

On the Stability Analysis of Open Federated Learning Systems

Youbang Sun¹ Heshan Fernando² Tianyi Chen² Shahin Shahrampour¹

Abstract—We consider the open federated learning (FL) systems, where clients may join and/or leave the system during the FL process. Given the variability of the number of present clients, convergence to a fixed model cannot be guaranteed in open systems. Instead, we resort to a new performance metric that we term the stability of open FL systems, which quantifies the magnitude of the learned model in open systems. Under the assumption that local clients' functions are strongly convex and smooth, we theoretically quantify the radius of stability for two FL algorithms, namely local SGD and local Adam. We observe that this radius relies on several key parameters, including the function condition number as well as the variance of the stochastic gradient. Our theoretical results are further verified by numerical simulations on synthetic data.

I. INTRODUCTION

Federated learning (FL) [1] is a machine learning setup where a group of clients work cooperatively to learn a statistical model. The learning process is coordinated by a central server which facilitates the exchange of model updates. FL algorithms enjoy the benefits of model sharing among clients while preserving data privacy, and they also reduce the number of communications without making too much sacrifice on the performance [2]. In a canonical FL algorithm, the central server broadcasts the initial model to all clients, and then, each client performs several steps of local updates before sending the model to the server. Once receiving models from a subset of clients, the server performs a global aggregation, typically in the form of average. The aggregated model will then be broadcasted again to all clients, and the algorithm continues.

FL emerges as a new paradigm to address the following four challenges in distributed machine learning [1], [3]. The first concern is *communication efficiency*, where FL algorithms often reduce the communication frequency in distributed architectures [4]. The second problem is that the local objective functions are different across clients, which is a consequence of having different data-sets (and optimal solutions) in respective clients. This challenge is termed as *statistical heterogeneity* [5]. The third challenge is *systems heterogeneity*, which accounts for the devices (rather than data) varying from one client to another, including

differences in processing speeds, communication bandwidth, and reliability [6]. Lastly, *privacy* is another concern in FL [7], which can be partially mitigated by preventing from directly transmitting local data to the central server.

In this work, we consider a new FL setting that we term the open FL system, where clients may join and/or leave the system during the learning process. More generally, the local clients' functions may change over time. Open FL systems are common in real-life scenarios. For example, a node failure in a large distributed system is almost inevitable, which translates to clients leaving the open FL system. In social platforms, users constantly join and leave. Also, it is possible that a present user might not be willing to share its model at a certain time. Therefore, addressing open systems is critical for FL architectures. There are recent works such as [8], [9], [10] that consider the client *drop-out*. However, these works mostly emphasize the partial cooperation of the clients, and the clients are still considered to be within the system. We view an open FL system through a different lens, where at certain times (when the system is still operating), new clients may join and old clients may leave the system; see an illustration in Fig. 1.

In open FL systems, due to the possible change in local objective functions for various clients, the global objective function also varies over time. As a result, analyzing the convergence to an exact statistical model (i.e., fixed optimization solution) is not possible for this type of systems. Additionally, when a client joins or leaves the system, the network size (i.e., number of clients) increases or decreases, respectively, which poses another challenge. In order to address this issue, we focus on the *stability* rather than the convergence of the open FL system. The concept of stability is precisely defined in Section II-D, which quantifies the magnitude of the learned model in open systems.

Our contributions are as follows

- **[Stability as a new FL metric]** We provide a formal definition of stability for open FL systems. Then, we provide analysis on the global stability of the open FL system under the assumption that the local optimization methods are stable.
- **[Stability of two FL algorithms]** We focus on two most common optimization methods used for FL (local SGD and local Adam) and provide theoretical guarantee on their stability, respectively.
- **[Empirical verification of stability]** Experiments on both synthetic data as well as real world datasets are conducted, and our empirical results further verify our theoretical findings.

The work of Y. Sun and S. Shahrampour was supported by NSF ECCS-2136206 Award. The work of H. Fernando and T. Chen was supported by National Science Foundation CAREER Award 2047177.

