

*Review*

# On Consensus-based Distributed Blind Calibration of Sensor Networks

Miloš S. Stanković<sup>1\*</sup>, Srdjan S. Stanković<sup>2</sup>, Karl Henrik Johansson<sup>3</sup>, Marko Beko<sup>4</sup> and Luis M. Camarinha-Matos<sup>5</sup>

<sup>1</sup> Innovation Center, School of Electrical Engineering, University of Belgrade, Serbia; Vlatacom Institute, Belgrade, Serbia; and School of Technical Sciences, Singidunum University, Belgrade, Serbia; milstank@gmail.com

<sup>2</sup> School of Electrical Engineering, University of Belgrade, Serbia; and Vlatacom Institute, Belgrade, Serbia; stankovic@etf.rs

<sup>3</sup> ACCESS Linnaeus Center, School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden; kallej@kth.se

<sup>4</sup> COPELABS, Universidade Lusófona de Humanidades e Tecnologias, Campo Grande 376, 1749-024 Lisboa, Portugal; and CTS/UNINOVA, Campus da FCT/UNL, Monte de Caparica, 2829-516 Caparica, Portugal; mbeko@uninova.pt

<sup>5</sup> Faculty of Sciences and Technology and UNINOVA CTS, NOVA University of Lisbon, Portugal; cam@uninova.pt

\* Correspondence: milstank@gmail.com

**Abstract:** This review paper deals with recently proposed algorithms for distributed blind macro-calibration of sensor networks based on consensus (synchronization), not requiring any fusion center. The basic algorithm, performing the estimation of the local calibration parameters, is derived commencing from appropriate local criteria, and developing the corresponding gradient descent scheme. It is shown that the estimated parameters of the calibration functions asymptotically converge, in the mean-square sense and with probability one (w.p.1), to such values that ensure consensus on calibrated sensor gains and calibrated sensor offsets. For the more realistic case in which additive measurement noise, communication dropouts and additive communication noise are present, two algorithm modifications are introduced: one using a simple compensation term, and a more robust one based on an instrumental variable. By utilizing stochastic approximation arguments it is shown that the modified algorithms also achieve asymptotic agreement for calibrated sensor gains and offsets, in the mean-square sense and w.p.1. Convergence rate is analyzed in terms of an upper bound of the mean-square error. It is also shown that the communications between nodes can be completely asynchronous, which is of substantial importance for real-world applications. Suggestions for design of *a priori* adjustable weights are given. Finally, it is shown that, if there is a subset of (precalibrated) reference sensors with fixed calibration parameters, the calibrated sensor gains and offsets of the rest of the sensors do not achieve consensus - they converge to different points dictated by the reference sensors and the network characteristics. Wide applicability and efficacy of these algorithms are illustrated on several simulation examples.

**Keywords:** Blind Calibration, Macro Calibration, Distributed Estimation, Sensor Networks, Consensus, Synchronization, Stochastic Approximation

## 1. Introduction

Recently emerged technologies dealing with networked systems, such as Internet of Things (IoT), Networked Cyber-Physical Systems (CPS), and Sensor Networks (SN), still have many conceptual and practical challenges intriguing to both researchers and practitioners [1–6]. New classes of problems in this area continuously arise driven by many new real-world applications. Particularly for the case of SNs, application examples include environment monitoring, wildfires detection, shop-floor

manufacturing, smart cities, etc. One of the most important challenges, limiting the performance, robustness and time-to-market of these new technologies, is *sensor calibration*. Micro-calibration can be performed only in relatively small SNs where every sensor is individually calibrated in a controlled environment. Typical SNs are of large scale, functioning in dynamic and partially unobservable environments, thus demanding new methods and algorithms for efficient calibration. The idea of *macro-calibration* is to calibrate the entire SN based on the total system response, so that there is no requirement to individually calibrate every sensor node. The typical approach is to formulate the calibration problem as a parameter estimation problem (e.g., [7,8]). Of significant interest are methods for automatic calibration of SNs which successfully perform even if there are no reference signals/sensors, or other sources of groundtruth information about the measured process. In these situations, the goal of the calibration is to achieve homogeneous behavior of all the nodes, possibly enforcing dominant influence of sensors that are *a priori* known to provide sufficiently good (calibrated) measurements. These types of calibration problems are known as the *blind calibration* problems [9–13]. Furthermore, in many applications of SNs, it is of essential importance that the network functions in a completely decentralized fashion, without the requirement for any kind of centralized information fusion. Hence, completely *distributed and decentralized* calibration algorithms are of paramount importance.

In this paper, we analyze recently proposed algorithms which possess all the mentioned desirable properties: they deal with *blind macro-calibration* of SNs based on completely *decentralized, real-time and recursive estimation* of the parameters of linear *calibration functions* [14–18]. Another advantageous property of these algorithms is that it is assumed that the underlying SN have *directed* communication links between neighboring nodes. A basic algorithm is developed by using a distributed optimization problem setup, with the local objectives formulated as weighted sums of mean-square differences between the corrected sensor readings of neighboring nodes. Consequently, the basic distributed calibration algorithm is derived by constructing a corresponding *distributed gradient recursive scheme*. A direct consequence of this problem setup is that the obtained algorithm can be studied as a *generalized consensus* problem. Hence, for the convergence analysis, it is not sufficient to use only standard arguments used in the consensus literature (e.g., [19]). Additionally, utilization of techniques dealing with the stability of diagonally quasi-dominant dynamical systems [20–22] is needed to demonstrate that the basic algorithm provides asymptotic agreement (consensus, synchronization) for both corrected sensor gains and corrected sensor offsets, in the mean-square sense and with probability one (w.p.1). The basic algorithm is extended by assuming presence of several factors which are of essential importance for practical applicability of the proposed method: 1) additive communication noise, 2) communication dropouts, 3) additive measurement noise, and 4) asynchronous communication.

Two possible modifications of the basic algorithm are introduced for solving the problems posed in the cases 1)–3). The first is based on the assumption that the noise variance is known a priori, which is used to design an appropriate compensation term. The second modification is more robust and is based on an instrumental variable usage [17]. In both cases, the attainment of the asymptotic consensus in the mean-square sense and w.p.1 are shown based on stochastic approximation arguments. In the case of completely asynchronous communication scenario, which is particularly important, we show how the algorithm can be implemented assuming a broadcast gossip communication scheme, which does not require clock synchronization among the agents, or any type of centralized information or coordination [15].

Another practically important situation arises when there are multiple nodes in the network that do not update (correct) their calibration parameters, but they still participate in the described distributed macro-calibration process. In this case, these nodes are called *reference nodes* since their only role is to provide reference information based on which other nodes should calibrate themselves. For example, this situation may arise in practice when a set of uncalibrated sensors is added to an already calibrated SN. We show that, in the case of more than one reference node, the corrected gains and offsets of the non-reference nodes, in general, do not converge to consensus, but to different points

which depend on the information dictated by the reference sensors and the network properties [15]. In the case of only one reference sensor, the corrected gains and offsets of the rest of the sensors converge to the same point imposed by the reference sensor.

Finally, an analysis is given which clarifies the influence of initially selected weights corresponding to particular nodes in the analyzed calibration parameters estimation recursions. Guidelines are formulated on how these weights should be chosen so that given requirements are satisfied. General discussion of the described results is provided from both theoretical and practical points of view, based on which several future research directions are proposed.

The outline of the rest of the paper is as follows. The following section briefly discusses related work. In Section 3 we introduce the distributed blind macro-calibration problem and derive the basic algorithm for the noiseless case. Section 4 is devoted to the analysis of the convergence properties of the base algorithm. In Section 5 certain assumptions about the measured signals, communication errors, and communication protocol are relaxed, and the appropriate algorithm modifications are introduced, together with their convergence analysis. In Section 6 a discussion on the convergence rate, the case of presence of reference sensors with fixed characteristics, and some design guidelines are presented. In Section 7 we present illustrative simulation results. Finally, Section 8 presents some conclusions.

## 2. Related Work

*Macro-calibration* is based on the idea of calibrating the whole SN based on the responses of all the nodes. The most frequent approaches to this problem are based on parameter estimation techniques (e.g., [8]). If controlled stimuli are not available the problem is usually referred to as *blind calibration* of SNs. In general, it is a difficult problem, which has certain similarities with more general problems of blind estimation, equalization, and deconvolution (e.g., [23–25] and references therein). In [9,10] a centralized and non-recursive blind calibration algorithm was analyzed assuming restrictive signal and sensor properties. In [26] the authors proposed to use sparsity and convex optimization for blind estimation of calibration gains. In [12] an approach to blind sensor calibration is adopted based on centralized consistency maximization at the network level assuming very dense deployment and only pairwise inter-node communications. In [13], the authors proposed a method which can manage situations in which density requirements are not met. The problem of distributed blind macro-calibration may have certain similarities with the clock synchronization approaches based on local data processing and communications with neighbors [27–34]. Finally, certain extended consensus algorithms have been applied to SN calibration problems, but in different settings than the one treated in this paper [35–38]. This paper is focused on the algorithms proposed recently in [14–18] representing the only completely distributed and decentralized blind macro-calibration algorithms for SNs having satisfactory performance under diverse deteriorating conditions.

## 3. Problem Definition and the Basic Algorithm

Assume that the SN to be calibrated consists of  $n$  nodes/sensors. In the base setup, it is assumed that each sensor is measuring the same signal  $x(t)$  in discrete-time instants  $t = \dots, -1, 0, 1, \dots$ ; this signal can be considered as a realization of a stochastic process  $\{x(t)\}$ . Note that we have implicitly assumed that the sensor nodes are functioning synchronously, since all the sensor nodes perform measurements in the same time instances  $t$ . We will relax this assumption in Subsection 5.3. The output (measurement) of the  $i$ -th node can be written as

$$y_i(t) = \alpha_i x(t) + \beta_i, \quad (1)$$

where  $\alpha_i$  is the unknown gain, and  $\beta_i$  the unknown offset of sensor  $i$ .

Calibration of a sensor is performed by applying an affine *calibration function* to the raw readings (1) which results in the following calibrated sensor output

$$z_i(t) = a_i y_i(t) + b_i = g_i x(t) + f_i, \quad (2)$$

where  $a_i$  and  $b_i$  are the calibration parameters to be obtained,  $g_i = a_i \alpha_i$  is the corrected gain and  $f_i = a_i \beta_i + b_i$  the corrected offset. The calibration objective is, ideally, to find parameters  $a_i$  and  $b_i$  for which  $g_i$  is equal to one and  $f_i$  equal to zero. In general, if we assume that there are no sensors which give perfect readings  $z_i(t) = x(t)$  and that the signal  $x(t)$  is unknown and cannot be obtained or measured by some other means, this objective is impossible to achieve. Hence, in our decentralized real-time blind macro-calibration problem setup, this ideal objective must be alleviated: we require that the calibration process asymptotically achieves equal calibrated outputs  $z_i(t)$  for all the nodes  $i = 1, \dots, n$ . In order to approach as close as possible to the ideal goal of achieving  $g_i = 1$  and  $f_i = 0$ , we could use certain a priori knowledge about the underlying SN, and try to adjust the algorithm, such that, loosely speaking, the “good” sensors (e.g., precalibrated or higher-quality sensors) correct, using the consensus strategy, the response of the rest of the sensors. For example, if, in a given SN, there is an *a priori* given perfectly calibrated reference sensor, the ideal asymptotic calibration ( $g_i = 1$  and  $f_i = 0$ ) of the rest of the sensor nodes will be achieved if the consensus goal is achieved.

It is assumed that the underlying SN have a predefined communication topology, defining possible inter-sensor communications, represented by a *directed graph*  $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ , where  $\mathcal{N}$  is the set of nodes (sensors) and  $\mathcal{E}$  the set of communication links (arcs). Define the adjacency matrix  $A = [a_{ij}]$ ,  $i, j = 1, \dots, n$ , where  $a_{ij} = 1$  if the  $j$ -th node is able to send messages to the  $i$ -th sensor, and  $a_{ij} = 0$  otherwise. Let  $\mathcal{N}_i$  be the set of in-neighboring nodes (or just neighbors) of the  $i$ -th node, i.e., the nodes  $j$  for which  $a_{ij} = 1$ . Similarly, let  $\mathcal{N}_i^{\text{out}}$  be the set of out-neighboring nodes of the  $i$ -th node, i.e., the nodes  $j$  for which  $a_{ji} = 1$ .

Let us now derive the basic calibration algorithm. The idea is to start with local criteria for each node, whose local minimization would lead to a network-level consensus on the corrected sensor outputs:

$$J_i = \sum_{j \in \mathcal{N}_i} \gamma_{ij} E\{(z_j(t) - z_i(t))^2\}, \quad (3)$$

$i = 1, \dots, n$ , where  $\gamma_{ij}$  are nonnegative scalar weights whose influence on the properties of the algorithm will be discussed later. Denoting  $\theta_i = [a_i \ b_i]^T$ , the following expression is obtain for the gradient of (3):

$$\text{grad}_{\theta_i} J_i = \sum_{j \in \mathcal{N}_i} \gamma_{ij} E\left\{(z_j(t) - z_i(t)) \begin{bmatrix} y_i(t) \\ 1 \end{bmatrix}\right\}. \quad (4)$$

From (4) we obtain the following stochastic *gradient recursion* for estimating  $\theta_i^*$  minimizing (3):

$$\hat{\theta}_i(t+1) = \hat{\theta}_i(t) + \delta_i(t) \sum_{j \in \mathcal{N}_i} \gamma_{ij} \epsilon_{ij}(t) \begin{bmatrix} y_i(t) \\ 1 \end{bmatrix}, \quad (5)$$

where  $\hat{\theta}_i(t) = [\hat{a}_i(t) \ \hat{b}_i(t)]^T$ ,  $\epsilon_{ij}(t) = \hat{z}_j(t) - \hat{z}_i(t)$ ,  $\hat{z}_i(t) = \hat{a}_i(t) y_i(t) + \hat{b}_i(t)$ , and  $\delta_i(t) > 0$  is a time-varying gain whose influence on the convergence properties of the algorithm will be discussed later. The initial conditions are assumed to be  $\hat{\theta}_i(0) = [1 \ 0]^T$ ,  $i = 1, \dots, n$ . We expect that the set of recursions (5) asymptotically achieve that all the local estimates of corrected gains  $\hat{g}_i(t) = \hat{a}_i(t) \alpha_i$  and corrected offsets  $\hat{f}_i(t) = \hat{a}_i(t) \beta_i + \hat{b}_i(t)$  converge to the *same values*  $\bar{g}$  and  $\bar{f}$ , respectively; this implies that the corrected sensor outputs of all the nodes are also equal  $\hat{z}_j(t) = \hat{z}_i(t)$ ,  $i, j = 1, \dots, n$ .

