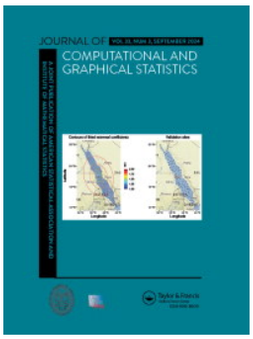


Taylor & Francis
Taylor & Francis Group

Journal of Computational and Graphical Statistics

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/ucgs20

Bayesian Federated Learning with Hamiltonian Monte Carlo: Algorithm and Theory

Jiajun Liang, Qian Zhang, Wei Deng, Qifan Song & Guang Lin

To cite this article: Jiajun Liang, Qian Zhang, Wei Deng, Qifan Song & Guang Lin (15 Jul 2024): Bayesian Federated Learning with Hamiltonian Monte Carlo: Algorithm and Theory, Journal of Computational and Graphical Statistics, DOI: 10.1080/10618600.2024.2380051

To link to this article: https://doi.org/10.1080/10618600.2024.2380051

Bayesian Federated Learning with Hamiltonian Monte Carlo: Algorithm and Theory

Jiajun Liang[a], Qian Zhang[b], Wei Deng[c], Qifan Song[b,*], Guang Lin[d]

[a]ByteDance Inc, China

[b]Department of Statistics, Purdue University, West Lafayette, IN

[c]Machine Learning Research, Morgan Stanley, New York, NY

[d]Department of Mathematics & School of Mechanical Engineering, Purdue University, West Lafayette, IN.

*qfsong@purdue.edu

Abstract

This work introduces a novel and efficient Bayesian federated learning algorithm, namely, the Federated Averaging stochastic Hamiltonian Monte Carlo (FA-HMC), for parameter estimation and uncertainty quantification. We establish rigorous convergence guarantees of FA-HMC on non-iid distributed data sets, under the strong convexity and Hessian smoothness assumptions. Our analysis investigates the effects of parameter space dimension, noise on gradient and momentum, and the frequency of communication between the central node and local nodes) on the convergence and communication costs of FA-HMC. Beyond that, we establish the tightness of our analysis by showing that the convergence rate cannot be improved even for continuous FA-HMC process. Moreover, extensive empirical studies demonstrate that FA-HMC outperforms the existing Federated Averaging-Langevin Monte Carlo (FA-LD) algorithm.

Keywords: Hamiltonian Monte Carlo, Federated Learning, Bayesian sampling, Federated averaging, Stochastic Gradient Langevin Dynamics

1 Introduction

Standard learning algorithms usually require centralizing the training data, in the sense that the learning machine can directly access all pieces of the data. Federated learning (FL), on the other hand, enables multiple parties to collaboratively train a consensus model without directly sharing confidential data (Konečnỳ et al., 2015, 2016; Bonawitz et al., 2019; Li et al., 2020a). The framework of FL is quite appealing to applications where data confidentiality is of vital importance, such as aggregating user app data from mobile phones to learn a shared predictive model (e.g., Tran et al., 2019; Chen et al., 2020a) or analyzing medical data from multiple healthcare stakeholders (e.g., hospitals, research centers, life science companies) (e.g., Li et al., 2020c; Rieke et al., 2020).

FL shares a similar algorithmic architecture to parallel optimization. First, parallel algorithms are commonly based on the divide-and-combine strategy, i.e., the learning system assigns (usually i.i.d.) training samples to each worker node, say via simple random sampling. As such, the training data sets are similar in nature across worker nodes. But under the FL framework, the data sets of each worker node are generated or collected locally and are not homogeneous, which poses challenges for convergence analysis. Secondly, parallel computing is commonly practiced in the same physical location, such as a data center, where high throughput computer networking communications are available between worker nodes. In contrast, FL has either a vast number of worker nodes (e.g., mobile devices) or geographically separated worker nodes (e.g., hospitals), which limits the connectivity between the central nodes and worker nodes. Due to the unavailability of fast or frequent communication, FL needs to be communication-efficient.

Federated Averaging (FedAvg, McMahan et al., 2017) is one of the most widely used FL optimization algorithms. It trains a global model by synchronously averaging multi-

step local stochastic gradient descent (SGD) updated parameters of all the worker nodes. Various attempts have been made to enhance the robustness and efficiency of FedAvg (e.g., Li et al., 2020b; Wang et al., 2020). However, optimization-based approaches often fail to provide proper uncertainty quantification for their estimations. Reliable uncertainty quantification, such as interval estimations or hypothesis testing, provides a vital diagnostic for both developers and users of an AI system.

The Bayesian counterpart naturally integrates an inference component, thus it provides a unified solution for both estimations and uncertainty quantification. This paper studies a Bayesian computing algorithm aiming to obtain samplers from the global posterior distribution by infrequently aggregating samples drawn from local posterior distributions. Unlike existing results that utilize stochastic gradient Langevin dynamics (Welling and Teh, 2011), this work considers (stochastic gradient) Hamiltonian Monte Carlo (HMC, Neal, 2012). While the second-order nature of HMC poses more theoretical difficulties, it has been demonstrated to be more computationally efficient through numerous empirical studies (see, e.g., Girolami and Calderhead, 2011; Chen et al., 2014). Readers can refer to Section A in Supplementary Material for a review of related literature on federated sampling and HMC.

The contributions of the presented work are three-fold:

(1) We propose the Federated Averaging Hamiltonian Monte Carlo (FA-HMC) algorithm which is effective for global posterior inferences in federated learning. It utilizes stochastic gradient HMC on individual local nodes and combines the local samples obtained infrequently to yield global samples.

(2) Under strong log-concavity and proper smoothness assumptions, we have proven a non-asymptotic convergence result under the Wasserstein metric for various training settings. Furthermore, we demonstrate that this upper bound of the convergence rate of the FA-HMC sampling algorithm is tight (i.e., best achievable for certain sampling problems).

(3) We conduct simulation and real data experiments to validate our theoretical findings. Additionally, the numerical studies show that FA-HMC is easy to tune, improves communication efficiency, and can outperform FA-LD in different settings.

Roadmap:

The paper is organized as follows: In Section 2, we summarize the problem setup and provide the necessary background on HMC. In Section 3, we present the FA-HMC algorithm and the assumptions used for its analysis. In Section 4, we provide the key theoretical findings and examine the effects of SGD noise and the correlation between momentum. Furthermore, we prove that our analysis is tight and cannot be improved for certain sampling problems, even for continuous FA-HMC. In Section 5, we compare the FA-HMC algorithm with the FA-LD algorithm through extensive simulations and real-data experiments. Finally, in Section 6, we conclude our work and suggest potential future directions.

2 Preiminary

2.1 Problem Setup

Let $z_i^c, 1 \le i \le n_c$ be the available data of the c-th node and $\ell(\theta; z_i^c)$ be a user-specified negative log-likelihood function. Define $n = \sum n_c$, $w_c = n_c / n$, and $f^{(c)}(\theta) := n \sum_{i=1}^{n_c} \ell(\theta; z_i^c) / n_c$ is the local loss function of parameter $\theta \in \mathbb{R}^d$ accessible to the c-th local node (e.g., the normalized negative log-likelihood function based on the data set available at c-th local node) for $1 \le c \le N$. The goal is to simulate the global target distribution $\pi(\theta) \propto \exp(-f(\theta))$, where $f(\theta) = \sum_{c=1}^{N} w_c f^{(c)}(\theta), w_c \ge 0$ and $\sum_c w_c = 1$.

2.2 Hamilton's Equations and HMC

Hamiltonian (Hybrid) Monte Carlo (HMC) was first proposed by Duane et al. (1987) for simulations of quantum chromodynamics and was then extended to molecular dynamics and neural networks Neal (2012). To alleviate the random-walk behavior in the vanilla Langevin dynamics, HMC simulates the trajectory of a particle according to Hamiltonian dynamics and obtains a much faster convergence rate than Langevin dynamics Mangoubi and Vishnoi (2018). In specific, HMC introduces a set of auxiliary momentum variables $p \in \mathbb{R}^d$ to capture second-order information, whereas Langevin Monte Carlo is only a first-order method. In this way, HMC generates samples from the following joint distribution

$$\pi(\theta, p) \propto \exp(-f(\theta) - \frac{1}{2} p' \Sigma^{-1} p),$$

where $f(\theta) + p' \Sigma^{-1} p / 2$ is the Hamiltonian function and quantifies the total energy of a physical system. To further generate more efficient proposals, HMC simulates according to the following Hamilton's equations

$$\frac{d\theta(t)}{dt} = \Sigma^{-1/2} p(t), \quad \frac{dp(t)}{dt} = -\Sigma^{1/2} \nabla_\theta f(\theta(t)), \quad (1)$$

which satisfy the conservation law and are time reversible. Such properties leave the distribution invariant and the nature of Hamiltonian conservation always makes the proposal accepted ideally. Note that commonly, one chooses $\Sigma = \mathbb{I}_d$ such that the momentum follows the standard multivariate normal distribution.

