

NetHE-FL: Network-Adaptive CKKS-Protected Federated Learning for Healthcare IoT Systems

Sarra Erregui^{1,*}, Otmane El Mouaatamid², El mehdi Loualid¹

¹National School of Applied Sciences, University Chouaib Doukkali, El Jadida, Morocco

²ICES Lab, ENSIAS, Mohammed V University, Rabat, Morocco

*Correspondence: erregui.sarra@ucd.ac.ma

Abstract—Federated learning (FL) allows hospitals to train shared diagnostic models without exchanging patient records, yet the transmitted model updates can still leak sensitive information, motivating the use of homomorphic encryption such as the CKKS scheme. CKKS protects real-valued updates at the cost of substantial ciphertext expansion (a packed CKKS ciphertext is about 327 KB, roughly an order of magnitude larger than the equivalent plaintext payload), which is prohibitive over the heterogeneous, bandwidth-constrained wireless links typical of healthcare Internet-of-Things (IoT) deployments. We argue that the dominant bottleneck of encrypted healthcare FL is not cryptographic computation alone, but the mismatch between static encrypted aggregation and dynamic wireless network conditions. We propose *NetHE-FL*, a network-adaptive CKKS-protected FL framework that executes encryption at the hospital edge gateway and jointly adapts aligned block sparsification, CKKS slot packing, transmission frequency, and client scheduling to both wireless link state and local update importance. We derive analytical communication, delay, and energy models for plaintext FL, full CKKS-FL, and *NetHE-FL*, and evaluate the framework on the UCI Heart Disease dataset under non-IID hospital partitions against four baselines. *NetHE-FL* reduces encrypted communication cost by about 67% and per-hospital transmission delay by about 33% relative to packed CKKS-FL, while keeping classification accuracy statistically indistinguishable from plaintext FL (mean difference under half a percentage point, within run-to-run variance), under an honest-but-curious aggregation server. A model-dimension analysis shows the communication saving growing to about 74% as model size approaches realistic scales. We further provide a deployment feasibility map for CKKS-protected FL across BLE, Zigbee, Wi-Fi, 4G, and 5G healthcare IoT networks.

Index Terms—Federated learning, homomorphic encryption, CKKS, healthcare IoT, communication efficiency, wireless networks, privacy-preserving machine learning.

I. INTRODUCTION

Healthcare Internet-of-Things (IoT) systems—wearable cardiac monitors, bedside telemetry, and clinic-grade sensors connected through hospital edge gateways—generate large volumes of sensitive physiological data. Training accurate diagnostic models on this data is clinically valuable, but centralizing patient records across institutions is restricted by privacy regulation and institutional policy. Federated learning (FL) addresses this by keeping data local: each hospital trains on its own records and shares only model updates, which a server aggregates into a global model [1], [2].

However, sharing model updates is not equivalent to preserving privacy. Gradient- and weight-level updates have been

shown to leak training samples through reconstruction attacks [3], so the updates themselves require protection. Homomorphic encryption (HE) enables the server to aggregate encrypted updates without observing their contents. The CKKS scheme [4] is particularly suited to FL because it operates natively on vectors of approximate real numbers, matching the representation of model parameters, and supports slot packing for amortized throughput. CKKS has accordingly been adopted in encrypted FL aggregation pipelines [5], [6].

The practical obstacle is communication. CKKS ciphertexts are substantially larger than the plaintext vectors they encode, and full encrypted FL transmits one or more ciphertexts per client per round. This overhead is tolerable in data-center cross-silo settings, but healthcare IoT links are heterogeneous and constrained: a rural clinic may connect over BLE or Zigbee at hundreds of kilobits per second while a tertiary hospital uses Wi-Fi or 5G, so transmitting full CKKS updates every round over the slower links inflates latency and energy to the point where encrypted FL becomes infeasible. Existing communication-efficient FL [7], [8] and wireless-FL work [9], [10] largely treat the plaintext case and do not jointly reason about ciphertext expansion and link heterogeneity.

This exposes a gap: *the dominant bottleneck of encrypted healthcare FL is not cryptographic computation alone, but the mismatch between static encrypted aggregation and dynamic wireless conditions*. A framework is needed that adapts encrypted-update transmission to both update importance and link capacity, without materially degrading accuracy or privacy. We therefore ask: *how can CKKS-protected federated learning be made communication-feasible over heterogeneous healthcare IoT networks without significantly degrading learning accuracy or privacy?*

We answer this with *NetHE-FL*. Our contributions are:

- **Architecture.** A network-adaptive CKKS-protected FL architecture that places encryption at the hospital edge gateway and augments the FL loop with an update importance estimator and a wireless network profiler (Section IV).
- **Algorithm.** An adaptive encrypted-transmission policy that jointly selects an active block budget, transmission frequency, and client schedule from link state and update importance, using a server-coordinated aligned sparsity mask so that sparse CKKS ciphertexts remain homomorphically additive (Section V).