In Fig. 1 an illustrating smart-city example sensor network is depicted. Completely decentralized network architecture is assumed, i.e., the nodes communicate according to the directed communication graph which is represented in the figure using arcs. The communication graph will typically depend





**Figure 1.** An example sensor network used in smart-city applications with decentralized communication topology. The inter-node communication is performed according to the depicted directed graph. The introduced distributed calibration algorithm achieves asymptotic calibration of all the sensor nodes in the network without using any type of fusion center.

on the mutual node distances, transmission power of individual nodes, channel conditions, presence of obstacles, etc. Each node in the network is equipped with the same type of sensor which measures certain physical quantity (e.g., certain atmospheric condition or air quality indicator). At each time instant  $t$ , a node  $i$  performs local reading of the raw sensor output  $y_i(t)$ , calculation of the corrected sensor output  $\hat{z}_i(t)$  according to (2) using current local estimates of the calibration parameters  $\hat{a}_i(t)$  and  $\hat{b}_i(t)$ , transmission of the corrected value  $\hat{z}_i(t)$  to the out-neighbors  $\mathcal{N}_i^{\text{out}}$ , reception of the values  $\hat{z}_j(t)$  from the in-neighbors  $j \in \mathcal{N}_i$ , and calculation of the updated estimates of the local calibration parameters  $\hat{a}_i(t+1)$  and  $\hat{b}_i(t+1)$  using (5). In the initial presentation we will assume that, at each iteration of the algorithm (5), local sensor measurement  $y_i(t)$  and the current messages of the neighboring nodes' corrected outputs  $\hat{z}_j(t)$  are available at node  $i$ . Possible communication dropouts and/or faulty/noisy sensor readings will be treated later. Local computational cost for each agent is minor since only two parameters are being estimated. Communication complexity depends on the number of neighboring agents, which is small in typical SNs with decentralized architecture.

For the sake of compact notations, suitable for convergence analysis of the derived algorithm, let us introduce

$$\hat{\phi}_i(t) = \begin{bmatrix} \hat{g}_i(t) \\ \hat{f}_i(t) \end{bmatrix} = \begin{bmatrix} \alpha_i & 0 \\ \beta_i & 1 \end{bmatrix} \hat{\theta}_i(t), \quad (6)$$

and

$$\epsilon_{ij}(t) = \begin{bmatrix} x(t) & 1 \end{bmatrix} (\hat{\phi}_j(t) - \hat{\phi}_i(t)), \quad (7)$$

so that (5) becomes

$$\hat{\phi}_i(t+1) = \hat{\phi}_i(t) + \delta_i(t) \sum_{j \in \mathcal{N}_i} \gamma_{ij} \Omega_i(t) (\hat{\phi}_j(t) - \hat{\phi}_i(t)), \quad (8)$$

where

$$\begin{aligned}\Omega_i(t) &= \begin{bmatrix} \alpha_i y_i(t) x(t) & \alpha_i y_i(t) \\ [1 + \beta_i y_i(t)] x(t) & 1 + \beta_i y_i(t) \end{bmatrix} \\ &= \begin{bmatrix} \alpha_i \beta_i x(t) + \alpha_i^2 x(t)^2 & \alpha_i \beta_i + \alpha_i^2 x(t) \\ (1 + \beta_i^2) x(t) + \alpha_i \beta_i x(t)^2 & 1 + \beta_i^2 + \alpha_i \beta_i x(t) \end{bmatrix},\end{aligned}\quad (9)$$

with the initial conditions  $\hat{\rho}_i(0) = [\alpha_i \ \beta_i]^T$ ,  $i = 1, \dots, n$ . Therefore, the following compact form for the recursions (8) is obtained

$$\hat{\phi}(t+1) = [I + (\Delta(t) \otimes I_2)B(t)]\hat{\phi}(t), \quad (10)$$

where  $\otimes$  is the Kronecker product,  $I$  is the identity matrix of dimension  $2n$ ,  $I_2$  is the dimension 2 identity matrix,  $\hat{\phi}(t) = [\hat{\phi}_1(t)^T \cdots \hat{\phi}_n(t)^T]^T$ ,  $\Delta(t) = \text{diag}\{\delta_1(t), \dots, \delta_n(t)\}$ ,  $\text{diag}\{\dots\}$  denotes the corresponding block diagonal matrix,

$$B(t) = \Omega(t)(\Gamma \otimes I_2),$$

$$\Omega(t) = \text{diag}\{\Omega_1(t), \dots, \Omega_n(t)\},$$

$$\Gamma = \begin{bmatrix} -\sum_{j,j \neq 1} \gamma_{1j} & \gamma_{12} & \cdots & \gamma_{1n} \\ \gamma_{21} & -\sum_{j,j \neq 2} \gamma_{2j} & \cdots & \gamma_{2n} \\ & & \ddots & \\ \gamma_{n1} & \gamma_{n2} & \cdots & -\sum_{j,j \neq n} \gamma_{nj} \end{bmatrix},$$

where  $\gamma_{ij} = 0$  when  $j \notin \mathcal{N}_i$ , and the initial condition is  $\hat{\phi}(0) = [\hat{\phi}_1(0)^T \cdots \hat{\phi}_n(0)^T]^T$ , according to (8). From the way in which we have constructed the vector  $\hat{\phi}(t)$  we conclude that the asymptotic value of  $\hat{\phi}(t)$  should be such that all of its odd components are equal, and all of its even components are equal.

In the next section, it will be shown that, under certain general assumptions, for any choice of the weights  $\gamma_{ij} \geq 0$  for  $j \in \mathcal{N}_i$  (and  $\gamma_{ij} = 0$  when  $j \notin \mathcal{N}_i$ ) the algorithm achieves convergence to consensus. However, if the underlying calibration objective is to achieve absolute calibration of the sensors (i.e.,  $\bar{g}$  close to one and  $\bar{f}$  close to zero), this can be done by trying to exploit sensors that are *a priori* known to have good characteristics. In a large SN, this can be achieved in two ways: 1) if the large number of sensors are “good” sensors, then  $\gamma_{ij}$ -s in all neighborhoods  $\mathcal{N}_i$  should be approximately the same, or 2) if there is a set of *a priori* chosen good sensors  $j \in \mathcal{N}^f \subset \mathcal{N}$  the goal is to enforce their dominant influence to the rest of the nodes. There are two possibilities to achieve this: a) to set high values of  $\gamma_{ij}$  to all neighborhoods  $\mathcal{N}_i$  in which node  $i \in \mathcal{N}_j^{\text{out}}$ , or b) to set all  $\gamma_{jk}$ ,  $k = 1, \dots, n$ ,  $k \neq j$ , to be small (which prevents large changes of  $\hat{\phi}_j(t)$ ). Subsection 6.3 deals with the guidelines on weights tuning, while Subsection 6.4 treats the case in which a set of reference sensors is kept with fixed calibration parameters.

#### 4. Convergence Analysis

In this section we analyze the convergence properties of the calibration scheme presented in the previous section, where it has been assumed that both local sensor measurements and inter-node communications are perfect, i.e., possible communication errors and/or measurement errors are not present. We first analyze this basic scheme in order to focus on structural characteristics of the algorithm; the case of lossy SNs will be analyzed in the subsequent sections. In the basic setup, without presence of any unreliability, it is sufficient to assume that the step sizes  $\delta_i(t)$  are constant:

A1)  $\delta_i(t) = \delta = \text{const}$ , for all  $i = 1, \dots, n$ .

For clearer initial presentation of the convergence results, we now adopt a simplifying assumption:

A2)  $\{x(t)\}$  is independent and identically distributed (i.i.d.) sequence, with  $E\{x(t)\} = \bar{x} < \infty$  and  $E\{x(t)^2\} = s^2 < \infty$ .

In practice, when the SNs are used to measure certain physical quantities, the assumption that  $\{x(t)\}$  is i.i.d. is almost never satisfied; hence it will be relaxed later.

Based on A1) and A2), the expectation of the parameter estimates  $\bar{\phi}(t) = E\{\phi(t)\}$  satisfies the following recursion

$$\bar{\phi}(t+1) = (I + \delta \bar{B})\bar{\phi}(t), \quad (11)$$

where  $\bar{\phi}(0) = \phi(0)$ ,  $\bar{B} = \bar{\Omega}(\Gamma \otimes I_2)$  and  $\bar{\Omega} = E\{\Omega(t)\} = \text{diag}\{\bar{\Omega}_1 \dots \bar{\Omega}_n\}$ , with

$$\bar{\Omega}_i = \begin{bmatrix} \alpha_i \beta_i \bar{x} + \alpha_i^2 s^2 & \alpha_i \beta_i + \alpha_i^2 \bar{x} \\ (1 + \beta_i^2) \bar{x} + \alpha_i \beta_i s^2 & 1 + \beta_i^2 + \alpha_i \beta_i \bar{x} \end{bmatrix}. \quad (12)$$

The following assumption, typical for consensus-based algorithms, is introduced:

A3) Graph  $\mathcal{G}$  has a spanning tree.

It implies that the matrix  $\Gamma$  has one zero eigenvalue and the rest eigenvalues with negative real parts, e.g., [19]. Hence, from the structure of matrix  $\bar{B}$ , we directly conclude that it has at least two zero eigenvalues. Its remaining eigenvalues can be characterized starting from the following assumption:

A4)  $s^2 - \bar{x}^2 = \text{var}\{x(t)\} > 0$ .

This assumption guarantees that the estimation recursions are sufficiently excited by the signal  $x(t)$ . Its important consequence is that  $-\bar{\Omega}_i$  defined by (12) is Hurwitz, for all  $i = 1, \dots, n$ . Indeed, using some simple algebra it can be derived that  $-\bar{\Omega}_i$  is Hurwitz iff

$$\alpha_i^2(s^2 - \bar{x}^2) > 0, \quad 2\alpha_i \beta_i \bar{x} + \alpha_i^2 s^2 + 1 + \beta_i^2 > 0. \quad (13)$$

Both inequalities hold if and only if A4) holds. This greatly simplifies further derivations which depend on somewhat complicate expression (12) for the  $2 \times 2$  diagonal blocks of the matrix  $\bar{\Omega}$ .

Because of the block structure of matrices  $\bar{\Omega}$  and  $\bar{B}$ , the properties of the main recursion (11) cannot be analyzed using standard linear consensus methodologies (see, e.g., [19,39] and references therein). To cope with this problem, a methodology based on the concept of *diagonal quasi-dominance of matrices decomposed into blocks* has been used [14,18,20–22] to obtain the following important result characterizing all the eigenvalues of the matrix  $\bar{B}$ .

**Lemma 1** ([14,18]). Assume that the assumptions A3) and A4) hold. Then, matrix  $\bar{B}$  in (11) has two zero eigenvalues and the rest eigenvalues have negative real parts.

Observe that vectors  $i_1 = [1 \ 0 \ 1 \ 0 \ \dots \ 1 \ 0]^T \in \mathbb{R}^{2n}$  and  $i_2 = [0 \ 1 \ 0 \ 1 \ \dots \ 0 \ 1]^T \in \mathbb{R}^{2n}$ , where  $\mathbb{R}$  is the set of real numbers, are the right eigenvectors of  $\bar{B}$  corresponding to the eigenvalue at the origin. Let  $\rho_1$  and  $\rho_2$  be the corresponding normalized left eigenvectors, satisfying  $\begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} [i_1 \ i_2] = I_2$ . The following lemma deals with a similarity transformation important for all the remaining derivations throughout the paper.

**Lemma 2** ([14,18]). Let  $T = \begin{bmatrix} i_1 & i_2 \\ \vdots & \vdots \end{bmatrix} T_{2n \times (2n-2)}$ , where  $T_{2n \times (2n-2)}$  is an  $2n \times (2n-2)$  matrix, such that  $\text{span}\{T_{2n \times (2n-2)}\} = \text{span}\{\bar{B}\}$  ( $\text{span}\{A\}$  denotes a linear space spanned by the columns of matrix  $A$ ). Then,  $T$  is nonsingular and

$$T^{-1} \bar{B} T = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times (2n-2)} \\ 0_{(2n-2) \times 2} & \bar{B}^* \end{bmatrix}, \quad (14)$$

where  $\bar{B}^*$  is Hurwitz, and  $0_{i \times j}$  denotes a  $i \times j$  zero matrix.

Notice that

$$T^{-1} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ S_{(2n-2) \times 2n} \end{bmatrix}, \quad (15)$$

where  $S_{(2n-2) \times 2n}$  can be determined from the definition of  $T$ .

From the structure of the matrices in (11), it can be concluded that the transformation  $T$  from Lemma 2, when applied to the original matrix  $B(t)$ , will produce a matrix which has the same structure as the transformed matrix given in equation (14).

**Lemma 3** ([14,18]). *For the matrix  $B(t)$  in (10) it holds that, for all  $t$ ,*

$$T^{-1}B(t)T = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times (2n-2)} \\ 0_{(2n-2) \times 2} & B(t)^* \end{bmatrix}, \quad (16)$$

where  $B(t)^*$  is an  $(2n-2) \times (2n-2)$  matrix and  $T$  is given in Lemma 2.

The following convergence theorem can now be formulated.

**Theorem 1** ([18]). *Assume that Assumptions A1)–A4) hold. Then there exists  $\delta' > 0$  such that for all  $\delta \leq \delta'$  in (10)*

$$\lim_{t \rightarrow \infty} \hat{\phi}(t) = (i_1 \rho_1 + i_2 \rho_2) \hat{\phi}(0) \quad (17)$$

in the mean square sense and w.p.1.

