Towards Energy Efficient Federated Meta-learning in Edge Network

Xubo Li*, Yuanjie Jia†, Yingyu Li‡, Yong Xiao*§¶
*School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China
†Wuhan Research Institute of Posts and Telecommunications, Wuhan, China
‡School of Mechanical Engineering and Electronic Information, China University of Geosciences, Wuhan, China
§Peng Cheng Laboratory, Shenzhen, China
¶Pazhou Lab, Guangzhou, China

***Abstract*—There is still lacking a simple and comprehensive framework to model and optimize the overall energy consumption of an FEI network, especially in heterogeneous scenarios. This paper proposes a comprehensive framework to characterize the overall energy consumption of FEI networks. The computation and communication overhead as well as the number of coordination rounds required to train a satisfactory model are analytically modeled and evaluated. We investigate and compare the energy consumption of FEI networks with two popular distributed algorithmic implementations: FedAvg and FedMeta. We observe that although FedMeta consumes more energy than FedAvg in each single coordination round, the overall energy consumption of FedMeta is much lower than that of FedAvg. Finally, we evaluate the energy consumption of both algorithms based on a hardware prototype. Numerical results show that the overall energy consumption of FedMeta is 77.9% less than that of FedAvg.**

***Index Terms*—Energy efficient, federated meta-learning, edge network.**

I. Introduction

Recently, the energy sustainability and impact to the climate change of telecommunication networks have attracted significant interest. Telecommunication network is already identified as one of the top carbon dioxide emission sources, counting around 2% of global carbon emissions [1]. Recent efforts of integrating AI and telecommunication network will further accelerate this trend, hindering the global efforts in reducing energy consumption and achieving net-zero emissions in 2023 and beyond [2].

Federated edge intelligence (FEI) is a promising paradigm that implements federated learning-based distributed machine learning solutions in an edge computing network. It enables collaborative construction of models based on decentralized datasets without compromising data privacy and is therefore considered as one of the key candidate technologies for implementing network AI in 6G [3].

Despite of its promising potential, recent results suggest that the FEI networks may cause a higher energy consumption and result in more carbon emissions compared to the traditional centralized approach. In particular, a recent study reports that a FEI network may emit up to two orders of magnitude more carbon dioxide when training the same performance model than the centralized approach [4]. This is because the model training process of an FEI network may involve a large number of edge servers deployed in a wide geographical area with various storage, communication, and computational capacities, which make it difficult to model and optimize the overall energy consumption of the networking systems. Also, the model training process involves multiple rounds of model coordination and the total number of coordination rounds required to train a satisfactory model is closely related to the probability distributions of datasets at the edge servers. Generally speaking, more number of global coordination rounds is required when the divergence between probability distributions of datasets at different edge servers is higher. In other words, there is still lacking a simple and comprehensive framework to model and optimize the overall energy consumption of an FEI network, especially for the heterogeneous edge server capabilities and high divergence of dataset distributions.

In this paper, we introduce a novel framework that can characterize the overall energy consumption of an FEI network. Based on our proposed framework, we investigate the energy consumption of FEI networks when different model training algorithmic solutions are implemented. In particular, we derive analytical framework for modeling and optimizing energy consumption of FEI networks with two popular algorithmic solutions: *FedAvg* which allows collaboratively training of a globally shared model by all edge servers, and *FedMeta* which allows all the edge servers to first collaboratively train a shared initial model which will then be distributed to each edge server for further training based on its local dataset. We conduct extensive experiments to evaluate and compare the overall energy required for both solutions in communication, i.e., model coordination, and computations, i.e., local model training for both solutions, for training the same performance model. Our results suggest that the overall energy consumption of FedMeta is 77.9% less than the FedAvg for FEI networks with non-i.i.d. datasets.

