BayGO: Decentralized **Bay**esian Learning and Information-Aware **G**raph **O**ptimization Framework

Tamara AlShammari, *Student Member, IEEE*, Chathuranga Weeraddana, *Member, IEEE*, and Mehdi Bennis, *Fellow, IEEE*

***Abstract*—Multi-agent Decentralized Learning (MADL) is a scalable approach that enables agents to learn based on their local datasets. However, it presents significant challenges related to the impact of dataset heterogeneity and the communication graph structure on learning speed, as well as the lack of a robust method for quantifying prediction uncertainty. To address these challenges, we propose BayGO, a novel fully-decentralized multi-agent local Bayesian learning with local averaging, usually referred to as non-Bayesian social learning, together with graph optimization framework. Within BayGO, agents locally learn a posterior distribution over the model parameters, updating it locally using their datasets and sharing this information with their neighbors. We derive an aggregation rule for combining received posterior distributions to achieve optimality and consensus. Moreover, we theoretically derive the convergence rate of agents' posterior distributions. This convergence rate accounts for both network structure and information heterogeneity among agents. To expedite learning, agents employ the derived convergence rate as an objective, optimizing it with respect to the network structure alternately with their posterior distributions. As a consequence, agents can successfully fine-tune their network connections according to the information content of their neighbors. This leads to a sparse graph configuration, where each agent communicates exclusively with the neighbor that offers the highest information gain, enhancing communication efficiency. Our simulations corroborate that the BayGO framework accelerates learning compared to fully-connected and star topologies owing to its capacity for selecting neighbors based on information gain.**

***Index Terms*—Bayesian learning, distributed learning, social learning, graph optimization, information heterogeneity, KL divergence, multi-agent systems.**

Manuscript received 12 September 2023; revised 23 February 2024; accepted 26 March 2024. Date of publication 10 April 2024; date of current version 24 April 2024. This work was supported in part by the European Commission under Grant 101095363 (Horizon Europe SNS JU ADROIT6G Project), in part by the European Union through the Projects 6G-INTENSE under Grant 101139266, in part by the Research Council of Finland (former Academy of Finland) 6G Flagship Programme under Grant 346208, and in part by the European Commission through the DESIRE6G (G.A. 101096466). A preliminary version of this work appears in the proceedings of IEEE ICC 2021 [DOI: 10.1109/ICC42927.2021.9500613]. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiao Fu. *(Corresponding author: Tamara Alshammari.)*

The authors are with Centre for Wireless Communication, University of Oulu, 90570 Oulu, Finland (e-mail: Tamara.Alshammari@oulu.fi; Chathuranga.Weeraddana@oulu.fi; Mehdi.Bennis@oulu.fi).

Digital Object Identifier 10.1109/TSP.2024.3387277

I. Introduction

NOWADAYS, many personal devices have powerful computational resources with a sufficient amount of data for locally training machine learning (ML) models. However, the local data of each agent is statistically insufficient to achieve satisfactory ML model performance. Moreover, sharing local data among agents violates privacy concerns. To address this issue, Federated Learning (FL) was proposed for training a global model without sharing private data. Instead, training is accomplished by sharing only the agents' local model parameters with the aid of a centralized server (i.e., centralized FL) [2], [3] or in a peer-to-peer manner over a decentralized graph. The latter is usually referred to as multi-agent decentralized learning (MADL) [4], [5].

The emergence of many distributed ML applications together with privacy and scalability concerns make MADL more appealing than centralized parameter-server based FL. MADL mandates agents to communicate with a subset of their neighbors, leveraging scalability, under communication constraints [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. In the context of Bayesian inference, agents adopt an approach akin to Bayesian principles by introducing a posterior distribution over a parameter space representing the unknown global model. This work specifically focuses on the Bayesian extension of MADL framework, where agents optimize their posterior distributions over the model parameters alternately with the communication graph, facilitating accelerated convergence, particularly in heterogeneous data settings. Furthermore, our modeling inherently accounts for uncertainty in model parameter predictions. In what follows, we discuss related works in MADL through the lens of Bayesian inference.

A. Related Works

Applying true Bayesian learning within MADL settings is intractable[1]. Instead, agents conduct local Bayesian updates, which are followed by a learning rule commonly termed aggregation [6], [10], [17], [18], [19]. During the aggregation step, local models from each agent are merged with those of its neighbors. This process is known as non-Bayesian social learning [7], [8], [9], [11], [12].

[1] Therefore, it is crucial to emphasize that the terminology Bayesian MADL pertains to the Bayesian update conducted locally at each agent, followed by an aggregation step.

In the literature, two variants of social learning have been proposed: non-adaptive social learning [12], [13], [14], [20] and adaptive social learning [21], [22], [23], [24]. The key distinction lies in the inclusion of a step-size adaptation parameter in adaptive social learning, which regulates the weighting assigned to recent observations relative to past observations. In particular, it has been demonstrated that the adaptation parameter governs a fundamental trade-off between the steady-state learning capability of an algorithm and its adaptability [21]. As a result, in adaptive settings, the steady-state error probability is non-zero [23], in contrast to non-adaptive social learning, where the error probability converges to zero almost surely [14]. In this paper, our settings align with non-adaptive social learning.

The studies in non-adaptive social learning [8], [12], [20] show that local Bayesian model updates performed by agents, when subjected to repeated interactions, can result in successful information aggregation. In particular, the updates yield an asymptotic convergence of local models to the true parameter, reaching optimal consensus. When aggregating information at an agent, the contribution from each neighbor is determined exclusively by weights associated with the corresponding edges of the communication graph. However, these works [6], [7], [8], [9], [10], [11], [12], [18], [19] have not examined the advantages of graph optimization to enhance the performance and convergence results.

In contrast, the authors in [13], [14], [15], [25] derive convergence rates and give insights into the potentials of weights of the communication graph for yielding a faster convergence. For example, [13], [14], [15] provides theoretical guarantees on the convergence of decentralized multi-agent Bayesian learning, where they analytically characterize the convergence rate of agents' posterior distributions as a function of the communication graph and agents' local heterogeneous dataset. More specifically, the expressions of the convergence rates rely on (i) the eigenvector centrality of the agents which is determined by *the structure of the weighted graph* and (ii) *the information content of agents' local datasets* which is measured by using the Kullback-Leibler (KL) divergence between the relative entropies of the marginal likelihood distributions among agents. Even though the works [13], [14], [15], [25] highlight the significance of the weighted graph for faster convergence, they do not investigate weight adaptation and optimization for potential improvements. In particular, the characterization of convergence rates in [13], [14], [15], [25] does not allow means of directly optimizing the convergence rate with respect to the weights of the underlying graph.

However, employing a weight adaptation of the underlying graph to expedite the convergence rate is of significant importance. This is investigated in the work by [22], demonstrating that enabling multi-agent systems to dynamically adjust their graph weights in real-time makes agents allocate either smaller or larger weights to their neighbors based on respective contributions to the inference task. In essence, the focus of [22] is on weight adaptation within multi-agent systems. Specifically, it focuses on scenarios where all agents receive streaming data from a fixed global distribution and the data is corrupted with additive noise. Moreover, the noise levels experienced by agents are different and consequently, the received data of an agent can be less or more noisier than that of another agent. To mitigate this, they propose the Hasting rule [*cf.* Lemma 12.2 in [22]], which enables each agent to adjust the weights assigned to interactions with neighbors based on their respective noise levels. This suggests that the agents' heterogeneity arises from the differences in the additive noise levels experienced by the agents despite that they all receive streaming data from the same global distribution. Whereas in prior works [13], [14], [15], [25], agents experience the same noise power, yet their data distributions might not be identical, or their signal structures may differ in informativeness [14]. Alternatively, agents may exhibit variation in the representation of the global dataset $\mathcal{D}$ within their local datasets $\mathcal{D}_i$, due to differences in the sizes of these local datasets. This results in a setting where the agents' heterogeneity emerges from variations in their local observations prior to the addition of noise. Our work focuses on optimizing the graph weights to accommodate the heterogeneity of datasets prior to the addition of noise, whereas the work in [22] addresses the optimization of the graph weights to accommodate the heterogeneity among agents in terms of their noise power. Therefore, the application of the Hasting rule [*cf.* Lemma 12.2 in [22]] for graph weights construction may not be ideal in systems where agents' heterogeneity originates from inherent disparities in their datasets before the introduction of noise. Specifically, under this notion of heterogeneity, the Hasting rule would treat all agents as uniformly informative due to their equal noise level, thus overlooking the inherited discrepancies in their local datasets.

B. Contributions and Organizations

The main contribution of this paper lies in introducing an alternating minimization framework, in which agents iteratively optimize their posterior distributions and network connections. Specifically, we derive a convergence rate of agents' posterior distributions which depends on both *the communication graph* and *the heterogeneity in information among agents*. What sets this approach apart from previous research is its flexibility in optimizing the derived convergence rate by taking into account the notion of heterogeneity in terms of agents' local model updates. This reflects the informativeness of their local datasets before the introduction of noise, as well as their past interactions within the network. This flexibility enables the alternating optimization of both the posterior distributions and the communication graph, leading to an enhanced learning speed. To provide a concise overview of our contributions, we summarize them as follows:

- We introduce BayGO, a fully-decentralized multi-agent Bayesian learning framework, which iteratively updates the agents' posterior distributions and the underlying communication graph, resulting in accelerated consensus and improved communication efficiency (**Algorithm 1**).

- We analytically derive an aggregation rule for the posterior distributions received from each agent, with the goal of attaining optimality and consensus (**Proposition 1**).
- We theoretically derive the rate of convergence of the agents' posterior distributions. This rate of convergence is influenced by the network's structure, represented by the weights of the connecting edges between agents, and the KL-divergences between the adjacent agents' posterior distributions. These KL-divergences reflect the differences in information content between neighboring agents; that is, the information heterogeneity throughout the network (**Theorem 2**).
- We optimize the derived rate of convergence with respect to the communication graph, alternately with the posterior distributions. This optimization results in a sparse graph configuration where each agent is connected solely to one neighbor at any particular instance. Consequently, this approach leads to an expedited learning and enhanced communication efficiency (**Section III-D**).
- We empirically substantiate that BayGO outperforms several baselines, such as the fully-connected and star topologies in terms of learning speed. This is attributed to the information-aware neighbor selection, which underscores the communication efficiency of BayGO (**Section IV**).

We would like to note that a preliminary version of this work was proposed in [1]. The main differences of this work over our prior work [1] are: (i) this work involves detailed derivations of the posterior distributions aggregation rule, (ii) we derive the convergence rate of the posterior distributions and use it as our graph optimization objective to boost the learning speed, (iii) finally, this work includes several simulation results based on neural networks.

Paper Organization: The rest of the paper is organized as follows. In section II, we introduce the system model and problem formulation. In section III, we describe our alternating minimization based algorithm to solve the proposed problem. In section III-C, we state our analytical results. In section IV, we introduce and discuss our simulation results. We conclude the paper in section V. Finally, in the appendix, we provide the mathematical proofs for our main results.

Notation: We use boldface lowercase symbol for vectors $\boldsymbol{s}$, and boldface uppercase symbol for matrices $\boldsymbol{S}$. In addition, for a probability distribution $\boldsymbol{p}$, $[p]_k$ notation is used to indicate a discrete probability distribution; i.e., it denotes the probability at point $\boldsymbol{\theta}_k$. Meanwhile, $p(\boldsymbol{\theta})$ notation denotes a continuous probability distribution over $\boldsymbol{\theta}$. Super-index will generally indicate the time index. We write as $[\boldsymbol{S}^t]_{ij}$ the i-th row and the j-th column entry of matrix $\boldsymbol{S}$ at time t. For a sequence of matrices $\{\boldsymbol{S}^t\}$, we let $\boldsymbol{S}_{t_f:t_i} \triangleq \boldsymbol{S}_{t_f} \cdots \boldsymbol{S}_{t_i+1}\boldsymbol{S}_{t_i}$ for all $t_f \geq t_i \geq 0$. Moreover, we refer to the Kullback-Leibler (KL) divergence between two probability distributions as $D_{\text{KL}}(\boldsymbol{p}(r)||\boldsymbol{p}'(r))$ such that $\boldsymbol{p}(r), \boldsymbol{p}'(r) \in \Delta R$ where ΔR denotes the set of all probability distributions on a set R. In addition, $\mathbb{N}_0$ denotes the set of natural numbers $\{1, 2, 3, \cdots\}$. Finally, $\mathcal{N}(\boldsymbol{m}, \boldsymbol{C})$ denotes a Gaussian distribution with mean vector $\boldsymbol{m}$ and covariance matrix $\boldsymbol{C}$.

