Distributed Network Size Estimation Executed by Average Consensus Bounded by Stopping Criterion for Wireless Sensor Networks

Martin Kenyeres
Institute of Informatics
Slovak Academy of Sciences
Bratislava, Slovakia
martin.kenyeres@savba.sk

Jozef Kenyeres
Sipwise GmbH
Brunn am Gebirge, Austria
jkenyeres@sipwise.com

***Abstract*—The exact information about the network size is crucial for the proper functioning of many distributed algorithms. In this paper, we analyze the average consensus algorithm for a distributed network size estimation bounded by the stopping criterion proposed for the wireless sensor networks. We analyze its four initial configurations over random geometric graphs of different connectivity under various parameters of the implemented stopping criterion. The performance is evaluated by the mean square error and the convergence rate expressed as the iteration number for the consensus. Finally, the results obtained under various conditions are compared to find the best performing configuration of both the average consensus algorithm and the implemented stopping criterion. Also, the results are compared to the distributed summing functionality.**

Keywords-distributed computing; distributed network size estimation; wireless sensor networks; stopping criterion

I. Introduction

Data aggregation poses an essential building block in various architectures (among others WSNs[1]) and may play a key role in the proper functioning of many distributed algorithms (e.g., the dimensioning of distributed hash table structures, the configuration of voting algorithms, load-balancing decisions, disk space estimation in peer-to-peer sharing systems, synchronization in ad-hoc systems, etc.) [1]. Distributed data aggregation does not present a trivial task, especially when no global identifiers or centralized algorithms are used [1]. One of the key demands required by the most distributed algorithms is to ensure the information about the network size n at each sensor node beforehand [2]. Therefore, this has instigated the formation of various techniques for this purpose (e.g., extrema propagation, distributed orthogonalization, consensus-based algorithms, etc.) [2].

As mentioned above, WSNs is a technology where data aggregation mechanisms are assumed to support the proper functioning of the executed applications [3]. WSNs, finding the application in various areas such as geographical surveillance, health-care, industrial monitoring, etc. [4, 5], are formed by geographically distributed sensor nodes to sense the adjacent environment and communicate together in order to make a meaningful decision about the monitored physical quantity [6]. As these sensor nodes are affordable devices, they suffer from insufficient communication, computation, energy, and memory capabilities [6]. Thus, one of their crucial designs issue is to ensure effective algorithm operation in short execution time (achieved by selection of an appropriate algorithm, an effective setup of the implemented stopping criterion, etc.) [6].

We focus our attention on the average consensus algorithm (AC) for a network size estimation, which finds a wide application in WSNs due to its specific character (e.g., lower energy, memory, communication demands, high robustness, easy implementation, etc.) [7]. We analyze its four initial configurations bounded by the stopping criterion from [7] proposed for WSNs over 30 random geometric graphs (RGGs) of either dense or sparse connectivity. This paper follows on our research[2] focused on the applicability of the mentioned stopping criterion for distributed summing in WSNs. We vary the parameters of the implemented stopping criterion (namely, *accuracy* and *counter threshold*) and evaluate the performance of four selected initial configurations using two metrics (namely, the mean square error (MSE) and the convergence rate expressed as the number of the iterations for the consensus) under various conditions. Our goal is to find the most suitable initial configuration of both AC and the implemented stopping criterion.

The next section is focused on the theoretical background, i.e, we introduce a model of AC over WSNs, its definition, the convergence conditions, and the implemented stopping criterion. In Section III., we introduce the used research methodology and the applied metrics. Section IV. is concerned with the experimental results from Matlab2016a and comparison with [8], where AC for distributed summing with the examined stopping criterion is analyzed (the research

This work was supported by the VEGA agency under the contract No. 2/0155/19 and by the Slovak Research and Development Agency under the contract No. APVV-17-0116.

[1] Wireless sensor networks

[2] presented in [8]

ISBN 978-80-261-0812-2, ©University of West Bohemia, 2019

methodology is the same as in this paper, allowing easy mutual comparison).

