

Privacy-Enhanced Zero-Order Federated Learning via xMK-CKKS over Wireless Channels

Anthony Ayli
GIC, ESIB

Université Saint-Joseph de Beyrouth
Beirut, Lebanon
anthony.ayli@net.usj.edu.lb

Khalil Harris
GIC, ESIB

Université Saint-Joseph de Beyrouth
Beirut, Lebanon
khalil.harris1@usj.edu.lb

Jihad Fahs

Electrical and Computer Engineering
American University of Beirut
Beirut, Lebanon
jihad.fahs@aub.edu.lb

Mohamad Assaad

Laboratory of Signals and Systems (L2S)
Centralesupelec, University of Paris-Saclay
Paris, France
mohamad.assaad@centralesupelec.fr

Abstract—Homomorphic encryption (HE) enables privacy-preserving aggregation in federated learning (FL) by allowing the server to operate on encrypted data without decryption. Existing HE-over-the-air (OTA) methods mainly rely on single-key HE schemes and require channel estimation or pre-equalization to compensate for wireless fading. However, single-key HE remains vulnerable to honest-but-curious (HBC) clients holding the shared secret key, while multi-key HE provides stronger client-level security by assigning each device its own secret key. We propose a four-phase protocol that enables the aggregation of xMK-CKKS over a shared wireless channel without channel estimation. The protocol retransmits partial public keys and ciphertexts through the same channel realization, so that the dominant large-modulus encryption terms cancel algebraically during decryption. We integrate this protocol with zero-order FL over slowly varying LoS-dominant channels, where each device transmits a single encrypted scalar per round and the communication/encryption overhead is independent of the model dimension. We show that the residual noise induced by encryption and wireless aggregation preserves the standard convergence rate $O(1/\sqrt{K})$ up to a negligible noise floor, where K is the number of communication rounds. The protocol assumes a non-trusted server and is secure against HBC clients, preventing any client from recovering the local updates of other participants. Numerical results on MNIST validate the theoretical analysis.

I. INTRODUCTION

Federated learning (FL) [1] enables edge devices to collaboratively train a shared model without sharing their local data. However, the model updates, gradients, or gradient surrogates transmitted by the devices may still reveal sensitive information and remain vulnerable to inference attacks [2]. Over-the-air (OTA) computation [3] and zero-order (ZO) gradient estimation [4], [5] reduce the uplink communication cost by exploiting the superposition property of the wireless channel. In particular, ZO FL methods [4], [5] can reduce the communication load per-device to one or two scalars per round. Nevertheless, these analog transmissions are non-encrypted and therefore do not provide cryptographic protection.

Homomorphic encryption (HE) offers a natural mechanism for privacy-preserving aggregation, since it allows the server to compute directly on encrypted data. Existing HE-over-the-air methods [6], [7] combine single-key HE with over-the-air aggregation but rely on channel estimation, pre-equalization, or beamforming to compensate for wireless fading. Moreover, because single-key HE places clients in a shared decryption domain, an honest-but-curious (HBC) client holding the shared secret key may decrypt or infer other clients updates.

Multi-key HE addresses this limitation by assigning each device its own secret key. In particular, xMK-CKKS [8] requires all participating devices to contribute partial decryption shares and remains secure against HBC devices. However, secure ring learning with errors (RLWE) based implementations require large ciphertext moduli, e.g., parameter sets such as $(n = 4096, q \approx 2^{109})$ and $(n = 8192, q \approx 2^{218})$ in CKKS deployments [9]. At this scale, channel-estimation errors and wireless distortions can leave q -scale decryption residuals.

This motivates the central question addressed in this paper: can multi-key HE be combined with over-the-air aggregation without estimating or compensating the wireless channel? We answer this question by proposing a four-phase protocol in which the partial public keys and ciphertexts are transmitted through the same channel realization. As a result, the dominant large-modulus encryption terms experience the same channel coefficients and cancel algebraically during decryption. The remaining channel and encryption noise are not amplified by q ; instead, they enter the learning algorithm as bounded perturbations.

The proposed protocol is particularly suited to slowly varying LoS-dominant links, such as short-range indoor THz communication scenarios [10], [11]. Such links are often highly directional and dominated by a small number of propagation paths. In fixed or low-mobility deployments, the effective channel can remain approximately constant over the short duration of the protocol phases.

The main contributions of this paper are as follows:

Anthony Ayli, Khalil Harris, and Jihad Fahs were supported by PHC CEDRE 2025, projet code: 52201TK.

- We propose a four-phase over-the-air protocol that enables xMK-CKKS aggregation over a shared wireless channel without channel state information (CSI) acquisition or pre-equalization.
- We show that retransmitting the partial public keys and ciphertexts through the same channel realization causes the dominant q -scale encryption terms to cancel algebraically during decryption, avoiding q -amplified residual errors.
- We integrate the protocol with ZO FL over slowly varying LoS-dominant channels, where each device transmits a single encrypted scalar per round and the communication/encryption overhead is independent of the model dimension.
- We prove that the residual encryption and channel noise preserves the $O(1/\sqrt{K})$ convergence rate, where K is the number of communication rounds, up to a negligible noise floor.
- We show that the protocol is secure against a non-trusted server and HBC devices. We validate the analysis using MNIST experiments under fading-channel models.

The remainder of the paper is organized as follows. Section II reviews existing HE-over-the-air methods and their limitations. Sections III and IV present the system model and the proposed protocol. Section V provides the convergence analysis results. Section VI presents numerical results, and Section VII concludes the paper.

II. RELATED WORK

Two recent methods combine HE with over-the-air computation. Wang et al. [6] proposed HEAirFed, which uses single-key CKKS: each device encrypts its full gradient vector and the server aggregates over the air using MIMO beamforming with CSI at the devices and the server. Xie et al. [7] proposed AirHE, a single-key LWE scheme that transmits ciphertext digits as nested-lattice codewords and relies on channel estimation and pre-equalization. Both show that HE can be combined with over-the-air aggregation in single-key settings, but did not address the multi-key setting of this paper, where each device keeps an independent secret key. Table I shows that the proposed protocol adopts a multi-client architecture, providing stronger resistance against HBC clients and collision-related attacks. Moreover, the scheme follows the Microsoft SEAL recommended RLWE parameters [9], where the ring dimension is set to $n = 4096$ or $n = 8192$, values that are considered computationally secure against practical attacks.

