Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC), Vol. 18, Iss. 1, pp. 15-34, 2020

Distributed Adaptive Filtering on Wireless Sensor Networks with Shared Medium Competition

Rafael Moura do Carmo
CEFET-RJ
rafaelm.carmo@hotmail.com
Luís Tarrataca, Jefferson Colares, Felipe da R. Henriques, Diego B. Haddad
CEFET-RJ
tarrataca@gmail.com, jcolares@gmail.com, felipe.henriques@cefet-rj.br, diego.haddad@cefet-rj.br
Raphael M. Guedes
LAND/COPPE/UFRJ
raphaelmguedes@gmail.com

Abstract – Wireless Sensor Networks (WSN) are of significant importance with increasingly diverse and viable applications. They gained even more traction after the IEEE 802.15.4 standard was defined. Distributed adaptive filtering algorithms have added statistical inference to WSN applications, employing techniques that extract data from distributed devices. In contrast, most adaptive filtering contributions do not consider realistic features of the subjacent telecommunications network protocols. Similarly, the telecommunications area typically does not take into account interesting abilities of adaptive filtering algorithms. In this paper, we explore this gap between the two study areas, allowing the development of network-protocol-aware distributed adaptive filtering techniques. In order to explore network realistic behaviors, this paper focuses on distributed inference problems. More specifically, we propose two new diffuse adaptive algorithms, aware of the characteristics of the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol, namely: (i) Variant Reuse of Coefficients Least Mean Squares (VRC-LMS) algorithm; and the (ii) Reuse of Coefficients Least Mean Squares (RC-LMS) algorithm in the Adapt-Then-Combine (ATC) modality. These two new algorithms will bring some advantages, specifically when information is delayed because of too much packet loss. Another advantage will be the addition of the spatial information diversity contribution in the VRC-LMS algorithm.

Keywords – Distributed adaptive filtering, Spatial diversity, Wireless sensor networks, Telecommunication protocols.

1. Introduction

Wireless Sensor Networks (WSN) composed of intelligent sensors (possibly distributed over vast geographic regions [1]) have presented key contributions to the area of *big data* [2]. The combination of the huge growth in data volume with the advent of more efficient communication technologies promotes powerful schemes that address *big data*-related challenges [1]. This paper focuses on: (i) distributed inference problems that generaly occur on such types of networks [3]; and (ii) the design of distributed adaptive filtering algorithms incorporating realistic features of communication protocols. Adaptive filtering algorithms were chosen because of their low computational complexity, robustness to dynamic configurations, and ability to handle noisy information [4]. Typically, researchers of signal processing area do not address communication techniques from the perspective of distributed inference problems [5]. Similarly, the theoretical models used in signal processing to predict the behavior of distributed algorithms ignore the characteristics of protocols normally implemented in real devices. This paper intends to bridge the gap between adaptive filters and WSN. Accordingly, we associate metrics normally employed in the area of adaptive filtering (*e.g.:* the mean deviation error) with measurements typically used in the area of sensor networks (*e.g.:* network lifetime), through simulations that evaluate the impact of shared access protocols. We chose to employ state-of-the-art simulations, implemented in Network Simulator 3 (NS-3) [6].

In addition, we also take into account in the design of adaptive filtering algorithms, realistic communication protocols, such as the CSMA/CA protocol commonly used in WSN [7]. More specifically, we propose the utilization of the coefficient reuse technique [8]. This allows for better performance in steady state. As older information is less reliable, we will adopt a coefficients reutilization aware of the current network characteristics. Such awareness will allow for a more intense reuse of recent information alongside a greater attenuation of the impact of older information, the latter of which may be the result of collisions and packet retransmission. Such a design of a distributed inference method, contemplating the specificities of communication modes in a shared environmnent, is an unexplored area, located at the intersection between sensor networks, signal processing and distributed optimization.

This paper is structured as follows. Section 2 depicts the fundamentals of adaptive filtering algorithms. Section 3 will describe one type of cooperation strategy in WSN which is referenced as the one with better performance among the main

techniques. Section 4 presents the concepts of coefficient reuse, which may benefit the estimation process under adverse conditions. Section 5 makes an overview of the IEEE 802.15.4 standard, highlighting its main characteristics such as the ones related to measuring parameters such as packet loss or power consumption. These features are then used as motivation for the proposed communication-protocol-aware adaptive filtering algorithm derived in Section 6.1. Section 7 shows the obtained results. Section 8 presents the concluding remarks of the paper.

2 Adaptive Filtering Fundamentals

The design of digital filters from a well-defined specification is a mature area of digital signal processing [9]. In practice, however, there are situations where specifications are not available and/or may vary over time. Usually, adaptive coefficients are used in these cases, resulting in the field known as adaptive filtering [10–12].

It is up to an adaptive filtering algorithm to update the filter coefficients. This is done in order to progressively perform the desired information extraction operation. Such updating is commonly guided by a cost function which, in supervised contexts, is dependent on a reference signal $d(k)$ [13]. The structure of an adaptive filter, in the context of system identification, is presented in Figure 1. Although the scope of this work focuses in the identification of systems, it should be emphasized that adaptive filtering techniques have a wide range of applications, such as adaptive control, echo cancellation [14], noise cancellation [15], channel equalization [16], beam conformation [17] and time series prediction [12].

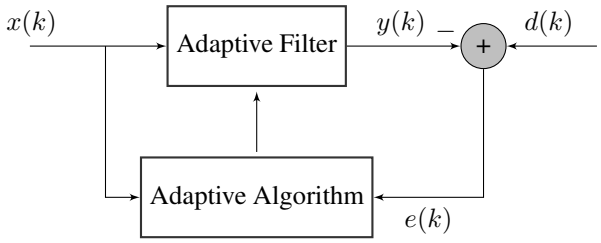

Figure 1: Classic structure of an adaptive filter.

In Figure 1, $x(k)$ represents the input signal (which must also excite the plant to be identified) and $y(k)$ consists of the filter output. This output must progressively emulate $d(k)$, called the desired signal. The magnitude of the discrepancy between the reference signal and the output of the filter is correlated with the quality of the filter at a given iteration. Such a discrepancy is usually incorporated into an error signal $e(k)$ defined as

$$e(k) \triangleq d(k) - y(k), \tag{1}$$

where

$$y(k) \triangleq \boldsymbol{w}_i^T(k)\boldsymbol{x}(k). \tag{2}$$

where $N \in \mathbb{N}$ is the filter size, the vectors $\boldsymbol{x}(k)$ and $\boldsymbol{w}_i(k)$ are as follows

$$\boldsymbol{x}(k) \triangleq \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k-N+1) \end{bmatrix}, \tag{3}$$

and

$$\boldsymbol{w}_i(k) \triangleq \begin{bmatrix} w_0(k) \\ w_1(k) \\ \vdots \\ w_{N-1}(k) \end{bmatrix}, \tag{4}$$

where $w_i(k)$ is the i-th adaptive coefficient at the k-th iteration. It should be noted that identity (2) assumes a tapped-delay structure, which is often the one adopted in adaptive filtering algorithms. An adaptive filter undertakes changes in its coefficients oriented by minimizing a stochastic cost function (or, equivalently, an instantaneous cost function). Among the cost functions frequently found in the literature, the most common is the *instant quadratic error*, which is a stochastic approximation of the *mean square error* [10, 11, 13, 18]. When this cost function is used, we have as result the Least Mean Squares (LMS) algorithm. This algorithm can be developed, therefore, from a simplification of the cost function, defined as

$$J(k) = \mathbb{E}\left[d(k) - \boldsymbol{w}^T(k)\boldsymbol{x}(k)\right]^2, \tag{5}$$

where $\mathbb{E}[\cdot]$ consists of the operator of statistical hope.

In the ideal case (*i.e.*, when we use the cost function of Equation (5)), the gradient vector of the cost function can be described as

$$\frac{\partial J(k)}{\partial \boldsymbol{w}(k)} = \boldsymbol{R}_x \boldsymbol{w}(k) - \boldsymbol{b}_x, \tag{6}$$

where $\boldsymbol{R}_x$ corresponds to the correlation matrix of the input signal

$$\boldsymbol{R}_x \triangleq \mathbb{E}\left[\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right] \tag{7}$$

and $\boldsymbol{b}_x$ is the autocorrelation vector between the desired signal and the input signal [19].

Equation (6) expresses the solution to the defined cost function in (5) as

$$\boldsymbol{w}^\star = \boldsymbol{R}_x^{-1}\boldsymbol{b}_x, \tag{8}$$

where $\boldsymbol{w}^*$ is a vector of dimensions $N \times 1$. This solution is known in the literature as *Wiener's solution*, which consists of an analytical formula that determines the coefficients responsible for minimizing the cost function (5), or that in other words, represents the solution for the minimization of the mean square error [11].

As the statistics $\boldsymbol{R}_x$ and $\boldsymbol{b}_x$ are usually unknown, it is necessary to stochastically approximate them by means of:

$$\boldsymbol{R}_{\boldsymbol{x}} \approx \boldsymbol{x}(k)\boldsymbol{x}^T(k), \tag{9}$$

$$\boldsymbol{b}_{\boldsymbol{x}} \approx d(k)\boldsymbol{x}(k). \tag{10}$$

Substituting equations (9) and (10) in (6), one obtains

$$\frac{\partial J(k)}{\partial \boldsymbol{w}(k)} = -d(k)\boldsymbol{x}(k) + \boldsymbol{x}(k)\boldsymbol{x}^T(k)\boldsymbol{w}(k). \tag{11}$$

In this way, by means of the stochastic gradient technique, one obtains the following update equation:

$$\begin{aligned} \boldsymbol{w}(k+1) &= \boldsymbol{w}(k) - \beta\frac{\partial J(k)}{\partial \boldsymbol{w}(k)} \\ &= \boldsymbol{w}(k) + \beta\boldsymbol{x}(k)\overbrace{[d(k) - \boldsymbol{x}^T(k)\boldsymbol{w}(k)]}^{=e(k)}, \end{aligned} \tag{12}$$

which is described in Algorithm 1.