The main contributions of this paper are summarized as follows:

- **Energy consumption modeling:** We establish a simple and comprehensive framework to characterize the overall energy consumption of an FEI network.
- **Analytical solutions:** Based on the proposed model, we

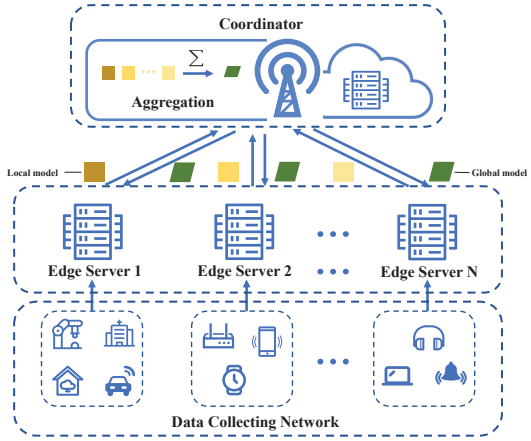


Fig. 1: System model

derive analytical solutions of overall energy consumption for two popular algorithm, FedAvg and FedMeta. We compare the overall energy required for both solutions in communication and computation.

- **Extensive experiments:** We develop a hardware prototype and conduct extensive measurements for the energy consumption under different setups. Numerical results verify our analysis.

II. System Model and Problem Formulation

A. System Model

Consider a model training in edge network involving a set $\mathcal{N}$ of N edge servers. The network consists of the following three key components as shown in Fig.1:

(1) *Data collecting network:* It comprises a diverse array of devices tasked with gathering local data samples to be uploaded to the corresponding edge servers.

(2) *Edge servers:* The edge servers are deployed to oversee the devices within the data collection network, dynamically updating the local model based on the incoming data samples from the data collecting network. In this paper, we posit the assumption that each edge server is associated with a distinct set of data collecting devices and has an exclusive data sample set $\mathcal{S}_n$.

(3) *Coordinator:* It can be deployed at the cloud data center or one of the edge servers, thereby enabling seamless coordination of model training across multiple edge servers.

We use subscript t to denote the model training related parameters between the t-th and the $(t+1)$-th round of global model coordination, i.e., let $w_{n,t}$ be the locally trained model parameters of edge server n and w_t be the globally updated model sent by the coordinator at the beginning of the t-th round of coordination. The main steps in each round t are described as follows:

(1) **Data collection:** Each device in the network first uploads the required number of data samples to its associated edge server n.

(2) **Local model training:** In each model training round, each edge server downloads the current model from the coordinator and then performs e steps of local model training.

(3) **Model uploading:** After e rounds of local training, each edge server n obtains an updated local model parameter $w_{n,t}^e$ which will then be uploaded to the coordinator.

(4) **Model aggregation and updating:** Upon receiving updated model parameters from all the edge severs, the coordinator calculates the updated global model parameters through model aggregation as follows:

$$w_{t+1} \leftarrow \frac{1}{N} \sum_{n \in \mathcal{N}} w_{n,t}^e \tag{1}$$

B. Problem Formulation

The energy consumption of each step of the edge servers during each round of global model coordination can be modeled as follows:

(1) **Energy Consumption of Data Collection:** is dominated by the energy consumed by the devices in transmitting the data samples. We use $E_n^{dc}(K)$ to denote the amount of energy consumed by a set of devices for uploading K data samples to edge server n. We can write $E_n^{dc}(K)$ as:

$$E_n^{dc}(K) = \eta K \tag{2}$$

where η is the normalized energy consumption for devices to upload each data sample.

(2) **Energy Consumption of Local Model Training:** is mainly affected by the energy consumption of local computation and data processing. Let $E_n^{mt}(e, K)$ be the energy consumption of edge server n to train e local steps. Following [5], we can write the energy consumption for each local step of model training at edge server n as $E_n^{mt} = c_0 K + c_1$, where c_0 characterizes the energy consumed for computing each data sample and c_1 is the constant capturing the energy consumption that is unrelated to the computational load, i.e., stationary energy for most computing devices. Based on the above discussion, we can write the energy consumption of local model training at edge server n as

$$E_n^{mt}(e, s) = c_0 e K + c_1 e \tag{3}$$

(3) **Energy Consumption of Model Uploading:** is the energy to transmit the local model to the coordinator. Let E_n^{mu} be the energy consumed by edge server n to upload its locally trained model. We can write the total energy consumed by the edge servers for model uploading as

