View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Agder University Research Archive

Cross-layer MAC Protocol for Unbiased Average Consensus under Random Interference

César Asensio-Marco[1,2], *Member, IEEE,* Daniel Alonso-Román[1,3], *Member, IEEE,* and Baltasar Beferull-Lozano[1], *Senior Member, IEEE*

***Abstract*—Wireless Sensor Networks have been revealed as a powerful technology to solve many different problems through sensor nodes cooperation. One important cooperative process is the so-called average gossip algorithm, which constitutes a building block to perform many inference tasks in an efficient and distributed manner. From the theoretical designs proposed in most previous work, this algorithm requires instantaneous symmetric links in order to reach average consensus. However, in a realistic scenario wireless communications are subject to interferences and other environmental factors, which results in random instantaneous topologies that are, in general, asymmetric. Consequently, the estimation of the average obtained by the gossip algorithm is a random variable, which its realizations may significantly differ from the average value. In the present work, we first derive a sufficient conditions for any MAC protocol to guarantee that the expected value of the obtained consensus random variable is the average of the initial values (unbiased estimator), while the variance of the estimator is minimum. Then, we propose a cross-layer and distributed link scheduling protocol based on carrier sense, which besides avoiding collisions, ensures both an unbiased estimation and close to minimum variance values. Extensive numerical results are presented to show the validity and efficiency of the proposed approach.**

***Index Terms*—Gossip algorithm, MAC protocol, Cross-layer design, Asymmetric communications, WSN**

I. Introduction

Gossip algorithms [1][2] are a clear example of the inherent capacity for distributed processing [3][4] that Wireless Sensor Networks (WSNs) possess. A common assumption in these algorithms is the existence of bidirectional communications so that a global common solution, such as average consensus, is reached. In a real implementation [5], ensuring bidirectional communications implies that any node j, after a transmission of data to any other node i, should wait until the corresponding acknowledgment, possibly including the data from i, is received. In addition, this latter node i should also wait for the acknowledgment from j. Then, every node waiting for an acknowledgment cannot mix information with any other node until the current data exchange is finished. Thus, while a node is waiting, the data received from other nodes, must be stored and processed after the current one. This procedure requires a control mechanism that introduces both a communication overhead and an uncontrolled delay which scales up with the number of nodes in the network, becoming prohibitive in many real applications.

Although executing gossip algorithms under asymmetric communications may still ensure consensus to a common (random) value [6][7], this may significantly differ from the average. Some conditions for an asymmetric gossip algorithm to reach consensus almost surely have been derived in [8] and an upper-bound on its resulting mean squared error (MSE) has been proposed in [9]. An interesting approach to ensure certain types of convergence is to maintain a companion variable in addition to the state variable. The companion variable allows the algorithm to correct the deviation introduced by each asymmetric exchange, as proposed in [10] and [11]. In particular, the work in [10] is based on applying a sequence of gossip algorithms to the state variable and successive companion variables until the deviation from the average is small enough. Similarly, the works of [12] and [13] exploit this additional information to ensure certain statistical properties during the consensus process so that convergence in expectation and in the mean square sense can be reached. Finally, the works of [14] and [15] use also additional variables together with a nonlinear algorithm to achieve average consensus under asymmetric communications. These last two approaches are based on computing the stationary distribution for the Markov chain characterized by the sensor network, and is thus different from consensus-type algorithms.

All these previous works [10-15] reduce the error introduced by the asymmetric exchanges of data at the expense of introducing a correction mechanism at the application layer that involves additional communications and processing. Intuitively, a similar correction is possible by properly choosing the order and the frequency of the data exchanges between neighboring nodes. In particular, the deviation from the average introduced by one asymmetric exchange can be compensated in the network by another exchange that creates a variation of similar magnitude but opposite sign. In practice, the order and the frequency of the communications depend on the MAC layer [16][17][18]. Although correcting each deviation would require global knowledge and complex control mechanisms, certain statistical properties can be introduced in a gossip process by designing the MAC protocol in a cross-layer fashion.

When designing a new MAC protocol, using accurate radio signal propagation and interference models is fundamental to ensure that collisions are minimized. The interference model is directly related to the complexity of the protocol, being important the distinction between considering only primary in-

[1]Department of Information and Communication Technology, University of Agder, Norway. [2]I2SysBio, CSIC-UVEG, Valencia, Spain. [3]ITI, Valencia, Spain. Email:{cesar.asensio,daniel.alonso,baltasar.beferull}@uia.no

This work was supported by the PETROMAKS Smart-Rig grant 244205/E30, the TOPPFORSK WISECART grant 250910/F20 and the IKTPLUSS INDURB grant 270730/O70 from the Research Council of Norway.

terference and taking also into account secondary interference. Considering only primary interference means that two links interfere with each other if and only if they share the receiver. Considering also secondary interference implies that every link can affect each other even if they do not share a receiver, which is a much more realistic model. In the first case, a scheduling following certain optimal criteria (e.g. minimum number of steps) can be found in polynomial time [19], while in the second case, it becomes NP-hard [20].

In this work, we propose a cross-layer and distributed link scheduling protocol based on a carrier sense strategy, that allows an asymmetric gossip algorithm to reach unbiased average consensus in the presence of secondary interference, while presenting close to minimum variance values. Based only on local information, such as the sensed power in the channel, nodes are able to self-organize in a time division basis, in such a way that, under the assumption of continuous time, free-collision patterns with a probabilistic symmetric structure are generated. More specifically, at every iteration of the gossip algorithm, several scheduling steps are executed, each of which implies the activation of a link and, following a novel criteria, the inhibition of several other links around it. This criteria, which is the key aspect of the proposed MAC protocol, is designed in a cross-layer basis by considering the performance of the random consensus.

Our main contributions can be summarized as follows:

- We propose a novel and realistic cross-layer scheme, where the performance of the consensus-based implementation of signal processing applications is improved by an appropriate design of the link layer. To that end, we consider all the relevant steps from the application to the link layer.

- First, we show how the computation of the average of the values of the nodes can be used as part of the distributed implementation of many signal-processing applications. In this way, we consider the result of the iterative random consensus as an estimator of the average of the values of the nodes, and show the dependence of the statistical properties of this estimator (expectation and covariance) as a function of the statistical properties of the random instantaneous connectivity patterns.

- Then, unlike other existing MAC protocols based on carrier sense, we consider as a design parameter, for each link, the area that includes all the links inhibited by its activation. We also derive the required size of this area so that the bias of the random instantaneous connectivity is zero and its variance is minimized whenever the links are activated following our approach.

- Finally, we propose a distributed implementation of this link scheduling protocol, obtaining numerical results that validate the cross-layer approach, and show that this approach outperforms other gossip algorithms over existing CSMA techniques in terms of MSE, convergence time and power consumption.

The remainder of this paper is structured as follows: some necessary background on graph theory and on consensus problems is presented in Section II. In Section III, the conditions for any MAC protocol to ensure that the consensus-based estimation of the average value is both unbiased and of minimum variance are derived. In Section IV, we present a cross-layer link scheduling protocol based on carrier sense that ensures unbiased average consensus, while presenting a small variance. We then present, in Section V, some numerical results about the performance of our approach, comparing it with existing schemes in the related literature. Finally, the conclusions are summarized in Section VI.

II. Problem definition and background

In this section, we first motivate the extensive use of consensus algorithms in many inference tasks, explaining its main drawbacks. Then, we introduce the interference model and the associated graph theory, which provide the framework that rules the design of our proposed cross-layer MAC protocol. Finally, we present the main concepts related to gossip algorithms and their convergence.

A. Motivation of this work

The distributed implementation of signal and information processing tasks in WSNs generally involves to split the global problem in simple local subtasks, each one consisting of a first step that involves local computation, followed by a second step of refinement through communication between nodes. The decomposition of certain tasks into separable functions that can be computed locally by nodes and executed in parallel, has been traditionally a popular topic in the computer science community [25]. In general, given both a network of N nodes, where each one i with access to local information given by the vector $\boldsymbol{\varsigma}_i$, and a global objective function $f_0(\boldsymbol{\varsigma_1}, \cdots, \boldsymbol{\varsigma_N})$ to be computed in a distributed fashion, the goal is to express the function f_0 as follows:

$$f_0(\boldsymbol{\varsigma_1}, \cdots, \boldsymbol{\varsigma_N}) = \sum_{i=1}^{N} f_i(\boldsymbol{\varsigma_i}) \tag{1}$$

If each node i takes $x_i(0) = N f_i(\boldsymbol{\varsigma_i})$ as its initial value, the objective function can be computed as the average of these values. Alternatively, when the objective function can be decomposed as a product of the form:

$$f_0(\boldsymbol{\varsigma_1}, \cdots, \boldsymbol{\varsigma_N}) = \prod_{i=1}^{N} \gamma_i(\boldsymbol{\varsigma_i}) \tag{2}$$

then, the average of the initial values $x_i(0) = N \log(\gamma_i(\boldsymbol{\varsigma_i}))$ results in $\log(f_0)$. It turns out that many statistical problems (Kalman filter [21], Neyman Pearson detector [22], optimization by ADMM [24], etc.) can be partially cast or decomposed as either a sum (1) or a product (2) of local functions. A common approach to implement this in a distributed fashion is the so-called randomized gossip algorithm [1][2], where the N nodes of the network aim at computing the average of their initial values by successive exchanging data with one hop neighbors in an iterative and random scheme.

