We propose a decentralized Bayesian learning algorithm when the data set $\mathbf{X}$ is held disjointly over n agents, i.e., $\mathbf{X} = \bigcup_{i=1}^{n} \boldsymbol{X}_i$ with $\boldsymbol{X}_i \bigcap \boldsymbol{X}_j = \emptyset$ for $j \neq i$. Thus the posterior satisfies $p(\boldsymbol{w}|\mathbf{X}) = \prod_{i=1}^{n} p(\boldsymbol{w}|\boldsymbol{X}_i)$. Similar formulations can be seen in almost all embarrassingly parallel MCMC algorithms (see [35,37,38] in main paper). We will add further discussions/references on various parallel MCMC schemes[1–4] in a Related Work section. However, they do not apply to the decentralized setting since they require a central node to combine the samples from individual Markov chains. In comparison, our formulation does not require a central node: each computing node i reconstructs an approximate posterior from $\boldsymbol{X}_i$ and prior information (the 3rd term on the r.h.s of (8)) while interacting with their neighbors as dictated by the undirected communication graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (the 2nd term on the r.h.s of (8), where $a_{i,j} = 1$ if the i-th node can receive $\boldsymbol{w}_j$ from j-th node and zero otherwise). We will clarify this point and mention similar techniques in consensus-optimization (e.g., decentralized SGD[6–9]). Though our proposed algorithm is built on ULA, analysis of even the centralized ULA (C-ULA) for non-log-concave target distributions requires restricting assumptions (see lines 51-60 & discussion on [13-20]). We will add discussions on variance reduced SGLD and second-order (underdamped) Langevin algorithms (e.g.,[10–13]). Here we relax aforementioned assumptions and our Assumption 3 is weaker than the uniform bound on gradient disagreement[7] and the bounded gradient assumption[6;9].

To the best of our knowledge, we propose the first-ever decentralized ULA (D-ULA) for general non-log-concave target distributions with time-varying step-sizes. The **3** main advantages of the proposed D-ULA include: **1)** it enables the individual computing nodes to approximate the posterior with an accuracy comparable to that of the C-ULA. **2)** by using decaying step-sizes, we are able to remove the constant bias term present in the KL-divergence and show that the rate of convergence is $\mathcal{O}\left((n^{1/3}(k+1)^{\delta_2-2\delta_1})^{-1}\right)$ (see (27)). In the final version we will discus how the convergence rate and the constants C_{F_i} depends polynomially in problem dimension d_w. **3)** similar to D-SGD[6–9], D-ULA experiences speedup with the number of agents as shown by the $n^{1/3}$ in the denominator of (27). These advantages will be highlighted in the final version. Our analysis of D-ULA is **novel/non-trivial** compared to the existing non-convex consensus-optimization and non-log-concave ULA literature because: **(i)** Consensus analysis and results in Theorem 1 are novel since we use time-varying step-sizes α_k and β_k and provide an explicit consensus rate in term of step-size decay rates (see (25)) (not just bounded consensus as in[7;8]). **(ii)** Compared to existing C-ULA analysis for non-log-concave target distributions, the continuous-time approximation to the D-ULA contains an additional consensus error term $\zeta(\cdot)$ (see (21)) that complicates the analysis. Requirements on the time-varying step sizes are also not straightforward to obtain as the existing literature is focused on fixed step-sizes. We will emphasize the novelty of our analysis in the final version. D-ULA requires the same number of communication rounds as the computation iterations to achieve a prescribed level of accuracy (Corollary 1). We hope this paper provides foundations to many open research problems, such as relaxing the synchronous, periodic communication requirement of D-ULA through local computation, compression, quantization, event-triggered and asynchronous communication as done in the D-SGD[6;9], and extensions of the proposed algorithm to SGLD and noisy, time-varying communication channel.

The goal of Bayesian learning is to estimate the epistemic uncertainty for assessing confidence in the model, which is not possible with MAP or ML point estimates as illustrated using the OOD detection example (Section 5.3). Though the histogram of probability of predicted labels across all MNIST test samples using SGD shows similar trend to that of Bayesian estimates (Fig. S3, Table 2), MAP cannot quantify the uncertainty associated with the predictions. To further illustrate this point, we will include the mean and standard deviation of prediction scores for individual test samples from both (in-distribution) MNIST and (OOD) SVHN datasets using Bayesian estimates in the final version. Though D-ULA replicates the true posterior with similar fidelity as C-ULA, our analysis proves the faster convergence of D-ULA as shown in Fig 2. For the GMM experiment, algorithm parameters were selected so as to have same step-sizes for both the distributed and centralized approach. However additional experiments have shown that C-ULA can distinguish both modes with further tuning of hyper-parameters. Step-sizes for each algorithm are obtained using grid search through feasible hyperparameter space, which is derived from theoretical results (Section 4). In response to reviewers' comments, we will include the following results in the final version 1) Results of all the experiments with more number of agents 2) Plots of empirical training loss and accuracy (for classification example) versus epochs for D-ULA with varying number of agents 3) An approximate discrepancy measure based on Wasserstein distance using estimated modes of the posterior distribution for GMM[5]. However, for complicated and intractable posteriors in regression and classification applications, we rely on the metrics based on test accuracy and OOD detection.

[1] Wang and Dunson, *Parallelizing MCMC . . .*, arXiv:1312.4605, 2013. [2] Neiswanger et al., *Asymptotically exact, embarrassingly parallel MCMC*, UAI, 2014. [3] Wang et al., *Parallelizing MCMC with random partition trees*, NIPS, 2015 [4] Chowdhury and Jermaine, *Parallel and distributed MCMC via shepherding . . .*, AISTATS, 2018. [5] Givens and Shortt, *A class of Wasserstein metrics . . .*, The Michigan Mathematical Journal, 1984. [6] Signh et. al., *SPARQ-SGD . . .* arXiv:1910.14280, 2019. [7] Lian et al., *Can decentralized algorithms outperform . . .*, NeurIPS, 2017. [8] Tang et al., *D^2: Decentralized Training. . .*, ICML, 2018. [9] Koloskova et. al., *Decentralized Deep Learning . . .*, ICLR 2020. [10] Dubey et. al., *Variance Reduction in SGLD*, NIPS, 2016. [11] Chatterji et. al., *On the Theory of Variance Reduction . . .*, ICML, 2018. [12] Cheng et. al., *Underdamped Langevin MCMC. . .*, COLT, 2018. [13] Şimşekli et. al., *Fractional Underdamped Langevin. . .*, arXiv:2002.05685, 2020.