Energy-Efficient Online Federated Learning in Wireless Networks

Jaemin Kim, Junsuk Oh, Wonjong Noh, and Sungrae Cho

***Abstract*—In this work, we study energy-efficient online federated learning. First, we analyzed the convergence performance of device selection in dynamic and non-stationary conditions, deriving the upper bound of the convergence rate. Building on this, we formulated an optimization problem that minimizes energy consumption while ensuring convergence and meeting latency constraints. Second, we developed an online allocation and scheduling-based iterative strategy (OASIS). Here, we designed a combinatorial upper-confidence-bound-based device scheduling algorithm with a newly designed reward function. We also derived a Lambert function-based power allocation to the scheduled devices in closed form. We performed a dynamic regret analysis, which reveals that the proposed algorithm effectively adapts to dynamic environments and maintains near-optimal decisions over time. It also demonstrates that the proposed algorithm achieves sublinear regret in a slowly changing dynamic environment and optimal regret in a static environment. Experimental results show that the proposed OASIS achieves faster convergence and provides significantly lower energy consumption than existing conventional baseline strategies. We also confirmed that the performance gain increases as the target accuracy level becomes higher. These results validate the energy efficiency and robustness of the proposed approach in realistic and time-varying federated learning environments.**

***Index Terms*—Data Distribution, Device Scheduling, Energy Efficiency, Multi-Armed Bandit, Online Federated Learning, Regret Analysis.**

I. Introduction

FEDERATED learning (FL) has emerged as a promising paradigm for decentralized machine learning, enabling edge devices to collaboratively train a global model while preserving local data privacy [1]. By performing local updates on devices and only transmitting model parameters, FL has gained significant attention as a method for training a global model across distributed devices without sharing raw data.

However, most traditional FL frameworks are implemented in environments where each device's dataset remains fixed throughout the process and has a static data distribution. Such environments cannot capture the dynamic data distribution of real-world data over time due to device behavior, streaming input, or environmental changes.

On the other hand, online learning (OL) is a framework in which learners make decisions sequentially over time, receive feedback after each decision, and adjust their strategies accordingly. A key performance metric in OL is regret, which measures how much worse the learner performs compared to an ideal benchmark, often the best fixed decision in hindsight. As real-world conditions shift unpredictably, OL approaches have proven successful in maintaining low regret in non-stationary environments [2]. By minimizing regret, OL algorithms ensure that cumulative performance remains competitive even in non-stationary or adversarial environments, making this paradigm well-suited for scenarios in which data and system conditions change over time.

Such an OL paradigm naturally fits FL in dynamic data distribution, as decisions (e.g., which devices to participate, how much power to allocate) must be updated sequentially at each communication round, without full knowledge of future data arrivals or distribution changes. Accordingly, the concept of online FL (OFL) has recently gained attention [3]–[5]. OFL integrates the principles of OL into FL, enabling models to be updated in real time as data is generated on each device.

Although OFL has attracted considerable attention recently, several new issues arise when transitioning from traditional FL to OFL, such as determining which device to involve as participants when the data distribution changes or balancing FL convergence rate and network efficiency when specific devices have large datasets but poor channel conditions. Even if selecting devices with more data and poor channel conditions or limited transmit power improves FL convergence, it may reduce energy efficiency. Conversely, selecting devices with less data and good channel conditions may boost network efficiency, but could hinder learning performance.

Since OFL faces these problems, we propose a novel OFL framework in this work that jointly optimizes device scheduling and power allocation under a dynamic data distribution environment. We observe that scheduling which devices participate in each round, given that their data sizes change over time, naturally forms an online decision scenario. The server must explore updated information while exploiting known high-quality participants (e.g., devices with larger datasets). The main novelty of this paper is the integration of device scheduling and power allocation principles into a single framework, enabling a sublinear-regret solution for FL under dynamic data conditions.

A. Related Work

1) Resource-Constrained FL: First, early studies primarily focused on selecting a subset of devices to accommodate limited communication and computation budgets. Wang *et al.* [6] proposed an intelligent device sampling framework that jointly accounts for heterogeneous resources and overlapping data. However, they assume fixed local datasets and solve

Jaemin Kim, Junsuk Oh, and Sungrae Cho are with the School of Computer Science and Engineering, Chung-Ang University, Seoul 06974, Republic of Korea (e-mail: jmkim@uclab.re.kr; jsoh@uclab.re.kr; srcho@cau.ac.kr) Wonjong Noh is with the School of Software, Hallym University, Chuncheon 24252, Republic of Korea. (email: wonjong.noh@hallym.ac.kr)

the scheduling problem using an offline-trained model. Chen *et al.* [7] developed a predictive online control scheme for hierarchical FL, using mixed-integer programming with a deep reinforcement learning policy to minimize long-term cost. This approach is adaptive, but it relies on accurate predictions and does not explicitly model the changing data distributions. Su *et al.* [8] proposed an online device scheduling algorithm under budget constraints using bandit theory. Although regret-minimizing, it abstracts communication costs and does not consider physical-layer dynamics such as power allocation. Moreover, Deressa and Hasan [9] introduced a multi-armed bandit (MAB)-based device scheduling algorithm that is robust against poisoning attacks, without requiring the modeling of communication constraints or power control. This study confirms the importance of device selection but leaves open the question of how to respond to both channel variation and changing data simultaneously.

Second, several works have begun to consider device scheduling and power allocation jointly. Xu *et al.* [10] proposed a Lyapunov-based algorithm for online joint scheduling and power control, considering non-independent and identically distributed (i.i.d.) and time-varying data. This method handled dynamic environments but focused on long-term time-average performance and did not provide regret bounds. Perazzone *et al.* [11] derived convergence bounds for FL under stochastic scheduling and proposed a scheduling-power allocation policy to minimize model error and delay. Their approach improved communication efficiency but lacked explicit online regret minimization.

Third, a separate line of work focused on dynamic data distributions. Jin *et al.* [12] modeled stochastic data arrivals at the edge device, so each device's local dataset grew over time. They developed a budget-aware controller that dynamically triggered a training round only when growth exceeded a specified threshold, demonstrating faster convergence than fixed-period schemes under an energy cap. While many works focused on growing data, Liu *et al.* [13] addressed the opposite form of dynamism: data deletion after training. Their framework efficiently removed a device's influence by replaying cached updates instead of costly full retraining, thus keeping the global model consistent in a shrinking dataset. Furthermore, Babendererde *et al.* [14] studied time-varying data distributions. They simulated gradual changes in input data at the device level (i.e., drift), followed by a sudden change in the relationship between inputs and labels across all devices (i.e., a concept shift). Their results showed that when past data is only kept for a short time, models suffer more from catastrophic forgetting, highlighting the need for learning methods that can adapt to short-lived and changing data patterns. In addition, Hu *et al.* [15] proposed scheduling based on data importance under dynamic streaming scenarios, using Lyapunov optimization. However, their model assumed predefined data variation metrics and did not include power allocation.

2) Online Federated Learning: Recently, some works have focused on OFL. Chen *et al.* [16] proposed an asynchronous OFL framework designed for edge devices with streaming non-i.i.d. data. Unlike traditional FL methods that assume each device's dataset is fixed, their system allowed devices to continuously receive new samples and asynchronously upload local models to the server after training. This setting reflects realistic conditions where data accumulates over time (e.g., sensor logs) and must accommodate partial or delayed uploads while still achieving effective convergence in the server. Thus, they demonstrated that leveraging OL principles and asynchronous communication protocols can significantly improve FL performance in truly dynamic data environments. Damaskinos *et al.* [5] introduced an OFL framework that addresses issues of stale and fresh updates from devices in a dynamic data setting. Since different devices may gather new data at varying rates and upload it on different schedules, some model contributions become stale due to varying transmission times. They employed a staleness-awareness mechanism and performance prediction to weigh device updates more effectively, ensuring that recent and relevant data receive greater influence in the aggregated model. In experiments simulating streaming asynchronous FL deployments, this approach showed improved adaptation and convergence. Ganguly *et al.* [17] investigated a non-stationary data scenario from an OFL perspective, emphasizing communication efficiency. They noted that devices may continue to gather new data, causing local distributions to shift across rounds. To address this, the proposed method adjusted the selection of participating devices and the frequency of local updates, thereby minimizing unnecessary transmissions while maintaining model accuracy. Their experiments demonstrated that carefully managing device participation and update intervals mitigates the adverse effects of data drift and limited bandwidth, underscoring the necessity for an OFL approach in practical and dynamic environments.

B. Motivation, Contribution, and Organization

There is active research on OFL. However, research on energy-efficient OFL under unpredictable data variations is still in its infancy. The main contributions of this work can be summarized as follows.

- First, we analyzed the relationship between the convergence rate and device scheduling in dynamic and non-stationary conditions, and derived the upper bound of the convergence rate. Based on this analysis, we designed a new objective function that promotes energy efficiency while guaranteeing convergence. Then we formulated an OFL optimization problem that jointly considers both device scheduling and power allocation simultaneously.
- Second, as a solution, we developed an online allocation and scheduling-based iterative strategy (OASIS). Here, we propose a combinatorial upper-confidence-bound-based device scheduling approach that leverages a newly designed reward function, which considers transmission energy, local data size, and the average number of selections. Then, we also derived a Lambert function-based power allocation for the scheduled devices in closed form.
- Lastly, we analyzed the dynamic regret and confirmed that the proposed algorithm provides a sublinear regret under slowly changing optimal paths. Through extensive

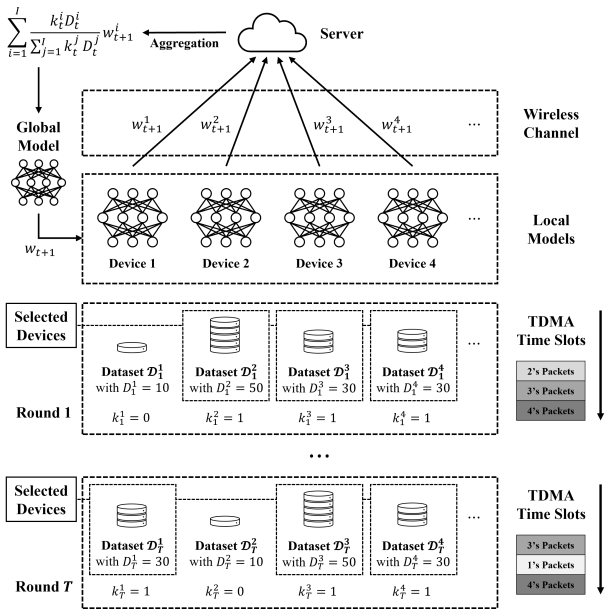


Fig. 1. System model.

simulations, we confirmed that the proposed control significantly outperforms the conventional baseline approach [18] in terms of energy efficiency. The performance gain of the proposed control becomes more pronounced as the target learning level increases.

The remainder of the work is organized as follows. Section II describes the system model, and Section III-A derives theoretical convergence bounds under dynamic data distribution and partially participating devices, illustrating how the online device scheduling and power allocation scheme influences both the rate and stability of model convergence. Next, Section IV details the proposed scheme, and Section V presents the theoretical analysis of dynamic regret. Finally, the performance evaluation is provided in Section VI, followed by a conclusion in Section VII.

II. System Model

In this section, we describe the system model, which is illustrated in Fig. 1. For the wireless FL, we consider a central server and a set of devices denoted as $\mathcal{I} = \{1, 2, \ldots, I\}$. Both the server and each device i are equipped with a single antenna.

A. Data Model

In each round $t \in [T]$, each device $i \in \mathcal{I}$ possesses a local dataset $\mathcal{D}_t^i$, which may change over time due to the online nature of data generation and deletion. The overall local dataset $\mathcal{D}_t$ is defined as

$$\mathcal{D}_t = \bigcup_{i=1}^{I} \mathcal{D}_t^i. \tag{1}$$

The size of device i's local dataset is denoted as $D_t^i = |\mathcal{D}_t^i|$ and $D_t^i \geq 0$. By considering data generation and deletion, which is the dynamic nature of online environments, the change in local dataset size is modeled as

$$D_t^i = D_{t-1}^i + \rho_t^i, \tag{2}$$

where ρ_t^i represents the net change in the number of samples for device i, which is defined as

$$\rho_t^i = \rho Z_t^i, \tag{3}$$

where $\rho \in \mathbb{Z}^+$ represents changeable size, and Z_t^i is

$$Z_t^i \in \{-1, 0, 1\}, \quad \mathbb{P}(Z_t^i = z) = (p_+^i, p_0^i, p_-^i), \tag{4}$$

where p_+^i, p_0^i, and p_-^i are the probabilities of dataset growth, retention, and reduction, respectively. These probabilities satisfy the normalization condition, which is defined as

$$p_+^i + p_0^i + p_-^i = 1. \tag{5}$$

Remark 1. In realistic online FL settings, only three fundamental types of dataset evolution can occur on a device between two consecutive communication rounds: *net growth*, *stasis*, or *net reduction*. The model in Eqs. (4)–(5) provides the simplest abstraction of these possibilities, while the parameter $\rho_t^i = \rho Z_t^i$ enables changes at the granularity of a mini-batch of newly arriving or expiring samples.