¹Youbang Sun and Shahin Shahrampour are with the Department of Mechanical and Industrial Engineering, Northeastern University, Boston, Massachusetts, 02115, USA {sun.youb, s.shahrampour}@northeastern.edu

²Heshan Fernando and Tianyi Chen are with Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, New York, 12180, USA {fernash, chent18}@rpi.edu

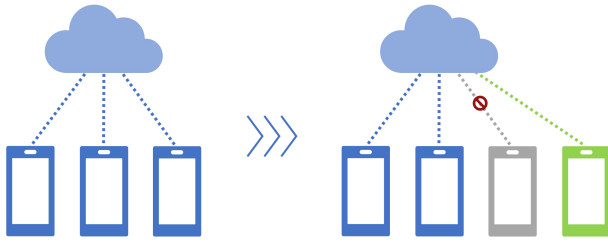


Fig. 1: An illustration of the open FL system

A. Related Work

We provide the literature review on several aspects of FL and distributed optimization.

Federated learning. Federated averaging (FedAvg) proposed by [1] was the first algorithm that introduced the federated learning setup. Since then, many variants have been proposed for a variety of motivations. FedAvg can be simplified to LocalSGD proposed by [11], which has been thoroughly analyzed in recent works ([12], [13], [14], [2]). Recently [15] established linear convergence while considering client heterogeneity and sparse gradients. [5] considered a regularized objective function in order to deal with statistical heterogeneity. [10] introduced SCAFFOLD, where the difference in clients are monitored to reduce the variance. [16] added momentum to the local clients and obtained better convergence results with the cost of extra communication. [17], [18] introduced adaptivity in local clients. A different approach to add adaptivity and acceleration to FedAvg was proposed by [19], [20], where the adaptivity is introduced in the server instead of clients. Apart from adaptive optimization methods, the communication can also be considered as adaptive, as shown in [21]. We refer to [22] for a more general overview of the literature.

Open multi-agent networks. In this literature, clients are often referred to as *agents*. The behavior of agents joining and leaving the system has long been studied by the community of distributed computing. [23] proposed FIRE and [24] proposed THOMAS, both addressing this mechanism in computations. Due to the recent resurgence of distributed optimization motivated by machine learning, a new line of work focusing on the dynamics of the system has been proposed. [25] provided analysis for open systems, and [26] provided further analysis for distributed gradient descent under more general network assumptions. The open system setup has also been discussed for algorithms such as online optimization [27] and dual averaging [28]. To the best of our knowledge, however, open FL systems have not been addressed.

Dynamical system stability. The stability of a dynamical system is an important topic in control and optimization. A very well-known notion of stability is the Lyapunov stability [29]. [30] introduced the notion of Lyapunov stability in the m -th mean for stochastic systems, which is akin to our definition of stability in Section II-D. Similar stability notions have also been discussed recently under different

setups. [31] and [32] considered the second-moment stability of discrete-time linear systems. The Markov jump system has also been discussed in [33]. However, all of these works touch upon the stability of the system dynamics rather than the stability of a FL architecture.

Another well-known line of work includes [34]. These works consider the stability of optimization methods in the sense of [35], which is closely related to the algorithm generalization. These works are not directly related to the notion of stability in this paper, and we only provide this remark to note the difference.

II. PROBLEM FORMULATION

Notation: We denote by $[n]$ the set $\{1, 2, 3, \dots, n\}$ for any integer n . We use $\langle x, y \rangle$ to denote the standard inner product between x and y , and $\|x\| = \sqrt{\langle x, x \rangle}$ is the Euclidean norm of vector x . We use $\text{col}\{v_1, \dots, v_n\}$ to denote the vector that stacks all vectors v_i for $i \in [n]$. We use $\mathbf{1}_d$ to denote a d -dimensional vector consists of ones. We use $\text{diag}\{a_1, \dots, a_n\}$ to represent an $n \times n$ diagonal matrix that has the scalar a_i in its i -th diagonal element. We denote the k -th element of a sequence $\{x^{(1)}, x^{(2)}, x^{(3)}, \dots\}$ by $x^{(k)}$. We write $\mathbf{B}(x, r) = \{y : \|y - x\| \leq r\}$ to denote a ball of radius r centered at x . The i -th element of a vector v is denoted as $[v]_i$.