TABLE I
NOTATION TABLE

Gossip parameter	Parameter description	Link Scheduling parameter	Parameter description
α	Constant link weight during consensus	β	SINR threshold
k	Consensus algorithm iteration	n	Link scheduling decision step
$\mathbf{A}(k)$	Instantaneous Adjacency matrix	η	Number of links scheduled simultaneously
$\mathbf{L}(k)$	Instantaneous Laplacian matrix	R_{max}	Maximum transmission range
$\mathbf{W}(k)$	Instantaneous Weight matrix	R_ρ	Intended transmission range
$\mathcal{N}_j(k)$	Instantaneous neighborhood of node j	Υ_{ji}	Collision area associated to link e_{ji}
$\mathbf{P}$	Matrix of connection probabilities	R_{inh}	Inhibition radius
$\mathbf{x}(0)$	Initial state of nodes	S^j_{inh}	Inhibition area associated to node j
$\mathbf{x}(k)$	State of nodes at iteration k	$\mathcal{S}_{ji}$	Set of links that inhibit link e_{ji}
x_{avg}	Average of initial values	P_t	Power transmission common to all nodes
$\hat{x}_{\text{avg}}$	Estimator of x_{avg}	$\mathcal{T}$	Set of active transmitters

Existing works in gossip algorithms for average consensus either assume that the underlying graph is undirected [1][2] or introduce some mechanism at application layer to correct the deviation produced by the asymmetric exchanges [10][11][12][13]. Implementing bidirectional communications or a corrective mechanism generally introduces certain overhead and delay in the process that may result prohibitive for certain applications.

In this work, we propose a scheme where the gossip algorithm is executed over a new cross-layer MAC protocol specifically designed to keep the random consensus value unbiased in the presence of secondary interference. Moreover, our design also leads to small variance values, keeping the overall error, measured in terms of the MSE, close to its minimum value. In order to ensure these properties, an appropriate connectivity pattern is created by our MAC protocol before each iteration of the gossip algorithm takes place. Note that our solution is compatible with the use of companion variables at the application layer.

B. Interference model

We consider a network composed of N nodes, each one equipped with an omni-directional antenna, and arbitrarily deployed in a square area of L square meters following a uniform distribution. The nodes perform unicast communications using a common transmission power P_t and each pair of nodes is linked by a single user channel. In this work, a simplified path loss model is adopted, since including fading does not significantly change the results in a real scenario, as shown in [16]. The channel gain between a transmitter node j and a receiver node i is approximated by $\frac{1}{d_{ji}^{\gamma}}$, where $\gamma \geq 2$ is the path-loss exponent and $d_{ji} = d_{ij}$ is the distance that separates nodes i and j.

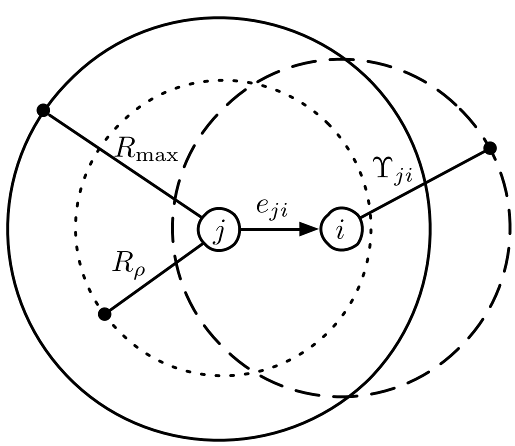


Fig. 1. Relation between the different areas and radii presented in this paper. In this example, it is assumed that the link between the transmitting node j and the receiving node i is active.

Additionally, we assume the SINR interference model according to which the successful reception of a packet sent by node j to node i depends on the SINR at node i, that is, a packet between j and i is correctly received if and only if:

$$\frac{\frac{P_t}{d_{ji}^{\gamma}}}{\sum\limits_{v \in \mathcal{T}, v \neq j} \frac{P_t}{d_{iv}^{\gamma}} + N_0} \geq \beta \tag{3}$$

where N_0 is the background noise, β is a constant threshold and $\mathcal{T}$ is a set containing all concurrent transmitting nodes. In practice, β is chosen to guarantee a certain quality in the communications. If inequality in (3) is satisfied for every scheduled link, it is said that a *feasible transmission pattern* has been formed. In this model, all the simultaneous transmissions are considered when evaluating whether a single transmission is feasible or not.

In the sequel, we introduce some definitions concerning the different areas that we consider associated to a transmission. These areas play a crucial role in the design of any MAC protocol. In our particular case, their proper design improves the performance of the gossip algorithm, as we show in the next section.

Definition 1. *The transmission range R_{max} is the maximum distance up to which a packet can be correctly received in absence of interference.*

Taking into account (3) and in the absence of interference, the maximum transmission radius R_{max} can be expressed as:

$$R_{\text{max}} = \left(\frac{P_t}{N_0 \beta}\right)^{\frac{1}{\gamma}} \tag{4}$$

Since a transmission to a node at distance equal to R_{max} implies that no other link can be simultaneously scheduled without collision, we force the length of the links to be scheduled to be shorter than $R_{\max}$. Accordingly, we define also a circular area of radius $R_\rho < R_{\max}$:

Definition 2. *The intended transmission range R_ρ of a node j defines the circular area containing all the neighbors that node j can communicate with.*

This intended transmission range can be expressed as:

$$R_\rho = \rho R_{\max} = \rho \left(\frac{P_t}{N_0 \beta}\right)^{\frac{1}{\gamma}} \tag{5}$$

with $0 < \rho < 1$. Notice that the value of ρ must be large enough for the network to be connected, namely, in order to guarantee a multi-hop path between every pair of nodes with high probability. It has been shown in [26] that for randomly and uniformly deployed large-scale networks, the critical radius for connectivity is $L\sqrt{\frac{\log N}{\pi N}}$, so we can choose the value of ρ in the following range $L\sqrt{\frac{\log N}{\pi N R_{\max}^2}} < \rho < 1$.

Definition 3. *The collision area associated to a specific link between transmitter node j and receiver node i is the circular area of radius Υ_{ji} and centered at i, inside which, no other node can transmit without corrupting the transmission from j to i.*

From (3), and considering that a collision occurs at node i if SINR $< \beta$, the following expression can be readily obtained:

$$\Upsilon_{ji} = \left(\frac{P_t}{\frac{P_t}{\beta d_{ji}^{\gamma}} - \sum_{v \in \mathcal{T}, v \neq j} \frac{P_t}{d_{iv}^{\gamma}} - N_0}\right)^{\frac{1}{\gamma}} \tag{6}$$

All the areas and radii described before ($R_{\max}$, R_ρ, Υ_{ji}) are shown in Fig. 1 and summarized in Table 1.

Finally, when a node j performs a transmission, and in order to protect the corresponding link, other nodes around it might decide not to transmit. This concept leads us to the following definition:

Definition 4. *The inhibition area S^j_{inh} of a transmitter node j is defined as the area around it that includes all nodes that are inhibited due to its transmission.*

The main objective of this area is to protect the transmission from node j to any node i within its intended transmission range R_ρ, by inhibiting as many nodes from the collision area Υ_{ji} as possible. However, since each area is centered at a different point, this objective entails a balance between the number of interfering nodes that are not inhibited (hidden terminal), and the number of non-interfering nodes that are inhibited (exposed terminal). As opposed to Υ_{ji}, whose value is completely determined by the wireless medium and the node transceiver sensibility, the inhibition area S^j_{inh} is a design parameter of the protocol, and its exact determination is a key point in the design of any MAC policy based on carrier sense. This issue is tackled in Section IV-B for our specific protocol design.

C. Graph Theory

The sequence of instantaneous connectivity patterns that result from applying any MAC protocol can be modeled as a time-varying graph $\mathcal{G}(k) = (\mathcal{V}, \mathcal{E}(k))$, consisting of a constant set $\mathcal{V}$ of N nodes and a set $\mathcal{E}(k) \subseteq \mathcal{E}$ of directed links that changes at each iteration k. The set $\mathcal{E} = \{e_{ji} \| d_{ji} \leq R_\rho\}$ denotes all the links that are susceptible to be scheduled, that is, the links between pairs of nodes that are at distance lower or equal to R_ρ and e_{ji} denotes a directed link from node j to node i, which indicates that there exists a directed information flow from node j to node i. While the set $\mathcal{E}$ includes enough links to ensure that the underlaying graph results in a strongly connected graph (possibly undirected), as we will explain later, each subset of links $\mathcal{E}(k)$ usually correspond to a directed and disconnected one.

Given a time-varying graph $\mathcal{G}(k)$, we can assign an $N \times N$ adjacency matrix $\mathbf{A}(k)$ where an entry $a_{ji} = [\mathbf{A}(k)]_{ji}$ is equal to 1 if $e_{ji} \in \mathcal{E}(k)$ and 0 otherwise. The set of neighbors of a node j is defined as $\mathcal{N}_j(k) = \{i \in \mathcal{V} : e_{ji} \in \mathcal{E}(k)\}$ and the degree matrix $\mathbf{D}(k)$ is a diagonal matrix whose entries are given by $d_j = [\mathbf{D}(k)]_{jj} = |\mathcal{N}_j(k)|$. Then, the Laplacian of a graph $\mathcal{G}(k)$ is a matrix defined as $\mathbf{L}(k) = \mathbf{D}(k) - \mathbf{A}(k)$, whose second smallest eigenvalue[1] $\lambda_2(\mathbf{L}(k))$ is the so-called algebraic connectivity. In general, and specifically for asymmetric gossip schemes, instantaneous matrices $\mathbf{A}(k)$ and $\mathbf{L}(k)$ are non symmetric. Besides, if we denote by $p_{ji} = [\mathbf{P}]_{ji}$ the probability of establishing a link from node j to node i, we have that $\mathbb{E}[\mathbf{A}(k)] = \bar{\mathbf{A}} = \mathbf{P}$, where $\mathbf{P}$ is the connection probability matrix. In Section IV-A, we tackle how this matrix $\mathbf{P}$ can be tuned by the MAC layer, such that the accuracy of the gossip algorithm is improved.