Our convergence bound (Eq. (24)) and dynamic regret bound (Eq. (46)) rely on Assumption 4, which requires the expected relative dataset variation to remain bounded. The proposed three-valued model naturally satisfies this condition and supports a clean theoretical analysis without losing the essential characteristics of online data evolution.

Although we adopt a three-valued random variable for clarity and analytical tractability, the proposed framework is *not restricted* to this model. More complex data dynamics can be incorporated as long as Assumption 4 is satisfied. The theoretical developments—including the gradient estimation error bound, convergence analysis, and dynamic regret—remain valid under these generalizations, because they depend only on the aggregate variation condition in Assumption 4 rather than on the specific form of Z_t^i.

B. Learning Model

In round t, each device i aims to minimize its local loss function $F^i(\mathbf{w}_t)$ using its local dataset $\mathcal{D}_t^i$ defined as

$$F^i(\mathbf{w}_t) = \frac{1}{D_t^i} \sum_{(\mathbf{x},y)\in\mathcal{D}_t^i} f(\mathbf{w}_t; \mathbf{x}, y), \tag{6}$$

where $\mathbf{w}_t \in \mathbb{R}^M$ is the parameter vector of the global model, and $f(\mathbf{w}_t; \mathbf{x}, y)$ is the sample-wise loss function with the feature vector $\mathbf{x}$ and the corresponding label y. The global loss function $F(\mathbf{w}_t)$ is defined by aggregating the local losses of all devices as

$$F(\mathbf{w}_t) = \frac{1}{D_t} \sum_{i=1}^{I} D_t^i F^i(\mathbf{w}_t), \tag{7}$$

where $D_t = \sum_{i=1}^{I} D_t^i$, and D_t^i/D_t denotes the device-wise aggregation weight. This weight ensures that each device's contribution to the global model is proportional to the size

of its local dataset, promoting fairness and efficiency in the learning process.

The objective of the OFL system is to collaboratively minimize (7), leveraging the local computational resources of devices while preserving data privacy. To achieve this, each device i performs online gradient descent (OGD) to compute the local gradient $\mathbf{g}_t^i$ as

$$\mathbf{g}_t^i = \frac{1}{D_t^i} \sum_{(\mathbf{x},y)\in\mathcal{D}_t^i} \nabla f(\mathbf{w}_t; \mathbf{x}, y), \tag{8}$$

where $\nabla f(\mathbf{w}_t; \mathbf{x}, y)$ denotes the gradient of $f(\mathbf{w}_t; \mathbf{x}, y)$. Each device then updates its local model as

$$\mathbf{w}_{t+1}^i = \mathbf{w}_t - \eta\, \mathbf{g}_t^i, \tag{9}$$

where η is the learning rate. This process enables each device to refine the global model using its local data.

In this work, the server performs device scheduling for each round. Here, we assume that the server has full information on the local dataset size and CSI of all devices. Subsequently, the server aggregates the updated local models from the scheduled devices to update the global model. This aggregation is performed using a weighted averaging approach similar to the Federated Averaging algorithm [1] as

$$\mathbf{w}_{t+1} = \sum_{i=1}^{I} \frac{k_t^i D_t^i}{\sum_{j=1}^{I} k_t^j D_t^j} \mathbf{w}_{t+1}^i, \tag{10}$$

where k_t^i indicates whether device i is scheduled as a participant in round t. Specifically, $k_t^i = 1$ if device i is selected and in round t, $k_t^i = 0$, otherwise. We assume that $\sum_{i=1}^{I} k_t^i \geq 1$ to ensure model convergence. Finally, the server broadcasts this global model to all devices.

C. Communication and Computation Model

We employ time division multiple access (TDMA), which divides the available communication time into orthogonal time slots, for uplink transmission in this federated learning, offering collision-free communication, latency predictability, and energy efficiency [19], [20]. It is particularly suitable for scenarios involving IoT devices, energy-constrained systems, or latency-sensitive applications such as industrial IoT or autonomous vehicles [21].

The wireless channel between device i and the server experiences Rayleigh fading, characterized by a channel coefficient h_t^i. We assume that the channel state information (CSI) is perfectly estimated and known to both device i and the server. This CSI remains constant during each round t but may vary between rounds.

In each round t, each selected device i transmits its updated model $\mathbf{w}_{t+1}^i$ to the server after local training. The required number of bits for transmission, B, is given by

$$B = n \times M, \tag{11}$$

where n is the number of bits used for floating-point precision (e.g., $n = 32$ for single-precision representation) and M is the number of model parameters. This represents the total size of the model update that the device needs to send back to the server. The latency $\tau_{\mathrm{Tx,t}}^i$, defined as the required time for transmission, is

$$\tau_{\mathrm{Tx,t}}^i = \frac{B}{R_t^i}, \tag{12}$$

where R_t^i is the transmission rate, defined by the Shannon capacity formula as

$$R_t^i = W \log_2 \left(1 + \frac{P_t^i |h_t^i|^2}{W N_0}\right), \tag{13}$$

where W is the bandwidth, P_t^i is the transmit power, $|h_t^i|^2$ denotes the channel gain, and N_0 represents the noise power spectral density. Equation (12) and (13) collectively show that the transmit power P_t^i directly or indirectly affects the latency and transmission rate. This highlights the necessity of determining optimal P_t^i to ensure energy-efficient OFL in a latency-constrained environment.

D. Energy Consumption Model

Based on the communication latency in (12), the energy consumption $E_{tx,t}^i$ for transmission is defined as

$$E_{\mathrm{Tx},t}^i(P_t^i) = \frac{P_t^i B}{R_t^i} = P_t^i \tau_t^i. \tag{14}$$

In addition to wireless transmission, local training itself introduces energy consumption. To model these costs, we define four device-specific parameters: ω^i denotes the number of CPU cycles required to process one data sample, χ^i represents the CPU frequency of device i in hertz, σ^i is an energy-per-cycle coefficient capturing the device's hardware efficiency, and e^i is the number of local epochs executed by device i in round t. Following a dynamic-voltage-and-frequency-scaling (DVFS) framework in [22], the energy consumption $E_{\mathrm{comp},t}^i$ for computation is defined as

$$E_{\mathrm{comp},t}^i = \sigma^i\, \omega^i\, e^i\, D_t^i \left(\chi^i\right)^2. \tag{15}$$

Then, the total energy consumption by device i in round t is given by

$$E_t^i(P_t^i) = E_{\mathrm{comp},t}^i + E_{\mathrm{Tx},t}^i(P_t^i). \tag{16}$$

III. Convergence Analysis and Problem Formulation

In this section, we analyze the convergence behavior of the OFL system with device scheduling and formulate our problem.

A. Convergence Analysis with Device Scheduling

To facilitate the convergence analysis, we make the following assumptions regarding the loss function, gradient, and data variation.

Assumption 1. *(L-smoothness):* The global loss function $F(\mathbf{w})$ is L-smooth, ensuring that its gradient does not change abruptly. Specifically, for all $\mathbf{w}, \mathbf{w}' \in \mathbb{R}^M$,

$$\|\nabla F(\mathbf{w}) - \nabla F(\mathbf{w}')\|_2 \leq L \|\mathbf{w} - \mathbf{w}'\|_2, \tag{17}$$

which is essential for controlling the step sizes during the optimization process. This assumption plays a critical role in deriving upper bounds for the convergence rate.

Assumption 2. *(Sample-wise Gradient Bounded):* For any sample, the sample-wise gradient norm, which is the norm of the stochastic gradient, is upper bounded by a function of the ideal global gradient norm. That is, for all samples $(\mathbf{x}, y)$ and for all $\mathbf{w} \in \mathbb{R}^M$,

$$\|\nabla f(\mathbf{w}; \mathbf{x}, y)\|_2^2 \leq \phi + \psi \|\nabla F(\mathbf{w})\|_2^2, \tag{18}$$

where $\phi \geq 0$ represents the intrinsic variance of the device-wise gradients, and $\psi \geq 1$ quantifies the extent to which the global gradient norm influences the stochastic gradient [23], [24].

Assumption 3. *(Unbiased Gradient Estimator):* The device scheduling policy ensures that the global gradient $\mathbf{g}_t$ aggregated through weighted averaging is an unbiased estimator of the ideal global gradient $\nabla F(\mathbf{w}_t)$ [25]. Formally,

$$\mathbb{E}[\mathbf{g}_t | \mathbf{w}_t] = \nabla F(\mathbf{w}_t). \tag{19}$$

This assumption guarantees that the expected value of $\mathbf{g}_t$ aligns with $\nabla F(\mathbf{w}_t)$. That is, on average, the optimization trajectory follows the direction of $\nabla F(\mathbf{w}_t)$ and results in stable and reliable convergence.

Assumption 4. *(Bounded Relative Data Variation):* The expected relative variation in local dataset size is upper bounded by a constant $\delta \in (0, 1)$. Formally,

$$\mathbb{E}\left[\sum_{i=1}^{I} \left| \frac{\rho_t^i}{D_t^i} \right| \right] \leq \delta, \quad \forall i, t. \tag{20}$$

This assumption ensures that the change ρ_t^i in local dataset size during round t remains small relative to the dataset size D_t^i. In addition, this prevents large fluctuations in the local dataset and thereby contributes to stable convergence analysis.

Given the above assumptions, we derive an explicit expression of the expected convergence bound of the OFL system. This explicitly incorporates the device scheduling variable k_t^i and local dataset size D_t^i.

Lemma 1. An upper bound on the norm of the global gradient estimation error $\mathbf{e}_t$ is given by

$$\mathbb{E}[\|\mathbf{e}_t\|^2] \leq I \left(\frac{\sum_{j=1}^{I} D_t^j}{\sum_{j=1}^{I} k_t^j D_t^j} - 1 \right) \times \left(\phi + \psi \mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2 \right] \right). \tag{21}$$

Proof: See Appendix **A**. ∎

Lemma 1 characterizes how the device scheduling in each round t affects the deviation of the estimated global gradient from the ideal global gradient. If all devices participate, $\mathbb{E}[\|\mathbf{e}_t\|_2^2] = 0$, meaning there is no estimation error. In contrast, if only a subset of devices participates, $\sum_{j=1}^{I} k_t^j D_t^j$ decreases while $\sum_{j=1}^{I} D_t^j$ remains unchanged, causing the estimation error to increase.

Theorem 1. Under **Assumptions 1-4** and **Lemma 1**, the convergence rate with device scheduling is given by

$$\frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] \leq \frac{2\left(\mathbb{E}[F(\mathbf{w}_1)] - F^*\right)}{\eta T \left(2 - L\eta\left(1 - I\psi D_t\right)\right)} + \sum_{t=1}^{T} \frac{I\phi\eta \sum_{j=1}^{I}(1 - k_t^j) D_t^j}{T\left(1 - I\psi D_t\right)}. \tag{22}$$

Proof: See Appendix **B**. ∎

From **Theorem 1**, we observe the following. First, the upper bound in (22) monotonically decreases as the maximum round T increases, indicating that all terms can converge to zero when T is sufficiently large. Second, as stated in **Lemma 1**, the second term of the upper bound in (22) decreases in proportion to the cumulative data size of the scheduled devices. This implies that utilizing a larger amount of local training data can accelerate the convergence rate, although it may also lead to higher energy consumption.