A. Federated Learning Setup

In the canonical FL setup, we have a set of N clients and one central server that together perform a distributed optimization. The ultimate goal for the entire FL architecture is to minimize the following global objective function

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad F(x) = \sum_{i=1}^N f_i(x) = \sum_{i=1}^N \mathbb{E}[f_i(x, \xi_i)], \quad (1)$$

which outputs a global statistical model learned by the entire network. However, consistent with privacy-preserving aspect of FL, each client $i \in [N]$ is associated with the local objective function $f_i(x) : \mathbb{R}^d \rightarrow \mathbb{R}$ that encapsulates its own data ξ_i . Therefore, clients perform local updates and every once in a while communicate with the central server, which performs averaging among local models. We classify the iterations into two separate categories, namely, the communication iterations and the local update iterations, where during the communication iterations, the clients do not perform any local update.

B. Open Federated Learning Architecture

In this work, we are concerned with *open* FL systems, where clients may join or leave the system, or more generally, the local functions for clients may slightly change over time. A common real-life example is a social platform where users become online/offline frequently with a user base that evolves over time. We are inspired to consider this approach to FL architectures following the work of [25] on open multi-agent systems. We note that when all local agents in the FL system perform gradient descent and communicate

every round, our setup reduces to the open system distributed gradient descent formulation, which is addressed in [26].

In order to adapt FL algorithms to the open system setting, we need to make some adjustments to the FedAvg algorithm. If a client joins the system in between two communications, it will not be initialized since the server might not be available for broadcasting. When the next communication happens, the FedAvg algorithm does not consider the newly joined client as an *eligible* client, and this new client will only start its local update after receiving broadcasting from the server in the coming communication round. For the ease of analysis, we assume that the availability of agents (i.e., joining/leaving the system) is modeled with i.i.d. random variables, and there is at least one eligible client available for every communication. However, this i.i.d. assumption is not necessary.

C. Local Optimization Methods

Recent works have studied many forms of FL; one variation of the original FedAvg algorithm is to change the local optimizer from gradient descent to other optimization methods. Our main focus is to study open FL systems, and to that end, we consider two standard local optimization methods, namely stochastic gradient descent (SGD) and Adam. SGD is the workhorse of training for many machine learning models, commonly embedded into FL architecture for large-scale data [1]. In SGD, the local clients perform the following update

$$x_i^{(k+1)} = x_i^{(k)} - \eta \nabla f_i(x_i^{(k)}, \xi_i^{(k)}), \quad (2)$$

where k is the iteration index, and $i \in [N]$ is the client index, respectively. On the other hand, Adam can exhibit a superior performance due to its adaptive nature without significantly increasing time/space costs [36]. In particular, for the Adam algorithm, at the k -th local iteration all local clients perform the following update

$$\begin{aligned} h_i^{(k+1)} &= \beta_1 h_i^{(k)} + (1 - \beta_1) \nabla f_i(x_i^{(k)}, \xi_i^{(k)}) \\ v_i^{(k+1)} &= \beta_2 \hat{v}_i^{(k)} + (1 - \beta_2) (\nabla f_i(x_i^{(k)}, \xi_i^{(k)}))^2 \\ x_i^{(k+1)} &= x_i^{(k)} - \frac{\eta}{(\epsilon I + \hat{V}_i^{(k+1)})^{1/2}} h_i^{(k+1)}, \end{aligned} \quad (3)$$

where $v_i^{(k)} \in \mathbb{R}^d$ is a vector that estimates the second moment of the stochastic gradient, and $\hat{V}_i^{(k)} \in \mathbb{R}^{d \times d}$ is a diagonal matrix, such that $\hat{V}_i^{(k+1)} := \text{diag}\{\hat{v}_i^{(k+1)}\}$ and $\hat{v}_i^{(k+1)} := \max\{v_i^{(k+1)}, \hat{v}_i^{(k)}\}$.

D. Stability in Open FL Systems

Given that in open FL architectures, clients may join and leave the system, exact convergence analysis is not possible. Here, we present the notion of stability in open FL systems with respect to the second-moment of optimization iterates.