Note here that the limit vector in (17)  $(i_1 \rho_1 + i_2 \rho_2) \hat{\phi}(0)$  have all the odd elements equal, and all the even elements equal, which means that the corrected gains of all the nodes converge to the same value, and the corrected offsets of all the nodes converge to the same value. It can be shown [14] that this value only depends on the unknown sensor parameters  $\alpha_i$  and  $\beta_i$ , and the weights  $\gamma_{ij}$  in  $J_i$ ,  $i, j = 1, \dots, n$ . For given initial conditions in (5),  $\rho_1 \hat{\phi}(0)$  and  $\rho_2 \hat{\phi}(0)$  are in the form of weighted sums of  $\alpha_i$  and  $\beta_i$ ,  $1, \dots, n$ , respectively. Assuming that the weights  $\gamma_{ij}$  are the same for all the nodes, and that  $\alpha_i$  have a distribution centered around one, and  $\beta_i$  around zero, these weighted sums will be close to one and zero, respectively.

The value of  $\delta' > 0$  in Theorem 1, which ensures convergence, may be restrictive. In practice, the choice of step size  $\delta$  in A1) should be based on the actual properties of the underlying SN; its value needs to be small enough to achieve convergence, but it should also be sufficiently large to achieve acceptable rate of convergence (as in the standard parameter estimation recursions [40]).

After clarifying the main structural properties of the algorithm, we now extend the analysis to the more realistic case of correlated sequences  $\{x(t)\}$ . We replace A2) with:

A2') The random process  $\{x(t)\}$  is weakly stationary, bounded w.p.1, and with bounded first and second moments, i.e.,  $|x(t)| \leq K < \infty$ ,  $E\{x(t)\} = \bar{x} < \infty$ ,  $E\{x(t-d)x(t)\} = m(d) < \infty$  for all  $d \in \{0, 1, 2, \dots\}$  ( $E\{\cdot\}$  is a sign of the mathematical expectation),  $m(0) = s^2 < \infty$ . It also holds that (w.p.1.)

$$a) \quad |E\{x(t)|\mathcal{F}_{t-\tau}\} - \bar{x}| = o(1), \quad (\text{w.p.1}) \quad (18)$$

$$b) \quad |E\{x(t-d)x(t)|\mathcal{F}_{t-\tau}\} - m(d)| = o(1), \quad (\text{w.p.1}) \quad (19)$$

when  $\tau \rightarrow \infty$ , for all  $d \in \{0, 1, 2, \dots\}$ ,  $\tau > d$  ( $\mathcal{F}_{t-\tau}$  denotes the minimal  $\sigma$ -algebra generated by  $\{x(0), x(1), \dots, x(t-\tau)\}$ , and  $o(1)$  denotes a function that converges to zero when  $\tau \rightarrow \infty$ ).

Hence, A2') requires stationarity, boundedness, and imposes a mixing condition on the signal  $\{x(t)\}$ . The explicitly used time shift parameter  $d$  will be used later for introducing a new algorithm based on an instrumental variable, capable of dealing with possible measurement noise.



The following theorem examines the convergence of the algorithm (11) under assumption A2'):

**Theorem 2** ([17]). Assume that the assumptions A1), A2'), A3) and A4) hold. Then there exists  $\delta'' > 0$  such that for all  $\delta \leq \delta''$  in (10)  $\lim_{t \rightarrow \infty} \hat{\phi}(t) = (i_1 \rho_1 + i_2 \rho_2) \hat{\phi}(0)$  in the mean square sense and w.p.1.

## 5. Extensions of the Main Algorithm

In this section, we introduce several modifications and generalizations of the main algorithm (5), so that it is possible to achieve distributed calibration under more challenging conditions, typically present in real-life SNs: communication dropouts, additive communication noise, measurement noise, and asynchronous communication. Convergence analyzes of the introduced modifications is presented in details.

### 5.1. Communication Errors

In this subsection, we assume that inter-node communication errors can be manifested in two ways: 1) communication dropouts (outages) and 2) additive communication noise. Communication dropouts typically occur in SNs using digital communication; additive noise can, in this case, model quantization effects. For example, in the case of smart city sensor networks, depicted in Fig. 1, the dropouts will happen relatively often because of the dynamic environment, where both physical obstacles and electronic interference can be persistent. In certain, less frequent practical situations, SNs can use analog communication (e.g., when certain types of energy harvesting are used [41]), when additive communication noise is dominant, and dropouts appear less frequently.

The communication errors are formally introduced using the following assumptions:

A5) The weights  $\gamma_{ij}$  in the algorithm (5) are now randomly time-varying, according to stochastic processes given by  $\{\gamma_{ij}(t)\} = \{u_{ij}(t)\gamma_{ij}\}$ , where  $\{u_{ij}(t)\}$  are i.i.d. binary random sequences, such that  $u_{ij}(t) = 1$  with probability  $p_{ij}$  ( $p_{ij} > 0$  when  $j \in \mathcal{N}_i$ ), and  $u_{ij}(t) = 0$  with probability  $1 - p_{ij}$ .

A6) Instead of receiving  $\hat{z}_j(t)$  from the  $j$ -th node, the  $i$ -th node receives  $\hat{z}_j(t) + \xi_{ij}(t)$ , where  $\{\xi_{ij}(t)\}$  is an i.i.d. random sequence with  $E\{\xi_{ij}(t)\} = 0$  and  $E\{\xi_{ij}(t)^2\} = (\sigma_{ij}^\xi)^2 < \infty$ .

A7) Processes  $\{x(t)\}$ ,  $\{u_{ij}(t)\}$  and  $\{\xi_{ij}(t)\}$  are mutually independent.

Based on the above assumptions, the communication dropout at any iteration  $t$ , when node  $j$  is sending to node  $i$ , will happen with probability  $1 - p_{ij}$ , independently of the additive communication noise process  $\{\xi_{ij}(t)\}$  and the measured signal  $\{x(t)\}$ .

Denoting

$$v_i(t) = \sum_{j \in \mathcal{N}_i} \gamma_{ij}(t) \xi_{ij}(t) \begin{bmatrix} \alpha_i y_i(t) \\ 1 + \beta_i y_i(t) \end{bmatrix},$$

and  $v(t) = [v_1(t) \dots v_n(t)]$ , one obtains from (10) that

$$\hat{\phi}(t+1) = [I + (\Delta(t) \otimes I_2) B'(t)] \hat{\phi}(t) + \Delta(t) v(t), \quad (20)$$

where  $B'(t) = \Omega(t)(\Gamma(t) \otimes I_2)$ , and  $\Gamma(t)$  is obtained from  $\Gamma$  by applying A5).

Convergence analyzes of the recursion (20), under the additional assumptions A5)–A7), is based on the results of the previous subsection. Due to the mutual independence of the random variables in  $B'(t)$ , it can be concluded that  $E\{B'(t)\} = \bar{B}' = \bar{\Omega}(\bar{\Gamma} \otimes I_2)$ , where  $\bar{\Gamma} = E\{\Gamma(t)\}$  is the same as  $\Gamma$  but with  $\gamma_{ij}$  replaced by  $\gamma_{ij} p_{ij}$ . Also, it follows that  $\tilde{B}'(t) \doteq B'(t) - \bar{B}'$ , is a martingale difference sequence (since  $E\{\tilde{B}'(t) | \mathcal{F}_{t-1}\} = 0$ ). Furthermore, it can be concluded that  $\bar{B}' = \bar{\Omega}(\bar{\Gamma} \otimes I_2)$  has the same spectrum as  $\bar{B}$  in (11): it has two zero eigenvalues and the rest eigenvalues are with negative real part.

Since the additive noise is now present in the recursions (20), A1) needs to be replaced with the following assumption, typical in the stochastic approximation literature (e.g., [42]):

A1')  $\delta_i(t) = \delta(t) > 0$ ,  $\sum_{t=0}^{\infty} \delta(t) = \infty$ ,  $\sum_{t=0}^{\infty} \delta(t)^2 < \infty$ ,  $i = 1, \dots, n$ .

Intuitively, A1') introduces diminishing gains  $\delta_i(t)$  which converge to zero slowly enough, so that the additive noise can be averaged out while asymptotic convergence to a consensus point is achieved (despite the presence of noise).

Therefore, we have

$$\hat{\phi}(t+1) = (I + \delta(t)\bar{B}')\hat{\phi}(t) + \delta(t)\tilde{B}'(t)\hat{\phi}(t) + \delta(t)v(t). \quad (21)$$

Similarly as in the noiseless case, let us introduce the similarity transformation

$$T' = \begin{bmatrix} i_1 & i_2 & T'_{2n \times (2n-2)} \end{bmatrix},$$

where  $T'_{2n \times (2n-2)}$  is an  $2n \times (2n-2)$  matrix, such that  $\text{span}\{T'_{2n \times (2n-2)}\} = \text{span}\{\bar{B}'\}$ . Then,  $(T')^{-1} =$

$$\begin{bmatrix} \rho'_1 \\ \rho'_2 \\ S'_{(2n-2) \times 2n} \end{bmatrix}, \text{ where } \rho'_1 \text{ and } \rho'_2 \text{ are the left eigenvectors of } \bar{B}' \text{ corresponding to the eigenvalue at the}$$

origin. By applying transformation  $T'$  to (21), and using stochastic Lyapunov stability arguments, along with the arguments typically used in analyzing stochastic approximation algorithms [14,18,43,44], the following theorem can be proved:

**Theorem 3** ([14,18]). *Let Assumptions A1'), A2)–A7) be satisfied. Then,  $\hat{\phi}(t)$  generated by (21) converges to  $i_1 w_1 + i_2 w_2$  in the mean square sense and w.p.1, where  $w_1$  and  $w_2$  are scalar random variables satisfying  $E\{w_1\} = \rho'_1 \hat{\phi}(0)$  and  $E\{w_2\} = \rho'_2 \hat{\phi}(0)$ .*

The theorem essentially states that, again, all the corrected drifts converge to the same point, and all the corrected offsets converge to the same point; however, because of the additive communication noise, these points are *random* and depend on the noise realization. The *mean* values of these possible convergence points depend on the sensor parameters  $\alpha_i$  and  $\beta_i$ , the design parameters  $\gamma_{ij}$ , as well as on the dropout probabilities  $p_{ij}$ ,  $i, j = 1, \dots, n$ .

## 5.2. Measurement Noise

In this subsection we, additionally to communication errors, assume that the signal  $x(t)$  is measured with additive measurement noise. This situation is of essential importance for practical applications since practically all the existing sensors contain certain measurement errors which are typically modeled using stochastic processes [3].

Formally, we model the additive noise stochastic process using the following assumption:

A8) Instead of  $y_i(t)$  given by (1), the sensor measurements are now contaminated by noise, and given by

$$y_i^\eta(t) = \alpha_i x(t) + \beta_i + \eta_i(t),$$

where  $\{\eta_i(t)\}$ ,  $i = 1, \dots, n$ , are zero mean i.i.d. random sequences with  $E\{\eta_i(t)^2\} = (\sigma_i^\eta)^2$ , independent of the measured signal  $x(t)$ .

By replacing  $y_i^\eta(t)$  instead of  $y_i(t)$  in the base algorithm (5), one obtains the following “noisy” version of (8):

$$\hat{\phi}_i(t+1) = \hat{\phi}_i(t) + \delta_i(t) \sum_{j \in \mathcal{N}_i} \gamma_{ij} \{ [\Omega_i(t) + \Psi_i(t)] [\hat{\phi}_j(t) - \hat{\phi}_i(t)] + N_{ij}(t) \hat{\phi}_j(t) - N_{ii}(t) \hat{\phi}_i(t) \}, \quad (22)$$

where  $\Psi_i(t) = \eta_i(t) \begin{bmatrix} \alpha_i x(t) & \alpha_i \\ \beta_i x(t) & \beta_i \end{bmatrix}$ ,  $N_{ij}(t) = \frac{\eta_j(t)}{\alpha_j} \begin{bmatrix} \alpha_i y_i(t) & 0 \\ \beta_i y_i(t) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_j(t)\eta_i(t)}{\alpha_j} & 0 \\ 0 & 0 \end{bmatrix}$  and  $N_{ii}(t) = \frac{\eta_i(t)}{\alpha_i} \begin{bmatrix} \alpha_i y_i(t) & 0 \\ \beta_i y_i(t) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_i(t)^2}{\alpha_i} & 0 \\ 0 & 0 \end{bmatrix}$ , assuming  $\alpha_i \neq 0$ ,  $i = 1, \dots, n$ . It is important to observe here that  $E\{\Psi_i(t)\} = 0$ ,  $E\{N_{ij}(t)\} = 0$ ; however  $E\{N_{ii}(t)\} = \begin{bmatrix} \frac{(\sigma_i^\eta)^2}{\alpha_i} & 0 \\ 0 & 0 \end{bmatrix}$ .

Assuming again that the step sizes  $\delta_i(t)$ ,  $i = 1, \dots, n$ , satisfy A1'), one can obtain the following equation analog to (10):

$$\hat{\phi}(t+1) = (I + \delta(t)\{[\Omega(t) + \Psi(t)](\Gamma \otimes I_2) + \tilde{N}(t)\})\hat{\phi}(t), \quad (23)$$

where  $\Psi(t) = \text{diag}\{\Psi_1(t), \dots, \Psi_n(t)\}$  and  $\tilde{N}(t) = [\tilde{N}_{ij}(t)]$  with  $\tilde{N}_{ij}(t) = -\sum_{k,k \neq i} \gamma_{ik} N_{ii}(t)$  for  $i = j$  and  $\tilde{N}_{ij}(t) = \gamma_{ij} N_{ij}(t)$  for  $i \neq j$ ,  $i, j = 1, \dots, n$ .

In an analogous way as in the previous section, instead of (11), the following equation is obtained for the mean of the corrected calibration parameters

$$\bar{\phi}(t+1) = [I + \delta(t)(\bar{B} + \Sigma_\eta)]\bar{\phi}(t), \quad (24)$$

where  $\bar{B}$  is as in (11) and  $\Sigma_\eta = -\text{diag}\{\frac{(\sigma_1^\eta)^2}{\alpha_1} \sum_j \gamma_{1j}, 0, \dots, \frac{(\sigma_n^\eta)^2}{\alpha_n} \sum_j \gamma_{nj}, 0\}$ . Because of the additional term  $\Sigma_\eta$ , the sums of the rows of the matrix  $\bar{B} + \Sigma_\eta$  are not equal to zero anymore, so that the convergence to consensus (as in Theorem 1) cannot be achieved in this case.