D. Gossip algorithms

A gossip algorithm is a distributed procedure, where, at each iteration k, one or several nodes send their value to a neighbour according to the decision of the MAC protocol being used. Then, every node i that receives a packet from a neighbor j updates its current value as follows:

$$x_i(k+1) = (1-\alpha)x_i(k) + \alpha x_j(k) \tag{7}$$

where α is the step size of the algorithm, which should be chosen to ensure convergence [9]. The rest of the nodes remain unchanged. By using matrix notation, we can write (7) in a more compact form as $\mathbf{x}(k+1) = \mathbf{W}(k)\mathbf{x}(k)$, where:

$$\mathbf{W}(k) = \mathbf{I} - \alpha \mathbf{L}(k) \tag{8}$$

is the instantaneous weight matrix at the k-th iteration (see Table I). Given some initial values, if we store them in a vector $\mathbf{x}(0) = [x_1(0) \dots x_N(0)]^T$, at each iteration k of the gossip algorithm, we have that $\mathbf{x}(k) = \mathbf{W}(k) \dots \mathbf{W}(0)\mathbf{x}(0)$.

The final objective of this gossip algorithm is that all nodes of the network asymptotically compute the average of the initial values of the nodes. In this way, if we define this average as $x_{\text{avg}} = \frac{1}{N}\sum_{i=1}^{N} x_i(0)$, the goal of the process is that the following holds:

$$\lim_{k\to\infty} \mathbf{x}(k) = x_{\text{avg}}\mathbf{1} \tag{9}$$

[1]We are assuming that the eigenvalues of a instantaneous Laplacian matrix are arranged in increasing order, i.e. $0 = \lambda_1(\mathbf{L}(k)) \leq \lambda_2(\mathbf{L}(k)) \leq \dots \leq \lambda_N(\mathbf{L}(k))$.

However, since $\mathbf{x}(k)$ is a random vector, at each iteration k of the algorithm, there is a deviation that can be quantified in terms of the mean square error (MSE), defined as:

$$\text{MSE}(k) = \frac{1}{N}\mathbb{E}\left[\|\mathbf{x}(k) - x_{\text{avg}}\mathbf{1}\|_2^2\right] \tag{10}$$

The matrices $\mathbf{W}(k)$ are random and independent at each iteration. By considering (8), it becomes clear that $\mathbf{W}(k)$ and $\mathbf{L}(k)$ have the same eigenvectors, and that their eigenvalues are related as: $\lambda_i(\mathbf{W}) = 1 - \alpha\lambda_{N-i+1}(\mathbf{L})$. Therefore, the largest eigenvalue of $\mathbf{W}(k)$ is equal to 1, and the associated right eigenvector is the all-one vector $\mathbf{1}$. However, and due to the random nature of $\mathbf{W}(k)$, nothing can be stated about the associated left eigenvector, which becomes a random vector and is different, in general, for each $\mathbf{W}(k)$. Then, provided that the graph is strongly connected on average, namely $\lambda_2\left(\overline{\mathbf{L}}\right) \neq 0$ with $\overline{\mathbf{L}} = \mathbb{E}[\mathbf{L}(k)]$, we have that [8]:

$$\lim_{k\to\infty}\mathbf{W}(k)\mathbf{W}(k-1)\ldots\mathbf{W}(0) = \mathbf{1}\mathbf{m}^T$$

where $\mathbf{m}$ is a random vector with $\sum_{i=1}^{N} m_i = 1$ and $0 \leq m_i \leq 1$. It means that the product of the weight matrices asymptotically converges to a rank-one matrix with equal rows, and (9) becomes:

$$\lim_{k\to\infty}\mathbf{x}(k) = \mathbf{1}\mathbf{m}^T\mathbf{x}(0) \tag{11}$$

which implies that the nodes reach consensus to the common random value:

$$\hat{x}_{\text{avg}} = \mathbf{m}^T\mathbf{x}(0) = \sum_{i=1}^{N} m_i x_i(0) \tag{12}$$

Then, by performing the dynamics in (7), every node is able to compute an estimator $\hat{x}_{\text{avg}}$ of x_{avg}, whose value depends on the left eigenvector associated to the eigenvalue 1 of each of the weight matrices applied during the process. The performance of the estimator is given by the asymptotic value of the MSE:

$$\text{MSE}(\hat{x}_{\text{avg}}) = \lim_{k\to\infty}\text{MSE}(k) = \mathbb{E}\left[(x_{\text{avg}} - \hat{x}_{\text{avg}})^2\right] \tag{13}$$

which can be decomposed as follows:

$$\text{MSE}(\hat{x}_{\text{avg}}) = \left(\mathbb{E}\left[x_{\text{avg}} - \hat{x}_{\text{avg}}\right]\right)^2 + \mathbb{E}\left[\left(\hat{x}_{\text{avg}} - \mathbb{E}\left[\hat{x}_{\text{avg}}\right]\right)^2\right] \tag{14}$$

where the first term on the right side denotes the bias of the estimator, and the second term represents its variance.

III. Sufficient conditions to ensure an estimation with minimum MSE value

Ideally, nodes should be able to obtain an estimation of the average such that the bias is zero and the variance is minimum. The minimum value that can be attained for this second term equals the value of the variance of the initial values, which is zero for the case of considering initial deterministic values. In the following, we present the conditions to ensure an estimation with minimum MSE value.

Proposition 1. *The estimator $\hat{x}_{avg}$ is unbiased if and only if $\mathbf{P}\mathbf{1} = \mathbf{1}^T\mathbf{P}$, that is, the sum of every column of the probability matrix $\mathbf{P}$ is the same as the corresponding row.*

Proof: If we compute the expectation of the value of $\mathbf{x}(k)$ in the limit, we have the following:

$$\mathbb{E}[\hat{x}_{\text{avg}}\mathbf{1}] = \mathbb{E}[\lim_{k\to\infty}\mathbf{W}(k)\ldots\mathbf{W}(0)\mathbf{x}(0)] = \lim_{k\to\infty}\overline{\mathbf{W}}^k\mathbf{x}(0)$$

where we have applied the independence between the matrices $\mathbf{W}(k)$ and also from vector $\mathbf{x}(0)$. By applying the Perron-Frobenius Theorem, and considering that $\mathbf{1}$ is the right eigenvector of $\overline{\mathbf{W}}$ associated to eigenvalue 1, we have that:

$$\lim_{k\to\infty}\overline{\mathbf{W}}^k = \frac{\mathbf{1}\overline{\mathbf{m}}^T}{\overline{\mathbf{m}}^T\mathbf{1}} \tag{15}$$

where $\overline{\mathbf{m}}$ is the left eigenvector of $\overline{\mathbf{W}}$ associated to the eigenvalue 1. If we express $\overline{\mathbf{L}} = \overline{\mathbf{D}} - \mathbf{P}$, and by considering that $\mathbf{1}$ is the right eigenvector of $\overline{\mathbf{L}}$ associated to eigenvalue 0, we have that:

$$\mathbf{P}\mathbf{1} = \overline{\mathbf{D}}\mathbf{1} \tag{16}$$

Moreover, since $\overline{\mathbf{W}}$ and $\overline{\mathbf{L}}$ have the same eigenvectors, the following holds:

$$\overline{\mathbf{m}}^T\mathbf{P} = \overline{\mathbf{m}}^T\overline{\mathbf{D}} \tag{17}$$

From (16) and (17), we have that if the sum of the rows of $\mathbf{P}$ is equal to the sum of the columns, then $\mathbf{1}\mathbf{P}^T = \mathbf{P}\mathbf{1} = \overline{\mathbf{D}}\mathbf{1} = \mathbf{1}\overline{\mathbf{D}}^T$. Therefore, $\overline{\mathbf{m}}$ is the all-one vector, which from (15) implies that:

$$\mathbb{E}[\hat{x}_{\text{avg}}\mathbf{1}] = \frac{1}{N}\mathbf{1}\mathbf{1}^T\mathbf{x}(0) = x_{\text{avg}}\mathbf{1}$$

which means that the estimator is unbiased.

On the other hand, if the estimator is unbiased, then $\overline{\mathbf{m}}$ must be the all-one vector, which implies that $\mathbf{1}\mathbf{P}^T = \mathbf{1}\overline{\mathbf{D}}^T = \overline{\mathbf{D}}\mathbf{1} = \mathbf{P}\mathbf{1}$, and the sum of the columns of $\mathbf{P}$ is the same as the sum of its rows.

■

Corollary 1. *If the probability matrix $\mathbf{P}$ is symmetric, the estimator $\hat{x}_{avg}$ is unbiased.*

Although ensuring an unbiased estimation is important, large values of the variance term in (14) could make the final value of the MSE unacceptable. The variance of the consensus estimation depends on both the variance of the initial values of the nodes and the covariance of vector $\mathbf{m}$. Since we cannot affect the former by applying any specific MAC design, we focus here on the covariance of vector $\mathbf{m}$, which is expressed in [8] as follows:

$$\begin{aligned}\text{vec}\left(\text{cov}\left(\mathbf{m}\right)\right) \quad = \quad & \mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]\right\} - \\ & -\mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\right]\right\}\otimes\mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\right]\right\}\end{aligned}$$

where $\text{vec}(\cdot)$ is the vectorization operator, $\otimes$ stands for the Kronecker product and $\mathbf{v}_1(\cdot)$ denotes the normalized left eigenvector corresponding to the unit eigenvalue.

