<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper presented at *8th International Conference on Future Internet of Things and Cloud, FiCloud 2021, Virtual, Online, 23 August 2021 through 25 August 2021*.

Citation for the original published paper:

Al-Saedi, A A., Casalicchio, E., Boeva, V. (2021)
An Energy-aware Multi-Criteria Federated Learning Model for Edge Computing
In: Younas M., Awan I., Unal P. (ed.), *Proceedings - 2021 International Conference on Future Internet of Things and Cloud, FiCloud 2021* (pp. 134-143). IEEE
<https://doi.org/10.1109/FiCloud49777.2021.00027>

N.B. When citing this work, cite the original published paper.

© 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:bth-22236>

An Energy-aware Multi-Criteria Federated Learning Model for Edge Computing

Ahmed A. Al-Saedi
Department of Computer Science
Blekinge Institute of Technology
Karlskrona, Sweden
Email: ahmed.a.al-saedi@bth.se

Emiliano Casalicchio
Department of Computer Science
Sapienza University of Rome, Italy &
Blekinge Institute of Technology, Sweden
Email: emiliano.casalicchio@uniroma1.it

Veselka Boeva
Department of Computer Science
Blekinge Institute of Technology
Karlskrona, Sweden
Email: veselka.boeva@bth.se

Abstract—The successful convergence of Internet of Things (IoT) technology and distributed machine learning have leveraged to realise the concept of Federated Learning (FL) with the collaborative efforts of a large number of low-powered and small-sized edge nodes. In Wireless Networks (WN), an energy-efficient transmission is a fundamental challenge since the energy resource of edge nodes is restricted. In this paper, we propose an Energy-aware Multi-Criteria Federated Learning (EaMC-FL) model for edge computing. The proposed model enables to collaboratively train a shared global model by aggregating locally trained models in selected representative edge nodes (workers). The involved workers are initially partitioned into a number of clusters with respect to the similarity of their local model parameters. At each training round a small set of representative workers is selected on the basis of multi-criteria evaluation that scores each node representativeness (importance) by taking into account the trade-off among the node local model performance, consumed energy and battery lifetime. We have demonstrated through experimental results the proposed EaMC-FL model is capable of reducing the energy consumed by the edge nodes by lowering the transmitted data.

I. INTRODUCTION

The development of Machine Learning (ML) techniques has shown great success in many tasks and applications, especially when powered by other technologies like IoT [1]. Unfortunately, training ML models in central cloud produces a high communication cost, which is not suitable for many resource-constrained IoT devices, and introduce privacy concerns as well. A distributed ML framework has been proposed by Google to move beyond the cloud-centric ML, which is called Federated Learning (FL), provides collaborative learning of a shared ML model, with the help of a large number of edge devices (workers hereafter). Meanwhile, Edge computing is a natural computational paradigm to run FL algorithms, specifically when used to deploy IoT applications. In FL context, edge devices such as smart phones, tablets and sensors' boards download a shared ML model, improve it by learning using their local data, and then upload only their updated models through the base station (BS) to the cloud for global model aggregation [2], [3]. Image classification in Vehicular Edge Computing [4], End-to-End Autonomous Driving [5], the Virtual Keyboard [6], out-of-vocabulary word learning [7] and Human Activity Recognition (HAR) [8] are examples of edge computing applications that use FL

to improve model accuracy and reduce latency in decision making. Furthermore, Artificial intelligence (AI) and FL are also proposed as innovative approaches for edge computing platform management tasks like privacy-preserving optimized service placement at the edge [9], or to optimize caching and communication in edge computing [10], [11]. In practice, to employ FL over IoT networks, edge nodes must repeatedly transmit their local FL models to a BS via wireless links [12]. Due to the constrained resources of IoT devices compared to resources in the cloud, reduction of the energy consumed by edge nodes while preserving the quality of the decision making is one of the main challenges in AI-based edge computing applications (e.g., [13]). Let us consider the edge computing scenario represented in Figure 1 with heterogeneous edge nodes connected to a cloud datacenter by means of (BSs). Each edge node has its own physical characteristics in terms of networking capabilities and energy consumption. We assume that the edge nodes run an application that leverages FL model to make decisions (e.g. image classification or HAR using a large amount of sensor data).

