The Upsides of Turbulence: Baselining Gossip Learning in Dynamic Settings

Antonio Di Maio
University of Bern, Switzerland
antonio.dimaio@unibe.ch

Mina Aghaei Dinani, Gianluca Rizzo
HES-SO Valais, Switzerland and Università di Foggia, Italy
name.surname@hevs.ch

ABSTRACT

In dynamic settings, fully distributed gossip-based learning schemes have recently gained interest due to their better scalability, robustness, and enhanced privacy protection compared to server-based architectures. However, existing approaches to their performance characterization either assume stable connectivity among nodes or are ad-hoc for specific trace-based mobility patterns. Thus, in dynamic settings, there is currently a poor understanding of the conditions under which gossip-based learning schemes are feasible, and of their main performance tradeoffs. In this work, we start addressing this issue by performing a first baselining of Gossip Learning (GL) on random Time-Varying Graphs (TVG), to get a first-order characterization of their main performance patterns in dynamic settings. The use of random TVG enables a fine-grained and accurate characterization of GL effectiveness as a function of the main system parameters while abstracting from scenario-specific features of patterns of communication and mobility (e.g., induced by road grids or measured mobility traces). Our results suggest that GL schemes are robust to node mobility and comparable in accuracy and convergence speed to Federated Learning architectures, over a wide range of operational conditions. We show that the final model accuracy is robust against data dispersion across nodes as well as against very low rates of exchanges across nodes.

KEYWORDS

Gossip Learning, Mobile Networks, Opportunistic Communications.

1 INTRODUCTION

The increasing amount of data generated in the network, due to the pervasive diffusion of the Internet of Things (IoT) and Smart City paradigms, is gradually pushing the bulk of the computing load required to elaborate these data towards the edge of the network [1]. This trend is at the origin of the recent interest in distributed Machine Learning (ML) [2] which, by removing the need to transfer data to the cloud, enables better privacy protection while potentially decreasing bandwidth utilization. However, many distributed schemes scale poorly, as they rely on infrastructure-based information exchanges or a central coordination server. These issues are at the origin of the growing interest towards fully distributed schemes, such as Gossip Learning (GL) [3–5] (also denoted as Decentralized Federated Learning [6]). They are based on a server-less, fully distributed model training approach, and on knowledge transfer and reuse among agents via direct, peer-to-peer exchanges of models. This allows for saving infrastructure-based computing and communication resources. They scale well with the number of agents, as each agent contributes to service capacity not only with data but also by adding computing and communication capacity. Several GL schemes have been proposed for a large variety of learning architectures ([4, 5]), scenarios and applications [7–9]. However, the majority of these schemes assume *static* network topologies (e.g., ring [10] or mesh [5, 11, 12]). In such networks, [4] shows that GL can converge and deliver comparable accuracy to that of server-based distributed schemes such as Federated Learning (FL). However, these results do not apply to dynamic settings, where node mobility implies a network topology that changes over time, such as in Vehicle-to-Vehicle (V2V) networks or robot swarms. Indeed, the network edge is progressively including a variety of devices (i.e., smartphones, Terrestrial and Unmanned Aerial Vehicles, IoT devices) characterized by high mobility and substantial sensing and computing capabilities. [13] assesses GL schemes in dynamic scenarios in which nodes are always fully interconnected, thus forming a stable topology. [8, 14] proposes GL schemes for fragmented and volatile vehicular networks, showing that they can converge and achieve high accuracy in various trace-driven mobility scenarios. However, these assessments are limited to very specific trace-based mobility patterns and scenarios, which are hard to extrapolate to other settings. As such, they do not offer insights into the relationship between the main system parameters and GL's key performance indicators. Specifically, when nodes move, it is currently unclear what are the operating conditions in which GL schemes converge and deliver satisfactory performance and how their performance compares to Centralized Learning or other distributed schemes such as Federated Learning. In the present work, we perform a first step toward addressing the aforementioned issues. We investigate the feasibility of GL in dynamic settings by elaborating a first characterization of the basic mechanisms affecting its performance on random TVG, as a function of the main system parameters and the main structural properties of the time-varying network. The use of synthetic graphs enables a fine-grained and accurate system characterization that abstracts from context-specific spatiotemporal patterns of communication and mobility, such as those found in measurement-based mobility traces. Specifically, our main contributions are:

- We characterize the performance of GL on dynamic random graphs, based on a GL scheme that generalizes Federated Learning (FL) [15] to fully decentralized dynamic settings, including FL as a special case. We show that GL schemes are robust to node mobility over a very wide range of scenarios, regarding the number of nodes and frequency of inter-node contacts. To the best of our knowledge, we are the first to characterize and assess GL feasibility over non-trivial, yet non-trace-driven connectivity patterns.
- We determine the impact of the main system parameters on GL performance. We show that, even in dynamic settings, GL

- accuracy and convergence speed are comparable to those of centralized Federated Learning schemes.
- We show that the final model accuracy is robust against data dispersion across nodes as well as against very low rates of exchanges across nodes.