B. Problem Formulation

In this work, motivated by the convergence bound (22) in **Theorem 1**, we construct an objective as a per-round penalty function that captures both convergence and energy consumption:

$$u_t(\mathbf{k}_t, \mathbf{P}_t) := \alpha \sum_{i=1}^{I} (1 - k_t^i) D_t^i + (1 - \alpha) \sum_{i=1}^{I} k_t^i E_t^i(P_t^i), \tag{23}$$

which is controlled by device scheduling $\mathbf{k}_t := [k_t^1, \ldots, k_t^I]$ and power allocation $\mathbf{P}_t := [P_t^1, \ldots, P_t^I]$. That is, we aim to solve the joint device scheduling and power allocation problem in OFL, targeting both fast convergence and energy efficiency. The first term $\sum_{i=1}^{I}(1 - k_t^i) D_t^i$ controls the convergence speed of the proposed algorithm, which belongs to the second term in the upper bound (22): reducing this quantity in each round tightens the theoretical bound on the average gradient norm. The second term $\sum_{i=1}^{I} k_t^i E_t^i(P_t^i)$ represents the total energy consumption of the scheduled devices. The weight factor $\alpha \in [0, 1]$ balances the impact of convergence and energy consumption: a larger α places more emphasis on faster convergence, whereas a smaller α prioritizes saving energy.

Based on this objective, we define the joint minimization problem as:

$$\textbf{P1:} \quad \min_{\mathbf{k}_t, \mathbf{P}_t} \quad u_t(\mathbf{k}_t, \mathbf{P}_t) \tag{24a}$$

$$\text{s.t.} \quad 0 \leq P_t^i \leq P_{\max}^i, \quad \forall i, \tag{24b}$$

$$\sum_{i=1}^{I} k_t^i \tau_t^i \leq \tau. \tag{24c}$$

The constraint (24b) ensures that the transmit power of each device i does not exceed its maximum transmit power $P_{\max}^i$, and constraint (24c) ensures that the total uploading latency by the selected devices remains below a threshold $\tau > 0$. In the subsequent section, we find the set of participating devices and the transmit power that minimizes the objective function.

IV. Proposed Solution

Since the discrete variables $\mathbf{k}_t$ and continuous variables $\mathbf{P}_t$ are coupled, problem **P1** is a mixed-integer programming problem. Solving this joint optimization problem is non-convex and intractable. Therefore, we find its solution using alternate optimization that iterates the process of determining the participating device set $\mathbf{k}_t$ and then allocating the transmit power $\mathbf{P}_t$. In the following, ℓ denotes an iteration index for the alternative optimization framework. For each round t, the proposed algorithm takes ℓ_{max} number of inner loops to find the solution.

A. Device Scheduling

Under fixed $\mathbf{P}_t$, minimizing $u_t(\mathbf{k}_t, \mathbf{P}_t)$ in (23) over $\mathbf{k}_t$ is equivalent to maximizing

$$\sum_{i=1}^{I} k_t^i \left(\alpha D_t^i - (1-\alpha) E_t^i(P_t^i)\right). \tag{25}$$

This expression shows that scheduling devices with higher D^i and lower $E_t^i(P_t^i)$. This device scheduling problem can be equivalently formulated as a combinatorial multi-armed bandit (CMAB) problem, where we introduce a device-wise score s_t^i. To solve it, we adopt the combinatorial upper confidence bound (CUCB) algorithm, which naturally balances exploration and exploitation [26], [27].

We first define a scheduled device set $\mathcal{K}_t(\ell) = \{i \in \mathcal{I} \mid k_t^i = 1\}$ satisfying the latency constraint (24c). In each iteration $\ell \in [1, \ell_{\max}]$ at each round t, the problem **P2** is formulated as:

$$\textbf{P2:} \quad \max_{\mathbf{k}_t \in \{0,1\}^I} \quad \sum_{i=1}^{I} k_t^i(\ell) s_t^i(\ell) \tag{26a}$$

$$\text{s.t.} \quad \sum_{i=1}^{I} k_t^i(\ell) \tau_t^i(\ell-1) \le \tau, \tag{26b}$$

where $\tau_t^i(\ell-1)$ are calculated as

$$\tau_t^i(\ell-1) = \frac{B}{W \log_2\left(1 + \frac{P_t^i(\ell-1)|h_t^i|^2}{W N_0}\right)}, \tag{27}$$

and $\tau_t^i(0)$ is initialized using $P_t^i(0) := P_{\max}^i$. In the objective function, $s_t^i(\ell)$ denotes the CUCB score, which is defined by augmenting the empirical means with the exploration bonuses as

$$s_t^i(\ell) = \hat{r}_t^i(\ell) + \kappa \sqrt{\frac{3 \ln t}{2 n_t^i(\ell)}}, \tag{28}$$

where $\kappa > 0$ is a tunable exploration coefficient that controls the trade-off between exploration and exploitation [28]. A higher κ encourages the selection of less frequently chosen devices by increasing the exploration bonus, while a lower κ favors devices with higher empirical mean rewards, thereby promoting exploitation. In addition, $n_t^i(\ell)$, which is the total count that device i has been selected up to round t, is defined as

$$n_t^i(\ell) = 1 + \sum_{j=1}^{t-1} k_j^i + \mathbb{1}_{i \in \mathcal{K}_t(\ell-1)}, \tag{29}$$

Algorithm 1 CUCB-Based Device Scheduling

1: **Input:** τ, κ, ℓ, $\mathcal{I}$, $\mathbf{P}_t(\ell-1)$, $\{\tau_t^i(\ell-1)\}_{i\in\mathcal{I}}$.
2: **Output:** $\mathcal{K}_t(\ell)$.
3: **Initialize** $\mathcal{K}_t(\ell) \leftarrow \{\varnothing\}$, $\beta \leftarrow 1$, $\mathbf{c}(\ell)$.
4: **while** $\beta \le I$ and (26b) is satisfied **do**
5: Append c_β to $\mathcal{K}_t(\ell)$.
6: $\beta = \beta + 1$.
7: **end while**
8: **Return** $\mathcal{K}_t(\ell)$.

where the first term of on the RHS aims to prevent $n_t^i(\ell) = 0$ (i.e., avoiding division by zero in the CUCB score) [26], and $\mathbb{1}_{i \in \mathcal{K}_t(\ell-1)}$ is an indicator function. Therefore, based on (29), the empirical mean reward $\hat{r}_t^i(\ell)$ in (28) is defined as

$$\hat{r}_t^i(\ell) = \frac{\sum_{j=1}^{t-1} r_j^i + \mathbb{1}_{\ell>1} \times r_t^i(\ell)}{n_t^i(\ell)}, \tag{30}$$

where $\mathbb{1}_{\ell>1}$ is the indicator function, and $r_t^i(\ell)$ is the device-wise partial reward defined as

$$r_t^i(\ell) = \begin{cases} -(1-\alpha) E_t^i(P_t^i(\ell-1)), & i \in \mathcal{K}_t(\ell-1), \\ -\frac{\alpha}{D_t^i}, & \text{otherwise.} \end{cases} \tag{31}$$

The reward is designed with a double disincentive: 1) selected devices are penalized in proportion to their energy consumption, and 2) unselected devices are penalized by the inverse of their dataset size. This reward design encourages devices with large local datasets for unselected devices or small consumption energy for selected devices to receive smaller penalties, thereby increasing their chances of being selected in subsequent rounds.

Using (28), the server (i.e., the scheduler) computes $\{s_t^i(\ell)\}_{i\in\mathcal{I}}$, and derived $\mathbf{c}(\ell) = [c_1, \ldots, c_I]$ as the permutation of indices that sorts $\{s_t^i(\ell)\}_{i\in\mathcal{I}}$ in descending order, i.e., $s_t^{c_1}(\ell) \ge s_t^{c_2}(\ell) \ge \ldots \ge s_t^{c_I}(\ell)$. The server sequentially includes the device i with larger $s_t^i(\ell)$ from $\mathbf{c}(\ell)$ into $\mathcal{K}_t(\ell)$ as long as the latency constraint (26b) is satisfied. Consequently, $\mathbf{k}_t(\ell)$ is determined through $\mathcal{K}_t(\ell)$. The proposed solution is summarized in **Algorithm 1**.

B. Transmit Power Allocation

For the scheduled devices, we allocate the transmit power $\mathbf{P}_t(\ell)$. For this, the power allocation problem of **P1** corresponds to the follows:

$$\textbf{P3:} \quad \min_{\mathbf{P}_t(\ell)} \quad \sum_{i=1}^{I} k_t^i E_t^i(P_t^i(\ell)) \tag{32}$$

$$\text{s.t.} \quad \text{(24b), (24c).}$$

In (32), the energy consumption of device i at ℓ-th iteration in round t, $E_t^i(P_t^i(\ell))$, is

$$\begin{aligned} E_t^i(P_t^i(\ell)) &= E_{\text{comp},t}^i + E_{\text{Tx},t}^i(\ell), \\ &= E_{\text{comp},t}^i + P_t^i(\ell)\tau_t^i(\ell). \end{aligned} \tag{33}$$

Here, the computation energy $E_{\text{comp},t}^i$ depends on local-training hyperparameters e^i and hardware parameters ω^i, χ^i,

and σ^i, but it is independent of the transmit power. Hence, for a fixed schedule $\mathbf{k}_t$ and fixed local-training settings, the term $E^i_{\mathrm{comp},t}$ is constant with respect to $\mathbf{P}_t(\ell)$ and can be dropped from (33) without affecting the minimization. Therefore, **P3** can be equivalently reduced to **P3-1**:

$$\textbf{P3-1:} \quad \min_{\mathbf{P}_t(\ell)} \quad \sum_{i=1}^{I} k_t^i P_t^i(\ell)\tau_t^i(\ell) \qquad (34)$$
$$\text{s.t.} \quad (24b), (24c).$$

Here, the objective function is a convex function, and (24b) and (24c) give convex sets so that the problem **P3-1** is a convex problem; this is formally stated in **Property 2** and proved in **Property 3.**

To solve this problem, we introduce dual variables $\lambda_1^i \geq 0$, $\lambda_2^i \geq 0$, and $\lambda_3 \geq 0$, and derive Lagrangian function as follows:

$$\begin{aligned}
&\mathcal{L}(\mathbf{P}_t(\ell), \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \lambda_3) \\
&= \sum_{i=1}^{I} k_t^i(\ell) P_t^i(\ell)\tau_t^i(\ell) + \sum_{i=1}^{I} k_t^i(\ell)\lambda_1^i(P_t^i(\ell) - P_{\max}^i) \\
&\quad - \sum_{i=1}^{I} k_t^i(\ell)\lambda_2^i P_t^i(\ell) + \lambda_3 \left(\sum_{i=1}^{I} k_t^i(\ell)\tau_t^i(\ell) - \tau \right), \qquad (35)
\end{aligned}$$

where $\boldsymbol{\lambda}_1 = [\lambda_1^1, \ldots, \lambda_1^I]$ and $\boldsymbol{\lambda}_2 = [\lambda_2^1, \ldots, \lambda_2^I]$ are associated with power constraint (24b), and λ_3 is associated with latency constraint (24c). Then, by solving this dual problem with KKT conditions [29], we derive **Theorem 2**.

Theorem 2. The device-wise transmit power that minimizes (35) is derived as

$$P_t^i(\ell) = \begin{cases} 0, & \lambda_1^i = 0, \lambda_2^i > 0, \\ \dfrac{1}{\gamma_t^i}\left[\dfrac{\Gamma_t^i(\ell)}{\mathbb{W}(\Gamma_t^i(\ell)/e)} - 1\right], & \lambda_1^i = \lambda_2^i = 0, \\ P_{\max}^i, & \lambda_1^i > 0, \lambda_2^i = 0, \end{cases} \qquad (36)$$

where $\gamma_t^i = |h_t^i|^2/WN_0$, $\Gamma_t^i(\ell) = \gamma_t^i \lambda_3 k_t^i(\ell) - 1$, and $\mathbb{W}(\cdot)$ denotes the Lambert-$\mathbb{W}$ function.

Proof: See Appendix **C**. ∎

Using **Theorem 2**, in each iteration ℓ at each round t, the server computes transmit power $P_t^i(\ell)$ and latency $\tau_t^i(\ell)$ for all devices, which are utilized in the next $(\ell+1)$-th iteration at round t.