To numerically implement the continuous HMC process, a popular numerical integrator is the "leapfrog" approximation, see Algorithm 1. Here, to enhance the computational efficiency, $\nabla \tilde{f}(\theta_k, \xi_k)$ and $\nabla \tilde{f}(\theta_{k+1}, \xi_{k+1/2})$ are the stochastic versions of $\nabla f(\theta_k)$ and $\nabla f(\theta_{k+1})$, respectively. The arguments ξ_k and $\xi_{k+1/2}$ denote random variables that control the randomness of the stochastic gradients. For example, given $f(\theta) = \sum_{i=1}^{n} \ell(\theta; z_i)$ with

data $\{z_i\}_{i=1}^n$, we let $\nabla\tilde{f}(\theta,\xi_k) = n\sum_{i\in S(\xi_k)}\nabla\ell(\theta;z_i)/|S(\xi_k)| + Z(\xi_k)$ where $S(\xi_k)$ is a random index subset, $Z(\xi_k)$ is an injected Gaussian noise, and ξ_k is the random seed. When the exact gradients are used, it holds that $\nabla\tilde{f}(\theta_k,\xi_k) = \nabla f(\theta_k)$ and $\nabla\tilde{f}(\theta_{k+1},\xi_{k+1/2}) = \nabla f(\theta_{k+1})$. Note that throughout this paper, when the exact gradient is used instead of a stochastic gradient, the algorithm is referred to as the vanilla version, e.g., *vanilla* FA-HMC.

Algorithm 1 Stochastic gradient leapfrog approximation $\tilde{h}_{\mathrm{LF}}$

Input: Energy function $f(\cdot)$; Initial parameters θ_0, momentum p_0; learning rate η; leapfrog step K; $k = 0$

while $k \le K$: **do**

$$\theta_{k+1} = \theta_k + \eta p_k - \frac{\eta^2}{2}\nabla\tilde{f}(\theta_k,\xi_k)$$

$$p_{k+1} = p_k - \frac{\eta}{2}\nabla\tilde{f}(\theta_k,\xi_k) - \frac{\eta}{2}\nabla\tilde{f}(\theta_{k+1},\xi_{k+\frac{1}{2}})$$

$$k = k+1;$$

Output: $\tilde{h}_{\mathrm{LF}}(f,\theta_0,p_0,\eta,K) = \theta_K$

For convenience in analysis, the leapfrog method without Metropolis correction (see Algorithm 2), is commonly studied in the literature (Mangoubi and Vishnoi, 2018; Chen and Vempala, 2022; Zou and Gu, 2021). One may also add an additional accept/reject step according to the Metropolis ratio (Chen et al., 2020b).

Algorithm 2 HMC algorithm (without Metropolis correction)

Input: Energy function $f(\cdot)$; Initial point θ_0; Stepsize function $\eta_t = \eta(t)$; Leapfrog step K; $t = 0$;

while the stopping rule is not satisfied **do**

sample momentum $p_t \sim N(0,\mathbb{I}_d)$

update $\theta_{t+1} = \tilde{h}_{\mathrm{LF}}(f, \theta_t, p_t, \eta_t, K), t = t+1$;

Output: $\{\theta_i\}_{i=1}^{t}$

Note that in the literature, Chen et al. (2014) proposed a different HMC algorithm, based on Euler integrator of Hamilton dynamics. Their implementation includes variance adjustment to counteract the noise of the stochastic gradient, which can negatively impact the stationary distribution. This adjustment eventually leads to an underdamped Langevin Monte Carlo algorithm with stochastic gradient (see also e.g., Ma et al., 2015; Zou et al., 2019; Chau and Rasonyi, 2022; Akyildiz and Sabanis, 2020; Nemeth and Fearnhead, 2021).

3 FA-HMC Algorithm and Assumptions

Ensuring the confidentiality of the data utilized for training a model is a vital concern in federated learning. To safeguard against potential gradient leakage (Zhu et al., 2019) and breaches of local data privacy, it is preferable to use noisy gradients and less-correlated momentum among local nodes (see Deng et al., 2021; Vono et al., 2022). This could make it more difficult to recover local data information through accumulated communication.

With these considerations, we propose Federated Averaging via HMC algorithm that utilizes general stochastic gradients and non-necessarily identical momentum across nodes. We let all local devices run HMC (Algorithm 1), and synchronize their model parameters every J iteration. All devices may use stochastic gradients and share part of the initial momentum of leapfrog approximation. Note that in practice, correlated momentum between devices can be easily achieved by sending a common random seed to all devices for momentum generation. This FA-HMC algorithm is formalized in Algorithm 3.

Algorithm 3 FA-HMC algorithm

Input: $\theta_0^{(c)} = \theta_0$, t = 0; stepsize function $\eta_t = \eta(t)$; Local update step T; leapfrog update step K;

while the stopping rule is not satisfied **do**

sample momentum $p_t^{(c)}$

if $t \equiv 0 (\mathrm{mod}\, T)$ **then**

Broadcast $\theta_t := \sum_{c=1}^{N} w_c \theta_t^{(c)}$ and set $\theta_{t+1,0}^{(c)} = \theta_t$

else

$\theta_{t+1,0}^{(c)} = \theta_t^{(c)}$

update $\theta_{t+1}^{(c)} = \tilde{h}_{\mathrm{LF}}(f^{(c)}, \theta_{t+1,0}^{(c)}, p_t^{(c)}, \eta_t, K)$ in parallel for all devices, $t = t+1$

It is worth mentioning that when leapfrog step K = 1, the leapfrog approximation of the unadjusted HMC algorithm (i.e., Algorithm 1) reduces to $\theta_{t+1} = \theta_t - (\eta_t^2/2)\nabla_\theta \tilde{f}(\theta_t, \xi_t) + \eta_t N(0, \mathbb{I}_d)$, which is exactly the unadjusted Langevin Monte Carlo with dynamic learning rate $\eta_t^2/2$, and the FA-HMC reduces to FA-LD Deng et al. (2021).

3.1 Assumptions

To establish the convergence performance of the aggregated model with respect to θ_t, we adopted the following assumptions.

Assumption 3.1 (μ-Strongly Convex). *For each* $c = 1, 2, \ldots, N$, $f^{(c)}$ *is μ-strongly convex for some* $\mu > 0$, *i.e.,* $\forall x, y \in \mathbb{R}^d$, $f^{(c)}(y) \geq f^{(c)}(x) + \langle \nabla f^{(c)}(x), y - x \rangle + \frac{\mu}{2} \| y - x \|_2^2$.

Assumption 3.2 (L-Smoothness). *For each* $c = 1, 2, \ldots, N$, $f^{(c)}$ *is L-smooth for some L > 0, i.e.,* $\forall x, y \in \mathbb{R}^d, \| \nabla f^{(c)}(y) - \nabla f^{(c)}(x) \| \leq L \| x - y \|$.

Assumption 3.3 (L_H-Hessian Smoothness). *For each* $c = 1, 2, \ldots, N$, $f^{(c)}$ *is* L_H *Hessian smoothness, i.e., for any* $\theta_1, \theta_2, p \in \mathbb{R}^d$, $\| \left(\nabla^2 f^{(c)}(\theta_1) - \nabla^2 f^{(c)}(\theta_2) \right) p \|^2 \leq L_H^2 \| \theta_1 - \theta_2 \|^2 \| p \|_\infty^2$.