Why Existing Methods Do Not Extend to xMK-CKKS

Two limitations separate these methods from the setting of this paper. The first is cryptographic and holds for any modulus: both use single-key HE, so any HBC client holding the shared key can decrypt every other client's update. Single-key HE protects against external adversaries, but does not give resistance to client-level collusion. xMK-CKKS assigns each device its own secret key and requires all devices to

TABLE I: Comparison of HE-over-the-air methods.

	AirHE [7]	HEAirFed [6]	Proposed
HE scheme	LWE	CKKS	xMK-CKKS
Key structure	Single	Single	Multi-key
RLWE dim. n	9	4096	4096, 8192
Ciph. mod. q	6560	2^{30}	$2^{109}, 2^{218}$
CSI required	Yes	Yes	No
Pre-equal.	Yes	Yes (beamf.)	No
Enc. payload	Scalar	$\nabla F \in \mathbb{R}^d$	Scalar
HBC client	Vuln.	Vuln.	Secure

contribute decryption shares, which gives resistance against HBC devices.

The second is tied to the modulus. Single-key CKKS with beamforming scales to a large modulus without difficulty. The multi-key case is different: the large gap between the modulus q and the scaling factor Λ means that any residual channel-estimation error is multiplied by the q -scale ciphertext terms and buries the Λ -scale message (details are presented in the appendix). For example, we will show in Fig. 3 of Section VI that both zero-forcing (ZF) and MMSE pre-equalization diverge from the first iteration at $q \approx 2^{109}$. To overcome this limitation, the proposed protocol retransmits the partial public keys and ciphertexts through the same channel realization, allowing the q -scaled terms to cancel algebraically while only small encryption and channel noise terms remain.

III. SYSTEM MODEL

Throughout, $i, j \in \{1, \dots, N\}$ index the devices and $k \in \{0, \dots, K\}$ the communication rounds.

A. Wireless Zero-Order Federated Learning Model

Consider an FL framework with N edge devices and a central server coordinating the training of a global model $\theta \in \mathbb{R}^d$ over a wireless network. Each device trains on its private local dataset. Let $\mathcal{N} = \{1, \dots, N\}$ denote the set of devices, and let $F_i : \mathbb{R}^d \rightarrow \mathbb{R}$ be the local loss associated with device i . The global objective is to minimize

$$F(\theta) = \sum_{i=1}^N F_i(\theta), \quad F_i(\theta) = \mathbb{E}_{\xi_i \sim \mathcal{D}_i} [f_i(\theta, \xi_i)], \quad (1)$$

where ξ_i is sampled from the local data distribution $\mathcal{D}_i$. The functions F , F_i , and f_i are allowed to be nonconvex.

Let $h_{i,k}$ denote the channel coefficient between device i and the server during communication round k . We consider slowly varying block-fading channels with a nonzero mean due to a LoS component:

$$\mathbb{E}[h_{i,k}] = \mu_i \neq 0, \quad \mathbb{E}[h_{i,k}^2] = \Omega_i, \quad 1 \leq i \leq N. \quad (2)$$

The channel coefficients are assumed independent, but not necessarily identically distributed, across devices. Within each round k , $h_{i,k}$ remains constant for each device i over all protocol phases. Thus, each round corresponds to one fading block, whereas independent fading is assumed from one round to the next. This model is motivated by short-range indoor THz links with fixed or low-mobility devices. Such links are often

LoS-dominant and highly directional, with sparse multipath and high Rician K -factors. When the transmitter, receiver, and dominant scatterers remain stationary over the duration of a communication round, the coherence time can cover all phases of the proposed protocol [12], [13].

We consider a ZO method in which the channel disturbance is incorporated into the learning process, in the same spirit as [4]. At each round k , every device i computes the standard two-point ZO difference of its local loss [4], [14], [15]:

$$\Delta f_{i,k} = f_i(\theta_k + \gamma_k \Phi_k, \xi_{i,k}) - f_i(\theta_k - \gamma_k \Phi_k, \xi_{i,k}), \quad (3)$$

where $\Phi_k = (\Phi_k^1, \dots, \Phi_k^d)^\top$ is a perturbation vector with i.i.d. entries satisfying $\mathbb{E}[(\Phi_k^j)^2] = b_1$ and $\|\Phi_k\| \leq b_2$, and $\gamma_k > 0$ is the smoothing parameter. The perturbation sequence is generated randomly and is made available to all devices.

When the channel has a nonzero mean, $\mathbb{E}[h_{i,k}] = \mu_i \neq 0$, in the ZO FL scheme [4], each device knows, or estimates, the long-term channel mean μ_i and transmits each round the single scalar $\Delta f_{i,k}/\mu_i$. The server receives

$$Y_k = \sum_{i \in \mathcal{N}} \frac{\Delta f_{i,k}}{\mu_i} h_{i,k} + n_k, \quad (4)$$

where $n_k \sim \mathcal{N}(0, \sigma_n^2)$, and broadcasts Y_k to all devices. Each device then forms the ZO gradient estimate

$$g_k = \Phi_k Y_k. \quad (5)$$

The channel coefficient $h_{i,k}$ is not estimated or removed; it enters the gradient estimate as a perturbation. In [4], it is shown that including the channel disturbance in the learning process does not change the convergence rate, which remains of order $O(1/\sqrt{K})$ in nonconvex settings. In Theorem 2, we extend the analysis to the encrypted setting and show that the proposed protocol preserves the same rate up to an explicit noise floor, namely $O(1/\sqrt{K}) + \rho$, where ρ is negligible under the considered parameter regime.

B. Why Naive OTA xMK-CKKS Fails

We now explain why directly reusing digitally generated xMK-CKKS public keys over a wireless channel fails. Suppose first that, in round k , the partial public keys are transmitted over the air through the same channel realization $h_{i,k}$. The server receives

$$\begin{aligned} \tilde{b}_k &= \sum_{i=1}^N h_{i,k} b_i + w_{b,k} \\ &= - \left(\sum_{i=1}^N h_{i,k} s_i \right) a + \sum_{i=1}^N h_{i,k} e_i + w_{b,k}. \end{aligned}$$

Define

$$\tilde{S}_k := \sum_{i=1}^N h_{i,k} s_i, \quad \tilde{E}_{\text{agg},k} := \sum_{i=1}^N h_{i,k} e_i + w_{b,k}. \quad (6)$$

Then the channel-weighted cancellation identity becomes

$$\tilde{b}_k + \tilde{S}_k a = \tilde{E}_{\text{agg},k}, \quad (7)$$

where the large q -scale term cancels, since $\tilde{b}_k$ and $\tilde{S}_k$ are generated with the same channel realization.