- **Modeling.** Analytical communication, delay, and energy models for plaintext FL, full CKKS-FL, and NetHE-FL (Section VI).
- **Evaluation.** An evaluation on the UCI Heart Disease dataset under non-IID hospital partitions against four baselines, measuring accuracy, communication cost, delay, and energy (Sections VII–VIII).
- **Deployment insight.** A feasibility map indicating when CKKS-protected FL is viable across BLE, Zigbee, Wi-Fi, 4G, and 5G healthcare IoT networks.

II. RELATED WORK

Federated learning for healthcare. FL was introduced as a communication-efficient way to train shared models from decentralized data [1], and surveys frame its challenges, methods, and open problems [11], [12]. It has been proposed for cross-institutional medical modeling without data sharing [2], [13] and for resource-constrained IoT deployments [14], and heterogeneity-robust optimizers such as FedProx [15] and adaptive edge schemes [16] improve convergence under non-IID data. All of these, however, exchange model updates in the clear.

Privacy of model updates. Updates are not privacy-neutral: reconstruction attacks recover training samples from shared gradients [3], [17], and this leakage extends to honest-but-curious servers in collaborative learning [18]. Secure aggregation masks updates so the server learns only their sum [19], but relies on multi-party masking with dropout-recovery overhead and does not provide a reusable encrypted representation of real-valued parameters.

Homomorphic encryption for FL. HE lets the server aggregate without decrypting. Additively homomorphic schemes were applied to privacy-preserving deep learning [20]. The CKKS scheme operates on approximate real vectors with slot packing [4], making it a natural fit for model parameters, and practical libraries such as TenSEAL [6] expose it to ML pipelines. BatchCrypt reduces HE overhead in cross-silo FL via quantized batch encoding [5], and packed multi-key HE [21] and multi-key FL schemes [22] distribute trust across parties; these target data-center links rather than constrained wireless networks.

Communication-efficient FL. Reducing uplink cost is a long-standing FL goal [23]. Gradient sparsification transmits only the largest update coordinates [7], [8], and non-IID-robust compression has been studied for FL [10]. FL over wireless channels has been analyzed with respect to fading and scheduling [9]. These lines reduce communication but operate on plaintext updates and do not consider CKKS ciphertext expansion.

Positioning. Table I summarizes the gap. In contrast to communication-efficient FL, which compresses *plaintext* updates and ignores ciphertext expansion, and to HE-based FL, which protects updates but treats the network as static, NetHE-FL adapts *encrypted*-update transmission jointly to link state and update importance while preserving homomorphic addability. Among the representative prior work we surveyed,

we did not find a framework that jointly provides CKKS-protected aggregation, a healthcare IoT focus, wireless-link awareness, and adaptive scheduling. NetHE-FL targets exactly this intersection.

III. SYSTEM AND THREAT MODEL

A. System Entities

We consider a cross-silo healthcare federation with the following entities.

- **IoT sensors.** Wearable and bedside devices that collect physiological signals. Sensors only acquire and forward data to their local gateway; they perform no cryptographic operations.
- **Hospital edge gateways.** Each of the N participating hospitals operates an edge gateway (or local healthcare server) that holds the institution’s local dataset, performs local model training, and executes CKKS encoding and encryption. Placing cryptography at the gateway—rather than on resource-limited sensors—is a deliberate design choice justified in Section IV.
- **Cloud aggregator.** An untrusted server that aggregates encrypted updates homomorphically and broadcasts the updated global model.
- **Authorized key authority.** A trusted medical authority that holds the CKKS secret key and decrypts only the aggregated result. The aggregator never holds the secret key.
- **Wireless links.** Hospitals connect to the aggregator over heterogeneous wireless technologies (BLE, Zigbee, Wi-Fi, 4G, 5G), each with distinct data rate, latency, and transmission power.

B. Threat Model

We assume an *honest-but-curious* cloud aggregator: it follows the protocol correctly but may attempt to infer private information from any data it observes. Because all updates are CKKS-encrypted under a public key whose secret counterpart is held only by the authorized authority, the aggregator observes ciphertexts and their homomorphic sum but not plaintext updates. We assume secure, authenticated channels between entities, so the adversary’s view is limited to ciphertexts and protocol metadata (such as which hospitals participate in a round).

