

Spectrum Breathing: Protecting Over-the-Air Federated Learning Against Interference

Zhanwei Wang, Kaibin Huang, and Yonina C. Eldar

Abstract

Federated Learning (FL) is a widely embraced paradigm for distilling artificial intelligence from distributed mobile data. However, the deployment of FL in mobile networks can be compromised by exposure to interference from neighboring cells or jammers. Existing interference mitigation techniques require multi-cell cooperation or at least interference channel state information, which is expensive in practice. On the other hand, power control that treats interference as noise may not be effective due to limited power budgets, and also that this mechanism can trigger countermeasures by interference sources. As a practical approach for protecting FL against interference, we propose *Spectrum Breathing*, which cascades stochastic-gradient pruning and spread spectrum to suppress interference without bandwidth expansion. The cost is higher learning latency by exploiting the graceful degradation of learning speed due to pruning. We synchronize the two operations such that their levels are controlled by the same parameter, *Breathing Depth*. To optimally control the parameter, we develop a martingale-based approach to convergence analysis of Over-the-Air FL with spectrum breathing, termed AirBreathing FL. We show a performance tradeoff between gradient-pruning and interference-induced error as regulated by the breathing depth. Given receive SIR and model size, the optimization of the tradeoff yields two schemes for controlling the breathing depth that can be either fixed or adaptive to channels and the learning process. As shown by experiments, in scenarios where traditional Over-the-Air FL fails to converge in the presence of strong interference, AirBreathing FL with either fixed or adaptive breathing depth can ensure convergence where the adaptive scheme achieves close-to-ideal performance.

Index Terms

Over-the-Air federated learning, gradient pruning, spread spectrum, interference suppression.

Z. Wang and K. Huang are with the Department of Electrical and Electronic Engineering, The University of Hong Kong (HKU), Hong Kong (Email: {zhanweiw, huangkb}@eee.hku.hk). Y. C. Eldar is with the Faculty of Math and CS, Weizmann Institute of Science, Rehovot, Israel (Email: yonina.eldar@weizmann.ac.il). The corresponding author is K. Huang.

I. INTRODUCTION

A key operation of the *sixth-generation* (6G) mobile network is to distill intelligence from enormous mobile data at the network edge using distributed machine learning algorithms, resulting in an active area termed edge learning [1], [2]. The obtained *Artificial Intelligence* (AI) is expected to empower many *Internet-of-Things* (IoT) applications ranging from smart cities to auto-pilots to extended reality. *Federated Learning* (FL) is arguably the most popular edge-learning framework for its preservation of data ownership and being considered for the 6G standard [3], [4]. FL protects users' data privacy by distributing the learning task and requiring users to upload local model updates instead of raw data [5]. Among others, two key challenges stymieing the deployment of FL in a mobile network are 1) a communication bottleneck resulting from the transmission of high-dimensional model updates and 2) exposure to interference from neighboring cells, and jammers [2], [6]–[8]. To simultaneously tackle these two challenges, we propose a spectrum-efficient method for suppressing interference to FL in mobile networks, termed *Spectrum Breathing*.

The key operation of FL is for a server to upload local updates from devices, which are computed using local data, for aggregation to update the global model. To overcome the resultant communication bottleneck, previous works focus on designing task-oriented wireless techniques for FL with the aim to alleviate the effects of channel hostility on learning performance. Diversified approaches have been proposed including radio resource management [9], [10], power control [11], [12], and device scheduling [13], [14]. On the other hand, the direct approach to reduce communication overhead is to prune local model updates, namely local models or stochastic gradients, and furthermore adapt the pruning operation to wireless channels [15]–[17]. Instead of incurring unrecoverable distortion, gradient pruning can be translated via randomization into longer learning latency with small accuracy degradation [18].

Recently, a new class of techniques, termed *Over-the-Air FL* (AirFL), has emerged to address the scalability issue in multi-access by many devices under a constraint on radio resources [19]–[21]. Underpinning AirFL is the use of so-called *Over-the-Air Computing* (AirComp) to realize over-the-air aggregation of local updates by exploiting the waveform superposition property of a multi-access channel, and thereby enable simultaneous access [22]. Building on AirComp, the efficiency of AirFL can be enhanced by beamforming [20], gradient pruning [21], broadband transmission [19] and power control [12], [23], and even the use of *Intelligent Reflecting Surface*

(IRS) [24]. Different from traditional designs, such techniques aim to realize the required signal-magnitude alignment at the server to implement AirComp despite channel distortion. For instance, an IRS can help to overcome such distortion to suppress the alignment error [24]. Furthermore, the optimization of AirFL techniques enables them to be adapted to not only channel states but also learning operations (e.g., gradient statistics in the current round [12]). The effectiveness of AirFL and AirComp at large hinges on the use of uncoded linear analog modulation for transmission. This exposes AirFL to interference and gives rise to the challenge of how to make AirFL robust. Gradient pruning techniques mentioned earlier do not address this challenge as they merely reduce communication overhead without making any attempt at interference suppression. An alternative approach is to treat interference as noise and regulate it using existing power-control techniques for AirFL [25], [26]. This class of techniques' effectiveness in dealing with interference is limited for two reasons. First, interference power may be comparable with that of the signal if not larger and far exceeds the noise power. Second, unlike noise, an interference source (e.g., a neighboring access point or a jammer) is active and can react to the power control of a signal source in a way that renders it ineffective.

An additional line of work is to adapt the rich set of existing interference-mitigation techniques to suit AirFL [6], [7], [27]–[29]. Previous works share the common principle of relying on cooperation between interfering nodes to mitigate the effects of their mutual interference on the learning performance. This principle is materialized in diversified techniques for multi-cell AirFL systems, including spatial interference cancellation [6], signal-and-interference alignment into orthogonal signal sub-spaces [7], and cooperative power control and devices scheduling [28], [29]. However, their implementation requires accurate *Channel State Information* (CSI) of interference channels. Acquiring such information can incur extra overhead and latency due to inter-cell messaging in multi-cell systems, and is infeasible in scenarios where the interference sources are in other networks or jammers.

One classic interference mitigation technique, called spread spectrum, has no such limitations but has not been explored in the context of AirFL due to its low efficiency in spectrum utilization [30]. This technique can reduce interference power by a factor, called the *Processing Gain* denoted as G , if a narrowband signal is spread in the spectrum by G via scrambling using a *Pseudo-Noise* (PN) sequence at the transmitter and using the same sequence to reverse the operation, called despreading, at the receiver. These operations neither require multi-cell cooperation nor interference CSI. Its invention served the purpose of anti-jamming for secured

communication in World War II while its commercial success was due to the use for mitigating multi-user interference in a resultant multi-access scheme, termed *Code-Division Multi-Access* (CDMA), for 3G [30], [31].

Gradient pruning and spread spectrum are two well-known techniques. The novelty of the proposed spectrum breathing approach lies in their integration to cope with interference in an AirFL system under a bandwidth constraint. Specifically, deployed at each device, the technique cascades two operations before transmission – *random pruning* of local gradient, called spectrum contraction, and *spread spectrum* on the pruned gradient. Note that random pruning is more suitable for AirFL than the alternative of magnitude-based pruning [32] (also see discussion in Sec. III). The spectrum contraction and spreading are governed by parameter, call *breathing depth*. As mentioned, the former mainly results in lengthened learning latency; the latter suppresses interference power by the factor of breathing depth. As a result, AirFL can converge even in the presence of strong interference. In the iterative FL algorithm, the alternating spectrum contraction and spreading are analogous to human breathing, giving the technique its name. We optimally control the spectrum breathing parameter so as to maximize its performance gain.

The contributions of this paper are summarized as follows.

- **Convergence Analysis:** Adjusting the breathing depth provides a mechanism for controlling AirFL using spectrum breathing, termed AirBreathing FL. To facilitate optimal control, we analyze the learning convergence by extending an existing supermartingale-based approach to account for AirComp, spectrum breathing and fading. The derived results reveal a tradeoff as regulated by the breathing depth. Specifically, increasing the parameter has two opposite effects – one is to improve the successful convergence probability by interference suppression and the other is to decrease it due to more aggressive gradient pruning. This gives rise to the need of optimal control.
- **Control of Spectrum Breathing:** Using the preceding tradeoff, the optimization of breathing depth yields two schemes for controlling AirBreathing FL under given receive SIR and model size. First, without CSI and *Gradient State Information* (GSI) at the server, the parameter is fixed over rounds and its optimal value is derived in closed form. Second, when CSI and GSI are available as in [12], the optimal strategy is designed to be adapted to CSI and GSI.
- **Experimental Results:** The results from experiments on AirBreathing FL demonstrate satisfactory learning performance even in the cases with strong interference that could fail

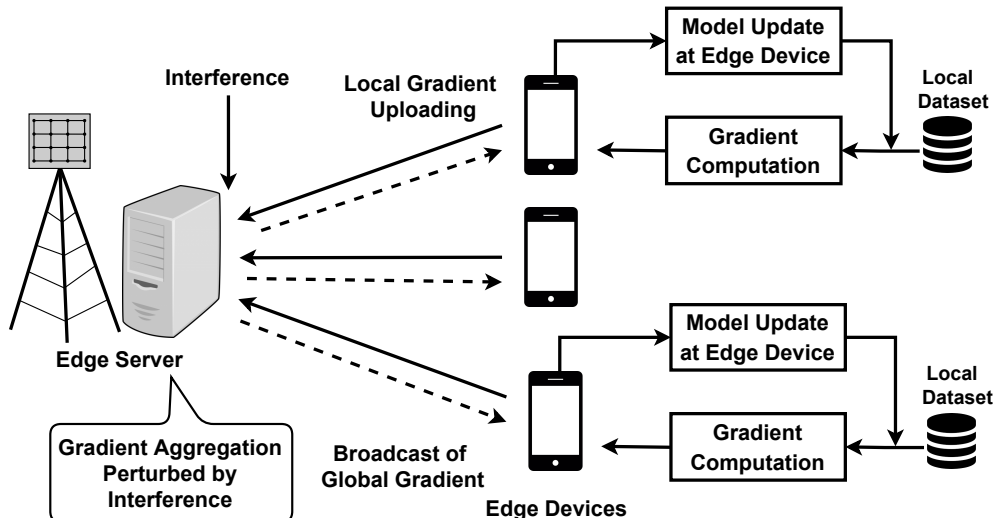


Fig. 1. System diagram of AirFL system perturbed by interference.

the learning task without spectrum breathing. Moreover, spectrum breathing with depth adaptation outperforms the case with a fixed breathing depth.