Motivated by the aforementioned challenges, we propose a FL model, entitled Energy-aware Multi-Criteria Federated Learning (EaMC-FL), that reduces the energy consumed by the edge node and transmitted data and additionally tries to

Fig. 1. The Edge computing scenario considered. Edge node with different colors have different hardware features/capabilities

improve the quality of the local ML model. The proposed FL model updates the global model only with local model parameters from few edge nodes that are considered to be representative. Such nodes are selected at each training round by identifying a trade-off between the quality of the local model produced and the physical characteristics of the node. The proposed EaMC-FL algorithm can be considered as an energy-aware improvement of the FL algorithm proposed in [2], FL-baseline hereafter. Hence, the EaMC-FL algorithm is benchmarked against the FL-baseline on two publicly available physical activity monitoring datasets under different experimental scenarios. These datasets have been selected since human activity monitoring and recognition is one of the IoT application field that benefits of federated learning algorithms that reduce energy consumption at the edge. Experimental results demonstrate that our model is capable of reducing the consumed energy by a factor between 5 and 12 with respect the FL-baseline model without compromising the quality of the built ML model.

The paper is organized as follows. Related works are analyzed in Section II. Backgrounds on Federated Learning and edge node (worker) selection policy are presented in Section III. The problem addressed is stated in Section IV along with the description of the energy model adopted. The proposed EaMC-FL approach is described in Section V. Finally, the experimental settings and the obtained evaluation results are presented and analysed in Section VI.

II. RELATED WORK

The communication overhead of FL mostly arises from FL's iterative nature and from the global model aggregation, which can be alleviated by effective resource allocation [14]. Recent research works address the FL challenges of reducing communication overhead (e.g., [13], [15], [16], [17], [18], [19]) and energy consumption (e.g., [13], [16], [20], [21]).

In [15] the authors propose to use analog aggregation to reduce energy consumption, that is the summation of local models can be carried out over-the-air if workers synchronize with each other and align the transmit power. That solution requires to properly schedule workers and to leverage network channel transmission properties. The authors do not compare their results against other energy aware solutions neither against the FL-baseline model [2]. A different approach to minimize energy consumption is proposed in [16]. In that work the authors have formulated an optimization problem whose goal is to minimize the total energy consumption of the system under a latency constraint. Results of the proposed solution are compared with a traditional federated learning schema over wireless channels. CMFL [17] is a FL model that let clients to send updates only if relevant enough to model improvement. That reduce the network footprint. The obtained results are compared with the FL-baseline. Gaia [18] is a geo-distributed ML system that employs an intelligent communication mechanism to efficiently utilize the WAN bandwidth, while retaining the accuracy and correctness guarantees of an ML algorithm. Error accumulation method is

considered to decrease the communication cost in [19]. A neural-structure-aware resource management approach with module-based federated learning is proposed in [13]. That solution assigns mobile clients with different subnetworks of the global model according to the status of their local resources with the goal of using more efficiently client resources, included network bandwidth. The proposed solution is compared against the traditional FL approach. A new approach for energy-efficient radio resource management (RRM) for federated learning is proposed in [20]. RMM performs energy-efficient bandwidth allocation and scheduling while preserving learning performance. RMM is benchmarked against the FL-baseline algorithm. HybridFL [21] adopts model aggregation at edge level and cloud level. Their client selection strategy for model update improves the FL training process significantly in terms of shortening the federated round length, speeding up the global model's convergence and reducing end device energy consumption.

Similarly to the solutions proposed in [13], [17], [21], we achieve energy efficiency by proposing an new edge node selection policy that splits edge nodes in optimal clusters based on local model similarity and that selects few representatives of the clusters trading-off model accuracy, battery lifetime and energy consumption. The selection of representative is heuristic, that has the advantage of not requiring the solution of complex optimization problems. The drawback is the achievement of a sub-optimal energy efficiency, that however is enough to outperform the FL-baseline model.

III. BACKGROUND

In this section, we first introduce the FL-baseline model [2]. We then provide a definition of k -medoids clustering algorithm used for initial grouping of the workers. Finally, we explain the Silhouette Index validation measure that is applied at each round for updating the worker clustering by assessing whether the representatives are still well tied to their current clusters.

