Can Distributed Consensus Algorithms with Asymptotic Convergence Be Applied to Estimate Size of Multi-agent Systems?

Martin Kenyeres[1,*], Ján Zelenka[1] and Nayden Chivarov[2]
Institute of Informatics, Slovak Academy of Sciences, Slovakia[1]
Institute of Information and Communication Technologies, Bulgarian Academy of Science, Sofia, Bulgaria[2]
martin.kenyeres@savba.sk

Abstract: *Estimating the size of a multi-agent system (MAS) is a fundamental task with applications in routing, resource management, distributed coordination, etc. Achieving accurate network size estimation in MASs is challenging due to the absence of a central authority, lack of global identifiers, dynamic network changes, communication constraints, etc. This paper investigates the applicability of distributed consensus algorithms with asymptotic convergence for estimating MAS size. Specifically, we examine the average consensus algorithm using four different weighting schemes: Maximum Degree (MD), Metropolis-Hastings (MH), Best Constant (BC), and Convex Optimized (OW) weights. Experimental results on a random geometric graph demonstrate that all weighting schemes enable agents' internal states to asymptotically converge to the value 1/n, where n is the total number of agents. Differences among the schemes are observed in convergence speed and early-stage oscillations, with BC providing the fastest convergence and OW exhibiting more initial fluctuations. Overall, the study confirms that distributed average consensus algorithms with the examined weights can effectively estimate network size in MASs.*
Keywords*: Consensus, Distributed algorithms, Multi-agent systems, Network size*

1. Introduction

Estimating the size of a network is essential for various real-world applications, including routing, synchronization, resource management, etc. [1]. However, achieving this in a distributed manner is challenging, as it must be done without central coordination or reliance on global identifiers [2]. Traditional approaches often require excessive communication overhead or make strong assumptions about the network topology [3]. Efficient estimation methods must balance accuracy, scalability, and robustness to dynamic changes in network structure [4].

Estimating the size of a network in multi-agent systems (MASs) is a challenging task due to several fundamental constraints and characteristics of distributed environments [5]. First, as already mentioned, MASs operate without a central authority, meaning no single agent has a global view of the network [6]. Each agent only has access to local information, making it difficult to obtain a global estimate of the network size. As mentioned again, many MASs do not assign unique identifiers to agents, which prevents direct counting. Without global identifiers, agents must rely on indirect methods, such as message passing or consensus algorithms, to estimate the total number of participants. The dynamic nature of these networks further complicates the task, as agents may join or leave the network unpredictably due to failures, mobility, changing conditions, etc. [7]. This constant flux makes it hard to maintain an accurate and up-to-date estimate of the network size. Communication constraints, such as limited bandwidth, energy limitations, and unreliable communication links, also play a significant role. Agents cannot constantly exchange information with all other nodes, leading to incomplete or delayed data for size estimation. Moreover, propagation delays and asynchronous updates cause information to spread gradually, and these delays can lead to outdated or inconsistent estimates across agents. Lastly, as the network grows, efficient estimation methods must avoid excessive communication overhead, with traditional flooding-based approaches becoming infeasible due to exponential message growth in large networks. To address these challenges, researchers have developed distributed estimation algorithms, such as consensus-based approaches, probabilistic sampling methods, etc., which aim to balance accuracy, convergence speed, and resource efficiency [8, 9].

This paper addresses whether distributed consensus algorithms with asymptotic convergence are applicable to estimate the size of MAS. The problem of achieving consensus, a long-standing research topic in computer science, has gained significant attention from both academia and industry in recent years. In general, "consensus" refers to reaching an agreement on a specific value based on the internal states of all agents in MAS [10]. The primary objective of distributed consensus algorithms is to enable agreement among independent, autonomous entities that communicate over potentially unreliable networked systems [11]. Asymptotic convergence in distributed consensus algorithms refers to the property that, as time approaches infinity, the states of all participating agents converge to a common value. In other words, even if perfect consensus is not reached in finite time, the differences between agents' states asymptotically vanish. This ensures long-term agreement across the network despite delays, noise, or limited communication. Over the past decades, consensus algorithms have been widely applied in diverse fields, including wireless sensor networks, robotic systems, unmanned aerial vehicles, clustered satellite constellations, etc. [12]. As shown in [13], the application of distributed consensus algorithms is beneficial in numerous real-world scenarios. In this context, estimating the network size using distributed consensus algorithms corresponds to computing the arithmetic mean, provided that one agent initializes its internal state to one while all others set theirs to zero [14]. The agent with an initial value of one is designated as the leader. Through iterative application of the update rule, the internal state of each agent asymptotically converges to the value $1/n$, where n represents the total number of agents in MAS.