C. Privacy Assumptions and Scope

NetHE-FL provides *confidentiality* of model updates against the honest-but-curious aggregator under the CKKS encryption setting. We explicitly do *not* claim protection against: a malicious aggregator that deviates from the protocol; model-poisoning or integrity attacks (CKKS provides confidentiality, not integrity); or collusion involving the key authority. The adaptive scheduler reveals participation metadata (which hospitals transmit in a given round), which we treat as a known leakage channel rather than a protected quantity. A single-key deployment also concentrates trust in the key authority; threshold or multi-key CKKS would distribute this

TABLE I
COMPARISON OF REPRESENTATIVE RELATED WORK. (Y = YES, N = NO, P = PARTIAL.)

Work	FL	HE	CKKS	Healthcare IoT	Wireless-aware	Adaptive comm.	Main limitation
McMahan <i>et al.</i> [1]	Y	N	N	N	N	P	No protection of model updates
Bonawitz <i>et al.</i> [19]	Y	N	N	N	N	N	Masking overhead; no reusable HE
Zhang <i>et al.</i> [5]	Y	Y	N	N	N	N	Data-center focus, not wireless
Rieke <i>et al.</i> [2]	Y	N	N	Y	N	N	Vision/survey; no encryption
Lin <i>et al.</i> [8]	Y	N	N	N	N	P	Plaintext sparsification only
Amiri & Gündüz [9]	Y	N	N	N	Y	P	No encryption of updates
NetHE-FL (ours)	Y	Y	Y	Y	Y	Y	Honest-but-curious, single-key; confidentiality only

trust and is left as future work. In practice the key authority is a governance role—for example a consortium oversight body or a hardware-security-module- or TEE-backed service—rather than a network participant, and it decrypts *only* the per-round aggregate; its compromise would therefore expose aggregated updates but never an individual hospital’s update or raw records. Threshold or multi-key CKKS removes this single point of trust by splitting the secret key so that no single party can decrypt, at additional key-management and communication cost; we regard this as the primary hardening path for deployment.

Scope. NetHE-FL is a confidentiality mechanism for encrypted aggregation, not a complete secure-FL framework, and we make no claim of clinical deployment or end-to-end system security. Because the per-round decrypted aggregate is revealed by design, a curious key authority observes the sequence of aggregated updates across rounds; bounding this aggregate-level leakage (e.g. via differential privacy on the aggregate) is complementary and orthogonal to the transmission confidentiality NetHE-FL provides.

IV. PROPOSED NETHE-FL ARCHITECTURE

NetHE-FL organizes encrypted federated training into seven layers, shown in Fig. 1.

- 1) **Healthcare IoT data.** Sensors stream physiological data to the hospital edge gateway; raw data never leaves the institution.
- 2) **Local learning.** The gateway trains a local model, producing update Δw_i^t .
- 3) **Update importance estimator.** The gateway estimates its update’s importance (Section V) to decide how much to transmit.
- 4) **Wireless network profiler.** It classifies the current link (rate, latency, power) as poor, medium, or good.
- 5) **Adaptive CKKS encoding decision.** From importance and link state, it selects an active block budget and a transmit/skip decision.
- 6) **Aligned sparsification, packing, encryption.** Selected coordinates are packed into CKKS slots and encrypted; a server-coordinated aligned mask preserves homomorphic additivity across hospitals (Section V).
- 7) **Homomorphic aggregation and authorized decryption.** The untrusted aggregator computes a weighted

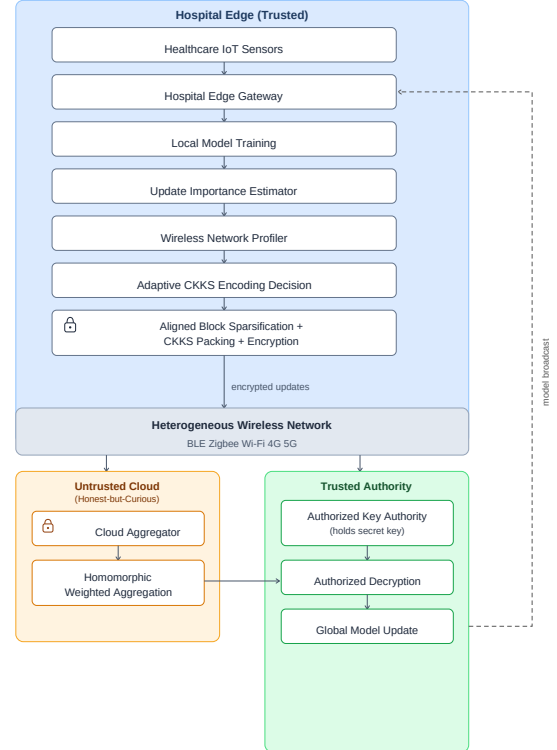


Fig. 1. Architecture of NetHE-FL (seven layers). Data acquisition and local training stay inside the trusted hospital zone, where the edge gateway performs CKKS encryption; the untrusted cloud aggregator sees *only* ciphertexts and their homomorphic sum, and a separate trusted key authority holds the secret key and decrypts *only* the aggregated update before it is broadcast back to all hospitals.

homomorphic sum; the key authority decrypts only the aggregate, which updates the broadcast global model.