C. Alternative Optimization

The proposed alternative optimization strategy, termed the online allocation and scheduling-based iterative strategy (OASIS), runs the above $\mathbf{k}_t$ and $\mathbf{P}_t$ optimizations iteratively until the maximum iteration $\ell_{\max}$ is reached or the convergence threshold ϵ is satisfied. Thus, if the termination condition is met at specific iteration ℓ^*, the proposed OASIS sets $\mathbf{k}_t(\ell^*)$, $\mathbf{P}_t(\ell^*)$, and $\{r_t^i(\ell^*)\}_{i\in\mathcal{I}}$ as $\mathbf{k}_t$, $\mathbf{P}_t$, and $\{r_t^i\}_{i\in\mathcal{I}}$, respectively. The pseudo-code of the proposed OASIS method, which solves **P1**, is summarized in **Algorithm 2**, where $u_t(\mathbf{k}_t(\ell), \mathbf{P}_t(\ell))$ is simply expressed as $u_t(\ell)$. On the other hand, the number of iterations ℓ on line 9 of **Algorithm 2** is as follows:

Algorithm 2 Proposed OASIS Method

1: **Input:** n, M, W, κ, α, τ, ϵ, $\mathcal{I}$, $\{D_t^i\}_{i\in\mathcal{I}}$, $\{\gamma_t^i\}_{i\in\mathcal{I}}$.
2: **Output:** $\mathbf{k}_t$, $\mathbf{P}_t$, $\{r_t^i\}_{i\in\mathcal{I}}$.
3: **Initialize:** $\ell \leftarrow 0$, $u_t(0) \leftarrow +\infty$, $\mathbf{P}_t(0)$, $\{\tau_t^i(0)\}_{i\in\mathcal{I}}$.
4: **Loop:**
5: $\quad \ell \leftarrow \ell + 1$.
6: $\quad$ Derive $\mathbf{k}_t(\ell)$ by **Algorithm 1**.
7: $\quad$ Derive $\mathbf{P}_t(\ell)$ and $\{\tau_t^i(\ell)\}_{i\in\mathcal{I}}$ by **Theorem 2**.
8: $\quad$ Derive $u_t(\ell)$ by $\mathbf{k}_t(\ell)$ and $\mathbf{P}_t(\ell)$.
9: **Until** ℓ, which satisfies $|u_t(\ell) - u_t(\ell-1)| < \epsilon$.
10: $\mathbf{k}_t, \mathbf{P}_t, \{r_t^i\}_{i\in\mathcal{I}} \leftarrow \mathbf{k}_t(\ell), \mathbf{P}_t(\ell), \{r_t^i(\ell)\}_{i\in\mathcal{I}}$.
11: **Return** $\mathbf{k}_t$, $\mathbf{P}_t$, $\{r_t^i\}_{i\in\mathcal{I}}$.

Property 1. *(Inner-iteration convergence)* Let $\ell_t(\epsilon)$ denote the minimal number of iterations required to satisfy the convergence condition, in round t, $|u_t(\ell) - u_t(\ell-1)| < \epsilon$. There exist constants $C_1, C_2 > 0$, such that

$$\ell_t(\epsilon) \leq C_1 + C_2 \log\left(\frac{1}{\epsilon}\right), \qquad (37)$$

where $C_1 = 2 + \frac{\log\left(u_t(0) - u_t^\star\right)}{|\log \Xi|}$ and $C_2 = \frac{1}{|\log \Xi|}$ in arbitrary constant $\Xi \in (0,1)$.

Proof: See Appendix **D**. ∎

Each iteration of the proposed OASIS algorithm consists of two main steps: CUCB-based device scheduling and Lambert-$\mathbb{W}$ function–based power allocation. In the worst case, the server computes the upper-confidence-bound indices for all I devices, which incurs a time complexity of $\mathcal{O}(I \log I)$. The subsequent power allocation step requires solving a closed-form expression for each selected device and therefore adds only an additional $\mathcal{O}(I)$ computational cost. Then, by Property 1, the per-global-round complexity becomes

$$\mathcal{O}\left(\log(1/\epsilon)\left(I \log I + I\right)\right),$$

which simplifies to

$$\mathcal{O}\left(\log(1/\epsilon)\, I \log I\right). \qquad (38)$$

Based on these results, the proposed OASIS algorithm is computationally efficient.

V. Dynamic Regret Analysis

A. Definition of Dynamic Regret

The dynamic regret is defined as the cumulative difference between the cost incurred by the algorithm and the optimal cost at each round:

$$\begin{aligned}
\mathrm{Reg}_T &= \sum_{t=1}^{T} \Big[\sum_{i=1}^{I} \big(u_t(k_t^i, P_t^i) - u_t(k_t^{i,*}, P_t^{i,*}) \big) \Big] \qquad (39) \\
&= \sum_{t=1}^{T} \left[u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_t^*, \mathbf{P}_t^*) \right], \qquad (40)
\end{aligned}$$

where $\mathbf{k}_t^* := [k_t^{1,*}, \ldots, k_t^{I,*}]$ and $\mathbf{P}_t^* := [P_t^{1,*}, \ldots, P_t^{I,*}]$ are the optimal solution of u_t at round t.

Assumption 5. *(Lipschitz-like bounded in u_t):* Each cost function $u_t(\mathbf{k}_t, \mathbf{P}_t)$ is L-Lipschitz-like bounded in every pair of decision variables. That is, there exists $L > 0$ such that for any two feasible decisions $(\mathbf{k}_t, \mathbf{P}_t)$ and $(\mathbf{k}_t^*, \mathbf{P}_t^*)$,

$$|u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_t^*, \mathbf{P}_t^*)| \leq L\|(\mathbf{k}_t, \mathbf{P}_t) - (\mathbf{k}_t^*, \mathbf{P}_t^*)\|_2. \quad (41)$$

This ensures that no single step change in decisions can cause an arbitrarily large change in the cost function.

Property 2. *(Strong Convexity of Energy Function)* For each device i, the energy function E_t^i is μ-strongly convex on the interval $P_t^i \in [P_{\min}^i, P_{\max}^i]$ with $P_{\min}^i \geq 0$. That is,

$$\frac{d^2 E_t^i}{d(P_t^i)^2} \geq \mu > 0, \quad \forall P_t^i. \quad (42)$$

Proof: See Appendix **E**. ∎

Property 3. *(Mixed-integer Convexity)* For each round t, the dynamic regret in (39) can be classified into four cases according to each value of k_t^i and $k_t^{i,*}$ as follows:

	$k_t^{i,*} = 0$	$k_t^{i,*} = 1$
$k_t^i = 0$	0	$\alpha D_t^i - (1-\alpha)E_t^{i,*}$
$k_t^i = 1$	$(1-\alpha)E_t^i - \alpha D_t^i$	$(1-\alpha)(E_t^i - E_t^{i,*})$

For any $\mathbf{k}_t$ and $\mathbf{k}_t^{i,*}$, the dynamic regret in (40) is convex with respect to $\mathbf{P_t}$ by **Property 2**. That is, (40) satisfies mixed-integer convexity [30].

Definition 1. *(Path-Length V_T and Regret)* We define the optimal decision sequence as having bounded cumulative variation. Using an optimal solution $(\mathbf{k}_t^*, \mathbf{P}_t^*)$ for u_t at round t, the path-length V_T over T rounds is defined as

$$V_T = \sum_{t=2}^{T} \left\| \left(\mathbf{k}_t^*, \mathbf{P}_t^*\right) - \left(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*\right) \right\|, \quad (43)$$

which sums the stepwise change in the optimal solution from one round to the next. We assume V_T is finite (and perhaps grows sub-linearly with T in a changing environment). This quantifies how rapidly the optimal decision drifts over time (i.e., smaller V_T means that the environment is more stable).

B. Regret Analysis from Decomposition

The total regret Reg_T is decomposed into two components, tracking term and drift term, by adding and subtracting $u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)$ inside the summation:

$$\mathrm{Reg}_T = \sum_{t=1}^{T} \underbrace{\left[u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)\right]}_{\text{1) Tracking term}} + \sum_{t=1}^{T} \underbrace{\left[u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*) - u_t(\mathbf{k}_t^*, \mathbf{P}_t^*)\right]}_{\text{2) Drift term}}, \quad (44)$$

where we define $(\mathbf{k}_0^*, \mathbf{P}_0^*) := (\mathbf{k}_1^*, \mathbf{P}_1^*)$, ensuring that the drift term is zero at $t = 1$.

1) **Tracking term** represents the regret incurred at round t due to not using the previous round's optimal solution.
2) **Drift term** accounts for the regret incurred because the optimal decision itself has changed from $t-1$ to t.

Lemma 2. Under CUCB, the tracking term in (44) can be bounded as:

$$\sum_{t=1}^{T} \left[u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)\right] \leq \sum_{i=1}^{I} \frac{6\kappa^2}{\Delta_t^i} \ln T + \mathcal{O}(1/\sqrt{T}) + \mathcal{O}(1), \quad (45)$$

where $i^\star = \arg\max_j \mathbb{E}[r_t^j]$ and $\Delta_t^i = \mathbb{E}[r_t^{i^\star}] - \mathbb{E}[r_t^i]$.

Proof: See Appendix **F**. ∎

Theorem 3. Under **Assumptions 5**, **Property 2**, **Lemma 2**, and **Definition 1**, the dynamic regret upon CUCB scheduling is bounded as:

$$\mathrm{Reg}_T \leq \sum_{i=1}^{I} \frac{6\kappa^2}{\Delta_t^i} \ln T + \mathcal{O}(1/\sqrt{T}) + LV_T. \quad (46)$$

Proof: See Appendix **G**. ∎

In **Theorem 3**, the bound consists of three components: The first term $\frac{6\kappa^2}{\Delta_t^i} \ln T$ arises from the CUCB rule, which selects a sub-optimal arm at most $\mathcal{O}(\ln T)$ times in expectation [31], [32]. The second term $\mathcal{O}(1/\sqrt{T})$ stems from approximating the dual multiplier λ_3 using a finite number of gradient or bisection steps. The third drift term LV_T depends on the cumulative change of the per-round optimum.

Remark 2. From Theorem 3, the dynamic regret satisfies $\mathrm{Reg}_T \leq \mathcal{O}(\ln T) + \mathcal{O}(1/\sqrt{T}) + V_T$. If the path-length V_T grows slowly enough, i.e., $V_T = o(T)$, then the average regret vanishes,

$$\mathrm{Reg}_T/T \to 0 \text{ as } T \to \infty. \quad (47)$$

That is, the regret grows sub-linearly in T, e.g., $\mathcal{O}(\ln T)$. In contrast, if $V_T = \Theta(T)$ due to rapidly changing data distributions, the term V_T becomes dominant and the dynamic regret can scale linearly in T, which is consistent with standard impossibility results for dynamic regret in fully adversarial online learning.

Remark 3. When the optimal scheduling and power decisions are time-invariant, i.e., $(\mathbf{k}_t^*, \mathbf{P}_t^*)$ is fixed, the optimum path-length V_T vanishes, yielding $V_T = 0$ [33], and the dynamic regret Reg_T reduces to static regret [34]:

$$\mathrm{Reg}_T \leq \mathcal{O}(\ln T) + \mathcal{O}(1/\sqrt{T}), \quad (48)$$

which matches the optimal static regret in [35].

VI. Experimental Evaluation

A. Experimental Setup

In this work, we used CIFAR-10 [36] for evaluation. Each device i starts with an initial dataset of size D_0^i, where D_0^i is uniformly sampled from the range $[10, 1010]$. Thus, some devices start with a relatively large local set, while others have only a handful of samples. At each communication round t, the local dataset size changes according to ρ_t^i in (2), the net change

in the number of data samples. More specifically, each device draws its net data change $\rho_t^i \in \{-50, 0, +50\}$. The probabilities p_-^i, p_0^i, and p_+^i are sampled once at initialization and satisfy (5), where p_-^i and p_+^i are generated from $\mathcal{U}[0.15, 0.45]$. It reflects device heterogeneity, with different data samples in each round. We run $T = 1000$ communication rounds. For local training, we employed a five-layer convolutional neural network: two convolutional layers (with a kernel size of 5×5 and ReLU activations, followed by 2×2 max-pooling) and three fully connected layers, reducing the dimension to 10 output classes. Each selected device runs $e^i = 5$ epochs per round, and we set the local learning rate η to 0.001 and the floating-point precision n to 32-bit.

At each round t, each device's channel coefficient h_t^i is drawn from i.i.d. $\mathcal{CN}(0, 1)$. Each device i has a maximum transmit power $P_{\max}^i$, which is uniformly chosen from the range $[0.05, 0.2]$. This reflects device heterogeneity, where some devices can afford higher power (potentially reducing uplink time while others are power-limited). The latency constraint τ is set to 8.0, meaning the proposed algorithm guarantees a maximum latency of τ regardless of the number of devices scheduled. The system bandwidth is set to $W = 1$ MHz, and the noise power spectral density N_0 is 3.98×10^{-21} W/Hz, corresponding to the thermal noise floor of -174 dBm/Hz (-114 dBm over 1 MHz), [37], [38]. The key simulation parameters are summarized in Table I.

