Deposited on 23 February 2024

Channel-Driven Monte Carlo Sampling for Bayesian Distributed Learning in Wireless Data Centers

Dongzhu Liu and Osvaldo Simeone

Abstract—Conventional frequentist learning, as assumed by existing federated learning protocols, is limited in its ability to quantify uncertainty, incorporate prior knowledge, guide active learning, and enable continual learning. Bayesian learning provides a principled approach to address all these limitations, at the cost of an increase in computational complexity. This paper studies distributed Bayesian learning in a wireless data center setting encompassing a central server and multiple distributed workers. Prior work on wireless distributed learning has focused exclusively on frequentist learning, and has introduced the idea of leveraging uncoded transmission to enable "over-the-air" computing. Unlike frequentist learning, Bayesian learning aims at evaluating approximations or samples from a global posterior distribution in the model parameter space. This work investigates for the first time the design of distributed one-shot, or "embarrassingly parallel", Bayesian learning protocols in wireless data centers via consensus Monte Carlo (CMC). Uncoded transmission is introduced not only as a way to implement "over-the-air" computing, but also as a mechanism to deploy *channel-driven MC sampling*: Rather than treating channel noise as a nuisance to be mitigated, channel-driven sampling utilizes channel noise as an integral part of the MC sampling process. A simple wireless CMC scheme is first proposed that is asymptotically optimal under Gaussian local posteriors. Then, for arbitrary local posteriors, a variational optimization strategy is introduced. Simulation results demonstrate that, if properly accounted for, channel noise can indeed contribute to MC sampling and does not necessarily decrease the accuracy level.

Index Terms—Distributed Bayesian learning, consensus Monte Carlo, over-the-air computation, federated learning, uncoded transmission, wireless data centers.

I. INTRODUCTION

The availability of massive data sets for machine learning has led to the development of distributed computing platforms that can improve the computational efficiency of training tasks by partitioning a global data set among multiple workers [1]–[6]. Decentralization may come at the cost of a possible degradation in accuracy due to the reliance on local optimization steps [7]. Moreover, when workers are connected to the central server via capacity-limited channels, communication-related impairments can cause additional performance degradation [6], [8]–[16].

As we will review, existing works on wireless distributed learning focus on a standard *frequentist* formulation of the learning problem, whereby the goal is to identify a single vector of model parameters by minimizing (a function of) the

training loss [9], [14], [15], [17], [18]. Much research has been specifically carried out on studying the advantages of *uncoded transmission* to enable "over-the-air" computing [8], [9], [14], [15]. Frequentist learning is known to be limited in its ability to quantify epistemic uncertainty, i.e., uncertainty in the model parameter space, yielding overconfident decisions [19], [20]. In contrast, *Bayesian* learning provides a principled way to obtain predictive models that account for epistemic uncertainty. This is done by optimizing not over individual model parameter vectors, but over distributions in the model parameter space [21], [22]. Other advantages of Bayesian learning over frequentist learning include the capacity to incorporate prior knowledge, to guide active learning, and to enable continual learning [21]. Practical implementations of Bayesian learning involve either approximating the posterior distribution over the model parameters – an approach known as *variational inference (VI)*, or producing approximate samples from the posterior – an approach known as *Monte Carlo (MC) sampling* [23].

This paper considers, for the first time, distributed Bayesian learning in wireless networks. We specifically focus on a wireless data center setting as illustrated in Fig. 1 [6], and on the standard class of "embarrassingly parallel" *consensus Monte Carlo (CMC)* protocols [24], [25]. In wireless data centers, the global data set is available at the central server, which distributes subsets of data points to the workers. Wireless data centers have been introduced as an alternative to conventional wired topologies to reduce cost and to cope with unbalanced traffic [6]. In CMC protocols, first, each worker produces samples from a local posterior based on the locally available data; then, the server carries out a one-shot aggregation of the received messages in order to approximate samples from the global posterior distribution.

As a key contribution, we introduce the novel idea of *channel-driven MC sampling*. We specifically demonstrate that, under uncoded wireless transmission, channel noise can be utilized as an integral component of MC sampling. As a result, if properly accounted for, channel noise may not cause harm to the performance of MC-based distributed Bayesian learning. This is in stark contrast to wireless distributed learning based on frequentist principles, for which channel noise is generally detrimental to the learning performance. An exception to this statement was demonstrated in [26], which argues that the presence of noise can be useful when training using non-convex objectives.

A. Wireless Distributed Learning

Most work on wireless distributed learning focuses on the "federated" setting, in which data originates at the workers and

it remains unknown to the server. In contrast, in a wireless data center system, the server has access to the global data set, and it distributes data to the workers to benefit from computational parallelism. In both cases, distributed learning protocols can be embarrassingly parallel, that is, one-shot, or iterative. In the former case, which we focus on, the workers carry out local computations and communicate only once with the server, which aggregates the results of the local computations [24], [25], [27]; while, in the latter, local computing and communication steps are alternated. To counteract hostile channel fading and additive channel noise conditions in wireless distributed learning, various strategies have been proposed including optimal power control [14]–[16] and over-the-air computing [8]–[10]. Over-the-Air computing utilizes non-orthogonal multi-access (NOMA) and uncoded transmission to enable the aggregation of signals transmitted by the workers "over-the-air", leveraging the waveform-superposition property of a multi-access channel. For the wireless data center setting, existing works address issues such as optimizing the number of workers to balance the tradeoff between communication latency and computation time [28], and designing data partition strategy to cope with straggling workers [29].

B. Distributed Bayesian Learning

To the best of our knowledge, as reviewed above, prior work on distributed wireless learning – in both federated and wireless data center settings – is based on frequentist learning. As mentioned, since exact Bayesian learning is generally intractable, existing Bayesian learning schemes are based on approximate methods leveraging either VI or MC sampling. VI assumes a parametric family for the posterior distribution of the model parameters, and its performance is generally limited by the bias due to the variational approximation [23]. In contrast, MC sampling aims at obtaining samples from the posterior distribution, and is asymptotically exact in the number of samples. Given their reliance on parametric optimization, distributed VI schemes can be implemented in a manner similar to existing frequentist protocols in federated learning, including local stochastic gradient descent (SGD) optimization and global aggregation [30], [31].

Distributed MC sampling can also be implemented either in a one-shot fashion or an iterative manner. Most common is the former class of protocols, which are known as CMC [24]. The key problem in the design of CMC schemes is how to aggregate the local samples provided by the workers [27]. In this regard, by approximating the posterior distribution as Gaussian, one can obtain optimal weights in closed form, which can be estimated using the local samples [24]. For general distributions, reference [25] proposed to design the aggregation function via a VI formulation. Less well studied are iterative protocols, which include distributed stochastic gradient Markov Chain MC strategies [32]. These protocols are inherently sequential, making parallel implementations challenging.

All prior works on CMC assume ideal communication. Nevertheless, the existence of channel noise in wireless data centers can severely distort the distribution of the received samples. This observation motivates us to seek optimized aggregation strategies that account for the impairments caused by wireless transmission. We refer to the resulting setting as *wireless CMC*.

C. Contributions, Organization, and Notations

In this paper, we study for the first time the design of wireless CMC protocols for distributed one-shot Bayesian learning in a wireless data center setting. As illustrated in Fig. 1, the global data set is partitioned into disjoint subsets for the distributed workers to enable the implementation of CMC. After local sampling, each worker uploads the produced local samples to the server via uncoded transmission via either orthogonal or non-orthogonal multi-accesses protocols. At the server, the received signals are aggregated by using an optimized function to produce approximated samples from the global posterior. The main findings and contributions of the paper can be summarized as follows.

• **Wireless Gaussian Consensus Monte Carlo (WGCMC):** To start, we study as in [24] the baseline case in which the local posteriors follow Gaussian distributions, and we introduce a wireless CMC protocol that is asymptotically optimal with respect to the number of produced samples. The solution automatically adapts to the signal-to-noise ratio (SNR) level, and, under Gaussian local posteriors, it is demonstrated to leverage channel noise as a component of the sample generation process, instead of a nuisance to be mitigated. We consider orthogonal multiple access (OMA), as well as NOMA with over-the-air computing to enhance spectral efficiency.

• **Wireless Variational Consensus Monte Carlo (WVCMC):** We then target the general case with arbitrary local posteriors. We propose a variational design methodology that aims at reducing the Kullback-Liebler (KL) divergence of the distribution of the aggregated samples with respect to the true global posterior. The proposed principled solution approach is based on SGD steps at the server.

• **Experiments:** We provide extensive numerical results on both synthetic and real data sets to demonstrate the advantages of channel-driven CMC.

Organization: The remainder of the paper is organized as follows. Section II introduces the system model. Section III introduces the necessary background on CMC. Section IV presents the general form of signal design for wireless CMC, while WGCMC is introduced in Section V and WVCMC in Section VI. Section VII provides numerical results, followed by conclusions in Section VIII.

Notations: The following notations are used throughout the paper. $\otimes$ denotes Kronecker product. $\mathbf{1}_l$ is all-ones $l \times 1$ vector. $\delta[\cdot]$ is the Dirac delta function. $\mathbf{A}^{-1/2} = \mathbf{U}\mathbf{\Lambda}^{-1/2}\mathbf{U}^{\mathsf{T}}$ is the operation for any positive semidefinite matrix $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathsf{T}}$, where $\mathbf{U}$ is comprised by singular vectors and $\mathbf{\Lambda}$ represents the non-zero diagonalized singular values. $[\mathbf{A}]^+$ sets the negative singular values of $\mathbf{A}$ to 0, and $\mathcal{I}(\cdot)$ is the indicator function.
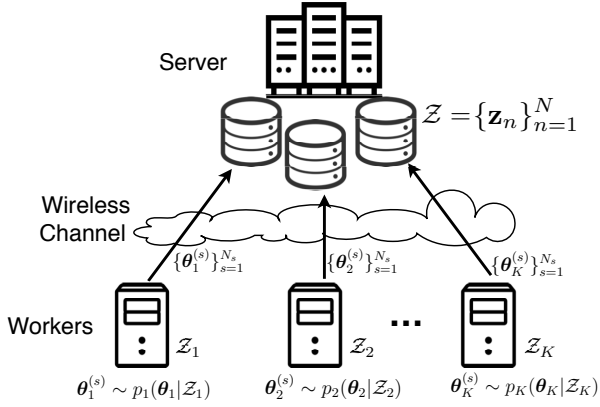
Figure 1. Bayesian learning via consensus Monte Carlo (CMC), i.e., embarrassingly parallel distributed MC sampling, in a wireless data center.

II. SYSTEM MODEL

A. Setting

As shown in Fig. 1, we consider a wireless data center setting [6] including a server and K workers connected to it via a shared wireless channel. The global dataset $\mathcal{Z}$ at the server consists of N data samples $\mathcal{Z} = \{\mathbf{z}_n\}_{n=1}^N$. For supervised learning applications, each data sample $\mathbf{z}_n = (\mathbf{u}_n, v_n)$ is in turn partitioned into a covariate vector $\mathbf{u}_n$ and a label v_n; while, for unsupervised learning applications such as generative modeling, it consists of a single vector $\mathbf{z}_n$ (see, e.g., [22]). The goal of the system is to carry out Bayesian learning via Monte Carlo (MC) sampling. This entails generating samples that are (approximately) distributed according to posterior distribution $p(\boldsymbol{\theta}|\mathcal{Z})$ of model parameter $\boldsymbol{\theta} \in \mathbb{R}^d$ given data set $\mathcal{Z}$. Samples from the posterior can be used to obtain ensemble predictive distributions that approximate the optimal Bayesian predictor (see, e.g., [21], [22], [33]).

Denote as $p(\mathbf{z}|\boldsymbol{\theta})$ the likelihood of model parameter vector $\boldsymbol{\theta}$ for data point $\mathbf{z}$, and as $p(\boldsymbol{\theta})$ the prior distribution of the model parameter vector. Under the standard assumption of conditionally independent and identical (i.i.d.) data points, the global posterior distribution can be written as

$$(\text{Global posterior}) \quad p(\boldsymbol{\theta}|\mathcal{Z}) \propto p(\boldsymbol{\theta}) \prod_{n=1}^N p(\mathbf{z}_n|\boldsymbol{\theta}). \quad (1)$$

Note that, for supervised learning problems, the likelihood can be written as $p(\mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{u}, v|\boldsymbol{\theta})$ for generative models and as $p(\mathbf{z}|\boldsymbol{\theta}) = p(v|\mathbf{u}, \boldsymbol{\theta})$ for discriminative models [22]. We make the key assumption that exact sampling from the posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ is computationally intensive due to the size of the data set $\mathcal{Z}$. Possible solutions include adaptive subsampling procedures [34] and efficient stochastic gradient-based algorithms such as stochastic gradient Langevin dynamics (SGLD) [35], which require limited access to data at the server at any given time. This paper considers an alternative, and potentially more effective approach that leverages the presence of the K distributed workers in the wireless data center system in Fig. 1.

We specifically propose, and study, the wireless implementation of distributed MC sampling solutions, whereby workers perform parallel MC sampling on distinct subsets of the data [23]. We focus on the class of embarrassingly parallel strategies, known as consensus Monte Carlo (CMC), in which communication from workers to server occurs only once, after local processing at the workers is completed. In CMC, the global dataset $\mathcal{Z}$ is partitioned into K non-overlapping subsets $\mathcal{Z}_1, \cdots, \mathcal{Z}_K$, each of size N_k, and each data set $\mathcal{Z}_k$ is allocated to the k-th worker. We have $\sum_{k=1}^K N_k = N$. Data delivery is offline, and hence one can consider data to be available without distortion at the workers. It is assumed that the k-th worker has sufficient computational resources to generate samples from the subposterior $p_k(\boldsymbol{\theta}|\mathcal{Z}_k)$. The subposterior depends on the allocated subset $\mathcal{Z}_k$ and on the underweighted prior $p(\boldsymbol{\theta})^{1/K}$ as

$$(\text{Subposterior}) \quad p_k(\boldsymbol{\theta}|\mathcal{Z}_k) \propto p(\boldsymbol{\theta})^{1/K} p(\mathcal{Z}_k|\boldsymbol{\theta}), \quad (2)$$

where we have defined the local likelihood function as $p(\mathcal{Z}_k|\boldsymbol{\theta}) = \prod_{\mathbf{z} \in \mathcal{Z}_k} p(\mathbf{z}|\boldsymbol{\theta})$. This assumption is justified by the fact that the complexity of MC sampling scales up with the size of the underlying data set [23]–[25] (see Appendix C for some additional discussion on this point). Note that the global posterior (1) can be expressed as the product of subposteriors (2), upon normalization, as

$$p(\boldsymbol{\theta}|\mathcal{Z}) \propto \prod_{k=1}^K p_k(\boldsymbol{\theta}|\mathcal{Z}_k). \quad (3)$$

Each worker $k = 1, \ldots, K$ generates S samples $\boldsymbol{\theta}_k^{(s)} \sim p_k(\boldsymbol{\theta}|\mathcal{Z}_k)$ for $s = 1, \ldots, S$ and transmits them one-by-one, as detailed in the next section, over the wireless channel to the server. Based on the received signals on the wireless channel, the server aims at obtaining samples $\boldsymbol{\theta}^{(s)}$ for $s = 1, \ldots, S$ that are approximately drawn from the global posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ by applying an aggregation function $F(\cdot)$. We note that the aggregation function $F(\cdot)$ may depend on the data set $\mathcal{Z}$, which is available at the server.