$$E^{mu} = \sum_{n \in \mathcal{N}} E_n^{mu} \tag{4}$$

In this paper, based on the above energy model, we focus on energy minimization problem of the system which can be written as follows:

$$\min_{K,T} \quad \mathbb{E}\left[E(K,T)\right] \tag{5a}$$

$$\text{s.t.} \quad \frac{1}{eT} \sum_{t=0}^{T-1} \sum_{\tau=0}^{e-1} \mathbb{E}\left[\left\|\nabla \mathcal{R}\left(\bar{w}_{t+1}^{\tau}\right)\right\|^2\right] \leq \epsilon, \tag{5b}$$

$$e, K, N, T \in \mathbb{Z}^+ \tag{5c}$$

where $E(K,T) = \sum_{t=1}^{T} \sum_{n\in\mathcal{N}} (E_n^{dc} + E_n^{mt} + E_n^{mu})$. $\bar{w}_{t+1}^{\tau} = \frac{1}{N} \sum_{n\in\mathcal{N}} w_{n,t+1}^{\tau}$. ϵ is the target model accuracy.

III. Energy Consumption Analysis

In this paper, we investigate two algorithmic implementation schemes of FEI: FedAvg and FedMeta. Before we present the detailed energy consumption models, let us first briefly summarize the training procedures of these two schemes.

A. FedAvg and FedMeta Schemes

The detailed training scheme for FedAvg is shown in algorithm 1. Each edge server minimizes the following objective function in each local update:

$$\textbf{FedAvg:} \quad \mathcal{R}_n(w_t) := f_n(w_t, \mathcal{S}_n) \tag{6}$$

where $f_n(w, \mathcal{S}_n) = \frac{1}{|\mathcal{S}_n|} \sum_{x\in\mathcal{S}_n} l_n(w;x)$ represents empirical loss according to the data samples of edge server n and the loss function $l_n(w;x)$ reflects the error of a model learned from a data point x parameterized by w for edge server n.

The detailed training scheme for FedMeta is shown in algorithm 2. Each edge server minimizes the following objective function in each local update:

$$\textbf{FedMeta:} \mathcal{R}_n(w_t) = f_n\left(w_t - \alpha\nabla f_n\left(w_t, \mathcal{S}_n^{in}\right), \mathcal{S}_n^{out}\right) \tag{7}$$

where α is the step size, $\mathcal{S}_n^{in}$ and $\mathcal{S}_n^{out}$ are two disjoint sets of size K for the inner and outer layer updates, respectively. These local updates will generate a local sequence $\left\{w_{n,t}^{\tau}\right\}_{\tau=0}^{e}$ where $w_{n,t+1}^{0} = w_t$ and, for $1 \le \tau \le e$,

$$w_{n,t+1}^{\tau} = w_{n,t+1}^{\tau-1} - \beta\nabla\mathcal{R}_n\left(w_{n,t+1}^{\tau-1}\right) \tag{8}$$

where β is the local learning rate.

B. Energy Consumption for FedAvg

For FedAvg, we adopt the convergence in [6].

Proposition 1: Suppose the following assumptions hold for arbitrary n: (1) the loss function $l_n(w;x)$ is L-smooth, i.e., $l_n(u) - l_n(v) + (u-v)^T \nabla l_n(v) \le \frac{L}{2}\|u-v\|^2$; (3) the gradient norm is bounded by B, i.e., $\|\nabla l_n(u)\| \le B$; (4) the gradient $\nabla l_n(w)$, i.e., $\mathbb{E}\left(\|\nabla l_n(w,x) - \nabla l_n(w)\|^2\right) \le \sigma_G^2$. By setting the learning rate $\beta \in \left(0, \frac{1}{L}\right]$, we have

$$\frac{1}{eT}\sum_{t=0}^{T-1}\sum_{\tau=0}^{e-1}\mathbb{E}\left[\|\nabla\mathcal{R}\left(\bar{w}_{t+1}^{\tau}\right)\|^2\right] \le \frac{A_1}{T} + \frac{A_2}{K} + A_3 \tag{9}$$

where $A_1 := \frac{2(\mathcal{R}(w_0) - \mathcal{R}(w^*))}{\beta}$, $A_2 := \frac{\beta L\sigma_G^2}{N}$, and $A_3 := 4\beta^2L^2e^2\sigma_G^2$.