II. Theoretical Background

A. Modeling of AC over WSNs

We consider WSNs to be indirect finite graphs G determined by the vertex set $\mathbf{V}$ and the edge set $\mathbf{E}$ [9]. The set $\mathbf{V}$ contains all the graph vertices, which represent the sensor nodes in WSN. A unique index is allocated to each vertex (v_i, i = 1,2....n) for easy identification. The parameter n represents the graph order, i.e, the number of the vertices in a graph. The set $\mathbf{E}$ consists of all the graph edges, indicating the direct connection between two vertices (referred to as (v_i,v_j) or e_{ij}).

At the beginning of the algorithm, each sensor node initiates its inner state with either "1" or "0" [2]. Only one of the sensor nodes takes "1" - it is so-called the leader. In our analyses, we appoint the node with the highest degree[3] as the leader. If there are two sensor nodes with the same degree, the one with a lower identity number is the leader. The inner states of the sensor nodes are gathered in a column vector $\mathbf{x}(k)$ variant over the iterations[4] and with the size of n. Thus, the initial states can be defined as follows[5]:

$$x_i(0) = \begin{cases} 1, & \text{if } v_i \text{ is the leader} \\ 0, & \text{if } v_i \text{ is not the leader} \end{cases} \tag{1}$$

In AC, at each iteration, all the sensor nodes update their inner states as a combination of the current inner state and the inner states collected from the adjacent sensor nodes. We assume the Constant weights of this algorithm, i.e., each edge in a graph is allocated the same weight equal to the mixing parameter ϵ. The described procedure can be modeled as the difference equation defined as follows [9]:

$$\mathbf{x}(k+1) = \mathbf{W} \times \mathbf{x}(k) \tag{2}$$

Here, $\mathbf{W}$ is the weight matrix, whose elements affect AC in several aspects (e.g., the convergence rate, the robustness, the convergence of the algorithm, etc.) [8]. As mentioned earlier, we analyze the Constant weights, whose weight matrix can be defined as follows [8]:

$$[W]_{ij}^{\mathrm{CW}} = \begin{cases} \epsilon, & \text{if } e_{ij} \in \mathbf{E} \\ 1 - d_i.\epsilon, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Here, d_i represents the degree of the corresponding sensor node v_i. The inner states of all the sensor nodes are iteratively exchanged between each two adjacent sensor nodes and asymptotically converge to the arithmetic mean calculated from all the inner states, i.e., to $\frac{1}{n}$, which can be expressed as follows [10]:

$$\lim_{k\to\infty} \mathbf{x}(k) = \lim_{k\to\infty} \mathbf{W}^k \times \mathbf{x}(0) = \frac{1}{n} \cdot \mathbf{1} \times \mathbf{1}^{\mathrm{T}} \times \mathbf{x}(0) \tag{4}$$

Here, $\mathbf{1}$ is an all-ones vector of a column shape [8]. Then, each sensor node is able to estimate the network size at each iteration as follows [2]:

$$x_i^{NS}(k) = \frac{1}{x_i(k)}, \forall v_i \in \mathbf{V} \tag{5}$$

The existence of the limit is necessary for AC to work correctly [10]. It is ensured when these three conditions hold [10]:

$$\mathbf{1}^{\mathrm{T}} \times \mathbf{W} = \mathbf{1}^{\mathrm{T}}, \mathbf{W} \times \mathbf{1} = \mathbf{1}, \rho(\mathbf{W} - \frac{1}{n} \cdot \mathbf{1} \times \mathbf{1}^{\mathrm{T}}) < 1 \tag{6}$$

Here, $\rho(\cdot)$ is the spectral radius of the analyzed matrix.

B. Implemented Stopping Criterion

As mentioned earlier, we assume that the execution of AC is bounded by the stopping criterion from [7]. This stopping criterion is determined by two pre-set constants, namely, *accuracy* and *counter threshold*, which are the same for each sensor node. Furthermore, each sensor node has its own independent *counter*, a variable initiated with "0" at the beginning of the algorithm. The implemented stopping criterion is fully-distributed, thereby finding application in WSNs. It works in such a way that each sensor node verifies whether its two subsequent inner states are smaller than pre-set *accuracy*, and the inner state of the corresponding sensor node is not equal to "0"[6]. If so, it increments *counter* by "1", otherwise, sets its value to "0" regardless of its current value. When *counter* reaches the value equal to *counter threshold* at a sensor node, this sensor node considers the algorithm to be locally completed and participates in AC no longer - it does not communicate with the other adjacent sensor nodes, and its inner state is updated no more.