Definition 1. For an iterative stochastic optimization algorithm $\mathcal{A}$, which produces a sequence of iterates $\{x^{(k)}\}$, we say that $\mathcal{A}$ is stable with respect to the second-moment if there exist a finite constant R and an iteration index k_0 , such that

Algorithm 1 Federated Learning in Open Systems

- 1: **client initialize:** Local optimizer with respective hyper-parameters, e.g. learning rate η .
- 2: **global initialize:** Initial model $x^{(0)}$, broadcast model to available clients, and mark all clients eligible.
- 3: **for** $k = 1, 2, \dots, K$, **do:**
- 4: **if** the iteration is local update iteration:
- 5: Eligible clients perform a local update.
- 6: **else:**
- 7: Server randomly selects a subset of clients from eligible clients and averages local parameters.
- 8: Broadcast parameters to all clients.
- 9: Mark all clients eligible.
- 10: **end if**
- 11: Current clients randomly leave the system and become ineligible.
- 12: New clients randomly enter the system; initialize local optimizer and set client state to ineligible; wait for broadcasted model parameters.
- 13: **end for**

$$\text{when } \|x^{(k)}\|^2 \leq R^2, \\ \text{then } \mathbb{E}[\|x^{(k+1)}\|^2 | x^{(k)}] \leq R^2 \text{ for all } k \geq k_0.$$

E. Technical Assumptions

In this paper, we impose the following assumptions on the local objective function.

Assumption 1. For any client $i \in [N]$ in the network, we assume that the local objective function $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ is μ -strongly convex and L -Lipschitz smooth.

The smoothness assumption is standard in FL systems. The strong convexity assumption is not typical, but we require that to analyze the dynamics of optimization iterates $x^{(k)}$. Without strong convexity, one cannot analyze the distance between $x^{(k)}$ and x^* given that the optimal solution might not be unique. Additionally, having both strong convexity and Lipschitz smoothness allows us to define the condition number $\kappa = \frac{L}{\mu} \geq 1$, which is crucial in the analysis of similar works [26]. Note that the strongly convex part of Assumption 1 implies that the global objective function $F(x)$ is also strongly convex, and hence, there exists a unique finite minimizer x^* .

Assumption 2. For any client $i \in [N]$ in the system, we denote the optimal solution of the local function $f_i(x)$ by x_i^* , and we assume that $x_i^* \in \mathbf{B}(0, r)$ with $f_i(x_i^*) = 0$.

Since all local functions satisfy Assumption 1, Assumption 2 holds without loss of generality by simply re-centering the functions. This assumption only simplifies our analysis.

III. MAIN RESULTS

A. System-wide Stability

FL systems generally consist of two components. First is the local update (e.g., SGD or Adam), and second is the federated averaging, wherein the central server aggregates the

information from a selection of local clients to construct an improved global model. In regards to the stability of open FL systems, our first result is to show that when the local update satisfies stability in the sense of Definition 1, the stability over the entire open FL system can be claimed consequently.

Proposition 1. *Given Assumptions 1-2, for a local optimizer $\mathcal{A}^{\text{loc}}$ defined in a Federated Learning system $\mathcal{A}^{\text{Fed}}$, if $\mathcal{A}^{\text{loc}}$ satisfies the stability in the sense of Definition 1, then the entire system $\mathcal{A}^{\text{Fed}}$ is also stable in the sense that*

$$\begin{aligned} & \text{when } \|x_i^{(k)}\|^2 \leq R^2 \text{ for all } i \in [N], \\ & \text{then } \mathbb{E}[\|x_i^{(k+1)}\|^2 | x_i^{(k)}] \leq R^2 \text{ for all } i \in [N], k \geq k_0. \end{aligned}$$

Proof. (i) If the current iteration k is a local update iteration, each of the clients performs a local update; therefore, stability follows from Definition 1.