Assumptions 3.1-3.2 are commonly used for the convergence analysis of gradient-based MCMC algorithms (e.g., Dalalyan, 2017; Mangoubi and Vishnoi, 2018; Dalalyan and Karagulyan, 2019; Erdogdu and Hosseinzadeh, 2021, and references therein). The strong convexity condition, in some theoretical literature of stochastic Langevin Monte Carlo, has also been relaxed to the dissipativity condition (e.g., Raginsky et al., 2017; Zou et al., 2021) for non-log-concave target distributions. But such an extension is beyond the scope of this paper and will be investigated in future works. Assumption 3.3 ensures second-order smoothness of energy functions beyond gradient Lipchitzness. Similar Hessian smoothness conditions are used in the literature. For example, Dalalyan and Karagulyan (2019); Chen et al. (2020b), Zou et al. (2021) required the Hessian matrix of energy function to be Lipchitz under ℓ_2 operator norm. In comparison, Assumption 3.3 is a stronger requirement since ℓ_∞ norm appears on the RHS. Our assumption is somewhat comparable to Assumption 1 of Mangoubi and Vishnoi (2018) which defines a semi-norm with respect to a set of pre-specified unit vectors.

We require an additional assumption to model stochastic gradients. Denote $\theta_{t,k}^{(c)}$ as the position parameter of the c-th local node at iteration t and leapfrog step k, and $\xi_{t,x}^{(c)}$ ($x = k - 1/2, k$) as the corresponding variable that controls the randomness of gradient.

Assumption 3.4 (σ_g-Bounded Variance). *For local device* $c = 1, 2, \ldots, N$, *and leapfrog step* $k = 1, 2, \ldots, K$, $t = 1, 2, \ldots$, *we have* $\max_{x=k-1/2,k} \mathrm{tr}(\mathrm{Var}(\nabla \tilde{f}^{(c)}(\theta_{t,k}^{(c)}, \xi_{t,x}^{(c)}) \mid \theta_{t,k}^{(c)})) \leq \sigma_g^2 L d$, *for some* $\sigma_g > 0$.

This is a common assumption in the literature (see Gürbüzbalaban et al., 2021; Vono et al., 2022; Deng et al., 2021). It is worth noting that in practice, the stochastic gradient

is computed based on a random subsample of the whole dataset, thus the variability of the stochastic gradient can be naturally controlled by adjusting the batch sizes.

Under our framework, we can also relax the above assumption to

$$\max_{x=k-1/2,k} \mathrm{tr}(\mathrm{Var}(\nabla\tilde{f}^{(c)}(\theta_{t,k}^{(c)},\xi_{t,x}^{(c)})\,|\,\theta_{t,k}^{(c)})) \le \sigma_g^2(G_{t,k}^{(c)}+d),$$

without significant changes to our proof, where $G_{t,k}^{(c)}$ denote $\|\nabla f^{(c)}(\theta_{t,k}^{(c)})\|^2$. The extension of the proof to accommodate this assumption is discussed in Section K in the appendix.

Before presenting our main result, we emphasize that this paper examines the convergence of the FA-HMC sampling algorithm, specifically in regard to dimension d and error ϵ. It also explores ways to adjust the algorithm to maintain its effectiveness when considering variations in gradient and momentum noise. Adapting the FA-HMC algorithm to more general settings like non-convexity will be our future study.

4 Theoretical Results

In Section 4.1, we describe the general convergence rate of FA-HMC on different settings and point out the setting where FA-HMC achieves the fastest speed and least communication cost. In Section C of the supplementary material, we argue that that the upper bound on the nearly ideal case is tight by giving a matching lower bound result. In Section 4.2, we present a detailed result of the convergence behavior of the FA-HMC algorithm.

4.1 Main Results

Define $\theta^* := \mathrm{argmin}_\theta f(\theta)$ and denote the marginal distribution of θ_t by π_t. Given two probability measures μ and ν, the 2-Wasserstein distance is $\mathcal{W}_2(\mu,\nu) = \inf_{X\sim\mu,Y\sim\nu}(\mathbb{E}\|X-Y\|^2)^{1/2}$. The following theorem describes the general convergence rate of FA-HMC.

Theorem 4.1. *Assume 3.1-3.4, and* $\mathcal{W}_2(\pi_0,\pi)^2 = O^1(d)$ *and* $\sum_{c=1}^{N} w_c \| \nabla f^{(c)}(\theta^*) \|^2 = O(d)$ *. For a given local iteration step T, there exists some constant C depending on* $L, L/\mu, L_H^2/L^3$ *such that if we choose* $\eta(t) \equiv \eta$ *and (denote* $\gamma = (K\eta)^2$ *)*

$$\eta^2 = \frac{\gamma}{K^2} = C \min\Big\{\frac{1}{K^2 L}, \frac{\epsilon}{K^2\sqrt{dT}}, \frac{\epsilon^2}{K^2 dT^2(1-\rho)N}, \frac{\epsilon^2}{Kd\sum_{c=1}^{N} w_c^2 \sigma_g^2}\Big\}$$

then $\mathcal{W}_2(\pi_{t_\epsilon}, \pi) \le \epsilon$ *for any* $\epsilon > 0$ *, with iteration number*

$$t_\epsilon = \frac{d\log(d/\epsilon^2)}{\epsilon^2}\tilde{O}^1\Big(T^2\big(\gamma + (1-\rho)N\big) + \frac{\sum_{c=1}^{N} w_c^2\sigma_g^2}{K}\Big)$$

and corresponding communication times

$$\frac{t_\epsilon}{T} = \frac{d\log(d/\epsilon^2)}{\epsilon^2}\tilde{O}\Big(T\big(\gamma + (1-\rho)N\big) + \frac{\sum_{c=1}^{N} w_c^2\sigma_g^2}{K}\Big).$$

When one uses small batch stochastic gradients (i.e., large σ_g) or less correlated momentum (i.e, small ρ) to improve computational feasibility and protect privacy, the proposed γ is negligible. Under this scenario, the required number of iterations is of rate $\tilde{O}(d/\epsilon^2)$ with respect to the dimension d and precision level ϵ.

Remark 4.2. Regarding the stopping rule of algorithms 2 and 3, Theorem 4.1 does provide a nonasymptotic choice of t_ϵ to achieve an ϵ-W_2 error in theory. But this bound is impractical, as it relies on the unknown distributional properties of the target distribution. For more practical rules, various suggestions have been made in the literature (e.g., Gelman et al., 1995). For example, (i) From a visual inspection perspective, we can randomly pick some dimensions and visually compare the trace plots between two parts of a single chain (by splitting one chain in half) or between two chains. We keep running the chains until they become "approximately" stationary; (ii)

From a quantitative perspective, we can compute the between- and within-sequence variances following the potential scale reduction factor $\hat{R}$ defined in Eq.(11.4) of Gelman et al. (1995), the stopping rule can be triggered when $\hat{R} \approx 1$. Note that it is beyond the scope of this paper to design a stopping rule with statistical guarantees.

The result of Theorem 4.1 also shows that for a fixed ϵ, under proper tuning, the communication cost t_ϵ / T may initially decrease and then increase as the number of local HMC iteration steps T increases (i.e., a 'U' curve w.r.t, T). Therefore, there is a trade-off between communication and divergence, and an optimal choice for local iteration can be made. Similar discoveries were also argued by Deng et al. (2021) for Bayesian Federated Averaging Langevin system. The above results provide a certain level of direction for optimizing the performance of FA-HMC algorithms, considering any well-defined federated learning loss that accounts for total running time, overall communication cost, and divergence.

For instance, by reducing the noise of the stochastic gradients and improving correlation between momentum to a certain level, we can achieve significant improvement on the convergence speed from $\tilde{O}(d/\epsilon^2)$ to $\tilde{O}(\sqrt{d}/\epsilon)$, which is argued by the following proposition.

Proposition 4.3. *With the assumptions as stated in Theorem 4.1, if we choose* $\eta(t) \equiv \eta$ *and (denote* $\gamma = (K\eta)^2$

$$\gamma = C\min\{\frac{1}{L}, \frac{1}{T\sqrt{L}}\} \quad \rho = 1 - O(\frac{\gamma}{N}), \quad \sigma_g^2 = O(K\gamma) \qquad (2)$$

then it achieves that $\mathcal{W}_2(\pi_{t_\epsilon}, \pi) \le \epsilon$*, where* π_t *denotes the marginal distribution of* θ_t*, with iteration t and corresponding communication times* t_ϵ / T *as*

$$t_\epsilon = \frac{\sqrt{d}\log(d/\epsilon^2)}{\epsilon}\tilde{O}(T), \qquad \frac{t_\epsilon}{T} = \tilde{O}\Big(\frac{\sqrt{d}\log(d/\epsilon^2)}{\epsilon}\Big).$$

Under the setting (2), referred to as vanilla FA-HMC, the obtained convergence rate matches that of the underdamped Langevin Monte Carlo algorithm on a single device in Cheng et al. (2018) and is superior to that of Federated Averaging of underdamped Langevin Monte Carlo algorithm under decentralized setting (i.e., rate $\tilde{O}(d/\epsilon^2)$ in Gürbüzbalaban et al., 2021). It also matches existing results about Federated Langevin algorithm tackling heterogeneity under the federated learning framework Plassier et al. (2023) and is better than those without hessian smoothness assumption Deng et al. (2021).