However, it can be seen from the structure of the recursion (24) that, if we assume that the measurement noise variances  $(\sigma_i^\eta)^2$  are *a priori* known, we can use them to modify the basic algorithm (5) in the following way, ensuring again the asymptotic convergence to consensus:

$$\hat{\theta}_i(t+1) = \hat{\theta}_i(t) + \delta(t)\left\{ \sum_{j \in \mathcal{N}_i} \gamma_{ij} \epsilon_{ij}^\eta(t) \begin{bmatrix} y_i^\eta(t) \\ 1 \end{bmatrix} + \begin{bmatrix} (\sigma_i^\eta)^2 \sum_{j \in \mathcal{N}_i} \gamma_{ij} & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \hat{\theta}_i(t) \right\}, \quad (25)$$

where  $\epsilon_{ij}^\eta(t) = \hat{z}_j^\eta(t) - \hat{z}_i^\eta(t)$  and  $\hat{z}_i^\eta(t) = \hat{a}_i(t)y_i^\eta(t) + \hat{b}_i(t)$ ,  $i = 1, \dots, n$ .

The following theorem deals with the convergence of the above modification of the main algorithm, when the measurement noise is present together with the communication errors. The convergence points will again depend on the measurement and communication noise realizations, in a similar way as in Theorem 3.

**Theorem 4 ([18]).** Assume that the assumptions A1'), A2)–A8) hold. Then,  $\hat{\phi}(t)$ , given by (25), converges to  $i_1 w_1 + i_2 w_2$  in the mean square sense and w.p.1, where  $w_1$  and  $w_2$  are scalar random variables satisfying  $E\{w_1\} = \rho'_1 \hat{\phi}(0)$  and  $E\{w_2\} = \rho'_2 \hat{\phi}(0)$ .

Notice that the above theorem was based on assumption A2): indeed, when both  $\{x(t)\}$  and  $\{\eta_i(t)\}$  are i.i.d. sequences, it is not surprising that the asymptotic consensus is achievable only provided  $\sigma_i^\eta$ ,  $i = 1, \dots, n$ , are known. However, we can replace the unrealistic assumption A2) with A2') (introduced in Section 4 in the noiseless case) allowing correlated sequences  $\{x(t)\}$  which is almost always case in practice. In such a way, the correlatedness problem present in the algorithm (24) can be overcome, without requiring any *a priori* information about the measurement noise process. The idea is to introduce *instrumental variables* in the main algorithm in the way analogous to the one often used in the field of recursive methods for system identification, e.g., [45,46]. Instrumental variables

have the basic property of being correlated with the measured signal, and uncorrelated with noise. If  $\{\zeta_i(t)\}$  is the instrumental variable sequence of the  $i$ -th agent, one has to ensure that  $\zeta_i(t)$  is correlated with  $x(t)$  and uncorrelated with  $\eta_j(t)$ ,  $j = 1, \dots, n$ . Under A2') a logical choice is to take the delayed sample of the measured signal as an instrumental variable, i.e., to take  $\zeta_i(t) = y_i^\eta(t-d)$ , where  $d \geq 1$ . Consequently, we present the following general *calibration algorithm based on instrumental variables* able to cope with measurement noise:

$$\hat{\theta}_i(t+1) = \hat{\theta}_i(t) + \delta(t) \sum_{j \in \mathcal{N}_i} \gamma_{ij} \epsilon_{ij}^\eta(t) \begin{bmatrix} y_i^\eta(t-d) \\ 1 \end{bmatrix}, \quad (26)$$

where  $d \geq 1$  and  $\epsilon_{ij}^\eta(t) = \hat{z}_j^\eta(t) - \hat{z}_i^\eta(t)$ ,  $\hat{z}_i^\eta(t) = \hat{a}_i(t)y_i^\eta(t) + \hat{b}_i(t)$ ,  $i = 1, \dots, n$ . Following the derivations from Section 3, one obtains from (26) the following relations involving explicitly  $x(t)$  and the noise terms:

$$\begin{aligned} \hat{\phi}_i(t+1) = \hat{\phi}_i(t) + \delta(t) \sum_{j \in \mathcal{N}_i} \gamma_{ij} \{ (\Omega_i(t,d) + \Psi_i(t,d)) (\hat{\phi}_j(t) - \hat{\phi}_i(t)) \\ + N_{ij}(t,d) \hat{\phi}_j(t) - N_{ii}(t,d) \hat{\phi}_i(t) \}, \end{aligned} \quad (27)$$

where

$$\Omega_i(t,d) = \begin{bmatrix} \alpha_i \beta_i x(t) + \alpha_i^2 x(t)x(t-d) & \alpha_i \beta_i + \alpha_i^2 x(t-d) \\ (1 + \beta_i^2)x(t) + \alpha_i \beta_i x(t)x(t-d) & 1 + \beta_i^2 + \alpha_i \beta_i x(t-d) \end{bmatrix},$$

$$\Psi_i(t,d) = \eta_i(t-d) \begin{bmatrix} \alpha_i x(t) & \alpha_i \\ \beta_i x(t) & \beta_i \end{bmatrix},$$

$$N_{ij}(t,d) = \frac{\eta_j(t)}{\alpha_j} \begin{bmatrix} \alpha_i y_i(t-d) & 0 \\ \beta_i y_i(t-d) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_j(t)\eta_i(t-d)}{\alpha_j} & 0 \\ 0 & 0 \end{bmatrix}$$

and

$$N_{ii}(t,d) = \frac{\eta_i(t)}{\alpha_i} \begin{bmatrix} \alpha_i y_i(t-d) & 0 \\ \beta_i y_i(t-d) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_i(t)\eta_i(t-d)}{\alpha_i} & 0 \\ 0 & 0 \end{bmatrix}.$$

In the same way as in (23), we have

$$\hat{\phi}(t+1) = (I + \delta(t) \{ [\Omega(t,d) + \Psi(t,d)] (\Gamma \otimes I_2) + \tilde{N}(t,d) \}) \hat{\phi}(t), \quad (28)$$

where  $\Omega(t,d) = \text{diag}\{\Omega_1(t,d), \dots, \Omega_n(t,d)\}$ ,  $\Psi(t,d) = \text{diag}\{\Psi_1(t,d), \dots, \Psi_n(t,d)\}$ ,  $\tilde{N}(t,d) = [\tilde{N}_{ij}(t,d)]$ , where  $\tilde{N}_{ij}(t,d) = -\sum_{k,k \neq i} \gamma_{ik} N_{ik}(t,d)$  for  $i = j$  and  $\tilde{N}_{ij}(t,d) = \gamma_{ij} N_{ij}(t,d)$  for  $i \neq j$ ,  $i, j = 1, \dots, n$ .

In order to formulate a convergence theorem for (28), the following modification of A4) is needed: A4')  $m(d) > \bar{x}^2$  for some  $d = d_0 \geq 1$ .

This assumption implies that the correlation  $m(d_0)$  should be large enough. Similarly as in the case of A4), it can be concluded that A4') implies that  $-\bar{\Omega}(d) = -E\{\Omega_i(t,d)\}$  is Hurwitz. Similarly as in the above cases, let us introduce the similarity transformation

$$T'' = \begin{bmatrix} i_1 & \vdots & i_2 & \vdots & T''_{2n \times (2n-2)} \end{bmatrix},$$

where  $T''_{2n \times (2n-2)}$  is an  $2n \times (2n - 2)$  matrix, such that  $\text{span}\{T''_{2n \times (2n-2)}\} = \text{span}\{\bar{B}(d)''\}$ . Then,  $(T'')^{-1} = \begin{bmatrix} \rho_1'' \\ \rho_2'' \\ S''_{(2n-2) \times 2n} \end{bmatrix}$ , where  $\rho_1''$  and  $\rho_2''$  are the left eigenvectors of  $\bar{B}(d)'' = E\{\Omega(t, d)(\Gamma(t) \otimes I_2)\} = \bar{\Omega}(d)(\bar{\Gamma} \otimes I_2)$  corresponding to the zero eigenvalue. The following theorem deals with the convergence of the instrumental variable algorithm (26). The convergence point, again, depends on the noise realization.

**Theorem 5 ([17]).** Assume that the assumptions  $A1')$ ,  $A2')$ ,  $A3)$ ,  $A4')$ ,  $A5)$ – $A8)$  hold. Then  $\hat{\phi}(t)$ , given by (28) with  $d = d_0$ , converges to  $i_1 w_1 + i_2 w_2$  in the mean square sense and w.p.1, where  $w_1$  and  $w_2$  are scalar random variables satisfying  $E\{w_1\} = \rho_1'' \hat{\phi}(0)$  and  $E\{w_2\} = \rho_2'' \hat{\phi}(0)$ .

### 5.3. Asynchronous Broadcast Gossip Communication

So far we have shown how to deal with most of the practical challenges which emerge when dealing with real life SNs, such as communication dropouts, communication additive noise and measurement noise. However, in all of the above discussed algorithms we have implicitly assumed that all the nodes in the network share a common clock, based on which the recursions in (5), (25) or (26) can be implemented synchronously. Indeed, when introducing the main algorithm we have assumed that the signal  $x(t)$  is being measured in discrete-time instances  $t$  by all the nodes. These instances are also used as time indexes of synchronous recursions of the above algorithms. Yet, there are many practical cases of SNs for which it is impossible or impractical to function synchronously. A typical example is the case when the nodes follow certain sleeping policies in order to minimize power consumption (e.g., [3]). For example, the nodes in SN shown in Fig. 1, measuring air pollution or atmospheric conditions, may be programmed to make measurements less often during periods in which there is less traffic in the city. These types of situations are rigorously treated in the rest of this subsection.

Instead of the problem setup introduced in Section 3, assume now that the sensors are measuring a continuous-time signal  $x(t)$  at discrete points  $t_k$ ,  $t_k \in \mathfrak{R}^+$ ,  $k = 1, 2, \dots$ ,  $t_{k+1} > t_k$ , producing the sensor outputs

$$y_i(t_k) = \alpha_i x(t_k) + \beta_i + \eta_i(t_k), \quad (29)$$

where the  $\alpha_i$  and  $\beta_i$  are the same unknown parameters as in the previous subsections, and we also assume that the measurement noise  $\eta_i(t_k)$ ,  $i = 1, \dots, n$ , is present in the sensor readings.

Furthermore, since the goal is to remove dependence on a common global clock, it is now assumed that every node  $j \in \mathcal{N}$  has its own local clock. For the sake of compact notation and simpler derivations, a single clock, called global virtual clock, is introduced, which ticks when any of the local clocks ticks. Hence,  $t_k$  in (29) can be considered as the time in which the  $k$ -th tick of the virtual clock happens. To have a well defined situation, it is formally assumed that the ticks of the local clocks are independent, and that the intervals between any two consecutive ticks are finite w.p.1. It is also assumed, for the sake of simpler derivations, that the unconditional probability that the  $j$ -th clock ticked at an instance  $t_k$  is  $q_j > 0$ , independently of  $k$ . It is easy to verify that these conditions are satisfied for a typical model used in SNs, where it is assumed that the local clocks tick according to independent Poisson processes with rates  $\mu_j$  (as in, e.g., [47,48]). This case will be adopted throughout this subsection. It directly follows that, in this case, the virtual global clock ticks according to a Poisson process with the rate  $\sum_{j=1}^n \mu_j$ .

According to the above assumptions, let us denote with  $t_l^j$  the ticks of the local clock  $j$ ,  $l = 1, 2, \dots$ . The communication protocol can then be defined in the following way. At each local clock tick, a node  $j$  makes the local sensor measurement, calculates the corrected sensor output  $z_j(t_l^j)$  (based on the current estimates of calibration parameters  $a_j$  and  $b_j$ ), and broadcasts it to its out-neighbors  $i \in \mathcal{N}_j^{\text{out}}$ .



We assume also that communication dropouts can happen, i.e., each node  $i \in \mathcal{N}_j^{\text{out}}$  receives the transmitted message with probability  $p_{ij} > 0$ . For the sake of clarity of presentation, we do not treat additive communication noise in this subsection. It is also assumed that the communication delay is negligible, so that, practically at the same time instant all the nodes which have received the broadcast, perform the local sensor reading, calculate their corrected outputs  $z_i(t_l^j)$ , and update the local estimates of their calibration parameters  $a_i$  and  $b_i$ . This procedure is repeated for any local clock tick. The index of the node whose clock has ticked at instant  $t_k$  is denoted by  $j(k)$ , and let  $J(k)$  be the subset of the out-neighbors  $i \in \mathcal{N}_{j(k)}^{\text{out}}$  which have received the broadcast message. Also, let  $x(k) = x(t_k) = x(t_l^{j(k)}), y_i(k) = y_i(t_k) = y_i(t_l^{j(k)}), y_j(k) = y_j(t_k) = y_j(t_l^{j(k)}), z_i(k) = z_i(t_k) = z_i(t_l^{j(k)}), z_j(k) = z_j(t_k) = z_j(t_l^{j(k)}), \eta_i(k) = \eta_i(t_k) = \eta_i(t_l^{j(k)})$  and  $\eta_j(k) = \eta_j(t_k) = \eta_j(t_l^{j(k)})$  for some  $l$ .

The measurement noise is treated as in the previous subsection, by using the delayed measurement  $y_i(d_i(k))$  as the instrumental variable

$$\zeta_i(k) = y_i(d_i(k)), \quad (30)$$

where  $d_i(k)$  is the global iteration number that corresponds to the closest past measurement of the node  $i$ . By using the same local criteria as in (3) and gradients as in (4), the following new recursion for updating the calibration parameters at node  $i$  is formulated:

$$\hat{\theta}_i(k) = \hat{\theta}_i(k-1) + \delta_i(k) \gamma_{i,j(k)} \epsilon_{i,j(k)}(k) \begin{bmatrix} y_i(d_i(k)) \\ 1 \end{bmatrix}, \quad (31)$$

where:

- $\hat{\theta}_i(k) = [\hat{a}_i(k) \ \hat{b}_i(k)]^T$ ,
- $\delta_i(k)$  is the step size given by  $\delta_i(k) = v_i(k)^{-c}$ , where  $v_i(k) = \sum_{m=1}^k I\{i \in J(m)\}$  is the number of parameter updates of node  $i$  up to the iteration  $k$ , with  $1/2 < c \leq 1$  ( $I\{\cdot\}$  denotes the indicator function),
- $\epsilon_{i,j(k)}(k) = \hat{z}_{j(k)}(k) - \hat{z}_i(k)$ , where

$$\hat{z}_{j(k)}(k) = \hat{a}_{j(k)}(k-1) y_{j(k)}(k) + \hat{b}_{j(k)}(k-1), \quad (32)$$

$$\hat{z}_i(k) = \hat{a}_i(k-1) y_i(k) + \hat{b}_i(k-1) \quad (33)$$

are the corrected outputs of node  $j(k)$  and node  $i$ .