A. Federated learning

As it has already been mentioned, the proposed EaMC-FL algorithm can be considered as an energy-aware improvement of the federated learning model introduced in [2]. Hence the latter FL-baseline algorithm (also abbreviated as FedAvg) is used as benchmark for EaMC-FL.

In a FL scenario, the learning task is conducted by having a coordinator (also referred as server and usually running in a cloud datacenter) that, throughout a set of iterations, collaborates with the participating workers to learn and build a shared ML model [22].

In detail, a model M is learned iteratively by using a randomly selected subset of workers $V_s \subset V$. The workers in V_s participate at each round and compute the gradient of the loss over the data held by them. Each worker $v \in V_s$ at round t has its own data D_v^t and a local model m_v^t . At each round t , each worker trains its local model by iterating the local update multiple times of Stochastic Gradient Descent (SGD) before sending the next local model m_v^{t+1} to the server. The

server, after aggregating the local models computed at round t , performs a synchronous update of the global model M_{t+1} . The global model update can be computed using different criteria. In this paper, we assume it is evaluated by means of federated averaging, that is the local model $m_v^t, v \in V_s$ and global model M_t are updated by the following equations: [23]:

$$m_v^{t+1} = m_v^t - \eta g_v^t \quad (1)$$

$$M_{t+1} = \sum_{v \in V_s} \left(\frac{n_v}{n} m_v^{t+1} \right), \quad (2)$$

where m_v^{t+1} is the local update, g_v^t are the updated weights on its local data at the current model m_v^t , M_{t+1} is the next global model, η is a learning rate computed by each worker, n is the sum of all data points and n_v is the number of local data points. The server then distributes the global model M_{t+1} to the workers that can perform another iteration of local training and model update.

B. K-medoids clustering

K-medoids clustering [24] allows to split the available workers (edge nodes) into groups of similar workers with respect to their local updates. The medoid is the most centrally located point in a given cluster. Due to this in the EaMC-FL algorithm, the medoid of each cluster can be used as the representative of the cluster.

A challenge in the partitioning algorithms like k -medoids is the identification of the number of clusters k in advance. One solution that may apply in such cases is to build a clustering model with a range of values for k and then evaluate the quality of the generated clustering partitions. For example, different internal cluster validation indices [25] (such as Silhouette Index introduced below) can be applied to recognize the optimal number of clusters.

C. Silhouette Index

The proposed EaMC-FL algorithm uses Silhouette Index (SI) [26] at each iteration round for assessing whether the worker representatives are still well tied to their current clusters. The Silhouette Index is a well-known and widely used internal cluster validation technique that can be applied for assessing compactness and separation properties of a given partitioning [26]. Suppose a_i represents the average distance of object i from all the other objects in the cluster to which the object i is assigned, and b_i represents the minimum of the average distances of object i from objects of the other clusters. Then the *Silhouette Index* $s(i)$ of object i can be calculated as

$$s(i) = (b_i - a_i) / \max\{a_i, b_i\}. \quad (3)$$

$s(i)$ measures how well object i matches the clustering at hand. $s(i) \in [-1, 1]$ and higher value points out a better quality of the clustering results. For example, when $s(i)$ is close to 1 this means that object i is assigned to a very appropriate cluster. A situation is different when $s(i)$ is about zero. Namely, object i lies between two clusters. The worst case is when $s(i)$ is close to -1 . Evidently, this object has been misclassified.

IV. PROBLEM STATEMENT

As mentioned earlier, in a FL setting when workers are edge nodes with limited battery lifetime, a reduction of the amount of transferred data to the master node in the cloud will decrease the energy consumed. Indeed, as outlined by [27] data transfer over radio channels is the process with higher energy consumption impact.

In this paper, we address the problem of reducing the quantity of data transferred by the workers to the master node and vice-versa trading-off the worker local model performance, the energy consumed and the battery lifetime.