III. Research Methodology

In this section, we introduce the used research methodology and the metrics for performance evaluation.

As mentioned above, we focus our attention on the Constant weights of AC, which is characterized by uniform edge weights. As discussed above, the existence of the limit provided in (4) is crucial for the proper functioning of AC. Thus, in this paper, we select four following values of the mixing parameters ϵ, ensuring the convergence of the algorithm in all our examined graphs [9]:

- ϵ = 0.25 . $1/\Delta$ (referred to as ϵ = 0.25)
- ϵ = 0,50 . $1/\Delta$ (referred to as ϵ = 0.50)
- ϵ = 0.75 . $1/\Delta$ (referred to as ϵ = 0.75)
- ϵ = $1/\Delta$ (referred to as ϵ = 1)

Here, Δ is the degree of the best-connected sensor node, i.e., the maximum degree of the graph.

Furthermore, we analyze AC over RGGs of either dense or sparse connectivity. We generate 30 densely connected unique graphs and 30 sparsely connected unique graph, both formed by 200 nodes.

[3] i.e., the sensor node with the most neighbors

[4] k is the label of an iteration

[5] we assume that $\mathbf{x}(k$ = 0) poses the initial inner states

[6] the second condition is added compared to distributed averaging and summing - it increases the precision of the final estimates