Under a given accuracy ϵ, we have

$$\frac{A_1}{T} + \frac{A_2}{K} + A_3 \le \epsilon \tag{10}$$

Then, in order to minimize the energy consumption, we need to minimize the global communication rounds T under the constraint of convergence. By rearranging inequality (10), we can obtain the optimal T^* for FedAvg as follows:

$$T_{Avg}^* = \frac{KA_1}{K\epsilon - A_2 - KA_3} \tag{11}$$

Algorithm 1 FedAvg

Input: Initial iterate w_0

for t: 1 to T **do**
  Coordinator sends w_t to all edge servers
  for all $n \in \mathcal{N}$ **do**
    Set local model parameter $w_{n,t}^0 = w_{t-1}$
    for local step $\tau = 1$ to e **do**
      Set $w_{n,t}^{\tau} = w_{n,t}^{\tau-1} - \beta\nabla f_n(w_{n,t}^{\tau-1}, \mathcal{S}_n)$
    end for
    Edge server *n* sends $w_{n,t}^e$ back to coordinator
  end for
  Coordinator updates its global model by averaging over received models: $w_{t+1} = \frac{1}{N}\sum_{n=1}^{N} w_{t,n}^e$
end for

Algorithm 2 FedMeta

Input: Initial iterate w_0

for t: 1 to T **do**
  Coordinator sends w_t to all edge servers
  for all $n \in \mathcal{N}$ **do**
    Set local model parameter $w_{n,t}^0 = w_{t-1}$
    for local step $\tau = 1$ to e **do**
      Set $\tilde{w}_{n,t}^{\tau} = w_{n,t}^{\tau-1} - \alpha\nabla f_n(w_{n,t}^{\tau-1}, \mathcal{S}_n^{in})$
      Set $w_{n,t}^{\tau} = w_{n,t}^{\tau-1} - \beta\nabla f_n(\tilde{w}_{n,t}^{\tau}, \mathcal{S}_n^{out})$
    end for
    Edge server *n* sends $w_{n,t}^e$ back to coordinator
  end for
  Coordinator updates its global model by averaging over received models: $w_{t+1} = \frac{1}{N}\sum_{n=1}^{N} w_{t,n}^e$
end for

Substituting (11) into problem (5a), we can obtain the optimal overall energy consumption for FedAvg as follows:

$$\mathbb{E}\left[\hat{E}(K)\right] = \frac{KA_1}{K\epsilon - A_2 - KA_3} N(B_0K + B_1) \tag{12}$$

where $N(B_0K + B_1)$ represents the energy consumption of all edge servers in each coordination round, and $B_0 = \mathbb{E}[c_0]\, e$ and $B_1 = \mathbb{E}[\eta]\, K + \mathbb{E}[c_1]\, e + \mathbb{E}[E_n^{mu}]$.

C. Energy Consumption for FedMeta

We do the same investigation for FedMeta, and adopt the convergence result in [7] with slight modification.

Proposition 2: Suppose the following assumptions hold for arbitrary n: (1) the loss function $l_n(w;x)$ is L-smooth, i.e., $l_n(u) - l_n(v) + (u-v)^T\nabla l_n(v) \le \frac{L}{2}\|u - v\|^2$; (2) the Hessian $\nabla^2 l_n$ is ρ-Lipschitz continuous, i.e., $\|\nabla^2 l_n(u) - \nabla^2 l_n(v)\| \le \rho\|u - v\|$; (3) the gradient norm is bounded by B, i.e., $\|\nabla l_n(u)\| \le B$; (4) the gradient $\nabla l_n(w)$ and Hessian $\nabla^2 l_n(w)$ have bounded variance, i.e., $\mathbb{E}\left(\|\nabla l_n(w,x) - \nabla l_n(w)\|^2\right) \le \sigma_G^2$, $\mathbb{E}\left(\|\nabla^2 l_n(w,x) - \nabla^2 l_n(w)\|^2\right) \le \sigma_H^2$. By setting $\alpha \in \left(0, \frac{1}{L}\right]$, $L_F := 4L + \alpha\rho B$, $\sigma_F^2 := \frac{3\alpha^2B^2\sigma_H^2 + 24\sigma_G^2}{K} + \frac{6\alpha^2\sigma_G^2\sigma_H^2}{K^2}$,