Proposition 2. *The covariance of vector $\mathbf{m}$ is zero if matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ is balanced.*

Proof: By properties of the Kronecker product, it is accomplished that:

$$\mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\right]\right\}\otimes\mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\right]\right\} = \mathbf{v}_1\left\{\mathbb{E}\left[\mathbf{W}(k)\right]\otimes\mathbb{E}\left[\mathbf{W}(k)\right]\right\}$$

Thus, a sufficient condition for $\mathrm{vec}\left(\mathrm{cov}\left(\mathbf{m}\right)\right)$ to be zero is that both matrices $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$ and $\mathbb{E}\left[\mathbf{W}(k)\right]\otimes\mathbb{E}\left[\mathbf{W}(k)\right]$ share the same left eigenvector corresponding to the unit eigenvalue. This directly holds when both matrices are the same, however this is not possible since it requires the entries of $\mathbf{W}(k)$ to be statistically independent, or in other words that there is no uncertainty in the process. Alternatively, the specific matrix design of $\mathbf{W}(k)$ in (8) implies that matrices $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$ and $\mathbb{E}\left[\mathbf{W}(k)\right]\otimes\mathbb{E}\left[\mathbf{W}(k)\right]$ are both row stochastic, that is, the right eigenvector corresponding to the unit eigenvalue is the all ones vector. For the specific case of a symmetric probability matrix $\mathbf{P}$, it follows that $\mathbb{E}[\mathbf{W}(k)]$ is also symmetric[2], which entails that $\mathbf{v}_1\left\{\mathbb{E}[\mathbf{W}(k)]\otimes\mathbb{E}[\mathbf{W}(k)]\right\}=\mathbf{1}$. Therefore, the covariance of vector $\mathbf{m}$ is zero if $\mathbf{v}_1\left\{\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]\right\}=\mathbf{1}$, that is, if matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ is balanced. ■

Corollary 2. *If the matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ is symmetric, the estimator $\hat{x}_{avg}$ presents minimum variance.*

Theorem 1. *If the matrices $\mathbf{P}$ and $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ are both balanced, then the value of MSE$(\hat{x}_{avg})$ is minimum.*

Proof: The result of the theorem is an immediate consequence of both Proposition 1 and Proposition 2. ■

In a distributed scenario, the easiest way to ensure that matrices $\mathbf{P}$ and $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ are both balanced is to make them symmetric, since this does not require any global information or coordination from a central entity. While a symmetric matrix $\mathbf{P}$ can be attained by properly designing the MAC layer, as we show in the next section, the symmetry of matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ cannot be attained under the SINR interference model, as shown by the following result.

Proposition 3. *The matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ is symmetric if and only if instantaneous symmetric links are ensured.*

Proof: If instantaneous symmetric links are always ensured, all the instantaneous matrices $\mathbf{W}(k)$ are symmetric and then $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$ is symmetric. Thus the instantaneous symmetry condition is sufficient, but nothing can be directly inferred about whether this condition is also necessary.

To see that this condition is also necessary, we elaborate on the expression of $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$. Since $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$ is formed by the $(N\text{x}N)\text{x}(N\text{x}N)$ elements of the form $\mathbb{E}\left[w_{ij}(k)w_{kl}(k)\right]$ for $i,j,k,l=1\ldots N$, namely the expected values of the products between all elements of matrix $\mathbf{W}$, we can claim that $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$ is symmetric if and only if:

$$\mathbb{E}[w_{ij}(k)w_{kl}(k)]=\mathbb{E}[w_{ji}(k)w_{lk}(k)]\ \ \forall i,j,k,l=1\ldots N$$

By considering (8), we have the following:

$$\begin{aligned} w_{ii}(k) =&\ 1-\epsilon\sum_{j=1}^{N}a_i(k), \quad i=j \\ w_{ij}(k) =&\ \epsilon a_{ij}(k), \quad \text{otherwise} \end{aligned}$$

[2]For the specific design of matrix $\mathbf{W}(k)$ in (8), the expectation matrix $\mathbb{E}[\mathbf{W}(k)]$ depends exclusively on the probability matrix $\mathbf{P}$ in the form $\mathbb{E}[\mathbf{W}(k)]=\mathbf{I}-\epsilon\left(\mathrm{diag}(\mathbf{P}\cdot\mathbf{1})-\mathbf{P}\right)$.

If we multiply for different indexes, we take expectations and by noting that $\mathbb{E}[a_{ij}]=[\mathbf{P}]_{ij}=p_{ij}$, the diagonal entries of $\mathbb{E}\left[\mathbf{W}(k)\otimes\mathbf{W}(k)\right]$ can be expressed as follows (we drop the time index k for the sake of simplicity):

$$\begin{aligned} \mathbb{E}[w_{ii}^2] &= 1-2\epsilon\sum_{j=1}^{N}p_{ij}+\epsilon^2\sum_{j=1}^{N}\sum_{k=1}^{N}\mathbb{E}[a_{ij}a_{ik}] \\ \mathbb{E}[w_{ii}w_{jj}] &= 1-\epsilon\sum_{k=1}^{N}\left(p_{ik}+p_{jk}\right)+\epsilon^2\sum_{k=1}^{N}\sum_{l=1}^{N}\mathbb{E}[a_{ik}a_{jl}] \end{aligned}$$

which do not affect the symmetry of the matrix $\mathbb{E}[\mathbf{W}(k)\otimes\mathbf{W}(k)]$. Similarly, the rest of the entries which do affect the symmetry of this matrix are the following:

$$\mathbb{E}[w_{ii}w_{ik}]=\epsilon p_{ik}-\epsilon^2\sum_{j=1}^{N}\mathbb{E}[a_{ij}a_{ik}] \tag{18}$$

$$\mathbb{E}[w_{ii}w_{kl}]=\epsilon p_{kl}-\epsilon^2\sum_{j=1}^{N}\mathbb{E}[a_{ij}a_{kl}],\ \ i\neq k,l \tag{19}$$

$$\mathbb{E}[w_{ij}w_{kl}]=\epsilon^2\mathbb{E}[a_{ij}a_{kl}]\ \ i\neq j,k\neq l \tag{20}$$

For the entries determined by (18), the condition of symmetry can be stated as:

$$\mathbb{E}[a_{ij}a_{ik}]=\mathbb{E}[a_{ij}a_{ki}] \tag{21}$$

In the particular case of $j=k$, the condition in (21) becomes $\mathbb{E}[a_{ij}^2]=\mathbb{E}[a_{ij}a_{ji}]$. Since $a_{ij}^2=a_{ij}$ and by applying Bayes theorem, it is equivalent to:

$$\mathrm{prob}\left\{a_{ij}=1\right\}=\mathrm{prob}\left\{a_{ij}=1\right\}\mathrm{prob}\left\{a_{ji}=1|a_{ij}=1\right\}$$

which holds if and only if $\mathrm{prob}\left\{a_{ji}=1|a_{ij}=1\right\}=1$, that is, both links e_{ij} and e_{ji} always simultaneously activate, which makes the condition of instantaneous symmetric links necessary. ■

The result stated by **Proposition 3** implies that we cannot reach minimum variance under our setting, since nodes with strong hardware limitations cannot transmit and receive simultaneously under the SINR model. However, we can still ensure that, except for the entries of the form (18) with $j=k$, the rest of the entries are symmetric, implying that the resulting matrix is a small perturbation [27] of a symmetric one, leading, in most cases, to left eigenvectors close to the all-ones vector and to small variance values as a consequence.

In the next section, we relate the conditions for an unbiased estimation and minimum variance to the design of a specific MAC protocol. Then, we describe in detail our particular design and we show that this protocol ensures an unbiased estimation, while maintaining close to minimum values of the variance.

IV. Description of the protocol

This section is devoted to a detailed description of the proposed distributed link scheduling protocol based on carrier sense. After outlining the main features of the scheduling procedure, we present the design rule that guarantees symmetric connection probabilities, that is, a symmetric matrix $\mathbf{P}$. Then, we explain how the inhibition area of the nodes can be tuned in such a way that this design rule is fulfilled. Finally, we show that our design ensures that most of the entries of the matrix $\mathbb{E}[\mathbf{W}(k) \otimes \mathbf{W}(k)]$ are symmetric, leading to small variance values.

Our proposed scheme involves the simultaneous execution of the link scheduling protocol and the average gossip algorithm. At every iteration k of the gossip algorithm, a realization of the scheduling protocol is performed. All the links in $\mathcal{E}$ are initially labeled as unclassified, defining the link demand, which is common to every gossip iteration. Thus, the link demand at iteration $k+1$ is reestablished after a subset $\mathcal{E}(k)$ of links is activated at iteration k. At each scheduling step n, a randomly chosen node j signals the activation of a link, which involves a randomly chosen neighbor node i from the circular area associated to R_ρ. The receiver i updates its value according to (7) at the current gossip iteration. When the link e_{ji} is marked as activated, it causes all the unclassified links inside the inhibition area S^j_{inh} of transmitter node j to be marked as inhibited. The subsequent repetition of this activation-inhibition step leads to the following concept:

Definition 5. *The length of a particular feasible transmission pattern, denoted by η, is the number of simultaneous links that has been scheduled in a particular realization while satisfying the feasibility condition (3) for all of them.*

Therefore, an iteration of the gossip algorithm involves η transmissions due to the scheduled activation of the corresponding η links, such that $|\mathcal{E}(k)| = \eta$.

Accordingly, during one realization, at each step n of the scheduling protocol, $0 \leq n \leq \eta$, every link $e_{ji} \in \mathcal{E}$ is classified into one of the following sets:

- ACTIVATED $\mathcal{A}(n)$: contains all links marked as activated up to the n-th step of the link scheduling protocol. Initially, $\mathcal{A}(0) = \emptyset$. Then, only a new link is marked as activated at each step n, so that, $|\mathcal{A}(n)| = n$. After the η scheduling steps, we have that $\mathcal{E}(k) = \mathcal{A}(\eta)$.
- INHIBITED $\mathcal{I}(n)$: contains all links marked as inhibited up to the n-th step of the link scheduling protocol. At the initial step, we have that $\mathcal{I}(n) = \emptyset$. After the η scheduling steps, we have that $\mathcal{I}(\eta) = \mathcal{E} \setminus \mathcal{E}(k)$.
- UNCLASSIFIED $\mathcal{U}(n)$: contains the links not belonging to any of the previous sets. The initial set at each gossip iteration is given by $\mathcal{U}(0) = \mathcal{E} = \{e_{ji} | d_{ji} \leq R_\rho\}$.

From (7) and according to the previous classification of links, it follows that only the receiver nodes of the links that have been activated after the last η scheduling steps update their values in the k iteration of the gossip algorithm, that is:

$$x_i(k+1) = \begin{cases} (1-\alpha)x_i(k) + \alpha x_j(k) & \text{if } e_{ji} \in \mathcal{A}(\eta) \\ x_i(k) & \text{otherwise} \end{cases}$$

Given this scheduling procedure, in the next subsection, we derive the condition under which symmetric connection probabilities (symmetric matrix $\mathbf{P}$) are obtained.