The remainder of this paper is organized as follows. Models and metrics are introduced in Sec. II. The effects of pruning on generic data and FL is demonstrated in Sec. III. Overview design of spectrum breathing is illustrated in Sec. IV. Convergence analysis and breathing depth optimization are analyzed in Sec. V and VI, respectively. Experimental results are provided in Sec. VII, followed by concluding remarks in Sec. VIII.

II. MODELS AND METRICS

We consider an AirFL system as illustrated in Fig. 1 that comprises one server and K devices. The learning process is perturbed by external interference (e.g. from other cells). The learning and communication models are described separately in the following sub-sections.

A. Learning Model

We first describe the FL process underpinning AirFL. Each device, say k , maintains its local dataset \mathcal{D}_k including $|\mathcal{D}_k|$ pairs of data sample \mathbf{x}_j and label y_j , denoted as $\{(\mathbf{x}_j, y_j)\} \in \mathcal{D}_k, j \in \{1, 2, \dots, |\mathcal{D}_k|\}$. The server coordinates K devices to optimize the weights of the global model $\mathbf{w} \in \mathbb{R}^D$ where D is the model size, under the criterion of minimizing a global loss defined as

$$F(\mathbf{w}) \triangleq \frac{1}{\sum_{k=1}^K |\mathcal{D}_k|} \sum_{k=1}^K \sum_{(\mathbf{x}_j, y_j) \in \mathcal{D}_k} f(\mathbf{w}, \mathbf{x}_j, y_j), \quad (1)$$

where $f(\mathbf{w}, \mathbf{x}_j, y_j)$ is the empirical loss function indicating the prediction error on model \mathbf{w} using a data sample (\mathbf{x}_j, y_j) . For simplicity, we denote $f(\mathbf{w}, \mathbf{x}_j, y_j)$ as $f_j(\mathbf{w})$. Distributed *Stochastic Gradient Descent* (SGD) is applied to minimize the global loss. Specifically, time is divided into N rounds with index $n \in \{0, 1, \dots, N - 1\}$. Considering round n , each device computes the gradient of the empirical loss function using a mini-batch of local dataset. The gradient of device k is given as

$$\mathbf{g}_k(n) = \frac{1}{|\mathcal{B}_k|} \sum_{j \in \mathcal{B}_k} \nabla f_j(\mathbf{w}(n)), \quad (2)$$

where $\mathcal{B}_k \subseteq \mathcal{D}_k$ is the selected mini-batch of \mathcal{D}_k , and ∇ represents the gradient operation. If local gradients can be reliably transmitted to the server, the global estimate of the gradient of the loss function in (1) is obtained as

$$\bar{\mathbf{g}}(n) = \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n). \quad (3)$$

Then, $\bar{\mathbf{g}}(n)$ is broadcast back to each device, by which the current model is updated via gradient descent:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta \cdot \bar{\mathbf{g}}(n), \quad (4)$$

where η denotes the learning rate. The distributed SGD is thus to iterate (2) and (4) until a convergence condition is met.

B. Communication Model

The uploading of the gradients using AirComp is perturbed by interference. To combat interference, each transmitted signal undergoes the operations of random pruning and spread spectrum. The two operations of AirBreathing FL are elaborated in Section IV. For the current exposition, some useful notation is introduced. Considering round n , the s -th element of the pruned local gradient transmitted by device k is scrambled by spread spectrum into a sequence, denoted $\tilde{\mathbf{g}}_{k,s}(n)$, with each element called a chip and the ℓ -th chip denoted as $[\tilde{\mathbf{g}}_{k,s}(n)]_\ell$.

Using the above notation, AirComp can be modelled as follows. Assume chip-level synchronization between devices using a standard technique such as Timing Advance [33]. The simultaneous transmission of the (s, ℓ) -th chips, i.e., $[\tilde{\mathbf{g}}_{k,s}(n)]_\ell$, enables AirComp to yield the corresponding received chip symbol, given as

$$[\mathbf{y}_s(n)]_\ell = \sum_{k=1}^K h_k(n) \sqrt{p_k(n)} [\tilde{\mathbf{g}}_{k,s}(n)]_\ell + [\mathbf{z}_s(n)]_\ell, \quad \forall (s, \ell), \quad (5)$$

where $h_k(n) \sim \mathcal{N}(0, 1)$ represents the k -th fading channel-gain that remains constant with round n , $p_k(n)$ the transmission power, $[\mathbf{z}_s(n)]_\ell \sim \mathcal{N}(0, P_I)$ is additive Gaussian interference. We consider the worst-case interference distribution that is Gaussian over chip duration [34]. Given an interference-limited system, channel noise is assumed negligible.

The downloading of the aggregated gradient can be implemented using digital or analog transmission [35]. Besides the availability of full bandwidth, transmission power at the server is much larger than that of the devices. Thus, gradient broadcasting is much more reliable than local gradient uploading such that the distortion to downlink is negligible.

C. Performance Metric

To quantify the distortion from gradient-pruning and interference, we introduce an AirComp error. Consider round n and active device set $\mathcal{K}(n) \subseteq \{1, 2, \dots, K\}$. After post-processing the received signal $\mathbf{y}(n)$ in (5), the output at the server is denoted as $\mathbf{y}'(n)$ specified in Sec. IV. Then AirComp error is defined as the *Mean Squared Error* (MSE) between $\mathbf{y}'(n)$ and its desired ground-truth, namely $\frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n)$, as

$$\text{MSE}(n) = \mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n) - \mathbf{y}'(n) \right\|^2 \right], \quad (6)$$

where the expectation is taken over the distributions of the transmitted symbols, interference, channel fading, and pruning pattern.

III. DISTORTION FROM PRUNING - GENERIC DATA VERSUS FEDERATED LEARNING

The spectrum-contraction operation of AirBreathing FL is realized via pruning local gradients in the FL process. Its effect on the system performance is fundamentally different from that of pruning a generic data sequence. This can be better understood by analyzing and comparing the two effects in the remainder of this section.

A. Generic Data Pruning

Consider gradient $\mathbf{g}_k(n) \in \mathbb{R}^D$, that is i.i.d. distributed with each element having zero-mean and variance of σ^2 . The desired ground truth $\frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n)$ is compressed by random pruning, namely a function that randomly replaces elements with zeros. Let $\mathbf{g}^{\text{sp}}(n)$ and γ denote the pruned gradient and the remaining fraction of nonzero elements, called the pruning ratio.

For generic data, the distortion from pruning is commonly quantified as the MSE between the pruned sequence and its ideal version:

$$\text{MSE}(n) = \mathbb{E} \left[\left\| \mathbf{g}^{\text{sp}}(n) - \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n) \right\|^2 \right] = (1 - \gamma) \frac{D\sigma^2}{|\mathcal{K}(n)|}. \quad (7)$$

One can see that the distortion increases linearly with the level of pruning $(1 - \gamma)$. If the pruning represents channel erasures, then the lost information can not be recovered at the server.

B. Stochastic Gradient Pruning

The reliability of a generic communication system is measured by data distortion as we have discussed. On the contrary, the performance of an FL system is measured using an *End-to-End* (E2E) metric such as convergence rate or learning accuracy. In such a system, the pruning of transmitted data has the effect of slowing down the learning speed. Recall that FL is essentially a distributed implementation of SGD. To substantiate the above claim, we consider randomly pruned SGD implemented using classic *Block Coordinate Descent* (BCD) [18]. Let the loss function, $F(\mathbf{w})$ comprise a smooth and convex loss function, $f(\mathbf{w})$, that is regularized by a block separable function, $\Phi(\mathbf{w})$:

$$F(\mathbf{w}) = f(\mathbf{w}) + \Phi(\mathbf{w}). \quad (8)$$

The regularization $\Phi(\mathbf{w}) = \sum_{b=1}^B \Phi_b(\mathbf{w}_b)$ is a sum of B convex, closed functions $\Phi_b(\mathbf{w}_b)$ where \mathbf{w}_b is a block of model parameters. The blocks are non-overlapping and together they constitute the whole model. Considering round n , the server selects one block randomly and notifies devices. Then each device computes the gradient locally based on $\mathbf{w}(n)$ and uploads the specified block of coefficients to the server. Thus, only one selected block is updated using a pruned gradient aggregated from devices, i.e., $\mathbf{g}^{\text{sp}}(n)$, while others remain unchanged. Mathematically,

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta \mathbf{g}^{\text{sp}}(n). \quad (9)$$

Equivalently, BCD can be seen as distributed SGD updated with pruned gradients where the pruning ratio is $\gamma = \frac{1}{B}$. Its convergence rate is measured by the required number of iterations, say $N_{\epsilon, \rho}$, guaranteeing ϵ -accuracy with probability of at least $1 - \rho$, $\rho \in (0, 1]$:

$$\Pr\{F(\mathbf{w}(N_{\epsilon, \rho})) - F(\mathbf{w}^*) \leq \epsilon\} \geq 1 - \rho, \quad (10)$$

where \mathbf{w}^* represents the global optimality point. It can be proved that [18],

$$N_{\epsilon, \rho} \leq \mathcal{O} \left(\frac{B}{\epsilon} \log \left(\frac{1}{\rho} \right) \right) = \mathcal{O} \left(\frac{1}{\gamma \epsilon} \log \left(\frac{1}{\rho} \right) \right). \quad (11)$$

One can observe that the required number of iterations (i.e, learning latency) is inversely proportional to the pruning ratio.

C. Why Random Pruning for AirFL?

Random gradient/model pruning is popularly adopted for FL (see e.g., [15], [32]). The alternative scheme, importance-aware pruning that prunes gradient coefficients with the smallest magnitudes, does not allow efficient implementation for several reasons discussed in [32]. First, AirComp requires local gradient coefficients pruned by different devices to have identical positions in the local gradient vectors. This cannot be guaranteed if devices perform independent importance-aware pruning. Second, doing so requires devices to upload indices of pruned/remaining coefficients to the server to facilitate aggregation, thereby incurring additional, significant communication overhead [36]. Finally, importance-aware pruning increases devices' computation loads due to the coefficient sorting of high-dimensional gradient vectors.

IV. OVERVIEW OF SPECTRUM BREATHING

As illustrated in Fig. 2, the proposed spectrum breathing technique consists of operations at the transmitter of a device and at the receiver of the server. They are described separately in the following sub-sections.