These results suggest that node mobility and the lack of coordination among nodes do not cause a performance penalty in GL compared to centralized architectures, such as FL, over a very broad set of system configurations.

2 SYSTEM MODEL

We consider a set V of mobile nodes modeling, e.g., smartphones, UAVs, and connected vehicles. We assume that each node is endowed with an ML model whose architecture is equal for all nodes and which needs to be trained and used by each node to perform a specific inference task. Assuming the same ML architecture for every model in the system is necessary to make model aggregation possible, as the aggregation operations between models are only possible between parameter vectors of the same dimension. Each node is also endowed with a set of data points, denoted as *local dataset.* We assume nodes can communicate directly among themselves through wireless Device-to-Device (D2D) communications, e.g. via Dedicated Short-Range Communications (DSRC), or Bluetooth Low Energy [14]. Communication between two nodes $v_i, v_j \in V$ may occur whenever they are in *contact*, i.e., within the transmission range of each other. We assume time is divided into intervals called *slots*, indexed with $t \in \mathbb{N}$.

2.1 Time-Varying Graph Model

Algorithm 1 Basic GL algorithm. The ML model weights and the set of neighbors of node v in time slot t are denoted by w_t^v and K_t^v, respectively. The loss of node k's model on node v's dataset is denoted by l_k^v and its formulation is task-specific (e.g., cross-entropy for classification tasks).

```
1: $w_0 \leftarrow$ INITIALIZE()
2: for $\forall v \in V$ do $w_0^v \leftarrow w_0$
3: loop                                  ▷ $\forall v \in V$ executes this loop in parallel, start from $t \leftarrow 0$
4:     $w_t^v \leftarrow$ TRAIN($w_t^v$)
5:     for $\forall k \in K_t^v$ do Send $w_t^v$ to $k$, Receive $w_t^k$ from $k$
6:     for $\forall k \in K_t^v \cup \{v\}$ do Compute $l_k^v$
7:     $w_{t+1}^v \leftarrow \left( \sum_{k \in K_t^v \cup \{v\}} w_t^k 2^{-l_k^v} \right) \cdot \left( \sum_{k \in K_t^v \cup \{v\}} 2^{-l_k^v} \right)^{-1}$     ▷ MERGE
8:     $t \leftarrow t+1$
```

We model the mobile nodes' connectivity graph and its evolution over time as a TVG, composed of a set V of nodes and a set E_t of edges between nodes, which varies over time. An edge $\in E_t$ models the existence of a direct wireless channel between two nodes. We assume the graph to be constant within each time slot and to (possibly) vary only from one slot to the following one. The resulting dynamic graph, denoted as $G = \{G_t = (V, E_t) : t \in \mathbb{N}\}$, is thus a sequence of graphs, each associated with a time slot. The volatility and dynamicity of the wireless channel are modeled by the fact that the set of edges E_t can be different at each slot. These dynamic graphs are often used to model opportunistic gossiping schemes because they simplify assumptions about the network structure while still capturing key characteristics of real-world networks. In particular, they allow varying the number of nodes, the connectivity patterns between nodes, and the frequency and duration of node interactions in a controlled and systematic way. In this paper, we model G as an Erdős-Rényi dynamic graph [16], a type of uniform random graph. Specifically, at any time slot t and for any two nodes $v_i, v_j \in V$, the probability p that an edge exists between them is the same for any i, j, t. This assumption gives the graph a homogeneous structure, ensuring stationary patterns of evolution over time. We further assume that the edges are *conditionally independent* of each other, i.e., $\forall t \in \mathbb{N}, v_i, v_j \in V$: $\mathbb{P}[(v_i, v_j) \in E_t] = \mathbb{P}[(v_i, v_j) \in E_t | (v_i, v_j) \in E_{t-1}] = p$. The choice of p determines the graph's degree of connectivity at any time slot and, thus, the rate at which new connections are established and terminated between nodes, which is a key aspect of network dynamicity. Moreover, p also determines the mean duration of a link between any two nodes and, thus, the edge density and clustering degree of the graph. We chose Erdős-Rényi dynamic random graphs because they are among the simplest dynamic random graph models of realistic D2D network topologies that provide mathematical guarantees on key graph metrics such as the average number of edges $\mathbb{E}\left[|E_t|\right] = p\binom{|V|}{2}, \forall t \in \mathbb{N}$, the nodes' degree distribution $\mathbb{P}\left(d(v) = k\right) = \binom{|V|-1}{k} p^k \left(1-p\right)^{|V|-1-k}$, and the p threshold for the almost-sure graph's connectivity $p > \frac{\ln |V|}{|V|}$. Assuming conditional independence of edges allows modeling "worst case" edge dynamics, where changes in connectivity patterns are abrupt and fast. In particular, mobile networks' connectivity can be modeled with a random graph when the GL message exchange dynamics are considerably slower than the nodes' physical mobility dynamics, which is our assumption in this work.