(ii) If the current iteration k is a communication iteration, consider the averaging setup discussed in Section II-A. The system takes an average over a random subset of clients. Since the selection of clients is independent of client number, we can write the averaged result as $\bar{x}^{(k)} = \frac{\sum_{i=1}^N \delta_i x_i^{(k)}}{\sum_{i=1}^N \delta_i}$, where $\delta_1, \dots, \delta_N$ are non-negative i.i.d. random variables. We note that for the vanilla FedAvg algorithm, δ_i follows a Bernoulli distribution though δ_i can also follow other distributions, which corresponds to the central server taking a weighted average over clients. Then, after averaging and broadcasting to all agents, we have

$$\mathbb{E}[\|x_i^{(k+1)}\|^2] = \mathbb{E}\left[\left\|\frac{\sum_{i=1}^N \delta_i x_i^{(k)}}{\sum_{i=1}^N \delta_i}\right\|^2\right] \leq R^2. \quad (4)$$

Therefore, the system has stability on both local update iterations and communication iterations. $\square$

Although the analysis of stability in Proposition 1 was for system variable x_i (i.e., client i), we note that this also provides us with insights about objective error. From Assumption 2 we know that since the optimal solution $x_i^* \in \mathbf{B}(0, r)$, when $x_i^{(k)} \in \mathbf{B}(0, R)$, the distance $\|x_i^{(k)} - x_i^*\|$ is upper bounded. Therefore, the optimization error is also upper bounded due to Lipschitz smoothness from Assumption 1.

B. Open FL System Stability - SGD

We now consider the stability of FL systems under the SGD algorithm, which is the most common algorithm used in FL. We impose the following assumption on the stochastic gradient of the objective function.

Assumption 3. *The stochastic gradient is unbiased with bounded variance, i.e.*

$$\begin{aligned} \mathbb{E}[\nabla f_i(x_i^{(k)}, \xi_i^{(k)})] &= \nabla f_i(x_i^{(k)}) \\ \mathbb{E}[\|\nabla f_i(x_i^{(k)}, \xi_i^{(k)}) - \nabla f_i(x_i^{(k)})\|^2] &\leq \sigma^2. \end{aligned}$$

The assumption above is quite standard and essential for the analysis in many previous works on SGD under various settings, including [1], [9], [37] and more. In the following

theorem, we quantify the radius of stability for open FL systems using SGD as the local optimizer. We remark that since our analysis holds for any available client, we present the result with a generic function $f(x)$, instead of a client-specific function $f_i(x)$.

Theorem 2. *For any function $f(x)$ that satisfies the Assumption 1 with the optimal solution $x^* \in \mathbf{B}(0, r)$, if the SGD algorithm runs with a learning rate $\eta \in (0, \frac{1}{L}]$, under Assumption 3 there exists a radius*

$$R = r + \sqrt{\max(3, \kappa)} \sqrt{2r^2 + \frac{\sigma^2}{L^2}},$$

for the FL system to have stability with respect to the second moment defined in Definition 1.

The complete proof of this theorem is given in [38]. We can see that the radius depends on κ and σ : if the stochastic gradient variance and/or the condition number increase, the radius also increases, which intuitively makes sense. For a special case where $\sigma = 0$, the FL algorithm will perform GD instead of SGD, and the stability results can be considered as a special case from [26], in a federated setup (i.e., centralized).

C. Open FL System Stability - Adam

We now quantify the radius of stability for open FL systems using Adam as the local optimizer. Similar to SGD, we first impose an assumption on the stochastic gradient used in the Adam algorithm, which is also commonly used in previous related works such as [18].

Assumption 4. *The stochastic gradient is unbiased and bounded, i.e.*

$$\begin{aligned} \mathbb{E}[\nabla f_i(x_i^{(k)}, \xi_i^{(k)})] &= \nabla f_i(x_i^{(k)}) \\ \|\nabla f_i(x_i^{(k)}, \xi_i^{(k)})\| &\leq \sigma. \end{aligned}$$

Again, since our analysis holds for any available client, we present the result with a generic function $f(x)$, instead of a client-specific function $f_i(x)$.

Theorem 3. *For any function $f(x)$ that satisfies Assumption 1 with the optimal solution $x^* \in \mathbf{B}(0, r)$, if the Adam algorithm runs with learning rate $\eta \in (0, \frac{1}{L}]$, under Assumption 4 there exists a finite radius*

$$R = C_5 r + \sqrt{\frac{1 + 3\kappa C_1}{(1 - \kappa C_1)^2} r^2 + \frac{2(C_2 + C_3 + C_1 C_4)}{\mu - LC_1}}, \quad (5)$$

with constants $C_1 - C_5$ for the FL system to have stability with respect to the second moment defined in Definition 1.