The initial conditions are adopted to be  $\hat{\theta}_i(0) = [1 \ 0]^T$ . Note that, according to the problem setup, at a given iteration  $k$  only the nodes  $i \in J(k)$  perform the above parameters update; for the rest of the nodes it holds that  $\hat{\theta}_i(k) = \hat{\theta}_i(k-1)$ .

Computationally, the algorithm is as simple as the basic one, requiring only a few additions and multiplications in one iteration. Information needed at node  $i$  are: the local sensor measurement, the local instrumental variable, and the current output sent by an in-neighbor  $j$ . Knowledge of the global iteration index  $k$  (or  $d_i(k)$ ) is not needed.

From the above definition of the step size  $\delta_i(k)$  it can be concluded that it depends only on the number of local clock ticks, which makes the algorithm completely decentralized.

It should also be noticed that the instrumental variables in (31) can be selected in several ways. For example, instead of choosing (30), it can be practical to choose  $\zeta_i(k) = y_i(\bar{t}_l^{j,i})$ , where  $\bar{t}_l^{j,i}$  is the time instant of a supplementary measurement of node  $i$ , just after the last step of the recursion (31) has been locally performed. This scheme is not assumed in the below analyzed setting, because of much more complicated notation; all the results can be easily transferred to this case.

Similarly as in the synchronous case, we introduce:

$$\hat{\phi}_i(k) = \begin{bmatrix} \hat{g}_i(k) \\ \hat{f}_i(k) \end{bmatrix} = \begin{bmatrix} \alpha_i & 0 \\ \beta_i & 1 \end{bmatrix} \hat{\theta}_i(k), \quad (34)$$

and

$$\epsilon_{i,j(k)}(k) = \begin{bmatrix} x(k) & 1 \end{bmatrix} (\hat{\phi}_{j(k)}(k) - \hat{\phi}_i(k)) + \hat{a}_{j(k)}(k)\eta_{j(k)}(k) - \hat{a}_i(k)\eta_i(k). \quad (35)$$

Consequently, we have

$$\begin{aligned} \hat{\phi}_i(k) = & \hat{\phi}_i(k-1) + \delta_i(k)\gamma_{i,j(k)}\{(\Omega_i(k) + \Psi_i(k))(\hat{\phi}_{j(k)}(k-1) - \hat{\phi}_i(k-1)) \\ & + N_{i,j(k)}(k)\hat{\phi}_{j(k)}(k-1) - N_{ii}(k)\hat{\phi}_i(k-1)\}, \end{aligned} \quad (36)$$

where

$$\Omega_i(k) = \begin{bmatrix} \alpha_i\beta_i x(k) + \alpha_i^2 x(k)x(d_i(k)) & \alpha_i\beta_i + \alpha_i^2 x(d_i(k)) \\ (1 + \beta_i^2)x(k) + \alpha_i\beta_i x(k)x(d_i(k)) & 1 + \beta_i^2 + \alpha_i\beta_i x(d_i(k)) \end{bmatrix}$$

$$\Psi_i(k) = \eta_i(d_i(k)) \begin{bmatrix} \alpha_i x(k) & \alpha_i \\ \beta_i x(k) & \beta_i \end{bmatrix},$$

$$N_{i,j(k)}(k) = \frac{\eta_{j(k)}(k)}{\alpha_{j(k)}} \begin{bmatrix} \alpha_i y_i^0(d_i(k)) & 0 \\ \beta_i y_i^0(d_i(k)) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_{j(k)}(k)\eta_i(d_i(k))}{\alpha_{j(k)}} & 0 \\ 0 & 0 \end{bmatrix}$$

and

$$N_{ii}(k) = \frac{\eta_i(k)}{\alpha_i} \begin{bmatrix} \alpha_i y_i^0(d_i(k)) & 0 \\ \beta_i y_i^0(d_i(k)) & 0 \end{bmatrix} + \begin{bmatrix} \frac{\eta_i(k)\eta_i(d_i(k))}{\alpha_i} & 0 \\ 0 & 0 \end{bmatrix},$$

where  $y_i^0(k) = \alpha_i x(k) + \beta_i$ , with the initial conditions  $\hat{\phi}_i(0) = [\alpha_i \ \beta_i]^T$ ,  $i \in J(k)$ .

Recursions (36) for  $i = 1, \dots, n$ , can be written compactly as

$$\hat{\phi}(k) = \{I + [\Omega(k) + \Psi(k)](\Delta(k)\Gamma(k) \otimes I_2) + (\Delta(k) \otimes I_2)\tilde{N}(k)\}\hat{\phi}(k-1), \quad (37)$$

where:

- $\hat{\phi}(k) = [\hat{\phi}_1(k)^T \dots \hat{\phi}_n(k)^T]^T$ ,
- $\Delta(k) = \text{diag}\{\delta_1(k), \dots, \delta_n(k)\}$ ,
- $\Omega(k) = \text{diag}\{\Omega_1(k), \dots, \Omega_n(k)\}$ ,
- $\Gamma(k) = [\Gamma(k)_{lm}]$ , with  $\Gamma(k)_{ll} = -\gamma_{l,j(k)}$  and  $\Gamma(k)_{l,j(k)} = \gamma_{l,j(k)}$  for all  $l \in J(k)$ ,  $\Gamma(k)_{lm} = 0$ , otherwise,
- $\Psi(k) = \text{diag}\{\Psi_1(k), \dots, \Psi_n(k)\}$ ,
- $\tilde{N}(k) = [\tilde{N}_{lm}(k)]$ , where  $\tilde{N}_{ll}(k) = -\gamma_{l,j(k)}N_{ll}(k)$  and  $\tilde{N}_{l,j(k)}(k) = \gamma_{l,j(k)}N_{l,j(k)}(k)$ , for all  $l \in J(k)$ ,  $\tilde{N}(k)_{lm} = 0$ , otherwise.

The initial condition is  $\hat{\phi}(0) = [\hat{\phi}_1(0)^T \dots \hat{\phi}_n(0)^T]^T = [[\alpha_1 \ \beta_1]^T \dots [\alpha_n \ \beta_n]^T]^T$ .

Since we have formulated a slightly different problem setup than in the Section 3, we introduce a new set of assumptions, and denote them using letter B:

B1)  $\{x(k)\}$  is a stationary random sequence, bounded w.p.1, and satisfying the  $\phi$ -mixing condition.

B2) Let  $\{t^{i,l}\}$ ,  $l = 1, 2, \dots$  represent time instants in which node  $i$  performs measurements. Then,  $\min_i \bar{r}_i > m^2$ , where  $m = E\{x(k)\}$  and  $\bar{r}_i = E\{x(t^{i,l})x(t^{i,l-1})\}$ ,  $i = 1, \dots, n$ .

B3) Graph  $\mathcal{G}$  has a spanning tree.

B4)  $\{\eta_i(k)\}$ ,  $i = 1, \dots, n$ , are zero-mean sequences of independent and bounded w.p.1 random variables.  $\{\eta_i(k)\}$  is independent of the process  $\{x(t)\}$ , with  $E\{\eta_i(k)^2\} = (\sigma_i^\eta)^2$  for all  $k$ .

Assumptions B3) and B4) are essentially the same as A3) and A8).

The  $\phi$ -mixing condition B1) represents one of the strong mixing conditions, usually satisfied for sensory signals [49–51].

Assumption B2) represents an extension of the assumption A4), adapted to the presence of the instrumental variable  $y_i(d_i(k))$  in (31). It guarantees the persistence of excitation in the sense that the variance of  $x(k)$  must be greater than zero (for all  $k$ , because of stationarity) so that constant signals are not allowed [14,40]. However, it also ensures sufficient correlation between the instrumental variable and the current measurement, so that e.g., white noise signals are also not allowed. It can be easily derived [15] that B2) is satisfied if the autocovariance function of  $x(t)$  is positive in a sufficiently large interval around zero. Also, if the rates  $\mu_j$  are adjustable we can choose  $\mu_{\min} = \min_{j \in \mathcal{N}} \mu_j$  large enough, such that B2) is always satisfied. Therefore, B2) is, in general, not restrictive for processes having dominant low frequency spectrum, which is typical in practical applications of SNs.

Based on the above modified problem definition, the following result was proved in [15], stating that both corrected gains and corrected offsets will converge to consensus points (which depend on the realizations of the stochastic processes) for all the nodes.

**Theorem 6 ([15]).** *Let Assumptions B1)–B4) be satisfied. Then  $\hat{\phi}(k)$  given by (37) converges to  $\hat{\phi}_{\infty} = \chi_1 i_1 + \chi_2 i_2$  in the mean square sense and w.p.1, where  $\chi_1$  and  $\chi_2$  are random variables with bounded second moments.*

## 6. Discussion

### 6.1. Rate of Convergence

In the above subsections we didn't analyze how quickly the presented algorithms can achieve convergence to the specified points. Since the basic equations (5) have constant step size, it can be concluded that the asymptotic convergence rate in the noiseless case is exponential. In the cases of measurement and additive communication noise, the convergence rate of the algorithms can be obtained following general methodology applied to the analysis of standard stochastic approximation algorithms. The following result gives an upper bound of the mean-square error with respect to the consensus point:

**Theorem 7 ([18]).** *Under the assumptions of any of the Theorems 3, 4 or 5, together with  $\lim_{t \rightarrow \infty} (\delta(t+1)^{-1} - \delta(t)^{-1}) = d \geq 0$ , there exists such a positive number  $\sigma' < 1$  that for all  $0 < \sigma < \sigma'$  the asymptotic consensus is achieved by the analyzed algorithms with the convergence rate  $o(\delta(t)^{\sigma})$ .*

It might be problematic to obtain the precise value of  $\sigma'$  in concrete applications. However, it can be shown that it directly depends on the sensor and network properties (encoded by matrix  $B(t)$  or  $B(t, d)$ ) and on the connectivity of the underlying communication graph [52]. On one hand, if the number of nodes is increased without increasing network connectivity, the rate of convergence will decrease; however, if the graph connectivity is increased, the convergence rate will also increase. For example, if the graph is fully connected, the convergence rate will be high at the expense of very large number of communication links. In practice, a compromise between the rate of convergence and the network complexity needs to be found.

Another compromise to be found is between algorithm's noise immunity and convergence rate. Indeed, according to [53], assuming that  $\delta(t)$  is given as  $\delta(t) = m_1 / (m_2 + t^{\mu})$ ,  $m_1, m_2 > 0$ ,  $\frac{1}{2} < \mu \leq 1$ , the values of  $\mu$  closer to  $\frac{1}{2}$  give larger rate of convergence but higher sensitivity to noise; the values of  $\mu$  closer to 1 give the opposite effect.

### 6.2. Stationarity of the measured signal

In the previous section, we made comments about all the introduced assumptions, explaining their practical applicability. Let us make some additional comments on the stationarity assumption for the random process  $\{x(t)\}$ , introduced in A2), A2') and B1). From the point of view of applications, it cannot be considered restrictive, since it encompasses a large variety of quickly and slowly varying

real signals. This assumption is not essential for proving convergence of the presented algorithms: it has been introduced primarily for the sake of focusing on the essential structural aspects of the algorithm and avoiding complex notation [14,15,18]. Notice, according to Lemmas 2 and 3, that the similarity transformation  $T$  can be applied in the above analysis even in the case of time varying matrix  $\bar{B}(t)$ , owing to its specific structure; namely, we have  $T^{-1}\bar{B}(t)T = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times (2n-2)} \\ 0_{(2n-2) \times 2} & \bar{B}(t)^* \end{bmatrix}$ , where  $\bar{B}(t)^*$  is Hurwitz and  $T$  is obtained from  $\bar{B}(t)$  for any selected  $t = t'$ . Moreover, notice that the conclusions of Theorem 1 hold, in general, provided the following unrestrictive condition holds:  $\lim_t \prod_\tau (I - \bar{B}(t - \tau)^*) = 0$ . Also, it is possible to conclude directly that the applied methodology of convergence analysis, the results of the above theorems hold for changes of  $\bar{B}(t)^*$  sufficiently slow. Moreover, it is not difficult to prove that the above convergence results exactly hold when the signal is asymptotically stationary.

### 6.3. Network Weights Design

As already discussed in the previous subsections, the implicit goal of the analyzed calibration scheme is to exploit the sensors with *a priori* good calibration properties by enforcing their dominating effect in the final consensus value to which all the nodes converge. This can be done in two ways, by adjusting the design weights  $\gamma_{ij}$ : 1) if the majority of sensors are “good”, we can set all  $\gamma_{ij}$  for the neighborhood of any node  $i$  to the same value, or 2) if there is a smaller subset  $\mathcal{N}^f \subset \mathcal{N}$  of *a priori* “good” sensors in the network we should appropriately tune the values of  $\gamma_{ij}$ . For this scenario, in this subsection, we give a detailed analysis of the weights adjusting problem for the case of asynchronous communications treated in Subsection 5.3.