We place CKKS at the edge gateway rather than on sensors because CKKS key generation, encoding, and encryption are computationally and memory-intensive relative to the capabilities of typical healthcare IoT sensors. The gateway is the first point with sufficient resources and a natural trust boundary for the institution’s data.

V. NETWORK-ADAPTIVE CKKS-FL ALGORITHM

A. Adaptive Decision Rules

Block partition. The model vector of dimension d is partitioned into B public, fixed, contiguous blocks. This partition is data-independent and identical for all hospitals, so it leaks no private information.

Update importance. Hospital i measures the importance of its round- t update as the ℓ_2 norm of the local update, $I_i^t = \|\Delta w_i^t\|_2$, min-max normalized across participating hospitals.

Wireless state classification. The profiler maps the measured link rate R_i^t into one of {poor, medium, good} using configurable thresholds.

Link-adaptive block budget and skipping. The link state sets the number of active blocks a hospital transmits: poor links send a single block ($\approx 10\%$ of the model), medium links send a mid-range subset, and good links send most or all blocks. A poor link combined with low normalized importance ($I_i^t < \tau_{\text{low}}$) triggers a *skip* for that round.

Network-adaptive aligned block sparsification. Unlike unaligned per-hospital top- k selection, which produces different coordinate sets across hospitals and prevents direct ciphertext addition, NetHE-FL uses a *public structured block mask*. The server selects the active block set M^t from public information only—the round index and the *previous* round's decrypted aggregate-block magnitudes (already revealed by design)—so the mask never depends on any single hospital's private update. Because all transmitting hospitals pack the same blocks into the same CKKS slots, the aggregator can sum ciphertexts directly, and only the small block-index set must be broadcast.

Error feedback. To prevent the systematic loss that fixed sparsification would otherwise cause, each hospital maintains a residual: the coordinates it did not transmit in a round are accumulated locally and added to its next update. Combined with round-robin block coverage, this guarantees every coordinate's information eventually reaches the aggregator, so block sparsification defers rather than discards information. This is the mechanism behind the negligible accuracy gap reported in Section VIII.

B. Algorithm

Algorithm 1 summarizes one communication round.

VI. COMMUNICATION, DELAY, AND ENERGY MODEL

We model the per-round uplink cost of transmitting model updates from N hospitals to the aggregator. Let d be the model dimension, b the bytes per plaintext parameter (e.g. $b = 4$ for 32-bit floats), s the CKKS slot capacity, and S_{ct} the serialized CKKS ciphertext size in bytes (measured from the encryption library, or analytically estimated in fallback mode).

Plaintext FL. Each hospital sends its full update in the clear:

$$C_{\text{plain}}^t = \sum_{i=1}^N d \cdot b. \quad (1)$$

Algorithm 1 NetHE-FL: Network-Adaptive Aligned Block Sparsification (round t)

Require: global model w^t , public key pk , B public blocks, prior aggregate $\bar{g}^{t-1}$

- 1: Server selects active block set M^t from public signals (round index, $\|\bar{g}^{t-1}\|$ per block)
- 2: Server broadcasts w^t and M^t to hospitals
- 3: **for** each hospital i in parallel **do**
- 4: train locally to obtain update Δw_i^t ; importance $I_i^t = \|\Delta w_i^t\|_2$
- 5: measure link state R_i^t ; set block budget b_i^t from R_i^t
- 6: **if** R_i^t poor **and** $I_i^t < \tau_{\text{low}}$ **then**
- 7: **continue** $\triangleright$ skip this round
- 8: **end if**
- 9: restrict Δw_i^t to the shared active blocks M^t (budget b_i^t)
- 10: pack into CKKS slots; encrypt under pk to obtain c_i^t ; transmit
- 11: **end for**
- 12: aggregator computes weighted homomorphic sum $\bar{c}^t = \sum_{i \in \mathcal{S}_t} \alpha_i c_i^t$
- 13: trusted authority decrypts $\bar{c}^t$ to obtain aggregated update $\bar{g}^t$
- 14: server updates w^{t+1} and broadcasts