A. Symmetric connection probabilities

Given a link e_{ji}, if we denote by $\mathcal{S}_{ji}(n)$ the set of unclassified links at step n whose activation imply its inhibition (transmission through link e_{ji} unfeasible), and based on the previous description of the link scheduling protocol, we have the following result:

Proposition 4. *Symmetric probabilities of communication are ensured, if the number of unclassified links that can inhibit link e_{ji} is the same as the number of those that can inhibit link e_{ij} at every scheduling step l. In other words, if $|\mathcal{S}_{ji}(l)| = |\mathcal{S}_{ij}(l)|$ $\forall l = 1 \ldots \eta$, then matrix $\mathbf{P}$ is symmetric.*

Proof: The probability $P^A_{ji}(n)$ of activating a link e_{ji} at the n-th step of the link scheduling is the combination of two events:

- The link e_{ji} remains unclassified after $n-1$ scheduling steps.
- The link e_{ji} is chosen for activation at step n.

Thus, we can write the following:

$$P^A_{ji}(n) = p\{e_{ji} \in \mathcal{U}(n-1)\}\, p\{e_{ji} \in \mathcal{A}(n) | e_{ji} \in \mathcal{U}(n-1)\}$$

The link e_{ji} belongs to $\mathcal{U}(n-1)$ if it does belong neither to $\mathcal{A}(n-1)$ nor to $\mathcal{I}(n-1)$:

$$\begin{aligned} & p\{e_{ji} \in \mathcal{U}(n-1)\} = \\ & = (1 - p\{e_{ji} \in \mathcal{A}(n-1)\})\,(1 - p\{e_{ji} \in \mathcal{I}(n-1)\}) \end{aligned}$$

The link e_{ji} does not belong to $\mathcal{A}(n-1)$ if it has not been activated in any of the previous $n-1$ steps:

$$(1 - p\{e_{ji} \in \mathcal{A}(n-1)\}) = \prod_{l=1}^{n-1} \left(1 - P^A_{ji}(l)\right)$$

Similarly, the link e_{ji} does not belong to $\mathcal{I}(n-1)$ if it has not been inhibited in any previous step. Furthermore, the link e_{ji} is inhibited at step $1 \leq l \leq n-1$ if a link from $\mathcal{S}_{ji}(l)$, among all candidates contained in $\mathcal{U}(l)$, is chosen for activation. Since both $|\mathcal{S}_{ji}(l)|$ and $|\mathcal{U}(l)|$ are random variables, we have the following:

$$\begin{aligned} & (1 - p\{e_{ji} \in \mathcal{I}(n-1)\}) = \\ & = \prod_{l=1}^{n-1} \left(1 - \sum_{\tau,\upsilon} \frac{\tau}{\upsilon}\; p\; \{|\mathcal{S}_{ji}(l)| = \tau, |\mathcal{U}(l)| = \upsilon\}\right) = \\ & = \prod_{l=1}^{n-1} \left(1 - \mathbb{E}\left[\frac{|\mathcal{S}_{ji}(l)|}{|\mathcal{U}(l)|}\right]\right) \end{aligned}$$

The probability that link e_{ji} is chosen for activation at scheduling step n, provided that it has remained unclassified during the previous $n-1$ steps, can be expressed as:

$$\begin{aligned} & p\{e_{ji} \in \mathcal{A}(n) | e_{ji} \in |\mathcal{U}(n)|\} = \\ & = \sum_{\upsilon} \frac{1}{\upsilon} p\{|\mathcal{U}(n)| = \upsilon\} = \\ & = \mathbb{E}\left[\frac{1}{|\mathcal{U}(n)|}\right] \end{aligned}$$

Accordingly, the probability of activation of a specific link e_{ji} at the n-th step of the scheduling process is given by the following recursive expression:

$$P^A_{ji}(n) = \mathbb{E}\left[\frac{1}{|\mathcal{U}(n)|}\right] \cdot \prod_{l=1}^{n-1} \left(1 - \mathbb{E}\left[\frac{|\mathcal{S}_{ji}(l)|}{|\mathcal{U}(l)|}\right]\right) \left(1 - P^A_{ji}(l)\right)$$

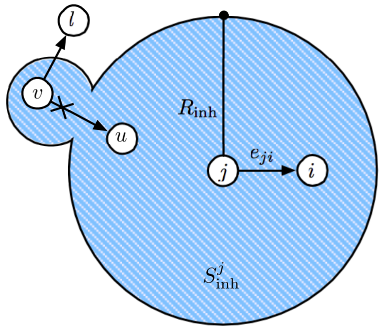
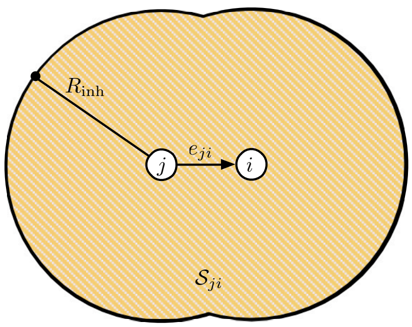


Fig. 2. Inhibition area S^j_{inh} of node j covers every link with an endpoint inside the circle of radius R_{inh} centered at j. Consequently, any node located at a distance shorter than R_{inh} from node j is inhibited not only for sending but also for receiving.



(a) $\mathcal{S}_{ji}$

Fig. 3. Set of nodes $\mathcal{S}_{ji}$ whose activation would cause the inhibition of the transmission from node j to i. This set is the same as the one that includes the nodes whose activation would inhibit the transmission from i to j.

Thus, it follows that if $|\mathcal{S}_{ji}(l)| = |\mathcal{S}_{ij}(l)|$ for $l = 1 \dots \eta$, then $P^A_{ji}(n) = P^A_{ij}(n)$ for $n = 1 \dots \eta$. Then, since $[\mathbf{P}]_{ij} = \sum_{n=1}^{\eta} P^A_{ij}(n)$, it is accomplished that $[\mathbf{P}]_{ij} = [\mathbf{P}]_{ji}$, which concludes the proof. ■

Next, we analyze how to design the inhibition area in order to achieve: a) a symmetric matrix $\mathbf{P}$ and b) collision-free connectivity patterns that lead to an unbiased estimation with small variance.

B. Design of the inhibition area

The inhibition area S^j_{inh} controlled by a transmitter node j, namely the nodes that are inhibited for transmitting while this node j is performing a transmission, is a key design aspect in any MAC protocol, since it entails a trade-off between the hidden and the exposed terminal problems. As explained in Section II, in order to completely avoid collisions and generate feasible transmission patterns, the inhibition area S^j_{inh} of a node j should cover the collision area Υ_{ji} of node i, inhibiting the transmission of any neighbor of node i. However, in our specific setting, the inhibition area has an additional requirement: it must be designed in such a way that **Proposition 4** holds, so that a symmetric probability matrix $\mathbf{P}$ is ensured. Accordingly, and assuming continuous time so that two nodes cannot transmit exactly at the same time, we propose the following result:

Theorem 2. *Given any active link e_{vu}, if every other potential link with an endpoint at a distance from the transmitter v shorter than the inhibition radius R_{inh} given by:*

$$R_{inh} = \left(\frac{(\eta - 1)P_t}{\frac{N_0}{\rho^\gamma} - N_0} \right)^{\frac{1}{\gamma}} + R_\rho \tag{22}$$

is inhibited, then feasible transmission patterns with an associated symmetric probability matrix $\mathbf{P}$ are generated.

Proof: We first show that symmetric probabilities of connection are generated. From what is stated in the Theorem, link e_{ji} is inhibited if either node j or node i are located at a distance shorter than (22) from an active transmitter v. Therefore, the set of links $\mathcal{S}_{ji}(n)$, whose activation implies the inhibition of link e_{ji}, becomes:

$$\mathcal{S}_{ji}(n) = \{e_{vu} : \min\{d_{vj}, d_{vi}\} \leq R_{\text{inh}}\} \tag{23}$$

Since the inhibition radius in (22) is a common value to all links, then $\mathcal{S}_{ji}(n) = \mathcal{S}_{ij}(n)$, which, by Proposition (4), implies that $P^A_{ji}(n) = P^A_{ij}(n)$, for $n = 1 \dots \eta$.

Now, we prove that, with the defined strategy, at each realization of the scheduling protocol, collision free patterns are obtained. Since the maximum number of simultaneous active links for a given realization is η, the collision area for a link between j and i defined by (6) must be particularized for the case of $\eta - 1$ interferers. Besides, since the distance between those $\eta - 1$ interferers and the receiver i cannot be predicted, we must assume the worst case scenario, that is, the $\eta - 1$ interferers are located at the same distance, and as close as possible to node i such that a collision is not provoked. By considering this worst case scenario, the radius for this new collision area can be computed from (3) as:

$$\Upsilon^\eta_{ji} = \left(\frac{\beta(\eta - 1)P_t}{\frac{P_t}{d^\gamma_{ji}} - N_0\beta} \right)^{\frac{1}{\gamma}} \tag{24}$$

This collision area is protected if the transmission from node j to node i inhibits the transmission of any other node in a circular area centered at j with radius $\Upsilon^\eta_{ji} + d_{ji}$. As explained before, a condition to ensure symmetric probabilities of communication is that the inhibition radius R_{inh} is common to every node j, hence the associated inhibition area S^j_{inh} should cover the collision area of the longest link in the network. Since $R_\rho \geq \max_{e_{ji} \in \mathcal{E}} \{d_{ji}\}$, the expression for R_{inh} in (22) implies that $R_{\text{inh}} \geq \max_{e_{ji} \in \mathcal{E}} \{\Upsilon_{ji} + d_{ji}\}$, and a feasible pattern is generated as a consequence, while symmetric probabilities of communication are also ensured. ■

Remark 1. *Based on this strategy, a transmission from node j to node i is inhibited not only by the activation of any transmitter node inside a circular area of radius R_{inh} centered at node j, but also by the activation of any transmitter node inside the same area centered at the intended receiver i (see Fig. 3). It is straightforward to see that the previous set of nodes is the same as the one that inhibits the transmission from node i to node j.*

Theorem 3. *Our inhibition area design ensures that all the entries of the matrix $\mathbb{E}[\mathbf{W}(k) \otimes \mathbf{W}(k)]$ are symmetric, except for the ones of the type (18) with $j = k$.*

Proof: Firstly, for our protocol to ensure that the entries of the type (18) are symmetric, it must be accomplished that:

$$\mathbb{E}[a_{ij}a_{ik}] = \mathbb{E}[a_{ij}a_{ki}]$$

For $j \neq k$, since the random variables $a_{ij}(k)$ can only take the values 0 and 1, we have that the expectation $\mathbb{E}[a_{ij}a_{ik}]$ is equal to prob $\{(a_{ij} = 1) \cap (a_{ik} = 1)\}$, namely the probability that both links e_{ij} and e_{ik} are simultaneously activated. Then, entries in (18) with $j \neq k$ are symmetric if and only if:

$$\text{prob}\{(a_{ij} = 1) \cap (a_{ik} = 1)\} = \text{prob}\{(a_{ij} = 1) \cap (a_{ki} = 1)\}$$

which for our setting are both equal to zero, since we are considering unicast communications and a SINR interference model, that is, no node can transmit and receive at the same time and a node can communicate with one and only one neighbour at a time.