Algorithm 1 LMS

```
procedure LMS ALGORITHM UPDATE
    w(0) = [0 0 ... 0]^T                ▷ Initialization
    for k ≥ 0 do
        e(k) = d(k) − x^T(k)w(k)
        w(k+1) = w(k) + βe(k)x(k)
    end for
end procedure
```

In the context of distributed adaptive filtering, it can be described as a degenerate case in which the update of the adaptive algorithm occurs in a single node. If necessary, the size of the learning factor β can be selected to vary with time, so that the new sequences (in the form $\beta(k)$) obey the following rules simultaneously[1] [10, 18]:

$$\sum_{k=0}^{\infty}\beta(k) = \infty, \qquad \sum_{k=0}^{\infty}\beta^2(k) < \infty. \tag{13}$$

However, these sequences slowly converge to zero, in addition to decreasing when $k \to \infty$, interrupting the adaptation process [18]. For this reason, we will adopt in this work a constant learning factor β. In order to deepen the performance analysis of the LMS algorithm in cooperative networks, it is necessary and relevant to perform an analysis of the Mean Square Deviation (MSD) metric. In Expression (12), $\boldsymbol{w}(k)$ approaches $\boldsymbol{w}^\star$ as time elapses. Formally speaking, we can not assert that $\boldsymbol{w}(k)$ tends to $\boldsymbol{w}^\star$, because *convergence* is impossible due to the ubiquitous presence of additive noise (or measuring) $\nu(k)$ [20].

It should be noted that the LMS algorithm can also be derived by means of a methodology complementary to the stochastic gradient strategy. Such a methodology employs a *deterministic* optimization problem whose solution is *exact* (without resorting to approximations). Thus, we have the following linear constraint optimization problem:

$$\min_{\boldsymbol{w}(k+1)} \mathcal{F}\left[\boldsymbol{w}(k+1)\right] \triangleq \|\boldsymbol{w}(k+1) - \boldsymbol{w}(k)\|^2 \tag{14}$$

[1] Such rules are only interesting for the case where the optimal filter is time invariant.

$$\text{s.t. } e_{\text{p}}(k) = (1 - \beta\|\boldsymbol{x}(k)\|^2)e(k),$$

where $e_{\text{p}}(k)$ is known as *posteriori* error, being defined by

$$e_{\text{p}}(k) \triangleq d(k) - \boldsymbol{w}^T(k+1)\boldsymbol{x}(k), \tag{15}$$

which evaluates the error of the linear prediction performed on the data pair $\{d(k), \boldsymbol{x}(k)\}$ *after* the update generated by this same pair of data [9].

The cost function $\mathcal{F}\left[\boldsymbol{w}(k+1)\right]$ in (14) penalizes solutions $\boldsymbol{w}(k+1)$ too far from the current vector $\boldsymbol{w}(k)$. Such a bias is important because $\boldsymbol{w}(k)$ incorporates information from many previous data pairs, and therefore it is unlikely that only a pair of data $\{d(k), \boldsymbol{x}(k)\}$ will be able to provide a solution that is very divergent from the current one. This conservative principle, however, is not enough to specify the new vector of adaptive coefficients $\boldsymbol{w}(k+1)$. This requires the specification of the linear constraint (14), whose locus is a hyperplane to which $\boldsymbol{w}(k+1)$ must belong (note that the *posteriori* error is a linear function of $\boldsymbol{w}(k+1)$). Geometrically, the intersection of this hyperplane with a sphere specifies the solution $\boldsymbol{w}(k+1)$. As a consequence of this result, the sphere becomes the one with the smallest radius (due to the principle of minimum distortion implicit in the definition of $\mathcal{F}\left[\boldsymbol{w}(k+1)\right]$).

The use of the Lagrange multiplier technique allows us to rewrite (14) equivalently as

$$\min_{\boldsymbol{w}(k+1)} \|\boldsymbol{w}(k+1) - \boldsymbol{w}(k)\|^2 + \lambda\left[e_{\text{p}}(k) - (1 - \beta\|\boldsymbol{x}(k)\|^2)e(k)\right], \tag{16}$$

which is an unrestricted problem whose solution coincides with that of (14). For the minimization of (16) to take effect, it is necessary to set its gradient with respect to $\boldsymbol{w}(k+1)$ to zero, which, after some manipulations, gives us:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \frac{\lambda}{2}\boldsymbol{x}(k). \tag{17}$$

In order to finally arrive at the LMS update equation - see Eq. (12) - it is necessary to prove that $\frac{\lambda}{2} = \beta e(k)$, which can be easily proven by applying (17) to the linear constraint of (14). For simplicity, we omit the details.

The energy of the deviation vector

$$\tilde{\boldsymbol{w}}(k) \triangleq \boldsymbol{w}^\star(k) - \boldsymbol{w}(k) \tag{18}$$

is measured by the MSD metric, and the definition of which is given by

$$\text{MSD}(k) \triangleq \mathbb{E}\left[\left\|\tilde{\boldsymbol{w}}(k)\right\|^2\right]. \tag{19}$$

The MSD is an important performance metric, as it measures the size of the deviation variation $\mathbb{E}\left[\left\|\tilde{\boldsymbol{w}}(k)\right\|^2\right]$ over the iterations [21]. It is widely known in the literature that small values of the β learning factor engender better performance (*i.e.*, a lower MSD) in the steady state [10, 18], and consequently a greater probability of finding small values (in magnitude) of the error $e(k)$. In this case, assuming β is small and some common hypotheses in the literature [12], we can conclude that

$$\text{MSD}_{\text{ncoop}} \approx \frac{\beta N}{2}\sigma_\nu^2, \tag{20}$$

where σ_ν^2 is the additive noise variance. It is worth noting that (20) corresponds to the non-cooperative mode of operation; *i.e.*, the metric of equation (20) denotes the MSD value individually measured on a node that autonomously implements the adaptive algorithm of [18].

The convergence rate r of $\mathbb{E}\left[\left\|\tilde{\boldsymbol{w}}(k)\right\|^2\right]$ towards the permanent regime is often measured by [10, 18, 21]:

$$r \approx 1 - 2\beta\lambda_{\text{min}}(\boldsymbol{R}_x). \tag{21}$$

As can be seen, (21) is explicitly dependent on the lower eigenvalue λ_{min} of $\boldsymbol{R}_x$; the lower the value of $r \in (0, 1)$, more faster will be the convergence.

Taking the mean over all nodes (and assuming that the measurement noise variance is the same for all nodes), we can rewrite equation (20) as

$$\text{MSD}^{\text{network}} \approx \frac{\beta N}{2}\left(\frac{1}{M}\sum_{i=1}^{M}\sigma_\nu^2\right). \tag{22}$$

Note that (22) corresponds to the performance of the nodes in a non-cooperative network topology [18]. Equation (22) allows one to observe that the average MSD is proportional to noise-node variances. However, as network nodes observe the growth of data transmission under the same model $\boldsymbol{w}^\star$, it is natural that cooperation between nodes is beneficial to the network with respect to data metrics or estimators [18]. Section 3 emphasizes that, in the context of diffuse adaptation, Eq. (12) is fundamental for elaborating alternative strategies [10].

3 Diffuse Adaptive Filtering

Among the modalities of cooperation and exchange of information, the diffuse mode has gained prominence in relation to the incremental and consensus strategy, since it presents superior performance in terms of traceability, convergence rate and stability [10, 18, 19]. Distributed strategies require the modeling of features associated with the processing, aggregation, and diffusion of information through graphs that are used to generate local interactions between neighboring nodes and allow greater resilience to failures [10]. They have attracted great interest because of their ability to provide organizational models capable of achieving high levels of performance through the use of local adaptive dynamics [22]. The effectiveness of any distributed model will depend on the mode of operation between the nodes [23]. As a result, we will start by describing the network (Section 3.1) and data models, and then proceed to the ATC model of cooperation in networks (Section 3.2).

3.1 Network Model

Consider a network composed by M nodes, endowed with computational capabilities. The interconnections are modeled by means of edges and vertices (nodes). The edges represents the bidirectional exchange of information between nodes [18]. The neighborhood of a node is defined as $\mathcal{N}_i$, that represents the set of nodes connected to the i-th node, including itself. An undirected graph is assumed, so that if the i-th node is a neighbor of the j-th one, then the j-th node is also a neighbor of the i-th one [18]. A node may have several neighbors connected by means of edges, as described in Figure 2.

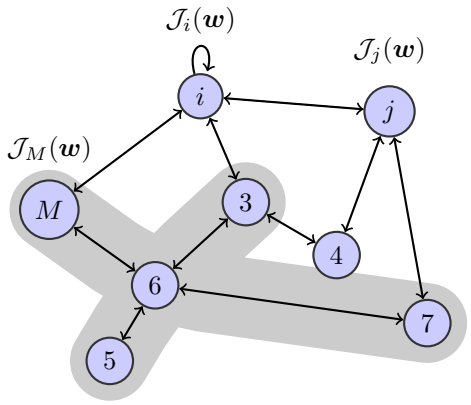

Figure 2: $\mathcal{N}_6$ is the set of nodes which includes $\{3,5,6,7,M\}$.

The purpose of the network is to estimate in real time, and in a distributed manner, the minimizer of the cost function [18]

$$\min_{\boldsymbol{w}} \sum_{i=1}^{M} J_i\left(\boldsymbol{w}\right), \tag{23}$$

of dimensions $N \times 1$, where these individual costs can be distinct across the nodes or they can be identical [18]. We are interested in a real-time distributed estimation of the optimized parameter vector $\boldsymbol{w}^{\star}$, which relies on local interactions between nodes.