A. Transmitter Design

The transmitter design is shown in Fig. 2(a), comprising three cascaded operations, i.e., spectrum contraction, channel inversion, and spectrum spreading.

1) *Spectrum Contraction*: The operation is to randomly prune the elements of a local gradient at each device. The purpose is to create extra bandwidth for the latter operation of spread spectrum. The operation compresses the spectrum required for transmitting a local gradient, giving the name of spectrum contraction. Consider round n and local gradient $\mathbf{g}_k(n)$ at device k . Let ψ_n , $S_n \triangleq |\psi_n|$, and Ω_n denote the selected element set, number of selected elements, and the set of all S_n -element subsets of $\{1, 2, \dots, D\}$, respectively. At the server, ψ_n is chosen randomly from Ω_n before being broadcast to devices. Using ψ_n , device k compresses $\mathbf{g}_k(n)$ into

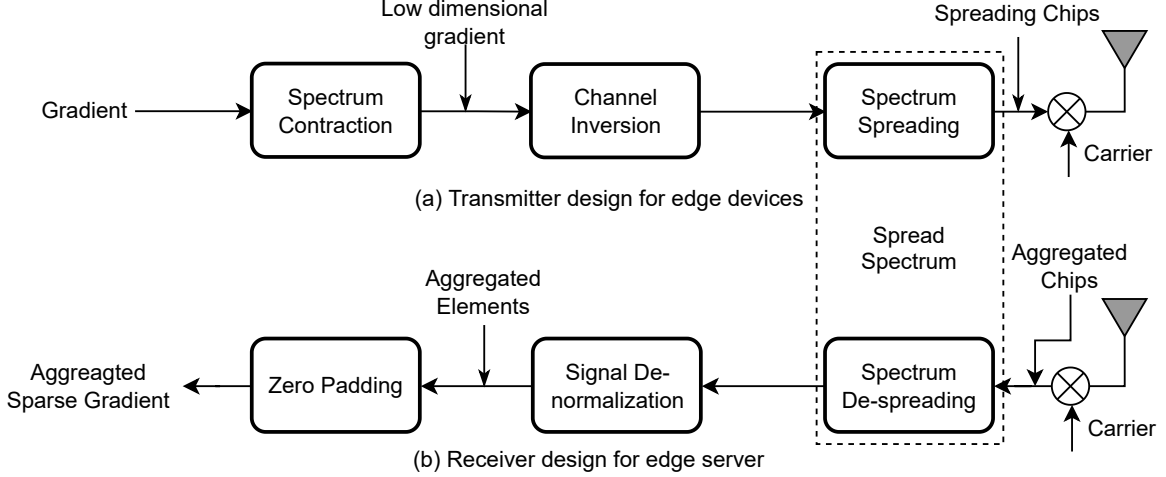


Fig. 2. Transceiver of the spectrum breathing system.

an $S_n \times 1$ vector, denoted as $\mathbf{g}_k^{\text{co}}(n) = \Psi(\mathbf{g}_k(n))$, using the pruning function $\Psi(\cdot) : \mathbb{R}^D \rightarrow \mathbb{R}^{S_n}$. Note that the pruning pattern ψ_n is identical for all devices as required for AirComp to realize element alignment. Since the gradient statistics may change over iterations, normalization is needed in each round to meet the power constraint [37]. The normalized pruned gradient of the compressed version is given as

$$\hat{\mathbf{g}}_k^{\text{co}}(n) = \frac{\mathbf{g}_k^{\text{co}}(n) - M(n)\mathbf{1}}{V(n)}, \quad (12)$$

where $\mathbf{1}$ is an all-one vector. Considering i.i.d data distribution as in [6], [19] and random pruning, the elements of $\mathbf{g}_k^{\text{co}}(n)$ can be modeled as identically distributed random variables over k with mean $M(n)$ and variance $V^2(n)$. This enables normalized gradient symbol power, i.e., $\frac{1}{S_n} \mathbb{E}[\|\hat{\mathbf{g}}_k^{\text{co}}(n)\|^2] = 1$.

2) *Channel Inversion*: Following [19], truncated channel inversion is performed to achieve amplitude alignment as required for AirComp. We consider block fading channels such that the channel state is constant in each round. To avoid deep fading, device k is inverted only if its gain exceeds a given threshold, denoted as G_{th} , or otherwise device k is absent in this round by setting its power as zero. Mathematically,

$$p_k(n) = \begin{cases} \frac{P_0}{|h_k(n)|^2}, & |h_k(n)|^2 \geq G_{th} \\ 0, & |h_k(n)|^2 < G_{th}, \end{cases}, \quad (13)$$

where P_0 is the signal-magnitude-alignment factor. Transmission of each device is subject to a long-term power constraint over N rounds:

$$\mathbb{E} \left[\sum_{n=0}^{N-1} G_n S_n p_k(n) \right] \leq P_{\max}, \quad (14)$$

where the expectation is taken over the randomness of channel coefficients and transmitted symbols. Given (5), $\frac{P_0}{P_I}$ determines the receive SIR of the model-update from each device [19]. The probability that device k avoids truncation, called activation probability, is denoted by ξ_a and obtained as

$$\xi_a = \Pr(|h_k(n)|^2 \geq G_{th}) = e^{-G_{th}}. \quad (15)$$

Due to random truncation, the random set of active devices of round n is denoted as $\mathcal{K}(n)$, which varies over rounds.

3) *Spectrum Spreading*: For interference suppression, spectrum spreading [30] is performed to expand the data bandwidth, denoted as $B_s(n)$, into the whole available bandwidth, denoted as B_c , using PN sequences. Let $T_s(n) = \frac{1}{B_s(n)}$ and $T_c = \frac{1}{B_c}$ denote the duration of one gradient symbol and one chip of PN sequences, respectively. Then a PN sequence comprises $G_n = T_s(n)/T_c$ chips, when G_n is called the processing gain. The key operation of the spreader is to upsample and scramble the input elements by the corresponding PN sequences to realize spectrum expansion. Given the s -th input element of the spreader, say $[\widehat{\mathbf{g}}_k^{\text{co}}(n)]_s$, the corresponding PN sequence is represented as $\mathbf{C}_s(n) \in \mathbb{R}^{G_n}$ wherein $[\mathbf{C}_s(n)]_\ell \in \{+1, -1\}$, $s \in \{1, 2, \dots, S_n\}$, $\ell \in \{1, 2, \dots, G_n\}$ is the ℓ -th chip and is generated at the server through Bernoulli trials with the probability 0.5. The set of PN sequences, denoted as $\mathcal{C}(n) \triangleq \{\mathbf{C}_1(n), \dots, \mathbf{C}_{S_n}(n)\}$, is broadcast to all devices. For device k , the output of the spreader is represented by a $G_n S_n \times 1$ vector, say $\tilde{\mathbf{g}}_k(n) = [[\tilde{\mathbf{g}}_{k,1}(n)]^T, \dots, [\tilde{\mathbf{g}}_{k,s}(n)]^T, \dots, [\tilde{\mathbf{g}}_{k,S_n}(n)]^T]^T$, where $\tilde{\mathbf{g}}_{k,s}(n) \in \mathbb{R}^{G_n}$ is a G_n -entry vector representing the spreading chips of $[\widehat{\mathbf{g}}_k^{\text{co}}(n)]_s$, given as

$$\tilde{\mathbf{g}}_{k,s}(n) = [\widehat{\mathbf{g}}_k^{\text{co}}(n)]_s \mathbf{C}_s(n), \quad \forall (s, k). \quad (16)$$

Note that for all $s \in \{1, 2, \dots, S_n\}$, $\frac{1}{G_n} \sum_{\ell=1}^{G_n} [\mathbf{C}_s(n)]_\ell^2 = 1$ holds.

B. Receiver Design

The receiver design is illustrated in Fig. 2(b) comprising three cascaded operations, i.e., spectrum de-spreading, signal de-normalization and zero padding.

1) *Spectrum De-spreading*: The operation targets mining the desired gradient symbols hidden in the interference using the *de-spreader* to be elaborated in the following. Perfect synchroniza-

tion between transmitters and receiver is assumed such that chip-level operations of spectrum de-spreading can be realized. Considering round n , the received signals at the server are the superimposed waveform due to the simultaneous transmission of devices. Let $\tilde{\mathbf{y}}(n) \in \mathbb{R}^{S_n}$ denote the output vector of spectrum de-spreading. By introducing the truncated channel inversion in (13), the s -th output element, say $[\tilde{\mathbf{y}}(n)]_s$, is given as

$$\begin{aligned} [\tilde{\mathbf{y}}(n)]_s &= \frac{1}{G_n} \sum_{\ell=1}^{G_n} [\mathbf{C}_s(n)]_{\ell} [\mathbf{y}_s(n)]_{\ell} = \sum_{k \in \mathcal{K}(n)} \frac{\sqrt{P_0}}{G_n} \sum_{\ell=1}^{G_n} [\mathbf{C}_s(n)]_{\ell}^2 [\hat{\mathbf{g}}_k^{\text{co}}(n)]_s + [\tilde{\mathbf{z}}(n)]_s, \\ &= \sqrt{P_0} \sum_{k \in \mathcal{K}(n)} [\hat{\mathbf{g}}_k^{\text{co}}(n)]_s + [\tilde{\mathbf{z}}(n)]_s, \end{aligned} \quad (17)$$

where $[\tilde{\mathbf{z}}(n)]_s \sim \mathcal{N}(0, \frac{P_I}{G_n})$ is zero-mean Gaussian interference, whose power is inversely proportional to G_n [38].

2) *Signal De-normalization*: This operation is performed to eliminate the impact of normalization and channel inversion to obtain the noisy averaged gradient symbols, denoted as $\hat{\mathbf{y}}(n) \in \mathbb{R}^{S_n}$, given as

$$\begin{aligned} \hat{\mathbf{y}}(n) &= \frac{V(n)}{\sqrt{P_0}|\mathcal{K}(n)|} \tilde{\mathbf{y}}(n) + |\mathcal{K}(n)|M(n)\mathbf{1} \\ &= \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k^{\text{co}}(n) + \frac{V(n)}{\sqrt{P_0}|\mathcal{K}(n)|} \tilde{\mathbf{z}}(n). \end{aligned} \quad (18)$$

3) *Gradient Zero-padding*: To facilitate global-model updating, zero padding executes the inverse of pruning $\Psi^{-1}(\cdot) : \mathbb{R}^{S_n} \rightarrow \mathbb{R}^D$ to restore the D -dimensional update by inserting zeros into the punctured dimensions. The zero-padded gradient, denoted as $\mathbf{y}'(n)$, is represented as

$$\mathbf{y}'(n) = R \left(\frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n) + \hat{\mathbf{z}}(n) \right), \quad (19)$$

where $\hat{\mathbf{z}}(n) \in \mathbb{R}^D$ represents the interference vector distributed as $\mathcal{N}(0, \frac{V^2(n)P_I}{P_0|\mathcal{K}(n)|^2G_n} \mathbf{I}_D)$, and $R(\cdot) : \mathbb{R}^D \times \Omega_n \rightarrow \mathbb{R}^D$ is the zero-padding operation; Finally, devices update the global model using the gradient $\mathbf{y}'(n)$ after its broadcasting from the server.