Secondly, for our protocol to ensure that the entries of the type (19) are symmetric, it must be accomplished that:

$$\mathbb{E}[a_{ij}a_{kl}] = \mathbb{E}[a_{ji}a_{lk}]$$

For the case $i = k, j = k, l$, and by noting that $a_{ij}^2 = a_{ij}$, the condition becomes $p_{ij} = p_{ji}$, which always holds because of the symmetry of the matrix $\mathbf{P}$. For the case $i, j \neq k, l$, since again the random variables $a_{ij}(k)$ can only take the values 0 and 1, we have that the expectation $\mathbb{E}[a_{ij}a_{kl}]$ is equal to prob $\{(a_{ij} = 1) \cap (a_{kl} = 1)\}$, namely the probability that both links e_{ij} and e_{kl} are simultaneously activated. Then, entries in (19) with $i, j \neq k, l$ are symmetric if and only if:

$$\text{prob}\{(a_{ij} = 1) \cap (a_{kl} = 1)\} = \text{prob}\{(a_{ji} = 1) \cap (a_{lk} = 1)\}$$

namely the probability of simultaneous activation of any two links is the same that the probability of simultaneous activation of their both symmetric counterparts. Since our protocol is ensuring that the inhibition areas of a link and its symmetric counterpart are exactly the same, their statistical behaviour is also the same, making the probability of simultaneously activating any two links is the same than the one of their symmetric counterparts.

Finally, for our protocol to ensure that the entries of the type (20) are symmetric, it must be accomplished that:

$$\mathbb{E}[a_{ij}a_{kl}] = \mathbb{E}[a_{ij}a_{lk}], \quad i \neq k, l$$

Similarly to the previous case, it can be expressed as follows:

$$\text{prob}\{(a_{ij} = 1) \cap (a_{kl} = 1)\} = \text{prob}\{(a_{ij} = 1) \cap (a_{lk} = 1)\}$$

which entails that the probability of simultaneously activation of any two links is the same that the probability of simultaneous activation of one of the links and the symmetric counterpart of the other. For the same reason stated in the previous case, our protocol ensures the symmetry of this type of entries.

■

The result of **Theorem 3** implies that most of the entries of the matrix $\mathbb{E}[\mathbf{W}(k) \otimes \mathbf{W}(k)]$ are symmetric, which means that the resulting matrix is a small perturbation of a symmetric one, which intuitively results in a small covariance of vector $\mathbf{m}$.

C. Implementation details

First, we describe our algorithm in global terms by using the three link sets ($\mathcal{A}$, $\mathcal{I}$, $\mathcal{U}$) introduced in the previous section. Then, we show that all the global concepts can be easily mapped to local operations to the network nodes and implemented in a distributed manner.

Globally, our algorithm works as follows. At each iteration of the gossip algorithm, the sets $\mathcal{A}(0)$ and $\mathcal{I}(0)$ are initialized with no links on them, while the set $\mathcal{U}(0)$ is initialized with every link in $\mathcal{E}$. At each scheduling step, a link between a transmitter node j and a receiver node i is randomly chosen from the set $\mathcal{U}(n)$ and this unique link is added to the set of active links $\mathcal{A}(n)$ at scheduling step n. Then, the activation of this new link implies that no other transmission is simultaneously scheduled in the inhibition area around the receiver node i. Hence, these links are added to the set $\mathcal{I}(n)$ and are removed from the set $\mathcal{U}(n)$. The value chosen for η determines when the algorithm finishes. There are two scenarios for finishing the link scheduling: 1) if the used value for η is large, the algorithm stops when the set $\mathcal{U}(n)$ is empty and 2) if the used value for η is small, the link scheduling protocol is finalized when η simultaneous links are obtained. Both stopping conditions satisfy the feasibility of the transmission pattern. However, the first stopping criteria implies that η would become a random variable, being difficult to make a proper design of the inhibition area. For that reason, we work with values of η generally smaller than the optimal one. An efficient methodology to compute a proper value for η based on a worst case scenario is presented in our previous work [23]. The details are omitted here to avoid extending the length of the manuscript unnecessarily.

Algorithm 1 Link scheduling described in terms of $\mathcal{A}$, $\mathcal{I}$, $\mathcal{U}$

Require: $\mathcal{A}(0)$, $\mathcal{I}(0)$, $\mathcal{U}(0)$, η
Ensure: $\mathcal{U}(n)$ is empty OR $n = \eta$
$n = 0$
while $\mathcal{U}(n)$ is not empty AND $n \leq \eta$ **do**
  $e_{ji} \leftarrow$ choose uniformly at random a link from $\mathcal{U}(n)$
  add e_{ji} to $\mathcal{A}(n)$
  remove e_{ji} from $\mathcal{U}(n)$
  for $u = 1$ to N **do**
    for $v = 1$ to N **do**
      if $d_{uj} \leq R_{\text{inh}}$ OR $d_{vj} \leq R_{\text{inh}}$ **then**
        add e_{uv} to $\mathcal{I}(n)$
        remove e_{uv} from $\mathcal{U}(n)$
      end if
    end for
  end for
  $n = n + 1$
end while

Although the sets $\mathcal{A}(n)$, $\mathcal{I}(n)$ and $\mathcal{U}(n)$ used to describe **Algorithm 1** and **Proposition 2** are defined in global terms, these have a direct correspondence to local concepts. In particular, each transmitter node j is able to classify as active, inhibited and unclassified every link e_{ji} in which node j acts as the transmitter. This is always a local operation that only requires a state variable per neighbor.

In order to implement **Algorithm 1** in a distributed manner, we can use a similar approach to [18]. In particular, we divide each scheduling time-slot into a control slot and a data slot. The purpose of the control slot is to generate a collision-free transmission pattern used for the exchange of the consensus data in the corresponding data slot. The set S_{ji} of conflicting links for each link e_{ji} is given by (23). Note that in order to satisfy the symmetry of $\mathbf{P}$, we have that $S_{ji} = S_{ij}$, which is also a condition stated in [18] for the algorithm to work. The control slots are divided in mini-slots n, where the INTENT messages[3] are sent and the scheduling sets are updated accordingly. The way that the control slot works is similar to any CSMA protocol based on the standard IEEE 802.15.4. In particular, nodes try to send INTENT messages with a frequency that depends of the back-off timer or contention windows (CW). The rest of the procedure is described in detail in [18] and summarized for convenience in **Algorithm 2** with the difference that in our setting all links must be equiprobable to satisfy the symmetry property of matrix $\mathbf{P}$. A simple solution to satisfy this last condition is to make the back-off timer of each node inversely proportional to its number of neighbors. The result of **Algorithm 2** is the final transmission schedule (set of links marked as active) at each consensus iteration k.

Algorithm 2 Link scheduling per node

Initializing phase:

Each node j starts its own random timer with a value T_j inversely proportional to its number of neighbors

Every link e_{ji} is locally marked as unclassified.

Link scheduling:

After T_j control mini-slots, the timer of node j expires.

if node j hears and INTENT message from a link in S_{ji}, before the $(T_j + 1)$-th control mini-slot **then**

  — Link e_{ji} is marked as inhibited and node j does not transmit an INTENT message for it

else if node j does not hear and INTENT message from a link in S_{ji}, before the $(T_j + 1)$-th control mini-slot **then**

  — It sends an INTENT message to all links in S_{ji} at the beginning of the $(T_j + 1)$-th control mini-slot

  if there is a collision (i.e., concurrent INTENT message in the same control mini-slot) **then**

    — Link e_{ji} is marked as inhibited

  else

    — Link e_{ji} is marked as active with certain probability if no link $e_{uv} \in \mathcal{S}_{ji}$ active in previous data slot, keeping previous state otherwise

  end if

end if

V. Numerical results

In this section, we numerically evaluate our cross-layer scheme for different parameters: P_t, η, α and ρ, showing, that our design improves the accuracy and the power consumption while maintaining competitive convergence rates as compared to state of the art work.

A. Simulation Scenario

We model a WSN as a uniformly random deployed network of $N = 1000$ nodes inside a $2D$ unit square area. The information is mixed as described in (7), where the instantaneous topology at the k-th iteration determines which data is mixed. Additionally, channel gains are computed based on node positions, and on the radio propagation model. Radio signal propagation is assumed to follow log-normal shadowing, with path loss exponent $\gamma = 3$. For a given background noise $N_0 = 10^{-9}$mW per meter and a given value of $\beta = 10$, we choose a combination of the values P_t and ρ that ensures connectivity on average, that is, the set of edges $\mathcal{E}$ contains enough links to ensure the existence of a multi-hop path between every pair of nodes. Finally, since the value of α plays a crucial role, we analyze the consensus performance for different values of this parameter.