2.2 Gossip Learning Operation

We detail the operation of the basic GL algorithm (Algorithm 1) run by each node. In this work, we assume all models are initialized with identical random weights, which has been shown to improve convergence speed and accuracy [15]. Starting from $t = 0$, in every time slot, the algorithm proceeds through three *phases*, which we assume to be synchronized across nodes. In the first phase (*training*), each node trains its local model instance on the local dataset. In the second phase (*exchanging*), each node sends its local model instance to its neighbors and receives their local instance. In this work, we assume model exchanges to be instantaneous. However, our approach can be easily extended to consider finite exchange durations. Finally, in the third phase (*merging*), each node *merges* the models received from neighbor nodes with its local model (i.e., it computes a linear combination of them) to produce a *meta-model*, similarly to what parameter servers do in centralized FL algorithms [15]. The weights of the merging operation are computed via the *Decentralized Powerloss* (DP) strategy [8], where each weight is a function of the loss computed over the context-specific validation set. The exact formula for the weights is shown in Algorithm 1, line 7. We chose the DP merging strategy as it has proven superior performance to other state-of-the-art merging approaches [8, 14]. These three phases are repeated until a termination criterion is met. For instance, after a maximum number of iterations is attained or when the average model's accuracy exceeds a threshold. In this work, we assumed the termination criterion is met when the global model's accuracy does not improve more than a fixed threshold over a fixed number of gossip rounds.

3 PERFORMANCE EVALUATION

3.1 Simulation Setup

To assess the performance of GL schemes on time-varying graphs, we considered the case in which a set V of homogeneous nodes in the system need to train an ML model to perform an inference task such as handwritten digit recognition (MNIST dataset [17], $m = 10$ classes, image size $n = 28$) or object recognition (CIFAR-10 dataset [18], $m = 10$ classes, image size $n = 32$) from a set of images. We assume that each node in the system is endowed with a *local dataset* of equal size for all nodes and that each data point in the system can belong to at most one local dataset. All local datasets are i.i.d. and partitioned in 85% training set and 15% validation set. Let us denote the union of all local datasets in the system as the *global dataset* of size γ. This study assumes the global dataset is built as a random sample of γ dataset samples from one of the two source datasets (MNIST or CIFAR-10). By varying γ, we thus modulate the total amount of data in the system available for the collaborative model training. To each global dataset, we associate a *global test set* obtained by random sampling 20% of the source datasets and ensuring that the global dataset and test set are disjoint. We assume that the global dataset is equally distributed across all nodes in the scenario, meaning that each node's local dataset size is $\gamma/|V|$. This choice allows us to assess the impact of information fragmentation on GL performance for a fixed global dataset size.

To perform both inference tasks, we assume that nodes use supervised models trained with Mini-Batch Stochastic Gradient Descent (SGD) with Categorical Cross-Entropy loss, early-stopping patience of 20 epochs, batch size of 32 [19], momentum of 0.9, and a 10^{-4} learning rate [20]. Specifically, we assumed every node in the system executes a Convolutional Neural Network (CNN), whose architecture and hyperparameters (identical for all nodes) are as follows. Layer 1 (input) is a 2D Convolution with 32 filters and 3x3 kernel. Layer 2 is a 2x2 Max Pooling. Layer 3 is a 100-neuron Dense layer with ReLu activation. Finally, Layer 4 (output) is a 10-neuron Dense layer with SoftMax activation. We choose this architecture as it is widely recognized as effective in extracting shape features from images [21]. Further hyperparameter optimization is out of the scope of this work and is left to future investigation.