V. CONVERGENCE ANALYSIS OF AIRBREATHING FEDERATED LEARNING

In this section, we analyze the convergence of AirBreathing FL. The results are useful for optimizing the spectrum breathing in the next section.

A. Assumptions, Definitions, and Known Results

For tractable analysis, some commonly used assumptions, definitions, and known results are provided below. First, we consider a strongly-convex loss function with bounded gradient estimates. These assumptions are commonly used in the literature (see, e.g., [36], [39], [40]).

Assumption 1. The differentiable loss function $F(\cdot)$ is c -strongly convex, i.e., $\forall \mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^D$,

$$F(\mathbf{w}_1) - F(\mathbf{w}_2) \geq \nabla F(\mathbf{w}_1)^T(\mathbf{w}_2 - \mathbf{w}_1) + \frac{c}{2} \|\mathbf{w}_2 - \mathbf{w}_1\|^2. \quad (20)$$

Assumption 2. Let $\mathbf{w}^* \in \mathbb{R}^D$ denote the optimality point of $F(\cdot)$. For $\varepsilon > 0$, there exists a success region indicating the convergence, defined as $\mathcal{S} = \{\mathbf{w} \mid \|\mathbf{w} - \mathbf{w}^*\|^2 \leq \varepsilon\}$.

Assumption 3. Local gradients $\mathbf{g}_k(n)$ are i.i.d. over devices $k \in \{1, 2, \dots, K\}$ with unbiased estimate of the ground truth $\mathbf{g}(n)$ and bounded variance, i.e.,

$$\mathbb{E}[\mathbf{g}_k(n)] = \mathbf{g}(n), \quad \mathbb{E}[\|\mathbf{g}_k(n) - \mathbf{g}(n)\|^2] \leq \sigma_g^2, \quad \mathbb{E}[\|\mathbf{g}(n)\|^2] \leq \zeta^2, \quad (21)$$

for all (n, k) , where σ_g^2 and ζ^2 are constants.

Next, we adopt the method of martingale-based convergence analysis in [39]. To this end, a useful definition and some known results from [39] are provided below.

Definition 1 ([39]). A non-negative process $W_n(\mathbf{w}(n), \mathbf{w}(n-1), \dots, \mathbf{w}(0)) : \mathbb{R}^{D \times (n+1)} \rightarrow \mathbb{R}$ is defined as a rate supermartingale with a scalar parameter A , called the horizon, if the following conditions hold.

- 1) It must be a supermartingale [41], i.e., for any sequence $\mathbf{w}(n), \mathbf{w}(n-1), \dots, \mathbf{w}(0)$ and $\forall n \leq A$,

$$\mathbb{E}[W_{n+1}(\mathbf{w}(n+1), \mathbf{w}(n), \dots, \mathbf{w}(0))] \leq W_n(\mathbf{w}(n), \dots, \mathbf{w}(0)). \quad (22)$$

- 2) For all rounds $N \leq A$ and for any sequence $\mathbf{w}(n), \mathbf{w}(n-1), \dots, \mathbf{w}(0)$, if the algorithm has not converged into the success region by N (i.e., $\mathbf{w}(n) \notin \mathcal{S}, \forall n \leq N$),

$$W_N(\mathbf{w}(N), \mathbf{w}(N-1), \dots, \mathbf{w}(0)) \geq N. \quad (23)$$

Lemma 1 ([39], Lemma 1). Consider an FL system updating as in (4) with a learning rate $\eta < 2c\varepsilon\mathbb{G}^2$. If the algorithm has not converged by round n , the process defined as

$$W_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) \triangleq \frac{\varepsilon}{2\eta c\varepsilon - \eta^2 \mathbb{G}^2} \log \left(\frac{e \|\mathbf{w}(n) - \mathbf{w}^*\|^2}{\varepsilon} \right) + n, \quad (24)$$

is a rate supermartingale with horizon $A = \infty$, where $\mathbb{G}^2 \geq \zeta^2 + \sigma_g^2$ is the upper bound of the squared norm of aggregated gradients. Under Assumptions 1-2, $W_n(\mathbf{w}(n), \dots, \mathbf{w}(0))$ is also H -Lipschitz smooth in the first coordinate, with $H = 2\sqrt{\varepsilon}(2\eta c\varepsilon - \eta^2 \mathbb{G}^2)^{-1}$. In other words, for any $n \geq 1$, $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^D$ and any sequence $\mathbf{w}(n-1), \dots, \mathbf{w}(0)$, it satisfies

$$\|W_n(\mathbf{w}_1, \mathbf{w}(n-1), \dots, \mathbf{w}(0)) - W_n(\mathbf{w}_2, \mathbf{w}(n-1), \dots, \mathbf{w}(0))\| \leq H \|\mathbf{w}_1 - \mathbf{w}_2\|. \quad (25)$$

Intuitively, the rate supermartingale represents the level of satisfaction for model weights $\mathbf{w}(n), \mathbf{w}(n-1), \dots, \mathbf{w}(0)$ over $n+1$ rounds. Some intuition into the preceding assumptions are as follows. First, (22) reflects the fact that obtained model weights are more satisfactory as they approach the optimality point. Second, as specified in (23), the satisfaction is reduced if the algorithm is executed for many rounds without convergence. FL updating as in (4) is considered as the vanilla SGD satisfying the properties of rate supermartingale. It is a commonly used analytical method in the convergence analysis of SGD [36], [39], [40].

B. Convergence Analysis

Based on the preceding assumptions, we further develop the mentioned rate-supermartingale approach to study the convergence of AirBreathing FL. The new approach is able to account for channel fading, and the system operations such as AirComp and spread spectrum. Specifically, several useful intermediate results are obtained as shown in the following lemmas.

We first upper bound the gap between the vanilla SGD and AirBreathing FL using the results on the AirComp error (see Lemmas 2 and 3). Next, based on Lemma 3, a supermartingale for AirBreathing FL is constructed in Lemma 4. Furthermore, the upper bound of the convergence rate is derived using the theory of martingale as shown in Lemma 5.

Lemma 2. The AirComp error defined in (6) for round n , can be expressed as the sum of the gradient-pruning error and interference-induced error:

$$\text{MSE}(n) = \underbrace{(1 - \gamma_n) \mathbb{E}[\alpha^2(n)]}_{\text{gradient-pruning error}} + \underbrace{\frac{\gamma_n D P_I}{G_n P_0} \mathbb{E} \left[\frac{V^2(n)}{|\mathcal{K}(n)|^2} \right]}_{\text{interference-induced error}}, \quad (26)$$

where $\gamma_n = \frac{S_n}{D}$ represents the pruning ratio in round n , and $\alpha^2(n)$ is defined as

$$\alpha^2(n) = \left\| \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n) \right\|^2. \quad (27)$$

Proof. See Appendix A.

Lemma 3. Considering round n , the gap between vanilla SGD and AirBreathing FL is defined as the expected difference between the update of vanilla SGD, namely $\frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n)$, and spectrum breathing, namely $\mathbf{y}'(n)$. The gap can be bounded as

$$\mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n) - \mathbf{y}'(n) \right\| \right] \leq u(n), \quad (28)$$

where $u(n)$ is defined as

$$u(n) = \frac{2 - \xi_a}{K \xi_a} \sigma_g^2 + \sqrt{\text{MSE}(n)}, \quad (29)$$

where ξ_a and σ_g^2 is the activation probability in (15) and gradient variance in (21), respectively. The first and second terms of the bound result from 1) the fading channel and gradient randomness and 2) the AirComp error in (26), respectively.

Proof. See Appendix B.

The process defined in Lemma 3, $\{u(n)\}$, serves as indicators of performance loss caused by the air interface and is hence termed *propagation-loss process*. To this end, we use the result in Lemma 3 to define a new stochastic process pertaining to spectrum breathing and show it to be a supermartingale. The details are as follows.

Lemma 4. Define a stochastic process, $\{U_n\}$, as

$$U_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) \triangleq W_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) - \eta H \sum_{i=0}^{n-1} u(i), \quad (30)$$

for $\forall i \leq n$ and $\mathbf{w}(i) \notin \mathcal{S}$, $\{U_n\}$ is a supermartingale process.

Proof. See Appendix C.

Note that, U_n has a negative term, which is a function of the propagation-loss process, removes from the model under training the effect of the air interface. Thereby, the result in Lemma 4 facilitates the use of supermartingale theory to quantify the convergence probability

of AirBreathing FL as shown below.

Lemma 5. Consider N rounds and AirBreathing FL for minimizing the loss function $F(\mathbf{w})$. If the learning rate satisfies

$$\eta < \frac{2\sqrt{\varepsilon} \left(c\sqrt{\varepsilon}N - \sum_{n=0}^{N-1} u(n) \right)}{N\mathbb{G}^2}, \quad (31)$$

the event of failing to converge to the success region, denoted as F_N , has a probability bounded as

$$\Pr\{F_N\} \leq \frac{\varepsilon \log(e\|\mathbf{w}(0) - \mathbf{w}^*\|^2 \varepsilon^{-1})}{(2\eta c\varepsilon - \eta^2\mathbb{G}^2)N - 2\eta\sqrt{\varepsilon} \sum_{n=0}^{N-1} u(n)}. \quad (32)$$

where \mathbb{G}^2 is the upper bound of aggregated gradient defined in Lemma 1.

Proof: See Appendix D.

Definition 2 (Breathing Depth). In the considered scenario of constrained bandwidth-and-latency, it is necessary to fix the product of processing gain and pruning ratio: $G_n\gamma_n = 1$. Under this constraint, the tradeoff between spread spectrum and gradient pruning can be regulated by the processing gain $G_n = \frac{1}{\gamma_n}$. To be more instructive, it is renamed the *Breathing Depth*, that is the most important control variable of spectrum breathing.