3.2 Data Model

Assume that the i-th node has access to a moving-average model [11]:

$$d_i(k) = \sum_{n=0}^{N-1} w^{\star}(n) x_i(k-n) + \nu_i(k) \tag{24}$$

$$= \left(\boldsymbol{w}^{\star}\right)^T \boldsymbol{x}_i(k) + \nu_i(k), \tag{25}$$

where $\nu_i(k)$ represents the noise sample (usually linked to measurement and/or modeling errors) associated with the i-th node at the k-th iteration and

$$\boldsymbol{x}_i(k) \triangleq \begin{bmatrix} x_i(k) & x_i(k-1) & \ldots & x_i(k-N+1) \end{bmatrix} \tag{26}$$

contains N consecutive samples of the i-th node input signal.

3.3 ATC Strategy

The ATC technique uses a data aggregation step that allows the incorporation of information collected from local neighbors into the adaptation mechanism, described mathematically as

$$\boldsymbol{w}_i(k+1) = \sum_{j \in \mathcal{N}_i} c_{ij} \boldsymbol{\psi}_j(k), \tag{27}$$

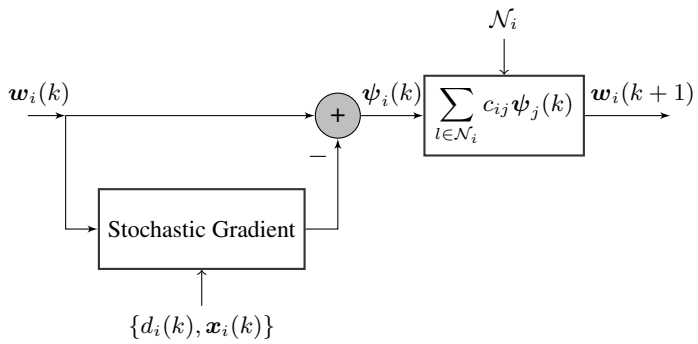

Figure 3: Block representation of the i-th node update in the ATC diffusion strategy.

where c_{ij} is the combination of the coefficients. Index j indicates one neighbor node attached to the i-th node and $\mathcal{N}_i$ indicates the set of neighboring nodes of the i-th sensor. Figure 3 depicts the block diagram of the ATC strategy.

In cooperative diffusion mode, nodes exchange their local estimates with neighbors and incorporate the estimates collected through linear combinations. These are commonly called in the literature as Constant Combined Weights [24, 25]. There are some common choices for selecting such weights in a network composed of M nodes. Many of these techniques are originally derived from graph theory [25].

Among the techniques that can be used, the *Metropolis* rule [19, 25, 26] takes the form presented in Expression (28).

$$c_{ij} = \begin{cases} \frac{1}{\max|\mathcal{N}_i|,|\mathcal{N}_j|}, & \text{if } \ i \neq j \text{ are neighbors} \\ 1 - \sum\limits_{j\in\frac{\mathcal{N}_i}{n}} c_{ij}, & \text{if } \ i = j \\ 0, & \text{otherwise} \end{cases} \tag{28}$$

where $\max|\mathcal{N}_i|, |\mathcal{N}_j|$ indicates the maximum cardinality of the neighborhood of i and j. The Laplacian rule [19, 27] is given by equation

$$\boldsymbol{C} = I_N - k\mathcal{L}, \tag{29}$$

and $\boldsymbol{C}$ is the combination matrix of the dimension coefficients $[a_{kl}]$, $\mathcal{L} = \mathcal{D} - A_d$, with $\mathcal{D} =$ diag $(|\mathcal{N}_1|, |\mathcal{N}_2|, ..., |\mathcal{N}_N|)$, $\alpha = \frac{1}{|\mathcal{N}_{max}|}$ and A_d is the adjacent network matrix formed by

$$[A_d]_{ij} = \begin{cases} 1, \text{ if } i \text{ and } j \text{ are neighbors} \\ 0, \text{ otherwise.} \end{cases} \tag{30}$$

In the case of the nearest neighbor rule [19, 28], the combination coefficient c_{ij} is denoted as follows

$$c_{ij} = \begin{cases} \frac{1}{\mathcal{N}_i}, \text{ if } i \text{ and } j \text{ are neighbors} \\ 0, \text{ otherwise.} \end{cases} \tag{31}$$

The combination coefficient c_{ij} must satisfy

$$\sum_{j\in\mathcal{N}_i \forall i} c_{ij} = 1. \tag{32}$$

The term diffusion is employed because the intermediate state $\boldsymbol{\psi}_i(k)$ allows data to spread across the network, bringing the i-th node information beyond its neighborhood [18]. Algorithm 2 synthesizes the ATC-based classical LMS algorithm.

Algorithm 2 ATC Strategy

1: **procedure** LMS ALGORITHM UPDATE
2: **for** For each k instant, each node i **do**
3: $\boldsymbol{\psi}_i(k) = \boldsymbol{w}_i(k) + \beta\boldsymbol{x}_i(k)[d_i(k) - \boldsymbol{x}_i^T(k)\boldsymbol{w}_i(k)]$. ▷ Adapt
4: $\boldsymbol{w}_i(k+1) = \sum\limits_{j\in\mathcal{N}_i} c_{ij}\boldsymbol{\psi}_j(k)$ ▷ Combine
5: **end for**
6: **end procedure**

4 Reuse Coefficients Technique

One of the main obstacles to the estimation process of an adaptive filtering algorithm is the measurement noise. The fact that it is not zero implies that its variance increases the variance of the adaptive estimator, and hence its mean square deviation. Particularly, in situations where signal-to-noise ratio (SNR) is low, the identification process may be severely impaired. One way to mitigate this problem is to use the coefficient reuse technique (RC) [29], which changes the principle of minimum distortion, taking into account the minimization of a weighted sum of the distances of the new solution vector to the last L solution vectors.

Typically, algorithms with coefficients reuse adopt a normalized update term, giving rise to the RC-NLMS algorithm and its variants, which can present a reuse factor and/or a time-varying learning factor [30, 31]. More recently, reference [32] detailed how to obtain a non-normalized algorithm that employs the RC technique.

In this section, we will extend the RC technique to the domain of diffuse adaptive filtering. This extension will be useful when measuring noise (which, we may recall, can incorporate modeling errors) has a high variance, resulting in a better steady-state performance.

Unlike the traditional LMS algorithm, which minimizes the MSE cost function, the NLMS algorithm minimizes the following cost function [32]

$$J(k) = \mathbb{E}\left[\frac{e^2(k)}{\left\|\boldsymbol{x}(k)\right\|^2}\right], \tag{33}$$

whose minimization result can possibly differ[2] from the MSE $\xi(k) \triangleq \mathbb{E}\left[e^2(k)\right]$, where $e(k)$, is defined in Section 2 as

$$e(k) \triangleq d(k) - y(k). \tag{34}$$

The RC-NLMS algorithm is usually derived using the Lagrange multipliers technique. Consider the following optimization problem:

$$\min_{\boldsymbol{w}(k+1)} = \sum_{l=0}^{L-1} \rho^l \|\boldsymbol{w}(k+1) - \boldsymbol{w}(k-l)\|^2 \tag{35}$$

$$\text{s.t. } e_p(k) = d(k) - \boldsymbol{w}^T(k+1)\boldsymbol{x}(k) = (1-\beta)\overline{e}(k),$$

where $\rho \in (0, 1]$ and $L \in \mathbb{N}$ are parameters at the discretion of the designer that regulate the reuse intensity, $\overline{e}(k)$ is defined by

$$\overline{e}(k) \triangleq d(k) - \theta(\rho) \sum_{l=0}^{L-1} \rho^l \boldsymbol{w}^T(k-l)\boldsymbol{x}(k), \tag{36}$$

and $\theta(\rho)$ can be expressed as

$$\theta(\rho) \triangleq \begin{cases} \left(\frac{\rho-1}{\rho^L-1}\right), & \text{for } \rho \neq 1 \\ \frac{1}{L}, & \text{for } \rho = 1 \end{cases}. \tag{37}$$

The optimization problem (35) can be converted into an equivalent one by the Lagrange multipliers technique, so that we can solve (35) by means of

$$\begin{gathered} \min_{\boldsymbol{w}(k+1)} \mathcal{F}_{\text{RC}}[\boldsymbol{w}(k+1)] = \\ \textstyle\sum_{l=0}^{L-1} \rho^l \|\boldsymbol{w}(k+1) - \boldsymbol{w}(k)\|^2 + \lambda[e_p(k) - (1-\beta)\overline{e}(k)], \end{gathered} \tag{38}$$

where λ is the Lagrange multiplier. To minimize (38), we can find the vector $\boldsymbol{w}(k+1)$ which zeroes the gradient:

$$\begin{gathered} \nabla_{\boldsymbol{w}(k+1)} \mathcal{F}_{\text{RC}}[\boldsymbol{w}(k+1)] \\ = 2\textstyle\sum_{l=0}^{L-1} \rho^l \left[\boldsymbol{w}(k+1) - \boldsymbol{w}(k)\right] - \lambda \boldsymbol{x}(k) = \boldsymbol{0}, \end{gathered} \tag{39}$$

or, equivalently:

$$\boldsymbol{w}(k+1) = \theta(\rho) \sum_{l=0}^{L-1} \rho^l \boldsymbol{w}(k-l) + \frac{\lambda\theta(\rho)}{2}\boldsymbol{x}(k). \tag{40}$$

The insertion of (40) in (35) allow us to find $\frac{\lambda\theta(\rho)}{2}$:

$$\frac{\lambda\theta(\rho)}{2} = \frac{\beta\overline{e}(k)}{\|\boldsymbol{x}(k)\|^2}, \tag{41}$$

[2]This difference tends to be magnified when the equalizer size is small, in which case the correlation between the random variables $\|\boldsymbol{x}\|^2$ and $e(k)$ tend to be larger [32].

so that the update equation of the RC-NLMS algorithm is described by

$$\boldsymbol{w}(k+1) = \theta(\rho) \sum_{l=0}^{L-1} \rho^l \boldsymbol{w}(k-l) + \frac{\beta \boldsymbol{x}(k) \overline{e}(k)}{\|\boldsymbol{x}(k)\|^2}. \tag{42}$$