We compare the performance of GL with the following baselines:

- *Centralized Learning*, where a server collects the local datasets from all participants, aggregates them into a global dataset, and locally trains a global model on it.
- *Local Learning*, in which each node trains its local model using only its local dataset without exchanging data or models with other nodes or a centralized server. It is derived from our GL reference scheme by considering TVGs with $p = 0$. When $p = 0$, each node's local model does not change over time compared to the one trained at $t = 0$.
- *Federated Learning* [15], in which a server collects the models from all participants at each iteration, aggregates them, and redistributes the aggregated model to all participants. For comparison, we assume models are aggregated with Decentralized Powerloss (as in Algorithm 1, line 7). FL can be derived as a special case of our GL scheme when applied to TVGs with $p = 1$ (i.e., complete graphs).

We verified that an early-stopping patience of 20 epochs for the local training was sufficient to detect convergence accurately. We assume the random coefficients of the ML model trained at $t = 0$ to be the same at all nodes. Indeed, [15] shows that models trained from independent random initializations lead to different weights minima, leading to less accurate meta-models when the independently trained models are merged. One key performance metric of our system is the *accuracy* A_t of each node's local model at time slot t, defined as the fraction of the model's correct predictions out of all predictions for the global test set. We denote a node's accuracy at $t = 0$, before gossiping starts, as A_0 (*initial accuracy*), and a node's accuracy at convergence as A_C (*final accuracy*). Another important performance indicator of our GL schemes is the gossip *convergence time* C, defined as the number of time slots required for the system to converge. We assumed our schemes to have converged when the local models' accuracy (averaged across all nodes) did not improve by more than 0.5% over the last 10 rounds (gossip patience).

3.2 Simulation Results

Figure 1 shows the detrimental effect of *information fragmentation* (i.e., the partitioning of the global dataset into $|V|$ local datasets) on the average accuracy of isolated local models, and how distributed learning schemes, such as Federated Learning and Gossip Learning can mitigate them. Information fragmentation also introduces an upper bound to the average accuracy achievable through distributed learning schemes, which decreases as the fragmentation (i.e., the number of nodes $|V|$) increases. We call *fragmentation loss* the distance between the Centralized Learning accuracy and the maximum Gossip Learning accuracy (i.e., achieved in scenarios with $p = 1$, equivalent to centralized Federated Learning). We show that the more a network is connected (i.e., larger p), the larger the opportunity for distributed-learning schemes to improve local model accuracy by aggregating other models' knowledge from external local datasets. Larger global dataset sizes increase accuracy for all compared learning methods (Centralized, Local, Centralized Federated, Gossip) and all topologies (number of nodes $|V|$ and connectivity p, confirming the expected behavior. In Figure 1, the green curve (Centralized Learning) is the same for all $|V|$, as it only depends on the global dataset size, and achieves the highest accuracy compared to the other baselines because the model is trained with all available information in the system. As the number of nodes $|V|$ increases, the size of the local dataset decreases, resulting in the gap between the accuracy A_C for centralized and local learning schemes becoming proportionally larger due to information fragmentation. As the probability of connection p increases, the network topology becomes more connected and each node has a higher expected number of neighbors (namely, in the order of $p|V|$) at each time slot with which to exchange trained local models and perform aggregation (gossiping). We call *gossip opportunity* the distance between the local learning curve $p = 0$ and the Federated Learning curve $p = 1$. We observe that, for every number of nodes and any global dataset size, the higher the probability of connection p the greater the average accuracy A_C of local models, showing