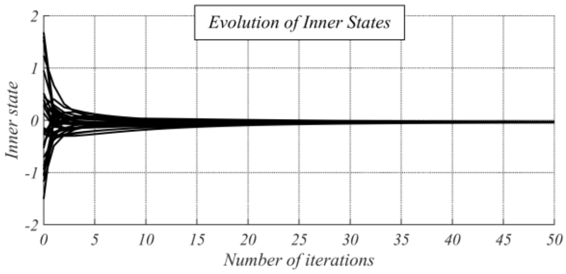

Fig. 1 *Evolution of inner states when distributed consensus algorithm with asymptotic convergence is applied.*

2. Theoretical Background into Topic

In this paper, we choose the average consensus algorithm with four weights (namely, the Maximum degree weights (MD), the Metropolis-Hastings weights (MH), the Best Constant weights (BC), and the Convex Optimized weights (OW)) [14]. An average consensus is a distributed method in which each agent updates its state based on weighted information (affected by the selected weight model) from its neighbors [13]. These weighting schemes guarantee that the resulting update matrix preserves the average of the initial states, enabling convergence to the global average even in networks with non-uniform connectivity. The average consensus relies on iterative neighbor-to-neighbor communication and can therefore be expressed by a difference equation as follows [14]:

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

In this context, **W** denotes the weight matrix, whose elements are determined by the used weights and govern the convergence rate, robustness, and overall stability of the algorithm, as well as its susceptibility to the initial configuration. In the average consensus algorithm, each agent's internal state asymptotically converges to the value of the estimated aggregate function—representing, in our case, the reverse value of the size of the multi-agent system [14].

The first type of weights considered in this study are the MD weights. They are among the simplest and most commonly used weighting schemes in distributed consensus algorithms. They ensure uniform convergence properties across networks with heterogeneous node degrees, as each node uses the same normalization factor corresponding to the global maximum degree. Although this approach provides good stability and predictable convergence behavior, it may lead to suboptimal convergence speed in sparse or highly irregular topologies, where the degree distribution varies significantly among nodes. Their initialization requires prior knowledge of the maximum node degree within the graph. The corresponding weight matrix can be expressed as the Perron matrix, defined as follows [14]:

$$[W]_{ij}^{\mathrm{MD}} = \begin{cases} \frac{1}{\Delta}, & \text{if } e_{ij} \in \mathbf{E} \\ 1 - d_i \cdot \frac{1}{\Delta}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

Here, Δ denotes the maximum degree of the network, i.e., the number of neighbors of the most highly connected node, d represents the degree of the specific node under consideration, e is the label of an edge (the indexes i and j represent particular vertexes in a graph), and **E** is the edge set, i.e., the set gathering all edges in a graph. Another commonly used weighting scheme is the MH weights. These weights are particularly attractive in distributed consensus algorithms because they do not require global knowledge of the network, unlike schemes such as the MD weights. By relying solely on local connectivity, they maintain robustness and guarantee convergence even in irregular or dynamically changing topologies. Furthermore, the MH design ensures that the resulting weight matrix is stochastic and symmetric, which is critical for achieving asymptotic convergence of each agent's state to the global average. These weights require only locally available information for proper initialization. Specifically, each agent needs to know its own degree as well as the degrees of its immediate neighbors. The corresponding weight matrix can thus be defined as follows [14]:

$$[W]_{ij}^{\mathrm{MH}} = \begin{cases} \frac{1}{\max\{d_i, d_j\}+1}, & \text{if } e_{ij} \in \mathbf{E} \\ 1 - \sum_{i \neq j} [W]_{ij}^{\mathrm{MH}}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