Substituting $G_n = \frac{1}{\gamma_n}$ into the result in Lemma 5 yields the following main result.

Theorem 1. Consider AirBreathing FL with breathing depths $\{G_n\}$ and N rounds. If the learning rate satisfies (31), the probability of failing to converge to the success region is bounded as

$$\Pr\{F_N\} \leq \frac{\varepsilon \log(e\|\mathbf{w}(0) - \mathbf{w}^*\|^2 \varepsilon^{-1})}{\left(2c\varepsilon - \eta\mathbb{G}^2 - 2\sqrt{\frac{(2-\xi_a)\varepsilon}{M\xi_a}}\sigma_g\right)\eta N - 2\eta\sqrt{\varepsilon}\beta_\Sigma}. \quad (33)$$

Here $\beta_\Sigma = \sum_{n=0}^{N-1} \sqrt{\beta_n(G_n)}$ is a sum of error terms where each term $\beta_n(G_n)$ represents the air-interface error in round n , given as

$$\beta_n(G_n) = \underbrace{\left(1 - \frac{1}{G_n}\right) \mathbb{E}[\alpha^2(n)]}_{\text{gradient-pruning error}} + \underbrace{\frac{DP_I}{G_n^2 P_0} \mathbb{E}\left[\frac{V^2(n)}{|\mathcal{K}(n)|^2}\right]}_{\text{interference-induced error}}. \quad (34)$$

Consider the air-interface error term in (34). One can see that increasing the breathing depth, G_n , corresponds to decreasing the pruning ratio and thus causes the pruning error to grow. On the

other hand, increasing G_n enhances the process gain and thereby reduces interference perturbation (and its corresponding error term). The above tradeoff suggests the need of optimizing $\{G_n\}$, which is the topic of the next section.

Comparing Theorem 1 to the convergence analysis in related works [25], [36], our results have two main differences: First, our results reflect the effect of spectrum breathing depth, G_n , in (34), which does not exist in prior work. When $G_n = 1$ (no breathing, pruning), (34) reduces to the mean squared norm of the introduced noise in [25, (18)]. Second, the effect of fading channels on convergence is characterized in (34) by the term $\frac{1}{|\mathcal{K}(n)|^2}$, while the $|\mathcal{K}(n)|$ in [36] is assumed to be constant.

VI. OPTIMIZATION OF SPECTRUM BREATHING

In this section, the results from the preceding convergence analysis are applied to the optimization of the breathing depth of AirBreathing FL. To enhance convergence, Theorem 1 imposes the need of minimizing (34). Before that, the assumptions on known and unknown parameters are specified as follows. The predefined parameters, i.e., model size D and receive SIR $\frac{P_0}{P_T}$, are assumed to be known. Let *Gradient State Information* (GSI) refer to the statistical parameters of the stochastic gradient in the current round, namely $\alpha(n)$ and $V(n)$ in (34), which are not accessible but can be estimated at each round using local gradients. Moreover, let CSI refer to the channel-dependent number of active devices in the current round, namely $|\mathcal{K}(n)|$. It is known to the server by assuming perfect channel estimation over rounds. Then we consider the optimization in two cases: (1) without GSI and CSI, and (2) with GSI and CSI at the server.

A. Breathing Depth Optimization without GSI and CSI Feedback

Without GSI and CSI feedback, we deploy fixed breathing depth for all rounds, i.e., $G_n = G, \forall n$. Consider the term β_Σ in Theorem 1, which is the only term related to the breathing depth, G . Then G is optimized to minimize β_Σ , thereby accelerating convergence. The difficulty of such optimization lies in the lack of the required GSI and CSI. To address the issue, we resort to minimizing an upper bound on β_Σ that requires no such information.

Lemma 6. Consider gradient $\mathbf{g}_k(n), k \in \mathcal{K}(n)$. There exists a positive constant $\Gamma(n)$, satisfying $\Gamma(n) \geq \frac{1}{D}(\|\mathbf{g}(n)\|^2 + \sigma_g^2)$, such that β_Σ is upper bounded as

$$\beta_\Sigma \leq \beta_F(G) \sum_{n=0}^{N-1} \sqrt{D\Gamma(n)}, \quad (35)$$

where $\beta_F(G)$ is a function of G , given as

$$\beta_F(G) = \sqrt{1 - \frac{1}{G} + \frac{6P_I}{G^2 K^2 \xi_a^2 P_0}}. \quad (36)$$

Proof. See Appendix E.

We next formulate the optimization problem

$$\begin{aligned} \min_G \quad & \beta_F(G) \\ \text{s.t.} \quad & G \in \{1, 2, \dots, D\}. \end{aligned} \quad (37)$$

Problem (37) can be solved by integer relaxation as follows. Given a continuous variable $x > 0$, setting $\nabla_x \beta_F(x) = 0$ yields the optimal solution $x^* = \frac{12P_I}{P_0 K^2 \xi_a^2}$. Thus, for the discrete function $\beta_F(G), \forall G \in \{1, 2, \dots, D\}$, the fixed breathing depth, denoted G^* , can be obtained approximately by

$$G^* = \begin{cases} 1, & x^* < 1, \\ \lfloor x^* \rfloor_{\beta_F(G)}, & 1 \leq x^* \leq D, \\ D, & x^* > D, \end{cases} \quad (38)$$

where $\lfloor \hat{x} \rfloor_{\beta_F(x)}$ is equal to $\lfloor \hat{x} \rfloor$ if $\beta_F(\lfloor \hat{x} \rfloor) \leq \beta_F(\lceil \hat{x} \rceil)$, and is otherwise equal to $\lceil \hat{x} \rceil$.

In the above results, G^* is a monotonous decreasing function of the receive SIR $\frac{P_0}{P_I}$. It implies that for a low receive SIR, we allocate more bandwidth resources for interference suppression to guarantee convergence at cost of more aggressive gradient compression. On the other hand, for a high receive SIR, the spectrum-breathing control favours uploading as many gradient dimensions as possible to attain faster convergence. Increasing the expected number of active devices $K\xi_a$ directly enhances the received signal power, which suppresses the interference by aggregation gain and hence reduces the need of interference suppression via spectrum spreading.

B. Breathing Depth Optimization with GSI and CSI Feedback

Given GSI and CSI feedback, the breathing depth can be adapted over rounds and hence re-denoted as G_n for round n . The optimization criteria is to minimize the estimate of the relevant term, $\beta_n(G_n)$, of the successful convergence probability in the Theorem 1. Let the estimate be denoted as $\hat{\beta}_n(G_n)$:

$$\hat{\beta}_n(G_n) = \left(1 - \frac{1}{G_n}\right) \hat{\alpha}^2(n) + \frac{DP_I \hat{V}^2(n)}{G_n^2 P_0 |\mathcal{K}(n)|^2}, \quad (39)$$

where $\hat{\alpha}(n)$ and $\hat{V}(n)$ are estimates of $\alpha(n)$ and $V(n)$, respectively, while the number of active devices, $|\mathcal{K}(n)|$, is perfectly known at the server from CSI. Based on feedback statistics of local gradients, the estimation is similar to that in [12] given as

$$\hat{\alpha}^2(n) = \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \|\mathbf{g}_k(n)\|^2, \quad \hat{V}^2(n) = \frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \hat{V}_k^2(n), \quad (40)$$

where $\hat{V}_k^2(n)$ is the local gradient variance:

$$\hat{V}_k^2(n) = \frac{1}{D} \sum_{d=1}^D \left([\mathbf{g}_k(n)]_d - \frac{1}{D} \sum_{d=1}^D [\mathbf{g}_k(n)]_d \right)^2. \quad (41)$$

Consider an arbitrary round n of AirBreathing FL. Based on (39) and (40), the problem of optimizing the breathing depth can be formulated as

$$\begin{aligned} \min_{G_n} \quad & \hat{\beta}_n(G_n) \\ \text{s.t.} \quad & G_n \in \{1, 2, \dots, D\}, \\ & \forall n \in \{0, 1, \dots, N-1\}. \end{aligned} \quad (42)$$

Again, by integer relaxation of G_n into $x > 0$, $\nabla_x \hat{\beta}_n(x) = 0$ yields the optimal solution $x_n^* = \frac{2P_I D \hat{V}^2(n)}{P_0 |\mathcal{K}(n)|^2 \hat{\alpha}^2(n)}$ such that $\hat{\beta}_n(x_n^*)$ is the minimum. Then an approximate of the adaptive breathing depth is given as

$$G_n^* = \begin{cases} 1, & x_n^* < 1, \\ \lfloor x_n^* \rfloor_{\hat{\beta}_n(G_n)}, & 1 \leq x_n^* \leq D, \\ D, & x_n^* \geq D. \end{cases} \quad (43)$$

Note that the adaptive breathing depth is a clipping function of x_n^* , truncated by the smallest and largest achievable value. For the general case, $1 \leq x_n^* \leq D$, G_n^* is found to be inversely proportional to the receive SIR $\frac{P_0}{P_I}$, number of active devices $|\mathcal{K}(n)|^2$ and estimate of gradient squared norm $\hat{\alpha}^2(n)$. Enlarging these parameters reduces the impact of interference on convergence so that breathing depth decreases accordingly. On the other hand, the breathing depth increases with the rise of gradient-variance estimate due to the scaling of interference in de-normalization. The resultant protocol for adaptive spectrum breathing is summarized in Algorithm 1.

Algorithm 1: Adaptive Breathing Depth Protocol

Input: Receive SIR P_0/P_I , Model size D ;

- 1: Initialisation : $\mathbf{w}(0)$ in all devices;
- 2: **for** Round: $n = 0$ to N **do**
- 3: **for** each device $k \in \mathcal{K}(n)$ in parallel **do**
- 4: Computes $\mathbf{g}_k(n)$ via (2);
- 5: Computes $\|\mathbf{g}_k(n)\|^2$;
- 6: Computes $\hat{V}_k^2(n)$ via (41);
- 7: Uploads $\|\mathbf{g}_k(n)\|^2$ and $\hat{V}_k^2(n)$ to server;
- 8: **end for**
- 9: Server estimates $\hat{\alpha}^2(n)$ and $\hat{V}^2(n)$ via (40);
- 10: Server computes adaptive breathing depth G_n^* via (43);
- 11: Server generates selected index set ψ_n and set of PN sequence $\mathcal{C}(n)$ w.r.t. G_n^* ;
- 12: Server broadcasts ψ_n and $\mathcal{C}(n)$ to all devices;
- 13: **Spectrum Breathing Process** returns $\mathbf{w}(n + 1)$
- 14: **end for**
- 15: **return** $\mathbf{w}(n + 1)$

VII. EXPERIMENTAL RESULTS

In this section, the preceding fixed and adaptive breathing depth protocols are simulated. Based on this, we evaluate the performance of AirBreathing FL by comparing it with six benchmarks to be specified below.