TABLE I
KEY EXPERIMENTAL PARAMETERS

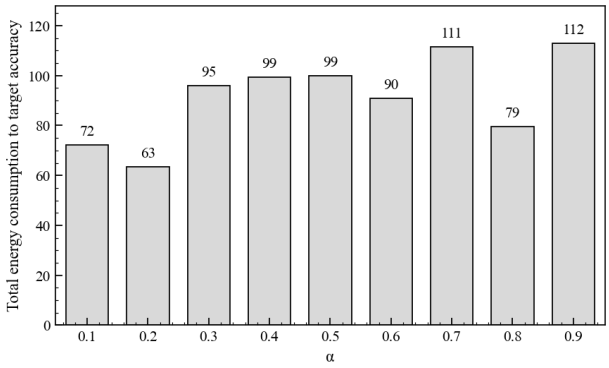
Parameter	Value
Number of Devices I	5, 25, 50, 100
Initial Data Size D_0^i	Uniform in $\{10, \ldots, 1010\}$
Data Variation ρ_t^i	$\rho_t^i \in \{-50, 0, +50\}$; $p_+^i \in [0.15, 0.45]$; $p_-^i \in [0.15, 0.45]$
Max Transmit Power $P_{\max}^i$	Uniform in $[0.05,\ 0.20]$ [W]
Latency Budget τ	8.0 secs
Maximum Round T	1000
Local Learning Model	$e^i = 5$, $\eta = 0.001$
Bandwidth W	10^6 [Hz]
Noise Spectral Density N_0	3.98×10^{-21} [W/Hz] ≈ -114[dBm/Hz]

In this work, we compare the proposed algorithm with the following baseline schemes:

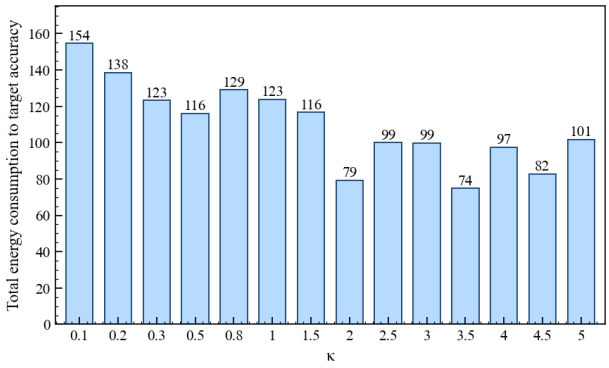
- *FedOGD and Random Selection (FGRS):* This baseline utilizes a federated online gradient descent (FedOGD) algorithm to update local models. Also, it employs random selection (RS) for device scheduling, where a subset of devices is randomly selected. This approach offers minimal overhead, but it does not account for heterogeneous device conditions or data distributions.
- *FedOGD and CS-UCB-Q (FGCUQ):* This baseline applies the FedOGD for local model update. It also employs the UCB policy and virtual queue technique (CS-UCB-Q) [18] based device scheduling mechanism.
- *ELASTIC:* This baseline implements the existing elastic scheme [39], which jointly considers device scheduling and power control under a per-round latency constraint. It represents a state-of-the-art approach that adaptively adjusts the transmit power and the number of participating devices.
- *Only Device Scheduling (OnlyDS):* This baseline uses only the device scheduling (DS) component of the proposed alternating optimization framework. In particular, it adopts the same CUCB-based scheduler as OASIS to select devices but employs a fixed (or heuristic) transmit power control policy without the Lambert-$\mathbb{W}$ based optimization. This isolates the benefit of intelligent device scheduling.
- *Only Power Allocation (OnlyPA):* This baseline uses only the power allocation (PA) component of the proposed framework. It applies the Lambert-$\mathbb{W}$ based power control given a baseline scheduling strategy (e.g., that of Elastic or random selection), but it does not use the CUCB-based scheduler. This isolates the benefit of optimizing the transmit power while keeping the scheduling policy unchanged.

B. Evaluation of the Proposed Algorithm

Fig. 2a and Fig. 2b compare the total cumulative energy consumption required to reach a target test accuracy of 65% for different values of the weight factor α and κ in the objective function. The minimum is observed at $\alpha = 0.2$ and $\kappa = 3.5$; thus, this is adopted as the default hyperparameter in all subsequent experiments.



(a) Factor α

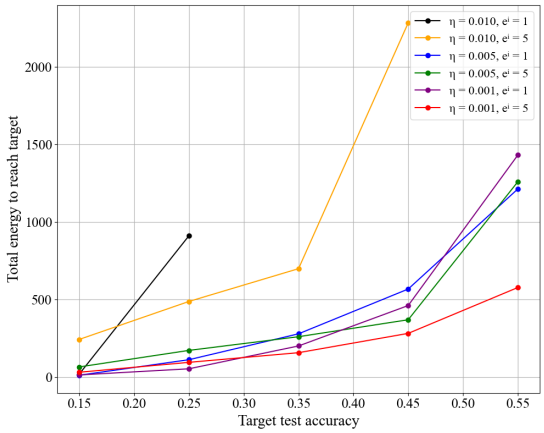


(b) Factor κ

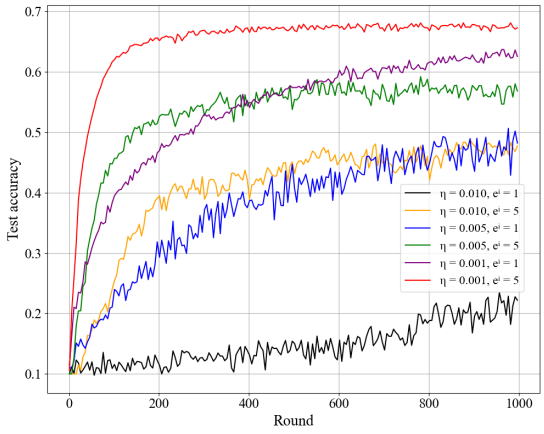
Fig. 2. Total energy to target accuracy over different factors, $I = 25$

Fig. 3a and Fig. 3b compare the energy–convergence trade-off and the convergence behavior with respect to the local

learning rate η and the number of local epochs e^i. From Fig. 3a and Fig. 3b, we observe that a larger e^i yields higher accuracy as well as faster and more stable convergence. However, as η increases, the convergence becomes slower, and the total energy consumption increases significantly to reach the target accuracy. That is, the parameters $\eta = 0.001, e^i = 5$ provide the best trade-off and convergence, i.e., achieving the target accuracy with fast and stable convergence and the lowest total energy consumption. Therefore, we selected the parameters $\eta = 0.001, e^i = 5$ as the default configuration of our proposed model.



(a) Energy-accuracy



(b) Convergence

Fig. 3. Compare with different η and e^i, $I = 25$

Fig. 4 demonstrates that the proposed OASIS algorithm maintains strong energy efficiency even as the number of devices in the network increases. Although the total cumulative energy consumption naturally rises with larger device populations, OASIS exhibits an almost linear growth pattern for the number of devices, indicating that the per-device energy cost remains nearly constant. This efficiency stems from OASIS's ability to judiciously schedule devices and allocate transmit power such that only the most informative yet energy-efficient devices are activated under the latency constraint. As a result, OASIS effectively prevents excessive energy usage despite having a larger selection pool. The results in Fig. 4 confirm that OASIS achieves excellent scalability, ensuring energy-efficient operation even in dense federated learning environments.

TABLE II
REQUIRED T FOR DIFFERENT I

Number of devices	Rounds to reach 70% accuracy
25	343 ± 85
50	168 ± 25
100	151 ± 7

Table II shows that OASIS achieves the target accuracy with fewer global communication rounds as the number of devices increases. Specifically, OASIS requires approximately 343, 168, and 151 rounds for networks with 25, 50, and 100 devices, respectively, demonstrating clear improvements in convergence speed as device count grows. This behavior stems from OASIS's ability to efficiently select the most informative devices and allocate power in a way that maximizes the learning contribution of each round. With a larger device pool, OASIS can exploit greater data diversity and richer local datasets while still satisfying the latency constraint, thereby accelerating convergence. Overall, it confirms that OASIS scales gracefully with the network size and consistently delivers faster convergence.

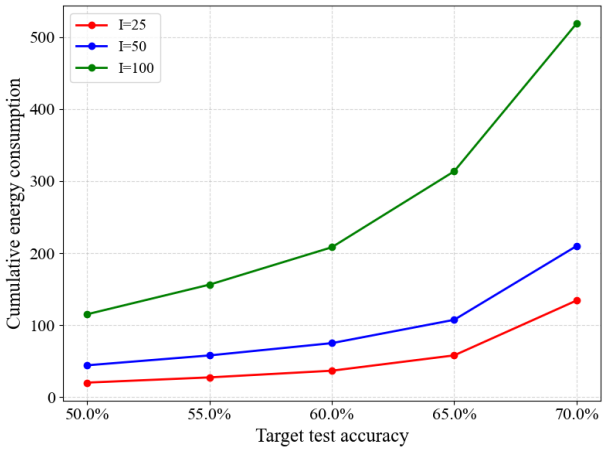


Fig. 4. Scalability: cumulative energy consumption to target accuracy

Fig. 5 illustrates that the proposed OASIS algorithm achieves rapidly decreasing average dynamic regret, demonstrating its strong adaptability to changing environments. From the earliest rounds, the regret Reg_T/T drops sharply, indicating that OASIS is able to closely track the optimal scheduling and power allocation decisions even under changing environments. As training progresses, the average regret continues to reduce and approaches zero when the path-length V_T grows sufficiently slowly, empirically validating the sublinear regret bound proven in the theoretical analysis. This behavior confirms that OASIS effectively balances exploration and exploitation, enabling it to make near-optimal decisions despite the dynamic nature of online federated learning. Overall, Fig. 5 verifies that OASIS maintains robust learning performance over time and adapts efficiently to evolving data distributions and device conditions.

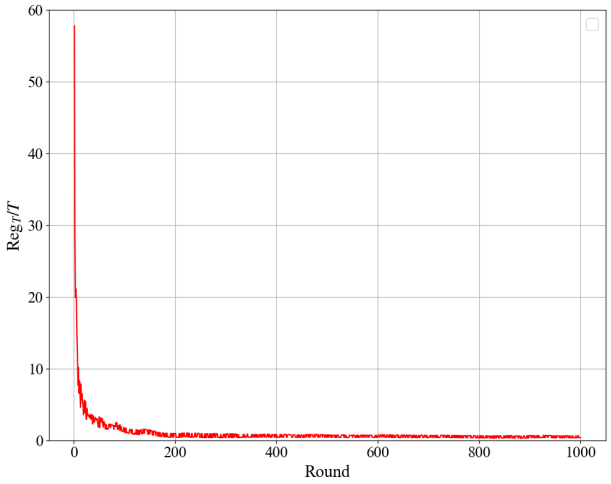


Fig. 5. Dynamic regret of the proposed approach, $I = 5$

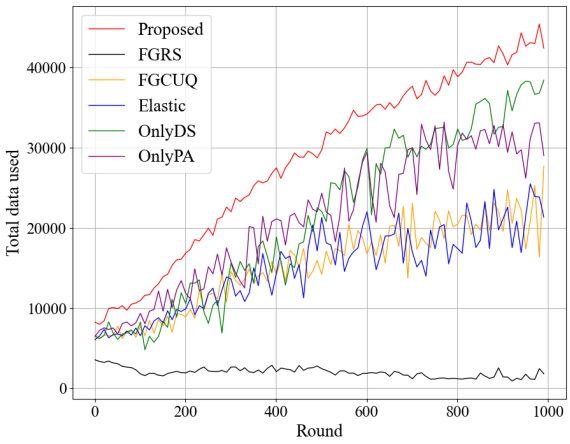


Fig. 6. Amount of data used, $\alpha = 0.2$, $I = 25$

C. Comparison with Benchmark Schemes

Fig. 6 clearly demonstrates that the proposed OASIS framework consistently exploits a substantially larger amount of data compared to all baseline methods. This performance gain stems from OASIS's CUCB-based device scheduling, which explicitly favors devices with larger and more informative local datasets while still respecting the latency constraint. As a result, OASIS systematically incorporates more fresh and diverse data into each training round, accelerating convergence and improving model quality. Next to the proposed OASIS, OnlyDS and OnlyPA show efficient data usage. However, OnlyPA lacks the principled selection mechanism that OASIS employs, and OnlyDS overlooks data-rich but slower devices, resulting in somewhat lower total data utilization. Other baselines, such as FGCUQ and ELASTIC, incorporate partial heuristics but fail to jointly optimize scheduling and power, leading to moderate performance. Random selection (FGRS) shows no meaningful improvement due to its inability to prioritize valuable devices. Overall, the results in Fig. 6 confirm that OASIS's joint optimization approach enables efficient data exploitation, which is a key factor behind its superior convergence speed and learning performance.