In existing CMC methods [24], [25], the samples $\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K$ are assumed to be received noiselessly to the server, and the aggregation function is chosen as the linear combination

$$\boldsymbol{\theta}^{(s)} = F(\boldsymbol{\theta}_1^{(s)}, \cdots, \boldsymbol{\theta}_K^{(s)}) = \sum_{k=1}^K \mathbf{W}_k \boldsymbol{\theta}_k^{(s)} \quad (4)$$

for some $d \times d$ weight matrices $\mathbf{W}_1, \cdots, \mathbf{W}_k$. CMC does not require feedback from the edge server or other communications beyond the transmission of the local posterior samples from workers to the server. The weight matrices are optimized based on the samples $\{\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K\}_{s=1}^S$ and on data set $\mathcal{Z}$ [24], [25]. CMC is successful when the distribution $q_F(\boldsymbol{\theta})$ of the samples $\boldsymbol{\theta}^{(s)}$ computed at the server is a good approximation of the global posterior $p(\boldsymbol{\theta}|\mathcal{Z})$.

In this paper, we will consider for the first time the case in which the samples $\{\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K\}_{s=1}^S$ have to be communicated to the server over a wireless channel, and demonstrate the novel use of channel noise for Monte Carlo sampling. The proposed channel-driven sampling is enabled by the use of analog communication for the transmission of the samples $\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K$, whereby channel noise is directly added to the samples produced and transmitted by the workers. The CMC

aggregation function (4) can be implemented using both orthogonal and non-orthogonal multiple access schemes, which have distinct advantages and disadvantages. With orthogonal multiple access (OMA) transmission, the local samples $\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K$ are received separately. Therefore, channel noise acts as a distinct source of randomness for each sample, and arbitrary weight matrices $\{\mathbf{W}_k\}_{k=1}^K$ can be applied at the server. In contrast, under non-orthogonal multiple access (NOMA) transmission, AirComp can increase communication efficiency, but at the costs of sharing the channel noise across samples and of limiting the choice of the weight matrices $\{\mathbf{W}_k\}_{k=1}^K$ (see Sec. IV).

B. Communication Model

Following recent work on communication for machine learning [8], [10], [17], [36], [37], we assume that workers communicate with the server via a general vector wireless communication model that may account for multi-carrier and/or single/multi-antenna setups. Accordingly, each worker transmits m_t symbols, over time, space, and/or frequency domains, and the server receives m_r symbols over a channel described by a real matrix and additive noise. We focus on a real channel model with no loss of generality, since complex vector channels can be described in terms of real vector channel of twice the size of the original model [14]. We will specifically consider two general classes of communication protocols, namely orthogonal multiple access (OMA) and non-orthogonal multiple access (NOMA).

1) Orthogonal multiple access (OMA): Under OMA, workers transmit over orthogonal communication resources, and the $m_r \times 1$ received signal from the k-th worker is

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{n}_k, \tag{5}$$

where $\mathbf{H}_k \in \mathbb{R}^{m_r \times m_t}$ is the channel matrix between worker k and server; $\mathbf{x}_k \in \mathbb{R}^{m_t}$ is the signal transmitted by worker k; and $\mathbf{n}_k \in \mathbb{R}^{m_r}$ is channel noise, with entries being independent and identically distributed (i.i.d.) $\mathcal{N}(0, N_0)$ random variables (rvs). Specifically, in each communication block, a given worker k is scheduled to transmit a sample $\boldsymbol{\theta}_k^{(s)}$. We focus on analog, i.e., uncoded transmission strategies, whereby $\boldsymbol{\theta}_k^{(s)}$ is mapped through a linear transformation to $\mathbf{x}_k$, as we will detail in Sec. IV.

2) Non-orthogonal multiple access (NOMA): Under NOMA, all workers transmit simultaneously, and the $m_r \times 1$ received signal given by the noisy superposition

$$\mathbf{y} = \sum_{k=1}^K \mathbf{H}_k \mathbf{x}_k + \mathbf{n}, \tag{6}$$

where $\mathbf{H}_k \in \mathbb{R}^{m_r \times m_t}$ is the channel matrix between worker k and server; $\mathbf{x}_k \in \mathbb{R}^{m_t}$ is the signal transmitted by worker k; and $\mathbf{n} \in \mathbb{R}^{m_r}$ is channel noise, with entries being i.i.d. $\mathcal{N}(0, N_0)$ random rvs. As for OMA, each signal $\mathbf{x}_k$ encodes a sample $\boldsymbol{\theta}_k^{(s)}$ produced by worker k through a linear transformation.

In both cases, the channel matrices generally change from one communication block, given by (5) or (6), to another. The channel matrix $\mathbf{H}_k$ and its statistics are assumed to be known at worker k. This can be achieved via the broadcasting of pilots by the server in time-division duplex (TDD) systems, or via uplink pilot transmission and feedback from the server for frequency-division duplex (FDD) systems. An important aspect of the model is that, in order to receive information from all devices, NOMA can operate with a fraction $1/K$ of the resources of OMA, since the server obtains a superposition of all K signals in a single communication block. This property has been successfully leveraged by AirComp strategies based on analog transmission [9], [10], [14], [38], [39].

For both OMA and NOMA, the transmit per-block power constraint is set as the inequality

$$\mathsf{E}[\|\mathbf{x}_k\|^2] \leq P, \tag{7}$$

where the expectation is taken with respect to the distribution of the channels and of the encoded samples. The average over the channel implies consideration of a long-term power constraint. This is sensible for the common setting in which the workers transmit multiple samples $S > 1$ in order to enable posterior averaging at the server, e.g., for ensemble prediction (see Sec. VII). A short-term power constraint can be accounted for in a similar way.

Throughout the paper, we assume the conditions $m_t \geq m_r \geq d$, where we recall that d is the dimension of the model vector $\boldsymbol{\theta}$. As is common in the Bayesian learning literature, the value of d is generally small in prior CMC works [24], [25]. For larger models, i.e., for larger d, the transmission of one sample would require multiple coherence blocks and/or use sufficiently large antenna arrays. This setting can be modelled by introducing block diagonal channel matrices, including the channel matrices for each block on the block diagonal. The integration of analog compression techniques (see, e.g., [40], [41]) with the wireless CMC strategies to be developed here can alleviate this problem and is left for future work. Finally, we assume that the channel matrices $\{\mathbf{H}_k\}_{k=1}^K$ have full rank with probability one. This is a common assumption in many studies on multi-antenna wireless communication (see, e.g., [42]).

III. BACKGROUND: GAUSSIAN CONSENSUS MONTE CARLO

In this section, we review the Gaussian CMC (GCMC) scheme introduced in [24] that assumes noiseless, ideal communication. Accordingly, GCMC assumes that the local samples $\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^K$, for $s = 1, \ldots, S$ are available at server. We will argue in Sec. V that a direct application of this scheme in the wireless setting under study is strictly suboptimal, even in the asymptotic regime of large number of samples, S, motivating the introduction of alternative schemes.

GCMC is based on the observation that, if each local sub-posterior $p_k(\boldsymbol{\theta}|\mathcal{Z}_k)$ is a Gaussian $\mathcal{N}(\boldsymbol{\mu}_k, \mathbf{C}_k)$ distribution with invertible covariance matrix $\mathbf{C}_k$, the global posterior is the Gaussian $\mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$ distribution, where $\boldsymbol{\mu} = \mathbf{C}(\sum_{k=1}^K \mathbf{C}_k^{-1} \boldsymbol{\mu}_k)$ and $\mathbf{C} = (\sum_{k=1}^K \mathbf{C}_k^{-1})^{-1}$. If follows that, given local samples $\boldsymbol{\theta}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \mathbf{C}_k)$, for $k = 1, \ldots, K$, a global

sample $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{C})$ can be obtained by applying the linear combination (4) with weight matrices

$$\mathbf{W}_k = \Big(\sum_{k=1}^{K} \mathbf{C}_k^{-1} \Big)^{-1} \mathbf{C}_k^{-1}. \tag{8}$$

Since the matrices $\mathbf{C}_1, \cdots, \mathbf{C}_K$ are not known, GCMC estimates them by using the S samples $\{\boldsymbol{\theta}_k^{(s)}\}_{k=1}^{K}$ with $s = 1, \ldots, S$ received from the workers. In particular, given the S samples $\{\boldsymbol{\theta}_k^{(s)}\}_{s=1}^{S}$ received form worker k, covariance $\mathbf{C}_k$ is estimated as

$$\widehat{\mathbf{C}}_k = \frac{1}{S-1} \sum_{s=1}^{S} \Big(\boldsymbol{\theta}_k^{(s)} - \widehat{\boldsymbol{\mu}}_k \Big) \Big(\boldsymbol{\theta}_k^{(s)} - \widehat{\boldsymbol{\mu}}_k \Big)^{\mathsf{T}}, \tag{9}$$

with $\widehat{\boldsymbol{\mu}}_k = \frac{1}{S} \sum_{s=1}^{S} \boldsymbol{\theta}_k^{(s)}$. Then, the function (4) is applied using the estimate $\widehat{\mathbf{C}}_k$ (9) in lieu of $\mathbf{C}_k$ in the weights (8). Accordingly, GCMC computes each global sample $\boldsymbol{\theta}^{(s)}$ as

$$\text{(GCMC)} \quad \boldsymbol{\theta}^{(s)} = \sum_{k=1}^{K} \underbrace{\Big(\sum_{k'=1}^{K} \widehat{\mathbf{C}}_{k'}^{-1} \Big)^{-1} \widehat{\mathbf{C}}_k^{-1}}_{\mathbf{W}_k} \boldsymbol{\theta}_k^{(s)}. \tag{10}$$

Note that, as a result, the function (10) applied by the server is non-linear in the samples $\{\{\boldsymbol{\theta}^{(s)}\}_{k=1}^{K}\}_{s=1}^{S}$. It is also noted that GCMC does not require the use of the data set $\mathcal{Z}$ at the server. The matrix inversion in (10) incurs prohibitive computational complexity for high-dimensional models. One possible solution, as proposed in [24], [25], is to restrict the weight matrices $\mathbf{W}_k$ to be diagonal, with each element corresponding to the inverse of the marginal variance of the samples.

While GCMC is defined under the assumption of Gaussian subposteriors, reference [24] argues that the choice (8)–(9) for the combining matrices $\{\mathbf{W}_k\}_{k=1}^{K}$ is suitable for other settings as well. This is theoretically well justified for large data sets, i.e., when the data set sizes $\{N_k\}_{k=1}^{K}$ are large as compared to the model dimension, since the subposteriors in this case can be well approximated by Gaussian distributions according to Bernstein-von Mises theorem [43]. In this paper, we will adopt throughout a linear aggregation as in (4), although the methods to be introduced in the next sections apply to more general parametric aggregation functions.

IV. WIRELESS CONSENSUS MONTE CARLO: SIGNAL DESIGN

Unlike prior works on CMC [24], [25], including the baseline GCMC scheme reviewed in the previous section, in the following sections, we propose solutions that account for the wireless communication channel between workers and server. A key novel idea introduced by this work is that the noise added by the wireless channel may not cause any performance degradation as compared to an ideal scenario with noiseless transmission: Noise can contribute to MC sampling, and is not necessarily a nuisance to be mitigated. In this section, we describe the general form of the considered uncoded transmission strategies for CMC. The next two sections propose specific wireless CMC protocols. Throughout, we focus on

analog, or uncoded, transmission strategies with the aims of leveraging channel noise for sampling and enabling AirComp solutions via NOMA. Comparison with digital schemes follow along similar lines as in [9], [18], and is left as future work.

A. Signal Design in OMA

In OMA, in each communication block, a single sample $\boldsymbol{\theta}_k^{(s)}$ is transmitted by the scheduled device k to the server. Dropping temporarily the superscript s to simplify the notation, we assume uncoded transmission of the form

$$\text{(Transmitted Signal)} \quad \mathbf{x}_k = \mathbf{U}_k \boldsymbol{\theta}_k, \tag{11}$$

where $\mathbf{U}_k$ is an $\mathbb{R}^{m_t \times d}$ precoding matrix. We assume that $\mathbf{U}_k$ factorizes into a zero-forcing pre-equalization matrix $\mathbf{H}_k^{\dagger} = \mathbf{H}_k^{\mathsf{T}}(\mathbf{H}_k \mathbf{H}_k^{\mathsf{T}})^{-1}$ and a $m_r \times d$ encoding matrix as $\mathbf{U}_k = \mathbf{H}_k^{\dagger} \mathbf{E}_k$. The use of zero-forcing is aligned with the majority of studies on uncoded transmission for machine learning [14], [38], [39], [44], although generalizations are possible. The encoding matrix $\mathbf{E}_k$ is fixed, and not subject to optimization. For instance, if m_r is a multiple of d, e.g., $m_r = ld$ for some integer l, one can set $\mathbf{E}_k \propto \mathbf{1}_l \otimes \mathbf{I}_d$ to implement a form of analog repetition coding. Optimization of analog coding strategies is left for future work. The encoding matrix $\mathbf{E}_k$ is subject to the power constraint (7), which imposes the inequality

$$\mathsf{E}[\|\mathbf{x}_k\|^2] = \mathsf{E}[\mathrm{tr}(\mathbf{U}_k \boldsymbol{\theta}_k \boldsymbol{\theta}_k^{\mathsf{T}} \mathbf{U}_k^{\mathsf{T}})] \leq P. \tag{12}$$

Since the distribution of the samples $\{\boldsymbol{\theta}_k\}_{k=1}^{K}$ is unknown, we will specifically impose (12) by averaging over the available S samples as

$$\frac{1}{S} \sum_{s=1}^{S} \mathrm{tr} \left(\mathsf{E}\big[(\mathbf{H}_k \mathbf{H}_k^{\mathsf{T}})^{-1} \big] \mathbf{E}_k \boldsymbol{\theta}_k^{(s)} (\boldsymbol{\theta}_k^{(s)})^{\mathsf{T}} \mathbf{E}_k^{\mathsf{T}} \right) \leq P. \tag{13}$$

The inequality (13) can be thought of an energy constraint across the transmission of the S samples for each worker k. For the example of analog repetition coding, constraint (13) is satisfied with $\mathbf{E}_k = \sqrt{P_k} \mathbf{1}_l \otimes \mathbf{I}_d$, where $P_k = PS / \sum_{s=1}^{S} \mathrm{tr}\big(\mathsf{E}\big[(\mathbf{H}_k \mathbf{H}_k^{\mathsf{T}})^{-1} \big] (\mathbf{1}_l \otimes \mathbf{I}_d) \boldsymbol{\theta}_k^{(s)} (\boldsymbol{\theta}_k^{(s)})^{\mathsf{T}} (\mathbf{1}_l \otimes \mathbf{I}_d)^{\mathsf{T}} \big)$.