A. Experimental Settings

The default experimental settings are as follows unless specified otherwise.

- **Communication Settings:** We consider an AirBreathing FL system comprising one server and 10 devices. In each round, the PN sequence shared by devices is generated by having i.i.d. chips following the unbiased Bernoulli distribution; the sequence is varied over rounds. Each chip spans unit time and a transmitted gradient coefficient occupies G_n chips with G_n being the breathing depth. The interference at the server's receiver is modelled as a sequence of i.i.d Gaussian symbols. Assuming Rayleigh fading, all channel coefficients are modelled as $\mathcal{CN}(0, 1)$ random variables. Consider the scenario of strong interference. The devices' fixed transmission power and the interference power are set such the expected receive SIR is -23dB , which can be enhanced by aggregation and spectrum de-spreading in AirBreathing. Finally, the threshold of truncated channel inversion is set as $G_{th} = 0.2$ and the resultant activation probability of each device is $\xi_a = 0.82$.

- **Learning Settings:** We consider the learning task of handwritten digit classification using the popular MNIST dataset. To model non-i.i.d data at devices, each of which comprises 3000 randomly drawn samples of one class. Two randomly chosen shards with different labels are assigned to each device. The task is to train a CNN model having 21,840 parameters. The model consists of two 5×5 convolutional layers with ReLU activation (with 10, 20 channels, respectively), and the ensuing 2×2 max pooling, a fully connected layer with 50 units and ReLU activation, and a final softmax output layer. Furthermore, the pruning for spectrum contraction is executed on model weights (99.8% of all parameters) but not bias to avoid divergence.

Six benchmarking schemes with their legends in brackets are described below.

- **Ideal Case:** The ideal FL system without pruning and channel distortion.
- **No Spectrum Breathing (No SB):** This is equivalent to AirBreathing FL with $G_n = 1, \forall n$. The resultant system is exposed to strong interference.
- **Pruning without Spectrum Spreading:** Gradients are pruned randomly with a fixed ratio, γ , and uploaded without spread spectrum. This results in the exposure of pruned gradients to strong interference.
- **Convergent OTA FL (COTAF) [25]:** Given the same power constraint and interference, time-varying scalar precoding (equivalent to power control) with full-dimensional gradient uploading is simulated by accounting for the gradually decreasing squared norm of gradient over rounds. Note that COTAF requires offline simulation to estimate the scalar precoder, while AirBreathing FL does not need this as a result of online estimation from GSI feedback with negligible communication overhead.
- **Optimal Fixed Breathing Depth (Optimal fixed BD):** The optimal fixed breathing depth is obtained using an exhaustive search, as opposed to using the closed-form result in (38).
- **AirBreathing with Importance-aware Pruning (AirBreathing with IP):** Random pruning is replaced with importance-aware pruning, namely pruning gradient coefficients with the smallest magnitudes. The difficulty in its implementation is overcome by alternating rounds of 1) full-gradient uploading to allow the devices to select from aggregated gradient coefficients an index subset to prune in the next round and 2) using the subset to perform importance-aware pruning at devices assuming temporal correlation in gradients [32].

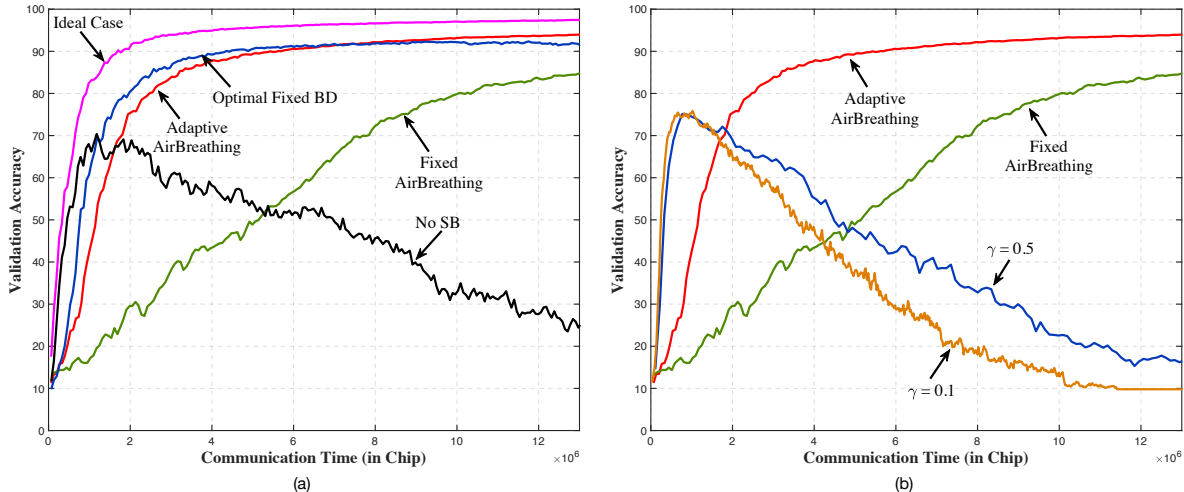


Fig. 3. (a) Performance comparison between AirBreathing FL (in both the cases of fixed and adaptive breathing depth) and benchmarking schemes; (b) Comparison between pruning without spreading ($\gamma = 0.5, 0.1$) and AirBreathing FL.

B. Performance of AirBreathing FL

The learning performance of AirBreathing FL is compared with the benchmarking schemes. For spectrum breathing, both the schemes of fixed and adaptive breathing depth are considered. The curves of validation accuracy versus communication time are plotted in Fig. 3 (a). Several observations can be made. First, AirBreathing in both the cases of fixed and adaptive breathing depth achieves convergence. This demonstrates its effectiveness in coping with strong interference. On the contrary, FL without spectrum breathing, which suffers from strong interference, fails to converge. Second, there exists a substantial performance gap between the scheme of fixed breathing depth from the ideal case due to approximation in the former design to obtain the closed-form result. Thus the gap is largely removed using the proposed scheme of adaptive breathing depth. Finally, AirBreathing with adaptive depth is observed to approach the ideal case within a reasonable performance gap. In particular, the converged accuracy for the former is 96.2% and 94.6% for the latter.

Fig. 3 (b) compares the performance of pruning without spectrum spreading and AirBreathing FL. One can see the former has a rapid increase in accuracy at the beginning, as a result from a higher communication rate in the absence of spectrum spreading. However, the corruption of gradients by interference eventually takes its toll and leads to unsuccessful learning. In contrast, despite a slower learning speed initially, AirBreathing FL ensures steady increase in accuracy to achieve convergence.

Fig. 4 (a) compares the performance between COTAF and adaptive AirBreathing FL in coping

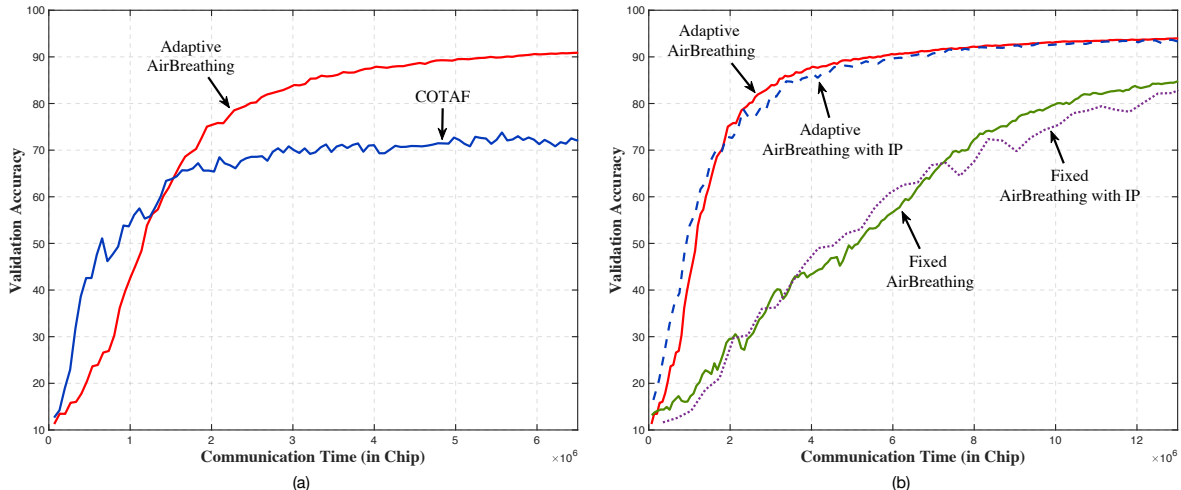


Fig. 4. (a) Performance comparison between COTAF and AirBreathing FL; (b) Comparison between random pruning and importance-aware pruning for AirBreathing FL.

with strong interference. Several observations can be made. When subjected to the same power constraints, COTAF avoids divergence by gradually increasing transmit power. Despite this, COTAF still struggles to achieve a high accuracy due to the discussed limitation of power control in suppressing interference. In contrast, adaptive AirBreathing FL achieves a significantly higher accuracy after convergence, albeit with a slower learning speed at the beginning, thanks to its interference-suppression capability.

C. Random Pruning versus Importance-aware Pruning

In Fig. 4 (b), we compare the learning performance of AirBreathing FL using the proposed random pruning with that of the benchmarking scheme using importance-aware pruning. The main observation is that the latter does not yield any performance gain over the proposed scheme. The reasons are two drawbacks of the importance-aware scheme. First, the full-gradient uploading in every other round in the benchmarking scheme increases communication overhead. Second, the pruned gradient coefficients in a round are selected based on those in the preceding round, resulting in inaccurate choices as gradients vary over rounds.

D. Effects of Network Parameters

We study the effects of two key network parameters, namely the number of devices and the receive SIR per device, on the learning performance of AirBreathing FL for a given communication time of 7×10^6 chips. To this end, the curves of validation accuracy versus varying network parameters are plotted in Fig. 5.