Furthermore, in Section C of the supplementary material, we establish a lower bound for $t_\epsilon = \Omega(\sqrt{d}T\log(d/\epsilon)/\epsilon)$ for some log-concave target distribution. In other words, our result in Proposition 4.3 is tight w.r.t. dimension *d* and local iteration *T*. This tight result implies that (1) Unlike the "U" curve with respect to *T* discovered in Theorem 4.1, when there are small stochastic gradients and large correlations between momentum, communication times have limited variations in *T*. Therefore, the tradeoff between communication and divergence will not exist for vanilla FA-HMC and it suggests a small local iteration *T* to minimize unnecessary computation; and (2)In terms of rate dependency w.r.t. the dimension, under similar conditions on the Hessian matrix, the rate of single-device HMC is as low as $O(d^{1/4})$ Mangoubi and Vishnoi (2018), which is strictly better than our rate $O(d^{1/2})$ under the federated learning setting. This intrinsic gap is caused by (i) FA algorithm design and (ii) the use of stochastic gradient.

4.2 Convergence Behaviour for FA-HMC Algorithm

For correlated momentum, for simplicity of analysis, we consider the following setting

$$p_t^{(c)} = \sqrt{\rho}\xi_t + \sqrt{1-\rho}\xi_t^{(c)}/\sqrt{w_c}, \quad \text{for all } c \in [N], t \geq 1,$$

where $\xi_t, \xi_t^{(c)}$ are independent standard Gaussian and the ξ_t are the shared across all local nodes and $\xi_t^{(c)}$'s are private to each local node *c*.

Here the factor $1/\sqrt{w_c}$ on $\xi_t^{(c)}$ is a scaling treatment such that the average momentum is a standard Gaussian. To see this, note that the average momentum $p_t = \sum_{c=1}^{N} w_c p_t^{(c)}$ has a smaller variance due to the correlation between $\{p_t^{(c)}\}_c$. By direct calculations, we have

$$\mathbb{E}\| p_t^{(c)} \|^2 = (\rho + \frac{1-\rho}{w_c})d, \qquad \mathbb{E}\| p_t \|^2 = d.$$

Note that for FA-HMC, the momentum of each local device is not standard Gaussian. This is to ensure that the center momentum (i.e., $p_t = \sum_{c=1}^{N} w_c p_t^{(c)}$ aggregated from local momentum) is close to the standard Gaussian. This is a special setting induced by distributed sampling and the goal of privacy preservation.

We define the aggregated global model $\theta_t := \sum_{c=1}^{N} w_c \theta_t^{(c)}$ for all $t \geq 1$. Note that θ_t, in practice, is not accessible unless $t \equiv 0 (\mathrm{mod}\, T)$. For $t \geq 0$, we also define θ_{t+1}^{π} as the parameter resulting from the evolution over Kn time following dynamic (1) with initial position θ_t^{π} and momentum p_t. With the above preparations, to intuitively understand the convergence of the distribution of θ_t, we take the vanilla FA-HMC as an example. We can decompose $\theta_{t+1}^{(c)} - \theta_{t+1}^{\pi}$ as follow:

$$\theta_{t+1}^{(c)} - \theta_{t+1}^{\pi} = (\mathrm{I}_1) - n^2 \sum_{k=1}^{K-1} (K-k)(\mathrm{I}_2)_k - (\mathrm{I}_3),$$

where

$$(\mathrm{I}_1) = \theta_{t,0}^{(c)} - \frac{(K\eta)^2}{2}\nabla f^{(c)}(\theta_{t,0}^{(c)}) - \frac{(K^3-K)\eta^3}{6}\nabla^2 f^{(c)}(\theta_{t,0}^{(c)})$$
$$\cdot p_t - \Big(\theta_t^{\pi} - \frac{(K\eta)^2}{2}\nabla f(\theta_t^{\pi}) - \frac{(K\eta)^3}{6}\nabla^2 f(\theta_t^{\pi})p_t\Big);$$
$$(\mathrm{I}_2)_k = \nabla f^{(c)}(\theta_{t,k}^{(c)}) - \nabla f^{(c)}(\theta_{t,0}^{(c)}) - \nabla^2 f^{(c)}(\theta_{t,0}^{(c)})\eta p_t k$$
$$(\mathrm{I}_3) = \int_0^{K\eta}\int_0^{s}\nabla f(\theta_t^{\pi}(u)) - \nabla f(\theta_t^{\pi}) - \nabla^2 f(\theta_t^{\pi})p_t u du ds.$$

Here (I_1) represents second-order random approximation of $\theta_{t+1}^{(c)} - \theta_{t+1}^{\pi}$ through $\theta_t^{(c)}$ and θ_t^{π}, and we expect that

$$\mathbb{E}\,\|\sum_{c=1}^{N} w_c(\mathrm{I}_1)\|^2 \le \alpha_t \mathbb{E}\,\|\theta_t - \theta_t^{\pi}\|^2 + \varepsilon_t^2,$$

where the contraction factor $\alpha_t \in (0,1)$ and one-iteration divergence error $\varepsilon_t > 0$.

On the other hand, $\|(\mathrm{I}_2)_k\|$ and $\|(\mathrm{I}_3)\|$ represent second-order approximation error and are expected to be $O((K\eta_t)^2\sqrt{d})$.

By utilizing Lemma D.1, the overall behavior is summarized in the following theorem.

Theorem 4.4 (Convergence). *Under Assumptions 3.1-3.4, if we set* $\eta_{t''} \le \eta_{t'} \le 1/(K\sqrt{L})$ *for any* $t' \le t''$ *in Algorithm 3, then* $\{\theta_t\}_t$ *satisfies*

$$\mathbb{E}\,\|\theta_{t+1} - \theta_{t+1}^{\pi}\|^2 \le (1 - \frac{\mu(K\eta_t)^2}{4})^t \mathbb{E}\,\|\theta_0 - \theta_0^{\pi}\|^2 + \eta_t^2\Delta_t$$

where there exist constants $C_1, C_2 > 0$ *depending on* $L, L/\mu, L_H^2/L^3$ *and* $c_d = \log^2(d)$, *such that*

$$\Delta_t = C_1 T^2 K^2 \sum_{c=1}^{N}\Big(\underbrace{\frac{w_c B_{\nabla}^{(c)}}{L}}_{\text{Bias}}(K\eta_t)^2 + \underbrace{\frac{1-\rho}{L}d}_{\text{Correlation}}\Big) + C_2 K \underbrace{\sum_{c=1}^{N} w_c^2\sigma_g^2 d}_{\text{Stoc. Grad.}}$$

with $B_{\nabla}^{(c)} := \sup_t \mathbb{E}\,\|\nabla f^{(c)}(\theta_{t,0}^{(c)})\|^2$.

The proof is postponed to Section G in the supplementary material. The divergence error is made up of three main components: error resulting from bias across local nodes (which includes heterogeneity and sampling cost), momentum noise, and gradient noise. In the absence of stochastic gradients and when momentum is identical across nodes, the only errors present are lower-order biases. Similar intermediate contraction results have been derived in the literature on gradient-based sampling algorithms (e.g., Deng et al., 2021; Plassier et al., 2023).

By the definition of Wasserstein metric, Theorem 4.4 immediately establishes a convergence result of the marginal distribution of θ_t, denoted by π_t, towards π under Wasserstein-2 distance. The convergence result involves a term $\sum_{c=1}^{N} w_c \sup_t \mathbb{E} \| \nabla f^{(c)}(\theta_{t,0}^{(c)}) \|^2 / L$. In Lemma D.7 in the appendix, we shows that uniformly, $\mathbb{E} \| \nabla f^{(c)}(\theta_{t,0}^{(c)}) \|^2 = \tilde{O}(\sum_{c=1}^{N} w_c \| \nabla f^{(c)}(\theta^*) \|^2 + L\mathbb{E} \| \theta_0 - \theta_0^\pi \|^2 + d)$, omitting its dependency on constants $L, L/\mu$ and L_H^2 / L^3, and in consequence, solving the two inequalities

$$(1-\mu(K\eta_t)^2/4)^t \mathbb{E} \| \theta_0 - \theta_0^\pi \|^2 \le \epsilon^2/2, \qquad \eta_t L \le \epsilon^2/2,$$

we obtain Theorem 4.1.