Fig. 7 shows that the proposed OASIS algorithm consistently achieves the highest test accuracy and reaches the target accuracy in the fewest communication rounds among all compared methods. While all schemes initially exhibit rapid accuracy improvement, OASIS maintains a clear advantage by continuously selecting informative devices and optimizing transmit power to ensure efficient model aggregation. As a result, OASIS not only converges faster but also attains a higher final accuracy with minimal fluctuation. In contrast, OnlyDS, OnlyPA, and ELASTIC demonstrate intermediate performance, as their partial or heuristic optimization limits their ability to balance data quality and energy efficiency effectively. FGRS performs the worst due to random device scheduling, which prevents it from consistently utilizing data-rich or reliable devices, leading to slow and noisy accuracy progression. Overall, Fig. 7 confirms that OASIS delivers superior learning performance in both convergence speed and final model accuracy, highlighting the benefit of its joint scheduling–power allocation design.

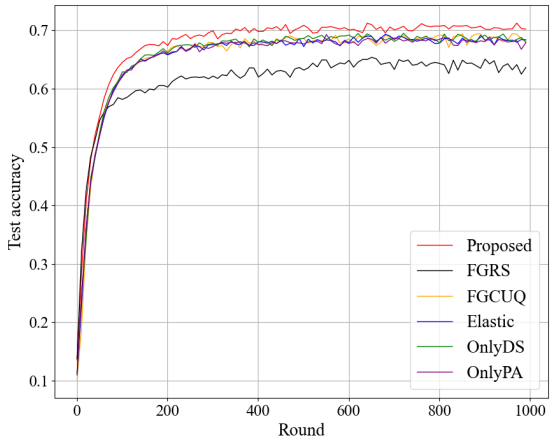


Fig. 7. Test accuracy, $\alpha = 0.2$, $I = 25$

Fig. 8 demonstrates that the proposed OASIS method achieves the lowest cumulative energy consumption across all target accuracy levels, clearly outperforming existing baseline schemes. This improvement results from the joint optimization of device scheduling and power allocation, which enables OASIS to effectively minimize unnecessary transmission energy while still leveraging high-quality data from selected devices. Notably, the energy gap between OASIS and the competing methods widens as the target accuracy increases, indicating that OASIS becomes even more advantageous in high-accuracy regimes. Compared with Elastic, for example, OASIS reduces the required energy by approximately 69.1% at a target accuracy of 0.675. Although OnlyPA shows better efficiency than ELASTIC, FGCUQ, and OnlyDS due to its optimized power control, it still lacks the coordinated scheduling mechanism that OASIS employs. Meanwhile, FGRS wastes

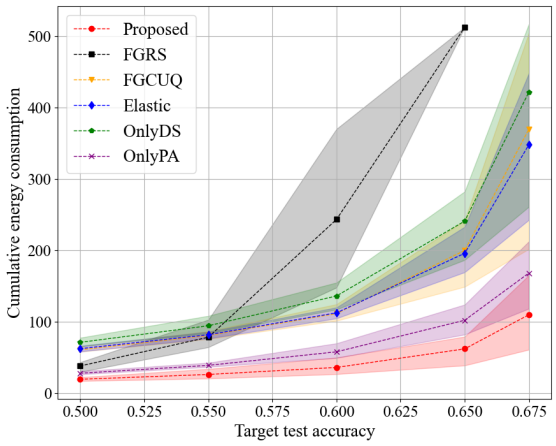


Fig. 8. Energy consumption over different methods, $\alpha = 0.2$, $I = 25$

energy by randomly selecting devices, and OnlyDS fails to reduce transmission energy because it does not optimize power. Overall, Fig. 8 confirms that the integrated scheduling–power control strategy of OASIS is essential for minimizing the total energy cost of federated learning under wireless latency constraints.

VII. Conclusion

In this work, we investigated an energy-efficient OFL approach. To achieve this, we formulated an OFL problem that jointly optimizes device scheduling and energy allocation under a per-round latency constraint. We developed a device scheduling and power allocation algorithm, referred to as OASIS. The proposed device scheduling is based on a CUCB-based bandit strategy. First, we designed a new reward function that considered transmission energy, local data size, and the average number of selections. We then derived a Lambert-$\mathbb{W}$ function-based power allocation for the scheduled devices in closed form. A dynamic-regret analysis demonstrates that the proposed approach adapts effectively to non-stationary environments and achieves sublinear dynamic regret in slowly changing optimal path environments. Experiments confirmed that the proposed algorithm achieves lower energy consumption and faster convergence compared to conventional baseline schemes. The performance gain increases as the target accuracy of the OFL becomes higher. In future work, we will investigate device- and time-adaptive strategies for selecting the learning rate and the number of local epochs based on local data characteristics and channel conditions. We also plan to extend the proposed framework to asynchronous or hierarchical OFL environments to further enhance its applicability.

Appendix

A. Proof of *Lemma 1*

Starting from the definition of the error term,

$$\mathbf{e}_t = \mathbf{g}_t - \nabla F(\mathbf{w}_t), \tag{49}$$

where $\mathbf{g}_t$ is defined by equation (8):

$$\mathbf{g}_t = \frac{1}{\sum_{j=1}^{J} k_t^j} \sum_{i=1}^{I} k_t^i \mathbf{g}_t^i. \tag{50}$$

Thus, the error term becomes

$$\mathbf{e}_t = \sum_{i=1}^{I} \sum_{(\mathbf{x},y)\in\mathcal{D}_t^i} \nabla f(\mathbf{w}_t;\mathbf{x},y) \left(\frac{k_t^i}{\sum_{j=1}^{I} k_t^j D_t^j} - \frac{1}{\sum_{j=1}^{I} D_t^j} \right). \tag{51}$$

We take the squared L_2 norm and expectation on $\mathbf{e}_t$. Then, according to Jensen's inequality, it becomes

$$\begin{aligned}
\mathbb{E}[\|\mathbf{e}_t\|^2] &= \mathbb{E}\Bigg[\Bigg\| \sum_{i=1}^{I} \left(\frac{k_t^i}{\sum_{j=1}^{I} k_t^j D_t^j} - \frac{1}{\sum_{j=1}^{I} D_t^j} \right) \\
&\quad \times \left(\sum_{(\mathbf{x},y)\in\mathcal{D}_t^i} \nabla f(\mathbf{w}_t;\mathbf{x},y) \right) \Bigg\|^2 \Bigg] \\
&\le \mathbb{E}\Bigg[I D_t \sum_{i=1}^{I} \left(\frac{k_t^i}{\sum_{j=1}^{I} k_t^j D_t^j} - \frac{1}{\sum_{j=1}^{I} D_t^j} \right)^2 \\
&\quad \times \sum_{(\mathbf{x},y)\in\mathcal{D}_t^i} \|\nabla f(\mathbf{w}_t;\mathbf{x},y)\|^2 \Bigg].
\end{aligned} \tag{52}$$

From **Assumption 2**, we have

$$\begin{aligned}
\mathbb{E}[\|\mathbf{e}_t\|^2] &\le \mathbb{E}\Bigg[I D_t \sum_{i=1}^{I} \left(\frac{k_t^i}{\sum_{j=1}^{I} k_t^j D_t^j} - \frac{1}{\sum_{j=1}^{I} D_t^j} \right)^2 \\
&\quad \times D_t^i \left(\phi + \psi \|\nabla F(\mathbf{w_t})\|^2 \right) \Bigg] \\
&\le \mathbb{E}\Bigg[\frac{I D_t}{\sum_{j=1}^{I} D_t^j} \left(\sum_{j=1}^{I} (1 - k_t^j) D_t^j \right) \\
&\quad \times \left(\phi + \psi \|\nabla F(\mathbf{w_t})\|^2 \right) \Bigg].
\end{aligned} \tag{53}$$

Here, we can take out the constant terms outside $\mathbb{E}(\cdot)$ and rearrange it. Then, we obtain

$$\mathbb{E}[\|\mathbf{e}_t\|^2] \le I \left(\sum_{j=1}^{I} (1 - k_t^j) D_t^j \right) \left(\phi + \psi \mathbb{E}\left[\|\nabla F(\mathbf{w_t})\|^2 \right] \right). \tag{54}$$

B. Proof of *Theorem 1*

Under **Assumption 1**, the function $F(\mathbf{w_{t+1}})$ is twice continuously differentiable. Then, the change in the function value from point $\mathbf{w}_t$ to $\mathbf{w}_{t+1}$ can be expressed using the second-order Taylor expansion as follows:

$$\begin{aligned}
F(\mathbf{w}_{t+1}) &= F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) \\
&\quad + \frac{1}{2} (\mathbf{w}_{t+1} - \mathbf{w}_t)^\top \nabla^2 F(\xi_t) (\mathbf{w}_{t+1} - \mathbf{w}_t),
\end{aligned} \tag{55}$$

where ξ_t is an arbitrary point on the line segment between $\mathbf{w}_t$ and $\mathbf{w}_{t+1}$. According to the definition of L-smoothness, the Hessian matrix $\nabla^2 F(\xi_t)$ satisfies:

$$\|\nabla^2 F(\xi_t)\|_2 \le L \quad \forall \mathbf{w} \in \mathbb{R}^M. \tag{56}$$

This implies that all eigenvalues of $\nabla^2 F(\xi_{\mathbf{t}})$ are bounded above by L. Therefore, we obtain the following inequality:

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) + \nabla F(\mathbf{w}_t)^\top (\mathbf{w}_{t+1} - \mathbf{w}_t) + \frac{L}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|^2. \tag{57}$$

Substituting the update rule $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{g}_t$,

$$F(\mathbf{w}_{t+1}) \leq F(\mathbf{w}_t) - \eta \nabla F(\mathbf{w}_t)^\top \mathbf{g}_t + \frac{L\eta^2}{2}\|\mathbf{g}_t\|^2. \tag{58}$$

Taking the expectation on both sides:

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t)] - \eta \mathbb{E}\left[\nabla F(\mathbf{w}_t)^\top \mathbf{g}_t\right] + \frac{L\eta^2}{2}\mathbb{E}\left[\|\mathbf{g}_t\|^2\right]. \tag{59}$$

Under the **Assumption 3** that the gradient estimator $\mathbf{g}_t$ is unbiased, it follows that:

$$\mathbb{E}\left[\nabla F(\mathbf{w}_t)^\top \mathbf{g}_t\right] = \mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right]. \tag{60}$$

Additionally, we can expand $\mathbb{E}\left[\|\mathbf{g}_t\|^2\right]$ as:

$$\begin{aligned} &\mathbb{E}\left[\|\mathbf{g}_t\|^2\right] \\ &= \mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] + 2\mathbb{E}\left[\langle \nabla F(\mathbf{w}_t), \mathbf{e}_t\rangle\right] + \mathbb{E}\left[\|\mathbf{e}_t\|^2\right]. \end{aligned} \tag{61}$$

Due to the unbiasedness of $\mathbf{g}_t$, the term $\mathbb{E}\left[\langle \nabla F(\mathbf{w}_t), \mathbf{e}_t\rangle\right]$ vanishes, i.e., $\mathbb{E}\left[\langle \nabla F(\mathbf{w}_t), \mathbf{e}_t\rangle\right] = 0$. Thus, the expression simplifies to:

$$\mathbb{E}\left[\|\mathbf{g}_t\|^2\right] = \mathbb{E}\left[\|\nabla F(\mathbf{w}_t)\|^2\right] + \mathbb{E}\left[\|\mathbf{e}_t\|^2\right]. \tag{62}$$

Substituting this back into the inequality, we obtain:

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t)] - \eta\left(1 - \frac{L\eta}{2}\right)\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] + \frac{L\eta^2}{2}\mathbb{E}[\|\mathbf{e}_t\|^2]. \tag{63}$$