[3]An INTENT message is sent for every link to be activated in order to announce this decision at the end of the corresponding node back-off timer. Then, the link is marked as active if and only if all the links in S_{ji} receive the INTENT message without collision in the control slot.

In order to compare our link scheduling protocol with a CSMA protocol based on the standard IEEE 802.15.4, we adopt the time equivalence illustrated in Figure 4. Particularly, the number of milliseconds required by our link scheduling to create a connectivity pattern, including both the control and the data slots, is what determines what an iteration is. In that period of time, a CSMA protocol activates a random number of links and a random number of collisions occur. In both type of protocols, since these are based on carrier sense, the number of links correctly activated and the energy efficiency mainly depend on the value of the contention window (CW). Note that the philosophy used in the control slot is similar to the normal operation of the standard IEEE 802.15.4. An important difference is that our protocol requires a different CW for each node. In particular, each node uses a value of CW inversely proportional to its degree, as explained in Section IV. In order to make a fair comparison, the value of CW used for the standard IEEE 802.15.4 equals the average of the values used for our protocol. The values of CW tested are [40, 80, 120, 160, 200, 400, 600, 800, 1000, 2000]. It is important to remark that the size of the control packets is, in general, significantly smaller than the data ones. Since this affects the performance of our protocol, this is also evaluated later.

Finally, since different combinations of gossip algorithms and MAC protocols may lead to different MSE values and in order to make a fair comparison, we operate as follows: i) a scheme is said to converge to a MSE value when its value changes less than 10^{-3} in two successive iterations and ii) the largest (worst) MSE value attained by any of the schemes considered is taken as the stoping criteria for the

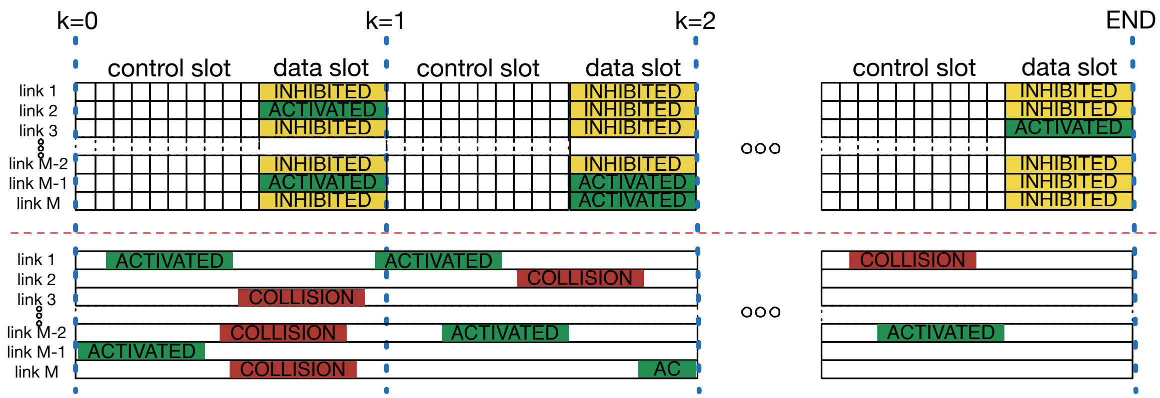


Fig. 4. Iteration equivalence between our link scheduling implementation (top) and the CSMA protocols based on the standard IEEE 802.15.4 (bottom).

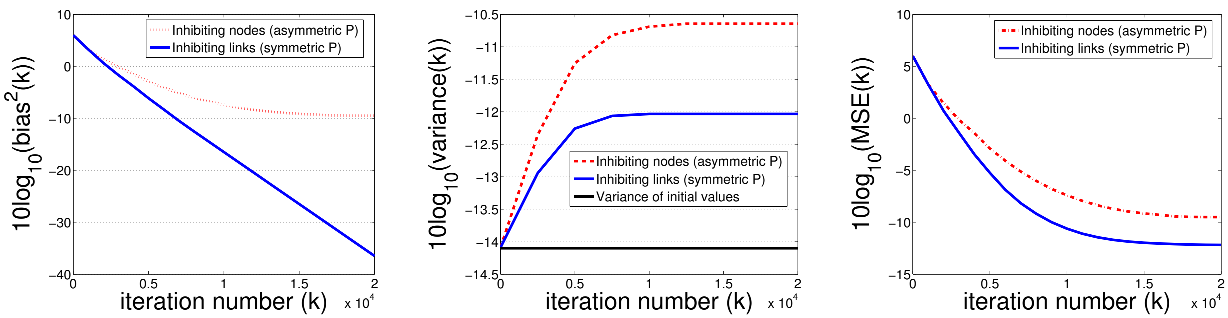


Fig. 5. Bias, variance and MSE of the cross-layer scheme proposed in this work when both nodes and links are inhibited. The network setting used is: $P_t = -2$ dBm, $\rho = 0.5$ and $\alpha = 0.002$.

rest of the schemes. In this way, both the convergence time and the power consumption are always compared for the same level of accuracy. For more implementation details about MAC protocols check [28].

B. Numerical Results

Fig. 5 shows the bias, the variance and the MSE for the cross-layer methodology proposed in this work. In order to illustrate the importance of obtaining a symmetric matrix **P**, we have compared two different settings that lead to different values of the MSE. In the first one, we have used the traditional approach of inhibiting nodes (under the inhibition radius R_{inh}), leading to an asymmetric matrix **P**. In the second one, we have used, the results of **Theorem 2**, inhibiting links and ensuring a symmetric matrix **P**. When the matrix **P** is ensured to be symmetric, the asymmetric gossip converges in expectation, namely, the term of the bias tends to zero. Accordingly, the MSE converge to the value of the total variance. Notice that this variance depends on two factors, such as the variance of the initial data and the covariance of vector $\mathbf{m}$. Oppositely, when the matrix **P** is asymmetric, the bias does not vanishes and it dominates the value of the MSE, since it present larger values than the variance. Note that as smaller the variance is, the more importance an unbiased estimation takes and viceversa.

Fig. 6 shows that the convergence speed and the final MSE value are slightly influenced by ρ and P_t. This is explained by the fact that our link scheduling protocol produces denser connectivity patterns when the average length of links is decreased by a reduction on the values of P_t or ρ and vice versa. It means that, in both cases, the information is mixed at similar rates, namely, when denser patterns are generated, there are more data exchanges between nodes, but these exchanges are more locally done than in the case of having less larger links. As a consequence, the consensus performance in terms of convergence speed and accuracy is almost independent of the parameters ρ and P_t. Note that for a given value of these parameters P_t and ρ, as larger the resulting value of η is, the faster convergence we obtain. Therefore, for a fixed average link size, the value of η is a measure about how good the link scheduling is for the convergence rate of the consensus process. This figure also shows the influence of α in the existing trade-off between the convergence rate and the accuracy of the consensus process. It is clear from Fig. 6 that larger values of α lead to faster convergence speed but also to larger values of the MSE, since the covariance of vector $\mathbf{m}$ is significantly increased.

Fig. 7 (a) shows a comparison between our cross-layer scheme and the gossip algorithm proposed in [14], which is executed over the IEEE 802.15.4. Each point of the depicted curves correspond to a different CW value. The influence of this parameter on the convergence time and the power consumption is as follows. When a large value of CW is used, there is no concurrence in the channel and a low throughput and a high PRR are obtained. This is associated to slow convergence rates with a high energy efficiency, since most of the few

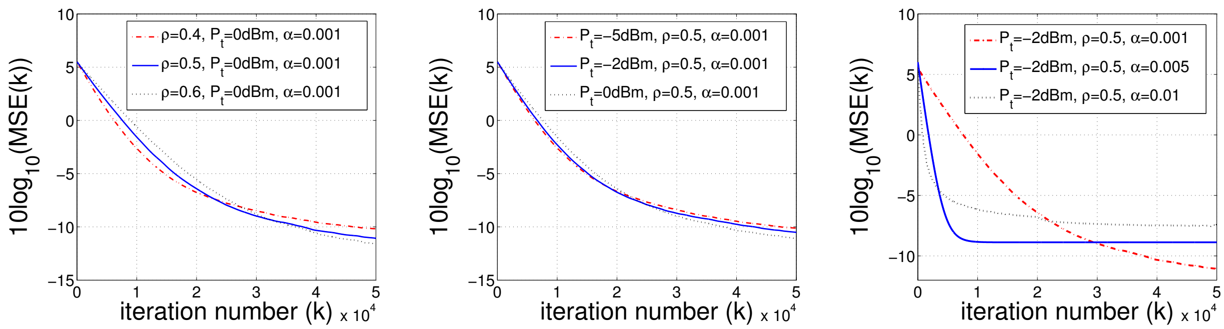
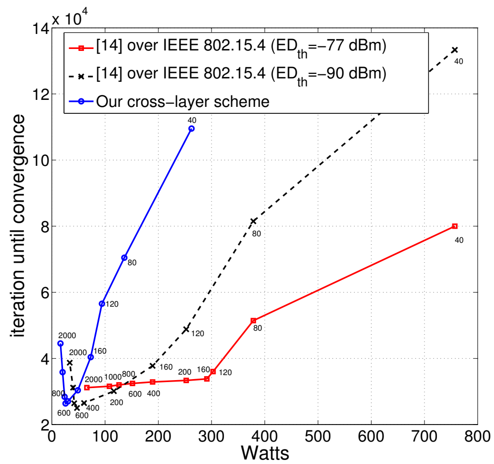
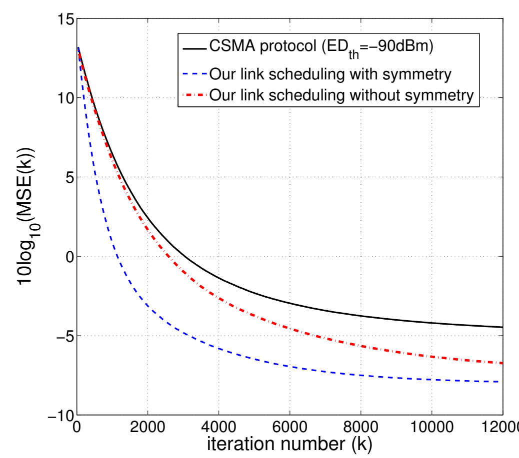


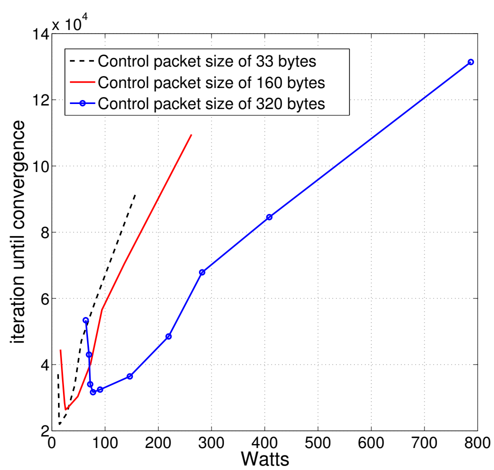
Fig. 6. Evolution of the MSE value for different combinations of the parameters: α, P_t and ρ ensuring a symmetric matrix $\mathbf{P}$.