At the receiver side, for OMA, we propose to aggregate the received signals, $\mathbf{y}_1, \cdots, \mathbf{y}_K$ in a manner akin to (4) as

$$\text{(Aggregation OMA)} \quad \boldsymbol{\theta} = F(\mathbf{y}_1, \cdots, \mathbf{y}_K) = \sum_{k=1}^{K} \mathbf{W}_k \mathbf{y}_k, \tag{14}$$

where $\mathbf{W}_1, \cdots, \mathbf{W}_K$ are $d \times m_r$ weight matrices. Note that the aggregation in (14) is applied separately for each set of received signals $\{\mathbf{y}_k^{(s)}\}_{k=1}^{K}$. The weights $\{\mathbf{W}_k\}_{k=1}^{K}$ are to be optimized based on the received signals $\{\mathbf{y}_k^{(s)}\}_{k=1}^{K}$ for all samples $s = 1, \ldots, S$, as well as, possibly, based on the data set $\mathcal{Z}$.

B. Signal Design in NOMA

Under NOMA, in each communication block, all workers transmit one of their local samples simultaneously by using uncoded transmission as in (11). As in OMA, matrix $\mathbf{U}_k$

is factorized as the product $\mathbf{U}_k = \mathbf{H}_k^{\dagger}\mathbf{E}_k$. Furthermore, to enable the implementation of AirComp in a manner similar to [14], [38], [44], we assume that all the transmitted signals are aligned by using the same encoding matrix, which denoted as $\mathbf{E}_k = \mathbf{E}$ for all $k = 1, \ldots, K$. This matrix is subject to power constraint (7) for all workers $k = 1, \ldots, K$ as

$$\frac{1}{S}\sum_{s=1}^{S}\operatorname{tr}\left(\mathsf{E}\big[(\mathbf{H}_k\mathbf{H}_k^{\mathsf{T}})^{-1}\big]\mathbf{E}\boldsymbol{\theta}_k^{(s)}(\boldsymbol{\theta}_k^{(s)})^{\mathsf{T}}\mathbf{E}^{\mathsf{T}}\right) \leq P. \quad (15)$$

For the mentioned example of repetition coding, the encoding matrix should be set as $\mathbf{E} = \sqrt{\min_k P_k}\mathbf{1}_l \otimes \mathbf{I}_d$ in order to satisfy the constraint in (15), where the value of P_k is same as for OMA.

The received signal (6) in NOMA carries out aggregation "on air". Applying a single weight matrix $\mathbf{W} \in \mathbb{R}^{d \times m_r}$ yields the global sample

$$\text{(Aggregation NOMA)} \quad \boldsymbol{\theta} = F(\mathbf{y}) = \mathbf{W}\mathbf{y}. \quad (16)$$

The weight matrix $\mathbf{W}$ is again applied separately for each received signal $\mathbf{y}^{(s)}$, and is optimized by using all received signals $\{\mathbf{y}^{(s)}\}_{s=1}^{S}$, as well as, possibly, data set $\mathcal{Z}$. Note that, while in OMA one can potentially apply any aggregation function to the received samples $\{\{\mathbf{y}_k^{(s)}\}_{s=1}^{S}\}_{k=1}^{K}$, over-the-air computing via NOMA limits the aggregation to the linear superposition (6) carried out by the channel.

To unify notation, we will write $\mathbf{W} = \{\mathbf{W}_k\}_{k=1}^{K}$ for the collection of weight matrices for OMA.

V. Wireless Gaussian Consensus Monte Carlo

In this section, we present a first wireless CMC protocol that is inspired by GCMC, and referred to as wireless Gaussian consensus Monte Carlo (WGCMC). Unlike GCMC, WGCMC can be proved to be asymptotically exact with respect to the number of samples, S, under the described wireless channel model when the subposteriors are Gaussian. The limitations of WGCMC will motivate the introduction of a general optimization procedure for the decoding matrices in the next section.

To proceed, as in GCMC (see Sec. III), we assume that each local subposterior $p_k(\boldsymbol{\theta}|\mathcal{Z}_k)$ is a Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{C}_k)$ distributions, so that the global posterior is the Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{C})$ distribution, where $\mathbf{C} = (\sum_{k=1}^{K} \mathbf{C}_k^{-1})^{-1}$. Furthermore, in order to simplify the notation, we set $m_t = m_r = d$, and we choose the trivial encoding matrices $\mathbf{E}_k = \sqrt{P_k}\mathbf{I}_d$ for OMA and $\mathbf{E} = \sqrt{\min_k P_k}\mathbf{I}_d$ for NOMA, where $P_k = PS/\sum_{s=1}^{S}\operatorname{tr}(\mathsf{E}[(\mathbf{H}_k\mathbf{H}_k^{\mathsf{T}})^{-1}]\boldsymbol{\theta}_k^{(s)}(\boldsymbol{\theta}_k^{(s)})^{\mathsf{T}})$. The generalization to any set of encoding matrices follows directly from the discussion below.

A. Wireless GCMC: OMA

Let us consider OMA first. For any weight matrices $\mathbf{W}$, the distribution of the output of the aggregated function (14) is given as

$$\boldsymbol{\theta} = \sum_{k=1}^{K}\mathbf{W}_k\mathbf{y}_k \sim \mathcal{N}\left(\mathbf{0}, \sum_{k=1}^{K}\mathbf{W}_k(P_k\mathbf{C}_k + N_0\mathbf{I})\mathbf{W}_k^{\mathsf{T}}\right). \quad (17)$$

Accordingly, if we know the covariance matrices $\{\mathbf{C}_k\}_{k=1}^{K}$, the power of encoded sample P_k, and noise variance N_0, optimal weight matrices can be obtained by solving the following equality

$$\sum_{k=1}^{K}\mathbf{W}_k(P_k\mathbf{C}_k + N_0\mathbf{I})\mathbf{W}_k^{\mathsf{T}} = \left(\sum_{k=1}^{K}\mathbf{C}_k^{-1}\right)^{-1}. \quad (18)$$

In fact, this condition ensures that the distribution of the global sample $\boldsymbol{\theta} = \sum_{k=1}^{K}\mathbf{W}_k\mathbf{y}_k$ is exactly the desired posterior, i.e., $\boldsymbol{\theta} = \sum_{k=1}^{K}\mathbf{W}_k\mathbf{y}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$. A solution to this system of equation is given by the matrices

$$\mathbf{W}_k = \left(\sum_{k=1}^{K}\mathbf{C}_k^{-1}\right)^{-1}\mathbf{C}_k^{-1/2}(P_k\mathbf{C}_k + N_0\mathbf{I})^{-1/2}. \quad (19)$$

Solution (19) is not computable since the matrices $\mathbf{C}_1, \cdots, \mathbf{C}_K$ are unknown. They can be estimated by using the noisy received samples $\{\mathbf{y}_1^{(s)}, \cdots, \mathbf{y}_K^{(s)}\}_{s=1}^{S}$ as

$$\widehat{\mathbf{C}}_k = \frac{1}{P_k}\left[\frac{1}{(S-1)}\sum_{s=1}^{S}\left(\mathbf{y}_k^{(s)} - \widehat{\boldsymbol{\mu}}_{\mathbf{y}_k}\right)\left(\mathbf{y}_k^{(s)} - \widehat{\boldsymbol{\mu}}_{\mathbf{y}_k}\right)^{\mathsf{T}} - N_0\mathbf{I}\right]^{+}, \quad (20)$$

where $\widehat{\boldsymbol{\mu}}_{\mathbf{y}_k} = \frac{1}{S}\sum_{s=1}^{S}\mathbf{y}_k^{(s)}$. Note that the estimate $\widehat{\mathbf{C}}_k$ in (20) is positive semidefinite thanks to the use of operation $[\,\cdot\,]^{+}$. Plugging (20) into (19) yields WGCMC scheme

(WGCMC for OMA)

$$\boldsymbol{\theta}^{(s)} = \sum_{k=1}^{K}\underbrace{\left(\sum_{k'=1}^{K}\widehat{\mathbf{C}}_{k'}^{-1}\right)^{-1}\widehat{\mathbf{C}}_k^{-1/2}(P_k\widehat{\mathbf{C}}_k + N_0\mathbf{I})^{-1/2}}_{\mathbf{W}_k}\mathbf{y}_k^{(s)} \quad (21)$$

The estimated covariance $\widehat{\mathbf{C}}_k$ in (20) approaches the true covariance $\mathbf{C}_k$ as $S \to \infty$. Under this condition, the distribution of the global sample $\boldsymbol{\theta}$ in (17) equals the desired global posterior. In contrast, the conventional GCMC scheme when applied directly to the received signals, i.e., by replacing scaled received signals $\{\mathbf{y}_k^{(s)}/\sqrt{P_k}\}_{s=1}^{S}$ for $\{\boldsymbol{\theta}_k^{(s)}\}_{s=1}^{S}$ in (10), is not asymptotically optimal as $S \to \infty$ since it disregards the presence of noise on the channel.

B. Wireless Gaussian CMC: NOMA

To develop an asymptotically optimal solution for NOMA, we further assume that local subposteriors are identical Gaussian $\mathcal{N}(0, \mathbf{C}_0)$ distributions, so that the global posterior is the Gaussian $\mathcal{N}(\mathbf{0}, \frac{1}{K}\mathbf{C}_0)$ distribution. This highly simplified scenario with homogeneous posteriors is considered here to illustrate the potential benefits, but also the possible limitations, of NOMA.

The distribution of the output of the aggregated function (14) is given as

$$\mathbf{W}\mathbf{y} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{W}\big(\min_k P_k K\mathbf{C}_0 + N_0\mathbf{I}\big)\mathbf{W}^{\mathsf{T}}\right). \quad (22)$$

In a manner similar to OMA, the optimal weight matrix can be obtained by solving the following equality

$$\mathbf{W}\left(\min_k P_k K \mathbf{C}_0 + N_0 \mathbf{I}\right)\mathbf{W}^\mathsf{T} = \frac{1}{K}\mathbf{C}_0, \qquad (23)$$

and one of solutions is given as

$$\mathbf{W} = \frac{1}{\sqrt{K}}\mathbf{C}_0^{1/2}\left(K \min_k P_k \mathbf{C}_0 + N_0 \mathbf{I}\right)^{-1/2}. \qquad (24)$$

The unknown matrix $\mathbf{C}_0$ can be estimated by using the noisy received samples $\{\mathbf{y}^{(s)}\}_{s=1}^S$ as

$$\widehat{\mathbf{C}}_0 = \frac{1}{K \min_k P_k}\left[\frac{1}{(S-1)}\sum_{s=1}^S\left(\mathbf{y}^{(s)} - \widehat{\boldsymbol{\mu}}_\mathbf{y}\right)\left(\mathbf{y}^{(s)} - \widehat{\boldsymbol{\mu}}_\mathbf{y}\right)^\mathsf{T} - N_0 \mathbf{I}\right]^+, \qquad (25)$$

where $\widehat{\boldsymbol{\mu}}_\mathbf{y} = \frac{1}{S}\sum_{s=1}^S \mathbf{y}^{(s)}$. Plugging (25) into (23) yields the WGCMC scheme under NOMA, i.e.,

(WGCMC for NOMA)

$$\boldsymbol{\theta}^{(s)} = \underbrace{\frac{1}{\sqrt{K}}\widehat{\mathbf{C}}_0^{1/2}\left(K \min_k P_k \widehat{\mathbf{C}}_0 + N_0 \mathbf{I}\right)^{-1/2}}_{\mathbf{W}}\mathbf{y}^{(s)}. \qquad (26)$$

As for OMA, the sample $\boldsymbol{\theta} = \mathbf{W}\mathbf{y}$ generated by of WGCMC under NOMA in (24) follows the desired global posterior distribution when $S \to \infty$. The potential advantage of NOMA versus OMA is that, with the same number of communication blocks, T, the number of received samples is $S = T$ in NOMA while it equals $S = T/K$ in OMA. The large number of samples is generally advantageous for the downstream applications that use the samples, e.g., for ensemble prediction. Furthermore, a larger S leads to a more accurate estimation of the relevant covariance matrices. On the flip side, optimality was only argued here for a simplified homogeneous setting, and one may expect a degraded performance in heterogeneous settings. A strategy that addresses the case of general subposterior is introduced next.

VI. WIRELESS VARIATIONAL CONSENSUS MONTE CARLO

The WGCMC schemes proposed in the previous subsection can be applied to any class of subposteriors in a manner similar to GCMC. However, apart form the special case of Gaussian subposteriors, this application is expected to be successful only in the regime of large data sets due to the Bernstein-von Mises Theorem. In general, the WGCMC scheme is suboptimal. In this section, we introduce an optimization strategy for the design of the weight matrices $\mathbf{W}$ for OMA and NOMA that is able to adapt to arbitrary subposteriors and channels.