We refer to [38] for explicit forms of constants $C_1 - C_5$ and the proof of this theorem. We remark that σ should be small enough for the radius to exist. More discussion can be found in [38] (Section VI-B).

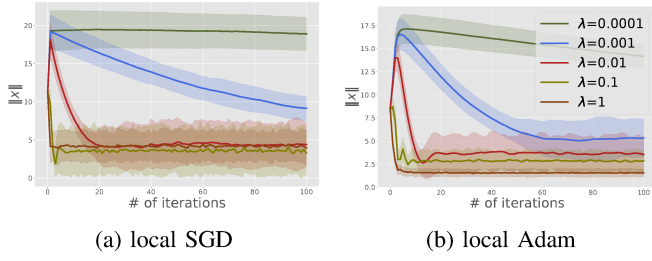


Fig. 2: Comparison of iterate norms for different λ .

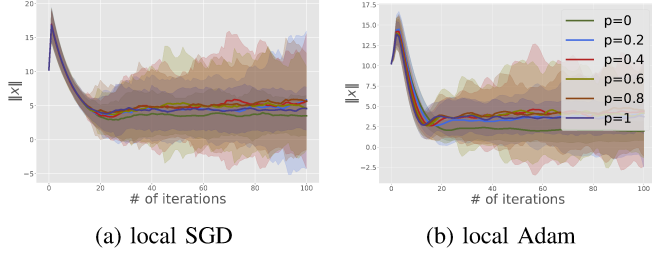


Fig. 3: Comparison of iterate norms for different p .

IV. NUMERICAL EXPERIMENTS

In order to verify our analysis, we conduct simulations using logistic regression on a synthetic dataset. Data are uniformly distributed across a group of dynamic clients during all tests. We implement local Adam and local SGD in the open FL setting, where clients perform local iterate updates independently, which are averaged periodically at a central server over all currently available clients. We specify the details of our simulation setting below.

Open FL system setup. We simulate a synthetic open FL system as follows. The open FL system consists of N clients in total, each containing m data. Initially, the system has N_0 clients that communicate with server. Each of the clients currently communicating with the server has a probability of leaving p before the next communication round, and similarly there is a probability p of a new client joining the system. Once a client leaves, it never joins the current system again. At the k -th communication round, the server communicates with N_k clients in the system. The newly joined clients will upload their local iterates to the server only after receiving the global iterate from the server in the previous communication iteration. For simplicity we allow only one client to leave and one client to join the system, at any given global iteration. We assume $N_k > 0$ for all $k = 0, 1, 2, \dots$, that is, the system will consist of at least one client at all times.

We plot the norm of the global iterate at each communication round under different choice of regularization λ of local loss function, probability p of clients joining/leaving the system, and in the synthetic setting the standard deviation of the data σ . We investigate the stability of the global iterate with changing λ , since λ determines the condition number κ of the loss function, which has a significant effect on R .

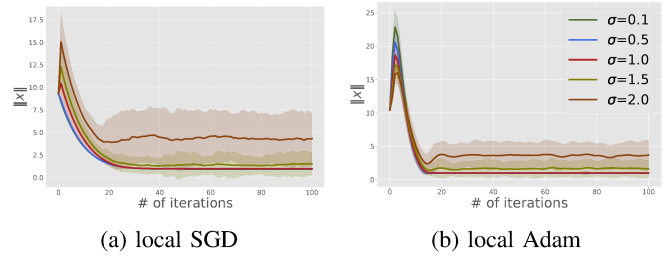


Fig. 4: Comparison of iterate norms for different σ .

A. Synthetic setting

In the synthetic data setting, the clients will have data of dimension $d = 100$ corresponding two label classes $y = -1$ and $y = 1$. The two label classes corresponds to two Gaussian distributions constructed as follows: the mean vector for each label class is sampled as $\mu = 2\text{Bern}(0.5) - 1$ element-wise, and the standard deviation of the data (element-wise) is σ . Each client will have $m = 100$ data.