The next set of weights under consideration are the Best Constant (BC) weights, which require precise knowledge of the second smallest eigenvalue and the largest eigenvalue of the corresponding Laplacian matrix. These weights are recognized as the fastest among all weight matrices that can be described by a Perron matrix [14]. The weight matrix for BC weights is formally defined as follows [14]:

$$[W]_{ij}^{\mathrm{BC}} = \begin{cases} \frac{2}{\lambda_1(\mathbf{L}) + \lambda_{N-1}(\mathbf{L})}, & \text{if } e_{ij} \in \mathbf{E} \\ 1 - d_i \cdot \frac{2}{\lambda_1(\mathbf{L}) + \lambda_{N-1}(\mathbf{L})}, & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}$$

Here, λs are Laplacian eigenvalues (the largest and the second smallest one in our case), N (also labelable as n) is the graph order, and **L** is the Laplacian of the network graph. By choosing the mixing parameter in this manner, the spectral radius of the update matrix is minimized, which maximizes the convergence rate of the distributed averaging process. The BC weights are particularly effective in networks where the global spectral properties of the corresponding Laplacian matrix are available, allowing for optimal balancing between stability and speed of consensus**.**

The final set of weights of interest in this study are the OW weights. In the context of average consensus, convex optimized weights ensure that the combination of node values at each iteration preserves convexity, i.e., the updated value at each node is a convex combination of its own value and those of its neighbors. This property guarantees that all node states remain within the convex hull of the initial values, preventing divergence and maintaining stability. Moreover, by carefully selecting the weights to minimize the spectral radius of the update matrix, the convergence rate of the distributed averaging process is maximized. This makes OW weights particularly suitable for networks where rapid consensus is critical, such as sensor networks, multi-agent coordination, distributed control systems, etc. Thus, these weights are specifically designed to accelerate the convergence of distributed linear averaging protocols. In particular, the problem of achieving fast average consensus is reformulated as a spectral radius minimization problem, where the objective is to minimize the spectral radius $\rho(\cdot)$ of the weight matrix, as discussed in [14].

$$\begin{array}{ll} \text{minimize} & \rho(\mathbf{W} - \frac{1}{n}\mathbf{1}\mathbf{1}^{\mathrm{T}}) = max\{\lambda_2(\mathbf{W}), -\lambda_n(\mathbf{W})\} \\ \text{subject to} & \mathbf{W} \in \mathbf{S}, \mathbf{1}^{\mathrm{T}}\mathbf{W} = \mathbf{1}^{\mathrm{T}}, \mathbf{W}\mathbf{1} = \mathbf{1} \end{array}$$

Here, **1** is an all-ones vector, and $\mathbf{W} \in \mathbf{S}$ denotes that the weight matrix **W** is subject to a sparsity pattern constraint, where the set **S** is defined as follows [14]:

$$\mathbf{S} = \{\mathbf{W} \in \mathbf{R}^{n \times n} \mid [W]_{ij} = 0 \text{ if } \{i,j\} \notin \mathbf{E} \text{ and } i \neq j\}$$

All these weights are tested in random geometric graph whose topology is depicted in Fig. 2.