II. System Model and Problem Formulation

Consider a set $\mathcal{M} = \{1, 2, \ldots, M\}$ of M agents, each holding a local dataset $\mathcal{D}_i$ whose elements are pairs of inputs and corresponding labels. In particular, $\mathcal{D}_i$ is a subset of $\mathcal{X}_i \times \mathcal{Y}$, where $\mathcal{X}_i$ is the local input space of agent i and $\mathcal{Y}$ is the set of all possible labels. Note that the union of all local input spaces is contained in a global input space $\mathcal{X}$, i.e., $\cup_{i=1}^{M} \mathcal{X}_i \subset \mathcal{X}$. The global input samples of $\mathcal{X}$ are assumed to be independent and identically distributed (i.i.d) according to a distribution $P_X \in P(\mathcal{X})$, where $P(\mathcal{X})$ is the set of all probability distributions over $\mathcal{X}$. Similarly, Agent i's local input samples, $\{\boldsymbol{x}_i^{(1)}, \boldsymbol{x}_i^{(2)}, \ldots, \boldsymbol{x}_i^{(D_i)}\}$ of $\mathcal{X}_i$ are assumed to be i.i.d according to a distribution $P_i \in P(\mathcal{X})$, and the corresponding labels are $\{y^{(1)}, y^{(1)}, \ldots, y^{(D_i)}\}$. Moreover, for all $\boldsymbol{x} \in \mathcal{X}$, the generating function of the global labels is viewed as a probabilistic model with a distribution $f(\,\cdot \mid \boldsymbol{x})$. In addition, for simplicity, and without loss of generality, we discretize the parameter space $\Theta \subseteq \mathbb{R}^d$ with K representative points and denote the set of these points by Θ_K[2].

For fixed $\boldsymbol{x} \in \mathcal{X}$, $y \in \mathcal{Y}$, and for some parameter vector $\boldsymbol{\theta} \in \Theta_K$ the global likelihood function is given by $l(y|\boldsymbol{x}, \boldsymbol{\theta})$. Similarly, the local likelihood function $l_i(y|\boldsymbol{x}, \boldsymbol{\theta})$ of agent i is defined as follows:

$$l_i(y|\boldsymbol{x}, \boldsymbol{\theta}) \triangleq \begin{cases} l(y|\boldsymbol{x}, \boldsymbol{\theta}), & \text{if } (\boldsymbol{x}, y) \in \mathcal{D}_i \\ \beta, & \text{if } (\boldsymbol{x}, y) \notin \mathcal{D}_i, \end{cases}$$

where $\beta \in (0, 1)$. This definition entails the likelihood function of agent i to be identical to the global likelihood function only when the data sample $(\boldsymbol{x}, y)$ is part of its local dataset $\mathcal{D}_i$.

The posterior distribution of agent i is denoted by $\boldsymbol{\mu}_i \in \Delta\Theta$, where $\Delta\Theta$ is the probability simplex or the set of all probability distributions over the set Θ. In an iterative setting, we use a superscript t with $\boldsymbol{\mu}_i$ to signify the time index. The goal of each agent i, which we refer to as the *per-agent* objective, is to learn a posterior distribution $\boldsymbol{\mu}_i^*$ that minimizes the KL divergence between the true labeling function $f(\,\cdot \mid \boldsymbol{x})$ and its predictive distribution $\sum_{k=1}^{K} l_i(\,\cdot \mid \boldsymbol{x}, \boldsymbol{\theta}_k)[\boldsymbol{\mu}_i]_k$. More specifically, $\boldsymbol{\mu}_i^*$ is the solution to the following optimization problem [13]:

$$\underset{\boldsymbol{\mu_i} \in \Delta\Theta}{\text{minimize}} \;\; \mathbb{E}_{\boldsymbol{x} \sim P_X}\left[D_{\text{KL}}\left(f(\,\cdot \mid \boldsymbol{x}) \middle\| \sum_{k=1}^{K} l_i(\,\cdot \mid \boldsymbol{x}, \boldsymbol{\theta}_k)[\boldsymbol{\mu}_i]_k\right)\right]. \tag{1}$$

The minimization above ensures that each agent makes statistically similar predictions as the true labeling function over the global dataset. In applied domains, it is commonplace to tackle the problem under the setting of *global identifiability* as defined below:

Definition 1: A decentralized multi-agent learning model is said to be globally identifiable if there exists $\boldsymbol{\theta}^* \in \Theta_K$ such that $l_i(y|\boldsymbol{x}, \boldsymbol{\theta}^*) = f(y|\boldsymbol{x})$ $\forall i \in \mathcal{M}$ and $(\boldsymbol{x}, y) \in \mathcal{D}_i$ [13], [20].

In other words, under the global identifiability setting, each agent's goal is to learn the true model parameter $\boldsymbol{\theta}^*$. Since we

[2] Similar assumptions have been previously made in [10], [15], [20].

assume that the model is globally identifiable, the per-agent objective in (1) boils down to learning $\boldsymbol{\theta}^*$ by minimizing the divergence between the true labeling function $f(y|\boldsymbol{x}) = l_i(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ [*cf.* Definition 1] and the likelihood at $\boldsymbol{\theta}$, $l_i(y|\boldsymbol{x}, \boldsymbol{\theta})$ [13], [20]. More specifically, we have $\boldsymbol{\theta}^* = \boldsymbol{\theta}_{k^*}$, where

$$k^* = \underset{k \in \{1,\ldots,K\}}{\operatorname{argmin}} \ \mathbb{E}_{\boldsymbol{x} \sim P_X} \left[D_{\mathrm{KL}} \left(f(\cdot \mid \boldsymbol{x}) \middle\| l_i(\cdot \mid \boldsymbol{x}, \boldsymbol{\theta}_k) \right) \right]. \tag{2}$$

It is worth pointing out that under the global identifiability setting, $[\boldsymbol{\mu}^*]_k = 1$ if $k = k^*$ and zero otherwise. Addressing the aforementioned problem in isolation does not lead to an optimal solution. This is primarily because of the significant challenges inherent in a fully decentralized learning paradigm, which can be summarized as follows.

- *Statistical Insufficiency of Local Datasets*: Agents' locally observed data are likely to be less rich than the global training set. In other words, agents' local datasets are statistically insufficient to learn the shared global model in isolation. In problem (1), this limitation is evident as each agent i only has access to its own distribution P_i, rather than the overall distribution P_X.
- *Limited Communication Resources*: For communication efficiency and scalability reasons, agents are likely to limit their interactions with a subgroup of their peers which can be viewed as their one-hop neighbors on the physical graph.
- *Information heterogeneity:* Information received from different peers should be viewed differently, requiring a heterogeneous updates aggregation mechanism.

Statistical insufficiency suggests agents to collaborate with other agents to enrich the view of the global dataset. On the other hand, the lack of communication resources suggests sparse and effective communication among neighbors, possibly with only the one-hop neighbors. Information heterogeneity suggests designing means of who is communicating to whom to realize a meaningful model aggregation. Thus, instead of the restricted isolated formulation (2), we consider the following optimization problem[3]:

$$\underset{k \in \{1,\ldots,K\}}{\text{minimize}} \ \sum_{j=1}^{M} [\boldsymbol{W}]_{ij} \, \mathbb{E}_{\boldsymbol{x} \sim P_j} \, D_{\mathrm{KL}} \left(f(\cdot \mid \boldsymbol{x}) \middle\| l_j(\cdot \mid \boldsymbol{x}, \boldsymbol{\theta}_k) \right). \tag{3}$$

Note that the weights $[\boldsymbol{W}]_{ij}$s are introduced into the modified formulation to capture the importance of each agent as seen by the others. The weights account for information heterogeneity and give means of differently weighting the information received from neighbors.

III. Distributed Learning With Learning Rate Optimization

In this section, we derive the Bayesian estimation learning rule for problem (3). We also derive the rate of convergence of our model in terms of both communication graph and level of informativeness among agents. Then, we optimize this rate with respect to the communication graph alternately, together with the model to increase the learning speed.

A. Modeling Agents' Communication

To enable designing a suitable method for solving problem (3) in our considered multi-agent setting, we model the agents' interactions by using a graph. More specifically, the interactions between agents at each time epoch t are modeled by the communication graph, $\mathcal{G}_c^t = (\mathcal{M}, \mathcal{E}_c^t)$, which is overlaid over a strongly-connected physical graph[4] $\mathcal{G}_p = (\mathcal{M}, \mathcal{E}_p)$. The set of edges $\mathcal{E}_p$ contains pairs of physically connected agents on $\mathcal{G}_p$. That is the set of pairs $\{(i,j),(j,i)\} \in \mathcal{E}_p$ if and only if agent i and agent j are permitted to communicate with each other through a physical communication link. The set of edges $\mathcal{E}_c^t$ contains $(j,i) \in \mathcal{E}_p$ if and only if agent j is communicating to agent i at time t.

Let us denote by $\mathcal{S}_i$ the set of neighbors of agent i in $\mathcal{G}_p$, i.e., $\mathcal{S}_i = \{j \in \mathcal{M} \mid (j,i) \in \mathcal{E}_p\}$. Similarly, let $\mathcal{V}_i^t$ denotes the set of neighbors of agent i in $\mathcal{G}_c^t$, i.e., $\mathcal{V}_i^t = \{j \in \mathcal{M} \mid (j,i) \in \mathcal{E}_c^t\}$. We directly associate $[\boldsymbol{W}^t]_{ij}$ used in problem (3)[5] to the weight of the directed edge (i,j) from agent j to agent i, which in turn models the real interaction between the two agents. It is worth noting that, for all $j \in \mathcal{M}$, $[\boldsymbol{W}^t]_{ij}$ represents the level of importance as seen by agent i when it blends the posterior distributions of agent j at time t. In other words, $[\boldsymbol{W}^t]_{ij}$ can be viewed as the influence of agent j on the learning process of the posterior distribution of agent i at time t. Recall that, each agent i influences its own learning via the self-loop weight $[\boldsymbol{W}^t]_{ii}$, which is referred to as self-reliance. In the literature, the weight matrices sequence $\{\boldsymbol{W}^t\}$ has been assumed to satisfy some technical connectivity conditions that are used in convergence analysis of distributed averaging algorithms [10]. To this end, we have the following assumptions regarding the communication graph $\mathcal{G}_c^t$.

Assumption 1: The graph sequence $\{\mathcal{G}_c^t\}$ and the weight matrix $\boldsymbol{W}^t$ are such that:

(a) $\boldsymbol{W}^t$ is row-stochastic with $[\boldsymbol{W}^t]_{ij} > 0$ if $j \in \mathcal{V}_i^t$, otherwise $[\boldsymbol{W}^t]_{ij} = 0$.
(b) $\boldsymbol{W}^t$ has strictly positive diagonal entries, $[\boldsymbol{W}^t]_{ii} > 0$.
(c) If $[\boldsymbol{W}^t]_{ij} > 0$ then $[\boldsymbol{W}^t]_{ij} > \eta$ for some positive constant η.
(d) $\{\mathcal{G}_c^t\}$ is B-strongly connected, i.e., there is an integer $B \geq 1$ such that the graph $\left\{\mathcal{M}, \cup_{z=tB}^{(t+1)B-1} \mathcal{E}_c^z\right\}$ is strongly connected for all $t \geq 0$.

B. Bayesian Parameter Estimation

First, let us begin by highlighting the following key remark when formulating the Bayesian estimation learning rule from (3) for all $i \in \mathcal{M}$.

[3] Since agents have access only to their own distributions, the proposed formulations allow agents to sample from P_j instead of P_X [13].

[4] A strongly connected graph is a directed graph in which paths exist in both directions, connecting any two distinct vertices within the graph.

[5] In this equation, dependency of $\boldsymbol{W}$ on t is suppressed.

Remark 1: Minimizing the KL divergence between the empirical global labeling distribution $f(\,\cdot\mid\boldsymbol{x})$, defined by the training set, and the likelihood distribution $l_j(\,\cdot\mid\boldsymbol{x},\boldsymbol{\theta}_k)$ with respect to $\boldsymbol{\theta}_k$ is equivalent to maximizing the conditional mean of the log-likelihood. More specifically,

$$\underset{k\in\{1,\ldots,K\}}{\operatorname{argmin}}\ D_{\mathrm{KL}}\left(f(\cdot|\boldsymbol{x})\middle\|l_j(\cdot|\boldsymbol{x},\boldsymbol{\theta}_k)\right) \tag{4}$$

$$=\underset{k\in\{1,\ldots,K\}}{\operatorname{argmax}}\ \mathbb{E}_{y\sim f(\cdot|\boldsymbol{x})}\log l_j(y|\boldsymbol{x},\boldsymbol{\theta}_k), \tag{5}$$

where (4) follows from the definition of D_{KL} and (5) follows from the fact that $\mathbb{E}_{y\sim f(\cdot|\boldsymbol{x})}\log f(y|\boldsymbol{x})$ does not depend on $\boldsymbol{\theta}$. By using Remark 1, problem (3) is simply given by

$$\underset{k\in\{1,\ldots,K\}}{\text{maximize}}\quad\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\,\mathbb{E}_{\boldsymbol{x}\sim P_j}\,\mathbb{E}_{y\sim f(\cdot|\boldsymbol{x})}\ \log l_j(y|\boldsymbol{x},\boldsymbol{\theta}_k). \tag{6}$$

Mathematically, the Maximum Likelihood Estimation (MLE) problem (6) can equivalently be cast as an optimization problem over the posterior distribution vector $\boldsymbol{\mu}_i$ [20]. In particular, we have the following problem:

$$\underset{\boldsymbol{\mu}_i\in\Delta\Theta}{\text{maximize}}\sum_{k=1}^{K}[\boldsymbol{\mu_i}]_k\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\,\mathbb{E}_{\boldsymbol{x}\sim P_j}\,\mathbb{E}_{y\sim f(\cdot|\boldsymbol{x})}\ \log l_j(y|\boldsymbol{x},\boldsymbol{\theta}_k). \tag{7}$$

The equivalence of between (6) and (7) follows from that $\Delta\Theta$ is a probability simplex. In particular, the solution $\boldsymbol{\mu}_i^\star$ of problem (7) conforms to $[\boldsymbol{\mu}_i^*]_k=1$ at $k=\tilde{k}^*$ and zero otherwise for all i, where $\tilde{k}^*$ is the solution of problem (6). Moreover, under the global identifiability setting [*cf.* Definition 1], the solution of problem (7) coincides with the solution of problem (1), i.e., $\tilde{k}^\star=k^\star$.