The failure occurs if the public keys are aggregated once during an initial setup phase and then reused in later rounds. In that case, the server receives $\tilde{b}_0$, which is tied to the setup channel realization $h_{i,0}$, whereas the decryption shares in round k produce $\tilde{S}_k$, which is tied to the current channel realization $h_{i,k}$. The cancellation becomes

$$\tilde{b}_0 + \tilde{S}_k a = (\tilde{S}_k - \tilde{S}_0) a + \tilde{E}_{\text{agg},0}.$$

where $\tilde{S}_0 := \sum_{i=1}^N h_{i,0} s_i$ is the channel-weighted secret from the setup round, so that $\tilde{b}_0 = -\tilde{S}_0 a + \tilde{E}_{\text{agg},0}$. The first term on the right-hand side is no longer an encryption-noise term. It is a channel-mismatch residual multiplied by the public polynomial $a \in R_q$. Since the remaining coefficients are reduced modulo q , with $q \approx 2^{109}$ or $q \approx 2^{218}$ following the Microsoft SEAL recommendations [9], the residual term may reach a magnitude proportional to q .

By contrast, the encoded message has a scale of approximately Λ , with $\Lambda = 2^{40}$. Thus, a q -scale residual can be roughly $q/\Lambda \approx 2^{70}$ times larger than the encoded signal, which destroys decryption. The proposed protocol avoids this failure by retransmitting the partial public keys in every communication round, through the same channel realization that carries the encrypted scalar and the decryption shares. The proposed protocol does not try to estimate or invert the wireless channel; instead, it forces the public-key and decryption-share terms to experience the same fading coefficients, so that the large-modulus terms cancel algebraically. The complete protocol is described in the next section.

IV. PROPOSED PROTOCOL

The key observation is that the cancellation identity present in Eq. (21) fails over the air because the public keys b_i are aggregated once during setup, while the channel coefficients $h_{i,k}$ change every round. To fix this, we retransmit b_i every round through the same channel realization $h_{i,k}$ that carries the ciphertext and decryption shares. This ensures that the cancellation identity survives in channel-weighted form.

The protocol consists of four phases per round, illustrated in Fig. 1. We add the following assumption on the HE randomness.

Assumption 1. (HE randomness): The fresh randomness $v_{i,k}$, $e_{0,i,k}$, $e_{1,i,k}$, $e_{j,k}^*$ and the channel noises $w_{b,k}$, $w_{0,k}$, $w_{1,k}$, $w_{D,k}$ are mutually independent, zero-mean, and independent of Φ_k , $h_{\cdot,k}$, $\xi_{\cdot,k}$.

We assume a perfect downlink: messages broadcasted by the server are received by all devices without errors. This is a standard assumption in OTA FL [3], [4].

A. Phase 1: Fresh Key Aggregation

Each client i transmits its partial public key b_i over the uplink. The server receives:

$$\tilde{b}_k = \sum_{i=1}^N h_{i,k} b_i + w_{b,k} = -\tilde{S}_k a + \tilde{E}_{\text{agg},k}, \quad (8)$$

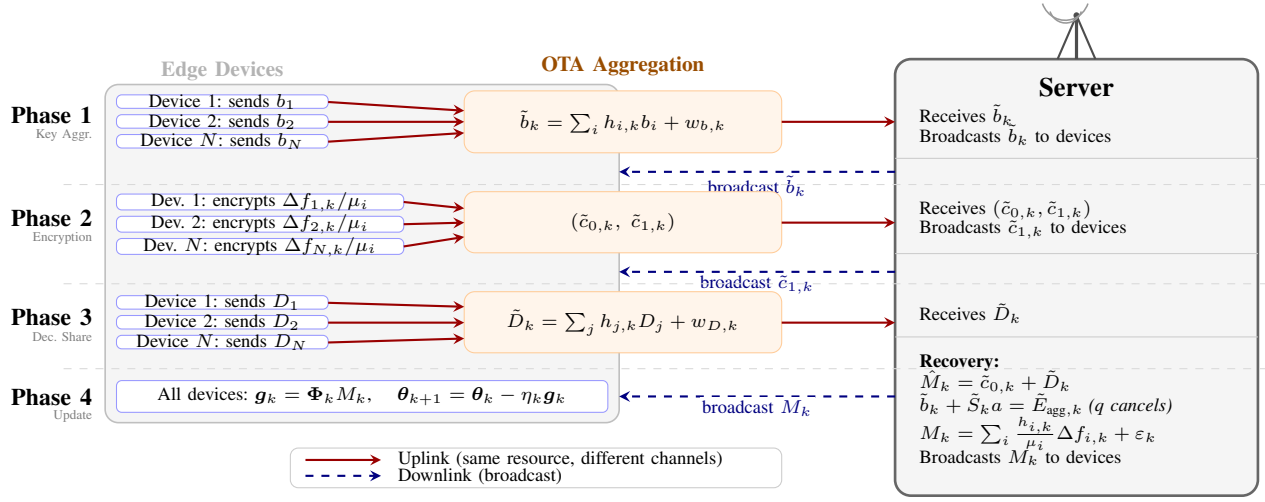


Fig. 1: The proposed four-phase protocol for one round k . All devices transmit simultaneously on the same wireless resource. Each device holds its own secret key s_i . The server does not estimate any channel coefficient. In Phase 4, the q -scale terms cancel algebraically because $\tilde{b}_k$ and $\tilde{S}_k$ use the same channel realization $h_{i,k}$.

where $\tilde{S}_k$ and $\tilde{E}_{\text{agg},k}$ are given by Eq. (6), and the channel-weighted cancellation identity (7) holds.

B. Phase 2: Encryption

The server broadcasts $\tilde{b}_k$ to all clients. Each client i encodes its ZO estimate as $m_{i,k} = \lfloor \Lambda \Delta f_{i,k} / \mu_i \rfloor$ and transmits the ciphertext pair $(c_0^{(i)}, c_1^{(i)})$:

$$c_0^{(i)} = v_{i,k} \tilde{b}_k + m_{i,k} + e_{0,i,k}, \quad c_1^{(i)} = v_{i,k} a + e_{1,i,k}, \quad (9)$$

where $v_{i,k}$ denotes a fresh encryption randomness, while $e_{0,i,k}$ and $e_{1,i,k}$ represent small error polynomials sampled according to a Gaussian distribution, as described in the appendix. The server receives the aggregated ciphertexts over the air:

$$\tilde{c}_{0,k} = V_k \tilde{b}_k + \sum_i h_{i,k} m_{i,k} + \sum_i h_{i,k} e_{0,i,k} + w_{0,k}, \quad (10)$$

$$\tilde{c}_{1,k} = V_k a + \sum_i h_{i,k} e_{1,i,k} + w_{1,k}, \quad (11)$$

where $V_k := \sum_i h_{i,k} v_{i,k}$.