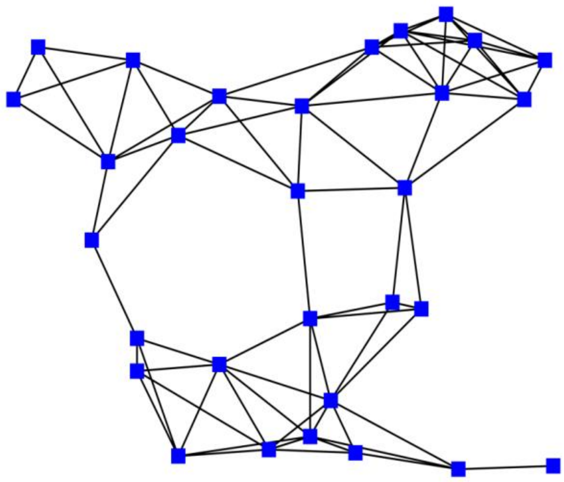

Fig. 2 *Used random geometric graph.*

3. Experimental results

In this section, we present the experimental results obtained in Matlab 2018b. The four figures (Fig. 3 – Fig. 6) illustrate the evolution of inner states over 70 iterations for the examined weights of the average consensus algorithm. In all cases, the inner states start with higher variability and gradually converge toward zero as iterations increase. The MD and MH strategies show a smooth, monotonic decay with minor fluctuations, while the BC approach converges slightly faster, reaching stability around iteration 20. The OW strategy, in contrast, exhibits more oscillations in the early iterations but eventually stabilizes near zero, indicating a more aggressive but oscillatory convergence pattern. Overall, all methods converge to similar stable inner states, but they differ in convergence speed and initial oscillatory behavior. Thus, in every case, the inner states converge to the value $1/n$; therefore, all examined weights can estimate the network size of MASs. These results confirm the robustness of the average consensus algorithm across different weighting schemes. Furthermore, they highlight that the choice of weight significantly affects the convergence rate, which can be crucial for time-sensitive distributed applications.

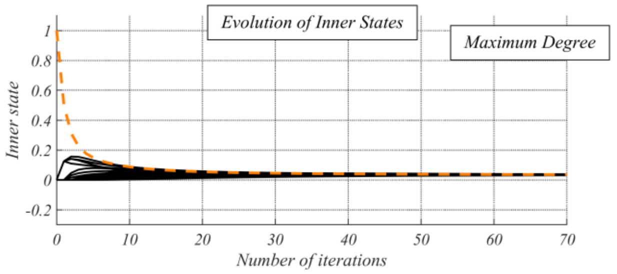

Fig. 3 *Evolution of inner states when MD is applied.*

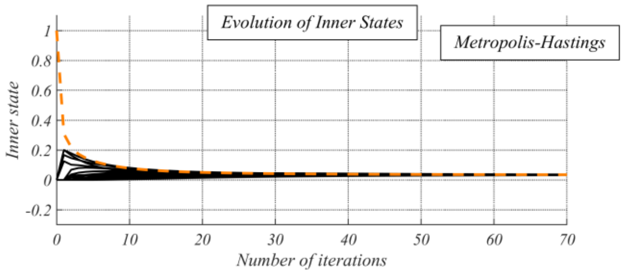

Fig. 4 *Evolution of inner states when MH is applied.*

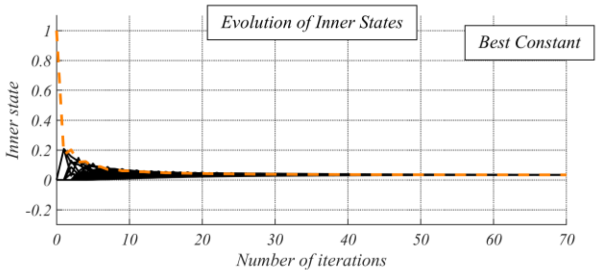

Fig. 5 *Evolution of inner states when BC is applied.*

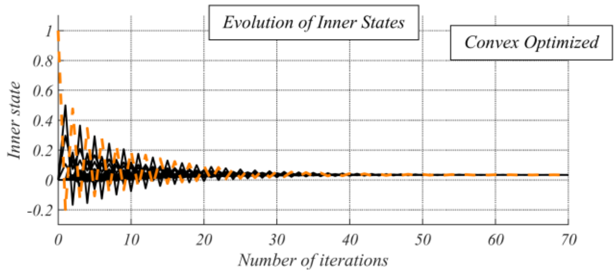

Fig. 6 *Evolution of inner states when OW is applied.*

Conclusion

This paper explored the use of distributed average consensus algorithms for estimating the size of multi-agent systems (MASs) under realistic constraints, including lack of global identifiers, dynamic network changes, and limited communication. Four weighting schemes—Maximum Degree (MD), Metropolis-Hastings (MH), Best Constant (BC), and Convex Optimized (OW)—were analyzed in terms of convergence behavior and stability. Experimental results demonstrated that all four methods successfully enable agents' internal states to asymptotically converge to $1/n$, effectively estimating the network size. While BC weights provided the fastest convergence and OW weights exhibited initial oscillations, MD and MH offered smoother, more stable convergence. These findings confirm that distributed consensus-based approaches are a viable and efficient solution for network size estimation in MASs, balancing accuracy, robustness, and communication efficiency. Future work may focus on optimizing convergence speed in highly dynamic or large-scale networks and exploring adaptive weighting strategies for further performance improvement.