Full CKKS-FL. Each hospital encrypts and sends its full update; a d -dimensional vector requires $m_i = \lceil d/s \rceil$ ciphertexts:

$$C_{\text{CKKS}}^t = \sum_{i=1}^N m_i \cdot S_{ct}, \quad m_i = \left\lceil \frac{d}{s} \right\rceil. \quad (2)$$

NetHE-FL. Only the selected hospitals $\mathcal{S}_t$ transmit, each sending a fraction k_i^t of coordinates packed into CKKS slots, plus a small per-round mask overhead O_{idx}^t :

$$C_{\text{NetHE}}^t = \sum_{i \in \mathcal{S}_t} \left\lceil \frac{k_i^t d}{s} \right\rceil S_{ct} + O_{\text{idx}}^t. \quad (3)$$

Transmission delay. For transmitted size C_i^t over a link of rate R_i^t with one-way latency ℓ_i :

$$D_i^t = \frac{C_i^t}{R_i^t} + \ell_i. \quad (4)$$

The delay model therefore combines the payload transmission time with the link's one-way latency ℓ_i , so technologies with similar rates but different latencies are distinguished.

Energy cost. With transmission power P_i :

$$E_i^t = P_i \cdot D_i^t. \quad (5)$$

Communication saving. Relative to full CKKS-FL:

$$\text{Saving} = 1 - \frac{C_{\text{NetHE}}}{C_{\text{CKKS}}}. \quad (6)$$

The accuracy/communication trade-off is then characterized by pairing the saving above with the resulting change in model accuracy, reported empirically in Section VIII. Because a small clinical model cannot, by itself, exercise the

communication regime where these differences matter, we additionally evaluate the cost models over a model-dimension sweep $d \in \{10^2, 10^3, 10^4, 10^5\}$; those rows are analytical and labeled as such.

VII. EVALUATION SETUP

Dataset. We use the UCI Heart Disease (Cleveland) dataset [24]: 303 patient records with 13 clinical features and a binarized target (presence vs. absence of disease). Features are standardized. The dataset validates classification accuracy; it is intentionally small and is not used to argue communication scaling on its own.

Synthetic scalability dataset. To stress-test scalability beyond the 13-feature clinical dataset, we additionally generate a synthetic healthcare-IoT binary-classification dataset with 10,000 samples and $d \in \{100, 1000\}$ features. Labels are produced from a latent linear signal; per-hospital class imbalance and feature-distribution shifts are introduced to emulate heterogeneous clinical populations across institutions, and clients are partitioned non-IID by Dirichlet label skew. This dataset is used only to evaluate learning stability and communication scalability; we make no claim of clinical validity from it.

Model. The primary model is logistic regression; a small single-hidden-layer MLP is used as secondary validation. Both are trained with FedAvg.

Federated setup. We simulate $N \in \{5, 10\}$ hospitals with *real* training, and report $N \in \{20, 50\}$ using the analytical cost model only, because the dataset is too small to partition meaningfully across 50 institutions. Data is partitioned non-IID via Dirichlet label-skew. Training runs for 40 communication rounds, averaged over five random seeds.

CKKS configuration. We use CKKS with `poly_modulus_degree` = 8192, `coeff_mod_bit_sizes` = [60, 40, 40, 60], and `global_scale` = 2^{40} , yielding 4096 slots at approximately 128-bit security. Using TenSEAL, the measured serialized ciphertext size is approximately 327 KB, i.e. about $20\times$ expansion relative to a 32-bit plaintext parameter per slot. The measured size depends on the serialized coefficient-modulus configuration; under the same TenSEAL setup the ciphertext-only serialized vector grows to ≈ 642 KB at `poly_modulus_degree` = 16384 (8192 slots, ≈ 128 -bit)—about $2\times$ the 8192-degree size—so larger degrees give more slots but proportionally larger ciphertexts, making wireless transmission harder and motivating the adaptive design. Encryption, decryption, and homomorphic-addition times are also measured; ciphertext size is the quantity that drives the communication model.

Baselines. Plain FL, full CKKS-FL, sparse FL (no CKKS), CKKS + packing, and the proposed NetHE-FL.

Metrics. Accuracy, F1-score, communication cost (bytes), transmission delay, energy, communication saving ratio, encryption time, and ciphertext size. *Communication saving* denotes the reduction in the encrypted *payload volume* (bytes transmitted per round) relative to a stated baseline—not the number of messages or rounds exchanged.

TABLE II
SIMULATION AND WIRELESS-PROFILE PARAMETERS. WIRELESS RATES AND POWERS ARE REPRESENTATIVE MODELING VALUES FROM STANDARDS AND SURVEYS [25]–[28], NOT DEVICE-CERTIFIED MEASUREMENTS.