C. Phase 3: Decryption Share

The server broadcasts $\tilde{c}_{1,k}$. Each client j computes and transmits a partial decryption share $D_{j,k} = s_j \tilde{c}_{1,k} + e_{j,k}^*$, where s_j denotes the secret key of client j and $e_{j,k}^*$ is an additional Gaussian noise term used to protect the privacy of the partial decryption share. The server receives:

$$\tilde{D}_k = \tilde{S}_k \tilde{c}_{1,k} + \sum_j h_{j,k} e_{j,k}^* + w_{D,k}. \quad (12)$$

D. Phase 4: Recovery

The server forms $\hat{M}_k := \tilde{c}_{0,k} + \tilde{D}_k$. Substituting Eq. (10), Eq. (11), and Eq. (12):

$$\begin{aligned} \hat{M}_k &= V_k \tilde{b}_k + \tilde{S}_k (V_k a + \sum_i h_{i,k} e_{1,i,k} + w_{1,k}) \\ &\quad + \sum_i h_{i,k} m_{i,k} + \sum_i h_{i,k} e_{0,i,k} + w_{0,k} \\ &\quad + \sum_j h_{j,k} e_{j,k}^* + w_{D,k} \\ &= V_k (\tilde{b}_k + \tilde{S}_k a) + \sum_i h_{i,k} m_{i,k} + \zeta_k, \end{aligned} \quad (13)$$

where

$$\begin{aligned} \zeta_k &:= \tilde{S}_k \sum_i h_{i,k} e_{1,i,k} + \tilde{S}_k w_{1,k} + \sum_i h_{i,k} e_{0,i,k} + w_{0,k} \\ &\quad + \sum_j h_{j,k} e_{j,k}^* + w_{D,k}. \end{aligned} \quad (14)$$

E. Ring-Scale Cancellation

The term $V_k (\tilde{b}_k + \tilde{S}_k a)$ contains two q -scale components $V_k \tilde{b}_k$ and $\tilde{S}_k V_k a$, each of order $q \approx 2^{109}$ or 2^{218} , far larger than the message $\Lambda \sum_i h_{i,k} \Delta f_{i,k}$ of order $\Lambda \approx 2^{40}$. Substituting $\tilde{b}_k = -\tilde{S}_k a + \tilde{E}_{\text{agg},k}$ from Eq. (8) gives $V_k (\tilde{b}_k + \tilde{S}_k a) = V_k \tilde{E}_{\text{agg},k}$: the q -scale terms cancel exactly, leaving a product of two small polynomials whose second moment is independent of q and of the order $n^2 \sigma_e^2 (\sum_i \Omega_i)^2$ (more details are found in [16, Appendix B]). The remaining expression is:

$$\hat{M}_k = \Lambda \sum_i \frac{h_{i,k}}{\mu_i} \Delta f_{i,k} + \hat{\varepsilon}_k, \quad \text{and} \quad \hat{\varepsilon}_k = V_k \tilde{E}_{\text{agg},k} + \zeta_k, \quad (15)$$

where ζ_k is given in Eq. (14). Decoding by $1/\Lambda$:

$$M_k = \frac{1}{\Lambda} \hat{M}_k = \sum_i \frac{h_{i,k}}{\mu_i} \Delta f_{i,k} + \varepsilon_k, \quad \varepsilon_k := \frac{1}{\Lambda} \hat{\varepsilon}_k. \quad (16)$$

Each ring coefficient is an integer in $\{0, \dots, q-1\}$, transmitted as an analog symbol; rounding the real-valued superposition recovers the correct ring element, and the algebraic identities of Phases 1-4 hold under modular arithmetic, when the noise $\hat{\varepsilon}$ is small with high probability [8].

F. Security of the OTA Protocol

Retransmitting b_i every round raises the question of whether the server can extract individual b_i from the K superpositions $\tilde{b}_k = \sum_i h_{i,k} b_i + w_{b,k}$. The system has K observations and $KN + N$ unknowns (channel coefficients and partial keys) and is under-determined since the server never observes individual $h_{i,k}$. Even if an adversary recovers each b_i , this information is already publicly available in digital xMK-CKKS. Recovering the corresponding secret key s_i from $b_i = -s_i a + e_i$ remains computationally hard under the RLWE assumption, whose security is reducible to hard lattice problems such as the Closest Vector Problem (CVP) [8]. The OTA setting therefore provides two layers of protection: the channel layer prevents the server from separating individual transmissions, while the cryptographic layer protects each b_i even if it were recovered. In the digital setting only the second layer is present.

G. Gradient Estimator

Each device forms the gradient estimate from the M_k :

$$\mathbf{g}_k = \Phi_k M_k = \Phi_k \left(\sum_i \frac{h_{i,k}}{\mu_i} \Delta f_{i,k} + \varepsilon_k \right), \quad (17)$$

matching the structure of Eq. (5) with the channel noise n_k replaced by the decoded HE noise ε_k . The model update is $\theta_{k+1} = \theta_k - \eta_k \mathbf{g}_k$.

H. Algorithm

Algorithm 1 summarizes the main steps of the proposed EEZOFL protocol based on xMK-CKKS.

Algorithm 1 EEZOFL

Input: initial values θ_0 , η_0 , γ_0 , channel mean μ_i

- 1: **for** $k = 0, \dots, K$ **do**
 - 2: Each device i computes $\Delta f_{i,k}$ using its local data and encodes $m_{i,k} = \lfloor \Lambda \Delta f_{i,k} / \mu_i \rfloor$.
 - 3: *Phase 1:* Each device transmits b_i . The server receives $\tilde{b}_k$ given in Eq. (8) and broadcasts it.
 - 4: *Phase 2:* Each device transmits $(c_0^{(i)}, c_1^{(i)})$ given in Eq. (9). The server receives $(\tilde{c}_{0,k}, \tilde{c}_{1,k})$ and broadcasts $\tilde{c}_{1,k}$.
 - 5: *Phase 3:* Each device j transmits $D_{j,k} = s_j \tilde{c}_{1,k} + e_{j,k}^*$. The server receives $\tilde{D}_k$ given in Eq. (12).
 - 6: *Phase 4:* The server computes M_k given in Eq. (16) and broadcasts M_k to all devices.
 - 7: Each device multiplies the received value by Φ_k to obtain $\mathbf{g}_k$ given in Eq. (17).
 - 8: Each device updates the model $\theta_{k+1} = \theta_k - \eta_k \mathbf{g}_k$.
 - 9: **end for**
-

I. Encryption Overhead and Feasibility

Table II presents two secure xMK-CKKS parameter sets, corresponding to $(n, \log_2 q) = (4096, 109)$ and $(n, \log_2 q) = (8192, 218)$, following the HE Standard [17] and Microsoft SEAL [9] recommendations, both providing an estimated 128-bit classical security level.