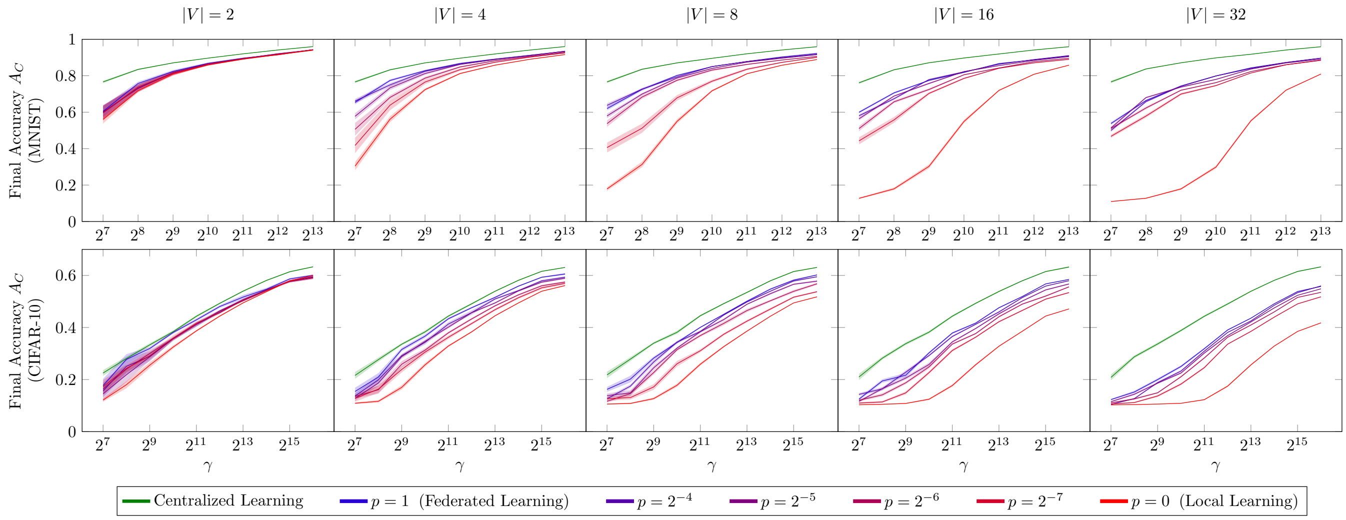


Figure 1: Impact of global dataset size γ, number of nodes $|V|$, and probability of connection p on the models' final accuracy A_C at convergence. Curves are mean central tendencies for distributions of A_C for each user in the system, surrounded by an asymptotic confidence interval at 95% level. The top row reports the plots for the MNIST dataset, whereas the bottom row reports the plots for the CIFAR-10 dataset. Measurements are aggregated over 15 repetitions.

the positive impact of network connectivity on average model accuracy. Even though the highest accuracy is always achieved in fully-connected scenarios (i.e., $p = 1$), the "speed" at which the accuracy improves from $p = 0$ scenarios to $p = 1$ scenarios (i.e., the function between p and the achievable accuracy) depends on $|V|$. In particular, we observe that both gossip opportunity and fragmentation loss increase with $|V|$, and that the distance between the accuracy for $p = 1$ scenarios and the accuracy for the other simulated scenarios (i.e., $0 < p \leq 1$) decreases as $|V|$ increases. This shows that for some values of connection probability p, the higher the fragmentation $|V|$, the closer the accuracy is to the maximum achievable.

Figure 2 shows the knowledge diffusion properties of GL for varying global dataset size and information fragmentation. We observe that, as $|V|$ increases, the initial accuracy's variability is progressively less explanatory for the final accuracy's variability. This effect shows that the larger the network size $|V|$, the more GL schemes can disseminate the knowledge contained in the nodes' trained local models throughout the system, providing each node with a local model with similar accuracy performance. We observe that the global dataset size is positively correlated with both initial and final accuracy and a moderate clustering effect around lower values of initial accuracy especially for lower global dataset sizes. This effect is due to the modest size of local datasets for scenarios where γ is small and $|V|$ is large, where local datasets do not have any sample for several classes to predict (for example, in MNIST, local datasets that do not include any dataset sample for certain digits).

Figure 3 shows a matrix heatmap in which each cell is the Spearman's correlation coefficients $\rho_{i,j}$ between a pair of system parameters or performance metrics i and j, across all collected data. Scenarios with a higher number of nodes, and therefore a higher information fragmentation, require a statistically higher number of gossip rounds to converge. We observe a negative correlation between the number of nodes $|V|$ and both the initial and final accuracy, and a positive correlation between the number of nodes $|V|$ and the difference between the final and initial accuracy. This evidence supports what Figure 1 suggested, i.e., that a higher information fragmentation harms the final accuracy by introducing a *fragmentation loss*, but it also represents the biggest opportunity for GL to improve model accuracy over the local training case. Furthermore, higher information fragmentation provides nodes with smaller local datasets, which reduces their initial accuracy.

Across different scenarios, the probability of connection p appears less than weakly correlated with the convergence time C, final accuracy A_C, and accuracy gain $A_C - A_0$ due to gossiping, i.e., $|\rho_{p,C}|, |\rho_{p,A_C}|, |\rho_{p,A_C - A_0}| < 0.15$. However, stratifying the collected data by different values of $|V|$ revealed stronger correlations between p and the three metrics C, A_C, and $A_C - A_0$ for some values of $|V|$. For example, for $|V| = 4$ we observe $\rho_{p,A_C - A_0} = 0.34$ and a $\rho_{p,C} = 0.16$, while for $|V| = 32$ we observe $\rho_{p,A_C - A_0} = 0.07$ and a $\rho_{p,C} = -0.17$. This effect shows that the probability of connection p impacts the convergence time C and the accuracy gain due to gossiping $A_C - A_0$ differently depending on the information fragmentation.