Dataset	UCI Heart Disease (Cleveland), 303/13		
Model	Logistic regression (+ small MLP)		
Hospitals N	5, 10 (real); 20, 50 (analytical)		
Partition	Non-IID, Dirichlet ($\alpha = 0.5$)		
FL rounds / seeds	40 / 5 (base 42, 42–46)		
Optimizer / LR / batch	Mini-batch SGD / 0.05 / 16		
CKKS	deg. 8192, [60, 40, 40, 60], scale 2^{40} (4096 slots)		
Measured ciphertext	≈ 327 KB		

Technology	Data rate	Latency	TX power
BLE	0.27 Mbps	50 ms	0.03 W
Zigbee	0.25 Mbps	50 ms	0.05 W
Wi-Fi	50 Mbps	10 ms	0.70 W
4G	20 Mbps	30 ms	1.50 W
5G	100 Mbps	10 ms	2.50 W
Rural	1 Mbps	100 ms	1.00 W

A. Sensitivity to Link Quality

To avoid dependence on a single assumed rate per technology, we sweep low/medium/high rate variants ($0.5\times$, $1\times$, $2\times$) of each profile. NetHE-FL delay scales as expected with link rate, and the constrained classes (BLE, Zigbee, rural) benefit from the adaptive block budget across the whole range, confirming the conclusions do not hinge on one rate assumption.

VIII. RESULTS AND DISCUSSION

All reported results are obtained from the implemented simulation pipeline. Large- N and high-dimensional communication results are analytical projections based on measured CKKS ciphertext sizes and are explicitly distinguished from the real FL training results.

A. Accuracy

Fig. 2 reports accuracy versus communication rounds (mean over five seeds, $\pm$ std bands) for Plain FL, full CKKS-FL, sparse FL, and NetHE-FL. CKKS-FL tracks Plain FL because the CKKS approximation error at scale 2^{40} is negligible. Because the block mask drops no information permanently—each hospital carries the un-transmitted residual forward through error feedback—NetHE-FL converges to the same operating point as Plain FL: 0.823 ± 0.041 accuracy and 0.797 ± 0.057 F1 at $N = 10$, identical within seed variance. Block sparsification warms up slightly slower but reaches the same plateau, so NetHE-FL preserves accuracy while transmitting far less per round.

To confirm this is not an artifact of the small clinical model, we repeat the comparison using *actual FL training* (not the analytical model) on the synthetic healthcare-IoT dataset (Section VII) at higher dimensions (Table III). The Plain-vs-NetHE-FL gap stays within noise— -0.03 pp at $d = 100$ and $+0.04$ pp at $d = 1000$, with the sign reversing—so error

TABLE III

SYNTHETIC HEALTHCARE-IOT ACCURACY ($N=10$, FIVE SEEDS). THE PLAIN-VS-NETHE-FL GAP STAYS WITHIN NOISE AS MODEL DIMENSION GROWS, CONFIRMING THE LOSSLESS RESULT IS NOT AN ARTIFACT OF THE SMALL CLINICAL MODEL.

Dim.	Plain FL	NetHE-FL	Gap
$d=100$	$.877 \pm .005$	$.877 \pm .005$	-0.03 pp
$d=1000$	$.916 \pm .004$	$.916 \pm .004$	$+0.04$ pp

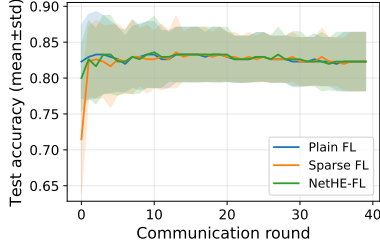


Fig. 2. Accuracy versus communication rounds (measured, five seeds).

feedback keeps the method effectively lossless well beyond the 13-feature clinical case.

B. Communication Cost

Fig. 3 shows communication cost as the number of hospitals grows, comparing Plain FL, full CKKS-FL, CKKS + packing, and NetHE-FL. At $N = 10$ hospitals and model dimension $d = 10^4$, packed CKKS-FL transmits about 10 MB per round, whereas NetHE-FL transmits about 3.3 MB, a saving of approximately 67% relative to packed CKKS-FL (about 60% at $N = 5$). Unpacked CKKS-FL is three to four orders of magnitude larger and is shown only to motivate packing. Realistic cross-silo consortia comprise about 5–20 hospitals; we include $N = 20, 50$ only to show the asymptotic scaling trend and report them as analytical projections (per-hospital cost is near-constant in N) rather than measured deployments.