The uplink communication per round is $4n \lceil \log_2 q \rceil$ bits per device. This includes the transmission of the aggregated public key $\tilde{b}_k$ in Phase 1, the two ciphertext components $(c_0^{(i)}, c_1^{(i)})$ in Phase 2, and the partial decryption share $D_{i,k}$ in Phase 3. Therefore, the transmission time per device at bandwidth B is given by $T_{\text{tx}} = \frac{4n \lceil \log_2 q \rceil}{B}$. Hence the transmission time reported in Table II corresponds to the communication performed during the first three phases of the proposed protocol described in Section IV. Therefore, increasing the security level also increases the communication overhead, which directly impacts the transmission time. At $B = 1$ THz, both configurations transmit in less than $8 \mu\text{s}$, which remains well within the coherence time of static indoor THz channels.

The encryption benchmarks were obtained on an Intel Core i9-14900HX platform (24 cores, 32 threads, 32 GB DDR5 RAM). Increasing the parameter set from $(n, \log_2 q) = (4096, 109)$ to $(8192, 218)$ raises the average encryption time from 23.92 ms to 55.56 ms due to the increased complexity of polynomial arithmetic over larger rings. Despite this increase, the encryption overhead remains practical for FL applications. From a computational perspective, encryption and decryption-share operations have a complexity of $O(n \log n)$ per device per round due to the use of the number theoretic transform (NTT). This explains the increase in encryption latency when moving from $(n, \log_2 q) = (4096, 109)$ to $(8192, 218)$. In addition, the decoded noise variance scales as $\sigma_{\varepsilon}^2 \propto n/\Lambda^2$ (as shown in the full paper [16, Appendix B]), while the resulting noise floor ρ remains negligible for both parameter sets.

Finally, as shown in Table II, the ciphertext size increases from 109 KB for $n = 4096$ to 446 KB for $n = 8192$, while the plaintext is represented by a floating-point value of only 8 bytes. This corresponds to communication expansion factors of approximately 1.36×10^4 and 5.58×10^4 , respectively. Although significant, such an increase is inherent to HE-based systems, where higher security levels and larger parameter sets lead to larger ciphertexts. The results therefore highlight the classical tradeoff between security and communication efficiency in HE.

V. CONVERGENCE ANALYSIS

We retain Assumptions 3.1–3.4 of [4]: L -smoothness of the global objective function $F(\cdot)$ and bounded Hessian $\|\nabla^2 F_i\|_2 \leq b$ (Assumption 3.1), Lipschitz continuity of $f_i(\cdot, \xi_i)$ with constant L (Assumption 3.2), step-size conditions $\sum_k \eta_k \gamma_k = \infty$, $\sum_k \eta_k \gamma_k^3 < \infty$, $\sum_k \eta_k^2 \gamma_k^2 < \infty$ (Assumption 3.3), and perturbation vector Φ_k with i.i.d. entries satisfying $\mathbb{E}[(\Phi_k^j)^2] = b_1$ and $\|\Phi_k\| \leq b_2$ (Assumption 3.4). We assume a perfect downlink: the server

TABLE II: HE parameter sets and protocol overhead.

	$n=4096$	$n=8192$
$\log_2 q$	109	218
Λ	2^{40}	2^{40}
Security (HE Std.)	128-bit	128-bit
Ciphertext size	109 KB	446 KB
Encryption time (mean)	23.92 ms	55.56 ms
Uplink / round	1.8 Mbit	7.1 Mbit
T_{tx} ($B=1$ THz)	$1.8 \mu s$	$7.1 \mu s$
Storage / device	109 KB	446 KB
Encrypt (NTT)	$O(n \log n)$ per round	
ρ (noise floor)	$\sim 2 \times 10^{-16}$	$\sim 4 \times 10^{-16}$

broadcasts $\tilde{b}_k$, $\tilde{c}_{1,k}$, and M_k to all devices without error. Let $\mathcal{H}_k = \{\boldsymbol{\theta}_0, \xi_0, \dots, \boldsymbol{\theta}_{k-1}, \xi_{k-1}, \boldsymbol{\theta}_k\}$ denote the history up to and including the model $\boldsymbol{\theta}_k$ but excluding the current sample ξ_k .

A. Preliminary results

Proposition 1. (Noise budget) Under Assumption 1 and the system model in (4), there exists a B_ϵ^2 such that $\hat{\epsilon}_k$ defined in Eq. (15) satisfies

$$\mathbb{E}[\hat{\epsilon}_k] = 0, \quad \mathbb{E}[\|\hat{\epsilon}_k\|^2 | \mathcal{H}_k] \leq B_\epsilon^2, \quad (18)$$

where the expression of B_ϵ^2 is given in [16, Appendix B]. The decryption correctness holds with probability greater than $1 - 4B_\epsilon^2/q^2$. The per-coefficient decoded noise variance is:

$$\bar{\sigma}_\epsilon^2 := \frac{B_\epsilon^2}{n\Lambda^2} = O\left(\frac{n\sigma_e^2(\sum_{i=1}^N \Omega_i)^2}{\Lambda^2}\right). \quad (19)$$

Proof. See [16, Appendix B]. $\square$

Eq. (19) implies that $\bar{\sigma}_\epsilon^2$ grows linearly in the ring dimension n and quadratically in the aggregated channel power, while remaining independent of the ciphertext modulus q .

The noise budget depends on the channel statistics only through the aggregate sums $\sum_i \Omega_i$, $\sum_i \Omega_i^2$, and $\sum_i \mu_i^4$. For example, in the case of identical channels, $\Omega_i = \Omega$ for all i and Eq. (19) reduces to $\bar{\sigma}_\epsilon^2 \approx \{6 \times 10^{-17} \Omega^2, 3 \times 10^{-17} \Omega^2\}$ for the parameters of Table II corresponding to $n = \{4096, 8192\}$ respectively. In both cases, the decoded noise variance is more than sixteen orders of magnitude below typical channel noise, so the encryption layer is invisible to the learning algorithm.