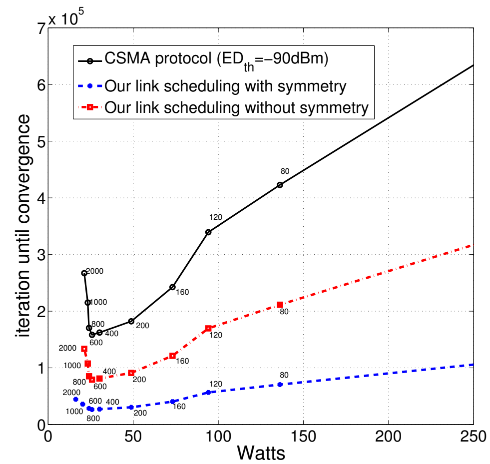
(a) comparative of different schemes



(a) MSE attained by different MAC protocols



(b) comparative of different control packet sizes



(b) time and power for different MAC protocols

Fig. 7. (a) Comparative between our cross-layer scheme and the gossip algorithm proposed in [14] executed over the IEEE 802.15.4 in terms of time and power. Two energy detection (ED) thresholds are evaluated. (b) Several control packet sizes have been tested for our cross-layer scheme.

Fig. 8. (a) Evolution of the MSE value for different medium access control protocols. (b) Comparative in terms of time and power between our link scheduling with symmetry (cross-layer scheme) and without symmetry it and the IEEE 802.15.4. All protocols execute the linear update of the state of the nodes in (7).

packets transmitted are correctly delivered. When CW starts decreasing, the throughput increases up to a maximum value from which it gets worse due to the occurrence of an increasing number of collisions. The behavior of the convergence time is the opposite of the one of the throughput. Finally, the power consumption always increases with the concurrence of the channel (smaller values of CW), since more collisions appear.

In general, our cross-layer scheme presents similar convergence times than [14] over the IEEE 802.15.4, while having significantly smaller energy consumption. This is explained by the fact that the control phase generally uses a small packets size, e.g., 32 bytes of header plus 1 byte of payload for ContikiOS. A comparison for different values of this parameters is presented in Fig. 7 (b). A small control packet size not only reduces the energy required for transmission, but also allows to schedule more links in one iteration, reducing also the convergence time.

Fig. 8 (a) shows the influence of the medium access control protocol in the the convergence rate and the accuracy of the consensus process. We execute two versions of our protocol, one ensuring collision avoidance and symmetric probabilities of communication and another ensuring only collision avoidance. We also compare these two approaches with a constant threshold technique, which is widely used in the hardware of real motes, where a node is allowed to transmit as long as the measured signal strength RSSI ≤ -90 dBm ($\mathrm{ED_{th}} = 90$ dBm). It is clear that ensuring an unbiased consensus value tends to reduce the MSE. It is important to remark that converging to a smaller final MSE value is associated to significantly smaller convergence time and power consumption, since the same level of accuracy can be attained in much less iterations, that is, less packet exchanges, see Fig. 8 (b).

VI. Conclusions

In this paper, we have considered the problem of average consensus in Wireless Sensor Networks under an accurate interference model, in which correct packet reception at a receiving node depends on the SINR. We propose a distributed and implementable cross-layer scheme, where the performance of the signal processing applications is improved by an appropriate design of the link layer. We go through the different steps of the end-to-end workflow (from application, consensus, topology, medium access control, to link scheduling). We also show dependences at each level, explaining how to tune each layer to influence the following one in an appropriate way. Moreover, we numerically evaluate the convergence rate, the power consumption and the MSE for different gossip schemes.

References

[1] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. "Randomized gossip algorithms." *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2508-2530, June 2006.

[2] N. Loizou and P. Richtárik. "A new perspective on randomized gossip algorithms." In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 440-444, December 2016.

[3] D. Alonso-Román and B. Beferull-Lozano, "Adaptive Consensus-Based Distributed Kalman Filter for WSNs with Random Link Failures." In *International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 187-192, 2016.

[4] D. Alonso-Román and B. Beferull-Lozano, "Adaptive consensus-based distributed detection in WSN with unreliable links." In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4438-4442, 2016.

[5] S. V. Macua, C. M. Leon, J. S. Romero, S. S. Pereira, J. Zazo, A. Pages-Zamora and S. Zazo. "How to implement doubly-stochastic matrices for consensus-based distributed algorithms." In *Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pp. 333-336, June 2014.

[6] 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, May 2008.

[7] T. Aysal, M. Yildiz, A. Sarwate, and A. Scaglione. "Broadcast gossip algorithms for consensus." *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 274-2761, July 2009.

[8] A. Tahbaz-Salehi and A. Jadbabaie, "Consensus Over Ergodic Stationary Graph Processes." in *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 225-230, January 2010.

[9] S. Pereira and A. Pages-Zamora. "Mean square convergence of consensus algorithms in random WSNs." *IEEE Transactions on Signal Processing*, vol. 58, no. 5, pp. 2866-2874, May 2010.

[10] Y. Chen, R. Tron, A. Terzis and R. Vidal. "Corrective consensus: Converging to the exact average." In *IEEE Conference on Decision and Control (CDC)*, pp. 1221-1228, December 2010.

[11] C. Asensio-Marco and B. Beferull-Lozano. "Fast average gossiping under asymmetric links in WSNs." In *European Signal Processing Conference (EUSIPCO)*, pp. 131-135, September 2014.

[12] K. Cai and H. Ishii. "Average consensus on general strongly connected digraphs." *Automatica*, vol. 48, no. 11, pp. 2750-2761, 2012.

[13] S. Wu and M. G. Rabbat. "Broadcast gossip algorithms for consensus on strongly connected digraphs." *IEEE Transactions on Signal Processing*, vol. 61, no. 16, pp. 3959-3971, 2013.

[14] F. Bénézit, V. Blondel, P. Thiran, J. Tsitsiklis and M. Vetterli, "Weighted gossip: Distributed averaging using non-doubly stochastic matrices." *In IEEE International Symposium on Information Theory (ISIT)*, pp. 1753-1757, June 2010.

[15] F. Iutzeler, P. Ciblat, W. Hachem and J. Jakubowicz. "New broadcast based distributed averaging algorithm over wireless sensor networks." In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3117-3120, March 2012.

[16] D. Alonso-Roman, E. Celada-Funes, C. Asensio-Marco and B. Beferull-Lozano. "Improving reliability and efficiency of communications in WSNs under high traffic demand." In *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 268-273, April 2013.

[17] E. Celada-Funes, D. Alonso-Román, C. Asensio-Marco and B. Beferull-Lozano. "A reliable CSMA protocol for high performance broadcast communications in a WSN." In *IEEE Global Communications Conference (GLOBECOM)*, pp.473-479, 2014.

[18] J. Ni, B. Tan and R. Srikant, "Q-CSMA: Queue-Length-Based CSMA/CA Algorithms for Achieving Maximum Throughput and Low Delay in Wireless Networks." *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 825-836, June 2012.

[19] B. Hajek and G. Sasaki. "Link scheduling in polynomial time." *IEEE Transactions on Information Theory*, vol. 34, no. 5, pp. 910-917, September 1988.

[20] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer. "Complexity in geometric SINR." *ACM International Symposium on Mobile Ad hoc Networking and Computing (MOBIHOC)*, pp. 100-109, 2007.

[21] S.S. Stankovic, M.S. Stankovic and D.M. Stipanovic. "Decentralized Parameter Estimation by Consensus Based Stochastic Approximation." *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp.531-543, March 2011.

[22] S.S. Stankovic, N. Ilic, M.S. Stankovic and K.H. Johansson, K.H. "Distributed Change Detection Based on a Consensus Algorithm." *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 5686-5697, December 2011.

[23] C. Asensio-Marco and B. Beferull-Lozano. "Asymmetric average consensus under SINR-based interference." In *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pp. 2571-2575, August 2012.

[24] W. Shi, Q. Ling, K. Yuan, G. Wu and W. Yin. "On the Linear Convergence of the ADMM in Decentralized Consensus Optimization." *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1750-1761, 2014.

[25] D. P. Bertsekas and J.N. Tsitsiklis. "Parallel and distributed computation: numerical methods." *Englewood Cliffs, NJ: Prentice hall*, 1989.

[26] P. Gupta and P.R. Kumar. "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388-404, March 2000.

[27] G. W. Stewart. "Matrix perturbation theory." 1990.

[28] C. Asensio-Marco and B. Beferull-Lozano. "Adaptive Medium Access Control for Distributed Processing in Wireless Sensor Networks." *IEEE Transactions on Signal and Information Processing over Networks*, vol. , no. , pp., August 2018.