It is commonplace to attack problem (7) by replacing expectation $\mathbb{E}_{\boldsymbol{x}\sim P_j}\,\mathbb{E}_{y\sim f(y|\boldsymbol{x})}(\cdot)$ with corresponding empirical averages. Specifically, in situations where we have a large sample, it is typical to choose any $D_{0j}\le D_j$ as batch size, and to construct a stochastic gradient by taking a subsample of indices $m_1,\ldots,m_{D_{0j}}$ uniformly at random, either with or without replacement, from $\{1,\ldots,D_j\}$ at time $t\ge 0$. Let us denote by $\boldsymbol{q}_j^t\in\mathbb{R}^K$ the resulting empirical average vector at time t. Consequently, the kth component of $\boldsymbol{q}_j^t\in\mathbb{R}^K$ is simply given by

$$[q_j^t]_k=\frac{1}{D_{0j}}\sum_{m=1}^{D_{0j}}\log l_j\left(y^{(m)}|\boldsymbol{x}^{(m)},\boldsymbol{\theta}_k\right).$$

Hence, the resulting stochastic gradient method, combined with appropriate regularization $\psi_{\boldsymbol{w}_i,\boldsymbol{\mu}^t}$, gives rise to the following proximal point algorithm for updating agent i's posterior distribution:

$$\boldsymbol{\mu}_i^t:=\underset{\boldsymbol{\nu}_i\in\Delta\Theta}{\operatorname{argmin}}\left[-\langle\boldsymbol{\nu}_i,\boldsymbol{g}_i^t\rangle+(1/2\alpha)\psi_{w_i,\boldsymbol{\mu}^{t-1}}(\boldsymbol{\nu}_i)\right],$$

where $\boldsymbol{g}_i^t$ is a vector in $\mathbb{R}^K$, whose kth component is given by $[g_i^t]_k=\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}[q_j^t]_k$ and

$$\psi_{\boldsymbol{w}_i,\boldsymbol{\mu}^{t-1}}(\boldsymbol{\nu})\triangleq\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}D_{\mathrm{KL}}\left(\boldsymbol{\nu}\|\boldsymbol{\mu}_j^{t-1}\right). \tag{8}$$

Note that $\boldsymbol{w}_i=[[\boldsymbol{W}]_{i1},\ldots,[\boldsymbol{W}]_{iM}]^{\mathrm{T}}$, $\boldsymbol{\mu}^t=[\boldsymbol{\mu}_1^t,\ldots,\boldsymbol{\mu}_M^t]^{\mathrm{T}}$, and t is used to denote the iteration index of the proximal point algorithm. Note that the regularization term defined in (8) encourages $\boldsymbol{\mu}_i^t$ of agent i not to be far from the posteriors of its neighbors $j\in\mathcal{N}_i$ whose significance is ranked with $[\boldsymbol{W}]_{ij}$. Thus, $\boldsymbol{\mu}_i^t$ is formally the solution of the following problem:

$$\begin{aligned}&\underset{\boldsymbol{\nu_i}}{\text{minimize}}-\langle\boldsymbol{\nu}_i,\boldsymbol{g}_i^t\rangle+(1/2\alpha)\psi_{\boldsymbol{w}_i,\boldsymbol{\mu}^{t-1}}(\boldsymbol{\nu}_i)\\&\text{subject to }\boldsymbol{\nu}_i\succeq 0,\quad\boldsymbol{\nu}_i^{\mathrm{T}}1=1.\end{aligned} \tag{9}$$

An iterative algorithm, in which each agent needs to communicate to its neighbors only, can be devised to yield the solution of problem (9). The iterates are established in the following proposition.

Proposition 1: The solution for problem (9) is given by:

$$[\mu_i^t]_k=\frac{\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\tilde{\nu}_j^t]_k\right)}{\sum_{q=1}^{K}\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\tilde{\nu}_j^t]_q\right)},\quad\forall i,j\in\mathcal{M}, \tag{10}$$

where

$$[\tilde{\nu}_j^t]_k=\frac{[q_j^t]_k[\mu_j^{t-1}]_k}{\sum_{l=1}^{K}[q_j^t]_l[\mu_j^{t-1}]_l},\quad\forall j\in\mathcal{M}. \tag{11}$$

Proof: See Appendix A. □

The sequence $\{[\mu_i^t]_k\}_{t\in\mathbb{N}_0}$ admits a linear convergence as established in [13], which we state in Theorem 1 for cohesion. We start by outlining a couple of assumptions.

Assumption 2: For all agents $i\in M$, let $\tilde{\mathcal{I}}_i:=\operatorname{argmin}_{k\in\{1,\cdots,K\}}\mathbb{E}_{\boldsymbol{x}\sim P_i}\left[D_{\mathrm{KL}}(f(\cdot|\boldsymbol{x})\|l_i(\cdot|\boldsymbol{x},\boldsymbol{\theta}_k))\right]$ and $\mathcal{I}^*:=\cap_{i=1}^{M}\tilde{\mathcal{I}}_i$. There exists a parameter $k^*\in\{1,\cdots,K\}$ that is globally identifiable; i.e. $\mathcal{I}^*\neq\varnothing$.

Assumption 3: For each agent $i\in\mathcal{M}$, assume (i) agent i's prior distribution at $t=0$ is $[\mu_i^0]_k>0,\ \forall\boldsymbol{\theta}_k\in\Theta$. (ii) There exists an $\alpha>0$, $L>0$ such that $\alpha<l_i(y|x,\boldsymbol{\theta}_k)<L$, for all $y\in\mathcal{Y},\boldsymbol{\theta}_k\in\Theta$ and $x\in\mathcal{X}$.

Assumption 2 guarantees that the collective observation of agents throughout the network provides statistically sufficient information to learn the global model. Assumption 3 safeguards against the occurrence of degenerate scenarios where a Bayesian prior or a likelihood of zero can hinder the learning process. Moreover, a bounded likelihood model is essential for convergence [10], [13], [20].

Theorem 1: Let Assumptions 1, 2, and 3 hold. Using the decentralized learning algorithm in equations (10) and (11), for any given confidence parameter $\gamma\in(0,1)$ and any arbitrary small $\epsilon>0$, we have

$$\max_{i\in\mathcal{M}}\max_{k\notin\mathcal{I}^*}[\tilde{\nu}_i^t]_k\le\exp -t\left(\Pi-\epsilon\right),$$

where the number of samples satisfies $t\ge\dfrac{8c\log(MK/\gamma)}{\epsilon^2(1-\lambda_{max}(W))}$. The rate of convergence Π of the posterior distribution is given by

$$\Pi=\min_{\boldsymbol{\theta}^*\in\{\boldsymbol{\theta}_k|k\in\mathcal{I}^*\},\boldsymbol{\theta}\notin\{\boldsymbol{\theta}_k|k\in\mathcal{I}^*\}}\sum_{j=1}^{M}v_jI_j(\boldsymbol{\theta}^*,\boldsymbol{\theta}),$$

where $I_j(\boldsymbol{\theta}^*, \boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}_j \sim \mathcal{X}_j}[D_{\text{KL}}(l_j(\cdot|\boldsymbol{\theta}^*, \boldsymbol{x}_j) \| l_j(\cdot|\boldsymbol{\theta}, \boldsymbol{x}_j))]$, v_j is the eigenvector centrality associated with agent $j \in \mathcal{M}$, and $\boldsymbol{v} = [v_1, \ldots, v_M]^{\text{T}}$ is the unique stationary distribution[6] of W with strictly positive components. Furthermore $\lambda_{max}(W) = \max_{1 \le i \le M-1} \lambda_i(W)$, where $\lambda_i(W)$ denotes the ith eigenvalue of W counted with algebraic multiplicity and $\lambda_0(W) = 1$, and $c = |\log(L/\alpha)|$[7].

The rate of convergence Π depends on two factors. The first is the structure of the weighted network, which is quantified by the eigenvector centrality $\boldsymbol{v}$ of the agents. The second one is the ability of agents to differentiate between parameters, a feature captured by the KL-divergences $[I_1, \ldots, I_M]$. These KL-divergences are calculated between the likelihood functions conditioned on the input, thereby measuring the extent to which parameter $\boldsymbol{\theta}^* \in \{\boldsymbol{\theta}_k | k \in \mathcal{I}^*\}$ can be distinguished from parameter $\boldsymbol{\theta} \notin \{\boldsymbol{\theta}_k | k \in \mathcal{I}^*\}$. As a result, each agent may influence the convergence rate in two distinct manners. First, the higher the eigenvector centrality of an agent, the better the position of the corresponding agent within the network to influence the posterior distributions of other agents, consequently having a greater impact on the rate of convergence. Secondly, agents with high KL-divergence, signifying highly informative local observations capable of distinguishing between parameters, also contribute to boosting the convergence rate [13], [14]. This suggests that optimizing Π with respect to the eigenvector centrality $\boldsymbol{v}$ of the agents would enhance the speed of learning. However, finding the optimal eigenvector centrality for each agent j is challenging due to its dependence on the unknown true parameter θ^* in $I_j(\theta^*, \theta)$. Our work overcomes this challenge by deriving a rate of convergence for the agents' posterior distributions $\boldsymbol{\nu}_i$ for all $i \in \mathcal{M}$ that is independent of the true parameter θ^*, thereby rendering it suitable for optimization. In the subsequent sections, we present our main assertion and the proposed optimization framework.

It's worth noting that the authors in [23], [24] demonstrates that the upper bound of the error exponent for posterior distributions can be achieved by the uniform Perron eigenvector, as illustrated in Corollary 2 of [23]. Since the uniform Perron eigenvector corresponds to a doubly-stochastic weight matrix, Corollary 2 of [23] concludes that any doubly-stochastic weight matrix will be optimal. Remarkably, this finding establishes the independence of the optimal Perron eigenvector from the choice of the global truth, thereby addressing the aforementioned challenge. However, it is important to emphasize that this observation does not directly translate to our context, as our focus lies in non-adaptive social learning, whereas the study in [23], [24] pertains to adaptive social learning. In more detail, in the adaptive settings, the steady-state error probability is non-zero and dependent on the eigenvector centrality. Therefore, the goal of [23], [24] is not only to consider the transient behaviors (i.e., convergence rate) but also the steady-state error probability when finding the optimal eigenvector centrality. Their conclusion is in contrast to the analogous result in non-adaptive social learning [14], where a positive relation between the informativeness of agents and the centrality of agents is highlighted for improving the learning performance [24].

C. A Modified Learning Rate Analysis and Optimization

Let us start by deriving a linear convergence rate for the error between the optimal posterior distribution and sequence $\{[\tilde{\nu}_i^t]_k\}_{t \in \mathbb{N}_0}$, which can be practically optimized. More specifically, we show that, at each agent i, the error convergence to zero is at least as fast as a geometric series with a rate dependent on the *weighted sum of discrimination information* between posterior distributions of agent i and its neighboring agents. Hereby, we introduce our main theorem.

Theorem 2: Let Assumptions 1, 2, and 3 hold. Also, let $\zeta \in (0, 1)$ be a confidence level and $\epsilon > 0$ be any arbitrary small number. Then, the decentralized learning algorithm in equations (10) and (11) has the following property: there exists an integer $N(\zeta)$ such that, with probability $1 - \zeta$, for all $t \ge N(\zeta)$ there holds that for any $k \notin \mathcal{I}^*$,

$$\max_{i \in \mathcal{M}} [\tilde{\nu}_i]_k^t < e^{-t\left(\Lambda(\boldsymbol{W}^t) - \epsilon\right)},$$

where $N(\zeta) \triangleq 2A\epsilon^{-2} \log\left(M(K-1)/\zeta\right)$ and

$$\Lambda(\boldsymbol{W}^t) = \sum_{\substack{j=1 \\ j \ne i}}^{M} [[\boldsymbol{W}^t]_{ij}] D_{\text{KL}}(\tilde{\boldsymbol{\nu}}_i^t \| \tilde{\boldsymbol{\nu}}_j^t). \tag{12}$$

Proof: See Appendix B. □

The rate of convergence given in (12), depends on both the KL-divergence between agents' posterior distributions and the structure of the weighted network defined by $[W^t]_{ij}$ for all agents $i \in \mathcal{M}$ and $j \in \mathcal{M}$ at time t. As a result, every agent can influence the convergence rate by assigning higher edge weights to neighbors that exhibit the most significant divergence in their posterior distributions compared to the agent's own distribution. The divergence reflects the degree of disparity in information content among agents. This information content is sourced either from the agents' local datasets or from their prior interactions within the network. Intuitively, this can be perceived as agents achieving more efficient learning when they combine diverse information, rather than redundantly processing data that does not contribute substantially to the overall learning.