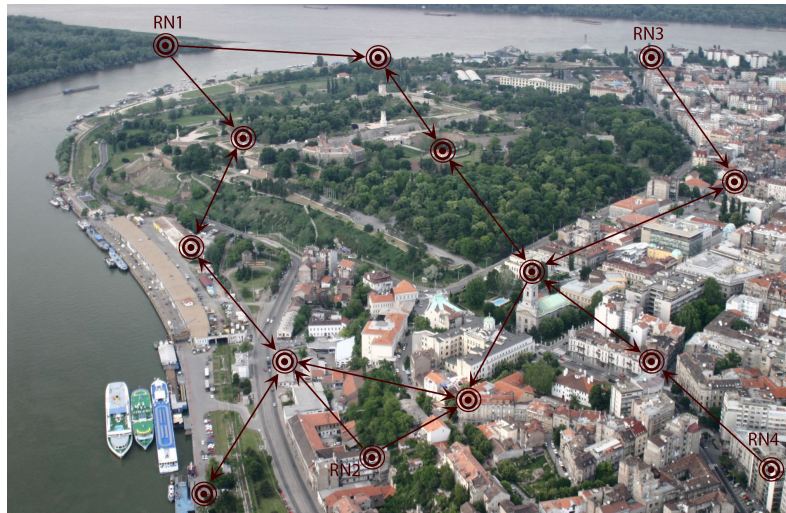
According to the theoretical analysis presented in details in [15], the dominant component of the random variables  $[\chi_1 \ \chi_2]$  in Theorem 6 is given by a weighted sum of the unknown sensor parameters  $\alpha_i$  and  $\beta_i$ . The positive weights are determined by the left eigenvectors  $w_1$  and  $w_2$  of  $\bar{B}$  corresponding to the zero eigenvalue. In turn, these weights are functions of the design parameters  $\gamma_{ij}$ , the wake-up probabilities  $q_j$ , and the dropout probabilities  $p_{ji}$ ,  $i, j = 1, \dots, n$ . Therefore, it is clear that the initial characteristics of a selected sensor  $i$  will have larger influence on the asymptotic consensus value if the appropriate elements of  $w_1$  and  $w_2$  are increased. This can be achieved in two ways:

1. By reducing the values of all the elements in the  $i$ -th row of  $\bar{\Gamma}$ , or
2. By increasing the values  $\gamma_{ji}$ ,  $j \neq i$ , from the  $i$ -th column (keeping in mind that  $\bar{\Gamma}$  must be row stochastic).

Probabilities  $q_j$  can in certain situations also be adjusted since they depend on the rate of the local clock of node  $j$ . By increasing the clock rate of the node  $j$ , the influence of that node on the asymptotic calibration parameter values achieved at consensus, is also increased. Adjusting dropout probabilities  $1 - p_{ji}$  might also be possible in certain situations: by decreasing the probability of a node  $i$  of receiving a message from a neighbor, we increase its influence on the asymptotic consensus. Hence, there are several design variables which can be adjusted so that the desired convergence point is achieved.

### 6.4. Macro Calibration for Networks with Reference Nodes

As discussed in the previous subsection, the selection of the weights in the matrix  $\bar{\Gamma}$  is important for attaining the calibration goal of emphasizing *a priori* selected “good” sensors (“leaders”). Besides the described methods, this can be ultimately done by leaving the nodes from a set  $\mathcal{N}^f \subset \mathcal{N}$  with unchanged calibration parameters (reference nodes), and only apply the recursions (31) (or (5), or (26)) to the rest of the nodes  $i \in \mathcal{N} - \mathcal{N}^f$ . An example of a SN with such topology corresponding to the smart city example in Fig. 1 is shown in Fig. 2. In practice, this situation emerges, for example, when a SN needs to be expanded, *i.e.*, when a number of uncalibrated sensors needs to be added to an already calibrated SN. In this subsection, an analysis of the convergence of the presented asynchronous calibration algorithm is presented for this case [15].



**Figure 2.** An example sensor network used in smart-city applications with multiple (four) reference nodes. The reference nodes (RNs) have fixed calibration parameters: only the rest of the nodes implement the given distributed sensor calibration recursions.

First, we analyze the special case in which  $|\mathcal{N}^f| = 1$ , *i.e.*, there is only one reference sensor, and we want to calibrate the rest of the SN so that their calibrated outputs converge to the output of the reference sensor. For this case, all of the above results still hold, since the resulting communication graph will again have a spanning tree (with the reference node as a center node), which implies that B3) (and A3)) holds. Therefore, by applying the above convergence theorems, one concludes that the corrected gains and offsets  $\hat{\phi}_i(k)$ ,  $i = 1, \dots, n$ , will converge to the same value, dictated by the “leader”.

In the general case, assume, without loss of generality, that  $\mathcal{N}^f = \{1, 2, \dots, n_f\}$ ,  $n_f = |\mathcal{N}^f| > 0$ , is the set of reference sensors which have fixed parameters:  $\phi_i^f = \begin{bmatrix} g_i^f \\ f_i^f \end{bmatrix}$ ,  $i \in \mathcal{N}^f$ . Assume that

$\bar{\phi}^f = [\phi_1^{fT} \dots \phi_{n_f}^{fT}]^T$ . The calibration algorithms above are now applied in the same way, except that the reference nodes do not change the calibration parameters:  $\hat{\theta}_i(k) = \hat{\theta}_i(k-1)$  for all  $i \in \mathcal{N}^f$ . Let  $\mathcal{N} - \mathcal{N}^f = \{n_f + 1, \dots, n\}$  and let  $\hat{\phi}^v(k) = [\hat{\phi}_{n_f+1}(k)^T \dots \hat{\phi}_n(k)^T]^T$  be the vector of all the calibration parameters to be tuned. In this case, all of the above theorems do not hold anymore, since, when  $n_f > 1$ , the communication graph does not necessarily satisfy B3) (the graph doesn’t have a center node anymore, since two or more reference nodes are not mutually reachable). Hence, a separate convergence theorem is needed which treats this case:

**Theorem 8 ([15]).** Let Assumptions B1)–B4) be satisfied and let all the nodes from  $\mathcal{N} - \mathcal{N}^f$  be reachable from all the nodes in  $\mathcal{N}^f$ . Then the algorithm (31), in which  $\gamma_{ij} = 0$  for all  $i \in \mathcal{N}^f$ , provides convergence of  $\hat{\phi}^v(k)$  in the mean square sense and w.p.1 to the limit defined by

$$\hat{\phi}_\infty^v = -(\bar{\Gamma}^v \otimes I_2)^{-1}(\bar{\Gamma}^{f,v} \otimes I_2)\bar{\phi}^f, \quad (38)$$

where matrices  $\bar{\Gamma}^v$  and  $\bar{\Gamma}^{f,v}$  are  $(n - n_f) \times (n - n_f)$  and  $(n - n_f) \times n_f$  submatrices of matrix  $P^{-c}\bar{\Gamma}$ , with indices  $i, j = n_f + 1, \dots, n$  and  $i = n_f + 1, \dots, n$ ,  $j = 1, \dots, n_f$ , respectively;  $P = \text{diag}\{p_1, \dots, p_n\}$  and  $c$  is defined in (31).

It can be easily concluded that if the calibration parameters of all the reference nodes are the same, equal to some  $\phi^f$ , then the calibration parameters of the rest of the nodes will also converge to  $\phi^f$ . If this is not the case, it can be seen from (38) that the calibration parameters of different nodes will



converge to different values. These values will be typically dictated by the reference sensors which are the closest to a given node.

### 6.5. Autonomous Gain Correction and Relationship with Time Synchronization

After presenting the most important aspects of the presented blind calibration methodology encompassing both noiseless and noisy environments, we will now make a few comments on the problem of its relationship with the algorithms for *time synchronization* in sensor networks, which has attracted a lot of attention (e.g., [27–34,54–56] and the references therein). Indeed, after coming back to the main measurement model, one easily realizes that in the case of time synchronization one has the form (1) with the absolute time  $t$  replacing the signal value  $x(t)$ . The estimation schemes used in the mentioned analogous time synchronization algorithms are, indeed, related to the estimation of the parameters of the calibration functions (2) based on the utilization of local time measurements, but, however, consists of one separate recursion for the relative drift estimation and one separate recursion for the estimation of offsets, relying on the obtained relative drifts. In [57] this scheme was reformulated in the light of the calibration problem and the above described methodology. One starts from the difference model  $\Delta y_i(t) = y_i(t+1) - y_i(t) = \alpha_i \Delta x(t)$ , where  $\Delta x(t) = x(t+1) - x(t)$ , and construct a gradient recursion for  $\hat{\alpha}_i$  following the above methodology, having in mind that  $\Delta y_i(t)$  does not depend on  $\beta_i$ . On the other hand, the estimation of  $b_i$  has to start from (1); it has the form of the recursion for  $\hat{b}_i$  in (5), but should use  $\hat{\alpha}_i$  generated by first recursion. Such a combined gradient algorithm based on utilization of  $\Delta y_i(t)$  resembles typical time synchronization algorithms. In general, it is important to observe that the introduction of  $t$  instead of  $x(t)$  in the basic relation (1) suffers from the very basic problem that unboundedness of the linear function contradicts the requirements for boundedness of the second order moments of  $x(t)$  (typical for stochastic approximation algorithms) and that it is not possible to guarantee convergence of the obtained recursions. This indicates that formal transfers of methodologies from one domain to the other should be done with extreme caution.

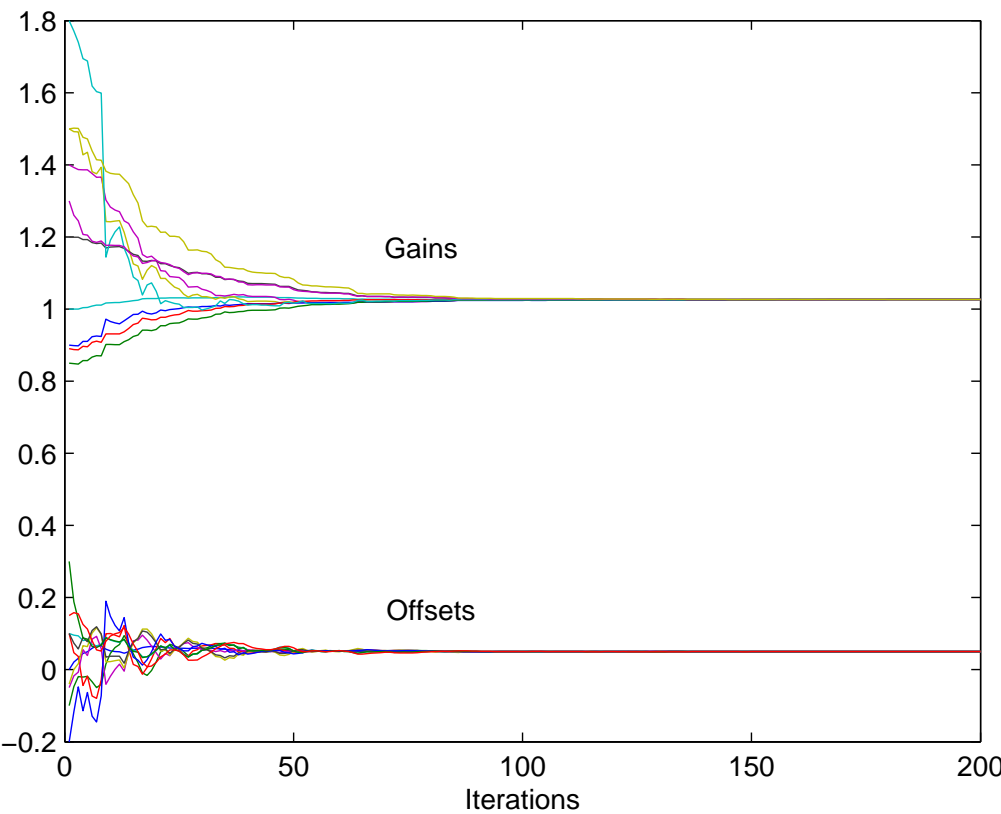
## 7. Simulation Results

In this section we present the results of extensive simulations, illustrating the above described properties of the analyzed algorithms. In all of the simulations, a SN with ten nodes, with randomly generated communication graph satisfying assumption A3), has been chosen. The sensor parameters  $\alpha_i$  and  $\beta_i$  have been generated randomly from uniform distribution with means one and zero, respectively, and with standard deviation 0.3.

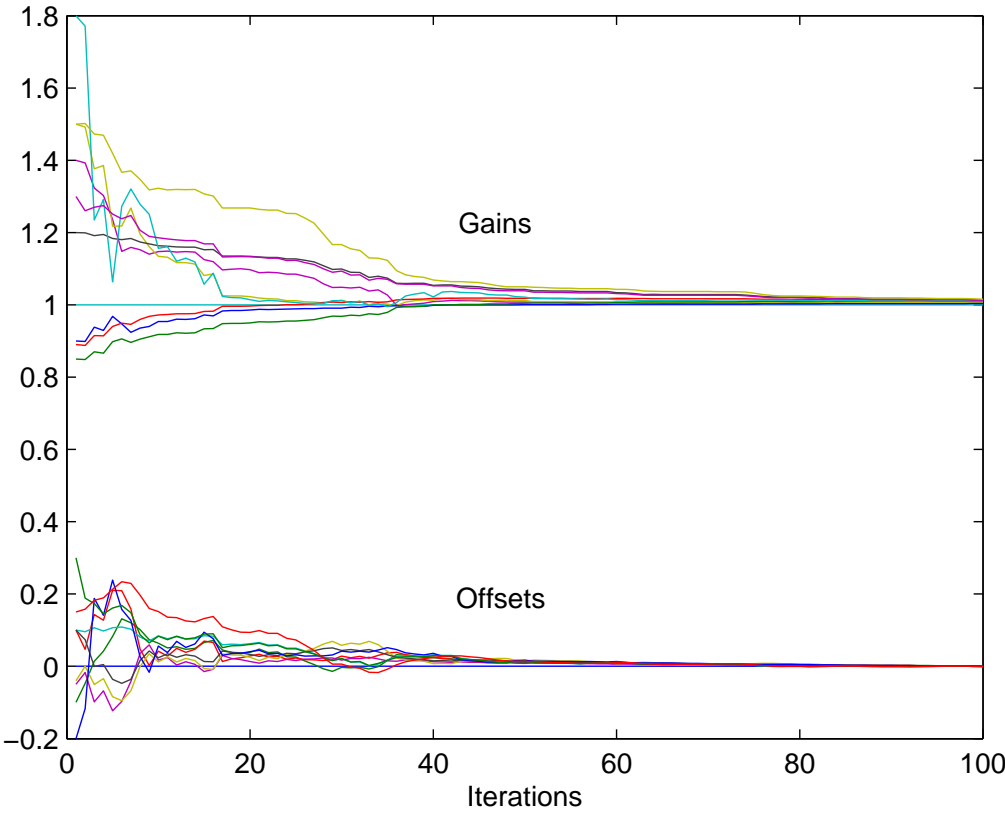
In Fig. 3 the corrected gains  $\hat{g}_i(t)$  and offsets  $\hat{f}_i(t)$  obtained by the presented algorithm (5) in the noiseless case are presented, for a preselected step size, equal for all the nodes,  $\delta = 0.01$ . It is clear from the figure that the convergence to consensus, and, hence, implicit asymptotic calibration is achieved with the asymptotic values close the desired (one for corrected gains, and zero for corrected offsets).