A. Design Principle

Inspired by [25], we adopt a variational approach to formulate the problem of ensuring that the distribution $q_F(\boldsymbol{\theta})$ of the aggregated samples (14) for OMA and (16) for NOMA is close to the true posterior $p(\boldsymbol{\theta}|\mathcal{Z})$. We recall that reference [25] assumed noiseless channels and hence focused on optimizing

the aggregation function (4) instead. This condition can be expressed as the approximate equality

(Mean Field Approximation for OMA)

$$p(\boldsymbol{\theta}|\mathcal{Z}) \approx q_F(\boldsymbol{\theta})$$

$$= \int\left(\prod_{k=1}^K p(\mathbf{y}_k)\right)\delta\left[\boldsymbol{\theta} = F(\mathbf{y}_1, \cdots, \mathbf{y}_K)\right] d\mathbf{y}_1, \cdots d\mathbf{y}_K \quad (27\text{a})$$

$$= \int\left(\prod_{k=1}^K p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)p(\mathbf{n}_k)\right)\delta\left[\boldsymbol{\theta} = F(\mathbf{y}_1, \cdots, \mathbf{y}_K)\right] d\boldsymbol{\theta}_1, d\mathbf{n}_1,$$
$$\cdots, d\boldsymbol{\theta}_K, d\mathbf{n}_K \quad (27\text{b})$$

$$= \mathsf{E}_{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_K, \mathbf{n}_1, \cdots, \mathbf{n}_K}\left[\delta\left[\boldsymbol{\theta} = F(\mathbf{y}_1, \cdots, \mathbf{y}_K)\right]\right], \quad (27\text{c})$$

for OMA, where $p(\mathbf{n}_k)$ is the $\mathcal{N}(0, N_0 \mathbf{I}_{m_r})$ probability density function (pdf) and $\delta[\cdot]$ is the Dirac delta function; and

(Mean Field Approximation for NOMA)

$$p(\boldsymbol{\theta}|\mathcal{Z}) \approx q_F(\boldsymbol{\theta})$$

$$= \int p(\mathbf{y})\delta\left[\boldsymbol{\theta} = F(\mathbf{y})\right] d\mathbf{y} \qquad (28\text{a})$$

$$= \int\left(p(\mathbf{n})\prod_{k=1}^K p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)\right)\delta\left[\boldsymbol{\theta} = F(\mathbf{y})\right] d\mathbf{n}, d\boldsymbol{\theta}_1, \cdots, d\boldsymbol{\theta}_K \qquad (28\text{b})$$

$$= \mathsf{E}_{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_K, \mathbf{n}}\left[\delta\left[\boldsymbol{\theta} = F(\mathbf{y})\right]\right], \qquad (28\text{c})$$

for NOMA, where $p(\mathbf{n})$ is the $\mathcal{N}(0, N_0 \mathbf{I}_{m_r})$ pdf. The distribution $q_F(\boldsymbol{\theta})$ in (27b) and (28b) of the obtained global sample is computed by averaging over the distribution of the local samples $(\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_K)$ and of the channel noise. Note that, for OMA, as per (27a), the received signal $\{\mathbf{y}_k\}_{k=1}^K$ are independent due to the independence of local samples and noise samples.

In order to enforce the approximate equalities (27) through the optimization of the weight matrices $\mathbf{W}$, we propose to tackle the problem of minimizing the distance of the distributions $q_F(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta}|\mathcal{Z})$ as measured by Kullback-Leibler (KL) divergence

$$\mathrm{KL}\left(q_F(\boldsymbol{\theta})\|p(\boldsymbol{\theta}|\mathcal{Z})\right) = \mathsf{E}_{q_F(\boldsymbol{\theta})}\left[\log\left(\frac{q_F(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathcal{Z})}\right)\right].$$

This is the standard objective function used in variational inference (see, e.g., [22], [25]). This problem is equivalent to the minimization of the variational free energy [45]

$$\min_\mathbf{W}\left\{\mathcal{L}(\mathbf{W}) = -\mathsf{E}_{\boldsymbol{\theta} \sim q_F(\boldsymbol{\theta})}\left[\log p(\boldsymbol{\theta}, \mathcal{Z})\right] - \mathsf{H}\left[q_F(\boldsymbol{\theta})\right]\right\}, \qquad (29)$$

which is also known as negative evidence lower bound (ELBO) [45]. Note that the variational free energy (29) does not depend on the unknown posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ but on the joint $p(\boldsymbol{\theta}, \mathcal{Z}) = p(\boldsymbol{\theta})p(\mathcal{Z}|\boldsymbol{\theta}) = p(\boldsymbol{\theta})\prod_{k=1}^K p(\mathcal{Z}_k|\boldsymbol{\theta})$.

In the following, we will develop algorithms that tackle problem (29) for OMA by using the received signals $\{\{\mathbf{y}_k^{(s)}\}_{k=1}^K\}_{s=1}^S$ and for NOMA by using the received signals $\{\mathbf{y}^{(s)}\}_{s=1}^S$. In both cases, the server can also use the data set $\mathcal{Z}$. The proposed approaches are based on stochastic gradient

descent (SGD) applied at the server. Specifically, the server applies SGD steps based on the gradient $\nabla_{\boldsymbol{\theta}} \log p(\mathbf{z}_n|\boldsymbol{\theta})$ for source data point $\mathbf{z}_n \in \mathcal{Z}$, which can be readily obtained at the server based on the global data set $\mathcal{Z}$. In fact, computing gradients $\nabla_{\boldsymbol{\theta}} \log p(\mathbf{z}_n|\boldsymbol{\theta})$ is much less computationally intensive than computing the posterior $p(\boldsymbol{\theta}|\mathcal{Z})$: The latter requires integrating over the space of the model parameters, while the former can be carried out via efficient algorithms such as backpropagation. For comparison, in Sec VII, we will consider the performance of stochastic gradient Langevin dynamics (SGLD), an MCMC scheme that also requires the computation of the same stochastic gradients.

B. Wireless Variational CMC: OMA

1) Approximation of the Objective Function: For OMA, the global sample $\boldsymbol{\theta} \sim q_F(\boldsymbol{\theta})$ is given as a function of the received signals in (14). Therefore, the objective in (29) can be rewritten as the expectation over the received signals $\{\mathbf{y}_k\}_{k=1}^K$ as

$$\mathcal{L}(\mathbf{W}) = -\mathsf{E}_{\mathbf{y}_1,\cdots,\mathbf{y}_K}\left[\log p\Big(\sum_{k=1}^K \mathbf{W}_k\mathbf{y}_k, \mathcal{Z}\Big)\right] - \mathsf{H}\big[q_F(\boldsymbol{\theta})\big]. \tag{30}$$

This step implements the "reparameterization trick" [22], moving the optimization variables $\mathbf{W}$ from the averaging distribution to the argument of the expectation. The expectation can now be estimated by averaging over the samples $\{\{\mathbf{y}_k^{(s)}\}_{k=1}^K\}_{s=1}^S$ as

$$\widehat{\mathcal{L}}(\mathbf{W}) = -\frac{1}{S}\sum_{s=1}^S\left[\log p\Big(\sum_{k=1}^K \mathbf{W}_k\mathbf{y}_k^{(s)}, \mathcal{Z}\Big)\right] - \mathsf{H}\big[q_F(\boldsymbol{\theta})\big]. \tag{31}$$

The direct minimization of (31) is still intractable due to the presence of differential entropy of the distribution $q_F(\boldsymbol{\theta})$. This issue is addressed via a tractable lower bound, as derived in the following lemma.

Lemma 1 (Lower Bound on Differential Entropy for OMA). With given encoding matrices $\{\mathbf{E}_k\}_{k=1}^K$, the differential entropy $\mathsf{H}[q_F(\boldsymbol{\theta})]$ of the distribution $q_F(\boldsymbol{\theta})$ of the global sample (14) under OMA is lower bounded as

$$\mathsf{H}[q_F(\boldsymbol{\theta})] \geq \frac{d}{2}\log\Big(2K\sqrt{2\pi e N_0}\Big) + \frac{1}{2K}\sum_{k=1}^K \log|\det(\mathbf{W}_k\mathbf{E}_k)|$$
$$+ \mathsf{H}[p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)] + \frac{1}{2}\log\det\big(\mathbf{W}_k\mathbf{W}_k^{\mathsf{T}}\big)$$
$$:= \widetilde{\mathsf{H}}[q_F(\boldsymbol{\theta})], \tag{32}$$

where $\mathsf{H}[p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)]$ is the differential entropy of the subposterior $p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)$.

Proof: The proof follows the steps of [25, Supplement A Proofs] and is detailed in Appendix A.

Using Lemma 1 in (31) yields the following upper bound on the empirical free energy (31) for OMA

$$\widehat{\mathcal{L}}(\mathbf{W}) \leq \widetilde{\mathcal{L}}(\mathbf{W}) := -\frac{1}{S}\sum_{s=1}^S\left[\log p\Big(\sum_{k=1}^K \mathbf{W}_k\mathbf{y}_k^{(s)}, \mathcal{Z}\Big)\right]$$
$$- \widetilde{\mathsf{H}}\big[q_F(\boldsymbol{\theta})\big]. \tag{33}$$

2) Stochastic Gradient Descent Algorithm: The proposed strategy, referred to as wireless variational consensus Monte Carlo (WVCMC), aims at minimizing the upper bound (33) via SGD. Given a mini-batch $\mathcal{B} \in \mathcal{Z}$ of $N_b \leq N$ samples, stochastic gradient of $\widetilde{L}(\mathbf{W})$ with respect to weight matrix $\mathbf{W}_k$ is given as

$$\widehat{\nabla}_{\mathbf{W}_k}\widetilde{\mathcal{L}}(\mathbf{W})$$
$$= -\frac{1}{S}\sum_{s=1}^S\Big[\nabla_{\boldsymbol{\theta}^{(s)}}\log p(\boldsymbol{\theta}^{(s)}) + \frac{N}{N_b}\sum_{\mathbf{z}_n\in\mathcal{B}}\nabla_{\boldsymbol{\theta}^{(s)}}\log p(\mathbf{z}_n|\boldsymbol{\theta}^{(s)})\Big](\mathbf{y}_k^{(s)})^{\mathsf{T}}$$
$$- \frac{1}{2K}\big[(\mathbf{W}_k\mathbf{E}_k)^{-\mathsf{T}}\mathbf{E}_k^{\mathsf{T}} + (\mathbf{W}_k^{\dagger})^{\mathsf{T}}\big], \tag{34}$$

where we recall that we have the global sample $\boldsymbol{\theta}^{(s)} = \sum_{k=1}^K \mathbf{W}_k\mathbf{y}_k^{(s)}$; and $\mathbf{W}_k^{\dagger}$ is the pseudoinverse of $\mathbf{W}_k$. This yields SGD algorithm based on (34) as summarized in Algorithm 1. Note that, unlike GCMC and WGCMC, WVCMC requires the use of data set $\mathcal{Z}$ at server.

Algorithm 1 WVCMC for OMA

Input: Data set $\mathcal{Z}$, received signals $\{\mathbf{y}_1^{(s)},\cdots,\mathbf{y}_K^{(s)}\}_{s=1}^S$, encoding matrices $\{\mathbf{E}_k\}_{k=1}^K$, step size η, number of iterations t_m.
Initialize: $\mathbf{W}^0$
For each iteration: $t = 1,\ldots,t_m$
    Estimate samples of global posterior using $\mathbf{W}^{t-1}$:
        $\boldsymbol{\theta}^{(s)} = \sum_{k=1}^K \mathbf{W}_k^{t-1}\mathbf{y}_k^{(s)}$ for $s = 1,\ldots,S$
    Update $\mathbf{W}^t$:
        Select a mini-batch of data $\mathcal{B} = \{\mathbf{z}_n\}_{n=1}^{N_b} \in \mathcal{Z}$
        $\mathbf{W}_k^t = \mathbf{W}_k^{t-1} - \eta\widehat{\nabla}_{\mathbf{W}_k}\widetilde{\mathcal{L}}(\mathbf{W}^{t-1})$
        with $\widehat{\nabla}_{\mathbf{W}_k}\widetilde{\mathcal{L}}(\mathbf{W}^{t-1})$ in (34)
Output: Global samples $\boldsymbol{\theta}^{(s)} = \sum_{k=1}^K \mathbf{W}_k^{t_m}\mathbf{y}_k^{(s)}$ for $s = 1,\ldots,S$.

In [25], matrices $\{\mathbf{W}\}_{k=1}^K$ were assumed to be positive semidefinite, resulting in a convex program when the joint distributions $p(\boldsymbol{\theta},\mathcal{Z})$ are log-concave in $\boldsymbol{\theta}$. In the more general setting considered here, the problem is not convex. The iteration complexity of Algorithm 1 to guarantee convergence to stationary points following directly from standard results on SGD, and it depends on the curvature properties of function $\log p(\boldsymbol{\theta},\mathcal{Z})$ [46, Theorem 4.8].

C. Wireless Variational CMC: NOMA

1) Approximation of the Objective Function: In a manner similar to OMA, we first approximate the objective in (29) by averaging over the received signal $\{\mathbf{y}^{(s)}\}_{s=1}^S$. To this end, we recall that the global samples $\boldsymbol{\theta}^{(s)}$ are obtained via (16), and hence we can write the empirical free energy (29) as

$$\widehat{\mathcal{L}}(\mathbf{W}) = -\frac{1}{S}\sum_{s=1}^S\left[\log p\Big(\mathbf{W}\mathbf{y}^{(s)}, \mathcal{Z}\Big)\right] - \mathsf{H}\big[q_F(\boldsymbol{\theta})\big]. \tag{35}$$

Again, the direct optimization of (35) is intractable given the presence of differential entropy $\mathsf{H}[q_F(\boldsymbol{\theta})]$, which is addressed by the tractable lower bound derived below.