Hence, our derived rate (12) suggests a practical means to estimate information diversity throughout the network. This is achieved by measuring information heterogeneity through KL-divergences among agents' posterior distributions, which agents can access during training. With this insight, we now optimize the network structure by using the derived convergence rate (12) as an objective function. Thus, in addition to the standard posterior distribution update (10), $\boldsymbol{W}_{ij}$s are also optimized, which we refer to as graph optimization.

[6] The unique stationary distribution of an irreducible and aperiodic matrix in the context of Markov chains is defined as a probability distribution over states that remains unchanged despite transitions, indicating the chain's long-term equilibrium [26].

[7] The ratio between the upper and lower bounds of agent i's likelihood models impacts the minimum number of samples required for convergence. The smaller the c, the smaller number of samples needed for convergence, and vice versa.

D. Decentralized Graph Optimization

In this subsection, we describe our proposed problem formulation and a decentralized algorithm for optimizing the graph weights $[\boldsymbol{W}^t]_{ij}$ for all $i, j \in \mathcal{M}$, given the posterior distributions $\tilde{\boldsymbol{\nu}}_i^t$ for all agents $i \in \mathcal{M}$. In particular, the graph optimization problem is given by

$$\underset{\{[\boldsymbol{W}^t]_{ij}\}}{\text{maximize}} \quad \sum_{i=1}^{M}\sum_{j=1}^{M}[\boldsymbol{W}^t]_{ij} D_{\text{KL}}(\tilde{\boldsymbol{\nu}}_i^t||\tilde{\boldsymbol{\nu}}_j^t) \tag{13a}$$

$$\text{subject to} \quad \sum_{j=1}^{M}[\boldsymbol{W}^t]_{ij} = 1, [\boldsymbol{W}^t]_{ii} \geq \delta, \tag{13b}$$

$$[\boldsymbol{W}^t]_{ij} \geq 0 \qquad \forall j \in \mathcal{S}_i,\, j \neq i, \tag{13c}$$

$$[\boldsymbol{W}^t]_{ij} = 0 \qquad \forall j \notin \mathcal{S}_i,\, j \neq i, \tag{13d}$$

where $\delta \in (0, 1)$ is a strictly positive constant and recall that $\mathcal{S}_i$ the set of neighbors of agent i in the physical graph $\mathcal{G}_p$. The above constraints imply that $\boldsymbol{W}^t$ is a row-stochastic matrix, agents' self-reliances $[\boldsymbol{W}^t]_{ii}$ are strictly positive, and $[\boldsymbol{W}^t]_{ij}$ is set to zero for all $j \notin \mathcal{S}_i$ for all $i \in \mathcal{M}$. Problem (13) is a standard linear programming (LP) problem, and the optimal solution is given by

$$[\boldsymbol{W}^t]_{ij} = \begin{cases} \delta, & j = i, \\ 1-\delta, & j = \text{argmax}_{\hat{j}\in\mathcal{S}_i} D_{\text{KL}}(\tilde{\boldsymbol{\nu}}_i^t||\tilde{\boldsymbol{\nu}}_{\hat{j}}^t) \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

The solution admits a sparse solution giving rise to a sparse graph where each agent selects exactly one neighbor that is the most valuable to its learning at time t. The chosen neighbor possesses the most distinct knowledge, potentially sourced from its local datasets and past interactions, compared to the agent's own knowledge. The choice of each agent agrees with intuition, in the following sense: agents learn more efficiently when they combine diverse pieces of information, as opposed to having redundant information that does not contribute to the holistic view. Moreover, it is worth highlighting that the sparsity solution (14) enables a light communication protocol.

In MADL, preserving the connectivity of the communication graph is a critical factor for the model to achieve consensus, as indicated in [27]. Traditionally, graph connectivity can be ensured by constraining the second largest eigenvalue, also known as the algebraic connectivity, of the optimized graph to be less than one, as discussed in [27], [28]. Specifically tailored to the proposed graph optimization and the consequent construction of $\boldsymbol{W}^t$ [*cf.* (14)], the following result can be established.

Lemma 1: Let $\boldsymbol{W}^t$ be the weight matrix constructed from (14) at time t and let $[\boldsymbol{W}]_i^t$ denote the i-th row of $\boldsymbol{W}^t$. Also, let $\hat{j} = \text{argmax}_{j\in\mathcal{S}_i} D_{\text{KL}}(\tilde{\boldsymbol{\nu}}_i^t||\tilde{\boldsymbol{\nu}}_j^t)$. For any $\delta \in (0, 1)$ and $t \geq 0$, the weight matrix $\boldsymbol{W}^t$ satisfies Assumption 1.

Proof: See Appendix C. □

Lemma 1 claims that the proposed weight matrix $\boldsymbol{W}^t$ guarantees existence of a B-strongly connected graph. This together with Lemma 3 [*cf.* page 13] ensures that B is always upper-bounded. More specifically, we have

$$B \leq \frac{1}{M}\frac{\log\rho}{\log\eta}, \tag{15}$$

where $\rho = \inf_{t\geq 0}\left(\min_{1\leq i\leq M}[\mathbb{1}_M'[[\boldsymbol{W}]_i]_{t:0}]\right)$ and $0 < \eta < \min(\delta, 1-\delta)$. The proof is straightforward.

Algorithm 1 BayGO Algorithm

```
Inputs: $\boldsymbol{\nu}_i^0 \in \Delta\Theta$ with $[\nu_i^0]_k > 0\ \forall k \in \{1, \cdots, K\}\ \forall i \in \mathcal{M}$, strongly-connected physical graph $\mathcal{G}_p$, and $0 < \delta < 1$.
for $t = 1$ to $T$ do
    for all $i \in \mathcal{M}$, in parallel do
        Draw a batch of samples $(\boldsymbol{X}_i^t, \boldsymbol{y}_i^t) \in \mathcal{D}_i$.
        Local Posterior Distribution Update: Form $\tilde{\boldsymbol{\nu}}_j^t$ using the following rule. For each $k \in \{1, \cdots, K\}$,
            $[\tilde{\nu}_j^t]_k = \frac{[q_j]_k[\mu_j^{t-1}]_k}{\sum_{l=1}^{K}[q_j]_l[\mu_j^{t-1}]_l}$    (16)
        Communication Step: Send $\tilde{\boldsymbol{\nu}}_i^t$ to neighbor $j$ for which $i \in \mathcal{S}_j$, and receive $\tilde{\boldsymbol{\nu}}_j^t$ from neighbor $j \in \mathcal{S}_i$.
        Graph Optimization: Update agent $i$'s edge weights; i.e., $[\boldsymbol{W}^t]_{ij}\ \forall j \in \mathcal{S}_i$ as follows:
            (a) Calculate KL-divergence between $\tilde{\boldsymbol{\nu}}_{i,t}$ and $\tilde{\boldsymbol{\nu}}_{j,t}$, $\forall j \in \mathcal{S}_i$.
            (b) Select the neighbor with the highest KL-divergence (denoted as agent $s$).
            (c) Update $[\boldsymbol{W}^t]_{is} = 1 - \delta$, and all $[\boldsymbol{W}^t]_{ij} = 0, \forall j \in \mathcal{M} \setminus \{i, s\}$.
        Consensus Step: Update local posterior distribution by averaging the log posterior distributions of agent $i$ and its neighbors as follows:
            $[\mu_i^t]_k = \frac{\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}^t]_{ij}\log[\tilde{\nu}_j^t]_k\right)}{\sum_{q=1}^{K}\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}^t]_{ij}\log[\tilde{\nu}_j^t]_q\right)}$.    (17)
    end for
end for
```

We are now ready to outline our proposed BayGO algorithm, *cf.* Algorithm 1. Fig. 1 provides an illustrative example that outlines the workflow of our algorithm. This example visually demonstrates several key aspects, including (i) the collaborative interactions among agents and their neighbors, (ii) the impact of local datasets and previous network interactions on the informativeness of each agent, (iii) the propagation of updates throughout the network, and (iv) the formation of a B-strongly connected graph for the agents' connections over time.

In particular, we examine a network consisting of 7 agents, each exhibiting varying levels of informativeness in their datasets, represented by different shades of color. The dashed grey arrows delineate the underlying strongly-connected physical graph $\mathcal{G}_p$. Initially, the agents possess no prior knowledge regarding the distribution of data across the network. Following the first update of their local posterior distributions, the agents calculate the KL-divergences between their respective updates. Each agent then establishes a connection with the agent displaying the highest divergence in terms of posterior distribution.

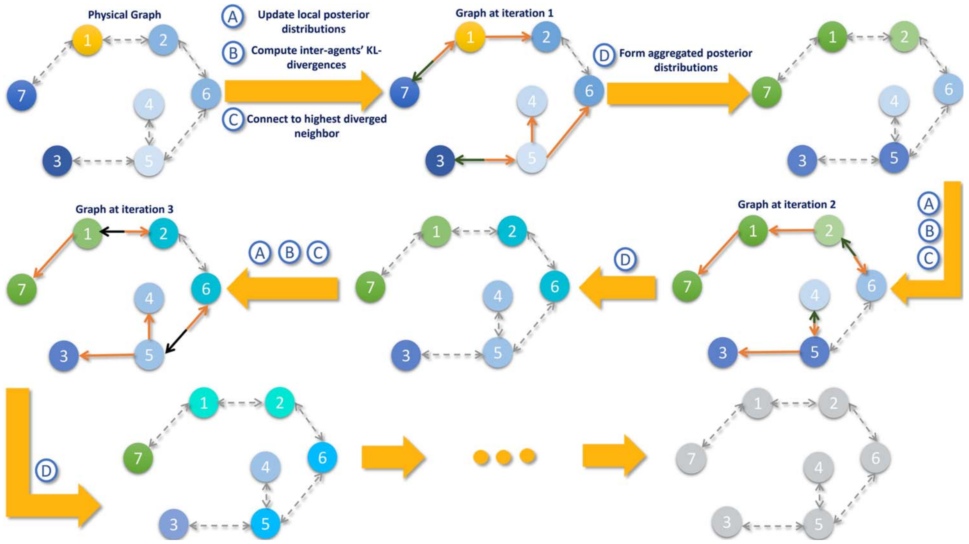
Physical Graph
A Update local posterior distributions
B Compute inter-agents' KL-divergences
C Connect to highest diverged neighbor
Graph at iteration 1
D Form aggregated posterior distributions
Graph at iteration 2
Graph at iteration 3
A B C
D
1
2
3
4
5
6
7

Fig. 1. An illustrative example describing the flow of BayGo.

From each agent's viewpoint, the most divergent agent is the one with a notably different shade. For instance, if we examine agent 5, it initially identifies agent 3 as its most divergent neighbor. Consequently, during iteration 1, agent 5 collaborates with agent 3.

Note that collaboration among agents is symbolized by blending their colors and adopting the resulting color as their new shade. For the sake of simplicity, we assume that each agent's color contributes equally, accounting for 50% of the resulting shade. Examining the entire network, it becomes evident that agent 1 stands out as the most divergent from all the others. However, agent 5 cannot directly establish a connection with agent 1. Nonetheless, agent 1's knowledge is transmitted to agent 5 through a relay of interactions involving agents 2 and 6. Put differently, in the initial iteration, agent 1 collaborates with agent 2, and in the subsequent iteration, it engages in collaboration with agent 6. This progression establishes agent 6 as the most diverged resource for agent 5's learning at that point. Consequently, in the third iteration, agent 5 collaborates with agent 6. This sequence of interactions continues among agents until they ultimately converge to a shared model, represented by a unified color shade. At this stage, neighbor selection simplifies to a random choice of one agent. Hence, one can observe that the combination of graphs after B steps is strongly connected.

Note that the posterior distributions $\boldsymbol{\mu}_i$s of the preceding discussions and in Algorithm 1 are discrete and finitely supported. Nevertheless, the derivations can be linked with continuous posterior distributions with infinite supports as pointed out in the following remark.

Remark 2: Suppose that $\boldsymbol{\theta}^* \in \Phi$, where parameter set Φ is a compact subset of $\mathbb{R}^d$. Furthermore, assume that there exists a set $\Theta \subset \Phi$ of cardinality K which is an r-covering of Φ, denoted as $\mathcal{B}_r(\boldsymbol{\theta})$. These assumptions indicate that the model that generates labels across the network can be parameterized by a continuous parameter $\boldsymbol{\theta}$ which belongs to a compact set $\Phi \subset \mathbb{R}^d$. Moreover, these assumptions imply that there exists a quantization of Φ with quantization points in Θ such that Θ is an r-covering of Φ. In [15], they have shown that with high probability we have $\boldsymbol{\theta}^* \in \mathcal{B}_r(\hat{\boldsymbol{\theta}}_i^t)$ where $\hat{\boldsymbol{\theta}}_i^t$ is the most probable parameter at time t and at agent i for all $i \in \mathcal{M}$.