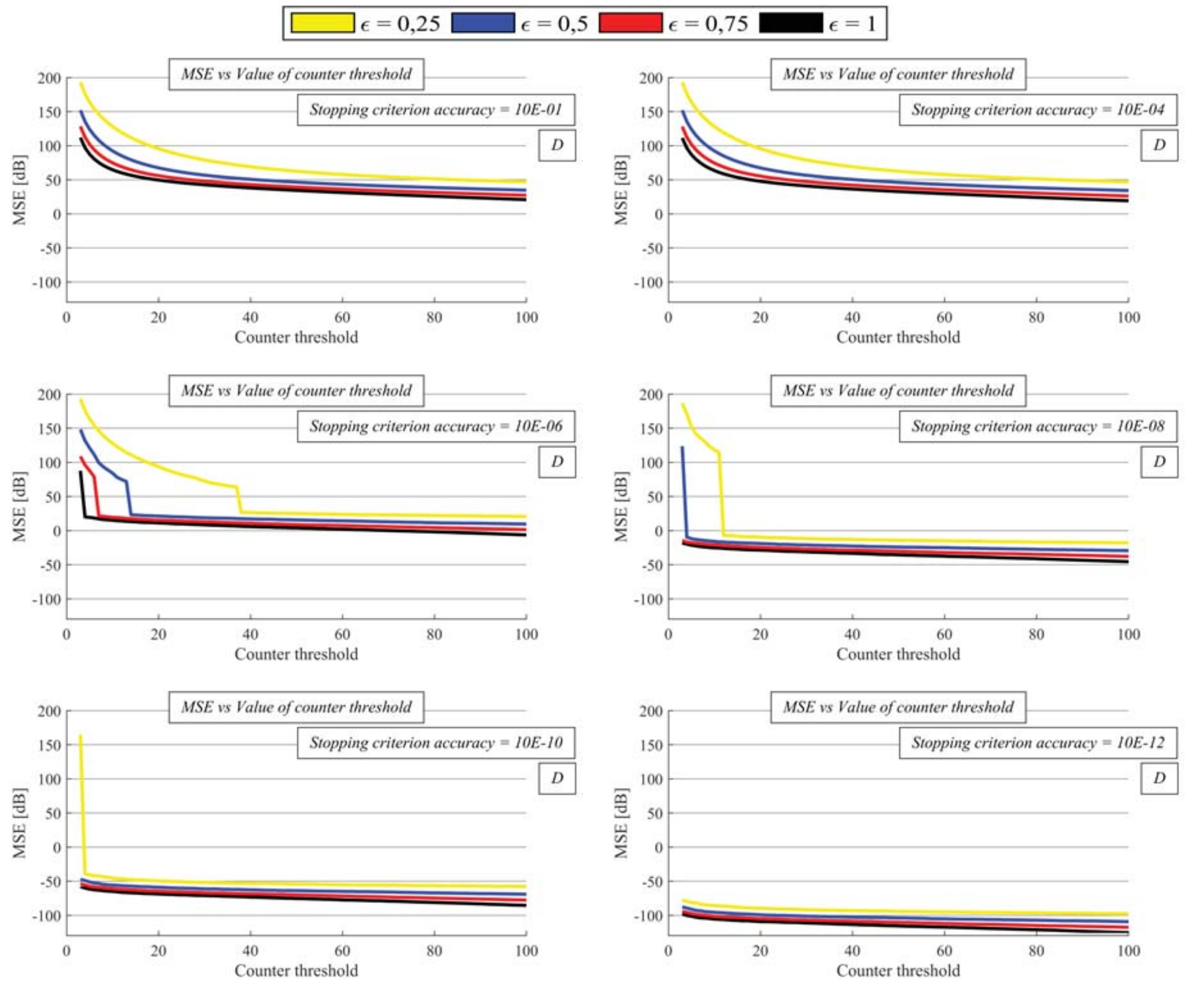

Fig. 1. MSE in decibels averaged over 30 densely connected RGGs

As mentioned above, we assume that each sensor node initiates its inner state with either ”1” (the leader) or ”0”. We appoint the best-connected sensor node as the leader.

As mentioned earlier, we assume that AC is bounded by the stopping criterion from [7], determined by two pre-set constants. In our analyses, they take these values:

- *accuracy* = 10^{-1}, 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}
- *counter threshold* = 3-100 with the step size = 1

We use two metrics for performance evaluation, namely, the mean square error over the iterations (MSE(k)) and the convergence rate expressed as the number of the iterations for the consensus achievement.

MSE is a frequently applied metric for performance evaluation in a wide spectrum of the scientific disciplines and is defined as follows [8]:

$$\text{MSE} = \frac{1}{n} \cdot \sum_{i=1}^{n} \left(x_i(k_{last}) - \mathbf{1}^{\text{T}} \times \frac{\mathbf{x}(0)}{n} \right)^2 \qquad (7)$$

Here, k_{last} is the label of the iteration when the last sensor node completes the algorithm. In the experimental section, we separately analyze MSE averaged over 30 densely connected and 30 sparsely connected RGGs.

The other applied metric is the convergence rate expressed as the number of the iterations. A higher number of the iterations indicates a lower convergence rate. Again, the convergence rate averaged over 30 densely connected and 30 sparsely connected RGGs is separately analyzed.

IV. Experimental section

In the first experiment, we analyze MSE over the iterations in densely connected RGGs (see Fig. 1). From the results, it can be seen that an increase in the mixing parameter ϵ results in lower MSE, and so the precision of the final estimates[7] is higher. Thus, AC with the highest analyzed mixing parameter (i.e., ϵ = 1) achieves the highest performance for each *accuracy* and *counter threshold*. Moreover, it is seen that an increase in *counter threshold* ensures a decrease in MSE for each *accuracy* and ϵ. However, we can see that the precision is very low for *accuracy* = 10^{-1} and *accuracy* = 10^{-4} for each analyzed *counter threshold*. It is because the leader completes AC so soon that the information about its inner state is not sufficiently diffused over the graphs. For *accuracy* = 10^{-6}, 10^{-8}, and 10^{-10}, we can see that there is the value of *counter threshold* for which a rapid increase