The update equation (42) specifies an adaptive filtering algorithm with a normalized update. As this work focuses on non-normalized algorithms, it is necessary to design an algorithm incorporating coefficient reuse without normalization. This was done recently in [32], an article that proposed the following optimization problem:

$$\min_{\boldsymbol{w}(k+1)} = \sum_{l=0}^{L-1} \rho^l \|\boldsymbol{w}(k+1) - \boldsymbol{w}(k-l)\|^2 \tag{43}$$

$$\text{s.t } e_p(k) = d(k) - \boldsymbol{w}^T(k+1)\boldsymbol{x}(k) = (1 - \beta\|\boldsymbol{x}(k)\|^2)\overline{e}(k),$$

whose solution gives rise to the update equation of the RC-LMS algorithm[3] [32]:

$$\boldsymbol{w}(k+1) = \theta(\rho) \sum_{l=0}^{L-1} \rho^l \boldsymbol{w}(k-l) + \beta \boldsymbol{x}(k) \overline{e}(k). \tag{44}$$

Note that both update equations (42)-(44) were derived using the Lagrange multipliers technique, rather than stochastic gradient. This means that if we want to insert the coefficient reuse technique into the diffuse adaptive filtering algorithms, we must first obtain them by a similar procedure. This work, as described in the previous section, focuses on the ATC algorithm [10]. In the case of ATC, we can interpret its adaptation equation (prior to the combination) as the resolution of the following optimization problem:

$$\min_{\boldsymbol{\psi}_i(k+1)} \|\boldsymbol{\psi}_i(k+1) - \boldsymbol{w}_i(k)\|^2 \tag{45}$$

$$\text{s.t. } d_i(k) - \boldsymbol{\psi}_i^T(k+1)\boldsymbol{x}_i(k) = (1 - \beta\|\boldsymbol{x}_i(k)\|^2) \left[d_i(k) - \boldsymbol{w}_i^T(k)\boldsymbol{x}_i(k)\right],$$

whose resolution gives rise to the well-known adaptation equation:

$$\boldsymbol{\psi}_i(k+1) = \boldsymbol{w}_i(k) + \beta \boldsymbol{x}_i(k) \left[d_i(k) - \boldsymbol{w}_i^T(k)\boldsymbol{x}_i(k)\right]. \tag{46}$$

It is known that the MSD has a tendency to increase when the noise variance is large [8, 33]. This impact is usually overcome by the adoption of coefficient reuse (RC) [31]. The main objective is to propose the minimization of the weighted sum of the square Euclidean norm of the difference between the updated coefficient vector and the L vectors of previous coefficients. The minimization is subject to a linear constraint on the new solution $\boldsymbol{w}(k+1)$ [29]. Such a method has the advantage of reducing the steady-state MSD with minimal impact at the convergence rate [34]. Therefore, we chose to employ time-varying reuse factors. These allow for a low reuse factor in the transient (thus avoiding losses to the convergence rate) and an intense utilization of coefficients reuse in steady-state. Overall, this contributes to a performance improvement in this region of operation.

5 IEEE 802.15.4 Standard

It is expected that the addition of protocols traditionally associated with the telecommunications area, the estimation process will integrate more reliable characteristics in real WSN scenarios. These characteristics will be discussed in the following subsections.

5.1 Overview

The architecture of the LR-WPAN (Low-Rate Wireless Personal Area Network) standard is structured in three blocks: PHY (Physical), MAC, and Upper Layers. The interfaces between the blocks serve to define their logical connections. Each LR-WPAN device needs at least the PHY layer, which contains the radio frequency transceiver, a low-level mechanism, and the MAC sublayer that provides access to the physical channels for all types of transfers. The upper layers provide support for network configuration, message routing, and the application layer which is responsible for the specific function of the device [7, 35, 36].

5.2 Energy Structure

Typically, the sensors that make up a network are powered by batteries, whose amount of power is limited. In this way, monitoring the energy usage and the useful life of the network become important metrics in WSN. In order to support simulations involving different forms of energy, NS-3 provides some models of power consumption for network devices. The models represent the energy sources connected to the nodes, and can assume linear characteristics[4], or nonlinear ones[5] [37]. Since WSN

[3] We do not explain the derivation steps because they are very similar to those of obtaining Eq. (42).
[4] They do not generate characteristics of real batteries.
[5] They are more precise when assuming discharge curves more faithful to the real cases.

are characterized by a very low transmission cycle, some articles on microsensors denote that the greatest energy expenditure is in the transitions of the radio between one state and another [38, 39]. In this way, the energy model used in this work takes into account the transition time and the energy between the transceiver states, which are presented by a state machine of the physical layer depicted in Figure 4. This representation will serve as a computational template for the network lifetime. The metrics employed, namely, stable state power, transient time, and power measurement consider the information extracted from the transceiver device datasheet CC2420 (40), which is compatible with the IEEE 802.15.4 network standard.

In order to have access to the average node consumption, it is necessary to trace the instantaneous power consumption when transitions between transceiver states occur. Among the transitions that can occur at the physical level, we are particularly interested in the end-of-transmission and reception operations, respectively defined as `BusyTx` $\rightarrow$ `TxOn` and `BusyRx` $\rightarrow$ `RxOn` (Figure 4). Each tracked event will be recorded in an array where we will indicate line by line each operation being performed, namely: `t` (transmission), `r` (reception) and `d` (drop); the index of each node of the network n_{id}, the end instants of each transition and the final energy level e_f and previous e_i. The energy attributes are measured in Joules. This is in accordance with the model of the basic-energy-source simulator [41].

Each network event mentioned in Figure 4 is associated with an energy level, the efficiency of which is associated with the type of transceiver to be used [42]. For most transmitters, there is no difference between the `On` and `Busy` states with respect to the transceiver circuit [42]. The level of energy consumed in each state depends on the current and time spent. This information can be extracted from the datasheets of the electronic elements, such as the transceiver radio CC2420:

(i) `TxOn` (transmition): 17.4 mA, 0 dBm;

(ii) `RxOn` (reception/idle): 18.8 mA.

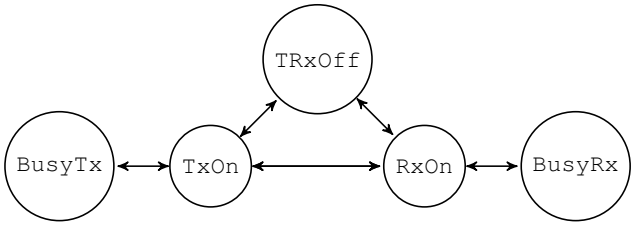

Figure 4: Physical Layer State Machine.

The lifetime of the network is calculated as soon as the power of the node assembly is completely dissipated, making more transmissions an impossible task. This method is ensured through a frame transmission replaying structure as $e_f > 0$, using the *lr-wpan-radio-energy-model*[6].

5.3 Shared Medium Protocol

The IEEE 802.15.4 standard can employ two mechanisms to increase the likelihood of success in data transmission [7], they are: ALOHA and CSMA/CA. In what concerns access to the shared medium, our emphasis will be placed on the CSMA/CA mechanism (Carrier Sense Multiple Access with Collision Avoidance). Its adoption is well established and its characteristics are well known [43–45]. Fundamentally, the CSMA/CA protocol is capable of identifying multiple carrier accesses to avoid possible collisions.

Only one node of the network is allowed to transmit information at a given instant. All nodes in wireless applications must compete through the medium using the CSMA/CA mechanism. As a result, a node that has an information to be sent should map the channel and check if it is free. If the channel is free, the information is transmitted, otherwise it should queue the request and continue mapping until it becomes free for a new request.

This work considers the realistic network behavior of the CSMA/CA, and evaluates the impact of the shared medium dispute on the proposed distributed adaptive filter.

6 Spatial Diversity of Information

The aggregation of data in a network is fundamentally important for solving distributed inference problems in several fields of science [18, 46]. In sensor network contexts, where the energy cost of communication is often greater than the energy cost of computing, and energy resources are limited, methods such as directed diffusion can achieve significant energy savings through the use of data aggregation identified by attribute/value pairs [47]. Thus, only nodes that have data that match the criteria of interest can forward the information to the nodes [47]. These benefits have been theoretically [48] and experimental [49] proven.

In the context of social networks, for example, it is said that social relations between individuals can affect the economic development of the community [46]. Although the correspondence between network diversity and social and economic well-being has not yet been quantified in general, recent studies [46] have shown the existence of a relationship between the composite

[6]This measurement structure is available at code.nsnam.org/vrege/ns-3-gsoc.

measure of diversity and the socio-economic percentile. Namely, it has been shown that social networking diversity is a strong signature of economic development .

In the area of network security, privacy is often guaranteed by encrypting the source data before propagating the signal to the environment [50]. This area of science is called the security encryption methodology and is usually implemented at the application layer of the network. However, these methods may experience cryptographic attacks because unauthorized nodes are under the same coverage area. In this sense, many studies have been intensifying network security studies in the physical transport layer, by allocating intermediate nodes distant from a possible network invader but still near the signal source [50, 51]. In this way, many intermediate nodes can safely decode the source signal, and then form a beam that hides the wanted signal from a possible intruder. This contributes to maximizing the secrecy of the information in the destination [50, 52].

As mentioned in previous paragraphs, the spatial diversity of information has been the object of study in several areas, bringing various contributions to the scientific community. In the following sections, we will formulate two algorithms that are aware of the characteristics of an WSN, namely: (i) the ATC algorithm with coefficient reuse, whose purpose is to reveal the credibility of the most recent information in detriment to older information in a distributed context; and (ii) the diffuse VRC-LMS algorithm. The latter of which has a $\varpi_{l,k}$ parameter. This was incorporated so that nodes that contemplate a greater diversity of spatial information in the neighborhood will have more credibility for the distributed inference process, when compared against nodes imbued with less spatial diversity of information.