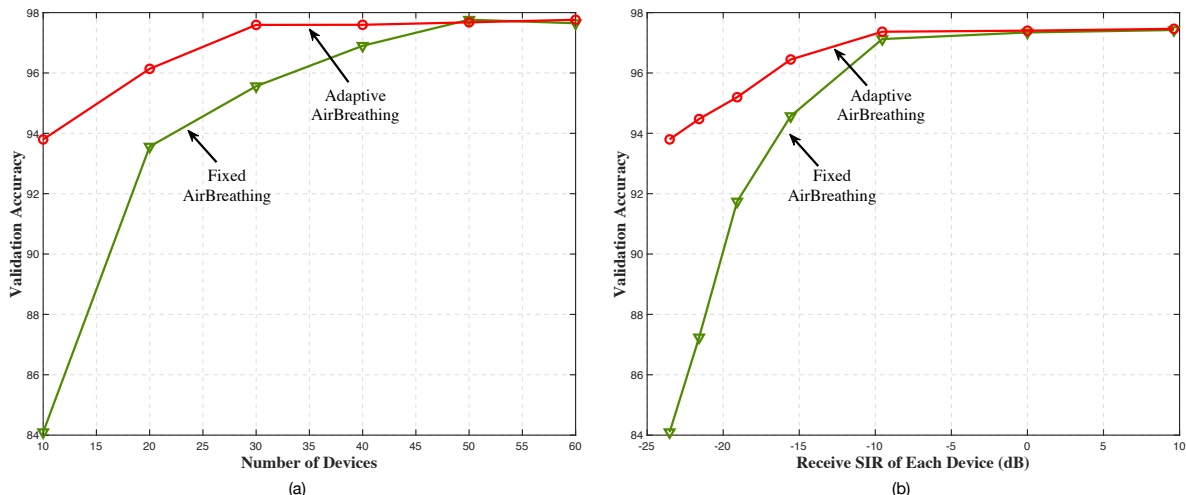


Fig. 5. The effects of (a) number of devices and (b) receive SIR on the learning performance for given communication time of 7×10^6 chips.

From Fig. 5 (a), one can see the continuous performance improvement as the number of devices increases, as the AirComp's aggregation over devices suppresses interference by averaging. Note that AirBreathing realizes interference suppression using a different mechanism of spread spectrum. On the other hand, as expected, the AirBreathing FL with either fixed or adaptive breathing depth sees growing performance improvement as the receive SIR per device (before aggregation) becomes larger.

VIII. CONCLUSION

In this work, we presented a spectrum-efficient method, called spectrum breathing. Leveraging the graceful degradation of learning performance due to pruning, the method exploits signal spectrum contraction via pruning to enable interference suppression via spread spectrum without requiring extra bandwidth. The breathing depth that controls spectrum contraction level is optimized and adapted to both the states of gradient descent and channels to amplify the learning performance gain.

This work establishes a new principle of designing robust AirFL by integrating gradient pruning and interference suppression. Beyond spread spectrum, this principle can be applied to other interference management techniques such as adaptive coding and modulation, MIMO beamforming, and cooperative transmission. The current AirBreathing FL method can also be generalized to more complex systems such as multi-cell or distributed AirFL. Practical issues such as synchronization errors and security warrant further investigation.

APPENDIX

A. Proof of Lemma 2

In this section, the AirComp error of AirBreathing FL using random pruning is derived as below. For the expression brevity, the round index n is omitted in the following equations.

First, the AirComp error of AirBreathing FL, denoted as $\text{MSE}(n)$, is the MSE between received signal $\mathbf{y}'(n)$ and the ideal version $\frac{1}{|\mathcal{K}(n)|} \sum_{k \in \mathcal{K}(n)} \mathbf{g}_k(n)$, given as

$$\begin{aligned}
\text{MSE}(n) &= \mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - \mathbf{y}' \right\|^2 \right] \\
&= \mathbb{E}_{\psi_n} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - R \left(\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right) \right\|^2 \right] + \mathbb{E}_{\hat{\mathbf{z}}} [\|R(\hat{\mathbf{z}})\|^2] \\
&\stackrel{(a)}{=} \mathbb{E}_{\psi_n} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - R \left(\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right) \right\|^2 \right] + \frac{S_n P_I}{G_n P_0} \mathbb{E} \left[\frac{V^2}{|\mathcal{K}|^2} \right] \\
&\stackrel{(b)}{=} \left(1 - \frac{S_n}{D} \right) \mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\|^2 \right] + \frac{S_n P_I}{G_n P_0} \mathbb{E} \left[\frac{V^2}{|\mathcal{K}|^2} \right] \\
&= (1 - \gamma_n) \mathbb{E}[\alpha^2(n)] + \frac{\gamma_n D P_I}{G_n P_0} \mathbb{E} \left[\frac{V^2}{|\mathcal{K}|^2} \right],
\end{aligned} \tag{44}$$

where the expectation is taken over ψ_n and $\hat{\mathbf{z}}(n)$. (a) is derived from the sum power of S_n i.i.d zero mean Gaussian random variables. (b) is derived from the expectation of ψ_n that is chosen at random from Ω_n . That is, for a generic vector $\mathbf{x} \in \mathbb{R}^D$, the MSE between \mathbf{x} and $R(\mathbf{x})$ over ψ_n can be represented as [42],

$$\begin{aligned}
\mathbb{E}_{\psi_n} [\|\mathbf{x} - R(\mathbf{x})\|^2] &= \frac{1}{|\Omega_n|} \sum_{\psi_n \in \Omega_n} \sum_{d=1}^D [\mathbf{x}]_d^2 \mathbb{I}\{d \notin \psi_n\} = \sum_{d=1}^D [\mathbf{x}]_d^2 \sum_{\psi_n \in \Omega_n} \frac{\mathbb{I}\{d \notin \psi_n\}}{|\Omega_n|} \\
&= \sum_{d=1}^D [\mathbf{x}]_d^2 \frac{\binom{D-1}{S_n}}{\binom{D}{S_n}} = \left(1 - \frac{S_n}{D} \right) \|\mathbf{x}\|^2 = (1 - \gamma_n) \|\mathbf{x}\|^2.
\end{aligned} \tag{45}$$

(b) trivially holds by replacing \mathbf{x} of (45) with $\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k$.

The proof of Lemma 2 is completed.

B. Proof of Lemma 3

Consider round n , the gap between vanilla SGD and AirBreathing FL is upper bounded as

$$\begin{aligned}
& \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n) - \mathbf{y}'(n) \right\| \right] = \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k - \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k + \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - \mathbf{y}' \right\| \right] \\
& \leq \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k - \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\| \right] + \mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - \mathbf{y}' \right\| \right] \\
& \stackrel{(c)}{\leq} \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k - \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\| \right] + \sqrt{\mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k - \mathbf{y}' \right\|^2 \right]} \\
& = \underbrace{\mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k - \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\| \right]}_{\nu(n)} + \sqrt{\text{MSE}(n)},
\end{aligned} \tag{46}$$

where (c) comes from Jensen's inequality on concave function $\sqrt{\cdot}$; $\nu(n)$ is upper bounded as

$$\begin{aligned}
\nu(n) &= \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k - \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\| \right] = \mathbb{E} \left[\left\| \frac{|\mathcal{K}| - K}{K|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k + \frac{1}{K} \sum_{k \notin \mathcal{K}} \mathbf{g}_k \right\| \right] \\
&\leq \sqrt{\mathbb{E}_{\mathbf{g}} \left[\left\| \frac{|\mathcal{K}| - K}{K|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k + \frac{1}{K} \sum_{k \notin \mathcal{K}} \mathbf{g}_k \right\|^2 \right]} \\
&\leq \sqrt{\mathbb{E}_{|\mathcal{K}|} \left[\left(\frac{1}{|\mathcal{K}|} - \frac{1}{K} \right)^2 \right] \sigma_g^2} \stackrel{(d)}{\leq} \sqrt{\frac{2 - \xi_a}{K \xi_a} \sigma_g},
\end{aligned} \tag{47}$$

where (d) is due to that $|\mathcal{K}|$ subjects to binomial distribution $\mathcal{B}(K, \xi_a)$ such that the inequality below holds [43]:

$$\mathbb{E} \left[\frac{1}{|\mathcal{K}|} \right] \leq \frac{2}{K \xi_a}, \quad \mathbb{E} \left[\frac{1}{|\mathcal{K}|^2} \right] \leq \frac{6}{K^2 \xi_a^2}. \tag{48}$$

The proof of Lemma 3 is completed.

C. Proof of Lemma 4

We consider the process defined as follows

$$U_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) \triangleq W_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) - \eta H \sum_{i=0}^{n-1} u(i), \tag{49}$$

where $u(i)$ is given in Lemma 3, and global model is updated using (19). When the algorithm has not entered the success region at round n , we have $\forall n \geq 0$,

$$\begin{aligned}
U_{n+1}(\mathbf{w}(n+1), \dots, \mathbf{w}(0)) &= W_{n+1}(\mathbf{w}(n) - \eta \mathbf{y}'(n), \mathbf{w}(n), \dots, \mathbf{w}(0)) - \eta H \sum_{i=0}^n u(i) \\
&\stackrel{(25)}{\leq} W_{n+1} \left(\mathbf{w}(n) - \eta \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n), \mathbf{w}(n), \dots, \mathbf{w}(0) \right) \\
&\quad + \eta H \left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n) - \mathbf{y}'(n) \right\| - \eta H \sum_{i=0}^n u(i),
\end{aligned} \tag{50}$$

where the scaling up results from the H -Lipschitz smooth in the first coordinate given in (25).

Then we take the expectation for both sides of the inequality and use the supermartingale property of W_n . The expectation of U_{n+1} is bounded by

$$\begin{aligned}
&\mathbb{E}[U_{n+1}(\mathbf{w}(n+1), \dots, \mathbf{w}(0))] \\
&\leq \mathbb{E} \left[W_{n+1} \left(\mathbf{w}(n) - \eta \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n), \mathbf{w}(n), \dots, \mathbf{w}(0) \right) \right] \\
&\quad + \eta H \mathbb{E} \left[\left\| \frac{1}{K} \sum_{k=1}^K \mathbf{g}_k(n) - \mathbf{y}'(n) \right\| \right] - \eta H \sum_{i=0}^n \mathbb{E}[u(i)] \\
&\leq W_n(\mathbf{w}(n), \dots, \mathbf{w}(0)) + \eta H u(n) - \eta H \sum_{i=0}^n u(i) = U_n(\mathbf{w}(n), \dots, \mathbf{w}(0)).
\end{aligned} \tag{51}$$

The inequality above still holds in the case when the algorithm has succeeded at round n . Thus, U_n is a supermartingale process for AirBreathing FL.