Lemma 2 (Lower Bound on Differential Entropy for NOMA). With given encoding matrices $\{\mathbf{E}_k\}_{k=1}^{K}$, the differential entropy $\mathsf{H}[q_F(\boldsymbol{\theta})]$ of the distribution $q_F(\boldsymbol{\theta})$ of the global sample (16) under NOMA is lower bounded as

$$\mathsf{H}[q_F(\boldsymbol{\theta})] \geq \frac{d}{2}\log\left[(K+1)\,(2\pi e N_0)^{\frac{1}{K+1}}\right]$$
$$+ \frac{1}{K+1}\left(K\log|\det(\mathbf{W}\mathbf{E})| + \frac{1}{2}\log\det(\mathbf{W}\mathbf{W}^{\mathsf{T}})\right.$$
$$\left. + \sum_{k=1}^{K}\mathsf{H}[p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)]\right)$$
$$:= \widetilde{\mathsf{H}}[q_F(\boldsymbol{\theta})]. \tag{36}$$

Using Lemma 2 in (35) yields the following upper bound on the empirical free energy for NOMA as

$$\widehat{\mathcal{L}}(\mathbf{W}) \leq \widetilde{\mathcal{L}}(\mathbf{W}) := -\frac{1}{S}\sum_{s=1}^{S}\left[\log p\left(\mathbf{W}\mathbf{y}^{(s)}, \mathcal{Z}\right)\right] - \widetilde{\mathsf{H}}\left[q_F(\boldsymbol{\theta})\right]. \tag{37}$$

2) Stochastic Gradient Descent Algorithm: The proposed WVCMC algorithm for NOMA aims at minimizing the upper bound (37) via SGD. Given a mini-batch $\mathcal{B} \in \mathcal{Z}$ of $N_b \leq N$ samples, stochastic gradient of $\widetilde{L}(\mathbf{W})$ with respect to weight matrix $\mathbf{W}$ is given as

$$\widehat{\nabla}_{\mathbf{W}}\widetilde{\mathcal{L}}(\mathbf{W})$$
$$= -\frac{1}{S}\sum_{s=1}^{S}\left[\nabla_{\boldsymbol{\theta}^{(s)}}\log p(\boldsymbol{\theta}^{(s)}) + \frac{N}{N_b}\sum_{\mathbf{z}_n\in\mathcal{B}}\nabla_{\boldsymbol{\theta}^{(s)}}\log p(\mathbf{z}_n|\boldsymbol{\theta}^{(s)})\right](\mathbf{y}^{(s)})^{\mathsf{T}}$$
$$- \frac{1}{K+1}\left[K(\mathbf{W}\mathbf{E})^{-\mathsf{T}}\mathbf{E}^{\mathsf{T}} + (\mathbf{W}^{\dagger})^{\mathsf{T}}\right], \tag{38}$$

where we have $\boldsymbol{\theta}^{(s)} = \mathbf{W}\mathbf{y}^{(s)}$. The resulting algorithm is summarized in Algorithm 2.

Algorithm 2 WVCMC for NOMA

Input: Data set $\mathcal{Z}$, received signals $\{\mathbf{y}^{(s)}\}_{s=1}^{S}$, encoding matrices $\mathbf{E}$, step size η, number of iterations t_m.
Initialize: $\mathbf{W}^0$
For each iteration: $t = 1, \cdots, t_m$
    Estimate samples of global posterior using $\mathbf{W}^{t-1}$:
      $\boldsymbol{\theta}^{(s)} = \mathbf{W}^{t-1}\mathbf{y}^{(s)}$ for $s = 1, \ldots, S$
    Update $\mathbf{W}^t$:
      Select a mini-batch of data $\mathcal{B} = \{\mathbf{z}_n\}_{n=1}^{N_b} \in \mathcal{Z}$
      $\mathbf{W}^t = \mathbf{W}^{t-1} - \eta\widehat{\nabla}_{\mathbf{W}}\widetilde{\mathcal{L}}(\mathbf{W}^{t-1})$
      with $\widehat{\nabla}_{\mathbf{W}}\widetilde{\mathcal{L}}(\mathbf{W}^{t-1})$ in (38)
Output: Global samples $\boldsymbol{\theta}^{(s)} = \mathbf{W}^{t_m}\mathbf{y}^{(s)}$, for $s = 1, \ldots, S$.

VII. EXPERIMENTS

In this section, we evaluate the performance of the proposed channel-driven CMC schemes for two standard problems: 1) CMC with Gaussian subposteriors [24]; and 2) CMC for Bayesian probit regression on synthetic data set and MNIST data set. To this end, we first introduce the performance metrics and reference schemes that are adopted throughout this section.

1) Performance Metrics: As in prior works on CMC [25], the performance of a CMC algorithm is evaluated by measuring the relative difference between two sample averages of a test function $f(\boldsymbol{\theta})$, namely the reference average obtained under the true global posterior (1) and the average computed using samples produced by the CMC algorithm. Given a test function $f(\cdot)$, denoting $\boldsymbol{\theta}_g^{(s)} \sim p(\boldsymbol{\theta}|\mathcal{Z})$ for $s = 1, \ldots, S_g$ as a set of samples from the distribution of global posterior (1), the test error is defined as

$$\mathsf{err}(f(\cdot)) = \frac{\left|\frac{1}{S}\sum_{s=1}^{S} f(\boldsymbol{\theta}^{(s)}) - \frac{1}{S_g}\sum_{s=1}^{S_g} f(\boldsymbol{\theta}_g^{(s)})\right|}{\left|\frac{1}{S_g}\sum_{s=1}^{S_g} f(\boldsymbol{\theta}_g^{(s)})\right|}, \tag{39}$$

where $\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}$ are the samples produced by the CMC algorithm under evaluation. Specifically, as in [25], we consider multiple test functions, with each function given by one entry of the outer product matrix $\boldsymbol{\theta}\boldsymbol{\theta}^{\mathsf{T}}$. Accordingly, we have the test functions $f_{ij}(\boldsymbol{\theta}) = \boldsymbol{\theta}[i]\boldsymbol{\theta}[j]$ for $i, j = 1, \ldots, d$, where $\boldsymbol{\theta}[i]$ is the i-th entry in vector $\boldsymbol{\theta}$. We then average the per-test function error (39) to obtain

$$\mathsf{err} = \frac{1}{d^2}\sum_{i=1}^{d}\sum_{j=1}^{d}\frac{\left|\frac{1}{S}\sum_{s=1}^{S} f_{ij}(\boldsymbol{\theta}^{(s)}) - \frac{1}{S_g}\sum_{s=1}^{S_g} f_{ij}(\boldsymbol{\theta}_g^{(s)})\right|}{\left|\frac{1}{S_g}\sum_{s=1}^{S_g} f_{ij}(\boldsymbol{\theta}_g^{(s)})\right|}. \tag{40}$$

Accordingly, the error (40) quantifies the accuracy of the empirical correlation matrix of the posterior distribution that can be obtained from the samples $\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}$. Similar results are obtained for the estimate of other moments of the posterior such as the mean vector.

We will also consider the inference performance of the ensemble predictive distribution

$$\widehat{p}_{\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}}(v|\mathbf{u}, \mathcal{Z}) = \frac{1}{S}\sum_{s=1}^{S} p(v|\boldsymbol{\theta}^{(s)}, \mathbf{u}, \mathcal{Z}) \tag{41}$$

obtained at the server given the samples $\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}$ for any test input $\mathbf{u}$. Specifically, the quality of predictive distribution (41) is measured by the Kullback-Leibler (KL) divergence between (41) and the ideal ensemble distribution $\widehat{p}_{\{\boldsymbol{\theta}_g^{(s)}\}_{s=1}^{S}}(v|\mathbf{u}, \mathcal{Z})$ computed by using the samples of true posterior distribution, i.e.,

$$\frac{1}{N_t}\sum_{i=1}^{N_t}\mathsf{KL}\Big(\widehat{p}_{\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}}(v|\mathbf{u}_i, \mathcal{Z})\Big\|\widehat{p}_{\{\boldsymbol{\theta}_g^{(s)}\}_{s=1}^{S}}(v|\mathbf{u}_i, \mathcal{Z})\Big), \tag{42}$$

where $\{\mathbf{u}_i\}_{i=1}^{N_t}$ is a given test set.

2) Reference Schemes: We evaluate the performance of: (*i*) GCMC in (10) with decoded received signals $\{\mathbf{E}^{\dagger}\mathbf{y}_k^{(s)}\}_{s=1}^{S}$ in lieu of $\{\boldsymbol{\theta}_k^{(s)}\}_{s=1}^{S}$; (*ii*) WGCMC in (21) for OMA and in (26) for NOMA; and (*iii*) WVCMC in Algorithm 1 and Algorithm 2. Furthermore, in order to evaluate the advantages of leveraging the K workers in the wireless data center, we also implement a state-of-the-art scheme for MC sampling at the server, namely SGLD [35]. SGLD is selected because it can be run with the same order of computational complexity at the server of the proposed WVCMC schemes. In fact, as detailed next, SGLD relies on the computation of stochastic gradients $\nabla\log p(\boldsymbol{\theta}, \mathbf{z}_n)$ for randomly selected examples

$\mathbf{z}_n \in \mathcal{Z}$. Therefore, we can compare SGLD and WVCMC under the assumption of (approximately) equal computation load at the server.

At each iteration t, SGLD updates the model parameter vector by using a subset of data in the global data set $\mathcal{Z}$ as

$$\text{(SGLD)} \quad \boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + \frac{\eta^{(t)}}{2}\left[\frac{N}{N_b}\sum_{\mathbf{z}_n \in \mathcal{B}} \nabla_{\boldsymbol{\theta}^{(t)}} \log p(\mathbf{z}_n|\boldsymbol{\theta}^{(t)})\right.$$
$$\left. + \nabla_{\boldsymbol{\theta}^{(t)}} \log p(\boldsymbol{\theta}^{(t)})\right] + \mathbf{n}^{(t)}, \quad (43)$$

where the learning rate is set as $\eta^{(t)} = \alpha(\beta + t)^{-\gamma}$ for some $\beta > 0$ and $\gamma \in (0.5, 1]$, and $\mathbf{n}^{(t)} \sim \mathcal{N}(0, \eta^{(t)}\mathbf{I})$ is injected noise. SGLD is known to produce, in the limit of a large number of iterations, samples from the desired posterior. After removing the first t_b samples, which we consider to constitute the burn-in phase, the remaining $\{\boldsymbol{\theta}^{(t)}\}_{t=t_b+1}^{t_m}$ samples are used for evaluation in (40). We emphasize that SGLD does not make use of the presence of the K workers. Distributed versions of SGLD also exist. These can be useful as alternatives to the CMC schemes studied in this paper when the workers do not have sufficient computing power to obtain samples from the local subposteriors – a condition that we have assumed. We refer to Sec. VIII for more discussion on this point. As for the computation at the server, consider C computed gradients $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \mathbf{z}_n)$, SGLD requires C/N_b iterations while WVCMC requires $C/N_b S$ iterations since WVCMC process S samples $\{\boldsymbol{\theta}^{(s)}\}_{s=1}^{S}$ in parallel.

A. Gaussian Subposteriors

We start with the baseline example studied in Sec. III and Sec. V, in which the local subposterior for each worker k follows a Gaussian distribution $\mathcal{N}(0, \mathbf{C}_k)$. The model dimension is $d = 5$, and each covariance matrix $\mathbf{C}_k$ is a symmetric Toeplitz matrix with first column given as $[1, \rho_k, \rho_k^2, \rho_k^3, \rho_k^4]$ for some $\rho_k \in [-1, 1]$. We consider $K = 10$ workers, and set $\rho_k = (k-1)/K$ for k-th worker. For reference, we also study a homogeneous setting in which the covariances are equal to $\mathbf{C}_0 = K(\sum_{k=1}^{K} \mathbf{C}_k^{-1})^{-1}$ for all workers.

To start, we consider frequency-flat channels with $\mathbf{H}_k = \mathbf{I}_d$, and the SNR $= P/(dN_0)$ is set as 5 dB by default. Unless stated otherwise, the number of communication blocks is equal to $T = 2000$; the number of per-output-sample iterations is $t_m = 30$ for NOMA and $t_m = 300$ for OMA, which ensures that the same total number of iterations, namely 300, are carried out for both OMA and NOMA; the learning rate is $\eta = 10^{-3}$ for NOMA and $\eta = 5 \times 10^{-3}$ for OMA, which have been separately optimized through a grid search; and the initial point used for WVCMC is $\mathbf{W}^0 = 1/K\mathbf{I}_d$ for NOMA, while in OMA it is set as GCMC solution. This initialization given by the GCMC solutions was found to be advantageous for OMA, but it is not applicable to NOMA. Note that, in this simple example, the global posterior covariance matrix is $\mathbf{C} = (\sum_{k=1}^{K} \mathbf{C}_k^{-1})^{-1}$, and hence we use this ground-truth matrix in (40) in lieu of the estimate $\frac{1}{S_g}\sum_{s=1}^{S_g} \boldsymbol{\theta}_g^{(s)}(\boldsymbol{\theta}_g^{(s)})^\mathsf{T}$. We use full mini-batches, i.e., we implement gradient descent, such that the gradients $\nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}, \mathcal{Z})$ in (34) and (38) are computed as $\nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}, \mathcal{Z}) = \nabla_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|\mathcal{Z})$. All results are averaged over 100 experiments.
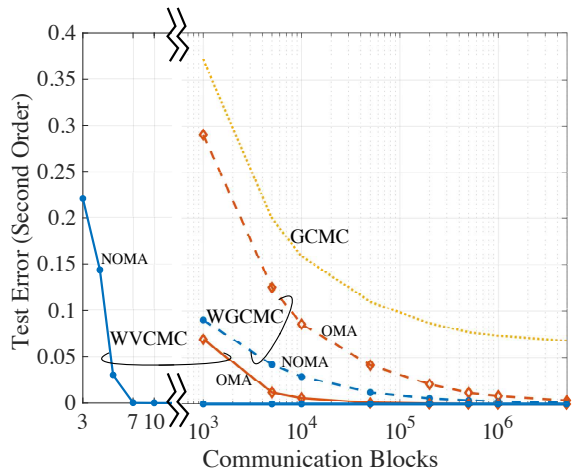

Figure 2. Second-order test error (40) versus the number of communication block, T, for wireless CMC (SNR = 5 dB, homogeneous subposteriors).
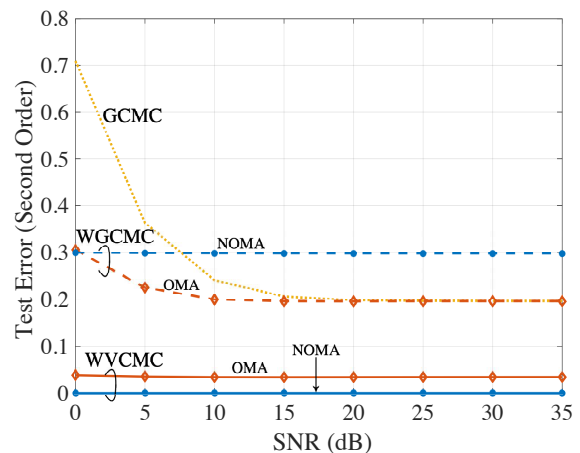

Figure 3. Second-order test error (40) versus SNR for wireless CMC ($T = 2000$, heterogeneous subposteriors).