On the other hand, in literature, people design settings for converging learning rate such that the extra logarithmic factor in the convergence result can be removed. We also obtain a similar result on a learning rate design as stated in the following proposition.

Proposition 4.5 (Dynamic stepsize). *Under Assumptions 3.1-3.4, there is a setting of* $\{\eta_t\}_t$ *for Algorithm 3 such that* $\mathbb{E} \| \theta_t - \theta_t^\pi \|^2 \le \epsilon^2$ *at some*

$$t \le C \log^2(d) d \left(T^2(\gamma + (1-\rho)N) + \sum_{c=1}^{N} w_c^2 \sigma_g^2 / K \right) / \epsilon^2$$

, with

$$\gamma = \min\{1/\sqrt{L}, \epsilon/\sqrt{d}T, \epsilon^2/(dT^2(1-\rho)N), \epsilon^2 K / (d\sum_{c=1}^{N} w_c^2 \sigma_g^2)\}$$

.

By this proposition, we see that the $\log(d/\epsilon^2)$ factors are removed in the convergent iteration compared to Theorem 4.1. One setting of η_t that satisfies the claims in Proposition 4.6 is specified in the proof (i.e., Section H in the supplement file).

5 Experiments

In this section, we first compare the empirical performance of FA-HMC and FA-LD on simulated data. Then we examine the relationship between dimension and communication round in our theoretical suggested setting of the learning rate. Last we present the performance of FA-HMC on the real datasets. We apply FA-HMC with constant stepsize η and the same momentum initialization across devices. We conduct the synchronization of the model parameters every T local leapfrog step in the implementation of FA-HMC. Due to the significant computational costs involved in evaluating performance at each cohort level, some results in this section are obtained from a single run and others are obtained by averaging multiple runs. We defer part of the results with error bars to Section L in the supplementary materials.

5.1 Simulation: FA-HMC vs FA-LD

We first sample from the posterior of a Bayesian logistic regression on a simulated dataset of dimension d = 1000 (Mangoubi and Vishnoi, 2018). Specifically, we split the dataset of size 1000 equally into 20 local nodes; we run the experiments using both exact gradients (i.e., vanilla version) and stochastic gradients, where the later ones are simulated by adding an independent zero-mean Gaussian noise of variance $\sigma^2 = 100$ to each coordinate of the true gradients. Note that simulating the randomness of the stochastic gradient by a normal variable is consistent with the experiment setting in Mangoubi and Vishnoi (2018). We argue that Gaussian noise is a reasonable approximation when invoking the central limit theorem with a large enough batch size.

As the benchmark, we run Metropolis-adjusted HMC (MHMC) for a sufficient number of iterations. To evaluate the performance of FA-HMC, we use the computable metric

$\frac{1}{d}\sum_{i=1}^{d}\mathcal{W}_1(\mu_i, \nu_i)$ as a measure of marginal error (ME) of two sets of samples, as proposed by Mangoubi and Vishnoi (2018); Faes et al. (2011). This metric compares the empirical distributions of the *i*-th coordinate of the two sets of samples, represented by μ_i and ν_i, respectively.

We first compare FA-HMC with FA-LD (i.e., FA-HMC with $K = 1$). Noticing that the communication limit is a major bottleneck for federated learning, we suppose the local computation cost is negligible compared with the communication cost. Therefore, the comparison between FA-HMC and FA-LD is based on the same number of communications, or equivalently, the same number of steps t. Fixing local step $T = 10$, we try different stepsizes η and leapfrog steps K. For FA-HMC, we set $K = \lfloor \pi/(3\eta) \rfloor$ following Mangoubi and Vishnoi (2018) when $\eta \leq 0.01$ and tune K (such that the performance is optimal w.r.t the choice of K) when $\eta \geq 0.02$. Each run consists of 2×10^7 steps and we collect the same number of samples from the last 10^7 steps. We plot the curves of the calculated MEs against η in Figure 1(a) (exact gradients, G) and 1(b) (stochastic gradients, SG). We observe that in this task, where FA-LD is already a competitive baseline, *FA-HMC still significantly outperforms FA-LD with around 5% improvement on the performance*. Moreover, we realize that a wide range of stepsizes for FA-HMC yields pretty decent performance. As such, FA-HMC appears to be more robust w.r.t. its hyperparameters around the optimal choices, suggesting that FA-HMC is easier to tune than FA-LD, and a small stepsize usually leads to a good performance.

Next, we study the impact of local steps T on communication efficiency in FA-HMC with SG. Fixing leapfrog step $K = 100$ and stepsize $\eta = 0.01$, we run FA-HMC with T ranging from 1 to 100. For each run, we collect one sample after a fixed number of communication rounds and calculate the MEs in an online manner. Then, we report the required rounds R_ϵ to achieve $\mathrm{ME} = \epsilon$ under different settings and present the results in Figure 1(c). As we can see, the optimal local step T is 70; setting T too large or too small leads to more communication costs. We also notice that under the optimal local step, a smaller ϵ leads to more improvement on the communication cost R_ϵ compared

with the result of T = 100. Moreover, compared with the communication efficiency of T = 1, the optimal *communication efficiency improves by more than 65 times when ϵ is around 0.101.*

Furthermore, we reduce the dimension d to 10 in the simulated data and run FA-HMC as well as FA-LD with different stepsizes η on this new dataset fixing local step T = 10. Apart from the dimension d, the other settings are the same as those in the experiments for Figure 1(b). To list a few, stochastic gradients are adopted, and we choose the leapfrog step $K = \lfloor \pi/(3\eta) \rfloor$ when $\eta \leq 0.01$ and tune $K > 1$ when $\eta \geq 0.02$ for FA-HMC. The curves of the MEs against η are plotted in Figure 1(d). We observe that the general pattern in Figure 1(d) is similar to Figure 1(b). The optimal performance of FA-HMC is better than that of FA-LD, and the performance gap is larger for smaller step sizes. Comparing Figure 1(d) with Figure 1(b), we comment that FA-HMC is more advantageous under high-dimensional settings. This observation is consistent with our theoretical results that FA-HMC has a better convergence rate in terms of the dimension.

5.2 Simulation: Dimension vs Communication for FA-HMC

In this experiment, under the suggested setting of learning rate in Proposition 4.3, we examine the relationship between communication rounds t_ϵ / T required to achieve a $\mathcal{W}_2(\theta_{t_\epsilon}, \theta^\pi)^2 < 0.1$ and dimension d.

To obtain an accurate computation of the $\mathcal{W}_2(\theta_{t_\epsilon}, \theta^\pi)$, we consider a distributed heterogeneous Gaussian model where the $\mathcal{W}_2(\theta_{t_\epsilon}, \theta^\pi)$ can be explicitly calculated in terms of the population mean and variance of the parameter. Specifically, we assume that the posterior distribution of half of the local nodes' parameters is $N(20\mathbf{1}_d, \mathbb{I}_d)$, and for the other half, it is $N(\mathbf{1}_d, 2\mathbb{I}_d)$. One can check that the overall posterior distribution of parameters is $N(16.2\mathbf{1}_d, 1.6\mathbb{I}_d)$. We use leapfrog steps K = 5, local steps T = 10, and a learning rate $\eta = 0.02/d^{1/4}$. For different dimensions $d = 2, 50, 100, 150, ..., 950, 1000$, we

repeat the experiment $200 \cdot d(d-1)/2$ times and sample the parameter θ_t at the last iteration t for each time. The sampled parameters allow us to estimate the population mean and variance on the calculation of $\mathcal{W}_2(\theta_{t_\epsilon}, \theta^\pi)$.

The simulation results in Figure 2 suggest that the square of communication round $(t_\epsilon / T)^2$ is approximately proportional to dimension d. This aligns well with our theoretical discovery in Proposition 4.3, where under the suggested learning rate setting $t_\epsilon / T = O(\sqrt{d}\log(d/\epsilon^2)/\epsilon)$.