[7] the estimates after AC is completed at each sensor node

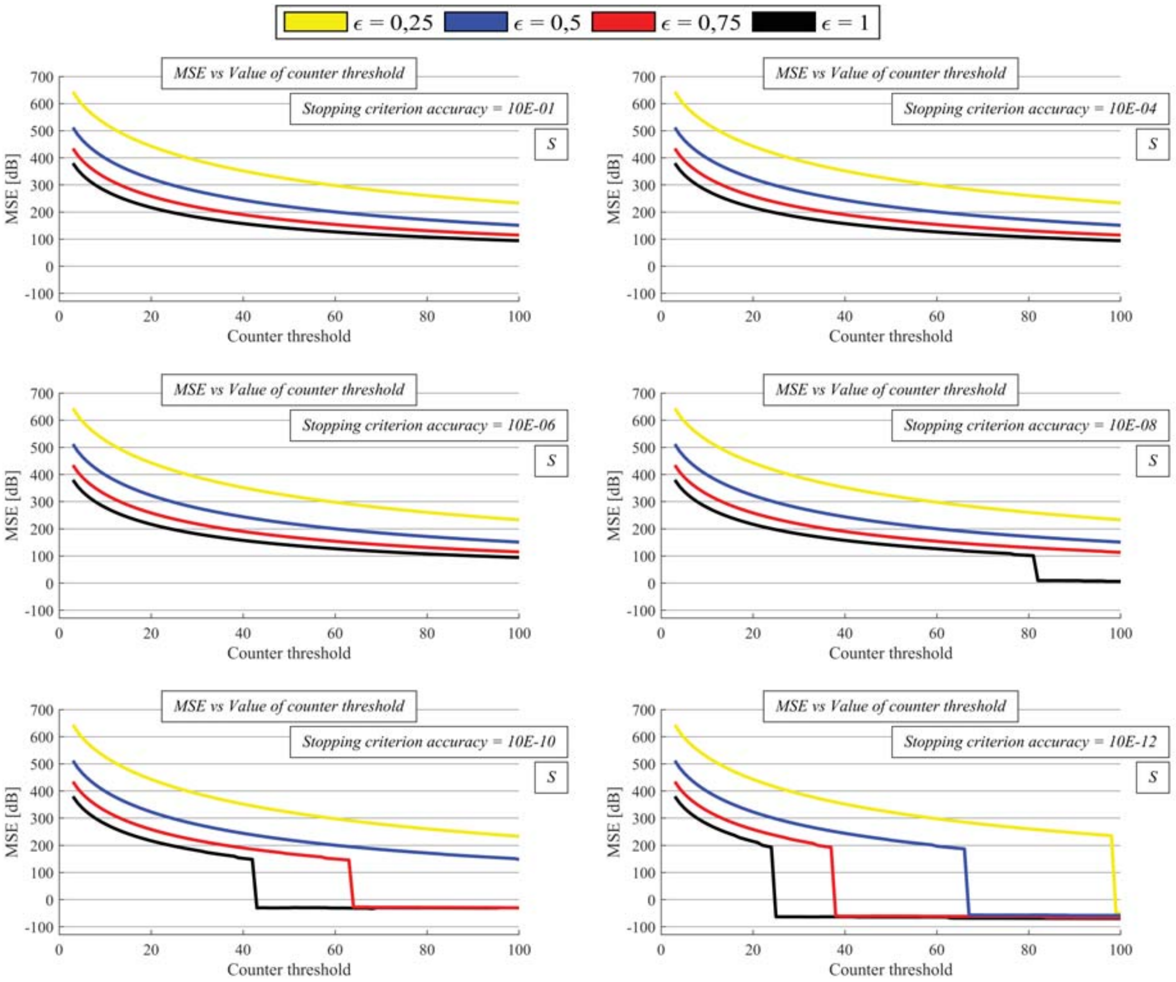

Fig. 2. MSE in decibels averaged over 30 sparsely connected RGGs

in the precision is visible. A higher value of ϵ and a lower value of *accuracy* cause that this rapid precision increase occurs for lower *counter threshold*. This rapid precision increase takes values between approx. 37 dB - 204 dB. This phenomenon is not observed in [8], where the distributed summing with the stopping criterion is analyzed. However, for *accuracy* = 10^{-10}, ϵ = 0.5, 0.75, and 1, high precision is seen for each *counter threshold*. For *accuracy* = 10^{-12}, high precision is achieved by each configuration for each *counter threshold*. Furthermore, we can see that the value of *accuracy* has only a marginal impact on MSE when the precision of the final estimates is small. For those values of *counter threshold* when MSE achieves low values, a decrease in *accuracy* ensures a decrease in MSE.

In the sparsely connected graphs (Fig. 2), the impact of a decrease in *accuracy* and an increase in *counter threshold* and the mixing parameter ϵ have the same character as in the previous analysis. Compared to the results from the dense graphs, the rapid precision increase has the same character, however, it is visible for higher values of *counter threshold* and lower values of *accuracy* (for ϵ = 1, it is observed the first time for *counter threshold* = 82, *accuracy* = 10^{-8} and for ϵ = 0.25, it is even seen the first time for *counter threshold* = 99, *accuracy* = 10^{-12}). It takes significantly higher values than in the dense graphs, and these values are between approx. 92 dB - 286 dB.