A. Energy model

For an edge node i , acting as a worker in a FL scenario, the power required to transmit forth and back the model parameters is $P_i^t = P_i^s + P_i^r$ Watt (W), where superscript t , s and r stand for transmit, send and receive, respectively. The energy E_i^t consumed by the edge node i in a time interval t is the power used in t , hence

$$E_i^t = E_i^s + E_i^r = P_i^s \times t^s + P_i^r \times t^r, \quad (4)$$

where t^s and t^r are respectively the time needed to transmit the data and to receive the data, and E_i^s (E_i^r) is the energy consumed for sending (receiving) data. The battery capacity, or energy budget, of an edge node is B_i Joule (J). Then, the battery lifetime (in seconds) is

$$L_i = B_i / P_i^t. \quad (5)$$

We recall that $1W = 1J/sec$. Each time an edge device sends/receives data for a certain amount of time t^s (t^r) the battery is drained of E_i^s (E_i^r) Joule. At one point in time T it is useful to express the available energy budget (or battery lifetime) as a percentage

$$L_i^a(T) = \frac{1}{B_i} \left(B_i - \sum_{\tau=0}^T \left(\dot{E}_i^t(\tau) \right) \right) \quad (6)$$

assuming $\sum_{\tau=0}^T E_i^t(\tau)$ is always less or equal than B_i . For example, if $B_i = 100J$, $P_i^s = P_i^r = 335.5\mu W$ then $L_i = 0.13 \times 10^6 sec$ that is approximately 36 hours. If the edge node at time $t = 0$ starts transmitting for 50 seconds and receiving for other 50 seconds, the energy drained at time $T \geq 100$ sec is $33550\mu J$ and $L_i^a(T) = 0.99\%$. Equation 6 allows to decide whether a node is suitable to transmit the model parameters. Indeed if $\sum_{\tau=0}^{T'} E_i^t(\tau) \geq B_i$ the node cannot be selected to send (receive) data at time T' .

The final step is to define E_i^t as a function of the size of the model parameters transmitted, and of the edge node characteristics like network bandwidth and latency. Let assume the size of the model parameters is s bytes, the network bandwidth for node i is b_i bytes per second, and the round trip network latency experienced in the communication with the master node is l_i seconds, the transmission time is

$$\tau_i = 2 \times \frac{s}{b_i} + l_i. \quad (7)$$

Assuming that $P_i^s = P_i^r = P_i$, Eq. 4 can be expressed as:

$$E_i^t = P_i \times \left(2 \times \frac{s}{b_i} + l_i \right) \quad (8)$$

In summary, Eq. 6, Eq. 8 and F-measure (see Section VI-B) are evaluation criteria that can be used to select an appropriate node for transmitting model parameters to the master node.

V. MULTI-CRITERIA FEDERATED LEARNING

A. Evaluation Criteria and Selection Policy

As introduced above, an edge node i is described by E_i^t , $L_i^a(T)$ (or L_i), along with a metric I_i that evaluates the model performance, e.g. F-measure can be used for this purpose. This three criteria are used to select an edge node as worker representative for a cluster of workers. Considering that for a node i the transmission power P_i , the size s of the model and the bandwidth b_i are constant where the latency l_i slightly varies from node to node in a given time frame, since there can be several factors such as edge node characteristics that can lead to a variation in the latency value in a realistic environment, the worker i score can be calculated as follows:

$$S_i = w_1 \times I_i + w_2 \times L_i^a(T) + w_3 \times \frac{E_i^* - E_i^t}{E_i^*}, \quad (9)$$

where $w_j \in [0, 1]$, $j \in \{1, 2, 3\}$ and $\sum_{j=1}^3 w_j = 1$, are relative weights that define the importance of the evaluation criteria. I_i and $L_i^a(T)$ are adimensional and vary in the range $[0, 1]$. Note that higher values mean a better quality of the model and more battery budget available to transmit model parameters to the master coordinator. The third term is an adimensional and normalization (in $[0, 1]$) factor accounting for the energy consumed to transfer the model parameters to the master coordinator in the cloud. Higher is its value lower is the energy consumed: $E_i^* = P_i(2(s/b_i)+1)$ is an upper bound of the energy consumed by the edge node i when the round trip network latency is equal to 1 second, that is the worst case, because the round trip latency between the edge and the cloud is usually of the order of few hundred of milliseconds [28] and values of the order of seconds means the master coordinator is not reachable (we assume it is always reachable). Hence S_i is an adimensional number in the range $[0, 1]$ used as an evaluation criteria for the workers' selection. The selection policy is assumed to be implemented in a cloud, i.e. a central node computes the S_i scores.