6.1 Proposal of an ATC Algorithm with Reuse of Coefficients

First, the ATC update process needs to be adapted in the derivation paradigm of adaptive filtering algorithms. This is done through the solution of a deterministic problem. Subsequently, we can insert the technique of coefficients reuse, through the definition of the following problem[7]:

$$\min_{\boldsymbol{\psi}_i(k+1)} \rho^l \sum_{l=0}^{L-1} \|\boldsymbol{\psi}_i(k+1) - \boldsymbol{w}_i(k-l)\|^2 \tag{47}$$

$$\text{s.t. } d_i(k) - \boldsymbol{\psi}_i^T(k+1)\boldsymbol{x}_i(k) =$$

$$(1-\beta\|\boldsymbol{x}_i(k)\|^2)\left[d_i(k) - \theta(\rho)\textstyle\sum_{l=0}^{L-1} \boldsymbol{w}_i^T(k-l)\boldsymbol{x}_i(k)\right].$$

Using the Lagrange multiplier allows for the minimization of (47) by finding the minimum of the following cost function:

$$\begin{aligned}\mathcal{F}_{\text{RC}}^{(\text{ATC})}[\boldsymbol{\psi}_i(k+1)] \quad \triangleq \quad & \sum_{l=0}^{L-1} \rho^l \|\boldsymbol{\psi}_i(k+1) - \boldsymbol{w}_i(k-l)\|^2 \\ & +\lambda\{d_i(k) - \boldsymbol{\psi}_i^T(k+1)\boldsymbol{x}_i(k) \\ & -(1-\beta\|\boldsymbol{x}_i(k)\|^2)\overline{e}_i(k)\},\end{aligned}$$

where

$$\overline{e}_i(k) \triangleq \left[d_i(k) - \theta(\rho)\sum_{l=0}^{L-1} \boldsymbol{w}_i^T(k-l)\boldsymbol{x}_i(k)\right]. \tag{48}$$

The minimization of (48) can be performed by zeroing its gradient with respect to $\boldsymbol{\psi}_i(k+1)$, that is

$$\nabla_{\boldsymbol{\psi}_i(k+1)}\mathcal{F}_{\text{RC}}^{(\text{ATC})}[\boldsymbol{\psi}_i(k+1)] =$$

$$2\sum_{l=0}^{L-1} \rho^l \boldsymbol{\psi}_i(k+1) - 2\sum_{l=0}^{L-1} \rho^l \boldsymbol{w}_i(k+1) - \lambda\boldsymbol{x}_i(k) = \boldsymbol{0}, \tag{49}$$

so that the following identity may be established

$$\boldsymbol{\psi}_i(k+1) = \theta(\rho)\sum_{l=0}^{L-1} \rho^l \boldsymbol{w}_i(k-l) + \frac{\theta(\rho)\lambda}{2}\boldsymbol{x}_i(k). \tag{50}$$

Applying the constraint of (47) to (50), it is possible to get the value of the Lagrange multiplier λ:

$$\lambda = \frac{2\beta\overline{e}_i(k)}{\theta(\rho)}. \tag{51}$$

[7]This derivation is unpublished, consisting of a contribution of this work.

Substituting λ into (50) (using (51)), one obtains

$$\boldsymbol{\psi}_i(k+1) \quad = \quad \theta(\rho)\sum_{l=0}^{L-1}\rho^l\boldsymbol{w}_i(k-l)+\beta\boldsymbol{x}_i(k)\left[d_i(k)-\theta(\rho)\sum_{l=0}^{L-1}\rho^l\boldsymbol{w}_i^T(k-l)\boldsymbol{x}_i(k)\right],$$

This equation describes the update of the *proposed* RC-LMS algorithm (in the ATC modality).

When the measurement noise presents high energy, the oscillations of the adaptive estimator contribute to increase the MSD, which can be critical in several applications [31]. In this case, the insertion of the coefficients reuse (as described in (52)) can improve the performance in the permanent regime. Notice that parameter L allows for controling the magnitude of reuse. More specifically, the higher the value of L, the greater the reuse (and therefore better performance in steady state). Parameter $\rho \in (0,1]$ in turn, regulates the weight given to more recent vectors, so that a value of ρ near 1 gives equal importance to L previously estimated vectors (i.e., $\boldsymbol{w}_i(k-l)$ to $l \in \{0,1,\ldots,L-1\}$, while values of ρ closer to zero give little importance to vectors $\boldsymbol{w}_i(k-l)$ that are more distant with respect to time.

6.2 VRC-LMS Algorithm

The previous subsection presented a new algorithm that inserted the RC methodology in the ATC method. However, such a proposal does not consider the characteristics of the communication protocol. In order to derive a protocol-aware distributed adaptive filtering method, it is necessary to note in the first place that in a context of shared medium access, not all of the transmited information will arrive to the nodes within range. This is due to package loss, which may be a result of collisions. In the following, a more sophisticated version of the proposed RC-ATC-LMS algorithm is advanced, which is aware of the packet loss phenomenon.

An undesirable feature of the RC-LMS algorithm is the reduction of the convergence rate [53]. Reference [30] compensated this behavior between the MSD and the convergence rate through the dynamic adjustment of the reuse order $L(k)$, which can be incorporated into the diffuse context as follows:

$$L_i(k) = \begin{cases} \min\{L_i(k-1)+1, L_i^{\max}\}, & \text{if } \overline{e}_i^2(k) < \eta \\ \max\{L_i(k-1)-1, 1\}, & \text{otherwise} \end{cases}, \tag{52}$$

so that the limit η is chosen according to the MSE of the theoretical steady state of the algorithm LMS [54],

$$\eta = \beta\sigma_v^2 \text{Tr}(\boldsymbol{R}_x)/2, \tag{53}$$

with constant β. In order to derive the update equation of the VRC-LMS algorithm, the proposed problem starts by minimizing the following equation:

$$\min_{\boldsymbol{\psi}_i(k+1)} \sum_{l=0}^{L_{i,k}-1}\rho^l\varpi_{l,k}\|\boldsymbol{\psi}(k+1)-\boldsymbol{w}_i(k-l)\|^2 \tag{54}$$

$$\text{s.t. } d_i(k)-\boldsymbol{\psi}_i^T(k+1)\boldsymbol{x}_i(k) = (1-\beta\|\boldsymbol{x}_i(k)\|^2)\overline{e}_i(k),$$

where $\overline{e}_i(k)$ is defined by

$$\overline{e}_i(k) \triangleq d_i(k) - \frac{1}{s_{i,k}}\sum_{l=0}^{L_{i,k}-1}\rho^l\varpi_{l,k}\boldsymbol{w}_i^T(k-l)\boldsymbol{x}_i(k), \tag{55}$$

where $s_{i,k} \triangleq \sum_{l=0}^{L-1}\rho^l\varpi_{l,k}$.

In the k-th iteration, the i-th node uses a time-variant reuse factor equal to $L_{i,k}$. Thus, $L_{i,k}$ previous vectors $\boldsymbol{w}_i^T(k-l)$ (for $l \in \{0,\ldots,L_{i,k}-1\}$) are taken into consideration. Our proposal consists in weighting each one of these vectors by a factor $\rho^l\varpi_{l,k}$, which consists of a product of two terms.

Assuming a parameter $\rho < 1$, the first of these terms (ρ^l) gives greater weights to more recent vectors, which are assumed to carry more reliable information. In the particular case where $\rho = 1$, such a term gives equal credibility to each of the $L_{i,k}$ vectors in the past. The adoption of this factor (ρ^l) is typical in coefficient reuse algorithms [31].

However, in the context of distributed adaptive filtering aware of shared access characteristics of the environment, one knows that vector $\boldsymbol{w}_i^T(k-l)$ is calculated by a combination of $|\mathcal{N}_{i,k,l}|$ vectors, where the neighborhood cardinality $|\mathcal{N}_{i,k,l}|$ is time variant due to the fact that packet loss is stochastic. Vectors $\boldsymbol{w}_i^T(k-l)$ whose neighborhood presents more cardinality are more reliable, as they aggregate more information in spatial terms. That motivates the proposal of the second term ($\varpi_{l,k}$), which emphasizes contributions of vectors that contain greater spatial diversity of information. It is worth emphasizing that the insertion of such a term is another contribution of this work.

Since the i-th node in the k-th iteration reuses $L_{i,k}$ previous vectors $\boldsymbol{w}_i(k-l)$, we define the largest value of neighborhood cardinality in this time window (with $L_{i,k}$ vectors) as

$$\varpi_{i,k}^{\max} = \max\{|\mathcal{N}_{i,k,0}|, |\mathcal{N}_{i,k,1}|, \ldots, |\mathcal{N}_{i,k,L_{i,k}-1}|\}. \tag{56}$$

Thus, using $\varpi_{i,k}^{\max}$ as the denominator of normalization, we propose that the weighting related to the amount of aggregate spatial information is defined by

$$\varpi_{l,k} \triangleq \frac{|\mathcal{N}_{i,k,l}|}{\varpi_{i,k}^{\max}}. \tag{57}$$

Therefore, the weighting factor associated with vector $\boldsymbol{w}_i(k-k)$ is given by

$$\rho^l \varpi_{l,k} = \rho^l \frac{|\mathcal{N}_{i,k,l}|}{\varpi_{i,k}^{\max}}. \tag{58}$$

After replacing the weighting factor (58) in equation (52) and adjusting the L vectors to be variant in time, that is, the algorithm starts to reuse $L_i(k)$ vectors, additionally, substituting $\theta(\rho)$ by $\frac{1}{s_{i,k}}$ in (52), the update equation from diffuse VRC-LMS algorithm can be denoted as:

$$\boldsymbol{\psi}_i(k+1) = \frac{1}{s_{i,k}} \sum_{l=0}^{L_i(k-1)} \rho^l \varpi_{l,k} \boldsymbol{w}_i(k-l) + \beta \boldsymbol{x}_i(k) \left[d_i(k) - \frac{1}{s_{i,k}} \sum_{l=0}^{L_i(k-1)} \rho^l \varpi_{l,k} \boldsymbol{w}_i^T(k-l) \boldsymbol{x}_i(k) \right]. \tag{59}$$

7 Simulation Structure and Results

7.1 Simulation Set–Up

The general simulation functions are illustrated in Figure 5 which are divided into:

(1) LR-WPAN model: divided in PHY, MAC, positioning and energy;

(2) Simulation: generates the necessary files for proposed algorithms;

(3) Proposed Algorithms: block where the simulation data is delivered as input data for the distributed VRC-LMS and RC-LMS algorithms;

(4) Metrics: responsible for generating comparative performance charts.