A model-dimension analysis clarifies the saving’s origin. For $d \leq 4096$ (one CKKS slot vector), sparsification removes no ciphertext, so the 50% saving at small d comes entirely from adaptive scheduling. As d exceeds slot capacity, sparsification removes whole ciphertexts and the saving rises to about 67% at $d = 10^4$ and 74% at $d = 10^5$ —so NetHE-FL’s advantage grows with model size, precisely where encrypted communication is most burdensome.

C. Delay and Energy Across Wireless Profiles

Fig. 4 reports transmission delay under the six wireless profiles. Because NetHE-FL adapts its block budget to link state, the reduction concentrates where it matters: on constrained links it sends fewer blocks, cutting delay by about 67% (BLE 29.8 $\rightarrow$ 9.9 s, Zigbee 32.1 $\rightarrow$ 10.7 s, rural 8.1 $\rightarrow$ 2.8 s), while on high-rate links (Wi-Fi, 4G, 5G) it sends nearly all blocks and delay is essentially unchanged and already sub-second. Energy follows the same pattern under the $E = P \cdot D$ model (e.g. Zigbee 1.61 $\rightarrow$ 0.54 J); NetHE-FL spends bandwidth when it is cheap and conserves it when it is scarce.

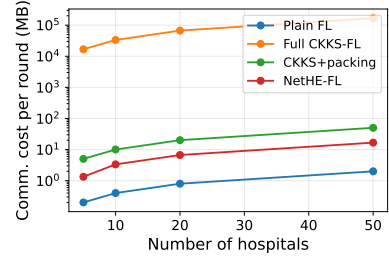
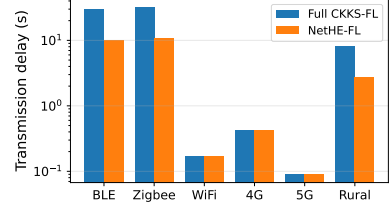
Fig. 3. Communication cost versus number of hospitals (analytical, from measured CKKS sizes); realistic consortia are $N \approx 5\text{--}20$, with $N = 20, 50$ shown as scaling projections.

Fig. 4. Transmission delay under wireless profiles. NetHE-FL reduces delay on constrained links (BLE, Zigbee, rural) while leaving high-rate links unchanged.

D. Ablation and Accuracy–Communication Trade-off

An ablation isolates the contribution of each mechanism relative to the packed CKKS-FL reference: packing only (0% saving), block sparsification only (33%), scheduling only (50%), link-aware adaptation only (33%), and full NetHE-FL (67%)—so scheduling and link-aware budgeting are complementary and stack to the full saving.

Measuring accuracy loss (mean $\pm$ std over seeds) against communication saving, we find that with error feedback the mean accuracy difference from Plain FL stays below the per-seed standard deviation at every operating point: no statistically significant penalty is observed across the 33% to 67% saving range (cf. Table IV). The savings stack as expected—scheduling and link-aware block budgeting are complementary—while none introduces a measurable accuracy penalty, because dropped blocks are deferred rather than discarded.

E. Sensitivity to Non-IID Severity

Across Dirichlet skew $\alpha \in \{0.1, 0.5, 1.0\}$ (smaller α = stronger heterogeneity), NetHE-FL tracks Plain FL at every level: the mean accuracy gap is 0.33 pp at the most skewed setting ($\alpha = 0.1$) and 0.0 pp otherwise, with overlapping error bars. The block-masking and scheduling mechanisms thus remain stable under strong non-IID conditions rather than amplifying client drift.

F. Limitations

We state limitations plainly. (i) Clinical accuracy uses a small dataset; higher-dimensional accuracy is validated with actual FL training on a synthetic dataset (Table III), while large- N communication figures are analytical projections. (ii) The guarantee is confidentiality against an honest-but-curious

TABLE IV

BASELINE COMPARISON AT $N = 10$, $d = 10^4$, UNDER THE CONSTRAINED ZIGBEE PROFILE. ACCURACY AND F1 ARE MEASURED OVER FIVE SEEDS; COMMUNICATION, DELAY, AND ENERGY ARE ANALYTICAL. PLAIN AND SPARSE FL ARE UNENCRYPTED, WHILE CKKS+PACKING AND NETHE-FL PROVIDE HE-BASED UPDATE CONFIDENTIALITY.