$\gamma_F^2 := 12B^2\alpha^2L^2 + 768B^2$, and the learning rate β satisfies $\beta \leq \frac{1}{10eL_F}$, we have

$$\frac{1}{eT}\sum_{t=0}^{T-1}\sum_{\tau=0}^{e-1}\mathbb{E}\left[\|\nabla\mathcal{R}\left(\bar{w}_{t+1}^{\tau}\right)\|^2\right] \leq \frac{C_1}{T} + \frac{C_2}{K} + C_3 \tag{13}$$

where $C_1 := \frac{4(\mathcal{R}(w_0)-\mathcal{R}(w^*))}{e\beta}$, $C_2 := \beta L_F\sigma_F^2 s + \alpha^2L^2\sigma_G^2$, and $C_3 := (\beta^2L_F^2\sigma_F^2 + \beta^2L_F^2\gamma_F^2)(e^2-e)$.

Under a given accuracy ϵ, we have

$$\frac{C_1}{T} + \frac{C_2}{K} + C_3 \leq \epsilon \tag{14}$$

Same as FedAvg, by rearranging inequality (14), we can obtain the optimal T^* for FedMeta as follows:

$$T_{Meta}^* = \frac{KC_1}{K\epsilon - C_2 - KC_3} \tag{15}$$

Substituting (15) into problem (5a), we can obtain the optimal overall energy consumption for FedMeta as follows:

$$\mathbb{E}\left[\hat{E}\left(K\right)\right] = \frac{KC_1}{K\epsilon - C_2 - KC_3}N\left(B_0'K + B_1'\right) \tag{16}$$

where $N\left(B_0'K + B_1'\right)$ represents the energy consumption of all edge servers in each coordination round, and $B_0' = 2\mathbb{E}\left[c_0\right]e$ and $B_1' = \mathbb{E}\left[\eta\right]K + \mathbb{E}\left[c_1\right]e + \mathbb{E}\left[E_n^{mu}\right]$.

Remark 1: Comparing the energy consumption of all edge servers in each coordination round, we can observe that the energy consumption of FedMeta in each coordination round is much greater than that of FedAvg. This is due to the fact that FedMeta requires twice as much data as FedAvg to achieve the same number of local update rounds. This results in more energy consumption for FedMeta to train local model during each global communication round.

Remark 2: We set the same hyper-parameters for FedAvg and FedMeta, and compare their convergence rates. In particular, we set the number of local updates as $e = \mathcal{O}\left(\epsilon^{-\frac{1}{2}}\right)$, the number of samples as $K = \mathcal{O}\left(\epsilon^{-1}\right)$, and stepsize as $\beta = \mathcal{O}\left(\epsilon\right)$. To achieve an O($\epsilon$)-first-order stationary point of $\mathcal{R}$, the FedAvg requires $T = \mathcal{O}\left(\epsilon^{-4}\right)$ rounds of communication between edge servers and the coordinator while the FedMeta requires $T = \mathcal{O}\left(\epsilon^{-2}\right)$ rounds. This result shows that FedMeta requires fewer communication rounds than FedAvg to achieve a satisfactory model.

IV. Experimental Results

To assess the energy consumption of the edge network, we develop a hardware prototype and conduct extensive experiments on the simulation platform. The overall energy consumption of the edge intelligence network using FedAvg and FedMeta algorithm are recorded and analyzed, respectively.

A. Experimental Setup

1) Hardware Prototype: As shown in Fig.2, we set a hardware prototype with 21 Raspberry Pis 4B mini-computers as edge networks. We connect a USB multi-meter to the power port of one server and set the power sample rate to 1 kHz.