The next analysis is concerned with the convergence rate in dense and sparse RGGs. From Fig. 3 and Fig. 4, we can see that an increase in *counter threshold* and a decrease in *accuracy* result in an increase in the iteration number necessary for the consensus and so decelerate the algorithm regardless of the value of the mixing parameter ϵ (the difference between various ϵ is negligible for *accuracy* = 10^{-1}). Moreover, an increase in ϵ ensures a higher convergence rate for each *counter threshold* and *accuracy*. In general, the precision and the convergence rate are higher in the densely connected graphs than in the sparsely connected ones.

Compared to AC for distributed summing with the same stopping criterion [7], AC for a network size estimation is less precise, and similar precision can be reached in a significantly higher number of the iterations. AC for a distributed network size estimation requires almost twice more iterations in the dense graphs and two and a half times more iterations in the sparse graphs to achieve MSE around zero decibels than AC for distributed summing [8].

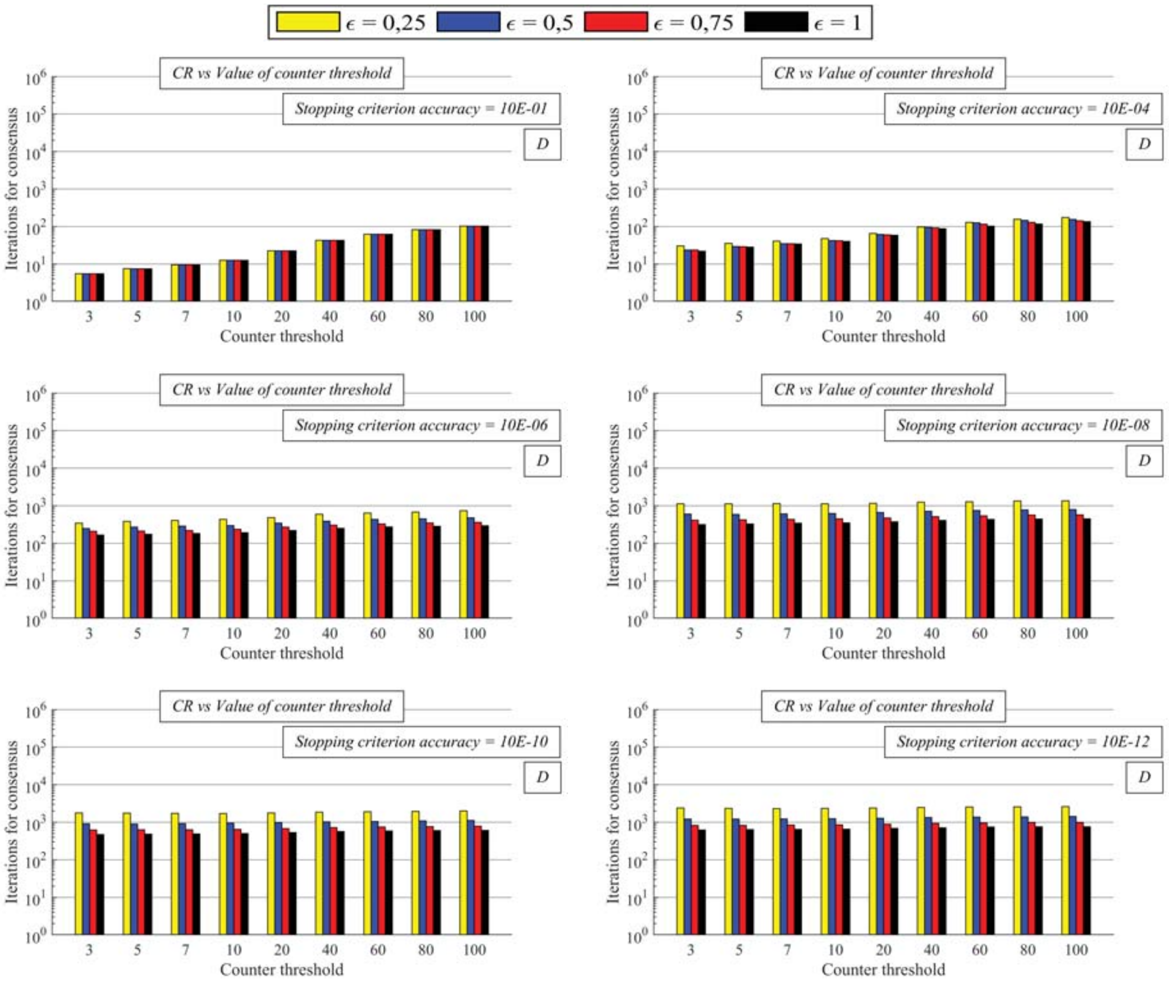

Fig. 3. Convergence rate expressed as number of iterations for consensus averaged over 30 densely connected RGGs

V. Conclusion

We analyze AC with four initial configurations bounded by a stopping criterion with varied parameters proposed for WSNs over RGGs of either dense or sparse connectivity. We show that a decrease in *accuracy* and an increase in *counter threshold* ensure lower MSE (i.e., higher precision of the final estimates) but decelerate the algorithm for each mixing parameter ϵ. However, we can see the interesting phenomenon - for higher *accuracy* and lower values of *counter threshold*, the precision of the algorithm is very low - even making the algorithm unusable. Then, a rapid precision increase dependent on *accuracy*, *counter threshold*, ϵ, and connectivity is observed. For lower *accuracy*, higher ϵ, and connectivity, this increase is visible for lower *counter