1) Error vs number of communication blocks: In Fig. 2, we consider homogeneous subposteriors, and plot the test error (40) versus the number of communication blocks. We recall that the number of samples is $S = T/K$ for OMA and $S = T$ for NOMA. WVCMC with NOMA is seen to be the most communication-efficient approach, as its test error vanishes for a number of blocks $T \simeq 7$, while the other schemes require a large number of communication blocks. Validating the analysis in Sec. V, WGCMC, under both OMA and NOMA, is observed to attain optimality as the number of communication blocks, and hence of samples, increases. We also note that, as mentioned in Sec. V, the conventional GCMC scheme is not asymptotically optimal with a large number of communication blocks.

2) Error vs SNR: We now study the impact of SNR in Fig. 3 under the assumption of heterogeneous subposteriors, i.e., subposteriors with different covariance matrices $\{\mathbf{C}_k\}_{k=1}^{K}$ as described above. WVCMC is seen to successfully adapt to different SNR levels, outperforming all the other schemes. In fact, the error of WVCMC is essentially unaffected by the SNR level in the considered range. This observation points to

a key property of the proposed channel-driven MC sampling: If properly accounted for, the randomness injected by the channel noise may not affect the performance of CMC algorithms. Channel noise can be considered as a component of the sample generation process, rather than a nuisance to be mitigated. Importantly, the results in Fig. 3 suggest that it is critical to properly optimize the aggregation function in order to leverage this unique property of MC sampling over noisy channels. We also note that the noise-invariant error of WVCMC is attained in the ideal case studied in this example, for which the subposterior and channel noise follow Gaussian distributions. As two further comments on Fig. 3, WGCMC under OMA is confirmed to be advantageous over conventional GCMC, with the performance of two schemes converging at high SNR. Moreover, WGCMC under NOMA is seen to be worse than that of OMA, due to the performance loss counted by the heterogeneity of the subposteriors (see Sec. V-B for further discussion). Note that this is unlike the proposed WVCMC NOMA scheme, which remains advantageous over OMA.

B. Bayesian Probit Regression

We turn to focus on the more complex problem of Bayesian probit regression. Accordingly, given observations $\{\mathbf{z}_n = (\mathbf{u}_n, v_n)\}_{n=1}^N$, the likelihood is given by the following binary regression model

$$p(v_n = 1 | \boldsymbol{\theta}, \mathbf{u}_n) = \Phi(\boldsymbol{\theta}^\mathsf{T} \mathbf{u}_n), \qquad (44)$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. Furthermore, the prior is the Gaussian distribution $\mathcal{N}(0, \sigma^2 \mathbf{I})$ for a fixed variance σ^2.

We first consider a synthetic data set following the model described in (44). This is comprised of $N = 8500$ labeled data points $\{\mathbf{z}_n = (\mathbf{u}_n, v_n)\}_{n=1}^N$. The covariates $\mathbf{u}_n \in \mathbb{R}^d$ are drawn i.i.d. from Gaussian distribution $\mathcal{N}(0, \mathbf{I}_d)$ where $d = 5$, and the binary label v_n is drawn as per (44) under the ground-truth model parameter $\boldsymbol{\theta}^* = [0.1103, -0.5832, 0.6417, 1.8279, 0.4968]^\mathsf{T}$. Following [25], we also consider a higher-dimensional data set comprised by $N = 15000$ data points with dimension $d = 300$ generated in the same manner.

The channels $\mathbf{H}_k$ are $2d \times (2d+2)$ random matrices with i.i.d standard Gaussian entries, and thus we have $\mathsf{E}\left[(\mathbf{H}_k \mathbf{H}_k^\mathsf{T})\right]^{-1} = \mathbf{I}_{2d}$. We implement analog repetition coding, i.e., we set the encoding matrices as $\mathbf{E}_k = \sqrt{P_k}[\mathbf{I}_d, \mathbf{I}_d]^\mathsf{T}$ for OMA and $\mathbf{E} = \min_k \sqrt{P_k}[\mathbf{I}_d, \mathbf{I}_d]^\mathsf{T}$ for NOMA, where $P_k = PS / \sum_{s=1}^S 2\|\boldsymbol{\theta}_k^{(s)}\|^2$. Unless stated otherwise, the data set is divided equally among $K = 20$ workers; the number of communication blocks is set to $T = 1000$; the number of per-output sample iterations to $t_m = 50$ for NOMA and OMA; the learning rate to $\eta = 10^{-7}$ for NOMA and $\eta = 10^{-6}$ for OMA upon separate optimizations; and the initial point used for WVCMC is $\mathbf{W}^0 = 1/K \mathsf{E}_k^\dagger$ for NOMA while in OMA it is set as GCMC solution as in the previous example. We use full mini-batches, i.e., $N_b = N$ in (34) and (38), and details about gradient computation and about the implementation of Gibbs sampling from (sub)posteriors are given in Appendix B. To show the effect of the partitions of local data set and
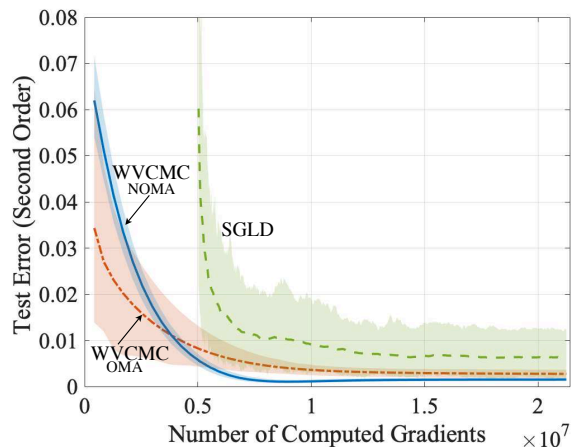

Figure 4. Second-order test error (40) versus the number of computed gradients $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \mathbf{z}_n)$ for wireless VCMC (SNR $= 15$ dB, $S = 50$) and centralized SGLD.

the necessity of consensus, we also include the performance of the best single worker in OMA for reference. In this case, the server only uses the samples from the worker that yields the smallest test error. Unless stated otherwise, the results are averaged over 40 experiments.

1) Computational Efficiency at the Server: In this experiment, we compare different solutions as a function of the total number of computed gradients at the server. As discussed, we are specifically interested in the comparison with SGLD. In Fig. 4, we plot the test error (40) versus the number of computed gradients $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \mathbf{z}_n)$, with the shaded area representing 90% high-probability interval to illustrate both mean and variance during the whole training process. We recall that the number of computed gradients at the server is NSt for WVCMC and $N_b t$ for SGLD where t is the iteration round. Note that we highlight high-probability intervals here and not in other figures since the other plots only show the performance at convergence.

For WVCMC, we use $S = 50$ samples, corresponding to the number of communication blocks $T = 1000$ for OMA and $T = 50$ for NOMA, and the SNR $= P/(2dN_0)$ is set as 15 dB. For SGLD, the initialization is randomized using a sample from the prior; mini-batch size in (43) is $N_b = 500$; the learning rate is $\eta^{(t)} = \alpha(\beta + t)^{-\gamma}$, with $\alpha = 0.01$, $\beta = 1$, $\gamma = 0.7$; and number of burn-in samples $t_b = 10^4$. All results are averaged over 100 experiments.

The most important observation is that, by leveraging the computing power of K parallel workers, WVCMC can vastly outperform SGLD for the same number of computed gradients at the server. WVCMC with NOMA is seen to achieve the best accuracy when the number of computed gradients are larger than 5×10^6, and such a performance is attainable with least variance among all the schemes. The performance of WVCMC with OMA has a larger variance which is caused by the additive channel noise accumulated across multiple communication blocks due to the more inefficient orthogonal access.

2) Error vs SNR: Fig. 5 plots the test error in (40) versus the SNR. WVCMC is again confirmed to outperform all the
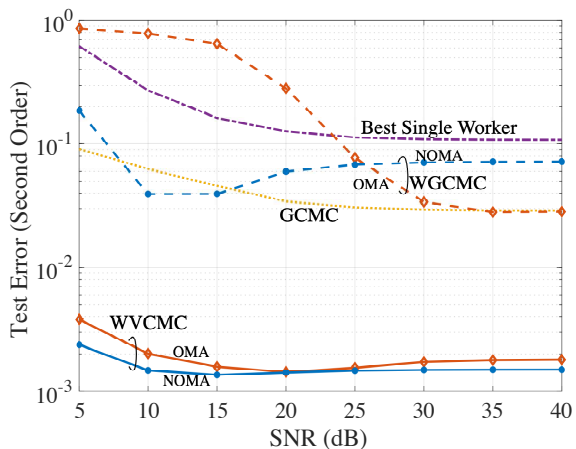
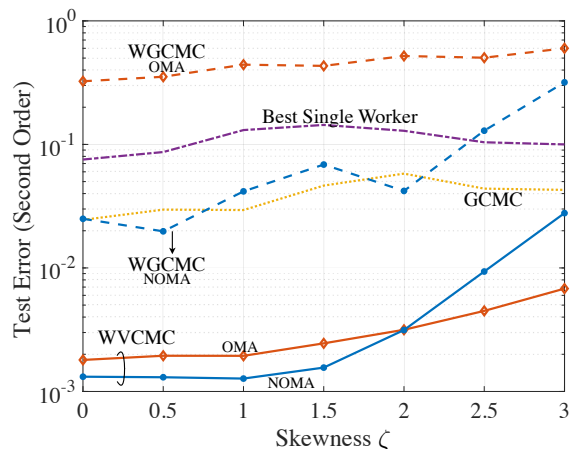Figure 5. Second-order test error (40) versus SNR for wireless CMC schemes ($T = 1000$, $K = 20$).

Figure 7. Second-order test error (40) versus the skewness of data heterogeneity for wireless CMC ($T = 1000$, SNR $= 20$ dB).
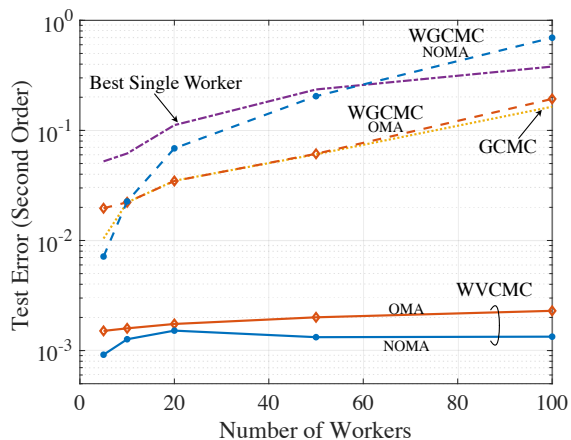

Figure 6. Second-order test error (40) versus the number of workers for wireless CMC ($T = 1000$, SNR $= 35$ dB).
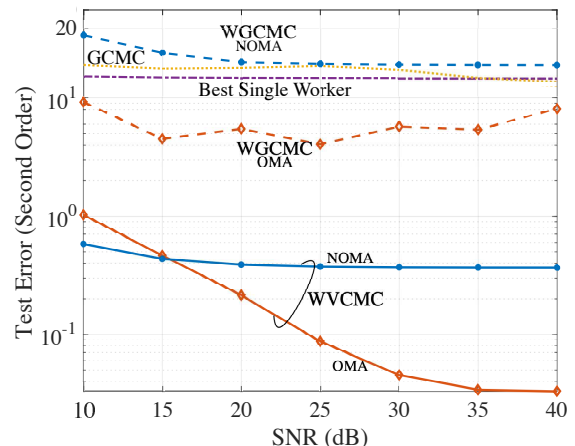

Figure 8. Second-order test error (40) versus SNR for wireless CMC schemes on high-dimensional data ($T = 10000$, $K = 20$).

other schemes. However, unlike the Gaussian example, for which the performance of WVCMC is unaffected by the SNR level, for the non-Gaussian subposteriors considered here, the performance degrades slightly in the low-SNR regime. Unlike WVCMC, the performance of WGCMC schemes is significantly decreased by the non-Gaussianity of the subposteriors.

3) Error vs Number of workers: We now study the impact of the number of workers K for SNR $= 35$ dB. The learning rate is set to adapt to the number of workers as $\eta = 2 \times 10^{-5}/K$ for OMA and $\eta = 2 \times 10^{-6}/K$ for NOMA. Fig. 6 shows that increasing the number of workers has a negative effect on all the schemes. This degradation is, in part, caused by the reduction in the size of local data sets, and by the resulting need to combine more local samples [24], [25]. For OMA, increasing K has the addition drawback of reducing the number of received samples for, which is seen to further degrade its performance for $K \geq 20$.

4) Data Heterogeneity: Fig. 7 plots the error versus a measure of the heterogeneity of the local data sets. To this end, the global data set is partitioned as follows. The k-th worker contains a fraction $\frac{1/k^\zeta}{\sum_{k=1}^{K} 1/k^\zeta}$ of data points from class

'1' and a fraction $\frac{1/(K-k+1)^\zeta}{\sum_{k=1}^{K} 1/k^\zeta}$ of data points from class '0'. The heterogeneity is described by the exponent ζ, with $\zeta = 0$ representing homogeneous local data sets. As shown in Fig. 7, increasing the heterogeneity of local data sets generally affects negatively to all schemes. Nevertheless, the heterogeneity of local data sets has a stronger impact for NOMA than for OMA, since, as discussed in Sec. V-B, NOMA ideally requires homogeneous subposteriors to achieve optimality.

5) Higher-Dimensional Data: We now consider the problem of Bayesian probit regression on higher-dimensional data, matching the scale of the experiments in [25]. The number of iterations is selected as a function of the SNR level to guarantee the convergence, and the learning rate is set to $\eta = 5 \times 10^{-9}$ for NOMA and $\eta = 10^{-7}$ for OMA. All the other settings are same as in the rest of this subsection, and the results are averaged over three realizations. Fig. 8 plots the test error in (40) versus the SNR. Confirming the results in Fig. 5, WVCMC is seen to significantly outperform all the other schemes. Furthermore, as compared to the case of lower-dimensional data, OMA is seen to be generally advantageous with respect to NOMA, suggesting that the restriction to equal
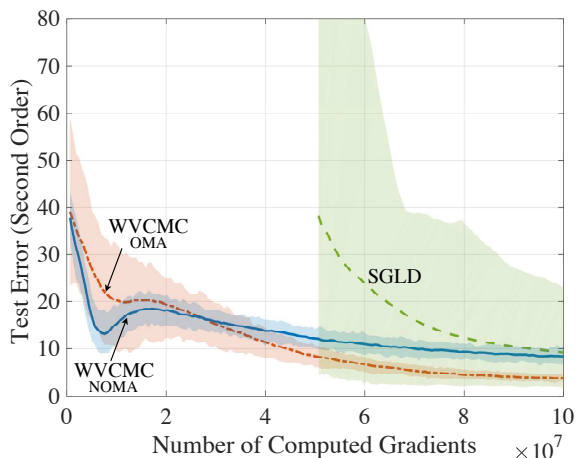
Figure 9. Second-order test error (40) versus the number of computed gradients $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \mathbf{z}_n)$ for wireless VCMC (SNR = 30 dB, $S = 50$) and centralized SGLD on MNIST dataset.

weights matrices in NOMA becomes more problematic as the problem dimension increases.