Figure 4 shows that larger global datasets reduce convergence time on average, that larger network sizes $|V|$ increase the convergence time average and standard deviation, and that models with higher initial accuracy tend to converge faster. This evidence matches what is shown by the correlation coefficients in Figure 3. However, stratifying the data by the size of the global dataset reveals that the initial accuracy does not influence the overall system's convergence speed, as evidenced by the negligible slope observed in the regression lines.

Figure 5 shows in the top row that, for a large range of different global dataset sizes, the probability of connection has a very limited

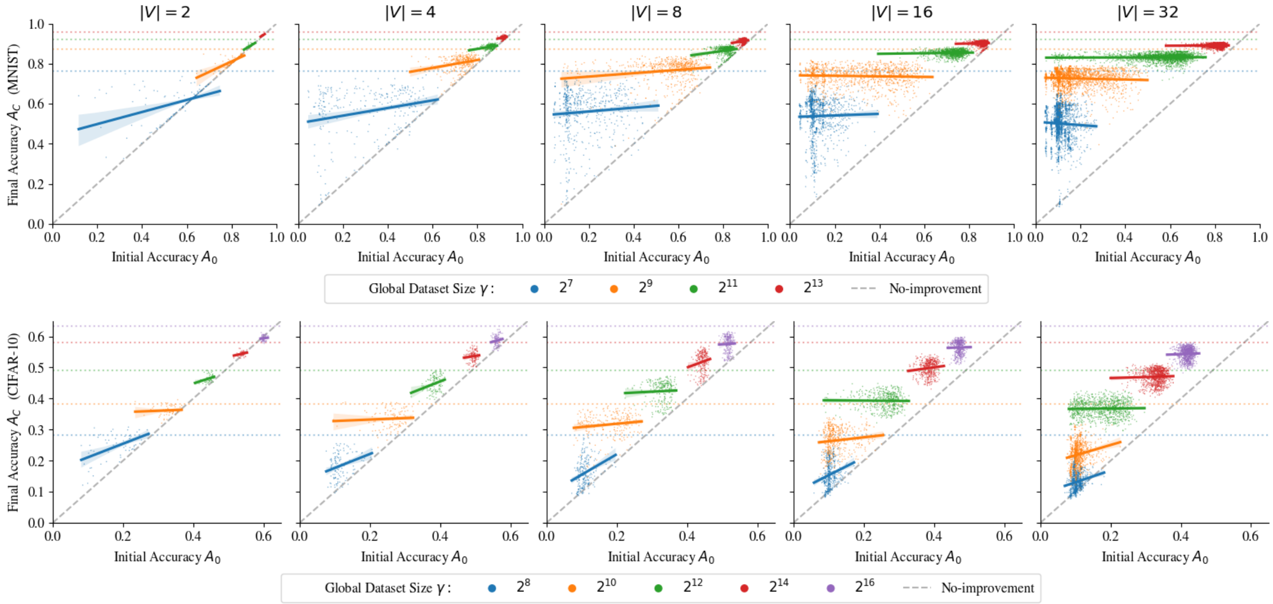
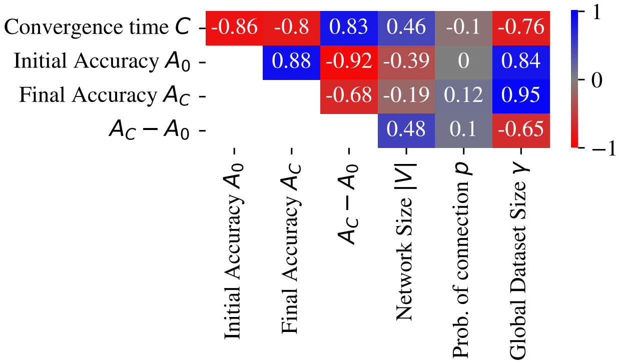


Figure 2: Relationship between global dataset size γ, initial accuracy A_0 (before the gossiping starts), and final accuracy A_C at gossip convergence for a varying number of nodes in the system. Each point in the scatter plot represents the initial and final accuracies for one user in the system. The top row reports the plots for the MNIST dataset, whereas the bottom row reports the plots for the CIFAR-10 dataset. Measurements are aggregated over 15 repetitions. For each global dataset size γ and number of nodes $|V|$, a linear regression line is shown and surrounded by a 95% confidence band computed using bootstrapping. The gray dashed line represents the "no improvement" line, where a node's accuracy at convergence is the same as its accuracy before gossiping. The dotted lines represent the average accuracy of a centralized model trained on the global dataset.