IV. Numerical Evaluation

In order to assess the effectiveness of the BayGO framework, in what follows, we consider two distinct tasks, namely, linear regression on a real body-fat dataset [29], and image classification on the MNIST dataset [30].

A. *Decentralized Bayesian Linear Regression*

Simulation Setup: We consider a multi-agent setting with $M = 12$ agents, who are to solve a linear regression problem with $\Theta = \mathbb{R}^2$, $\mathcal{X} \in \mathbb{R}$, and $\mathcal{Y} \in \mathbb{R}$. More specifically, $\mathcal{X} = [70, 120]$ represents the set of abdomen features and $\mathcal{Y}$ represents the body fat percentage where $y = \boldsymbol{\theta}^{*T}\boldsymbol{x} + n$ for all $y \in \mathcal{Y}$ and n denotes the additive Gaussian noise $n \sim \mathcal{N}(0, \vartheta^2)$. The first component of the parameter $\boldsymbol{\theta} \in \Theta$ is the intercept and the second component is the gradient of the model. A Gaussian prior is assumed on $\boldsymbol{\theta}$ for each agent i with a mean vector $\boldsymbol{m}_i^0 = 0$ and a diagonal covariance matrix $\boldsymbol{C}_i^0$ having all diagonal elements equal 0.5. In this respect, parameters are continuous (*cf.* Remark 2) and the local posterior distribution $\tilde{\boldsymbol{\nu}}_i^t$ of agent i at time t, which is characterized by $\boldsymbol{m}_i^t$ and $\boldsymbol{C}_i^t$, after a local Bayesian update remain Gaussian. More specifically, (16) at step 4 of Algorithm 1 is replaced by the following:

$$\tilde{\boldsymbol{m}}_i^t = \tilde{\boldsymbol{C}}_i^t \left((\boldsymbol{C}_i^{t-1})^{-1} \boldsymbol{m}_i^{t-1} + (\vartheta^2)^{-1} (\hat{\boldsymbol{X}}^t)^T \boldsymbol{y}_i^t \right),$$

$$(\tilde{\boldsymbol{C}}_i^t)^{-1} = (\boldsymbol{C}_i^{t-1})^{-1} + (\vartheta^2)^{-1} (\hat{\boldsymbol{X}}^t)^T (\hat{\boldsymbol{X}}^t),$$

where $\hat{\boldsymbol{X}}^t$ is the design matrix of size $D_{0i} \times 2$. Moreover, the posterior distribution (17) of agent i at step 7 of Algorithm 1 is replaced by the following:

$$\boldsymbol{m}_i^t = \boldsymbol{C}_i^t \sum_{j=1}^{M} [\boldsymbol{W}^t]_{ij} \left(\tilde{\boldsymbol{C}}_i^t\right)^{-1} \tilde{\boldsymbol{m}}_j^t, \ \left(\boldsymbol{C}_i^t\right)^{-1} = \sum_{j=1}^{M} [\boldsymbol{W}^t]_{ij} \left(\tilde{\boldsymbol{C}}_i^t\right)^{-1}. \tag{18}$$

As a result, the beliefs acquired following the consensus step maintain a Gaussian distribution (i.e., $\boldsymbol{\mu}_i^t \sim \mathcal{N}[\boldsymbol{m}_i^t, \boldsymbol{C}_i^t)]$, thereby indicating that the associated predictive distribution also retains its Gaussian characteristic [13], [15]. We make the observations not to be uniformly distributed among agents by manually splitting body-fat dataset [29] to yield the local data sets $\mathcal{D}_i$s of the users as follows. With $\mathcal{D}_1$, the local data set of agent 1, we associate all the input samples of $\mathcal{X}$ whose values are in $[85, 120]$ and their corresponding labels in $\mathcal{Y}$. Similarly, with local data sets $\mathcal{D}_i$, $i = 2, \ldots, M$ of the rest of the agents, we associate subsets of input samples of $\mathcal{X}$ with values in $[70, 85)$ and the corresponding labels in $\mathcal{Y}$ so that they are statistically similar. The mean-squared error (MSE) between the observed data and the values predicted by the model is computed as a performance metric to evaluate predictions over a test dataset.

Baselines: The following baselines are considered (i) a Centralized setting (Cen), where a single agent has access to all the training data, (ii) a setting with no collaboration (NoCol), where agents learn individually without any collaboration, (iii) a setting with a fully-connected topology (FulCon), where $\mathcal{G}_p$ is a fully-connected graph with uniform eigenvector centrality across all agents (iv) a setting with positive matching star topology (PosM) with agent 1 serving as the head of the star and (v) a setting with negative matching star topology (NegM) with agent 1 serving at the edge. In both PosM and NegM, $\mathcal{G}_p$ is a star graph of order M. The PosM baseline represents an ideal case where the agent 1 is the most influential agent in the network, whereas the NegM baseline represents a scenario where other agents hold the most influence within the network. Lastly, we would like to highlight some technical details regarding the FulCon baseline. Specifically, we employ a fully connected graph where each agent is assigned an equal weight of $1/M$, resulting in a doubly stochastic matrix with uniform eigenvector centrality. This selection of graph weights aligns with the conclusions drawn in [23], [24], where it was identified that in adaptive settings, any doubly-stochastic weight matrix would be optimal. It is noteworthy that non-adaptive settings can be regarded as a special case of the adaptive one. More specifically, the non-adaptive setting is achieved by setting the positive parameter δ of equation (10) of [23] to 0.5[8]. Therefore, FulCon can be viewed as an instantiation of the proposed solution in [23], [24]. Furthermore, if we aim to establish a baseline that emulates the Hastings rule [*cf.* Lemma 12.2 in [22]] under the assumption of all agents experiencing equal noise levels, then FulCon also serves as a valid representation, particularly in the scenario where all agents have an equal number of neighbors.

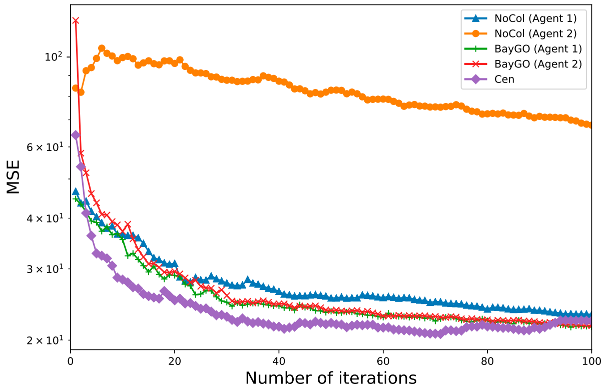

Fig. 2. Comparing BayGO to benchmarks (Cen and NoCol) in accuracy and convergence speed.

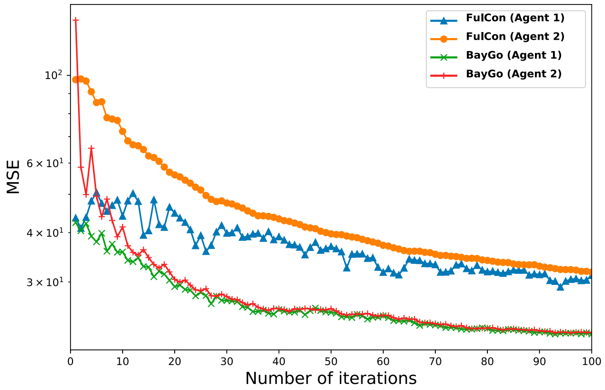

Fig. 3. Comparing inference accuracy and convergence speed between FulCon and BayGO.

Effect of Collaboration Between Agents on Learning Accuracy and Speed: Fig. 2 presents the MSE over training iterations for both agent 1 and agent 2[9], using our framework and two baselines Cen and NoCol. Despite information heterogeneity, both agents, 1 and 2, exhibit comparable performance in terms of MSE and convergence speed when trained under BayGO. The performance is very comparable to the Cen convergence plot. These results suggest that BayGO can learn in a decentralized manner, without each agent having access to all the training data, even in the presence of data set heterogeneity. On the other hand, in the case of NoCol, even though agent 1 performs well, agent 2 performs poorly. This is because agent 1 possesses a relatively more informative dataset whereas agent 2 suffers from local statistical insufficiency. Therefore, BayGO is a reasonable trade-off between making light communication only with neighbors, while keeping the convergence properties comparable to Cen.

Effect of Communication Graph Topology on Learning Speed: Fig. 3 displays the Mean Squared Error (MSE) over training iterations for both agent 1 and agent 2 using BayGO and FulCon. Recall that FulCon serves as a realization of optimal

[8] The positive parameter δ of [23] must not be confused with δ used in (14) of our paper.

[9] Note that agent 2 to 12 are statistically similar in terms of their local data sets $\mathcal{D}_2 - \mathcal{D}_{12}$. To simplify the presentation, we display the performance of agent 2 since other agents 3 to 12 exhibit similar performance.

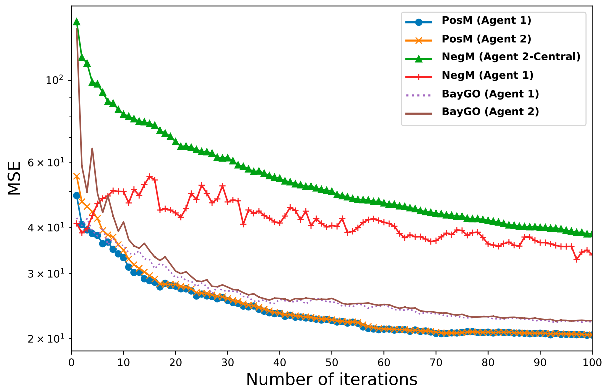

Fig. 4. Impact of learning using star topology with the informative agent at the center or edge compared to BayGo.

weights proposed in [22], [23], [24] under certain conditions [*cf.* Baselines on page IV-A]. It is evident that under the FulCon setting, characterized by a fully-connected topology, the convergence of MSE for each agent is notably slower compared to that achieved by BayGO. Furthermore, the results indicate a lack of consensus between agent 1 and agent 2 even after 100 iterations. These observations can be attributed primarily to dataset heterogeneity. Specifically, the FulCon baseline treats all agents as uniformly informative, thereby disregarding inherent disparities in their local datasets. Conversely, BayGO yields not only a faster convergence in terms of MSE, but also a faster consensus[10] between agents' posterior estimates in a few tens of iterations.

Fig. 4 depicts the MSE versus training iterations by using PosM and NegM star topologies. Under PosM setting, results show that both agents 1 and 2 achieve almost identical performance in terms of MSE convergence. This is expected since agent 1, the most informative agent, resides at the center of the star topology giving one-hop access to the least informative agents (i.e., 2 to 12). At the same time, all the less informative agents are at a one-hop distance to agent 1. As a result, information diffuses effectively among all the agents, yielding fast convergence and consensus of their posteriors. Conversely, in the NegM scenario, both agents 1 and 2 exhibit not only slow convergence but also poor consensus. This is mainly due to that the informative agent 1 is no longer at a one-hop distance to the noninformative agent 2 to yield an effective diffusion of information. Results show that BayGO enables fast convergence of MSE and better consensus compared to NegM. Notably, the performance of BayGO is comparable to the PosM star topology, despite the position of agent 1 in graph $\mathcal{G}_p$. This is a consequence of the optimization framework that dynamically updates the graph $\mathcal{G}_t$, see §III-D.

B. Decentralized Image Classification

Training Neural Network Model: Recall that each iteration of Algorithm 1 comprises a local Bayesian update [i.e., (16)], followed by a consensus step [i.e., (17)]. In the context of most real-world scenarios, calculating the normalizing constants precisely within these update rules is quite challenging from a computational standpoint. In this section, we present an added step along with an adjustment to Algorithm 1 to enhance its applicability in the context of training Neural Network (NN) models. More specifically, when performing the local Bayesian update, we leverage variational inference (VI) techniques outlined in [[31], Section 3]. In this respect, (16) of Algorithm 1 is replaced by the following: for each $\boldsymbol{\theta}\in\Theta$,

$$\tilde{\pi}_j^t(\boldsymbol{\theta}) = \frac{\boldsymbol{q}_j(\boldsymbol{\theta})\mu_j^{t-1}(\boldsymbol{\theta})}{\int_\Theta \boldsymbol{q}_j(\boldsymbol{\psi})\mu_j^{t-1}(\boldsymbol{\psi})} \tag{19}$$

$$\tilde{\boldsymbol{\nu}}_j^t = \underset{\boldsymbol{\pi}\in\boldsymbol{Q}}{\text{argmin}}\, D_{\text{KL}}(\boldsymbol{\pi}\|\tilde{\boldsymbol{\pi}}_j^t), \tag{20}$$

where $\boldsymbol{Q}$ is a permitted family of distributions over Θ. It's important to highlight that the consensus step is executed by applying equations (18).