The positioning model has the task of allocating node positions in a given area. In this model, we consider that the position of the nodes must be randomly allocated within a disk area. The respective control attributes are: random variable ρ, $\mathcal{X}$ and $\mathcal{Y}$. These respectively represent: (i) radius of a position on a random disk; (ii) x axis of the center of the random position disc; and (iii) y axis of the center of the random position disk.

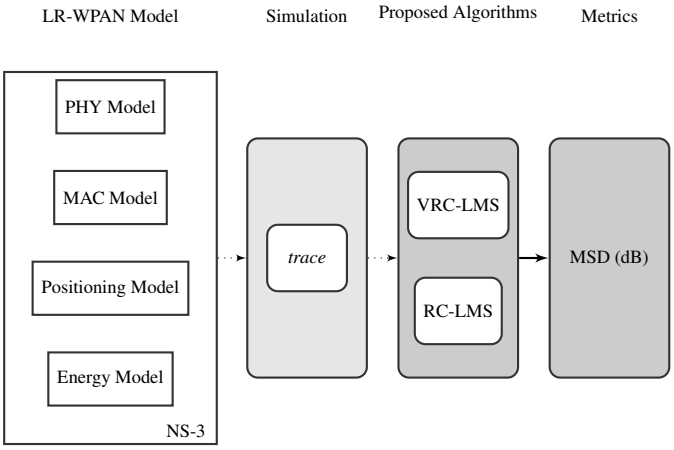

Figure 5: General Simulation Structure.

The PHY model tracks physical layer transitions (*i.e.,*the state machine of Figure 4) namely `BusyTx` $\rightarrow$ `TxOn` and `BusyRx` $\rightarrow$ `RxOn`. These correspond, respectively, to the end of transmission and reception. Associated with the transitions of the physical layer is the energy calculation. The energy model is responsible for making the product between the time spent and the energy level of each state. This calculates the amount of energy spent by a transmission or reception in the physical layer incorporating features of the CC2420 device (40). The energy model also calculates the network lifetime, which corresponds to the time required for the energy of all nodes to reach zero (at which time the simulation is terminated).

The MAC Model performs all the operations of the second network layer, that is, all the results from the media access control performed by the CSMA/CA protocol. Through a *Trace* tool (also shown in Figure 5), the MAC sublayer events are tracked.

These are structured through a matrix composed by S lines and m columns, with $\{n = 1, \ldots, 5\}$ and $m \to \infty$. The data in the first column corresponds to the operating mode op_{id} of the physical layer at the given instant, where the value may vary between `t` (transmition), `r` (reception) and `d` (*drop*). The second column corresponds to time instant `T`, in seconds, where the operation of the first column is running. The third column corresponds to the node index n_{id}, whose operation is being carried out. The fourth column indicates the sequence number of the frame being transmitted, received or that has been dropped (*drop*). Often in networks, the data stream is not ordered, so the sequence number guarantees the reconstruction of the information in the recipient. Through this method, we can consider whether a frame is older or newer. Finally, the fifth column indicates the physical address associated to the communication interface of each node. This address is unique and registered in hexadecimal format, *i.e.,* no two devices have the same information.

$$\begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} & S_{15} \\ S_{21} & S_{22} & S_{23} & S_{24} & S_{24} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_{m1} & S_{m2} & S_{m3} & S_{m4} & S_{m5} \end{bmatrix} = \begin{bmatrix} \texttt{t} & \texttt{1.00256} & \texttt{0} & \texttt{212} & \texttt{00:00} \\ \texttt{r} & \texttt{1.0032} & \texttt{1} & \texttt{212} & \texttt{00:00} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \texttt{d} & \texttt{111.417} & \texttt{4} & \texttt{213} & \texttt{00:04} \end{bmatrix}$$

In the block where the proposed algorithms act (shown in Figure 5), a function called *readDataTrace* was created, whose role is to read the data from matrix S. A second function called *extractInformationLine*, is responsible for importing the reference data into the algorithms.

7.2 Results

We consider that all nodes are within the range of communication, that is, they are positioned in a geographical area so that all devices can communicate with each other. Accordingly, the positioning model is configured so that the nodes are located in radius circle equivalent to 50 m. In the following sections, we will evaluate the performance of the proposed algorithms by combining adaptive filtering and telecommunications metrics. The results will be presented through a table containing the summarization of the main network attributes presented after the simulation in NS-3. The attributes list contains: number of collisions, transmissions and receptions, number of nodes, average frame size, network life (which corresponds to the end of simulation after the energy level of all nodes reach zero), transfer rate and energy level of nodes. Performance comparisons between the RC-LMS algorithms distributed in the ATC mode will be presented, and the VRC-LMS algorithm will be aware of the characteristics of a WSN.

7.2.1 Scenario 1

The first scenario consists of a network containing 20 nodes, according to Figure 6. In this scenario, we verified the performance of the RC-LMS distributed algorithm in the ATC mode and the VRC-LMS algorithm using the following adaptive filtering parameters: RC-LMS with $L = [1, \ldots, 4]$ and VRC-LMS with $L_i^{\max} = 4$. The other parameters are applicable to both algorithms: $\beta = 10^{-1}$, $\rho = 0.9$, $\sigma_\nu^2 = 10^{-6}$, $\boldsymbol{w}^*(z) = 1 - 0.8z^{-1} + 0.2z^{-2}$. The results were obtained through a single network experiment in NS-3, that is, the simulation of the LR-WPAN Model runs out one time and results are printed into the *trace* tool, which we may refer as the NS-3 domain simulations. These results are carried out to the proposed algorithms, which are optimized through 200 independent Monte Carlo attempts. Metrics are represented in Figure 7. As for the network parameterizations, we summarize them in Table 1.

In this point-to-point topology, we ensure that all nodes are under the reach of each other. The communication method in this case consists of each node sending 40 bytes of information to the *broadcast* address, with a 100 ms interval between each transmission, that is, maintaining a transfer rate of approximately 3.2 kbits/s. The goal is to get the message sent by a node to be received by the other nodes in the network. Because the transmissions are broadcast to all nodes in 100 ms intervals between each send, we will call this *intercalated broadcast* scenario. It can be seen that the performance of the VRC-LMS algorithm (yellow solid curve) shows a convergence rate in the transient close to the RC-LMS case in the ATC mode with $L = 1$ (corresponding to the traditional LMS algorithm), and a performance in the steady state above the other cases of the curves of the RC-LMS algorithm in the ATC modality with $L = [1, \ldots, 4]$. Since the VRC-LMS algorithm adopts the use of $L_i^{\max}$, a possible loss of the convergence rate in the transient is compensated by a reduced reuse of coefficients. Consequently, a higher reuse of coefficients in the steady state results in improved performance of the MSD. Due to the configuration of the network parameters summarized in Table 1, this scenario does not include many collisions because it does not present shared medium saturation. In the next scenario, we will explore a case of greater channel utilization by reducing the window between one transmission and another, while maintaining the average frame size, network topology, number of nodes, and power level.

7.2.2 Scenario 2

Scenario 2 uses the same adaptive filtering specification as Scenario 1 with the same network topology. However, in order to explore a possible channel saturation, we reduced window time between one transmission and another to 50 ms, which generates a transfer rate approximately twice as large as the first scenario (≈6.4 kbits/s).

In the Figure 8, for $L = [2, \ldots, 4]$, we have a slower convergence rate in the transient, coupled with a more stable performance in the permanent regime. As in Scenario 1, through the adoption of $L_i^{\max}$, the VRC-LMS algorithm compensates for a possible

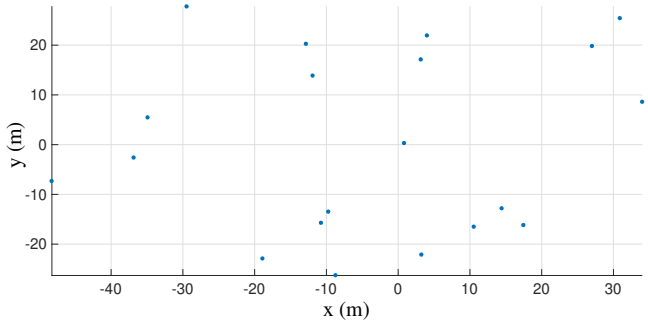

Figure 6: Network topology considered in Scenarios 1, 2 and 3.

Table 1: Network Parameters - Scenario 1

Parameters	**Values**
Number of nodes	20
Average frame size (Bytes)	40
Method of transmission	Broadcast
Throughput	3.2 kbits/s
Nodes energy level (Joules)	20
Network lifetime	323.971
Number of transmissions	3239
Number of receptions	61541
Number of collisions	3240

loss of the convergence rate in the transient by: (i) the reuse of coefficients less intense in that stretch; and (ii) a more intense reuse in the permanent regime. As a result, the VRC-LMS algorithm has improved performance in both the transient and the steady state, compared to the RC-LMS case in the ATC modality with $L = 1$.

7.2.3 Scenario 3

Scenario 3 uses the same adaptive filtering settings as Scenarios 1 and 2: RC-LMS with $L = [1, \ldots, 4]$ and VRC-LMS with $L_i^{\max} = 4$. The remaining attributes are applicable to both algorithms: $\beta = 10^{-1}$, $\rho = 0.9$, $\sigma_\nu^2 = 10^{-6}$, $\boldsymbol{w}^*(z) = 1 - 0.8z^{-1} + 0.2z^{-2}$. The network topology also follows the allocation illustrated in Figure 6.