threshold*. Furthermore, an increase in the mixing parameter ϵ ensures both higher precision and a higher convergence rate. Compared to AC for distributed summing stopped by the same implemented stopping criterion, AC for a network size estimation (the stopping criterion is slightly edited) achieves a significantly lower performance and requires more iterations to achieve similar precision than AC for distributed summing. In general, the precision and the convergence rate are higher in densely connected RGGs than in the sparsely connected ones.

Thus, the final conclusion is that the mixing parameter ϵ taking $1/\Delta$ achieves the highest performance in terms of both the precision of the final estimates and the convergence rate. Moreover, the setting of the stopping criterion parameters depends on whether higher precision or a faster execution of the algorithm is required - this depends on a particular application. Furthermore, higher values of *accuracy* and lower values of *counter threshold* may lead to unusable precision of the final estimates.

References

[1] C. Baquero, P. S. Almeida, R. Menezes, and P. Jesus, “Extrema propagation: Fast distributed estimation of sums and network sizes. ,” IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 4, pp. 668–675, April 2012.

[2] O. Sluciak and M. Rupp, “Network size estimation using distributed orthogonalization,”IEEE Signal Process. Lett., vol. 20, no. 4, pp. 347–350, April 2013.

[3] D. Izadi, J. H. Abawajy, S. Ghanavati, and T. Herawan, “A data fusion method in wireless sensor networks,” Sensors, vol. 15, no. 2, pp. 2964–2979, January 2015

[4] N. Zhao, X. Chi, L. Zhao, and Y. Zhu, ‘A Spectrum Efficient Self-Admission Framework for Coexisting IEEE 802.15. 4 Networks under Heterogeneous Traffics,” Radioengineering, vol. 27, no. 1, pp. 326–334, April 2018

[5] R. Mishra, V. Jha, R. K. Tripathi, and A. K. Sharma, “Design of Probability Density Function Targeting Energy Efficient Network for Coalition Based WSNs,” Wireless Pers. Commun., vol. 99, no. 2, pp. 651–680, March 2018