Simulation Setup: A multi-agent setting with $M=6$ agents was considered, where they perform an image classification task. We have $\Theta=\mathbb{R}^H$, where H is the number of weight parameters in the neural network (NN), a fully connected standard feed-forward network with one hidden layer consisting of 1024 ReLU activation units. Moreover, $\mathcal{X}\in\mathbb{R}^{784}$ represents the 28×28 pixel image and $\mathcal{Y}=\{0,1,\dots,9\}$ represents the labels of the digits in MNIST. That is, the input of the NN is a flattened 28×28 pixel image, while the output layer consists of 10 neurons with each neuron corresponding to one digit. We initialize the NN parameters at each agent by utilizing a Gaussian prior distribution with a zero mean vector and an identity covariance matrix. Variational inference [*cf.* (19) and (20)] is performed on local posteriors to obtain distributions that are constrained to the commonly used Gaussian mean-field approximate distribution family $\mathcal{Q}$ with a probability density function parameterized by $\boldsymbol{m},\boldsymbol{\sigma}$, where $\boldsymbol{m}$ is the mean and $\boldsymbol{\sigma}$ is the covariance matrix [32], [33].

We partitioned the dataset into 6 subsets $\mathcal{D}_1-\mathcal{D}_6$. The local dataset $\mathcal{D}_1$ contains 80% of the observations and the remaining 5 subsets contain the rest of the dataset, equally split. Thus, agent 1 is relatively more informative than other agents who are relatively non-informative. The classification accuracy between the observed data and the labels predicted by the model is computed as a performance metric to evaluate predictions over a test dataset. To this end, after the consensus step of Algorithm 1, for each agent $i\in\mathcal{M}$, first we sample from their aggregated posterior distributions n times to yield $\boldsymbol{\theta}_{i1},\dots,\boldsymbol{\theta}_{iN}$. Then for any image $\boldsymbol{x}_i\in\mathcal{X}_i$, the predicted label $\hat{y}_i$ is given by $\hat{y}_i=\text{argmax}_{y\in\mathcal{Y}}(1/N)\sum_{n=1}^N\frac{\exp\boldsymbol{f}^y_{\boldsymbol{\theta}_{in}}(\boldsymbol{x}_i)}{\sum_{y'\in\mathcal{Y}}\exp\boldsymbol{f}^{y'}_{\boldsymbol{\theta}_{in}}(\boldsymbol{x}_i)}$ where $\boldsymbol{f}^{y'}_{\boldsymbol{\theta}_h}(.)$ denotes the value of node y' of the output layer of the NN. We also compute the confidence in the prediction $\hat{y}$, denoted $c(\hat{y})$, which is given by

$$c(\hat{y})=\sum_{n=1}^N\frac{\exp\boldsymbol{f}^y_{\boldsymbol{\theta}_{in}}(\boldsymbol{x}_i)}{\sum_{y'\in\mathcal{Y}}\exp\boldsymbol{f}^{y'}_{\boldsymbol{\theta}_{in}}(\boldsymbol{x}_i)}. \tag{21}$$

[10] The closer the MSE values agent 1 and 2, the better the consensus of their posterior distributions, which is empirically validated.

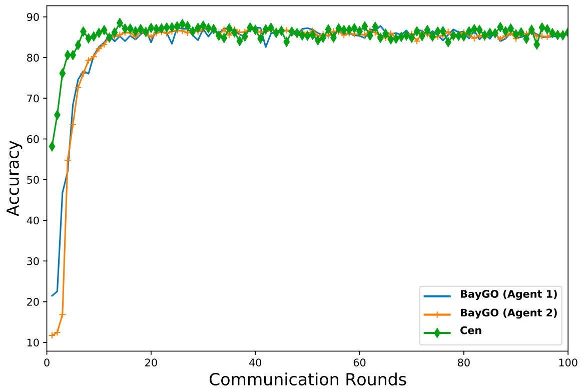

Fig. 5. Comparing BayGO and Cen in terms of accuracy and convergence speed under a classification task.

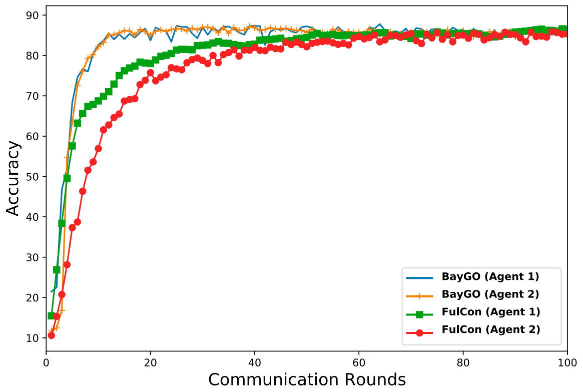

Fig. 6. Comparing inference accuracy and convergence speed in a classification task between FulCon and BayGO.

When running the algorithm, each agent performs (19)-(20) change at step 4 of Algorithm 1 for $m = 5$ times[11].

BayGO Performance Compared to Ideal Benchmark (Cen): Fig. 5 shows the classification accuracy over training iterations for both agent 1 and agent 2[12] using BayGO and Cen. Results show that both agent 1 and agent 2 under BayGO yield almost identical levels of accuracy compared to Cen. This is achieved within a few tens of iterations even though Cen admits a slightly faster convergence speed. Thus, the results suggest that BayGO is comparable to Cen despite the fact that BayGO has distributed non-homogeneous datasets. Moreover, the results show that both agent 1 and agent 2 perform similarly under BayGO. This indicates that the posterior distributions of both agent 1 and 2 are mixed effectively during the iterations. These observations are consequences of the graph optimization framework which dynamically changes the graph $\mathcal{G}_t$.

Effect of Communication Graph Topology on Speed of Learning: Fig. 6 shows the classification accuracy over training iterations for agent 1 and agent 2 by using BayGO and FulCon. The performance of BayGO compared to FulCon is similar to that observed in Fig. 3. In particular, the convergence of the accuracy under FulCon is slower compared to that of BayGO. Moreover, there are substantial mismatches between the classification accuracies of agent 1 and 2 under FulCon roughly until 60 iterations. This is expected because the FulCon baseline assumes uniform informativeness across all agents, thereby overlooking inherent differences in their local datasets. FulCon, once more, embodies the optimal weight distribution proposed in [22], [23], [24] under specific conditions [*cf.* Baselines on page IV-A]. Observing BayGO, both agents demonstrate not only a faster convergence but also similar classification accuracies throughout the communication rounds. For example, BayGo just within 12 communication rounds, BayGO achieved its maximum accuracy, while FulCon takes around 60 communication rounds to reach the same level of accuracy. Another way to view the benifit is that when the number of agents $M = 6$, the FulCon incurs $60 \times M \times (M - 1) = 1800$ message exchanges between agents while BayGO incurs only $12 \times M = 72$ message exchanges to reach the same level of accuracy. This further illustrates the ability of our proposed graph optimization to handle heterogeneous datasets by dynamically blending the local posterior distributions within a set of carefully chosen agents only without unnecessarily flooding communication between all the agents.

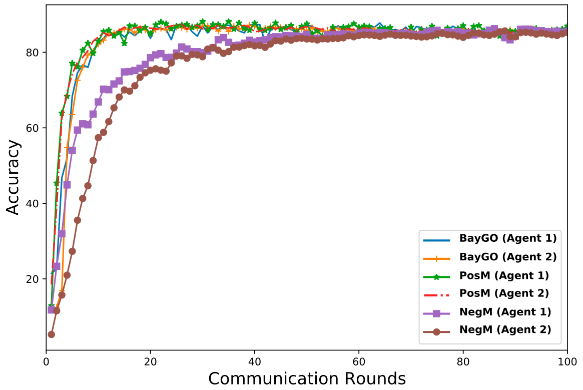

Fig. 7. Impact of learning using star topology with the informative agent at the center or edge compared to BayGO under a classification task.

Fig. 7 plots the test accuracy of agent 1 and 2 versus communication rounds for BayGO, PosM, and NegM. Under PosM, in which agents 1 and 2 lie at the center and the edge of the star graph, respectively, both agents have similar performance and they converge in a very few communication rounds [e.g., 12]. For example, the maximum accuracy level is achieved just within 18 communication rounds. This is intuitively expected since the agent placement seems to be a very desirable setting (in terms of the information spread) in the sense that the relatively more informative agent [i.e., 1] is at the center of the star and the less informative agents [i.e., 2-6] are at the edge nodes. On the other hand, results show that even a slight change in the agent placement can significantly aggravate

[11] Repeating the Bayesian update for a few iterations is beneficial to reduce the communication overhead. Naturally, m is a hyperparameter and its choice can depend on the application [13].

[12] Similar to the previous simulation setting, agent 2 to 6 are statistically similar in terms of their local data sets $\mathcal{D}_2 - \mathcal{D}_6$. Therefore, to simplify the presentation, in addition to the informative agent 1, we display the performance of only one non-informative agent, in particular, the agent 2 since other agents 3 to 6 exhibit similar performance.

TABLE I
PERFORMANCE COMPARISON BETWEEN FULCON, POSM, NEGM, AND BAYGO IN TERMS OF CONFIDENCE IN PREDICTING IMAGES WITH LABEL 2. C.R. DENOTES THE NUMBER OF COMMUNICATION ROUNDS, A1 AND A2 DENOTE AGENT 1 AND AGENT 2, RESPECTIVELY

	FulCon		PosM		NegM		BayGO	
C.R.	A1	A2	A1	A2	A1	A2	A1	A2
1	0.13	0.09	0.34	**0.31**	0.25	0.08	**0.35**	0.09
10	0.34	0.23	0.64	**0.63**	0.44	0.38	**0.66**	0.6
20	0.4	0.36	0.66	0.65	0.52	0.41	**0.74**	**0.73**

the convergence properties, see the plot for NegM in which the agents 1 and 2 both lie at the edge of the star graph. For example, NegM requires $70-80$ communication rounds to reach the maximum accuracy level. Nonetheless, the outcomes indicate that BayGO's performance is quite comparable to that of PosM, even though BayGO does not require any predefined agent placement. Thus the BayGO framework seems to be more robust for dataset heterogeneity.

Effect of Communication Graph Topology on Confidence over Predictions: In the sequel, we compare the confidence of predictions, see (21), of BayGO with the benchmarks FulCon, PosM, and NegM. Table I shows the confidence of both agent 1 and 2 predicting digit '2' (i.e., Label 2) of MNIST test dataset after communication rounds 1, 10, and 20[13]. The results demonstrate that BayGO outperforms the benchmark settings in almost all cases, see boldface numbers for dominant figures. This highlights how the graph optimization within BayGO excels in managing diverse datasets by dynamically merging local posterior distributions. As already pointed out, PosM by design is a setting where the agents are placed in a desired manner to enhance the fast consensus of posterior distributions. Even in this setting, BayGO outperforms the benchmark for agent 1, the most informative agent, while showing comparable performance for agent 2, the less informative agent.

V. CONCLUSION

In this article, we proposed BayGO, a novel fully-decentralized multi-agent Bayesian learning with local averaging and graph optimization framework. BayGO addresses challenges stemming from data heterogeneity and the need for uncertainty quantification, which are common obstacles encountered in MADL systems. It was shown that, BayGO ensures fast convergence over a heterogeneous and sparse communication graph without any assumptions on the prior knowledge of the data distribution across agents. This is done through an iterative process that optimizes agents' posterior distributions alternately with their connections on the graph. Indeed, within BayGO, agents are able to optimize their connections according to the information content of their neighbors; thus, accounting for information heterogeneity throughout the network. Our theoretical analysis demonstrates that when agents optimize their graph connections based on variations in their information, it leads to faster convergence of agents' posterior distributions. Furthermore, this strategy results in the formation of a sparse graph, wherein each agent exclusively communicates with the neighbor offering the greatest learning advantage. Finally, our simulations confirm that the BayGO framework outperforms traditional topologies, such as fully-connected and star configurations, demonstrating the practical benefits of our approach for scalable multi-agent learning.