5.3 Application: Logistic Regression Model for FMNIST

In this section, we apply FA-HMC to train a logistic regression on the Fashion-MNIST dataset. The data points are randomly split into 10 subsets of equal size for $N = 10$ clients. We run FA-HMC under different settings of local step T and leapfrog step K with stochastic gradients that are calculated using a batch size of 1000 in each local device. In each run, one parameter sample is collected after a fixed number of communication rounds, and the predicted probabilities made by all the previously collected parameter samples are averaged to calculate four test statistics: prediction accuracy, Brier Score (BS) (Brier et al., 1950), Expected Calibration Error (ECE) (Guo et al., 2017), and Negative Log Likelihood (NLL) on the test dataset. We tune the step size η in each setting for the best test statistic. We conduct 5 independent runs in each setting and report the average results of these chains. The standard deviations of the results across multiple runs are displayed in Section L in the supplementary materials.

Specifically, to study the impact of leapfrog step K on the performance of FA-HMC, we fix local step $T = 50$, run FA-HMC with $K = 1$, 10, 50, and 100, and plot the curves of the calculated test statistics (accuracy, BS, ECE, and NLL) against communication rounds in Figure 3. As we can see, under the same budgets of communication and computation, FA-LD ($K = 1$) performs the worst in terms of BS, ECE, and NLL and the second worst in terms of accuracy, which shows the superiority of FA-HMC with $K > 1$ over FA-LD. Moreover, FA-HMC with $K = 50$ performs the best in terms of accuracy,

BS, and NLL and achieves a small ECE. In particular, the improvement on ECE and NLL over $K = 1$ can be as large as 26% and 2% respectively, indicating that the optimal choice of leapfrog step is around 50 in this setting.

To study the impact of local step T on the performance of FA-HMC, we fix leapfrog step $K = 10$, run FA-HMC with $T = 1$, 10, 20, 50, and 100, and plot the curves of the calculated test statistics (accuracy, BS, ECE, and NLL) against communication rounds in Figure 4. According to the figure, FA-HMC with $T = 1$ performs the worst in terms of all four statistics, which shows the necessity of multiple local updates in this setting. Besides, the optimal local step T differs with testing evaluation metrics, e.g., the optimal T is 50 in terms of BS, while the optimal T is 20 in terms of NLL.

5.4 Application: Neural Network Model for FMNIST

To further assess the performance of FA-HMC on non-convex problems, we apply FA-HMC and FA-LD to train a fully connected neural network with two hidden layers[2] and the ReLU activation function on the Fashion-MNIST dataset. Other settings of the experiments are the same as the logistic regression experiments in Section 5.3 except that only one chain is simulated in each case. We also calculate prediction accuracy, Brier Score (BS), and Expected Calibration Error (ECE) on the test dataset. The step size η is tuned in each setting for the best test statistic. Fixing local step $T = 50$ and choosing leapfrog step $K = 1$, 10, 50, and 100, the curves of the calculated test statistics against communication rounds are plotted in Figure 5(a), 5(b), and 5(c). As is shown in the figures, the optimal leapfrog step differs among different test statistics. For accuracy and ECE, the optimal $K = 10$ (i.e., FA-HMC notably outperforms FA-LD), while the optimal $K = 1$ for BS (i.e., FA-LD sightly outperforms FA-HMC). Fixing $K = 10$ and choosing $T = 1$, 10, and 50, the curves of the calculated test statistics against communication rounds are plotted in Figure 5(d), 5(e), and 5(f). We can see that the best local step is $T = 50$ and the worst local step is $T = 1$, indicating that the communication cost can be greatly reduced.

5.5 Application: Logistic Regression Model on KMNIST/CIFAR2

We also apply FA-HMC to train logistic regression on the Kuzushiji-MNIST (KM) (Clanuwat et al., 2018) and CIFAR10 dataset (Krizhevsky et al., 2009). Specifically, we only use the first two classes (airplane and automobile) of the CIFAR10 dataset in the experiments to simplify the problem and denote it by CF2. The data points in each dataset are randomly split into 10 subsets of equal size for $N = 10$ clients. We run FA-HMC under different settings of local step T and leapfrog step K with stochastic gradients that are calculated using a batch size of 1000 in each local device. As usual, we tune the step size η in each setting and report the best statistics: prediction accuracy (AC), Brier Score (BS), and Expected Calibration Error (ECE) on the test dataset. The choices of local step T and leapfrog step K are the same as those in Section 5.4.

The performance of FA-HMC using different leapfrog steps K is shown in Figure 6. We see that the optimal leapfrog step K varies with different test statistics and datasets, for example, the best K is 10 for AC and BS and the best K is larger than 10 for ECE on the CF2 dataset, and none of the experiments support $K = 1$ (i.e., FA-LD) as the optimal leapfrog step, showcasing the advantage of FA-HMC (i.e., $K > 1$) over FA-LD. We also study the impact of different local steps T as shown in Figure 7. We observe that except for the AC metric on CF2, federated learning with $T > 1$ outperforms the standard baseline $T = 1$ on the rest of the metrics on both the CF2 and KM datasets.

6 Conclusions and Future Work

In this paper, we develop a tight theoretical guarantee for FA-HMC and provide suggestions to speed up FA-HMC. Through experimentation, we demonstrate that FA-HMC outperforms FA-LD. We believe that FA-HMC potentially captures the similarities between local nodes, giving it an advantage over FA-LD. For future directions, it would be interesting to explore if further improvements can be achieved by addressing heterogeneity in local leapfrog steps. Note that for second-order methods, one would need to tackle heterogeneity both on local positions and local momentum parameters. For example, motivated by Karimireddy et al. (2020), suppose at t_0-th iteration

(communication round), each local device obtain θ_{t_0+T} and $\nabla f(\theta_{t_0}) := \sum_{c=1}^{N} w_c \nabla f^{(c)}(\theta_{t_0})$. Then each local device with local loss function $f^{(c)}$ is going to perform the following update for $k = 0,1,\ldots,K-1, t = t_0+T+1,\ldots,t_0+2T$

$$\theta_{t,k+1}^{(c)} = \theta_{t,k}^{(c)} + \eta_t p_{t,k}^{(c)} - \frac{\eta_t^2}{2}\left(\nabla f^{(c)}(\theta_{t,k}^{(c)}) - \nabla f^{(c)}(\theta_{t_0}) + \nabla f(\theta_{t_0})\right),$$

$$p_{t,k+1}^{(c)} = p_{t,k}^{(c)} - \frac{\eta_t}{2}\left(\nabla f^{(c)}(\theta_{t,k}^{(c)}) + \nabla f^{(c)}(\theta_{t,k+1}^{(c)}) - 2\nabla f^{(c)}(\theta_{t_0}) + 2\nabla f(\theta_{t_0})\right).$$

The complete version is deferred to Algorithm B.2 in the Supplementary Material.

Another direction we are working on is to consider the privacy guarantee of the sampling algorithms and compare them with optimization algorithms.

It would also be interesting to examine the above directions and the application of underdamped Langevin Monte Carlo algorithms in federated learning as future research.

Acknowledgement

Dr. Lin gratefully acknowledges the support of the National Science Foundation (DMS-2053746, DMS-2134209, ECCS-2029241, and OAC-2311848), and U.S. Department of Energy (DOE) Office of Science Advanced Scientific Computing Research program DE-SC0023161, and DOE - Fusion Energy Science, under grant number: DE-SC0024583.

Conflict of interest statement

The authors of this paper report that there are no competing interests to declare.

Notes

[1] As $d \to \infty$, we say $f = O(g) iff \leq Cg$ for some constant C, and say $f = \tilde{O}(g) for C$ being a polynomial of $\log(d)$.

[2] The widths of two layers are 512 times input dimension and 512 times the number of classification labels respectively.

References

Akyildiz, Ö. D. and Sabanis, S. (2020), "Nonasymptotic Analysis of Stochastic Gradient Hamiltonian Monte Carlo under Local Conditions for Nonconvex Optimization," *arXiv preprint arXiv:2002.05465* .

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečnỳ, J., Mazzocchi, S., and McMahan, B. (2019), "Towards Federated Learning at Scale: System design," *Proceedings of Machine Learning and Systems* , 1, 374–388.

Brier, G. W. et al. (1950), "Verification of Forecasts Expressed in Terms of Probability," *Monthly weather review* , 78, 1–3.

Chau, H. N. and Rasonyi, M. (2022), "Stochastic gradient Hamiltonian Monte Carlo for non-convex learning," *Stochastic Processes and their Applications* , 149, 341–368.