Differently from the previous scenarios, we have changed the transmission method, in which all nodes simultaneously transmit to the *broadcast* address. As in Scenario 2, since the algorithm that holds the access control to the channel is saturated due to many requests, we intend to increase the number of collisions in order to evaluate the performance of the proposed algorithms. Therefore, we will call this scenario as the *simultaneous broadcast*. Table 3 summarizes the network parameters. One can readly see that the number of collisions is greater than the number of successful transmissions. In addition, we kept the information transmission rate at 3.2 kbits/s, average frame size at 40 Bytes, power level and number of network nodes is similar to previous scenarios. In Figure 9 we can see the evolution of the MSD curve, which has a behavior similar to Scenario 2 in the transient when using the RC-LMS algorithm with $L = [1, \ldots, 4]$. The VRC-LMS algorithm also presented a superior performance compared to $L = 1$ RC-LMS case, both in the transient and in the permanent regime due to the use of time-varying characteristics of the network that are smoothed by $L_i^{\max}$ variable.

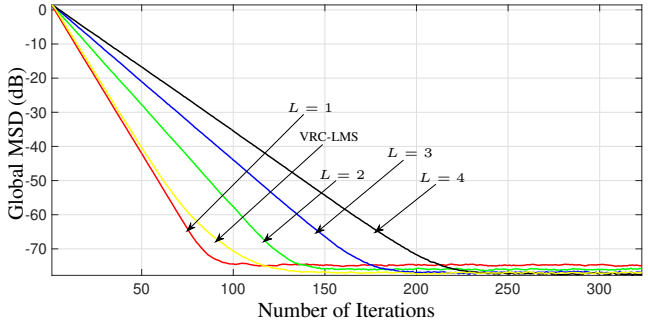

Figure 7: Global MSD (dB) evolution of the proposed algorithms - Scenario 1.

Table 2: Network Parameters - Scenario 2

Parameters	**Values**
Number of nodes	20
Average frame size (Bytes)	40
Throughput	6.4 kbits/s
Method of transmission	Broadcast interspersed
Nodes energy level (Joules)	20
Network lifetime (s)	324.796
Numer of transmitions	1728
Number of receptions	19950
Number of collisions	63939

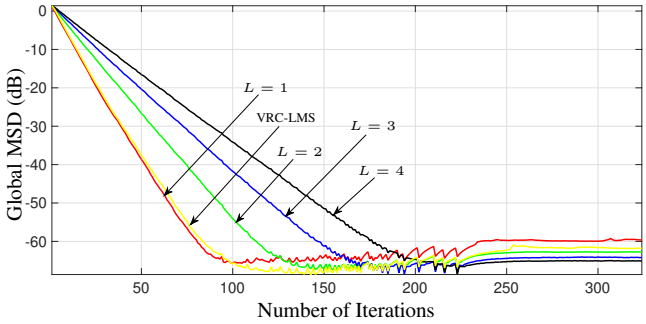

Figure 8: Global MSD (dB) evolution of the proposed algorithms - Scenario 2.

7.2.4 Scenario 4

In this simulation set–up, the number of nodes is increased to 50 (see Figure 10) and the average frame size is restricted to 40 bytes, with a transfer rate of 3.2 kbits/s. The adaptive filtering parameters used were: RC-LMS with $L = [1, \ldots, 4]$ and VRC-LMS with $L_i^{\max} = 4$. The other attributes are applicable to both algorithms: $\beta = 10^{-1}$, $\rho = 0.9$, $\sigma_\nu^2 = 10^{-6}$, $\boldsymbol{w}^*(z) = 1 - 0.8z^{-1} + 0.2z^{-2}$. The results were also obtained through a single experimental network realization in NS-3, while the results of the proposed algorithms are the result of 200 independent Monte Carlo attempts. It is worth to make a parallel between the evolution curve of MSD *versus* the number of iterations shown in Figure 12 and the collision dispersion presented in Figure 11. In the abscissa axis of the Figure 11, we have the time scale in seconds, and in the coordinate axis we have the number of collisions. We can see a concentration of collisions between $(100, 200)$. Both parameters proposed in this work contribute to the inference process: $L_i^{\max}$ contributes to a more vertiginous rate of convergence in the transient allied to a better performance in the permanent regime by increasing the coefficients reuse. $\varpi_{l,k}$ contributes to the use of spatial information diversity. Namely, the high number of collisions contributes so that more nodes have less relevant information for the estimation process. In this way, the performance of the solid curve in yellow presented similar learning to the case of the RC-LMS algorithm in the ATC modality with $L = 1$ (red solid curve) in the transient, and a superior performance in the permanent regime. In addition, it is noticed that, although the signal-to-noise ratio has increased (due to the number of collisions), there was not a large deviation of the MSD that compromised the stability of the learning process of the algorithm.

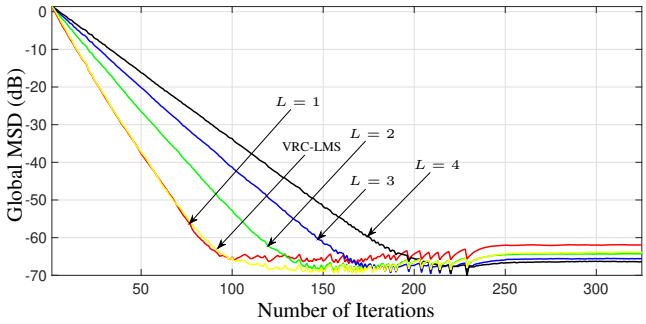

Figure 9: Global MSD (dB) evolution of the proposed algorithms - Scenario 3.

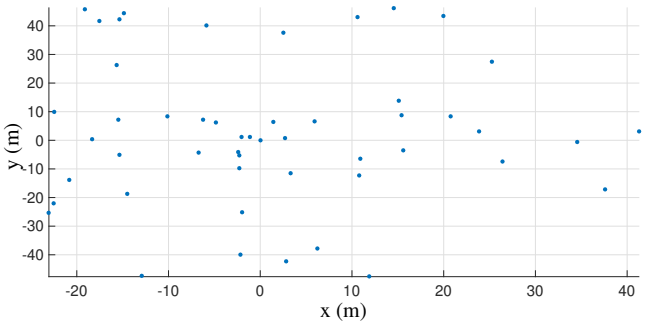

Figure 10: Network Topology considered in Scenario 4.

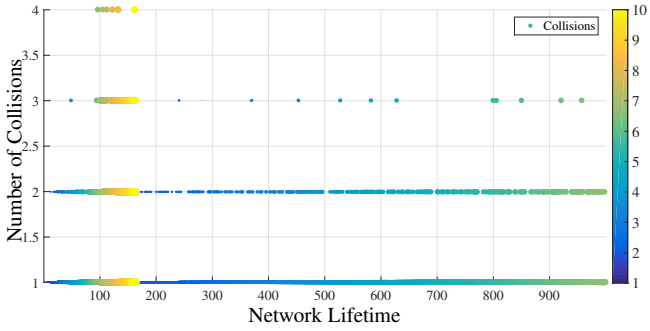

Figure 11: Collision dispersion of Scenario 4.

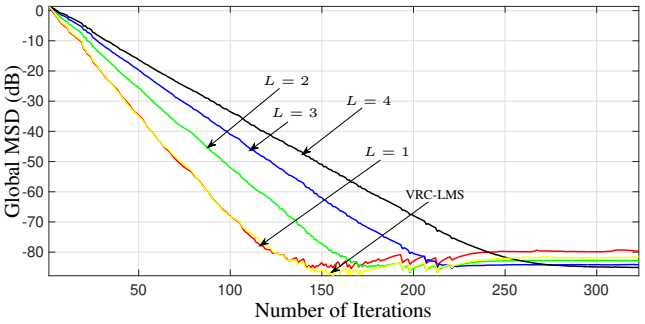

Figure 12: Global MSD (dB) evolution of the proposed algorithms - Scenario 4.

Table 3: Network Parameters - Scenario 3

Parameters	Values
Number of nodes	20
Average frame size (Bytes)	40
Throughput	3.2 kbits/s
Method of transmission	Simultaneous broadcast
Nodes energy level (Joules)	20
Network lifetime (s)	326.423
Number of transmitions	1612
Number of receptions	14539
Number of collisions	64414

Table 4: Network Parameters - Scenario 4

Parameters	Values
Number of nodes	50
Average frame size (Bytes)	40
Throughput	3.2 kbits/s
Method of transmition	Simultaneous broadcast
Nodes energy level (Joules)	20
Network lifetime (s)	999.294
Number of transmition	2942
Number of receptions	28804
Number of collisions	154469

8 Conclusions

Adaptive filtering algorithms typically have robust self-learning ability and tracking features. Due to these characteristics, they have been applied in several problems, especially when the statistical characteristics of the signal under study are not known or are time-variant.

This paper incorporates features derived from realistic behaviors that occur in communication processes into the design of a distributed adaptive identification algorithm. Namely, when the information is delayed, which may happen due to collisions or other network effects, older information becomes less reliable, which permits one to adopt a reuse of coefficients based on the current characteristics of the network. This premise allowed a more intense reuse of recent information in the context of diffuse adaptive filtering, whose implementation is one of the contributions of this work. Also aligned to WSN scenarios, we adopted a coefficient of spatial diversity, in order to add the estimates of network sensors in the neighborhood. This term had not been used yet within the VRC-LMS algorithm in the literature.

The distributed adaptive filtering methods with coefficients reuse incorporate the dispute for the shared medium considering a more realistic fashion. This was done through the NS–3 simulator. To the best of our knowledge, this is the first time that such a method is discussed. We ran simulations, varying several network parameters, and the proposed method outperforms the traditional ATC distributed LMS algorithm.

REFERENCES

[1] D. Takaishi, H. Nishiyama, N. Kato and R. Miura. "Toward Energy Efficient Big Data Gathering in Densely Distributed Sensor Networks". *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 3, pp. 388–397, Sept 2014.

[2] C. Zhou, C. K. Tham and M. Motani. "Finding Decomposable Models for Efficient Distributed Inference over Sensor Networks". *IEEE Transactions on Mobile Computing*, pp. 1–1, 2018.