Proof of Lemma 4 is completed.

D. Proof of Theorem 5

We denote the failure to enter the success region by N as F_N , otherwise, the success as $\neg F_N$. Consider the same model initialization $\mathbf{w}(0)$ for U_n and W_n , we have

$$\begin{aligned}
\mathbb{E}[W_0] &= \mathbb{E}[U_0] \geq \mathbb{E}[U_n] = \mathbb{E}[U_n|F_N] \Pr\{F_N\} + \mathbb{E}[U_n|\neg F_N] \Pr\{\neg F_N\} \\
&\geq \mathbb{E}[U_n|F_N] \Pr\{F_N\} = \left(\mathbb{E}[W_N|F_N] - \eta H \sum_{n=0}^{N-1} u(n) \right) \Pr\{F_N\} \\
&\geq \left(N - \eta H \sum_{n=0}^{N-1} u(n) \right) \Pr\{F_N\}.
\end{aligned} \tag{52}$$

Hence, we obtain

$$\Pr\{F_N\} \leq \frac{\mathbb{E}[W_0]}{N - \eta H \sum_{n=0}^{N-1} u(n)}, \quad (53)$$

where $\mathbb{E}[W_0]$ can be obtained by setting $n = 0$ in (24) and taking expectation.

Proof of theorem 5 is completed.

E. Proof of Lemma 6

The round index is omitted for expression brevity. By taking the expectation over the gradient and number of active devices, the upper bound of $\mathbb{E}[\alpha^2(n)]$ is given as

$$\mathbb{E}[\alpha^2(n)] = \mathbb{E} \left[\left\| \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \mathbf{g}_k \right\|^2 \right] \leq \mathbb{E} \left[\|\mathbf{g}\|^2 + \frac{\sigma_g^2}{|\mathcal{K}|} \right] \stackrel{(48)}{\leq} \|\mathbf{g}\|^2 + \frac{2\sigma_g^2}{K\xi_a} \stackrel{(e)}{\leq} \|\mathbf{g}\|^2 + \sigma_g^2, \quad (54)$$

where (e) trivially holds when no less than 2 devices are expected to be active at each round such that $K\xi_a \geq 2$. By taking the expectation over selected element set ψ_n , $\mathbb{E}[V^2(n)]$ is upper bounded as below:

$$\begin{aligned} \mathbb{E}[V^2(n)] &= \mathbb{E} \left[\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} V_k^2 \right] = \mathbb{E}[V_k] = \mathbb{E} \left[\frac{1}{S_n} \sum_{s=1}^{S_n} ([\mathbf{g}^{\text{co}}]_s - M_k)^2 \right] \\ &= \mathbb{E} \left[\frac{1}{S_n} \sum_{s=1}^{S_n} [\mathbf{g}^{\text{co}}]_s^2 - M_k^2 \right] \leq \mathbb{E} \left[\frac{1}{S_n} \sum_{s=1}^{S_n} [\mathbf{g}^{\text{co}}]_s^2 \right] \\ &= \frac{1}{D} \sum_{d=1}^D [\mathbf{g}_k]_d^2 \frac{D}{S_n} \sum_{\psi_n \in \Omega_n} \frac{\mathbb{I}\{d \in \psi_n\}}{|\Omega_n|} \\ &= \frac{1}{D} \|\mathbf{g}_k\|^2 \frac{D}{S_n} \frac{\binom{D-1}{S_n-1}}{\binom{D}{S_n}} = \frac{1}{D} \|\mathbf{g}_k\|^2 \leq \frac{1}{D} (\|\mathbf{g}\|^2 + \sigma_g^2), \end{aligned} \quad (55)$$

where $M_k = \frac{1}{S_n} \sum_{s=1}^{S_n} [\mathbf{g}^{\text{co}}]_s$ is the mean of local sparse gradient. Building on the analysis above, $\mathbb{E}[\alpha^2(n)] \leq D\Gamma(n)$ and $\mathbb{E}[V^2(n)] \leq \Gamma(n)$ holds if the upper bound of GSI is chosen as $\Gamma(n) \geq \frac{1}{D} (\|\mathbf{g}\|^2 + \sigma_g^2)$. Last, the upper bound for CSI can be simply obtained by (48).

Proof of Lemma 6 is completed.

REFERENCES

- [1] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The roadmap to 6G: AI empowered wireless networks," *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, 2019.

- [2] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 19–25, 2020.
- [3] 3GPP, "Study on 5G system support for AI/ML-based services," 3GPP, Tech. Rep. TR 23.700-80, 2022.
- [4] T. Gafni, N. Shlezinger, K. Cohen, Y. C. Eldar, and H. V. Poor, "Federated learning: A signal processing perspective," *IEEE Signal Process. Mag.*, vol. 39, no. 3, pp. 14–41, 2022.
- [5] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 2020.
- [6] Z. Lin, X. Li, V. K. N. Lau, Y. Gong, and K. Huang, "Deploying federated learning in large-scale cellular networks: Spatial convergence analysis," *IEEE Trans. Wireless Commun.*, vol. 21, no. 3, pp. 1542–1556, 2022.
- [7] Q. Lan, H. S. Kang, and K. Huang, "Simultaneous signal-and-interference alignment for two-cell Over-the-Air computation," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1342–1345, 2020.
- [8] R. Ruby, H. Yang, and K. Wu, "Anti-jamming strategy for federated learning in internet of medical things: A game approach," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 888–899, 2023.
- [9] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, 2021.
- [10] D. Wen, M. Bennis, and K. Huang, "Joint parameter-and-bandwidth allocation for improving the efficiency of partitioned edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8272–8286, 2020.
- [11] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1935–1949, 2021.
- [12] N. Zhang and M. Tao, "Gradient statistics aware power control for Over-the-Air federated learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5115–5128, 2021.
- [13] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 453–467, 2021.
- [14] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 317–333, 2020.
- [15] D. Wen, K.-J. Jeon, and K. Huang, "Federated dropout—a simple approach for enabling federated learning on resource constrained devices," *IEEE Wireless Commun. Lett.*, vol. 11, no. 5, pp. 923–927, 2022.
- [16] Y. Jiang, S. Wang, V. Valls, B. J. Ko, W.-H. Lee, K. K. Leung, and L. Tassiulas, "Model pruning enables efficient federated learning on edge devices," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, 2022.
- [17] G. Yan, T. Li, S.-L. Huang, T. Lan, and L. Song, "AC-SGD: Adaptively compressed SGD for communication-efficient distributed learning," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 9, pp. 2678–2693, 2022.
- [18] P. Richtárik and M. Takáč, "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," *Mathematical Programming*, vol. 144, no. 1-2, pp. 1–38, 2014.
- [19] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, 2020.
- [20] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via Over-the-Air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, 2020.
- [21] M. M. Amiri and D. Gunduz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, 2020.
- [22] G. Zhu, J. Xu, K. Huang, and S. Cui, "Over-the-Air computing for wireless data aggregation in massive IoT," *IEEE Wireless Commun.*, vol. 28, no. 4, pp. 57–65, 2021.

- [23] X. Cao, G. Zhu, J. Xu, and K. Huang, "Optimized power control for Over-the-Air computation in fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7498–7513, 2020.
- [24] Z. Wang, J. Qiu, Y. Zhou, Y. Shi, L. Fu, W. Chen, and K. B. Letaief, "Federated learning via intelligent reflecting surface," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 808–822, 2021.
- [25] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-Air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.
- [26] X. Cao, G. Zhu, J. Xu, and S. Cui, "Transmission power control for Over-the-Air federated averaging at network edge," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1571–1586, 2022.
- [27] Z. Lin, Y. Gong, and K. Huang, "Distributed Over-the-Air computing for fast distributed optimization: Beamforming design and convergence analysis," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 274–287, 2022.
- [28] R. Hamdi, M. Chen, A. B. Said, M. Qaraqe, and H. V. Poor, "Federated learning over energy harvesting wireless networks," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 92–103, 2022.
- [29] X. Cao, G. Zhu, J. Xu, and K. Huang, "Cooperative interference management for Over-the-Air computation networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2634–2651, 2020.
- [30] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-spectrum communications - a tutorial," *IEEE Trans. Commun.*, vol. 30, no. 5, pp. 855–884, 1982.
- [31] K. Gilhousen, I. Jacobs, R. Padovani, A. Viterbi, L. Weaver, and C. Wheatley, "On the capacity of a cellular CDMA system," *IEEE Trans. Veh. Technol.*, vol. 40, no. 2, pp. 303–312, 1991.
- [32] E. Ozfatura, K. Ozfatura, and D. Gunduz, "Time-correlated sparsification for communication-efficient federated learning," in *Proc. of IEEE Int. Symp. Inf. Theory (ISIT)*, Melbourne, Australia, Jul. 12-20, 2021.
- [33] 3GPP Specifications, "Timing advance (TA) in LTE," Sep. 2010. [Online]. Available: <http://4g5gworld.com/blog/timingadvance-ta-lte>
- [34] I. Shomorony and A. S. Avestimehr, "Worst-case additive noise in wireless networks," *IEEE Trans. Inf. Theory*, vol. 59, no. 6, pp. 3833–3847, 2013.
- [35] X. Wei and C. Shen, "Federated learning over noisy channels: Convergence analysis and design examples," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 2, pp. 1253–1268, 2022.
- [36] M. Mohammadi Amiri and D. Gunduz, "Machine learning at the wireless edge: Distributed stochastic gradient descent Over-the-Air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, 2020.
- [37] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "BEV-SGD: Best effort voting SGD against Byzantine attacks for analog-aggregation-based federated learning over the air," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 18 946–18 959, 2022.
- [38] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge UK: Cambridge university press, 2005.
- [39] C. M. De Sa, C. Zhang, K. Olukotun, C. Ré, and C. Ré, "Taming the wild: A unified analysis of hogwild-style algorithms," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, Canada, Dec. 7-10, 2015.
- [40] D. Alistarh, T. Hoeffler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli, "The convergence of sparsified gradient methods," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal, Canada, Dec. 2-8, 2018.
- [41] T. R. Fleming and D. P. Harrington, *Counting processes and survival analysis*. Hoboken, NJ, USA: John Wiley & Sons, 2011.
- [42] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Montreal Canada, Dec. 2-8, 2018.
- [43] Z. Zhang, G. Zhu, R. Wang, V. K. Lau, and K. Huang, "Turning channel noise into an accelerator for Over-the-Air principal component analysis," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 7926–7941, 2022.