Using **Lemma 1** to bound $\mathbb{E}[\|\mathbf{e}_t\|_2^2]$,

$$\mathbb{E}[F(\mathbf{w}_{t+1})] \leq \mathbb{E}[F(\mathbf{w}_t)] - \eta\left(1 - \frac{L\eta}{2}\right)\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] + \frac{IL\eta^2}{2}\sum_{j=1}^{I}(1 - k_t^j)D_t^j \times \left(\phi + \psi \mathbb{E}\left[\|\nabla F(\mathbf{w_t})\|^2\right]\right). \tag{64}$$

Simplifying the terms involving $\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|_2^2]$, and summing over $t = 1$ to T,

$$\begin{aligned} &\mathbb{E}[F(\mathbf{w}_1)] - \mathbb{E}[F(\mathbf{w}_{T+1})] \\ &\geq \eta \sum_{t=1}^{T}\left(1 - \frac{L\eta}{2}\left(1 - I\psi\sum_{j=1}^{I}(1 - k_t^j)D_t^j\right)\right) \\ &\quad \times \mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] - \sum_{t=1}^{T}\frac{IL\phi\eta^2}{2}\sum_{j=1}^{I}(1 - k_t^j)D_t^j. \end{aligned} \tag{65}$$

Assuming that $F(\mathbf{w}_{T+1}) \geq F^*$, we have

$$\begin{aligned} &\mathbb{E}[F(\mathbf{w}_1)] - F^* \\ &\geq \eta \sum_{t=1}^{T}\left(1 - \frac{L\eta}{2}\left(1 - I\psi\sum_{j=1}^{I}(1 - k_t^j)D_t^j\right)\right) \\ &\quad \times \mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] - \sum_{t=1}^{T}\frac{IL\phi\eta^2}{2}\sum_{j=1}^{I}(1 - k_t^j)D_t^j. \end{aligned} \tag{66}$$

Let us define A_t as follows:

$$A_t = \eta\left(1 - \frac{L\eta}{2}\left(1 - I\psi\sum_{j=1}^{I}(1 - k_t^j)D_t^j\right)\right). \tag{67}$$

Since $k_t^j \in \{0, 1\}, \forall j \in \mathcal{I}$, we know that:

$$\eta\left(1 - \frac{L\eta}{2}\left(1 - I\psi D_t\right)\right) \leq A_t \leq \eta\left(1 - \frac{L\eta}{2}\right). \tag{68}$$

As a result, we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla F(\mathbf{w}_t)\|^2] \leq \frac{2\left(\mathbb{E}[F(\mathbf{w}_1)] - F^*\right)}{\eta T\left(2 - L\eta\left(1 - I\psi D_t\right)\right)} + \sum_{t=1}^{T}\frac{I\phi\eta\sum_{j=1}^{I}(1 - k_t^j)D_t^j}{T\left(1 - I\psi D_t\right)}. \tag{69}$$

*C. Proof of **Theorem 2***

For (35), the Karush–Kuhn–Tucker (KKT) condition is

$$\frac{\partial \mathcal{L}}{\partial P_t^i(\ell)} = 0, \tag{70}$$

$$\lambda_1^i k_t^i(\ell)(P_t^i(\ell) - P_{\max}^i) = 0, \tag{71}$$

$$\lambda_2^i k_t^i P_t^i(\ell) = 0, \tag{72}$$

$$\lambda_3\Big(\sum_{i=1}^{I} k_t^i(\ell)\tau_t^i(\ell) - \tau\Big) = 0, \tag{73}$$

$$(24b), (24c) \tag{74}$$

$$\lambda_1^i, \lambda_2^i, \lambda_3 \geq 0. \tag{75}$$

These conditions imply (i) $P_t^i(\ell) = 0$ or (ii) $P_t^i(\ell) = P_{\max}^i$ or (iii) $\lambda_1^i = \lambda_2^i = 0$. For cases (i) and (ii), power is fixed at the boundary. For the case (iii), the following can be obtained from $\frac{\partial \mathcal{L}}{\partial P_t^i(\ell)} = 0$:

$$R_t^i(\ell) = \frac{W\gamma_t^i}{\ln 2}\frac{P_t^i(\ell) + \lambda_3 k_t^i(\ell)}{1 + \gamma_t^i P_t^i(\ell)}, \tag{76}$$

where $R_t^i(\ell)$ denotes the transmission rate R_t^i in each iteration ℓ. Let $v_t^i := 1 + \gamma_t^i P_t^i$. Then, we obtain

$$v_t^i\left[\ln v_t^i - 1\right] = \Gamma_t^i(\ell), \tag{77}$$

where $\gamma_t^i = |h_t^i|^2/WN_0$ and $\Gamma_t^i(\ell) = \gamma_t^i\lambda_3 k_t^i(\ell) - 1$. This can be solved via the Lambert-$\mathbb{W}$ function as

$$v_t^i = \frac{\Gamma_t^i(\ell)}{\mathbb{W}\big(\Gamma_t^i(\ell)/e\big)}. \tag{78}$$

This completes the derivation as,

$$P_t^i(\ell) = \begin{cases} 0, & \lambda_1^i = 0, \lambda_2^i > 0, \\ \dfrac{1}{\gamma_t^i}\left[\dfrac{\Gamma_t^i(\ell)}{\mathbb{W}(\Gamma_t^i(\ell)/e)} - 1\right], & \lambda_1^i = \lambda_2^i = 0, \\ P_{\max}^i, & \lambda_1^i > 0, \lambda_2^i = 0. \end{cases} \tag{79}$$

The multiplier λ_3 is iteratively determined with $P_t^i(\ell)$ by a bisection search on the $c(\lambda_3) := \sum_{i=1}^{I} k_t^i(\ell)\tau_t^i(\ell) - \tau$. When $\lambda_3 = 0$ yields $c(\lambda_3) > 0$, we iteratively increase λ_3 via bisection until a positive value satisfying $c(\lambda_3) = 0$ is found.

D. Proof of **Property 1**

Let $u_t^\star$ denote the minimum value of $u_t(\cdot)$ in round t. Under **Assumption 5** and **Property 2**, OASIS is updated by exact minimization. Standard results on alternating minimization for strongly convex and smooth objectives then guarantee a linear convergence rate: there exists a constant $\Xi \in (0,1)$, such that

$$u_t(\ell) - u_t^\star \;\le\; \Xi\big(u_t(\ell-1) - u_t^\star\big), \qquad \forall \ell \ge 1. \tag{80}$$

By recursion,

$$u_t(\ell) - u_t^\star \;\le\; \Xi^\ell\big(u_t(0) - u_t^\star\big), \qquad \forall \ell \ge 0. \tag{81}$$

Since $u_t(\ell)$ is nonincreasing, we have

$$|u_t(\ell) - u_t(\ell-1)| \;\le\; u_t(\ell-1) - u_t^\star \;\le\; \Xi^{\ell-1}\big(u_t(0) - u_t^\star\big). \tag{82}$$

From (82), a sufficient condition for the stopping criterion $|u_t(\ell) - u_t(\ell-1)| < \epsilon$ is

$$\Xi^{\ell-1}\big(u_t(0) - u_t^\star\big) \le \epsilon. \tag{83}$$

Since $0 < \Xi < 1$ and $u_t(0) > u_t^\star$, (83) is equivalent to

$$\Xi^{\ell-1} \le \frac{\epsilon}{u_t(0) - u_t^\star}. \tag{84}$$

Taking the natural logarithm on both sides yields

$$(\ell-1)\log\Xi \;\le\; \log\epsilon - \log\big(u_t(0) - u_t^\star\big). \tag{85}$$

Because $\log\Xi < 0$, dividing by $\log\Xi$ reverses the inequality and gives

$$\ell \;\ge\; 1 + \frac{\log\big(u_t(0) - u_t^\star\big)}{|\log\Xi|} + \frac{1}{|\log\Xi|}\log\left(\frac{1}{\epsilon}\right). \tag{86}$$

Define

$$L_t(\epsilon) := 1 + \frac{\log\big(u_t(0) - u_t^\star\big)}{|\log\Xi|} + \frac{1}{|\log\Xi|}\log\left(\frac{1}{\epsilon}\right). \tag{87}$$

Then any integer $\ell \ge L_t(\epsilon)$ satisfies (83), and hence $|u_t(\ell) - u_t(\ell-1)| < \epsilon$. Let

$$\ell_t(\epsilon) := \min\{\ell \in \mathbb{N} : |u_t(\ell) - u_t(\ell-1)| < \epsilon\}$$

denote the number of iterations required to reach this accuracy. By definition of $\ell_t(\epsilon)$ and the fact that $\lceil x \rceil \le x+1$, we obtain

$$\begin{aligned} \ell_t(\epsilon) \;&\le\; \big\lceil L_t(\epsilon)\big\rceil \;\le\; L_t(\epsilon) + 1 \\ &= \underbrace{\left(2 + \frac{\log\big(u_t(0) - u_t^\star\big)}{|\log\Xi|}\right)}_{=:C_1} + \underbrace{\frac{1}{|\log\Xi|}}_{=:C_2}\log\left(\frac{1}{\epsilon}\right). \end{aligned} \tag{88}$$

Thus, $\ell_t(\epsilon) \le C_1 + C_2\log(1/\epsilon)$ for some constants $C_1, C_2 > 0$ that depend only on Ξ and the initial gap $u_t(0) - u_t^\star$, but are independent of ϵ. This completes the proof.

E. Proof of **Property 2**

We compute the second derivative of $E_t^i(P_t^i)$ with respect to P_t^i, as follows:

$$\frac{d^2 E_t^i(P_t^i)}{d(P_t^i)^2} = \frac{B}{W} \cdot \frac{\log_2(e)}{(\log_2(1+\gamma_t^i P_t^i))^3} \cdot H(P_t^i), \tag{89}$$

where $H(P_t^i) = v_t^i \log_2 v_t^i - \gamma_t^i P_t^i$. Since $H(P_t^i) \ge 0$ for $P_t^i \ge 0$, we conclude that

$$\frac{d^2 E_t^i}{d(P_t^i)^2} \ge 0. \tag{90}$$

F. Proof of **Lemma 2**

Let $\zeta_t^i = \sqrt{\frac{3\ln t}{2\sum_{p=1}^{t-1} k_p^i}}$, by Hoeffding's inequality for bounded rewards, we have:

$$\mathbb{P}\big(|\hat{r}_{t-1}^i - \mathbb{E}[r_t^i]| > \zeta_t^i\big) \le 2e^{-2\sum_{p=1}^{t-1} k_p^i \zeta_t^{i2}}. \tag{91}$$

If a suboptimal device $i \ne i^\star$ is chosen in round t while all estimates lie within their confidence intervals, then

$$\mathbb{E}[r_t^i] + \kappa\sqrt{\frac{3\ln t}{2\sum_{p=1}^{t-1} k_p^i}} \ge \mathbb{E}[r_t^{i^\star}] + \kappa\sqrt{\frac{3\ln t}{2\sum_{p=1}^{t-1} k_p^{i^\star}}}. \tag{92}$$

Rearranging the previous inequality gives, for all $t \le T$,

$$\sum_{t=1}^{T} k_t^i \le \frac{6\kappa^2}{(\Delta_t^i)^2}\ln T. \tag{93}$$

Then we obtain:

$$\begin{aligned} \sum_{t=1}^{T}\big[u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)\big] &\le \sum_{i=1}^{I}\Delta_t^i \sum_{t=1}^{T} k_t^i + \mathcal{O}(1) \\ &\le \sum_{i=1}^{I}\frac{6\kappa^2}{\Delta_t^i}\ln T + \mathcal{O}(1). \end{aligned} \tag{94}$$

The power allocation relies on **Property 2** and **Assumption 5**. Consequently, the optimization error satisfies:

$$u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_t, \mathbf{P}_t^*) \le \frac{L^2}{2\mu} \cdot \frac{1}{\sqrt{T}}. \tag{95}$$

Therefore,

$$\begin{aligned} &\sum_{t=1}^{T}\big[u_t(\mathbf{k}_t, \mathbf{P}_t) - u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)\big] \\ &\qquad \le \sum_{i\in\mathcal{I}}\frac{6\kappa^2}{\Delta^i}\ln T + \mathcal{O}(1/\sqrt{T}) + \mathcal{O}(1). \end{aligned} \tag{96}$$

G. Proof of **Theorem 3**

The subgradient satisfies the first-order optimality condition:

$$\big(\nabla u_t(\mathbf{k}_t^*, \mathbf{P}_t^*)\big)^\top\big((\mathbf{k}_t, \mathbf{P}_t) - (\mathbf{k}_t^*, \mathbf{P}_t^*)\big) \ge 0, \quad \forall x. \tag{97}$$