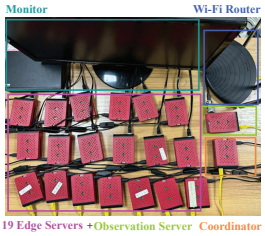
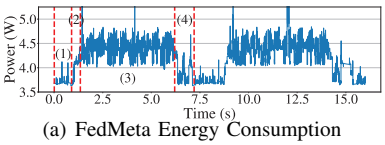
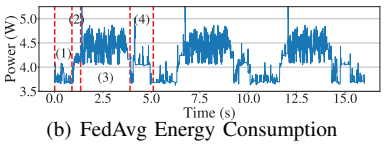


Fig. 2: Hardware prototype with the power measurement device (POWER-Z KM001C multi-function USB multi-meter), 20 Raspberry Pi as edge servers, another as coordinator, and the TP-Link Wi-Fi Router.

2) Experimental Setup: For each of the 20 edge servers, MNIST data is evenly split across 20 edge servers, each with two labels, and a two-layer CNN with ReLU is used for training. In FedMeta, we use SGD with learning rates of 0.005 for the inner layer and 0.004 for the outer layer. For comparison, in FedAvg, we set the learning rate as 0.005.



(a) FedMeta Energy Consumption



(b) FedAvg Energy Consumption

Fig. 3: The energy consumption of different algorithms in a single edge server over the same duration was captured. FedMeta completed three cycles and FedAvg completed two. Each cycle can be distinctly divided into four sections: (1) Waiting, (2) Model Downloading, (3) Training and (4) Model Uploading.

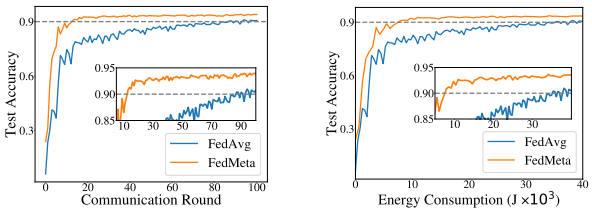
B. Experimental Results

We run FedAvg and FedMeta on our platform for 100 communication rounds, and record the energy consumption for each edge server during all training procedures. Observations reveal the following four stages:

(1) **Waiting:** Before one local training round begins, there is a waiting period for each edge server, attribute to the different computational power and communication efficiencies with the coordinator among the edge servers. This waiting period allows time for the coordinator to collect and integrate the model parameters uploaded by all edge servers before redistributing them. This process matches step (1) in Fig. 3. Both algorithms have a similar power consumption, averaging 3.7W, which results in an energy consumption of 3.33J.

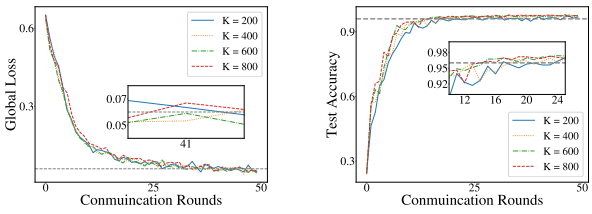
(2) **Model Downloading:** Once the coordinator finishes aggregating the model, each edge server starts downloading the model parameters, as shown in Step (2) Fig.3. The two algorithms on average consume 4.1W, resulting in 1.845J of energy per download phase.

(3) **Training:** As shown in Step (3) of Fig.3, after each edge server successfully receives the model parameters, training starts. In FedMeta, $2K$ data samples from the collection phase undergo mini-batch training with e gradient descent



(a) (b)

Fig. 4: Comparison of the convergence performance between FedMeta and FedAvg during the training process. (a) and (b) respectively show the comparison results of the accuracy of the two algorithms under the same number of communication rounds and the same overall energy consumption.



(a) Loss in various K (b) Accuracy in various K

Fig. 5: Training performance of global accuracy and loss using CNN and the MNIST dataset under various K.

steps, while FedAvg uses e steps on K samples. Notably, the training duration for FedMeta is about twice as long as for FedAvg. During this phase, the average power consumption for both algorithms is 4.4W, leading to energy consumption of 21.34J for FedMeta and 11.22J for FedAvg.