In Fig. 4 the simulation results are presented for the situation in which the first node is set to be a reference node (“leader”) with perfect parameters  $\alpha_1 = 1$  and  $\beta_1 = 0$ . Obviously, all of the rest of the nodes converge to this ideal characteristic.

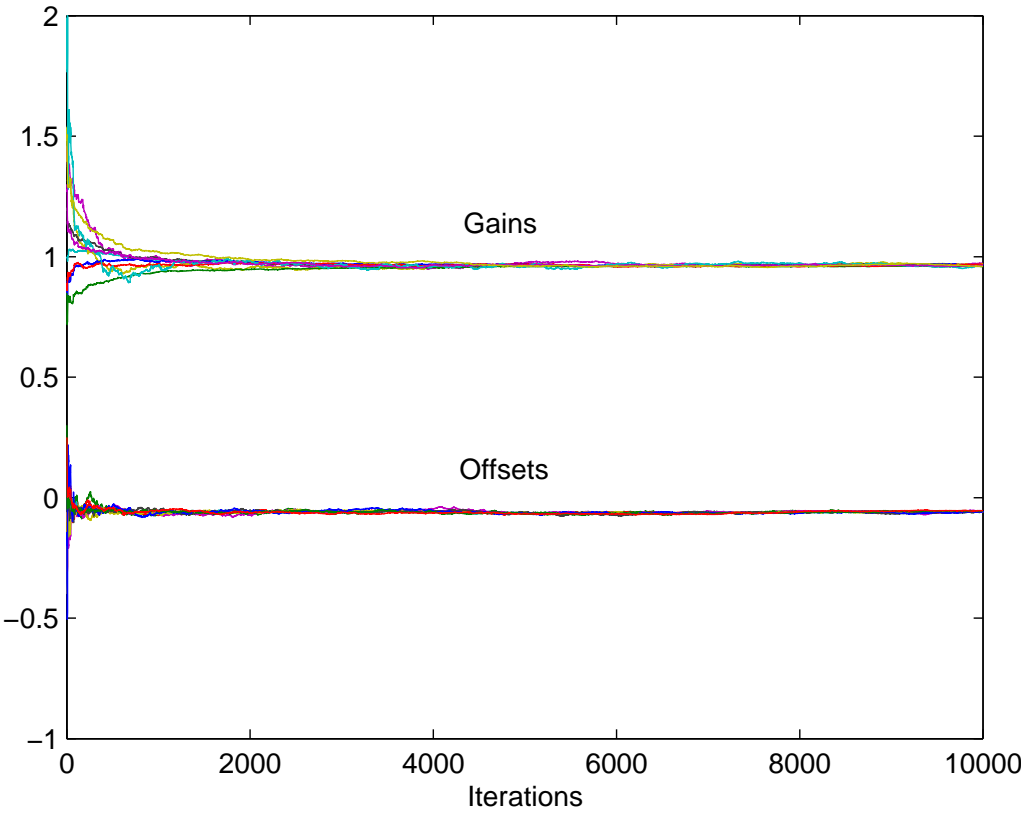
In Fig. 5 the corrected gains  $\hat{g}_i(t)$  and offsets  $\hat{f}_i(t)$  are depicted for the case in which all the theoretically discussed unreliabilities are included: communication dropouts with  $p = 0.2$  for all the links in the network, normally distributed communication additive noise with variance 0.1, and normally distributed measurement noises with variances different for all the nodes, uniformly generated in the range  $(0, 0.1)$ . In this case, in order to achieve convergence in the presence of noise, the time varying diminishing step-size sequences are chosen  $\delta(t) = 0.01/t^{0.6}$ , equal for all the nodes. The algorithm which works in this case is the one based on the instrumental variables (26), so that the measured signal  $x(t)$  is assumed to be generated by a second order linear system with white noise at the input, resulting in a correlated sequence with zero mean and standard deviation one. It is obvious from the figure that the convergence is achieved in spite of the presence of all the introduced unreliabilities.



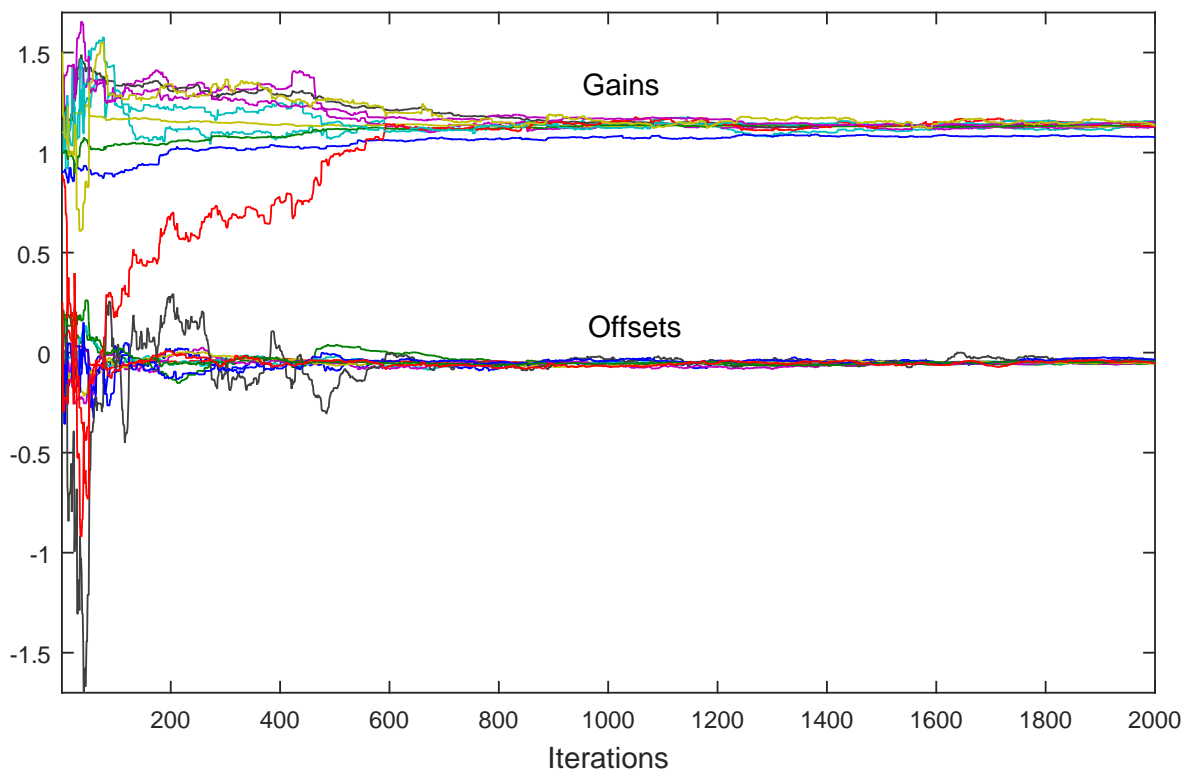
**Figure 3.** Noiseless synchronous algorithm without references: convergence to consensus is achieved for corrected gains and corrected offsets



**Figure 4.** Noiseless synchronous algorithm with one reference sensor: convergence to the reference is achieved.



**Figure 5.** The modified algorithm (25): convergence to consensus is achieved for corrected gains and corrected offsets in spite of measurement noise presence.



**Figure 6.** The asynchronous algorithm based on instrumental variables without reference sensors: convergence to consensus is achieved for corrected gains and corrected offsets.

Next we simulate the asynchronous algorithm (31) which includes instrumental variables. It is assumed that the local clocks of all the nodes are driven by Poisson processes with the same rates. In Fig. 6, the corrected gains  $\hat{g}_i(k)$  and offsets  $\hat{f}_i(k)$  generated by (31) are depicted assuming the steps  $\delta(k) = 0.01/k^{0.6}$  and the presence of communication dropouts with probabilities  $p_{ij} = 0.2$  for each link. The measurement noises and the signal  $x(k)$  are assumed to be the same as in Fig. 5.

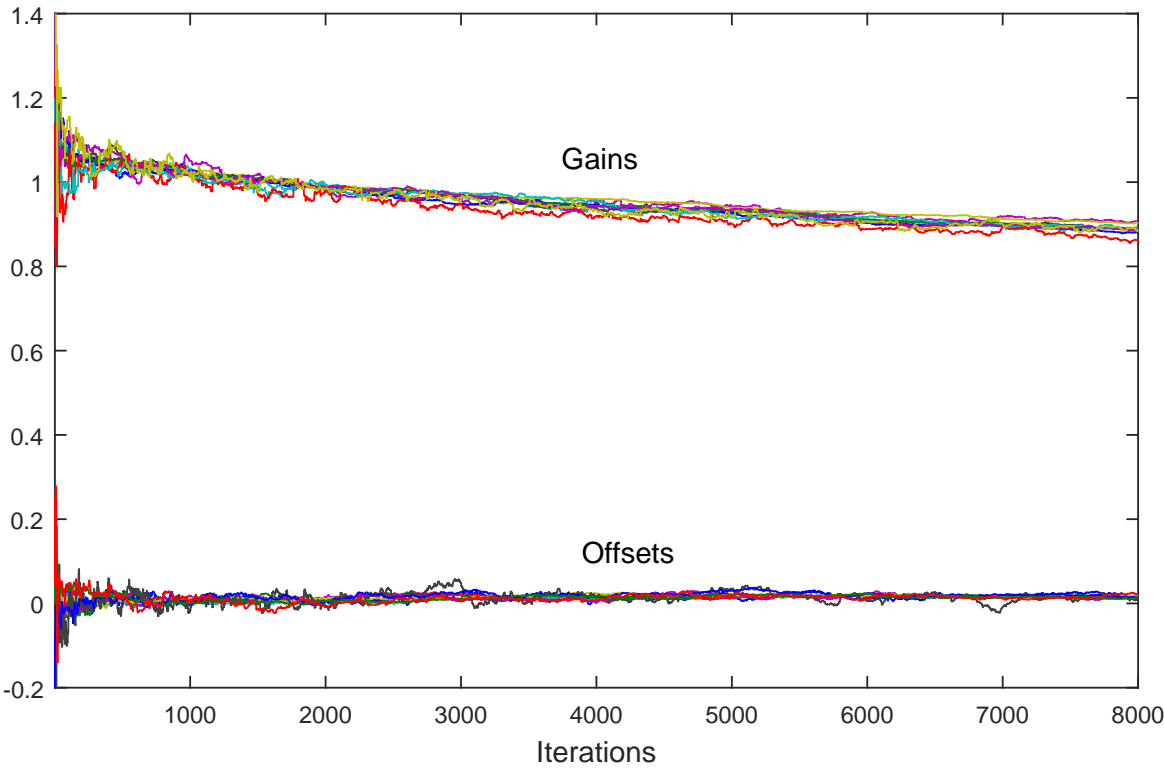
In Fig. 7 the necessity of introducing the instrumental variables is demonstrated when the noise is present. The basic algorithm (5), without instrumental variables, has been simulated and the convergence is not achieved in this case. It can be seen that all the corrected gains  $\hat{g}_i(k)$  slowly converge to zero which is very undesirable.

Fig. 8 illustrates the case discussed in Subsection 6.4 in which there are two reference sensors in the given SN. It can be seen from the figure that, as predicted by Theorem 8, the consensus is not achieved, and that all the calibration parameters converge to different values, determined by equation (38).

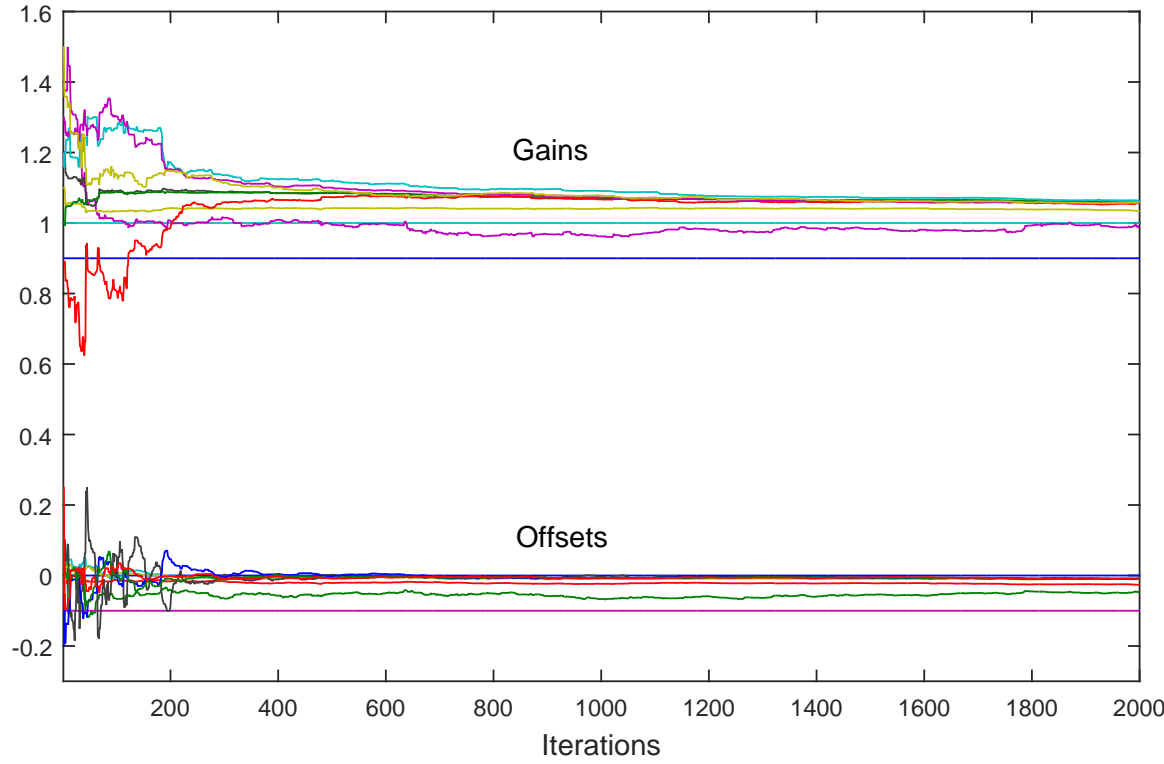
## 8. Conclusion

In this review, distributed recursive blind macro-calibration algorithms based on consensus have been treated. It was shown that the algorithms successfully perform on lossy sensor networks, characterized by unreliable communications, limited only to the local neighbors. The algorithms are completely decentralized, not requiring any fusion center. Convergence properties have been analyzed under both noiseless conditions, and the conditions typical for noisy and unreliable environments. The practically important case of asynchronous communication based on a broadcast gossip scheme has also been treated. Extensive discussions have been provided, explaining in details several important practical issues: convergence rate, design of tunable network weights, and calibration of sensor





**Figure 7.** Stochastic gradient algorithm: convergence to consensus is not achieved.



**Figure 8.** The asynchronous algorithm with two reference sensors with different characteristics: both the corrected gains and the corrected offsets converge to different values determined by (38).

networks with multiple reference nodes. Extensive simulation results illustrating the behavior of the analyzed algorithms have been presented.