	Initial Accuracy A_0	Final Accuracy A_C	$A_C - A_0$	Network Size $\lvert V\rvert$	Prob. of connection p	Global Dataset Size γ
Convergence time C	-0.86	-0.8	0.83	0.46	-0.1	-0.76
Initial Accuracy A_0		0.88	-0.92	-0.39	0	0.84
Final Accuracy A_C			-0.68	-0.19	0.12	0.95
$A_C - A_0$				0.48	0.1	-0.65

Figure 3: Spearman's correlation coefficient matrix between system parameters (network, dataset) and performance metrics (accuracy, convergence) for the MNIST dataset results. Each correlation coefficient belongs to a 95% confidence interval computed using the Fisher transform, with a difference between the upper and lower bound always < 0.022. Correlations between system parameters are zero and hidden.

impact on the convergence speed, as the associated ECDFs are very close and their confidence intervals often overlapping. Conversely, Figure 5 shows in the bottom row that, independently from the probability of connection, the global dataset size has a much larger impact on convergence time, as the ECDFs significantly shift to lower values for larger dataset sizes. Compared to Figure 4, Figure 5 stratifies the convergence time analysis by the probability of connection, highlighting its limited impact on the metric.

4 CONCLUSION

In this article, we performed a first assessment of Gossip Learning in a class of dynamic networks modeled by Time-Varying Graphs, advancing the knowledge in the field, which so far mainly targeted static networks. Our results suggest that Gossip Learning significantly enhances the average model accuracy compared to local learning in networked systems whose topology varies over time or when communication capacity between nodes is constrained. Gossip Learning improves accuracy over non-gossip learning along with the number of nodes in the network for a fixed global dataset, highlighting Gossip Learning's scalability. We show that Gossip Learning can provide network nodes with an accuracy comparable with that of more communication-intensive distributed learning schemes, even a very modest probability of connection between network nodes.

REFERENCES

[1] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE communications surveys & tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[2] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *ACM Comput. Surv.*, vol. 53, mar 2020.

[3] R. Ormándi, I. Hegedűs, and M. Jelasity, "Gossip learning with linear models on fully distributed data," *Concurrency and Computation: Practice and Experience*,

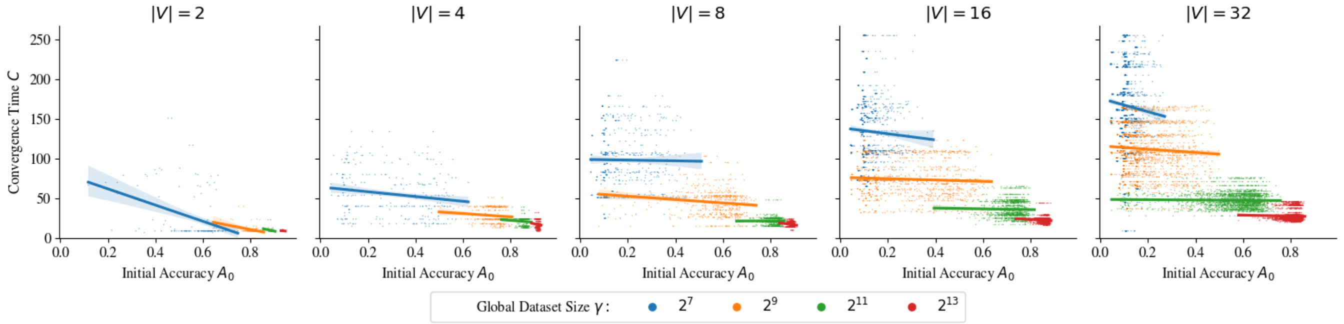


Figure 4: Scatter plot of convergence time C against initial accuracy A_0, for the MNIST dataset. Each point represents one user in the system. We show a linear regression line for the points belonging to each global dataset size γ and number of nodes $|V|$, surrounded by a 95% confidence band computed with bootstrapping.