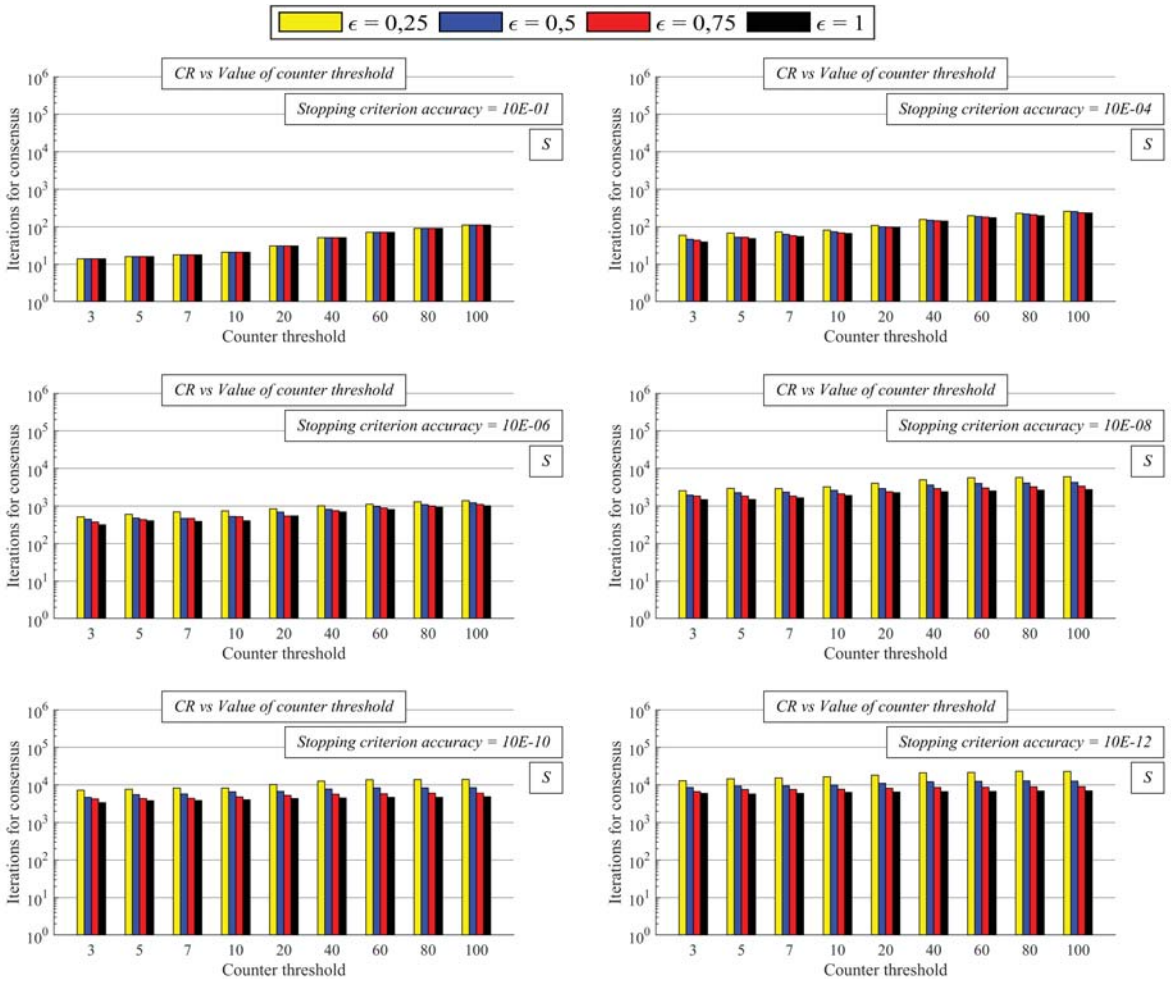

Fig. 4. Convergence rate expressed as number of iterations for consensus averaged over 30 sparsely connected RGGs

[6] N. Ayoub et al., “MAHEE: Multi-hop advance heterogeneity-aware energy efficient path planning algorithm for wireless sensor networks,” In Proc. of PACRIM, Victoria, BC, Canada, 2017, pp. 1–6, Aug. 2017.

[7] J. Kenyeres, M. Kenyeres, M. Rupp, and P. Farkas, “WSN implementation of the average consensus algorithm,” in Proc. of the 17th European Wireless 2011 - Sustainable Wireless Technologies, Vienna, Austria, 2011, pp. 1–8, April 2011.

[8] M. Kenyeres, J. Kenyeres, and I. Budinska “Distributed linear summing in wireless sensor networks,” in Proc. of SAMI, Herlany, Slovakia, 2019, pp. 23–30, Jan. 2019.

[9] V. Schwarz, G. Hannak, and G. Matz, “On the convergence of average consensus with generalized Metropolis-Hasting weights,” in Proc. of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 2014, pp. 5442–5446, May 2014.

[10] L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” Syst. Control Lett., vol. 53, no. 1, pp. 65–78, September 2004.