C. MNIST Data Set

We now consider the problem of Bayesian probit regression on the MNIST data set by using $N = 12665$ data points from the classes of 0 and 1. The original data with dimension $d = 784$ is projected to the lower dimension $d = 30$ via principal component analysis (PCA). We note that the application of Bayesian learning to lower-dimensional data embeddings is a common approach to enhance scalability [47]–[50]. The prior distribution, channel statistics, and initial points used for learning are same as in Sec. VII-B. We use the mini-batch size $N_b = N/20$ in (34) and (38). Unless stated otherwise, the data set is divided equally among $K = 10$ workers; the number of communication blocks is set to $T = 500$; the number of per-output sample iterations to $t_m = 250$ for NOMA and OMA; and the learning rate to $\eta = 2 \times 10^{-7}$ for NOMA and $\eta = 10^{-6}$ for OMA upon separate optimizations.

1) Computational Efficiency at the Server: We start by validating the experiment in Fig. 4 on this data set. To this end, in this experiment, we use $S = 50$ samples for WVCMC, corresponding to the number of communication blocks $T = 500$ for OMA and $T = 50$ for NOMA, and the SNR = 30 dB. For SGLD, mini-batch size in (43) is $N_b = N/20$; the learning rate is $\eta^{(t)} = \alpha(\beta + t)^{-\gamma}$, with $\alpha = 0.01$, $\beta = 1$, $\gamma = 0.52$; and number of burn-in samples $t_b = 8 \times 10^4$. All results are averaged over 100 experiments. In Fig. 9, we plot the test error (40) versus the number of computed gradients $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}, \mathbf{z}_n)$, with the shaded area representing 90% high-probability interval. Confirming the results in Fig. 4, WVCMC can vastly outperform SGLD for the same number of computed gradients at the server. However, unlike in the experiment on synthetic data set, for MNIST, WVCMC with OMA is seen to achieve the best accuracy in terms of mean and variance performance. This is due to the heterogeneity of the subposteriors entailed by the use of real data, which degrades the performance of NOMA.
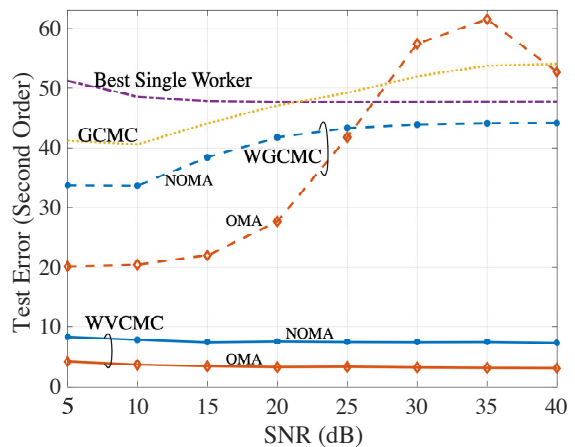

Figure 10. Second-order test error (40) versus SNR for wireless CMC schemes on MNIST dataset ($T = 500$, $K = 10$).
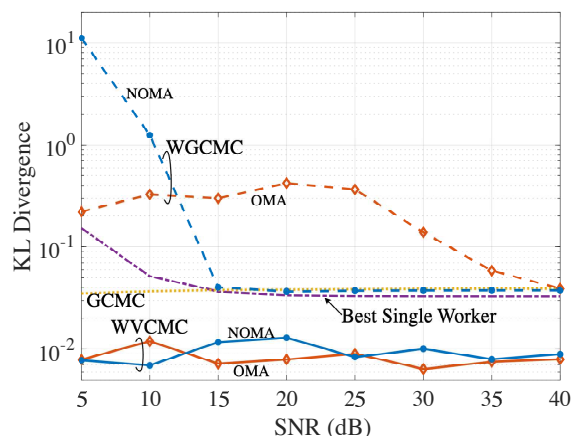

Figure 11. KL divergence (42) measuring the accuracy of the ensemble prediction (41) versus SNR for wireless CMC schemes on MNIST dataset ($T = 500$, $K = 10$).

2) Error vs SNR: Finally, Fig. 10 plots the test error in (40) versus the SNR. WVCMC is confirmed to outperform all the other schemes, and OMA outperforms NOMA due to the heterogeneous subposteriors. Note that, for GCMC and WGCMC, the error is seen to increase in high SNR regime, suggesting that the presence of Gaussian noise in low SNR is actually advantageous to schemes that are designed under the assumption of Gaussian subposteriors. Furthermore, the limited received samples in OMA cause the inaccurate estimation of covariance matrix, and WGCMC with OMA and GCMC is seen to perform worse than the single-worker benchmark.

3) Evaluation of Inference Performance: We now study the inference performance of the ensemble prediction (41) via the KL divergence (42) evaluated on a test set of $N_t = 2115$ PAC-pre-processed MNIST images. The results in Fig. 11 confirm the conclusions shown from Fig. 5 and Fig. 8 in terms of the advantages with respect to all benchmarks.

VIII. CONCLUSIONS

In this paper, we have considered, for the first time, consensus Monte Carlo (CMC) solutions for distributed one-

shot Bayesian learning in wireless data centers. Two main ideas were leveraged by the proposed designs. The first is the novel concept of channel-driven sampling: under uncoded wireless transmission of samples from workers to server, channel noise can contribute to MC sampling, and, if properly accounted for, is not necessarily a nuisance to be mitigated. In particular, we have argued that channel-driven sampling is asymptotically optimal with respect to the number of samples under the assumption that the subposteriors are Gaussian. The second idea we build on is over-the-air computing in order to enhance spectral efficiency. This is in line with previous works on federated (frequentist) learning in wireless systems [8]–[10], [14]. although the goal here is synthesizing samples from the posterior and not computing an average. For arbitrary subposteriors, building on these two ideas, we have proposed a general variational strategy termed wireless variational CMC (WVCMC) that aims at minimizing the KL divergence between the distributions of the output of the aggregation function at the server and of the global posterior. Simulation results show that the proposed approach can vastly outperform existing solutions.

We note that performance of the proposed WCMC schemes depends on the aggregation function. Although only linear mappings are considered in this paper in line with prior work on the topic, the introduced variational framework can be directly applied to other functions. As for future work, one may consider studying channel-driven sampling for digital transmission via lattice coding [51]. It would also be interesting to alleviate the assumption that workers can generate samples from local subposteriors. A possible approach is to carry out approximate MC sampling at workers, e.g., distributed SGLD [32] or distributed SVGD [31]. As another possible extension of the current work, optimal analog encoding and analog compression in [40], [41], could be tailored for wireless CMC strategies so as to further enhance the communication efficiency. One could also enforce a short-term power constraint in each communication block. A possible way to do this involves introducing a power control threshold, in a manner similar to [14], [15]. Accordingly, workers whose channel gain is too small refrain from transmitting. The optimization of the thresholds is an interesting open problem.

APPENDIX

A. Proof of Lemma 1

According to aggregated function in (14) and received signal in (5), we have

$$\mathsf{H}[q_F(\boldsymbol{\theta})]$$

$$= \mathsf{H}\Big[\sum_{k=1}^{K} \mathbf{W}_k \mathbf{y}_k\Big]$$

$$= \mathsf{H}\Big[\sum_{k=1}^{K} \mathbf{W}_k (\mathbf{E}_k \boldsymbol{\theta}_k + \mathbf{n}_k)\Big]$$

$$\overset{(a)}{\geq} \frac{d}{2} \log \Big[\sum_{k=1}^{K} e^{\frac{2}{d}\mathsf{H}[\mathbf{W}_k \mathbf{E}_k \boldsymbol{\theta}_k]} + e^{\frac{2}{d}\mathsf{H}[\mathbf{W}_k \mathbf{n}_k]}\Big]$$

$$\overset{(b)}{\geq} \frac{d}{2} \log \Big[2K e^{\frac{1}{2K}\sum_{k=1}^{K} \frac{2}{d}\mathsf{H}[\mathbf{W}_k \mathbf{E}_k \boldsymbol{\theta}_k] + \sum_{k=1}^{K} \frac{2}{d}\mathsf{H}[\mathbf{W}_k \mathbf{n}_k]}\Big]$$

$$= \frac{d}{2} \log(2K) + \frac{1}{2K}\Big(\sum_{k=1}^{K} \mathsf{H}[\mathbf{W}_k \mathbf{E}_k \boldsymbol{\theta}_k] + \mathsf{H}[\mathbf{W}_k \mathbf{n}_k]\Big)$$

$$= \frac{d}{2} \log(2K) + \frac{1}{2K}\Big(\sum_{k=1}^{K} \log|\det(\mathbf{W}_k \mathbf{E}_k)| + \mathsf{H}[p_k(\boldsymbol{\theta}_k|\mathcal{Z}_k)]$$

$$+ \frac{1}{2}\log\det 2\pi e N_o \mathbf{W}_k \mathbf{W}_k^{\mathsf{T}}\Big)$$

where (a) is obtained by using entropy power inequality, and (b) follows from Jensen's inequality for the convex exponential function. Rearranging terms in the result yields Lemma 1.

B. Implementation of Bayesian Probit Regression

1) Gradient Computation: We assume the prior follows Gaussian distribution $\mathcal{N}(0, \sigma^2 \mathbf{I})$, and the log prior is

$$\log p(\boldsymbol{\theta}) = -\frac{1}{2\sigma^2}\|\boldsymbol{\theta}\|^2 \tag{45}$$

According to (44), we have log likelihood as

$$\log p(v_n|\boldsymbol{\theta}, \mathbf{u}_n) = v_n \log \Phi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n) + (1-v_n)\log\big(1 - \Phi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n)\big), \tag{46}$$

where $\Phi(\cdot)$ is cumulative distribution function of the standard normal distribution. Accordingly, their gradients are given as

$$\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}) = -\frac{1}{\sigma^2}\boldsymbol{\theta}, \tag{47}$$

$$\nabla_{\boldsymbol{\theta}} \log p(v_n|\boldsymbol{\theta}, \mathbf{u}_n) = \frac{\varphi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n)\big[v_n - \Phi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n)\big]}{\Phi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n)\big[1 - \Phi(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n)\big]}\mathbf{u}_n \tag{48}$$

where $\varphi(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$.

2) Gibbs Sampling from Posterior: The analytical approach to perform Bayesian inference on probit model (44) is challenging due to the lack of conjugate prior for the parameter $\boldsymbol{\theta}$. An equivalent model can be provided by introducing an auxiliary variable κ_n as

$$\kappa_n = \boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n + \epsilon \tag{49a}$$

$$v_n = \begin{cases} 1, \text{if } \kappa_n > 0 \\ 0, \text{otherwise}, \end{cases} \tag{49b}$$

where $\epsilon \sim \mathcal{N}(0, 1)$. The equivalent model in (49) enables an efficient Gibbs sampling for obtaining the posterior statistics. The Gibbs sampling, as introduced in [52], involves iterative sampling of the following two posterior distributions of $\boldsymbol{\theta}$ and κ_n respectively.

For N observations, given the latent variables $\{\kappa_n\}_{n=1}^{N}$ and the covariates $\{\mathbf{u}_n\}_{n=1}^{N}$, the posterior distribution of $\boldsymbol{\theta}$ is derived as

$$p(\boldsymbol{\theta}|\{\kappa_n\}_{n=1}^{N}, \{\mathbf{u}_n\}_{n=1}^{N}) \propto p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\kappa_n|\boldsymbol{\theta}, \mathbf{u}_n) \tag{50}$$

$$= \mathcal{N}_{\boldsymbol{\theta}}(0, \sigma^2 \mathbf{I}) \prod_{n=1}^{N} \mathcal{N}_{\kappa_n}(\boldsymbol{\theta}^{\mathsf{T}}\mathbf{u}_n, 1) \tag{51}$$

$$\propto \mathcal{N}_{\boldsymbol{\theta}}\Big(\mathbf{C}\sum_{n=1}^{N} \mathbf{u}_n \kappa_n, \mathbf{C}\Big) \tag{52}$$

where $\mathbf{C} = \left(\sum_{n=1}^{N} \mathbf{u}_n \mathbf{u}_n^{\mathsf{T}} + \sigma^{-2} \mathbf{I} \right)^{-1}$.

On the other hand, with the knowledge of $\boldsymbol{\theta}$, $\mathbf{u}_n$ and v_n, the posterior distribution of κ_n is truncated Gaussian distribution

$$p(\kappa_n | \boldsymbol{\theta}, \mathbf{u}_n, v_n) \propto \begin{cases} \mathcal{N}_{\kappa_n}(\boldsymbol{\theta}^{\mathsf{T}} \mathbf{u}_n, 1) \mathcal{I}(\kappa_n > 0, v_n = 1) \\ \mathcal{N}_{\kappa_n}(\boldsymbol{\theta}^{\mathsf{T}} \mathbf{u}_n, 1) \mathcal{I}(\kappa_n \leq 0, v_n = 0) \end{cases}, \tag{53}$$

where $\mathcal{I}(\cdot)$ is the indicator function.

To start the sampling, the initial value of $\boldsymbol{\theta}^{(0)}$ can be set as the maximum likelihood estimate. The samples of (sub)posterior are drawn from (52) while iterating between (52) and (53). The samples of global posterior $\{\boldsymbol{\theta}_g^{(s)}\}_{s=1}^{S_g}$ are obtained by using global data set and prior $\mathcal{N}(0, \sigma^2 \mathbf{I})$, and $S_g = 20000$, while the samples of subposterior $\{\boldsymbol{\theta}_k^{(s)}\}_{s=1}^{S}$ are based on the local data set and prior $\mathcal{N}(0, K\sigma^2 \mathbf{I})$. The first 100 samples are considered to be burn-in phase, which are excluded in the evaluation.