Chen, M., Yang, Z., Saad, W., Yin, C., Poor, H. V., and Cui, S. (2020a), "A Joint Learning and Communications Framework for Federated Learning over Wireless Networks," *IEEE Transactions on Wireless Communications* , 20, 269–283.

Chen, T., Fox, E., and Guestrin, C. (2014), "Stochastic Gradient Hamiltonian Monte Carlo," in *International Conference on Machine Learning (ICML)* .

Chen, Y., Dwivedi, R., Wainwright, M. J., and Yu, B. (2020b), "Fast Mixing of Metropolized Hamiltonian Monte Carlo: Benefits of Multi-step Gradients." *Journal of Machine Learning Research* , 21, 92–1.

Chen, Z. and Vempala, S. S. (2022), "Optimal convergence rate of hamiltonian monte carlo for strongly logconcave distributions," *Theory of Computing* , 18, 1–18.

Cheng, X., Chatterji, N. S., Bartlett, P. L., and Jordan, M. I. (2018), "Underdamped Langevin MCMC: A Non-asymptotic Analysis," in *Conference on Learning Theory (COLT)* .

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. (2018), "Deep learning for classical japanese literature," *arXiv preprint arXiv:1812.01718* .

Dalalyan, A. S. (2017), "Theoretical Guarantees for Approximate Sampling from Smooth and Log-concave Densities," *Journal of the Royal Statistical Society: Series B* , 79, 651–676.

Dalalyan, A. S. and Karagulyan, A. (2019), "User-friendly Guarantees for the Langevin Monte Carlo with Inaccurate Gradient," *Stochastic Processes and their Applications* , 129, 5278–5311.

Deng, W., Ma, Y.-A., Song, Z., Zhang, Q., and Lin, G. (2021), "On Convergence of Federated Averaging Langevin Dynamics," *arXiv preprint arXiv:2112.05120* .

Duane, S., Kennedy, A., Pendleton, B., and Roweth, D. (1987), "Hybrid Monte Carlo," *Physics Letters B* , 195, 216–222.

Erdogdu, M. A. and Hosseinzadeh, R. (2021), "On the Convergence of Langevin Monte Carlo: the Interplay between Tail Growth and Smoothness," in *Proc. of Conference on Learning Theory (COLT)* .

Faes, C., Ormerod, J. T., and Wand, M. P. (2011), "Variational Bayesian inference for parametric and nonparametric regression with missing data," *Journal of the American Statistical Association* , 106, 959–971.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (1995), *Bayesian data analysis* , Chapman and Hall/CRC.

Girolami, M. and Calderhead, B. (2011), "Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* , 73, 123–214.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017), “On Calibration of Modern Neural Networks,” in *International Conference on Machine Learning (ICML)* .

Gürbüzbalaban, M., Gao, X., Hu, Y., and Zhu, L. (2021), “Decentralized Stochastic Gradient Langevin Dynamics and Hamiltonian Monte Carlo,” *Journal of Machine Learning Research* , 22, 1–69.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. (2020), “ Scaffold: Stochastic Controlled Averaging for Federated Learning,” in *International Conference on Machine Learning (ICML)* .

Konečnỳ, J., McMahan, B., and Ramage, D. (2015), “Federated Optimization: Distributed optimization Beyond the Datacenter,” *arXiv preprint arXiv:1511.03575* .

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtarik, P., Suresh, A. T., and Bacon, D. (2016), “Federated Learning: Strategies for Improving Communication Efficiency,” in *NIPS Workshop on Private Multi-Party Machine Learning* , URL https://arxiv.org/abs/1610.05492.

Krizhevsky, A., Hinton, G., et al. (2009), “Learning multiple layers of features from tiny images,” Technical report, University of Toronto, ON, Canada.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2020a), “Federated Learning: Challenges, Methods, and Future Directions,” *IEEE Signal Processing Magazine* , 37, 50–60.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smithy, V. (2020b), “ Federated Optimization in Heterogeneous Networks,” in *Proceedings of the 3rd MLSys Conference* .

Li, X., Gu, Y., Dvornek, N., Staib, L. H., Ventola, P., and Duncan, J. S. (2020c), “Multi-site fMRI Analysis using Privacy-preserving Federated Learning and Domain Adaptation: ABIDE Results,” *Medical Image Analysis* , 65, 101765.

Ma, Y.-A., Chen, T., and Fox, E. (2015), "A Complete Recipe for Stochastic Gradient MCMC," in *Advances in Neural Information Processing Systems (NeurIPS)* , volume 28.

Mangoubi, O. and Vishnoi, N. K. (2018), "Dimensionally Tight Running Time Bounds for Second-order Hamiltonian Monte Carlo," in *Advances in Neural Information Processing Systems (NeurIPS)* .

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017), "Communication-efficient Learning of Deep Networks from Decentralized Data," in *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)* .

Neal, R. M. (2012), "MCMC Using Hamiltonian Dynamics," in *Handbook of Markov Chain Monte Carlo* , volume 54, Chapman and Hall/CRC, 113–162.

Nemeth, C. and Fearnhead, P. (2021), "Stochastic Gradient Markov Chain Monte Carlo," *Journal of the American Statistical Association* , 116, 433–450.

Plassier, V., Moulines, E., and Durmus, A. (2023), "Federated averaging langevin dynamics: Toward a unified theory and new algorithms," in *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)* .

Raginsky, M., Rakhlin, A., and Telgarsky, M. (2017), "Non-convex Learning via Stochastic Gradient Langevin Dynamics: a Nonasymptotic Analysis," in *Proc. of Conference on Learning Theory (COLT)* .

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., and Maier-Hein, K. (2020), "The Future of Digital Health with Federated Learning," *NPJ digital medicine* , 3, 1–7.

Tran, N. H., Bao, W., Zomaya, A., Nguyen, M. N., and Hong, C. S. (2019), "Federated Learning over Wireless Networks: Optimization Model Design and Analysis," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications* , IEEE.

Vono, M., Plassier, V., Durmus, A., Dieuleveut, A., and Éric Moulines (2022), “QLSD: Quantised Langevin Stochastic Dynamics for Bayesian Federated Learning,” in *Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS)* .

Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., and Khazaeni, Y. (2020), “ Federated Learning with Matched Averaging,” in *International Conference on Learning Representations (ICLR)* .

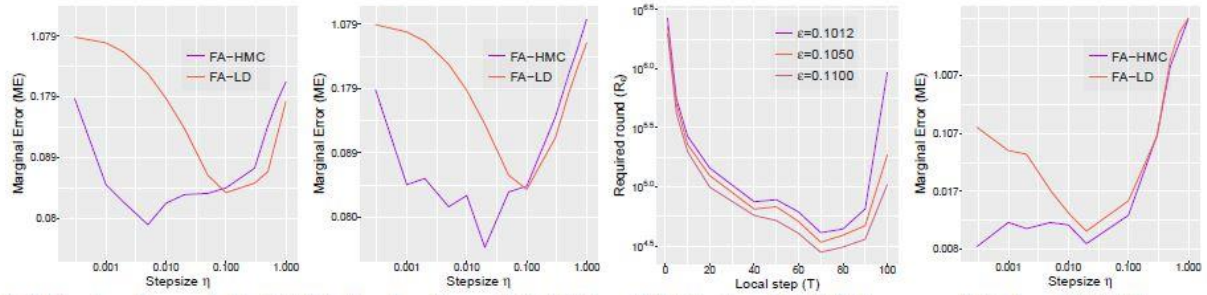
Welling, M. and Teh, Y. W. (2011), “Bayesian Learning via Stochastic Gradient Langevin Dynamics,” in *International Conference on Machine Learning (ICML)* .

Zhu, L., Liu, Z., and Han, S. (2019), “Deep Leakage from Gradients,” in *Advances in Neural Information Processing Systems (NeurIPS)* , volume 32.

Zou, D. and Gu, Q. (2021), “On the Convergence of Hamiltonian Monte Carlo with Stochastic Gradients,” in *International Conference on Machine Learning (ICML)* .

Zou, D., Xu, P., and Gu, Q. (2019), “Stochastic Gradient Hamiltonian Monte Carlo Methods with Recursive Variance Reduction,” *Advances in Neural Information Processing Systems (NeurIPS)* , 32.

— (2021), “Faster Convergence of Stochastic Gradient Langevin Dynamics for Non-log-concave Sampling,” in *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)* .