APPENDIX

A. Proof of Proposition 1

Leaving the positivity constraint implicit in equation (9), the equation can be rewritten as the maximization of the following Lagrangian,

$$L_i(\boldsymbol{\nu},\lambda)=\boldsymbol{\nu}_i^T\boldsymbol{g}_i^t-\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\sum_{k=1}^{K}[\nu_i]_k\log\frac{[\nu_i]_k}{[\mu_j^{t-1}]_k}+\lambda(\boldsymbol{\nu}_i^T\mathbf{1}-1), \tag{22}$$

where $\mathbf{1}$ is vector of all ones. Differentiating (22) we get

$$\frac{\partial}{\partial[\nu_i]_k}L_i(\boldsymbol{\nu},\lambda)=[g_i^t]_k-\log[\nu_i]_k+\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\mu_j^{t-1}]_k+\lambda-1=0,$$

$$\frac{\partial}{\partial\lambda}L_i(\boldsymbol{\nu},\lambda)=\boldsymbol{\nu}_i^T\mathbf{1}-1=0.$$

Solving the above equations yields:

$$[\nu_i]_k=\exp\left([g_i^t]_k+\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\mu_j^{t-1}]_k\right)\exp(\lambda-1), \tag{23}$$

$$\boldsymbol{\nu}_i^T\mathbf{1}=1, \tag{24}$$

and replacing $\boldsymbol{\nu}_i$ in (24) by (23) we have

$$\exp(\lambda-1)=\left[\sum_{q=1}^{K}\exp\left([g_i^t]_q+\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}[\mu_j^{t-1}]_q\right)\right]^{-1}. \tag{25}$$

Hence, from (23) and (25) we have

$$[\mu_i^t]_k=\frac{\exp\left([g_i^t]_k+\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\mu_j^{t-1}]_k\right)}{\sum_{q=1}^{K}\exp\left([g_i^t]_q+\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\mu_j^{t-1}]_q\right)}. \tag{26}$$

Substitute $[g_i^t]_k=\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}[q_j^t]_k$ in (26) we get

$$[\mu_i^t]_k=\frac{\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log\left([q_i^t]_k[\mu_j^{t-1}]_k\right)\right)}{\sum_{q=1}^{K}\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log\left([q_i^t]_k[\mu_j^{t-1}]_q\right)\right)},\quad\forall j\in\mathcal{M}. \tag{27}$$

To simplify the equation, we initially compute each agent's private posterior update by setting their self-reliances to one. This results in the following private posterior update:

$$[\tilde{\nu}_i^t]_k=\frac{[q_i^t]_k[\mu_j^{t-1}]_k}{\sum_{l=1}^{K}[q_i^t]_l[\mu_i^{t-1}]_l},\quad\forall i\in\mathcal{M}.$$

[13] The choice of Digit 2 is arbitrary.

To obtain the agent's aggregated posterior update, we rewrite (27) in terms of agents' private posteriors $[\tilde{\nu}_j^t]_k \forall j$ as follows:

$$[\mu_i^t]_k = \frac{\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\tilde{\nu}_j^t]_k\right)}{\sum_{q=1}^{K}\exp\left(\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}\log[\tilde{\nu}_j^t]_q\right)},$$

and this concludes the proof.

B. Proof of Theorem 2

For the ease of exposition, let $[\tilde{\nu}_i^0]_k = [\tilde{\nu}_j^0]_k = \dfrac{1}{|\Theta|}$ for all $i \in \mathcal{M}, j \in \mathcal{M}, \boldsymbol{\theta}_k \in \Theta$. Let $\boldsymbol{\theta}^* \in \Theta^*$, for each agent $i \in \mathcal{M}$, and for any $\boldsymbol{\theta}_q \notin \Theta^*$. The proof of Theorem 1 makes use of the following quantities: for all $i = 1, \cdots, M$ and $t \geq 0$,

$$\varphi_i^t(\boldsymbol{\theta}_q) \triangleq \log\frac{[\tilde{\nu}_i^t]_*}{[\tilde{\nu}_i^t]_q},\ \ \varphi_{ij}^t(\boldsymbol{\theta}_q) \triangleq \log\frac{[\tilde{\nu}_i^t]_q}{[\tilde{\nu}_j^t]_q},$$
$$[\mathcal{L}_i^t]_q = \log\frac{l_i(y^{(t)}|\boldsymbol{x}^{(t)},\boldsymbol{\theta}^*)}{l_i(y^{(t)}|\boldsymbol{x}^{(t)},\boldsymbol{\theta}_q)},$$

with dependence on $\boldsymbol{\theta}^*$ is suppressed. Also, the proof makes use of the following:

Assumption 4: We assume that $\left|[\mathcal{L}_{ij}^t]_k\right| = \left|\log\left(\dfrac{l_i(y^{(t)}|\boldsymbol{x}^{(t)},\boldsymbol{\theta}_k)}{l_j(y^{(t)}|\boldsymbol{x}^{(t)},\boldsymbol{\theta}_k)}\right)\right| \leq A$ for any $(i,j) \in \mathcal{M}$ and for any $\boldsymbol{\theta}_k \in \Theta$.

Lemma 2: ([10]) Under Assumption 1, for a graph sequence $\{\mathcal{G}_c^t\}$ and each $\tau \geq 0$, there is a stochastic vector ϕ^τ (meaning its entries are nonnegative and sum to one) such that for all i, j and $t \geq \tau$,

$$|[[\boldsymbol{W}]_{ij}]_{t:\tau} - \phi_j^t| \leq 2\lambda^{t-\tau} \qquad \forall t \geq \tau \geq 0,$$

where $\lambda \leq (1-\eta^{MB})^{\frac{1}{B}}$ and $0 < \eta < \min(\delta, 1-\delta)$.

Lemma 3: ([10]). Let the graph sequence $\{\mathcal{G}_c^t\}$ satisfy Assumption 1. Define

$$\rho \triangleq \inf_{t\geq 0}\left(\min_{1\leq i\leq M}[\mathbb{1}'_M[[\boldsymbol{W}]_i]_{t:0}]\right),$$

where $[\boldsymbol{W}]_i$ is the vector $[[\boldsymbol{W}]_{i1}, \cdots, [\boldsymbol{W}]_{iM}]$. Then, $\rho \geq \eta^{MB}$. Furthermore, the sequence ϕ^t from Lemma 2 satisfies $\phi_j^t \geq \dfrac{\rho}{M}$ for all $t \geq 0$, $j = 1, \cdots, M$.

Now we begin with the following recursion:

$$\begin{aligned}\log\frac{[\tilde{\nu}_i^t]_*}{[\tilde{\nu}_i^t]_q} &= \sum_{\tau=1}^{t}\sum_{j=1}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}[\mathcal{L}_j^\tau]_q + \sum_{j=1}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\varphi_j^0(\boldsymbol{\theta}_q)\\ &= \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}[\mathcal{L}_j^\tau]_q + \sum_{\tau=1}^{t}[[\boldsymbol{W}]_{ii}]_{t:\tau}[\mathcal{L}_i^\tau]_q\\ &\quad + \sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\varphi_j^0(\boldsymbol{\theta}_q) + [[\boldsymbol{W}]_{ii}]_{t-1:0}\varphi_i^0(\boldsymbol{\theta}_q).\end{aligned} \tag{28}$$

Let $\sum_{j=1}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau} = C_{t:\tau}^i$ for all $\tau \geq 0$, $\tau \leq t$, $i \in \mathcal{M}$. Then, (28) can be written as:

$$\begin{aligned}\log\frac{[\tilde{\nu}_i^t]_*}{[\tilde{\nu}_i^t]_q} &= \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}\Big[[\mathcal{L}_j^\tau]_q - [\mathcal{L}_i^\tau]_q\Big] + \sum_{\tau=1}^{t}C_{t:\tau}^i[\mathcal{L}_i^\tau]_q\\ &\quad + \sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\Big[\varphi_j^0(\boldsymbol{\theta}_q) - \varphi_i^0(\boldsymbol{\theta}_q)\Big]\\ &\quad + C_{t-1:0}^i\varphi_i^0(\boldsymbol{\theta}_q).\end{aligned} \tag{29}$$

Moreover, we have,

$$\begin{aligned}[\mathcal{L}_j^\tau]_q - [\mathcal{L}_i^\tau]_q = \log&\left(\frac{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}\right)\\ &+ \log\left(\frac{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}\right).\end{aligned} \tag{30}$$

We denote the first, second, third, and fourth terms on the right-hand side of equation (29) as $T_1, T_2, T_3,$ and $T4$ respectively. Using (30), we can rewrite T_1 as follows:

$$\begin{aligned}T_1 = &\sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}\log\left(\frac{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}\right)\\ &+ \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}\log\left(\frac{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}\right).\end{aligned}$$

Adding and subtracting $\sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}\phi_j^\tau\log\left(\dfrac{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}\right)$ to and from T_1 yields

$$\begin{aligned}T_1 \geq &\sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}\phi_j^\tau\log\left(\frac{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}\right)\\ &- \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}\left|[[\boldsymbol{W}]_{ij}]_{t:\tau} - \phi_j^\tau\right|\left|\log\left(\frac{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}^*)}\right)\right|\\ &+ \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}\log\left(\frac{l_i(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}{l_j(y^{(\tau)}|\boldsymbol{x}^{(\tau)},\boldsymbol{\theta}_q)}\right)\\ &\overset{(a)}{\geq} \sum_{\tau=1}^{t}\sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}[\mathcal{L}_{ij}^\tau]_q - \frac{A\rho(M-1)t}{M} + 2At\lambda^{t-\tau},\end{aligned} \tag{31}$$

where (a) follows from Lemma 2 and the boundedness assumption of log-likelihood ratios. On the other hand, taking the term T_3 in equation (29) and simplifying it, we get

$$T_3 = \sum_{\substack{j=1\\ j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\Big[\varphi_{ji}^0(\boldsymbol{\theta}_*) + \varphi_{ij}^0(\boldsymbol{\theta}_q)\Big]. \tag{32}$$

Substituting (31) and (32) back into equation (29), and since $[\tilde{\nu}_i^t]_* \leq 1$, equation (29) can be written as

$$\begin{aligned}-\log[\tilde{\nu}_i]_q^t \geq & \sum_{\tau=1}^{t}\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}[\mathcal{L}_{ij}^{\tau}]_q - \frac{A\rho(M-1)t}{M}\\ & + 2At\lambda^{t-\tau} + \sum_{\tau=1}^{t} C_{t:\tau}^{i}[\mathcal{L}_i^{\tau}]_q\\ & + \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\varphi_{ji}^{0}(\boldsymbol{\theta}_*)\\ & + \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\varphi_{ij}^{0}(\boldsymbol{\theta}_q) + C_{t-1:0}^{i}\varphi_i^{0}(\boldsymbol{\theta}_q).\end{aligned}$$

The fifth and seventh terms in the above equation goes to zero as all agents have equal uniform prior distributions. We also have the following:

$$\begin{aligned}&\sum_{\tau=1}^{t}\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t:\tau}[\mathcal{L}_{ij}^{\tau}]_q + \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t-1:0}\varphi_{ij}^{0}(\boldsymbol{\theta}_q)\\ &= \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) + \frac{t\rho A}{M}.\end{aligned}$$

The last term of the right-hand side of the above equation was obtained from normalization constants of $[\tilde{\nu}_i^t]_q$ and $[\tilde{\nu}_j^t]_q$ and using Assumption 4, Lemma 2 and Lemma 3. Thus, we have:

$$\begin{aligned}-\log[\tilde{\nu}_i]_q^t \geq & \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) + \frac{t\rho A}{M} - \frac{A\rho(M-1)t}{M}\\ & + 2At\lambda^{t-\tau} + \sum_{\tau=1}^{t} C_{t:\tau}^{i}[\mathcal{L}_i^{\tau}]_q.\end{aligned} \tag{33}$$

Since $\lambda \in (0,1)$, the fourth term of the right-hand side in the above equation converges to zero. Also, by Lemma (2) and Lemma 3, $\lim_{t\to\infty} C_{t:\tau}^{i} = \mathbb{1}_M \phi_j^t \geq \mathbb{1}_M \frac{\rho}{M} = \rho$. with assumption 4, the last term of the right hand side in (33) goes to $-A\rho t$. Thus, (33) becomes:

$$-\frac{1}{t}\log[\tilde{\nu}_i]_q^t \geq \frac{1}{t}\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) - \left(\frac{2A\rho t(M-1)}{M}\right).$$

Now let $\epsilon = \left(\frac{4A\rho t(M-1)}{M}\right)$. Then we have

$$-\frac{1}{t}\log[\tilde{\nu}_i]_q^t \geq \frac{1}{t}\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) - \frac{\epsilon}{2}.$$

Furthermore, we have

$$\begin{aligned}&P\left(-\frac{1}{t}\log[\tilde{\nu}_i]_q^t \leq \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\,\mathbb{E}[\varphi_{ij}^{t}(\boldsymbol{\theta}_q)] - \epsilon\right)\\ &\leq P\left(\frac{1}{t}\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) \leq \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\,\mathbb{E}[\varphi_{ij}^{t}(\boldsymbol{\theta}_q)] - \frac{\epsilon}{2}\right).\end{aligned}$$

By applying McDiarmid's inequality $\forall\epsilon > 0$ and $\forall t \geq 1$, we have

$$\begin{aligned}&P\left(\left(\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\varphi_{ij}^{t}(\boldsymbol{\theta}_q) - t\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\,\mathbb{E}[\varphi_{ij}^{t}(\boldsymbol{\theta}_q)]\right) \leq -\frac{\epsilon t}{2}\right)\\ &\leq e^{\frac{-t\epsilon^2}{2A}}.\end{aligned}$$

Hence, we have

$$P\left(-\frac{1}{t}\log[\tilde{\nu}_i]_q^t \leq \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\,\mathbb{E}[\varphi_{ij}^{t}(\boldsymbol{\theta}_q)] - \epsilon\right) \leq e^{\frac{-t\epsilon^2}{2A}},$$

which implies

$$P\left([\tilde{\nu}_i]_q^t \geq e^{-t\left(\sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}]_{ij}]_{t}\,\mathbb{E}[\varphi_{ij}^{t}(\boldsymbol{\theta}_q)] - \epsilon\right)}\right) \leq e^{\frac{-t\epsilon^2}{2A}}.$$

Using this we obtain a bound on the error for any $\boldsymbol{\theta}_q$ across the entire network as follows:

$$P\left(\max_{i\in\mathcal{M}}\,[\tilde{\nu}_i]_q^t \geq e^{-t\left(\Lambda(\boldsymbol{W}^t) - \epsilon^{o}\right)}\right) \leq M(K-1)e^{\frac{-t\epsilon^2}{2A}},$$

where

$$\Lambda(\boldsymbol{W}^t) = \sum_{\substack{j=1\\j\neq i}}^{M}[[\boldsymbol{W}^t]_{ij}]D_{\mathrm{KL}}(\tilde{\boldsymbol{\nu}}_i^t|||\tilde{\boldsymbol{\nu}}_j^t).$$

Since $\Lambda(\boldsymbol{W}^t)$ is positive, with probability $1-\zeta$ we have $\max_{i\in\mathcal{M}}\,[\tilde{\nu}_i]_q^t < e^{-t\left(\Lambda(\boldsymbol{W}^t) - \epsilon\right)}$ when $t \geq 2A\epsilon^{-2}\log\left(M(K-1)\zeta^{-1}\right)$, which concludes the proof.