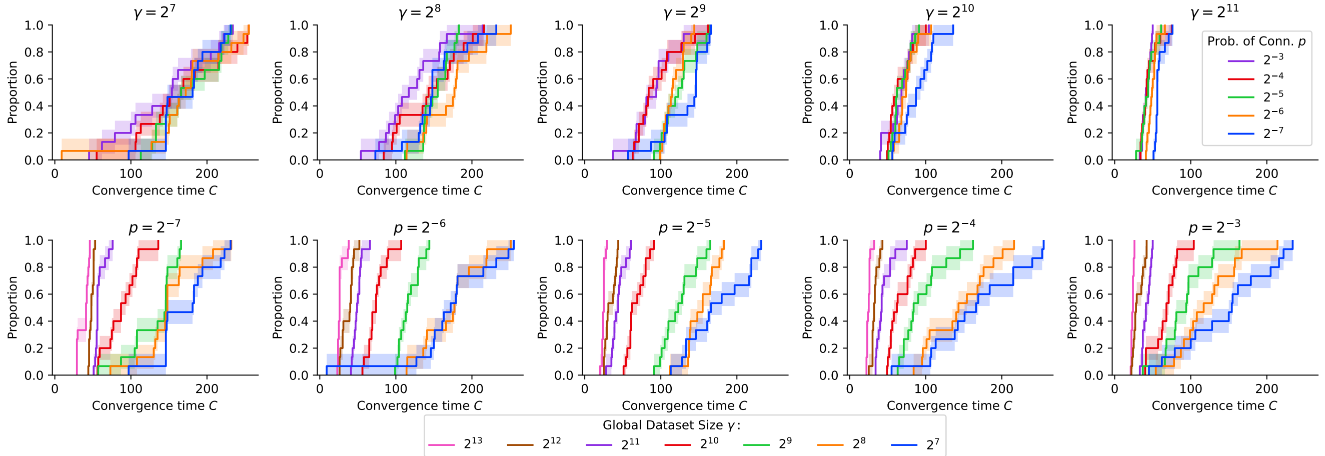


Figure 5: Empirical Cumulative Distribution Functions (ECDFs) of convergence time C for $|V| = 32$ number of vertices and MNIST dataset, at different values of global dataset size γ and probability of connection p. The top row groups data by global dataset size, while the bottom row by probability of connection. Confidence intervals are Kolmogorov-Smirnov bounds at 95% level.

vol. 25, no. 4, pp. 556–571, 2013.

[4] I. Hegedűs, G. Danner, and M. Jelasity, "Decentralized learning works: An empirical comparison of gossip learning and federated learning," *Journal of Parallel and Distributed Computing*, vol. 148, pp. 109–124, 2021.

[5] V. Zantedeschi, A. Bellet, and M. Tommasi, "Fully decentralized joint learning of personalized models and collaboration graphs," in *International Conference on Artificial Intelligence and Statistics*, pp. 864–874, PMLR, 2020.

[6] E. T. M. Beltrán, M. Q. Pérez, P. M. S. Sánchez, S. L. Bernal, G. Bovet, M. G. Pérez, G. M. Pérez, and A. H. Celdrán, "Decentralized federated learning: Fundamentals, state-of-the-art, frameworks, trends, and challenges," *arXiv preprint arXiv:2211.08413*, 2022.

[7] C. Li, G. Li, and P. K. Varshney, "Decentralized federated learning via mutual knowledge transfer," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1136–1147, 2022.

[8] M. A. Dinani, A. Holzer, H. Nguyen, M. A. Marsan, and G. Rizzo, "Gossip learning of personalized models for vehicle trajectory prediction," in *IEEE WCNC Workshop*, pp. 1–7, IEEE, 2021.

[9] A. A. Alkathiri, L. Giaretta, S. Girdzijauskas, and M. Sahlgren, "Decentralized word2vec using gossip learning," in *23rd Nordic Conference on Computational Linguistics (NoDaLiDa 2021)*, 2021.

[10] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[11] A. G. Roy, S. Siddiqui, S. Pölsterl, N. Navab, and C. Wachinger, "Braintorrent: A peer-to-peer environment for decentralized federated learning," *arXiv preprint arXiv:1905.06731*, 2019.

[12] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi, "Personalized and private peer-to-peer machine learning," in *International Conference on Artificial Intelligence and Statistics*, pp. 473–481, PMLR, 2018.

[13] N. Majcherczyk, N. Srishankar, and C. Pinciroli, "Flow-FL: Data-driven federated learning for spatio-temporal predictions in multi-robot systems," in *IEEE ICRA*, pp. 8836–8842, IEEE, 2021.

[14] M. A. Dinani, A. Holzer, H. Nguyen, M. A. Marsan, and G. Rizzo, "Vehicle position nowcasting with gossip learning," in *IEEE WCNC*, pp. 728–733, IEEE, 2022.

[15] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*, pp. 1273–1282, PMLR, 2017.

[16] A. Chaintreau, A. Mtibaa, L. Massoulie, and C. Diot, "The diameter of opportunistic mobile networks," in *Proceedings of the 2007 ACM CoNEXT conference on - CoNEXT '07*, (New York, New York), p. 1, ACM Press, 2007.

[17] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov. 1998. Conference Name: Proceedings of the IEEE.

[18] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," *Master Thesis, University of Toronto, CA*, 2009.

[19] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural networks: Tricks of the trade*, pp. 437–478, Springer, 2012.

[20] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, 2019.

[21] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, pp. 1–74, 2021.