We use the logistic loss as the local loss function at each client, given by

$$f_i(x) = \frac{\lambda}{2} \|x\|^2 + \frac{1}{m} \sum_{j=1}^m \log(1 + \exp(-y_{i,j} x^\top \xi_{i,j})),$$

where x is the local iterate, λ is the regularization parameter, $\{(\xi_{i,j}, y_{i,j})\}_{j=1}^m$ is the local dataset.

For the synthetic data setting, we use $N_0 = 10$ and $N = 1000$. We run 100 Monte-Carlo simulations for each choice of p , λ and σ , and plot the average results. The local optimization parameters for local SGD are set as $\eta = 1$, and for local Adam as $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-3}$ and $\eta = 0.1$. Fig. 2 shows the norm of the global iterate for different λ , where the remaining system parameters are set to $p = 1$ and $\sigma = 2$. Here we can observe that in both local Adam and local SGD cases, the norm of the global iterate converge to a stable value faster when the regularization coefficient is larger. Fig. 3 shows the norm of the global iterate for different p , where the remaining system parameters are set to $\lambda = 0.01$ and $\sigma = 2$. Here, it can be observed that when the system is highly dynamic, i.e. when p is closer to 1, the expectation of the stable value of the iterate is larger. The smallest global iterate norm corresponds to $p = 0$, when the system is static.

Fig. 4 shows the global iterate norm for different p , where the remaining system parameters are set to $p = 1$ and $\sigma = 2$. It can be seen that when the variance of the data is high, the global iterate norm is larger.

We further verify our results with extensive experiments in various settings including real-world data, where details of these experiments are provided in [38]. From the results for synthetic and real world data settings, the global iterate of the algorithms local SGD and local Adam maintains stability in systems with dynamic clients. The empirical results show how the norm of the global iterate changes with dynamic nature of the system and also with local loss function parameters such as λ and σ . The experiment results align

with the relationship between stability radius and function parameters given in Section III.

V. CONCLUSIONS

We introduced a novel formulation for FL, named open FL systems, where clients are able to join and/or leave the system during execution. We provided a formal definition for stability in open FL systems. We then provided theoretical analysis for stability for two commonly used optimization methods. The analytical results of this paper was further validated by numerical experiments on synthetic data.