Lemma 1. (Bias and second moment) Under Assumptions 3.1–3.4 and Assumption 1,

$$\begin{aligned} \mathbb{E}[\mathbf{g}_k | \mathcal{H}_k] &= c_1 \gamma_k (\nabla F(\boldsymbol{\theta}_k) + \boldsymbol{\delta}_k) \\ \mathbb{E}[\|\mathbf{g}_k\|^2 | \mathcal{H}_k] &\leq \tilde{C}'_\gamma \gamma_k^2 + \tilde{C}'_\epsilon, \end{aligned}$$

with $\|\boldsymbol{\delta}_k\| \leq c_3 \gamma_k$, and where $c_1 := 2b_1$, $c_3 := b b_2^3 N / (2b_1)$, and

$$\tilde{C}'_\gamma = 4L^2 b_2^4 \left(\sum_{i=1}^N \frac{\Omega_i}{\mu_i^2} + N(N-1) \right), \quad \tilde{C}'_\epsilon = b_2^2 \bar{\sigma}_\epsilon^2.$$

Proof. See [16, Appendix C]. $\square$

B. Asymptotic Convergence Rate

Next, we present our main convergence results.

Theorem 1. Under Assumptions 3.1–3.4 and Assumption 1, if $\sum_k \eta_k^2 < \infty$, then $\lim_{k \rightarrow \infty} \mathbb{E}[\|\nabla F(\boldsymbol{\theta}_k)\|^2] = 0$.

Proof. See [16, Appendix D]. $\square$

Theorem 2. Let $\eta_k = \eta_0 K^{-1/4}$, $\gamma_k = \gamma_0 K^{-1/4}$, where $\eta_0, \gamma_0 > 0$. Under Assumptions 3.1–3.4 and Assumption 1, after K iterations:

$$\min_{k=1:K} \mathbb{E}[\|\nabla F(\boldsymbol{\theta}_k)\|^2] \leq \frac{R}{\sqrt{K}} + \rho,$$

where $R := \frac{2\hat{\Delta}}{c_1 \eta_0 \gamma_0} + c_3^2 \gamma_0^2 + \frac{\tilde{C}'_\gamma L \eta_0 \gamma_0}{c_1}$, $\hat{\Delta} := F(\boldsymbol{\theta}_0) - F(\boldsymbol{\theta}^*)$ such that $F(\boldsymbol{\theta}^*) = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta}) > -\infty$, and

$$\rho := \frac{\tilde{C}'_\epsilon L \eta_0}{c_1 \gamma_0} = \frac{L b_2^2 \bar{\sigma}_\epsilon^2 \eta_0}{2b_1 \gamma_0} \quad (20)$$

is the noise floor introduced by the encryption. $\tilde{C}'_\gamma$ and $\tilde{C}'_\epsilon$ are the same constants defined in Lemma 1.

Furthermore, for any $\epsilon > \rho$ and $0 < \beta < 1$, if $K = R^2 / (\epsilon\beta - \rho)^2$, then $\Pr(\min_{k=1:K} \|\nabla F(\boldsymbol{\theta}_k)\|^2 < \epsilon) \geq 1 - \beta$.

Proof. The global objective F is L -smooth by Assumption 3.1 of [4], which gives the descent inequality

$$F(\boldsymbol{\theta}_{k+1}) \leq F(\boldsymbol{\theta}_k) - \eta_k \langle \nabla F(\boldsymbol{\theta}_k), \mathbf{g}_k \rangle + \frac{L}{2} \eta_k^2 \|\mathbf{g}_k\|^2.$$

Taking $\mathbb{E}[\cdot | \mathcal{H}_k]$, applying Lemma 1, summing over $1 \leq k \leq K$, and substituting $\eta_k = \eta_0 K^{-1/4}$ and $\gamma_k = \gamma_0 K^{-1/4}$ yields

$$\min_{k=1:K} \mathbb{E}[\|\nabla F(\boldsymbol{\theta}_k)\|^2] \leq \frac{R}{\sqrt{K}} + \rho,$$

where the $R/\sqrt{K}$ term originates from the initial gap, the bias, and the gradient's second moment, while the constant ρ originates from the decoded HE noise $\tilde{C}'_\epsilon$. Since $\min_{k=1:K} \|\nabla F(\boldsymbol{\theta}_k)\|^2$ is nonnegative, Markov's inequality gives, for any $\epsilon > \rho$,

$$\Pr\left(\min_{k=1:K} \|\nabla F(\boldsymbol{\theta}_k)\|^2 \geq \epsilon\right) \leq \frac{1}{\epsilon} \left(\frac{R}{\sqrt{K}} + \rho \right),$$

so that, taking complements, $\Pr(\min_{k=1:K} \|\nabla F(\boldsymbol{\theta}_k)\|^2 < \epsilon) \geq 1 - \frac{1}{\epsilon} (R/\sqrt{K} + \rho)$; setting $\frac{1}{\epsilon} (R/\sqrt{K} + \rho) = \beta$ gives $K = R^2 / (\epsilon\beta - \rho)^2$. The detailed proof is given in [16, Appendix D]. $\square$

Theorem 2 implies that the algorithm converges to a neighborhood of size ρ at rate $O(1/\sqrt{K})$, where ρ is given in Eq. (20). For the specific setup used in our experiments, namely the HE parameters of Table II ($\Lambda = 2^{40}$, $\sigma_e = 3.2$, $N = 10$), identical channels with $\mu_i = \mu$ and $\Omega_i = \Omega$, and step sizes $\eta_0 = \gamma_0 = 0.05$, the decoded noise variance is $\bar{\sigma}_\epsilon^2 \approx 4 \times 10^{-16} \Omega^2$, giving a noise floor of order $\rho \approx 10^{-16}$. This is more than sixteen orders of magnitude below the channel noise variance σ_n^2 , and is therefore negligible for all practical purposes. The number of iterations needed to reach $\min_{k=1:K} \|\nabla F(\boldsymbol{\theta}_k)\|^2 < \epsilon$ is then $K \approx R^2 / (\epsilon\beta)^2$, the same as for the non-encrypted algorithm.

