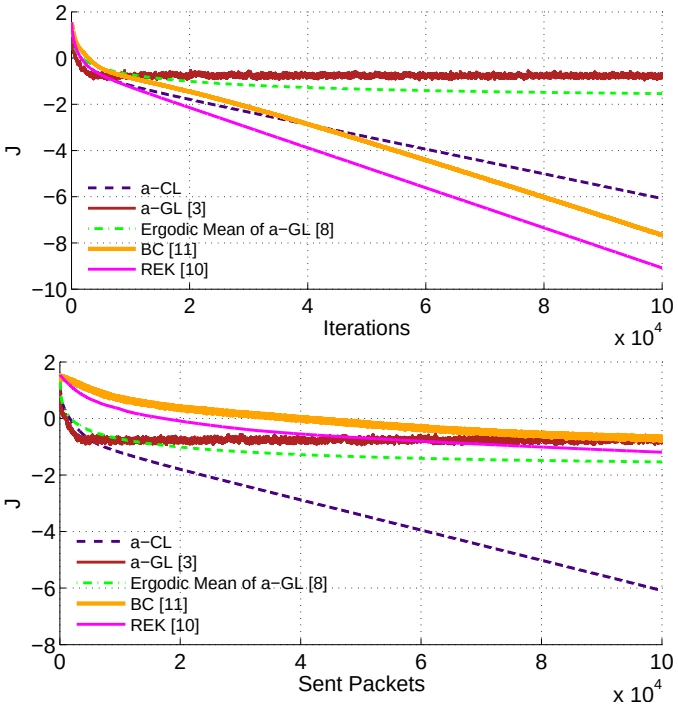

Fig. 2: Comparison of various algorithms considering the number of iteration (*top panel*) and number of sent packets (*bottom panel*).

VIII. Conclusion

In this paper we have introduce a consensus-based strategy to solve in a distributed way the problem of optimally estimating the position of each agent in a network from relative noisy distances with its neighbors. We have first introduced the algorithm in its synchronous version showing that it exponentially converges to the optimal centralized least squares solution. We have then proposed a more realistic asynchronous implementation which is still shown to be exponentially convergent under both uniform persistent communication and randomized persistent communication protocols. In the randomized scenario we have performed a theoretical analysis of the rate of convergence in mean-square, providing general lower and upper bounds; a more detailed analysis has been performed restricting to communication graphs which are regular graphs. Additionally we have shown that our novel asynchronous algorithm is robust to packet losses and delayes. Finally through numerical simulations we have tested the effectiveness of our approach as compared to other strategies recently proposed in the literature.

References

[1] P. Barooah and J. P. Hespanha, "Distributed estimation from relative measurements in sensor networks," in *Proceedings of the 2nd International Conference on Intelligent Sensing and Information Processing*, Dec. 2005.

[2] P. Barooah, "Estimation and control with relative measurements: Algorithms and scaling laws," Ph.D. dissertation, University of California, Santa Barbara, 2007.

[3] S. Bolognani, S. D. Favero, L. Schenato, and D. Varagnolo, "Consensus-based distributed sensor calibration and least-square parameter identification in WSNs," *International Journal of Robust and Nonlinear Control*, vol. 20, no. 2, pp. 176–193, 2010.

[4] R. Solis, V. Borkar, and P. R. Kumar, "A new distributed time synchronization protocol for multihop wireless networks," in *45th IEEE Conference on Decision and Control (CDC'06)*, San Diego, December 2006, pp. 2734–2739.

[5] D. Borra, E. Lovisari, R. Carli, F. Fagnani, and S. Zampieri, "Autonomous calibration algorithms for networks of cameras," in *Proceedings of the Americal Control Conference, ACC'12.*, July 2012.

[6] R. Tron and R. Vidal, "Distributed image-based 3-d localization of camera sensor networks," in *Proceedings of the 49th IEEE Conference on Decision and Control CDC'09*, 2009, pp. 901–908.

[7] P. Barooah and J. Hespanha, "Distributed estimation from relative measurements in sensor networks," in *Intelligent Sensing and Information Processing, 2005. ICISIP 2005. Third International Conference on*, 2005, pp. 226–231.

[8] W. Rossi, P. Frasca, and F. Fagnani, "Transient and limit performance of distributed relative localization," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012, pp. 2744–2748.

[9] C. Ravazzi, P. Frasca, H. Ishii, and R. Tempo, "A distributed randomized algorithm for relative localization in sensor networks," in *Proceedings of European Conference on Control (ECC'13)*, 2013.

[10] A. Zouzias and N. Freris, "Randomized extended Kaczmarz for solving least-squares," University of Toronto, Tech. Rep., 2013.

[11] A. Giridhar and P. Kumar, "Distributed clock synchronization over wireless networks: Algorithms and analysis," in *Decision and Control, 2006 45th IEEE Conference on*, 2006, pp. 4915–4920.

[12] J. Liu, B. Anderson, M. Cao, and A. S. Morse, "Analysis of accelerated gossip algorithms," in *Proceedings of the IEEE Conference on Decision and Control CDC'09*, 2009, pp. 871–876.

[13] R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri, "Communication constraints in the average consensus problem," *Automatica*, vol. 44, no. 3, pp. 671–684, 2008.

[14] R. Olfati-Saber, J. Fax, and R. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–913, 2007.

[15] F. Garin and L. Schenato, *Networked Control Systems*, ser. Lecture Notes in Control and Information Sciences. Springer, 2011, vol. 406, ch. A survey on distributed estimation and control applications using linear consensus algorithms, pp. 75–107.

[16] P. Barooah and J. Hespanha, "Estimation on graphs from relative measurements: Distributed algorithms and fundamental limith," *IEEE Control Systems Magazine*, vol. 27, no. 4, 2007.

[17] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 4, pp. 634–649, 2008.

[18] R. Carli, A. Carron, L. Schenato, and M. Todescato, "An exponential-rate consensus-based algorithms for estimation from relative measurements: implementation and performance analysis," University of Padova, Tech. Rep., 2013.

[19] A. Nedic and A. Ozdaglar, "Convergence rate for consensus with delays," *Journal of Global Optimization*, vol. 47, no. 3, pp. 437–456, 2008.

[20] A. Carron, M. Todescato, R. Carli, and L. Schenato, "Adaptive consensus-based algorithms for fast estimation from relative measurements," in *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems (NecSys'13)*, March 29 2013.