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2017, pp. 1273–1282.
- [2] B. Woodworth, K. K. Patel, S. Stich, Z. Dai, B. Bullins, B. McMahan, O. Shamir, and N. Srebro, "Is local sgd better than minibatch sgd?" in *International Conference on Machine Learning (ICML)*. PMLR, 2020, pp. 10 334–10 343.
- [3] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.
- [4] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *Advances in Neural Information Processing Systems; Private Multi-Party Machine Learning Workshop*, 2016.
- [5] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.
- [6] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 7611–7623, 2020.
- [7] A. Bhowmick, J. Duchi, J. Freudiger, G. Kapoor, and R. Rogers, "Protection against reconstruction and its applications in private federated learning," *arXiv preprint arXiv:1812.00984*, 2018.
- [8] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith, "On the convergence of federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, vol. 3, p. 3, 2018.
- [9] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2020, pp. 2021–2031.
- [10] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *International Conference on Machine Learning (ICML)*. PMLR, 2020, pp. 5132–5143.
- [11] M. Zinkevich, M. Weimer, A. J. Smola, and L. Li, "Parallelized stochastic gradient descent," in *Neural Information Processing Systems (NIPS)*, vol. 4, no. 1, 2010, p. 4.
- [12] S. U. Stich, "Local sgd converges fast and communicates little," in *International Conference on Learning Representations (ICLR)*, no. CONF, 2019.
- [13] S. U. Stich and S. P. Karimireddy, "The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates," *The Journal of Machine Learning Research (JMLR)*, vol. 21, no. 1, pp. 9613–9648, 2020.
- [14] A. Khaled, K. Mishchenko, and P. Richtárik, "Tighter theory for local sgd on identical and heterogeneous data," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2020, pp. 4519–4529.
- [15] A. Mitra, R. Jaafar, G. J. Pappas, and H. Hassani, "Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 14 606–14 619, 2021.
- [16] H. Yu, R. Jin, and S. Yang, "On the linear speedup analysis of communication efficient momentum sgd for distributed non-convex optimization," in *International Conference on Machine Learning (ICML)*. PMLR, 2019, pp. 7184–7193.
- [17] C. Xie, O. Koyejo, I. Gupta, and H. Lin, "Local adaalter: Communication-efficient stochastic gradient descent with adaptive learning rates," *OPT2020: 12th Annual Workshop on Optimization for Machine Learning*, 2020.
- [18] J. Wang, Z. Xu, Z. Garrett, Z. Charles, L. Liu, and G. Joshi, "Local adaptivity in federated learning: Convergence and consistency," *International Conference on Machine Learning (ICML)*, 2021.
- [19] S. J. Reddi, Z. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konečný, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," in *International Conference on Learning Representations (ICLR)*.
- [20] J. Wang, V. Tantia, N. Ballas, and M. Rabbat, "Slowmo: Improving communication-efficient distributed sgd with slow momentum," in *International Conference on Learning Representations (ICLR)*.
- [21] T. Chen, Z. Guo, Y. Sun, and W. Yin, "Cada: Communication-adaptive distributed adam," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*. PMLR, 2021, pp. 613–621.
- [22] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.
- [23] T. D. Huynh, N. Jennings, and N. Shadbolt, "Fire: An integrated trust and reputation model for open multi-agent systems," *Journal of Autonomous Agents and Multi-Agent Systems*, vol. 13, pp. 18–22, 01 2004.
- [24] C. Carrascosa, A. Giret, V. Julian, M. Rebollo, E. Argente, and V. Botti, "Service oriented mas: an open architecture," in *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, 2009, pp. 1291–1292.
- [25] J. M. Hendrickx and S. Martin, "Open multi-agent systems: Gossiping with random arrivals and departures," in *IEEE Conference on Decision and Control (CDC)*, 2017, pp. 763–768.
- [26] J. M. Hendrickx and M. G. Rabbat, "Stability of decentralized gradient descent in open multi-agent systems," in *IEEE Conference on Decision and Control (CDC)*, 2020, pp. 4885–4890.
- [27] Y.-G. Hsieh, F. Iutzeler, J. Malick, and P. Mertikopoulos, "Multi-agent online optimization with delays: Asynchronicity, adaptivity, and optimism," *Journal of Machine Learning Research (JMLR)*, vol. 23, no. 78, pp. 1–49, 2022.
- [28] —, "Optimization in open networks via dual averaging," in *IEEE Conference on Decision and Control (CDC)*, 2021, pp. 514–520.
- [29] A. M. Lyapunov, "The general problem of the stability of motion," *International journal of control*, vol. 55, no. 3, pp. 531–534, 1992.
- [30] F. Kozin, "A survey of stability of stochastic systems," *Automatica*, vol. 5, no. 1, pp. 95–112, 1969.
- [31] Y. Hosoe and T. Hagiwara, "Equivalent stability notions, lyapunov inequality, and its application in discrete-time linear systems with stochastic dynamics determined by an iid process," *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4764–4771, 2019.
- [32] —, "On second-moment stability of discrete-time linear systems with general stochastic dynamics," *IEEE Transactions on Automatic Control*, 2021.
- [33] O. Costa and D. Z. Figueiredo, "Stochastic stability of jump discrete-time linear systems with markov chain in a general borel space," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 223–227, 2013.
- [34] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *International Conference on Machine Learning (ICML)*. PMLR, 2016, pp. 1225–1234.
- [35] O. Bousquet and A. Elisseeff, "Stability and generalization," *The Journal of Machine Learning Research (JMLR)*, vol. 2, pp. 499–526, 2002.
- [36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations (ICLR)*, 2015.
- [37] A. Spiridonoff, A. Olshevsky, and I. C. Paschalidis, "Local sgd with a communication overhead depending only on the number of workers," 2020.
- [38] Y. Sun, H. Fernando, T. Chen, and S. Shahrampour, "On the stability analysis of open federated learning systems," 2022. [Online]. Available: <https://arxiv.org/abs/2209.12307>
- [39] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.