(a) Study of η and K (G) (b) Study of η and K (SG) (c) Study of T (SG) (d) $d = 10$ (SG)

Fig. 1 Experimental results of FA-HMC and FA-LD on the simulated dataset using exact gradients (G) and stochastic gradients (SG). Dimension d = 1000 in Figure (a)-(c) and d = 10 in Figure (d).

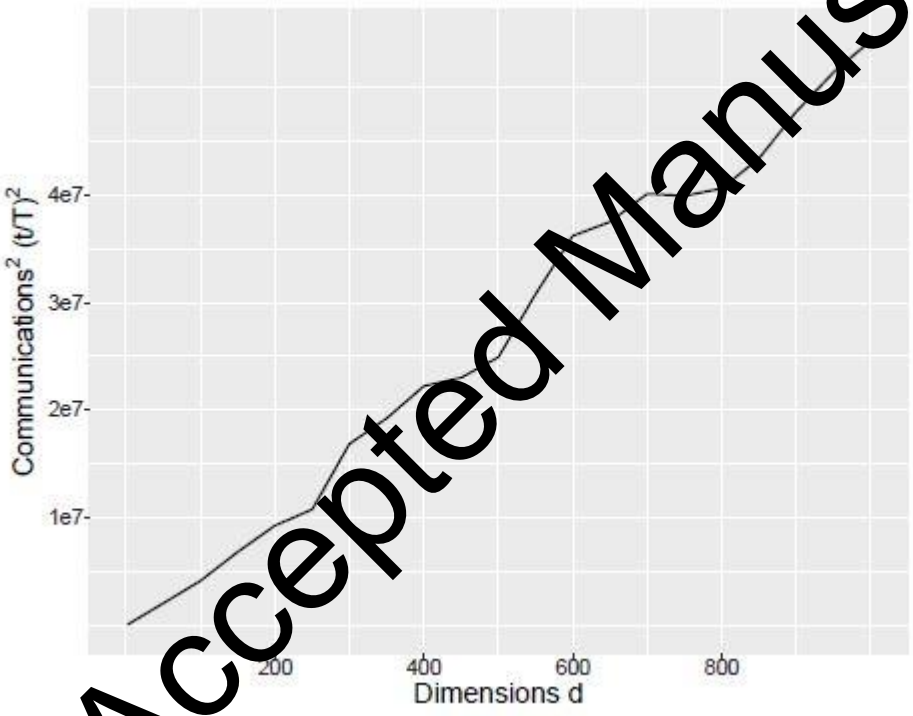


Fig. 2 Experimental results of FA-HMC to achieve $\mathcal{W}_2^2 < 0.1$ at different dimensions d.

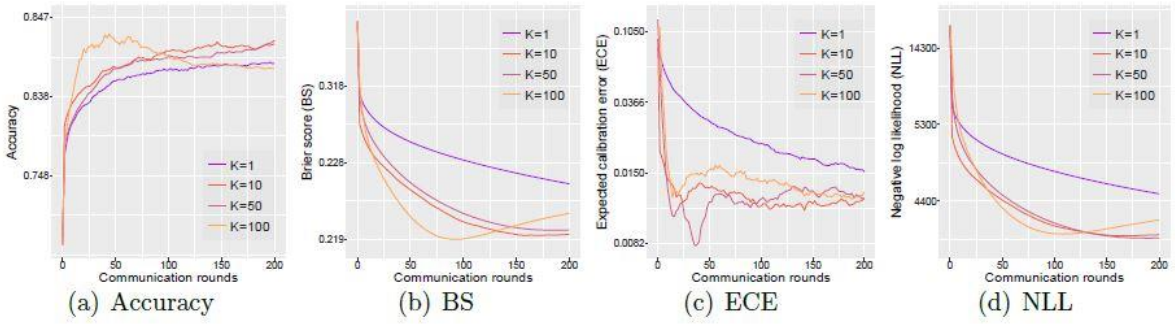


Fig. 3 The impact of leapfrog steps K on FA-HMC applied on the Fashion-MNIST dataset.

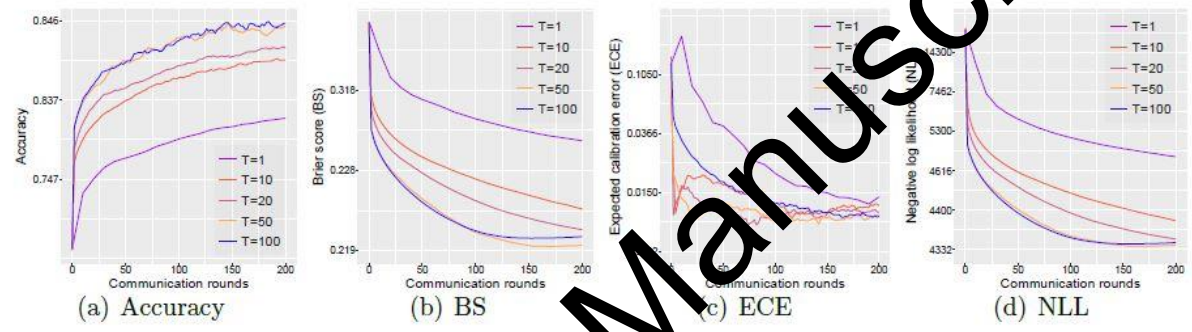


Fig. 4 The impact of local steps T on FA-HMC applied on the Fashion-MNIST dataset.

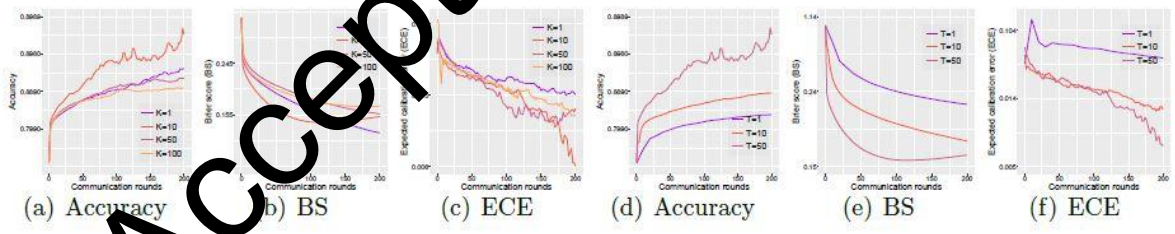


Fig. 5 The impact of leapfrog step K and local step T on FA-HMC applied to train a two-hidden-layer neural network on the Fashion-MNIST datasets.

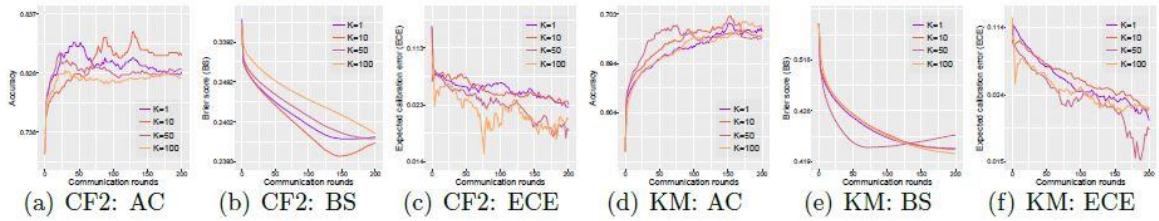


Fig. 6 The impact of leapfrog step K on FA-HMC applied on the CIFAR2 and KMNIST datasets.

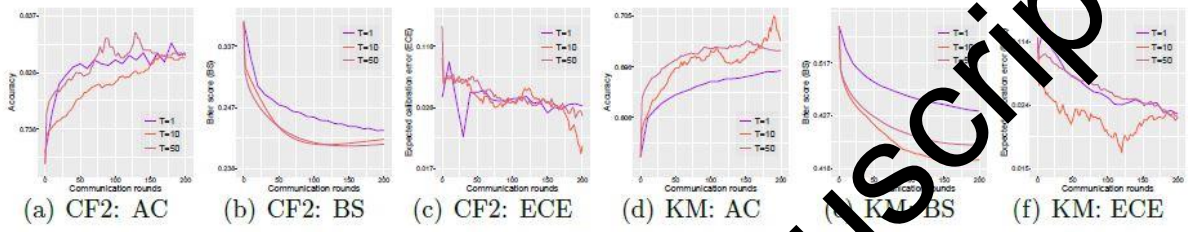


Fig. 7 The impact of local step T on FA-HMC applied on the CIFAR2 and KMNIST datasets.