C. Proof of Lemma 1

(a) From (14), we have $[\boldsymbol{W}]_{ii}^t = \delta$, $[\boldsymbol{W}]_{i\hat{j}}^t = 1-\delta$, and $[\boldsymbol{W}]_{ij}^t = 0$ for all $j \in \mathcal{M}\setminus\{i,\hat{j}\}$, where $\hat{j} = \mathrm{argmax}_{j\in\mathcal{S}_i} D_{\mathrm{KL}}(\tilde{\boldsymbol{\nu}}_i^t||\tilde{\boldsymbol{\nu}}_j^t)$. Thus, if $j \in \mathcal{V}_i^t = \{i,\hat{j}\}$, $[\boldsymbol{W}]_{ij}^t > 0$, otherwise $[\boldsymbol{W}]_{ij}^t = 0$. Moreover, $\boldsymbol{W}^t$ is row-stochastic, since for all i we have

$$\begin{aligned}\sum_{j=1}^{M}[\boldsymbol{W}]_{ij}^t &= [\boldsymbol{W}]_{ii}^t + [\boldsymbol{W}]_{i\hat{j}}^t + \sum_{\substack{j=1\\j\neq i,\hat{j}}}^{M}[\boldsymbol{W}]_{ij}^t\\ &= \delta + 1 - \delta + 0\\ &= 1.\end{aligned}$$

(b) $\boldsymbol{W}^t$ has strictly positive diagonal entries since for all i, $[\boldsymbol{W}]_{ii}^t = \delta$.

(c) Let $0 < \eta < \min(\delta, 1-\delta)$. For all $i,j \in \mathcal{M}$, if $[\boldsymbol{W}]_{ij}^t > 0$, then $[\boldsymbol{W}]_{ij}^t > \eta$.

(d) First recall that $\mathcal{G}_c^t$ is constructed from $\boldsymbol{W}^t$ such that the set of edges $\mathcal{E}_c^t$ of $\mathcal{G}_c^t$ contains $(j,i) \in \mathcal{E}_p$ if and only if agent j is communicating to agent i at time t [*cf.* Section III-A]. We need to show that $\exists B < \infty$ such that $\left\{\mathcal{M}, \cup_{z=tB}^{(t+1)B-1}\mathcal{E}_c^z\right\}$ is strongly connected for all $t \geq 0$. Suppose that $\forall B < \infty$, $\left\{\mathcal{M}, \cup_{z=tB}^{(t+1)B-1}\mathcal{E}_c^z\right\}$ is *not* strongly connected for some $t \geq 0$. Let $B \geq 1$

be arbitrary. The preceding statement means that there exists at least one pair of agents (a_N^B, a_M^B) engaged in communication while disconnected from the rest of the network for the period $z \in [t_B B, (t_B+1)B-1]$ for some t_B[14]. Given that the physical graph is strongly connected, it follows that either a_N^B or a_M^B (or both) must have additional neighboring agents. Since (a_N^B, a_M^B) is disconnected from the rest of the agents for the period $[t_B B, (t_B+1)B-1]$, then it means that $a_N^B = \text{argmax}_{j \in \mathcal{S}_{a_M^B}} D_{\text{KL}}(\tilde{\boldsymbol{\nu}}^z_{a_M^B} || \tilde{\boldsymbol{\nu}}^z_j)$ for all $z \in [t_B B, (t_B+1)B-1]$. This suggests that a_M^B perceives the posterior distribution of a_N^B as the most dissimilar compared to its own among its neighboring nodes for all $B-1$ iterations. But note that B is arbitrary, and therefore one can choose it to be sufficiently large to yield whatever the similarity level between the posterior distributions of a_M^B and a_N^B, which is a contradiction.

ACKNOWLEDGMENT

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

[1] T. Alshammari, S. Samarakoon, A. Elgabli, and M. Bennis, "BayGo: Joint Bayesian learning and information-aware graph optimization," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artif. Intell. Statist.,* Ft. Lauderdale, FL, USA, Apr. 2017, pp. 1273–1282.

[3] J. Konecny, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.

[4] J. Park, S. Samarakoon, M. Bennis, and M. Debbah, "Wireless network intelligence at the edge," *Proc. IEEE*, vol. 107, no. 11, pp. 2204–2239, Nov. 2019.

[5] P. Kairouz et al., "Advances and open problems in federated learning," 2019, *arXiv:1912.04977*.

[6] M. Al-Shedivat, J. Gillenwater, E. Xing, and A. Rostamizadeh, "Federated learning via posterior averaging: A new perspective and practical algorithms," in *Proc. Int. Conf. Learn. Representations*, Vienna, Austria, May 2021, pp. 1–23.

[7] A. Nedić, A. Olshevsky, and C. A. Uribe, "A tutorial on distributed (non-Bayesian) learning: Problem, algorithms and results," in *Proc. IEEE Conf. Decis. Control*, Las Vegas, NV, USA, Dec. 2016, pp. 6795–6801.

[8] A. Jadbabaie, P. Molavi, A. Sandroni, and A. Tahbaz-Salehi, "Non-Bayesian social learning," *Games Econ. Behav.*, vol. 76, no. 1, pp. 210–225, Sep. 2012.

[9] A. Nedic, A. Olshevsky, and C. Uribe, "A tutorial on distributed (non-Bayesian) learning: Problem, algorithms, and results," in *Proc. Conf. Decis. Control,* Las Vegas, NV, USA, Dec. 2016, pp. 6795–6801.

[10] A. Nedić, A. Olshevsky, and C. A. Uribe, "Nonasymptotic convergence rates for cooperative learning over time-varying directed graphs," in *Proc. Amer. Control Conf.*, Chicago, IL, USA, Jul. 2015, pp. 5884–5889.

[11] A. Lalitha, T. Javidi, and A. Sarwate, "Social learning and distributed hypothesis testing," *IEEE Trans. Inf. Theory*, vol. 64, no. 9, pp. 6161–6179, Sep. 2018.

[12] A. Nedić, A. Olshevsky, and C. A. Uribe, "Fast convergence rates for distributed non-Bayesian learning," *IEEE Trans. Automat. Control*, vol. 62, no. 11, pp. 5538–5553, Nov 2017.

[13] A. Lalitha, X. Wang, O. Kilinc, Y. Lu, T. Javidi, and F. Koushanfar, "Decentralized Bayesian learning over graphs," 2019, *arXiv:1905.10466*.

[14] A. Jadbabaie, P. Molavi, and A. Tahbaz-Salehi, "Information heterogeneity and the speed of learning in social networks," Columbia Business School, New York, NY, USA, Research Paper no. 13–28, May 2013. Accessed: Jul. 2020. [Online]. Available: https://ssrn.com/abstract=2266979

[15] A. Lalitha, O. Kilinc, T. Javidi, and F. Koushanfar, "Peer-to-peer federated learning on graphs," 2019, *arXiv:1901.11173*.

[16] S. Kar, M. F. Moura, and H. V. Poor, "*QD*-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1848–1862, Apr. 2013.

[17] S. Kar and M. Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1383–1400, Mar. 2010.

[18] V. Saligrama, M. Alanyali, and O. Savas, "Distributed detection in sensor networks with packet losses and finite capacity links," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4118–4132, Nov. 2006.

[19] R. Rahman, M. Alanyali, and V. Sligrama, "Distributed tracking in multihop sensor networks with communication delays," *IEEE Trans. Signal Process.*, vol. 55, no. 9, pp. 4656–4668, Sep. 2007.

[20] S. Shahrampour and A. Jadbabaie, "Exponentially fast parameter estimation in networks using distributed dual averaging," in *Proc. IEEE Conf. Decis. Control*, Firenze, Italy, Dec. 2013, pp. 6196–6201.

[21] V. Bordignon, V. Matta, and A. H. Sayed, "Adaptive social learning," *IEEE Trans. Inf. Theory*, vol. 67, no. 9, pp. 6053–6081, Sep. 2021.

[22] A. H. Sayed, Adaptation, learning, and optimization Over networks. *Found. Trends Mach. Learn.*, vol. 7, nos. 4–5, pp. 311–801, 2014.

[23] P. Hu, V. Bordignon, S. Vlaski, and A. H. Sayed, "Optimal aggregation strategies for social learning over graphs," *IEEE Trans. Inf. Theory*, vol. 69, no. 9, pp. 6048–6070, Sep. 2023.

[24] P. Hu, V. Bordignon, S. Vlaski, and A. H. Sayed, "Optimal combination policies for adaptive social learning," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)* , 2022, pp. 5842–5846.

[25] S. Shahrampour, A. Rakhlin, and A. Jadbabaie, "Distributed detection: Finite-time analysis and impact of network topology," *IEEE Trans. Autom. Control*, vol. 61, no. 11, pp. 3256–3268, Nov. 2016.

[26] S. Ross, *Stochastic Processes*, 2nd ed. Hoboken, NJ, USA: Wiley, 1996.

[27] M. H. DeGroot, "Reaching a consensus," *J. Am. Statist. Assoc.*, vol. 69, no. 345, pp. 118–121, Apr. 2012.

[28] J. Wang, A. Sahu, Z. Yang, G. Joshi, and S. Kar, "MATCHA: Speeding up decentralized SGD via matching decomposition sampling," in *Proc. 6th Indian Control Conf.*, Hyderabad, India, Dec. 2019, pp. 1–2.

[29] "Bodyfat database." StatLib. Accessed: Jul. 2020. [Online]. Available: https://www.csie.ntu.edu.tw/cjlin/libsvmtools/datasets/regression/bodyfat

[30] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010. Accessed: Jun. 2021. [Online]. Available: http://yann.lecun.com/exdb/mnist

[31] Y. Gal, *Uncertainity in Deep Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2016.

[32] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *Proc. Int. Conf. Mach. Learn.*, Lille, France, Jul. 2015, pp. 1–10.

[33] C. Nguyen, T. Li, T. D. Bui, and R. Turner, "Variational continual learning," in *Proc. Int. Conf. Mach. Learn.*, Vancouver, BC, Canada, Apr. 2018, pp. 1–18.

Tamara AlShammari (Student Member, IEEE) received the B.Sc. degree in computer engineering from the University of Jordan, Amman, Jordan, where she was a top student at class, in 2012, and the M.Sc. degree in electrical and computer engineering from Oregon State University, Oregon, USA, in 2014. She is currently working toward the Ph.D. degree with the University of Oulu, Oulu, Finland. Her research interests include Bayesian learning, multi-agent systems, semantic communication, machine learning, and convex optimization.

[14] Owing to the construction of $\boldsymbol{W}^t$, each agent must communicate with exactly one other agent, thereby preventing any single agent from being isolated from the rest.

Chathuranga Weeraddana (Member, IEEE) received the Ph.D. degree from the University of Oulu, Oulu, Finland, in 2011. From 2012 to 2014, he was a Postdoctoral Researcher with the KTH Royal Institute of Technology, Stockholm, Sweden. From 2015 to 2017 and from 2018 to 2022, he was a Senior Lecturer with Sri Lanka Institute of Information Technology, Sri Lanka, and the University of Moratuwa, Sri Lanka, respectively. He is currently a Senior Research Fellow with the Center for Wireless Communications, University of Oulu.

Mehdi Bennis (Fellow, IEEE) is a Professor with Centre for Wireless Communications, University of Oulu, Finland, and Head of the Intelligent Connectivity and Networks/Systems Group (ICON). His research interests are in radio resource management, game theory and distributed AI in 5G/6G networks. He has published more than 300 research papers in international conferences, journals and book chapters. He has been the recipient of several prestigious awards including the 2015 Fred W. Ellersick Prize from IEEE Communications Society, the 2016 Best Tutorial Prize from IEEE Communications Society, the 2017 EURASIP Best paper Award for *Journal of Wireless Communications and Networks*, the all-University of Oulu award for research, the 2019 IEEE ComSoc Radio Communications Committee Early Achievement Award and the 2020–2023 Clarviate Highly Cited Researcher by the Web of Science.