Acknowlegment

This work was supported by the Slovak Scientific Grant Agency VEGA under the contract 2/0135/23 “Intelligent sensor systems and data processing”, by the Slovak Research and Development Agency under the contract No. SK-SRB-23-0038 and No. APVV23-0292, and by Cyber-Physical System for environmental monitoring and data analysis (BAS-SAS-2024-01).

References

1. Lee, D., Kim, T., Lee, S., & Shim, H. (2024). Fully decentralized design of initialization-free distributed network size estimation. *arXiv preprint arXiv:2401.07472*.
2. Terelius, H., Varagnolo, D., & Johansson, K. H. (2012, December). Distributed size estimation of dynamic anonymous networks. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)* (pp. 5221-5227). IEEE.
3. Chatterjee, S., Pandurangan, G., & Robinson, P. (2019, May). Network size estimation in small-world networks under Byzantine faults. In *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)* (pp. 855-865). IEEE.
4. Chen, L., Karbasi, A., & Crawford, F. W. (2016). Estimating the size of a large network and its communities from a random sample. *Advances in neural information processing systems*, *29*.
5. Kenyeres, M., & Kenyeres, J. (2021). Comparative study of distributed consensus gossip algorithms for network size estimation in multi-agent systems. *Future Internet*, *13*(5), 134.
6. Lee, D., Kim, T., Lee, S., & Shim, H. (2024). Fully decentralized design of initialization-free distributed network size estimation. *arXiv preprint arXiv:2401.07472*.
7. Deplano, D., Franceschelli, M., & Giua, A. (2021). Dynamic min and max consensus and size estimation of anonymous multiagent networks. *IEEE Transactions on Automatic Control*, *68*(1), 202-213.
8. Habibi, G., Kingston, Z. K., Wang, Z., Schwager, M., & McLurkin, J. (2015, May). Pipelined Consensus for Global State Estimation in Multi-Agent Systems. In *AAMAS* (pp. 1315-1323).
9. Nken, A. T. A., Mckeever, S., Corcoran, P., & Ullah, I. (2025, September). Probabilistic Sampling with Frobenius Norm for Action Recognition. In *2025 IEEE International Conference on Image Processing (ICIP)*. IEEE.
10. Lee, D., Lee, S., Kim, T., & Shim, H. (2018, December). Distributed algorithm for the network size estimation: Blended dynamics approach. In *2018 IEEE Conference on Decision and Control (CDC)* (pp. 4577-4582). IEEE.
11. de Galland, C. M., & Hendrickx, J. M. (2022). Fundamental performance limitations for average consensus in open multi-agent systems. *IEEE Transactions on Automatic Control*, *68*(2), 646-659.
12. Makridis, E., Grammenos, A., Oliva, G., Kalyvianaki, E., Hadjicostis, C. N., & Charalambous, T. (2024, December). Average consensus over directed networks in open multi-agent systems with acknowledgement feedback. In *2024 IEEE 63rd Conference on Decision and Control (CDC)* (pp. 3051-3056). IEEE.
13. Kenyeres, M., Kenyeres, J., & Hassankhani Dolatabadi, S. (2025). Distributed consensus gossip-based data fusion for suppressing incorrect sensor readings in wireless sensor networks. *Journal of Low Power Electronics and Applications*, *15*(1), 6.
14. Kenyeres, M., Kenyeres, J., & Budinská, I. (2021). On performance evaluation of distributed system size estimation executed by average consensus weights. In *Recent advances in soft computing and cybernetics* (pp. 15-24). Cham: Springer International Publishing.