[3] P. K. Varshney. *Distributed detection and data fusion*. Springer Science & Business Media, 2012.

[4] S. Haykin and B. Widrow. *Least-mean-square adaptive filters*, volume 31. John Wiley & Sons, 2003.

[5] S. P. Talebi, S. Werner and D. P. Mandic. "Distributed Adaptive Filtering of α-Stable Signals". *IEEE Signal Processing Letters*, vol. 25, no. 10, pp. 1450–1454, Oct 2018.

[6] G. F. Riley and T. R. Henderson. "The ns-3 Network Simulator." In *Modeling and Tools for Network Simulation*, edited by K. Wehrle, M. Günes and J. Gross, pp. 15–34. Springer, 2010.

[7] "IEEE Standard for Low-Rate Wireless Networks", April 2016.

[8] S. E. Kim, J. W. Lee and W. J. Song. "Steady-state analysis of the NLMS algorithm with reusing coefficient vector and a method for improving its performance". In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4120–4123, May 2011.

[9] P. S. R. Diniz, E. A. B. da Silva and S. L. Netto. *Digital Signal Processing: System Analysis and Design*. Cambridge University Press, second edition, 2010.

[10] A. H. Sayed. "Adaptive Networks". *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, April 2014.

[11] P. S. R. Diniz. *Adaptive Filtering - Algorithms and Practical Implementation*. Federal University of Rio de Janeiro, 2008.

[12] S. Haykin. *Adaptive Filter Theory*. Pearson, 2014.

[13] D. B. Haddad. "Estruturas em Subbandas para Filtragem Adaptativa e Separação Cega e Semi-Cega de Sinais de Voz". Ph.D. thesis, UFRJ/COPPE, Jun. 2013.

[14] M. Ho, J. M. Cioffi and J. A. C. Bingham. "Discrete multitone echo cancelation". *IEEE Transactions on Communications*, vol. 44, no. 7, pp. 817–825, July 1996.

[15] S. Javed and N. A. Ahmad. "An adaptive noise cancelation model for removal of noise from modeled ECG signals". In *2014 IEEE REGION 10 SYMPOSIUM*, pp. 471–475, April 2014.

[16] H. Kim. *Channel Estimation and Equalization*. Wiley, 2015.

[17] D. Tarrazó-Serrano, S. Pérez-López, S. Castiñeira Ibáñez, P. Candelas and C. Rubio. "USING NUMERICAL MODELS TO UNDERSTAND ACOUSTIC BEAM CONFORMATION". In *12th International Technology, Education and Development Conference*, pp. 5610–5616, 03 2018.

[18] A. H. Sayed. "Diffusion strategies for adaptation and learning over networks: an examination of distributed strategies and network behavior". *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 155–171, May 2013.

[19] S. Xu. *Distributed Signal Processing Algorithms for Wireless Networks*. Communications and Signal Processing Research Group, Electronics, University of York, 2015., 2015.

[20] S. G. Sankaran and A. A. L. Beex. "Convergence behavior of affine projection algorithms". *IEEE Transactions on Signal Processing*, vol. 48, no. 4, pp. 1086–1096, Apr 2000.

[21] A. Sayed. "Adaptation, Learning, and Optimization over Networks". *Found. Trends Mach. Learn.*, vol. 7, no. 4-5, pp. 311–801, July 2014.

[22] G. Yang and V. Danos. "Learning in Open Adaptive Networks". In *2016 IEEE 10th International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pp. 50–59, Sept 2016.

[23] C. G. Lopes and A. H. Sayed. "Incremental Adaptive Strategies Over Distributed Networks". *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4064–4077, Aug 2007.

[24] N. Takahashi, I. Yamada and A. H. Sayed. "Diffusion Least-Mean Squares With Adaptive Combiners: Formulation and Performance Analysis". *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4795–4810, Sep. 2010.

[25] A. H. Sayed. "Diffusion Adaptation Over Networks". *Academic Press Library in Signal Processing*, vol. 3, 05 2012.

[26] L. Xiao and S. Boyd. "Fast linear iterations for distributed averaging". In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 5, pp. 4997–5002 Vol.5, Dec 2003.

[27] R. Olfati-Saber and R. M. Murray. "Consensus problems in networks of agents with switching topology and time-delays". *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, Sept 2004.

[28] A. Jadbabaie, J. Lin and A. S. Morse. "Coordination of groups of mobile autonomous agents using nearest neighbor rules". In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 3, pp. 2953–2958 vol.3, Dec 2002.

[29] H. Cho, C. W. Lee and S. W. Kim. "Derivation of a new normalized least mean squares algorithm with modified minimization criterion". *Signal Processing*, vol. 89, no. 4, pp. 692 – 695, 2009.

[30] S. E. Kim, J. W. Lee and W. J. Song. "Steady-state analysis of the NLMS algorithm with reusing coefficient vector and a method for improving its performance". In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4120–4123, May 2011.

[31] L. Resende, D. Haddad and M. R Petraglia. "A Variable Step-Size NLMS Algorithm with Adaptive Coefficient Vector Reusing". In *IEEE INTERNATIONAL CONFERENCE on ELECTRO/INFORMATION TECHNOLOGY, At Rochester, Michigan, USA*, 10 2018.

[32] L. Resende, R. Pimenta, N. Norat, D.B.Haddad and M.R.Petraglia. "LMS Algorithm with Reuse of Coefficients and Robustness Against Impulsive Noise". In *XXII Congresso Brasileiro de Automática*. XXII Congresso Brasileiro de Automática, 2018.

[33] P. Park, C. H. Lee and J. W. Ko. "Mean-Square Deviation Analysis of Affine Projection Algorithm". *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 5789–5799, Dec 2011.

[34] J. Ni. "Improved normalised subband adaptive filter". *Electronics Letters*, vol. 48, no. 6, pp. 320–321, March 2012.

[35] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci. "Wireless sensor networks: a survey". *Computer Networks*, vol. 38, no. 4, pp. 393 – 422, 2002.

[36] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta and Y. F. Hu. "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards". *Computer Communications*, vol. 30, no. 7, pp. 1655 – 1695, 2007. Wired/Wireless Internet Communications.

[37] H. Wu, S. Nabar and R. Poovendran. "An Energy Framework for the Network Simulator 3 (NS-3)". In *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, SIMUTools '11, pp. 222–230, ICST, Brussels, Belgium, Belgium, 2011. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[38] R. Min and A. Chandrakasan. "A Framework for Energy-scalable Communication in High-density Wireless Networks". In *Proceedings of the 2002 International Symposium on Low Power Electronics and Design*, ISLPED '02, pp. 36–41, New York, NY, USA, 2002. ACM.

[39] E. Shih, S. Cho, F. S. Lee, B. H. Calhoun and A. Chandrakasan. "Design Considerations for Energy-Efficient Radios in Wireless Microsensor Networks". *J. VLSI Signal Process. Syst.*, vol. 37, no. 1, pp. 77–94, May 2004.

[40] C. P. from Texas Instruments. "2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver", 2018.

[41] NS3. "NS3 Model Library", 2019.

[42] P. T. G. Norberto Barroca and F. J. Velez. "Impact of Switching Latency Times in Energy Consumption of IEEE 802.15.4 Radio Transceivers". *The 9th Conference on Telecommunications*, 2012.

[43] L. Kleinrock and F. Tobagi. "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics". *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1400–1416, December 1975.

[44] J. W. Chong, D. K. Sung and Y. Sung. "Cross-Layer Performance Analysis for CSMA/CA Protocols: Impact of Imperfect Sensing". *IEEE Transactions on Vehicular Technology*, vol. 59, no. 3, pp. 1100–1108, March 2010.

[45] H. Takagi and L. Kleinrock. "Throughput Analysis for Persistent CSMA Systems". *IEEE Transactions on Communications*, vol. 33, no. 7, pp. 627–638, July 1985.

[46] N. Eagle, M. Macy and R. Claxton. "Network Diversity and Economic Development". *Science*, vol. 328, no. 5981, pp. 1029–1031, 2010.

[47] C. Intanagonwiwat, D. Estrin, R. Govindan and J. Heidemann. "Impact of network density on data aggregation in wireless sensor networks". In *Proceedings 22nd International Conference on Distributed Computing Systems*, pp. 457–458, July 2002.

[48] J. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin and D. Ganesan. "Building Efficient Wireless Sensor Networks with Low-level Naming". *SIGOPS Oper. Syst. Rev.*, vol. 35, no. 5, pp. 146–159, October 2001.

[49] L. Krishnamachari, D. Estrin and S. Wicker. "The impact of data aggregation in wireless sensor networks". In *Proceedings 22nd International Conference on Distributed Computing Systems Workshops*, pp. 575–578, July 2002.

[50] T. T. Tran and H. Y. Kong. "A method enabling exploitation of spatial diversity and physical layer security in an extreme case of source-wiretapping without a jamming beamformer". *Journal of Communications and Networks*, vol. 17, no. 5, pp. 482–490, Oct 2015.

[51] M. Abolfath Beigi and S. Mohammad Razavizadeh. "Cooperative beamforming in Cognitive Radio networks". In *2009 2nd IFIP Wireless Days (WD)*, pp. 1–5, Dec 2009.

[52] S. Mijovic, C. Buratti, A. Zanella and R. Verdone. "A Cooperative Beamforming Technique for Body Area Networks". *Procedia Computer Science*, vol. 40, pp. 181 – 189, 2014. Fourth International Conference on Selected Topics in Mobile & Wireless Networking (MoWNet'2014).

[53] L. C. Resende, D. B. Haddad and M. R. Petraglia. "A Variable Step-Size NLMS Algorithm with Adaptive Coefficient Vector Reusing". In *2018 IEEE International Conference on Electro/Information Technology (EIT)*, pp. 0181–0186, May 2018.

[54] A. H. Sayed. *Adaptive Filters*. Wiley-IEEE Press, 2008.