(4) **Model Uploading:** Each edge server uploads its model parameters to the coordinator, a process corresponding to Step (4) in Fig.3. Similar to Step (2), the power consumption for both algorithms is nearly equivalent, averaging 3.9W. Consequently, this equates to an energy consumption of 3.9J for Step (4).

We present a comparison of the convergence rates of FedAvg and FedMeta under the same number of rounds in Fig.4. At a target accuracy of 0.9, FedAvg requires 89 rounds and 36,125J of energy, while FedMeta needs only 13 rounds and 7,970J, just 22.1% of FedAvg consumption. These results demonstrate that although FedMeta has a higher energy consumption per training round, its rapid convergence feature significantly reduces the overall energy consumption.

In Fig.5, we illustrate the global loss and accuracy corresponding to each communication round under various K. Fig.5(a) demonstrates that global loss decreases and then increases with larger K values. Fig. 5(b) shows that accuracy improves notably up to 0.96 but plateaus beyond this point. Increasing K is beneficial for quicker high accuracy, supporting our theory that a specific minimum K optimizes energy efficiency for desired accuracy.

In Fig.6, we show the gap between theoretical predictions from (16), derived via least squares from experimental data, and actual simulations, highlighting both theoretical and real optimal outcomes. The gap between sampled data and actual

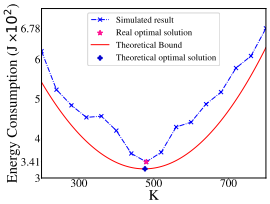


Fig. 6: Comparison of the theoretical bounds with the actual measured trajectories. The pink stars represent the optimal values on the simulation results, while the blue crosses denote the theoretical optimal values K^*.

distribution, though consistent in trend, suggests room for optimization. Furthermore, optimizing K as demonstrated in Fig.6 cuts energy use by nearly 49.7%.

V. Conclusion

This paper studied the overall energy consumption of FEI networks. We established a simple and comprehensive energy consumption model for FEI networks. Based on our proposed framework, we derived analytical solutions of overall energy consumption for two popular algorithms, FedAvg and FedMeta. Subsequently, we compared the overall energy required for both algorithms in communication and computation. Furthermore, we developed a hardware prototype and conducted extensive measurements for the energy consumption under different setups. The overall energy consumption of FedMeta is 77.9% less than that of FedAvg.

Acknowledgment

Y. Xiao and Y. Li were supported in part by the Major Key Project of Peng Cheng Laboratory under Grant PCL2023AS1-2. Yong Xiao was also supported in part by in part by the National Natural Science Foundation of China (NSFC) under Grant 62071193. Y. Li was also supported in part by the NSFC under Grant 62301516, in part by the Key Research and Development Program of Hubei Province under Grant 2023EHA009.

References

[1] A. S. Andrae and T. Edler, "On global electricity usage of communication technology: trends to 2030," *Challenges*, vol. 6, pp. 117–157, Apr. 2015.

[2] P. Zhang, Y. Xiao, Y. Li, X. Ge, G. Shi, and Y. Yang, "Towards net-zero carbon emissions in network AI for 6G and beyond," *IEEE Communications Magazine*, pp. 1–7, Sep. 2023.

[3] Y. Xiao, G. Shi, Y. Li, W. Saad, and H. V. Poor, "Toward self-learning edge intelligence in 6G," *IEEE Communications Magazine*, vol. 58, no. 12, pp. 34–40, Dec. 2020.

[4] C.-J. Wu, R. Raghavendra, U. Gupta, B. Acun *et al.*, "Sustainable AI: Environmental implications, challenges and opportunities," in *MLSys*, Santa Clara, USA, Aug. 2022.

[5] Y. Xiao, Y. Li, G. Shi, and H. V. Poor, "Optimizing resource-efficiency for federated edge intelligence in iot networks," in *WCSP*, Nanjing, China, Oct. 2020.

[6] H. Yu, S. Yang, and S. Zhu, "Parallel restarted SGD with faster convergence and less communication: Demystifying why model averaging works for deep learning," in *AAAI*, Hawaii, USA, Jan. 2019.

[7] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," in *NeurIPS*, Virtual, Dec. 2020.