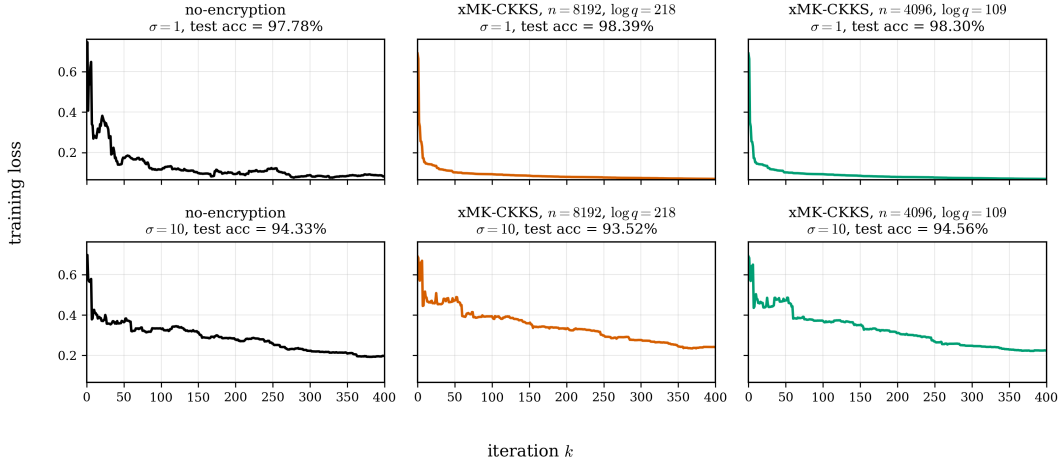


Fig. 2: Training loss on MNIST 0-vs-1 under the proposed protocol for $n \in \{4096, 8192\}$ and the non-encrypted baseline, in both channel regimes $\sigma \in \{1, 10\}$.

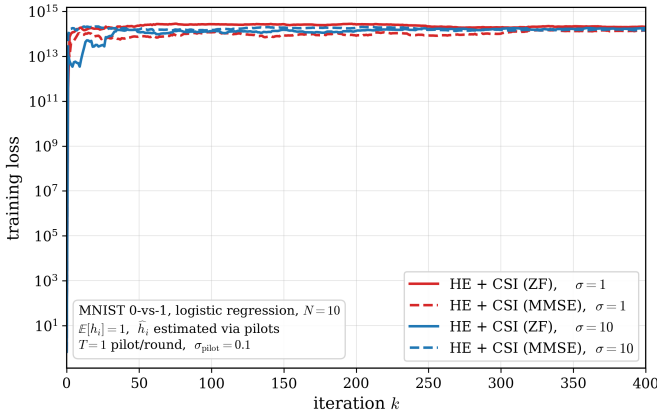


Fig. 3: Training loss under HE with CSI-based pre-equalization at $q \approx 2^{109}$ ($\sigma_{\text{pilot}} = 0.1$). Both ZF and MMSE equalization do not converge for either channel variance, since the residual estimation error is multiplied by q at decryption.

VI. NUMERICAL RESULTS

We evaluate the proposed protocol on MNIST 0-vs-1 binary classification with a logistic regression model ($d = 785$). The channel coefficient is $h_{i,k} \sim \mathcal{N}(\mu_i, \sigma_i^2)$, $1 \leq i \leq N$, with $\mu_i = \mu = 1$ and $\sigma_i = \sigma \in \{1, 10\}$, giving $\Omega_i = \Omega = 1 + \sigma^2$; the channel noise standard deviation is $\sigma_n = 1$. Step sizes are $\eta_k = \eta_0(1+k)^{-0.50}$ and $\gamma_k = \gamma_0(1+k)^{-0.25}$, $1 \leq k \leq K$, with $\eta_0 = \gamma_0 = 0.05$, batch size 128, $K = 400$ rounds, $N = 10$ devices.

Fig. 2 shows the training loss under the proposed protocol for the two HE setups of Table II and the non-encrypted baseline. Within each regime, the encrypted and non-encrypted configurations follow comparable trajectories and reach test accuracies that differ by less than 1% (Table III), confirm-

ing that the decoded HE noise $\bar{\sigma}_\epsilon^2 \approx 2 \times 10^{-16} \Omega^2$ and $\bar{\sigma}_\epsilon^2 \approx 4 \times 10^{-16} \Omega^2$ for the cases $n = 4096$ and $n = 8192$ is invisible to the learning algorithm. Trajectory differences within a regime arise from independent channel realizations rather than from encryption. We also compare to the CSI-based pre-equalization considered in Section III-B: each device estimates $h_{i,k}$ from a pilot (with the standard deviation of the channel-estimation error $\sigma_{\text{pilot}} = 0.1$) and pre-equalizes by ZF or MMSE. The residual estimation error is multiplied by q at decryption, dominating the decoded gradient by a term of order $q/\Lambda \approx 2^{70}$. Fig. 3 confirms this conclusion: the training loss saturates near 10^{14} from the first iteration regardless of estimator or channel variance. Our proposed protocol avoids this by never estimating the channel.

Additional experiments, including a larger-scale evaluation on CIFAR-10 with a LeNet-style CNN ($d = 61,241$), are reported in [16] and confirm that the encryption and communication overhead of the proposed protocol remains independent of the model dimension.

TABLE III: Test accuracy on MNIST 0-vs-1 across the three HE parameter setups and the non-encrypted baseline.

Setup	n	$\log_2 q$	$\sigma = 1$	$\sigma = 10$
A (HE)	8192	218	98.39%	93.52%
B (HE)	4096	109	98.30%	94.56%
no-enc	—	—	97.78%	94.33%

VII. CONCLUSION

We showed that multi-key HE and over-the-air aggregation can coexist: the additive structure of xMK-CKKS matches the wireless superposition, and the encryption noise vanishes after decoding. Among N users, each user's device incorporates the known channel mean $\{\mu_i\}_{1 \leq i \leq N}$ into its encoding, and

the gradient estimate is formed directly from the decrypted aggregate. The decoded HE noise introduces a noise floor that is negligible for all practical purposes. The numerical results on MNIST confirm this, and the ZO structure keeps the encrypted payload at a single scalar per device, so the encryption overhead does not grow with the model dimension d .

Several directions remain open: reducing the key retransmission overhead by exploiting slow channel variation, extending the protocol to approximate block fading with controlled error, and combining HE with differential privacy where the channel noise serves as a privacy mechanism on top of the encryption layer.