Taking $(\mathbf{k}_t, \mathbf{P}_t) = (\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*)$, we obtain:

$$\begin{aligned} &u_t(\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*) - u_t(\mathbf{k}_t^*, \mathbf{P}_t^*) \\ &\quad \le \nabla u_t(\mathbf{k}_t^*, \mathbf{P}_t^*)^\top\big((\mathbf{k}_{t-1}^*, \mathbf{P}_{t-1}^*) - (\mathbf{k}_t^*, \mathbf{P}_t^*)\big). \end{aligned} \tag{98}$$

Applying Cauchy–Schwarz and using the **Assumption 5**, we can bound the drift term:

$$u_t(\mathbf{k}^*_{t-1}, \mathbf{P}^*_{t-1}) - u_t(\mathbf{k}^*_t, \mathbf{P}^*_t) \leq L\|(\mathbf{k}^*_{t-1}, \mathbf{P}^*_{t-1}) - (\mathbf{k}^*_t, \mathbf{P}^*_t)\|. \tag{99}$$

Summing over t gives in **Definition 1**:

$$\sum_{t=2}^{T} \left[u_t(\mathbf{k}^*_{t-1}, \mathbf{P}^*_{t-1}) - u_t(\mathbf{k}^*_t, \mathbf{P}^*_t)\right] \leq L \sum_{t=2}^{T} \|(\mathbf{k}^*_{t-1}, \mathbf{P}^*_{t-1}) - (\mathbf{k}^*_t, \mathbf{P}^*_t)\| = LV_T. \tag{100}$$

Thus, the drift term is bounded by LV_T, where V_T is the total variation in the optimal decision sequence. Then, Reg_T is derived as follows by using the bound on the drift term above and the bound on the tracking term provided in **Lemma 2**,

$$\mathrm{Reg}_T \;\leq\; \sum_{i\in\mathcal{I}} \frac{6\kappa^2}{\Delta^i} \ln T + LV_T + \mathcal{O}(1/\sqrt{T}). \tag{101}$$

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[2] B. Zhao, L. Wang, Z. Liu, Z. Zhang, J. Zhou, C. Chen, and M. Kolar, "Adaptive Client Sampling in Federated Learning via Online Learning with Bandit Feedback," *Journal of Machine Learning Research*, vol. 26, pp. 1–67, 2025.

[3] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Foundations and Trends® in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2012.

[4] V. C. Gogineni, S. Werner, Y.-F. Huang, and A. Kuh, "Communication-efficient online federated learning framework for nonlinear regression," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 5228–5232.

[5] G. Damaskinos, R. Guerraoui, A.-M. Kermarrec, V. Nitu, R. Patra, and F. Taiani, "Fleet: Online federated learning via staleness awareness and performance prediction," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 13, no. 5, pp. 1–30, 2022.

[6] S. Wang, M. Lee, S. Hosseinalipour, R. Morabito, M. Chiang, and C. G. Brinton, "Device sampling for heterogeneous federated learning: Theory, algorithms, and implementation," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[7] X. Chen, Z. Li, W. Ni, X. Wang, S. Zhang, Y. Sun, S. Xu, and Q. Pei, "Towards dynamic resource allocation and client scheduling in hierarchical federated learning: A two-phase deep reinforcement learning approach," *IEEE Transactions on Communications*, 2024.

[8] L. Su, R. Zhou, N. Wang, G. Fang, and Z. Li, "An online learning approach for client selection in federated edge learning under budget constraint," in *Proceedings of the 51st International Conference on Parallel Processing*, 2022, pp. 1–11.

[9] B. Deressa and M. A. Hasan, "Trustbandit: Optimizing client selection for robust federated learning against poisoning attacks," in *IEEE INFOCOM 2024-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2024, pp. 1–8.

[10] W. Xu, B. Liang, G. Boudreau, and H. Sokun, "Clipper: Online joint client sampling and power allocation for wireless federated learning," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 2024.

[11] J. Perazzone, S. Wang, M. Ji, and K. S. Chan, "Communication-efficient device scheduling for federated learning using stochastic optimization," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1449–1458.

[12] Y. Jin, L. Jiao, Z. Qian, S. Zhang, and S. Lu, "Budget-aware online control of edge federated learning on streaming data with stochastic inputs," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3704–3722, 2021.

[13] G. Liu, X. Ma, Y. Yang, C. Wang, and J. Liu, "Federaser: Enabling efficient client-level data removal from federated learning models," in *2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS)*. IEEE, 2021, pp. 1–10.

[14] N. Babendererde, M. Fuchs, C. Gonzalez, Y. Tolkach, and A. Mukhopadhyay, "Jointly exploring client drift and catastrophic forgetting in dynamic learning," *Scientific Reports*, vol. 15, no. 1, p. 5857, 2025.

[15] C.-H. Hu, Z. Chen, and E. G. Larsson, "Dynamic scheduling for federated edge learning with streaming data," in *2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*. IEEE, 2023, pp. 1–5.

[16] Y. Chen, Y. Ning, M. Slawski, and H. Rangwala, "Asynchronous online federated learning for edge devices with non-iid data," in *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2020, pp. 15–24.

[17] B. Ganguly and V. Aggarwal, "Online federated learning via non-stationary detection and adaptation amidst concept drift," *IEEE/ACM Transactions on Networking*, vol. 32, no. 1, pp. 643–653, 2023.

[18] W. Xia, T. Q. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu, "Multi-armed bandit-based client scheduling for federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7108–7123, 2020.

[19] D. Xu, "Latency minimization for tdma-based wireless federated learning networks," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 13 974–13 979, 2024.

[20] X. Mo and J. Xu, "Energy-efficient federated edge learning with joint communication and computation design," 2020. [Online]. Available: https://arxiv.org/abs/2003.00199

[21] M. Fu, Y. Shi, and Y. Zhou, "Federated learning via unmanned aerial vehicle," *IEEE Transactions on Wireless Communications*, vol. 23, no. 4, pp. 2884–2900, 2024.

[22] K. Wang, Y. Ma, M. B. Mashhadi, C. H. Foh, R. Tafazolli, and Z. Ding, "Convergence acceleration in wireless federated learning: A stackelberg game approach," *IEEE Transactions on Vehicular Technology*, 2024.

[23] 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 transactions on wireless communications*, vol. 20, no. 1, pp. 269–283, 2020.

[24] M. P. Friedlander and M. Schmidt, "Hybrid deterministic-stochastic methods for data fitting," *SIAM Journal on Scientific Computing*, vol. 34, no. 3, pp. A1380–A1405, 2012.

[25] Y. Sun, Z. Lin, Y. Mao, S. Jin, and J. Zhang, "Channel and gradient-importance aware device scheduling for over-the-air federated learning," *IEEE Transactions on Wireless Communications*, vol. 23, no. 7, pp. 6905–6920, 2023.

[26] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *International conference on machine learning*. PMLR, 2013, pp. 151–159.

[27] O. Besbes, Y. Gur, and A. Zeevi, "Stochastic multi-armed-bandit problem with non-stationary rewards," *Advances in neural information processing systems*, vol. 27, 2014.

[28] S. Basu, R. Sen, S. Sanghavi, and S. Shakkottai, "Blocking bandits," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[29] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[30] R. S. Trindade, C. d'Ambrosio, A. Frangioni, and C. Gentile, "Comparing perspective reformulations for piecewise-convex optimization," *Operations Research Letters*, vol. 51, no. 6, pp. 702–708, 2023.

[31] K. Zhu, F. Zhang, L. Jiao, B. Xue, and L. Zhang, "Client selection for federated learning using combinatorial multi-armed bandit under long-term energy constraint," *Computer Networks*, vol. 250, p. 110512, 2024.

[32] B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari, "Tight regret bounds for stochastic combinatorial semi-bandits," in *Artificial Intelligence and Statistics*. PMLR, 2015, pp. 535–543.

[33] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 647–662, 2015.

[34] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 928–936, 2003.

[35] T. Yang, L. Zhang, R. Jin, and J. Yi, "Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient," in *International Conference on Machine Learning*. PMLR, 2016, pp. 449–457.

[36] W. Shi, S. Zhou, Z. Niu, M. Jiang, and L. Geng, "Joint device scheduling and resource allocation for latency constrained wireless federated learn-

ing," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 453–467, 2020.
[37] J. Yao and N. Ansari, "Enhancing federated learning in fog-aided iot by cpu frequency and wireless power control," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3438–3445, 2020.
[38] M. S. Al-Abiad, M. Z. Hassan, and M. J. Hossain, "Energy-efficient resource allocation for federated learning in noma-enabled and relay-assisted internet of things networks," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 24 736–24 753, 2022.
[39] L. Yu, R. Albelaihi, X. Sun, N. Ansari, and M. Devetsikiotis, "Jointly optimizing client selection and resource management in wireless federated learning for internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 6, pp. 4385–4395, 2021.

Jaemin Kim received the B.S. and M.S. degrees in the school of computer science and engineering from Chung-Ang University, Seoul, South Korea, in 2023 and 2025, respectively. He is currently pursuing a Ph.D. degree in the school of computer science and engineering at Chung-Ang University, Seoul, South Korea. His research interests include wireless communication networks, semantic communications, O-RAN, online convex optimization, online learning, and federated learning.

Junsuk Oh received the B.S. and M.S. degrees in the school of computer science and engineering from Chung-Ang University, Seoul, South Korea, in 2021 and 2023, respectively. He is currently pursuing a Ph.D. degree in the school of computer science and engineering at Chung-Ang University, Seoul, South Korea. His research interests include wireless communication networks, distributed computing systems, over-the-air computation, signal processing, data compression, and federated learning.

Wonjong Noh received the B.S., M.S., and Ph.D. degrees from the Department of Electronics Engineering, Korea University, Seoul, Korea, in 1998, 2000, and 2005, respectively. From 2005 to 2007, he conducted his postdoctoral research at Purdue University, IN, USA, and the University of California at Irvine, CA, USA. From 2008 to 2014, he was a Principal Research Engineer with Samsung Advanced Institute of Technology, Samsung Electronics, Korea. He is currently an Professor at the School of Software at Hallym University, College of Information Science, Korea. His current research interests include fundamental capacity analysis and optimizations in 5G/6G wireless communication and networks, intelligent LEO satellite networks, federated mobile edge computing, machine learning-based systems control, and big-data based healthcare and medical system design. He received a government postdoctoral fellowship from the Ministry of Information and Communication, Korea, in 2005. He was also a recipient of the Samsung Best Paper Gold Award in 2010, the Samsung Patent Bronze Award in 2011, and the Samsung Technology Award in 2013. He has served numerous international conferences as a TPC member or an organizing committee member such as IEEE Globecom, WCNC, ICOIN, ICTC, ICUFN, ICMIC, JCCI and ICAIIC.

Sungrae Cho is a professor with the school of computer sciences and engineering, Chung-Ang University (CAU), Seoul. Prior to joining CAU, he was an assistant professor with the department of computer sciences, Georgia Southern University, Statesboro, GA, USA, from 2003 to 2006, and a senior member of technical staff with the Samsung Advanced Institute of Technology (SAIT), Kiheung, South Korea, in 2003. From 1994 to 1996, he was a research staff member with electronics and telecommunications research institute (ETRI), Daejeon, South Korea. From 2012 to 2013, he held a visiting professorship with the national institute of standards and technology (NIST), Gaithersburg, MD, USA. He received the B.S. and M.S. degrees in electronics engineering from Korea University, Seoul, South Korea, in 1992 and 1994, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2002. He has been an associate member of National Academy of Engineering of Korea (NAEK) since 2025 and a KICS fellow since 2021. He received numerous awards including the Haedong Best Researcher of the Year in Telecommunications in 2022 and the Award from Korean Ministry of Science and ICT in 2021.

His current research interests include wireless networking, network intelligence, and network optimization. He has been an editor-in-chief (EIC) of ICT Express (Elsevier) since 2024, a subject editor of IET Electronics Letter since 2018, an executive editor of Wiley Transactions on Emerging Telecommunications Technologies since 2023, and was an area editor of Ad Hoc Networks Journal (Elsevier) from 2012 to 2017. He has served numerous international conferences as a general chair, TPC chair, or an organizing committee chair, such as IEEE ICC, IEEE SECON, IEEE ICCE, ICOIN, ICTC, ICUFN, APCC, TridentCom, and the IEEE MASS.