C. On the Complexity of Sampling from the Posterior

The complexity of drawing one or multiple samples from the posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ for some data set $\mathcal{Z}$, prior $p(\boldsymbol{\theta})$ and likelihood $p(\mathcal{Z}|\boldsymbol{\theta})$ depends strongly on the choice of prior and likelihood. At one extreme, if prior and likelihood are conjugate, the per-sample complexity is constant, and it does not depend on the size of the data set $\mathcal{Z}$. At the other extreme, we have general-purpose MCMC algorithms such as rejection sampling, Metropolis-Hastings, or Langevin Monte Carlo that apply to any prior $p(\boldsymbol{\theta})$ and likelihood $p(\mathcal{Z}|\boldsymbol{\theta})$. In between, we have solutions like Gibbs sampling – a special instance of Metropolis-Hastings algorithms – that is only relevant for priors and likelihoods that have suitable factorizations.

The problem of quantifying the complexity of general-purpose MCMC schemes is generally open [53], and recent work has provided bounds on iteration complexity for specific classes of schemes such as SG-MCMC [54], [55]. It is noted that these are approximate MC sampling schemes that produce samples from the posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ only asymptotically as the number of iterations grows. Methods such as SG-MCMC are justified by the fact that the per-iteration complexity scales linearly with the size of the mini-batch.

While, as mentioned, the general problem of relating accuracy, complexity, and number of data points is open, some insight on this issue can be obtained by considering the simple rejection sampling scheme. Rejection sampling produces exact samples from the posterior $p(\boldsymbol{\theta}|\mathcal{Z})$ by first drawing samples from a proposal distribution, and then accepting each candidate sample with some probability. The complexity of rejection sampling can hence be directly measured by the average number of candidate samples needed to produce one sample from $p(\boldsymbol{\theta}|\mathcal{Z})$.

A standard choice for the acceptance probability of a sample $\boldsymbol{\theta}$ is

$$p(\mathrm{acc}|\boldsymbol{\theta}) = \frac{p(\mathcal{Z}|\boldsymbol{\theta})}{p(\mathcal{Z}|\boldsymbol{\theta}^{ML})} = \prod_{n=1}^{N} \frac{p(\mathbf{z}_n|\boldsymbol{\theta})}{p(\mathbf{z}_n|\boldsymbol{\theta}^{ML})},$$

where $\boldsymbol{\theta}^{ML}$ is the maximum likelihood (ML) solution $\boldsymbol{\theta}^{ML} = \arg\max_{\boldsymbol{\theta}} p(\mathcal{Z}|\boldsymbol{\theta})$. By definition of the ML solution, the ratio

$p(\mathbf{z}_n|\boldsymbol{\theta})/p(\mathbf{z}_n|\boldsymbol{\theta}^{ML})$ is generally smaller than one for a non-negligible fraction of data points, showing that the probability of acceptance can potentially decrease exponentially quickly to zero as N increases. In turn, this implies that the complexity of generating a sample can grow as quickly as exponentially with the number of data points N.

REFERENCES

[1] I. Stoica, D. Song, R. A. Popa, D. Patterson, M. W. Mahoney, R. Katz, A. D. Joseph, M. Jordan, J. M. Hellerstein, J. E. Gonzalez, *et al.*, "A Berkeley view of systems challenges for AI," *[Online]. Available: https://arxiv.org/pdf/1712.05855.pdf*, 2017.

[2] M. Li, D. G. Andersen, J. W. Park, A. J. Smola, A. Ahmed, V. Josifovski, J. Long, E. J. Shekita, and B.-Y. Su, "Scaling distributed machine learning with the parameter server," in *Proc. 11th USENIX Symp. Operating Sys. Design and Implementation (OSDI)*, (Broomfield, Colorado), pp. 583–598, Oct. 2014.

[3] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *[Online]. Available: https://arxiv.org/pdf/1603.04467.pdf*, 2016.

[4] S. Dhar, J. Guo, J. Liu, S. Tripathi, U. Kurup, and M. Shah, "On-device machine learning: An algorithms and learning theory perspective," *[Online]. Available: https://arxiv.org/pdf/1911.00623.pdf*, 2019.

[5] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Commun. Surv. Tutor.*, vol. 21, pp. 2224–2287, March 2019.

[6] A. Celik, B. Shihada, and M.-S. Alouini, "Wireless data center networks: Advances, challenges, and opportunities," *[Online]. Available: https://arxiv.org/pdf/1811.11717.pdf*, 2018.

[7] T. Ben-Nun and T. Hoefler, "Demystifying parallel and distributed deep learning: An in-depth concurrency analysis," *ACM Comput. Surv. (CSUR)*, vol. 52, pp. 1–43, Aug. 2019.

[8] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 491–506, Oct. 2019.

[9] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, March 2020.

[10] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 2022–2035, Jan. 2020.

[11] Q. Lan, Y. Du, P. Popovski, and K. Huang, "Capacity of remote classification over wireless channels," *[Online]. Available: https://arxiv.org/pdf/2007.15480.pdf*, 2020.

[12] J. Ren, Y. He, D. Wen, G. Yu, K. Huang, and D. Guo, "Scheduling for cellular federated edge learning with importance and channel awareness," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7690–7703, 2020.

[13] H. H. Yang, Z. Liu, T. Q. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, pp. 317–333, Sep. 2019.

[14] D. Liu and O. Simeone, "Privacy for free: Wireless federated learning via uncoded transmission with adaptive power control," *IEEE J. Sel. Areas Commun.*, vol. 39, pp. 170–185, Nov. 2020.

[15] X. Cao, G. Zhu, J. Xu, and S. Cui, "Optimized power control for over-the-air federated edge learning," *[Online]. Available: https://arxiv.org/pdf/2011.05587.pdf*, 2020.

[16] N. Zhang and M. Tao, "Gradient statistics aware power control for over-the-air federated learning in fading channels," in *Proc. IEEE Intl. Conf. Commun. Workshops*, pp. 1–6, IEEE, June 2020.

[17] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Trans. Wireless Commun.*, 2020.

[18] H. Xing, O. Simeone, and S. Bi, "Federated learning over wireless device-to-device networks: Algorithms and convergence analysis," *[Online]. Available: https://arxiv.org/pdf/2101.12704.pdf*, 2020.

[19] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Sydney, Australia), pp. 1321–1330, Aug. 2017.

[20] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *[Online]. Available: https://arxiv.org/pdf/1612.01474.pdf*, 2016.

[21] D. Barber, *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

[22] O. Simeone, "A brief introduction to machine learning for engineers," *Foundation and Trends in Signal Processing*, vol. 13, Aug. 2018 (slides available at https://nms.kcl.ac.uk/osvaldo.simeone/ml4eng.html).

[23] E. Angelino, M. J. Johnson, and R. P. Adams, "Patterns of scalable bayesian inference," *[Online]. Available: https://arxiv.org/pdf/1602.05221.pdf*, 2016.

[24] S. L. Scott, A. W. Blocker, F. V. Bonassi, H. A. Chipman, E. I. George, and R. E. McCulloch, "Bayes and big data: The consensus Monte Carlo algorithm," *Int. J. Manag. Sci. Eng. Manag*, vol. 11, pp. 78–88, Feb. 2016.

[25] M. Rabinovich, E. Angelino, and M. I. Jordan, "Variational consensus monte carlo," in *Proc. NeurIPS*, (Montreal, Canada), pp. 1207–1215, Dec. 2015.

[26] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, June 2021.

[27] W. Neiswanger, C. Wang, and E. Xing, "Asymptotically exact, embarrassingly parallel mcmc," *[Online]. Available: https://arxiv.org/pdf/1311.4780.pdf*, 2013.

[28] J. Song and M. Kountouris, "Wireless distributed edge learning: How many edge devices do we need?," *IEEE J. Sel. Areas Commun.*, 2020.

[29] X. Lyu, C. Ren, W. Ni, H. Tian, R. P. Liu, and E. Dutkiewicz, "Optimal online data partitioning for geo-distributed machine learning in edge of wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 37, pp. 2393–2406, Aug. 2019.

[30] L. Corinzia and J. M. Buhmann, "Variational federated multi-task learning," *[Online]. Available: https://arxiv.org/pdf/1906.06268.pdf*, 2019.

[31] R. Kassab and O. Simeone, "Federated generalized bayesian learning via distributed stein variational gradient descent," *[Online]. Available: https://arxiv.org/pdf/2009.06419.pdf*, 2020.

[32] S. Ahn, B. Shahbaba, and M. Welling, "Distributed stochastic gradient mcmc," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Beijing, China), pp. 1044–1052, June 2014.

[33] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[34] R. Bardenet, A. Doucet, and C. Holmes, "An adaptive subsampling approach for mcmc inference in large datasets," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Beijing, China), pp. 405–413, June 2014.

[35] M. Welling and Y. W. Teh, "Bayesian learning via stochastic gradient langevin dynamics," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Bellevue, Washington), pp. 681–688, June 2011.

[36] M. M. Amiri, T. M. Duman, D. Gunduz, S. R. Kulkarni, and H. V. Poor, "Blind federated edge learning," *[Online]. Available: https://arxiv.org/pdf/2010.10030.pdf*, 2020.

[37] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: wireless communication meets machine learning," *IEEE Commun. Mag.*, vol. 58, pp. 19–25, Jan. 2020.

[38] G. Zhu and K. Huang, "MIMO over-the-air computation for high-mobility multimodal sensing," *IEEE Internet Things J.*, vol. 6, pp. 6089–6103, Sep. 2018.

[39] G. Zhu, J. Xu, and K. Huang, "Over-the-air computing for wireless data aggregation in massive IoT," *[Online]. Available: https://arxiv.org/pdf/2009.02181.pdf*, 2020.

[40] A. Abdi, Y. M. Saidutta, and F. Fekri, "Analog compression and communication for federated learning over wireless mac," in *Proc. IEEE Intl. Workshop Signal Process. Advances Wireless Comm. (SPAWC)*, pp. 1–5, IEEE, May 2020.

[41] A. Abdi and F. Fekri, "Quantized compressive sampling of stochastic gradients for efficient communication in distributed deep learning.," in *Proc. AAAI*, (New York, USA), pp. 3105–3112, Feb. 2020.

[42] S. A. Jafar, *Interference alignment: A new look at signal dimensions in a communication network*. Now Publishers Inc, 2011.

[43] I. M. Johnstone, "High dimensional bernstein-von mises: simple examples," *Inst. Math. Stat. Collect*, vol. 6, p. 87, 2010.

[44] D. Wen, G. Zhu, and K. Huang, "Reduced-dimension design of MIMO over-the-air computing for data aggregation in clustered IoT networks," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 5255–5268, Aug. 2019.

[45] S. T. Jose and O. Simeone, "Free energy minimization: A unified framework for modelling, inference, learning, and optimization [lecture notes]," *[Online]. Available: https://arxiv.org/pdf/2011.14963.pdf*, 2020.

[46] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *Siam Review*, vol. 60, pp. 223–311, May 2018.

[47] M. Binois, D. Ginsbourger, and O. Roustant, "On the choice of the low-dimensional domain for global optimization via random embeddings," *Journal of global optimization*, vol. 76, no. 1, pp. 69–90, 2020.

[48] A. Nayebi, A. Munteanu, and M. Poloczek, "A framework for bayesian optimization in embedded subspaces," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Long Beach, USA), pp. 4752–4761, June 2019.

[49] R. Moriconi, M. P. Deisenroth, and K. S. Kumar, "High-dimensional bayesian optimization using low-dimensional feature spaces," *Machine Learning*, vol. 109, no. 9, pp. 1925–1943, 2020.

[50] A. Kristiadi, M. Hein, and P. Hennig, "Being bayesian, even just a bit, fixes overconfidence in relu networks," in *Proc. Intl. Conf. Mach. Learning (ICML)*, (Vitrual), pp. 5436–5446, July 2020.

[51] R. Zamir, *Lattice Coding for Signals and Networks: A Structured Coding Approach to Quantization, Modulation, and Multiuser Information Theory*. Cambridge University Press, 2014.

[52] J. H. Albert and S. Chib, "Bayesian analysis of binary and polychotomous response data," *J. American Stat. Assoc.*, vol. 88, no. 422, pp. 669–679, 1993.

[53] N. Bhatnagar, A. Bogdanov, and E. Mossel, "The computational complexity of estimating mcmc convergence time," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 424–435, Springer, 2011.

[54] M. Raginsky, A. Rakhlin, and M. Telgarsky, "Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis," in *Proc. Conference on Learning Theory*, (Amsterdam, Netherlands), pp. 1674–1703, PMLR, July 2017.

[55] A. S. Dalalyan and A. Karagulyan, "User-friendly guarantees for the langevin monte carlo with inaccurate gradient," *Stochastic Processes and their Applications*, vol. 129, no. 12, pp. 5278–5311, 2019.

Dongzhu Liu received the B.Eng. degree from the University of Electronic Science and Technology of China (UESTC) in 2015, and the Ph.D. degree from The University of Hong Kong in 2019. She is now a postdoctoral research associate in the Dept. of Engineering at King's College London. Her research interests include edge intelligence, federated learning, and wireless communications.

Osvaldo Simeone (Fellow, IEEE) received an M.Sc. degree (with honors) and a Ph.D. degree in information engineering from Politecnico di Milano, Milan, Italy, in 2001 and 2005, respectively. Presently, he is a Professor of Information Engineering with the Centre for Telecommunications Research at the Department of Engineering of King's College London, where he directs the King's Communications, Learning and Information Processing lab. From 2006 to 2017, he was a faculty member of the Electrical and Computer Engineering (ECE) Department at New Jersey Institute of Technology (NJIT), where he was affiliated with the Center for Wireless Information Processing (CWiP). His research interests include information theory, machine learning, wireless communications, and neuromorphic computing.

Dr Simeone is a co-recipient of the 2019 IEEE Communication Society Best Tutorial Paper Award, the 2018 IEEE Signal Processing Best Paper Award, the 2017 JCN Best Paper Award, the 2015 IEEE Communication Society Best Tutorial Paper Award and of the Best Paper Awards of IEEE SPAWC 2007 and IEEE WRECOM 2007. He was awarded a Consolidator grant by the European Research Council (ERC) in 2016. His research has been supported by the U.S. NSF, the ERC, the Vienna Science and Technology Fund, as well as by a number of industrial collaborations. He currently serves in the editorial board of the IEEE Signal Processing Magazine and is the vice-chair of the Signal Processing for Communications and Networking Technical Committee of the IEEE Signal Processing Society. He was a Distinguished Lecturer of the IEEE Information Theory Society in 2017 and 2018. Dr Simeone is a co-author of two monographs, two edited books published by Cambridge University Press, and more than one hundred research journal papers. He is a Fellow of the IET and of the IEEE.