APPENDIX

The xMK-CKKS scheme is considered one of the most prominent and suitable HE schemes for FL scenarios, as it provides multi-key security, supports floating-point computations, and relies on the RLWE lattice problem, for which no practical attacks are currently known under recommended parameters. The xMK-CKKS scheme has been proposed in [8] and operates over the ring $R_q = \mathbb{Z}_q[X]/(X^n + 1)$, where n denotes the polynomial ring degree (RLWE dimension) and q represents the ciphertext coefficient modulus.

The encoded message uses a scaling factor Λ to balance numerical precision and noise growth. The different cryptographic components and parameters of the xMK-CKKS scheme can be found in [8] and are summarized in the following steps for completeness.

- **Secret-key generation.** Each device d_i independently samples its secret key $s_i \in \{-1, 1\}^n$ according to Bernoulli(1/2).
- **Partial public-key generation.** Each device computes a partial public key $b_i = -s_i a + e_i \mod q$, where a is a common public polynomial and $e_i \sim \mathcal{N}(0, \sigma_e^2)^{\otimes n}$, with $\sigma_e = 3.2$.
- **Aggregated public-key construction.** The partial public keys are summed to form the aggregated public key $\tilde{b} = \sum_{i=1}^N b_i$, which gives

$$\tilde{b} + a \sum_{i=1}^N s_i = \tilde{b} + aS = \sum_{i=1}^N e_i := E_{agg}, \quad (21)$$

where $S := \sum_{i=1}^N s_i$.

- **Encryption.** Each device encrypts its plaintext m_i using the aggregated public key $ct_i = (c_{0,i}, c_{1,i})$, where $c_{0,i} = v_i \tilde{b} + m_i + e_i^{(0)}$, and $c_{1,i} = v_i a + e_i^{(1)} \mod q$, where v_i is a fresh random polynomial used to randomize the encryption process.
- **Homomorphic aggregation.** The server aggregates the ciphertexts $C_{\text{sum}} = \sum_{i=1}^N ct_i = (C_{\text{sum},0}, C_{\text{sum},1})$.
- **Partial decryption.** Each device computes a decryption share $D_i = s_i C_{\text{sum},1} + e_i^* \mod q$, where e_i^* is the effective decryption-noise term associated with device d_i , modeled as a discrete Gaussian random variable with variance determined by the underlying encryption errors.

- **Collaborative reconstruction.** The server combines the aggregated ciphertext and all decryption shares $C_{\text{sum},0} + \sum_{i=1}^N D_i \mod q$. After expansion, the large key-dependent terms involving a cancel algebraically, leaving $\sum_{i=1}^N m_i + \text{bounded noise}$. Therefore, $C_{\text{sum},0} + \sum_{i=1}^N D_i \approx \sum_{i=1}^N m_i$.

A detailed correctness analysis of the collaborative decryption procedure in xMK-CKKS is provided in the full version of the paper [16, Appendix A].

ACKNOWLEDGMENT

This work was supported by PHC C  dre 2025 (project code 52201TK) and by Universit   Saint-Joseph de Beyrouth (USJ).

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Ag  ria y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. AISTATS*, ser. Proceedings of Machine Learning Research, vol. 54, 2017, pp. 1273–1282.
- [2] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. NeurIPS*, 2019.
- [3] M. M. Amiri and D. G  nd  z, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, 2020.
- [4] M. Assaad, Z. Nehme, and M. Debbah, "Communication-efficient zero-order and first-order federated learning methods over wireless networks," *arXiv:2508.08013*, 2025.
- [5] E. Mhanna and M. Assaad, "Countering the communication bottleneck in federated learning: A highly efficient zero-order optimization technique," *J. Mach. Learn. Res.*, vol. 25, no. 418, pp. 1–53, 2024.
- [6] Y. Wang, B. Zhang, J. Zhang, and C. Li, "Efficient privacy-preserving federated learning via homomorphic encryption-enabled over-the-air computation," *IEEE Trans. Mobile Comput.*, vol. 24, no. 12, pp. 13 743–13 759, 2025.
- [7] X. Xie, J. Hong, C. Hua, and Y. Xu, "Communication efficient ciphertext-field aggregation in wireless networks via over-the-air computation," *IEEE Trans. Inf. Forensics Security*, vol. 20, pp. 31–45, 2025.
- [8] J. Ma, S.-A. Naas, S. Sigg, and X. Lyu, "Privacy-preserving federated learning based on multi-key homomorphic encryption," *Int. J. Intell. Syst.*, vol. 37, no. 9, pp. 5880–5901, 2022.
- [9] Microsoft Research, "Microsoft seal, release 4.1," [Online]. Available: <https://github.com/microsoft/SEAL>, 2023.
- [10] Y. Chen, C. Han, Z. Yu, and G. Wang, "Channel measurement, characterization, and modeling for terahertz indoor communications above 200 ghz," *IEEE Trans. Wireless Commun.*, vol. 23, no. 6, pp. 6518–6532, 2024.
- [11] J. M. Jornet, V. Petrov, H. Wang, Z. Popovic, D. Shakya, J. V. Siles, and T. S. Rappaport, "The evolution of applications, hardware design, and channel modeling for terahertz (THz) band communications and sensing: Ready for 6g?" *Proc. IEEE*, vol. 113, no. 9, pp. 920–951, 2025.
- [12] S. Priebe and T. K  rner, "Stochastic modeling of THz indoor radio channels," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4445–4455, 2013.
- [13] Y. Chen, Y. Li, C. Han, Z. Yu, and G. Wang, "Channel measurement and ray-tracing-statistical hybrid modeling for low-terahertz indoor communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 12, pp. 8163–8176, 2021.
- [14] J. C. Duchi, M. I. Jordan, M. J. Wainwright, and A. Wibisono, "Optimal rates for zero-order convex optimization: The power of two function evaluations," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2788–2806, 2015.
- [15] A. Agarwal, O. Dekel, and L. Xiao, "Optimal algorithms for online convex optimization with multi-point bandit feedback," in *Proc. Conf. Learning Theory (COLT)*, 2010, pp. 28–40.
- [16] A. Ayli, K. Harris, J. Fahs, and M. Assaad, "Privacy-enhanced zero-order federated learning via xMK-CKKS over wireless channels," 2026. [Online]. Available: <https://arxiv.org/abs/2605.30123>
- [17] M. Albrecht *et al.*, "Homomorphic encryption standard," in *Protecting Privacy through Homomorphic Encryption*. Springer, 2021, pp. 31–62.