Method	Acc.	F1	Comm.	Delay	Energy	Priv.
Plain FL	.823±.041	.797±.057	0.04 MB	1.4 s	0.07 J	None
Sparse FL	.823±.041	.797±.057	0.03 MB	0.9 s	0.05 J	None
CKKS + packing	.823±.041	.797±.057	10.0 MB	32.1 s	1.61 J	HE
NetHE-FL	.823±.041	.797±.057	3.3 MB	10.7 s	0.54 J	HE

aggregator under single-key CKKS; integrity, poisoning resistance, malicious servers, and key-authority compromise are out of scope. (iii) The wireless model captures rate and latency but not packet loss or contention. (iv) Update skipping leaks participation metadata and may bias aggregation under strong non-IID skew. (v) Error feedback defers information, so heavily constrained links converge over more rounds, which our fixed round budget does not separately account for.

IX. CONCLUSION

We presented NetHE-FL, a network-adaptive CKKS-protected federated learning framework for healthcare IoT systems. From the observation that the dominant obstacle to encrypted healthcare FL is the mismatch between static encrypted aggregation and dynamic wireless conditions, NetHE-FL places CKKS at the hospital edge gateway and adapts aligned block sparsification, slot packing, transmission frequency, and client scheduling to link state and update importance, using a server-coordinated aligned mask so that sparse ciphertexts remain homomorphically additive. Evaluated on the UCI Heart Disease dataset under non-IID partitions against four baselines, NetHE-FL reduced encrypted communication by about 67% (up to 74% at larger model dimensions) and per-hospital delay by about 33% relative to packed CKKS-FL, while keeping accuracy statistically indistinguishable from plaintext FL. Future work includes threshold or multi-key CKKS to distribute trust, integrity and poisoning defenses, a packet-loss-aware channel model, and validation on larger clinical models and datasets.

REFERENCES

- [1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.
- [2] N. Rieke, J. Hancox, W. Li, F. Milletari, H. R. Roth, S. Albarqouni, S. Bakas, M. N. Galtier, B. A. Landman, K. Maier-Hein *et al.*, "The future of digital health with federated learning," *npj Digital Medicine*, vol. 3, no. 1, p. 119, 2020.
- [3] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.
- [4] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Advances in Cryptology – ASIACRYPT*, 2017, pp. 409–437.
- [5] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "BatchCrypt: Efficient homomorphic encryption for cross-silo federated learning," in *Proc. USENIX Annual Technical Conf. (ATC)*, 2020, pp. 493–506.
- [6] A. Benaissa, B. Retiat, B. Cebere, and A. E. Belfedhal, "TenSEAL: A library for encrypted tensor operations using homomorphic encryption," in *ICLR Workshop on Distributed and Private Machine Learning*, 2021.
- [7] A. F. Aji and K. Heafield, "Sparse communication for distributed gradient descent," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2017, pp. 440–445.
- [8] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2018.
- [9] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Communications*, vol. 19, no. 5, pp. 3546–3557, 2020.
- [10] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Trans. Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3400–3413, 2020.
- [11] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.
- [12] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji *et al.*, "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.
- [13] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," *Journal of Healthcare Informatics Research*, vol. 5, no. 1, pp. 1–19, 2021.
- [14] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for internet of things: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021.
- [15] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Machine Learning and Systems (MLSys)*, vol. 2, 2020, pp. 429–450.
- [16] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.
- [17] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients – how easy is it to break privacy in federated learning?" in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 16937–16947.
- [18] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS)*, 2015, pp. 1310–1321.
- [19] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS)*, 2017, pp. 1175–1191.
- [20] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Trans. Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, 2018.
- [21] H. Chen, W. Dai, M. Kim, and Y. Song, "Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference," in *Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS)*, 2019, pp. 395–412.
- [22] Y. Cai, W. Ding, Y. Xiao, Z. Yan, X. Liu, and Z. Wan, "SecFed: A secure and efficient federated learning based on multi-key homomorphic encryption," *IEEE Trans. Dependable and Secure Computing*, 2023.
- [23] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.
- [24] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J.-J. Schmid, S. Sandhu, K. H. Guppy, S. Lee, and V. Froelicher, "International application of a new probability algorithm for the diagnosis of coronary artery disease," *The American Journal of Cardiology*, vol. 64, no. 5, pp. 304–310, 1989.
- [25] Bluetooth SIG, "Bluetooth core specification v5.2," Bluetooth Special Interest Group, 2019, IE PHY data rates.
- [26] IEEE, *IEEE Standard for Low-Rate Wireless Networks (802.15.4)*, IEEE Std., 2020.
- [27] —, *IEEE Standard for Information Technology—Wireless LAN MAC and PHY Specifications (802.11)*, IEEE Std., 2021.
- [28] E. Dahlman, S. Parkvall, and J. Sköld, "5G NR: The next generation wireless access technology," *Academic Press*, 2020.