### 8.1. Future Work

The presented results can be extended in several directions.

The first future direction, which is imposed naturally, is to extend the presented algorithms, or develop new ones, for the case when the nodes measure spatially varying but correlated signals. Treating this situation would drastically increase the practical applicability of the described methodology. The performance of such a scheme would highly depend on *a priori* knowledge about the interrelatedness of the measurements of different nodes.

Another direction of possible future work is to extend the results to include the possibility that individual nodes measure vector values (instead of scalars treated in this review). This case would arise in practically frequent situations when there are multiple diverse sensors on each node, and when each sensor possibly measures different (overlapping) subsets of all the available sensed values. One possible idea for treating these situations is to use the wide existing literature in overlapping decentralized estimation (e.g. [21,58–61]). One typical example where this situation arises are networks of cameras with different view angles and scene coverage.

Third future direction emerges if one interprets the above results not in the context of calibration, but as pure synchronization (consensus) results for certain special types of linear time-varying systems. Indeed, the derivations and results of the presented convergence theorems open up a possibility of extending them to the case of synchronization of general higher order linear parameter-varying systems, which is a topic of high theoretical importance (e.g. [62] and references therein).

Finally, sensor networks often operate in hostile environments where battery usage and power consumption of the sensor nodes are of crucial importance. In this context it would be of high importance to analyze, for a given particular sensor network mission, how to formulate optimization problem and achieve optimal compromise between sensor network calibration and the nominal operation dictated by the given mission.

This work was partially supported by Fundação para a Ciência e a Tecnologia under Project UID/EEA/00066/2013, Project foRESTER PCIF/SSI/0102/2017, and Program Investigador FCT under Grant IF/00325/2015.

## References

1. Kim, K.D.; Kumar, P.R. Cyber-physical systems: A perspective at the centennial. *Proc. IEEE* **2012**, *100*, 1287–1308.
2. Holler, J.; Tsiatsis, V.; Mulligan, C.; Avesand, S.; Karnouskos, S.; Boyle, D. *From Machine-to-machine to the Internet of Things: Introduction to a New Age of Intelligence*; Academic Press, 2014.
3. Akyildiz, I.F.; Vuran, M.C. *Wireless sensor networks*; John Wiley & Sons, 2010.
4. *Proceedings of the IEEE, Special issue on sensor networks and applications*; Vol. 91, 2003.
5. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: a survey. *Computer Networks* **2002**, *38*, 393–422.
6. Speranzon, A.; Fischione, C.; Johansson, K.H. Distributed and collaborative estimation over wireless sensor networks. *Proc. IEEE Conf. on Decision and Control*, 2006, pp. 1025–1030.
7. Whitehouse, K.; Culler, D. Calibration as parameter estimation in sensor networks. *Proc. 1st ACM International Workshop on Wireless sensor networks and applications*, 2002, pp. 59–67.
8. Whitehouse, K.; Culler, D. Macro-calibration in sensor/actuator networks. *Mobile Netw. Applicat.* **2003**, *8*, 463–472.
9. Balzano, L.; Nowak, R. Blind calibration. Technical Report TR-UCLA-NESL-200702-01, Networked and Embedded Systems Laboratory, UCLA, 2007.
10. Balzano, L.; Nowak, R. Blind calibration of sensor networks. *Proc. Intern. Conf. Inf. Proc. in Sensor Networks*, 2007, pp. 79–88.

11. Römer, K.; Blum, P.; Meier, L., Time Synchronization and Calibration in Wireless Sensor Networks. In *Handbook of Sensor Networks*; John Wiley & Sons, Inc., 2005; pp. 199–237.
12. Bychkovskiy, V.; Megerian, S.; Estrin, D.; Potkonjak, M. A Collaborative Approach to In-Place Sensor Calibration. *Proc. Intern. Conf. Inf. Proc. in Sensor Networks*, 2003, pp. 301–316.
13. Takruri, M.; Challa, S.; Yunis, R. Data Fusion Techniques for Auto Calibration in Wireless Sensor Networks. *Proc. International Conference on Information Fusion*, 2009, pp. 132–139.
14. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed Blind Calibration in Lossy Sensor Networks via Output Synchronization. *IEEE Trans. Autom. Control* **2015**, *60*, 3257–3262.
15. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Asynchronous Distributed Blind Calibration of Sensor Networks under Noisy Measurements. *IEEE Transactions on Control of Network Systems* **2018**, *5*, 571–582.
16. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed Macro Calibration in Sensor Networks. *Proc. Mediterr. Conf. Control and Automation*, 2012, pp. 1049 – 1054.
17. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed Calibration for Sensor Networks under Communication Errors and Measurement Noise. *Proc. IEEE Conf. Decision and Control*, 2012, pp. 1380 – 1385.
18. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. A consensus-based distributed calibration algorithm for sensor networks. *Serbian Journal of Electrical Engineering* **2016**, *13*, 111–132.
19. Ren, W.; Beard, R. Consensus seeking in multi-agent systems under dynamically changing interaction topologies. *IEEE Trans. Autom. Control* **2005**, *50*, 655–661.
20. Ohta, Y.; Šiljak, D. Overlapping block diagonal dominance and existence of Lyapunov functions. *J. Math. Analysis Appl.* **1985**, *112*, 396–410.
21. Šiljak, D.D. *Decentralized Control of Complex Systems*; Academic Press: New York, 1991.
22. Pierce, I.F. Matrices with dominating diagonal blocks. *Journ. of Economic Theory* **1974**, *9*, 159–170.
23. Levin, A.; Weiss, Y.; Durand, F.; Freeman, W. Understanding and evaluating blind deconvolution algorithms. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 1964–1971.
24. Yu, C.; Xie, L. On Recursive Blind Equalization in Sensor Networks. *IEEE Trans. Signal Process.* **2015**, *63*, 662–672.
25. Nandi, A. *Blind Estimation Using Higher-Order Statistics*; Kluwer Academic Publishers: Boston, MA, 1999.
26. Bilen, C.; Puy, G.; Gribonval, R.; Daudet, L. Convex Optimization Approaches for Blind Sensor Calibration Using Sparsity. *IEEE Trans. Signal Process.* **2014**, *62*, 4847–4856.
27. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed drift estimation for time synchronization in lossy networks. *Proc. Mediterr. Conf. Control and Automation*, 2016, pp. 779 – 784.
28. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed time synchronization for networks with random delays and measurement noise. *Automatica* **2018**, *93*, 126 – 137.
29. Giridhar, D.; Kumar, P.R. Distributed clock synchronization over wireless networks: algorithms and analysis. *Proc. IEEE Conf. Decision and Control*, 2006, pp. 263–270.
30. Sommer, P.; Wattenhofer, R. Gradient clock synchronization in wireless sensor networks. *Proc. Int. Conf. Inf. Proc. in Sensor Networks*, 2009, pp. 37–48.
31. Schenato, L.; Fiorentin, F. Average TimeSynch: a consensus-based protocol for time synchronization in wireless sensor networks. *Automatica* **2011**, *47*, 1878–1886.
32. Carli, R.; Chiuso, A.; Schenato, L.; Zampieri, S. Optimal Synchronization for Networks of Noisy Double Integrators. *IEEE Trans. Autom. Control* **2008**, *56*, 1146–1152.
33. Liao, C.; Barooah, P. Distributed clock skew and offset estimation from relative measurements in mobile networks with Markovian switching topologies. *Automatica* **2013**, *49*, 3015–3022.
34. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed time synchronization in lossy wireless sensor networks. *Proc. 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems*, 2012, pp. 25–30.
35. Ravazzi, C.; Frasca, P.; Tempo, R.; Ishii, H. Ergodic randomized algorithms and dynamics over networks. *IEEE Trans. Control Netw. Syst.* **2015**, *2*, 78–87.
36. Carron, A.; Todescato, M.; Carli, R.; Schenato, L. An asynchronous consensus-based algorithm for estimation from noisy relative measurements. *IEEE Trans. Control Netw. Syst.* **2014**, *1*, 283–295.
37. Bolognani, S.; Favero, S.D.; Schenato, L.; Varagnolo, D. Consensus-based distributed sensor calibration and least-square parameter identification in WSNs. *International Journal of Robust and Nonlinear Control* **2010**, *20*.

38. Miluzzo, E.; Lane, N.D.; Campbell, A.T.; Olfati-Saber, R. CaliBree: A Self-calibration System for Mobile Sensor Networks. *Proc. 4th IEEE Int. Conf. Distributed Computing in Sensor Systems*, 2008, pp. 314–331.
39. Olfati-Saber, R.; Fax, A.; Murray, R. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE* **2007**, *95*, 215–233.
40. Ljung, L.; Söderström, T. *Theory and practice of recursive identification*; MIT Press: Cambridge, Mass., 1983.
41. Xiao, Y.; Xiong, Z.; Niyato, D.; Han, Z. Distortion minimization via adaptive digital and analog transmission for energy harvesting-based wireless sensor networks. *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 518–521.
42. Chen, H.F. *Stochastic approximation and its applications*; Kluwer Academic: Dordrecht, the Netherlands, 2002.
43. Li, T.; Zhang, J.F. Consensus conditions of multi agent systems with time varying topologies. *IEEE Trans. Autom. Control* **2010**, *55*, 2043–2056.
44. Huang, M.; Manton, J.H. Stochastic consensus seeking with noisy and directed inter-agent communications: fixed and randomly varying topologies. *IEEE Trans. Autom. Control* **2010**, *55*, 235–241.
45. Ljung, L. *System Identification - theory for the user*; Prentice Hall International: englewood Cliffs, N.J., 1989.
46. Söderström, T.; Stoica, P. *System Identification*; Prentice Hall International: Hemel Hempstead, UK, 1989.
47. Aysal, T.C.; Yildriz, M.E.; Sarwate, A.D.; Scaglione, A. Broadcast gossip algorithms for consensus. *IEEE Trans. Signal Process.* **2009**, *57*, 2748–2761.
48. Nedić, A. Asynchronous broadcast-based convex optimization over a network. *IEEE Trans. Autom. Control* **2011**, *56*, 1337–1351.
49. Bradley, R.C. Basic Properties of Strong Mixing Conditions A Survey and Some Open Questions. *Probability Surveys* **2005**, *2*, 107–144.
50. Ibragimov, I. Some limit theorems for stochastic processes stationary in the strict sense. *Dokl. Akad. Nauk SSSR* **1959**, *125*, 711–714.
51. Rosenblatt, M. A central limit theorem and a strong mixing condition. *Proc. Natl. acad. Sci. U.S.A.* **1956**, *42*, 43–47.
52. Godsil, C.; Royle, G. *Algebraic Graph Theory*; Springer Verlag: New York, 2001.
53. Borkar, V.; Meyn, S.P. The ODE method for convergence of stochastic approximation and reinforcement learning. *SIAM Journal on Control And Optimization* **2000**, *38*, 447–469.
54. Stanković, M.S.; Stanković, S.S.; Johansson, K.H. Distributed Offset Correction for Time Synchronization in Networks with Random Delays. *Proc. European Control Conference*, 2018.
55. Tian, Y.P.; Zong, S.; Cao, Q. Structural modeling and convergence analysis of consensus-based time synchronization algorithms over networks: non-topological conditions. *Automatica* **2016**, *65*, 64–75.
56. Chen, J.; Yu, Q.; Zhang, Y.; Chen, H.H.; Sun, Y. Feedback-Based Clock Synchronization in Wireless Sensor Networks: A Control Theoretic Approach. *IEEE Trans. Vehicular Techn.* **2010**, pp. 2963–2973.
57. Stanković, M. Distributed Asynchronous Consensus-based Algorithm for Blind Calibration of Sensor Networks with Autonomous Gain Correction. *IET Control Theory and Applications* **2018**.
58. Stanković, S.S.; Stanković, M.S.; Stipanović, D.M. Consensus based overlapping decentralized estimator. *IEEE Trans. Autom. Control* **2009**, *54*, 410–415.
59. Stanković, S.S.; Stanković, M.S.; Stipanović, D.M. Consensus Based Overlapping Decentralized Estimation With Missing Observations and Communication Faults. *Automatica* **2009**, *45*, 1397–1406.
60. Stanković, M.S.; Stanković, S.S.; Stipanović, D.M. Consensus-based decentralized real-time identification of large-scale systems. *Automatica* **2015**, *60*, 219 – 226.
61. Stanković, S.S.; Šiljak, D.D. Model abstraction and inclusion principle: a comparison. *IEEE Trans. Autom. Control* **2001**, *8*, 816–832.
62. Seyboth, G.; Schmidt, G.; Allgöwer, F. Output synchronization of linear parameter-varying systems via dynamic couplings. *Proc. IEEE Conf. Decision and Control*, 2012, pp. 5128–5133.