Based on the above assumptions, the **cluster representative selection policy** at round t is described in what follows. Let $C^t = \{C_1^t, C_2^t, \dots, C_k^t\}$ is the clustering at round t , k is the number of clusters and $\{v_{j1}^t, v_{j2}^t, \dots, v_{jk_j}^t\}$ is the set of edge nodes belonging to cluster C_j^t , $j = 1, 2, \dots, k$. Then

- 1) For each $C_j^t \in C^t$
 - compute $\bar{C}_j^t = C_j^t \setminus \{v_{ji}^t \mid l_{ji}^t \geq 1\}$
 - for each $v_{ji}^t \in \bar{C}_j^t$ compute the edge node scores S_i by applying Eq. 9
- 2) For each $\bar{C}_j^t$ select the highest scored edge node v_{ji}^t for which $L_{ji}^a(T) > 0$ (to be conservative could be selected a node with $L_{ji}^a(T) > X\%$).

B. Energy-aware multi-criteria federated learning algorithm

Let us assume N edge nodes are involved in training of a FL model. Initially, the three evaluation criteria defined in Section V-A (see also Section VI-B) are analyzed and relatively weighted with respect to the studied applied scenario, i.e. weights w_1, w_2 and w_3 are defined. We denote with $V = \{v_1, v_2, \dots, v_N\}$ the set of available workers. A set V^t ($t = 0$ and $|V^t| < N$) of randomly selected workers is initially used to build the initial global model M_t . The local updates received at the initialization round $\{m_v^t \mid v \in V^t\}$ are analyzed and partitioned into k groups of similar updates by applying k -medoids.

The basic operations conducted by our EaMC-FL algorithm at each round t ($t > 0$) are explained below:

- 1) **Multi-criteria evaluation of workers:** Each worker $v \in V^{t-1}$ is evaluated under the predefined criteria by calculating its edge score S_v (see Eq. 9);
- 2) **Selection of representative workers:** The workers in each cluster C_i^{t-1} , $i = 1, 2, \dots, k$ are ranked in descent order with respect to the calculated edge scores. Using the selection policy defined in Section V-A the admissible top-ranked worker is selected as a representative of each cluster C_i^{t-1} , $i = 1, 2, \dots, k$. In that way, the selected representatives form a new set of workers $V^t = \{v_1^t, v_2^t, \dots, v_k^t\}$, where $k < |V^0|$.
- 3) **Updating of local models** Each worker $v \in V^t$ receives the global model M_{t-1} and optimizes its parameters locally (Eq. 1), i.e. m_v^t update is produced and sent back to the server;
- 4) **Building of global model:** M_t is built by aggregating the local updates $\{m_v^t \mid v \in V^t\}$ received at round t (Eq. 2);
- 5) **Updating of clustering model:** C^{t-1} is updated by evaluating the local update of each $v \in V^t$ and whether it is still tied to its current cluster or must be assigned to an other (Eq. 3). As a result a new updated clustering C^t is produced.

Then steps 1 – 5 are repeated until reaching either convergence or the maximum number of rounds (T). The proposed EaMC-FL algorithm is illustrated in Figure 2.

VI. EXPERIMENTAL EVALUATION

A. Data

For initial evaluation of the proposed EaMC-FL algorithm, we have used publicly available real-world datasets from UCI Machine Learning repository. We use mHealth [29] and Pamap2 [30] physical activity monitoring datasets. The both datasets involve motion sensor data of several physical activities. These provide an opportunity to simulate a multi-class classification problem of HAR, which is a suitable machine learning task for studying FL scenarios.

The mHealth dataset has 215040 instances distributed in 7 classes, while Pamap2 dataset has 16819 instances that contain information about 18 different physical activities.

Fig. 2. A schematic illustration of EaMC-FL algorithm. A global model is trained on each representative's data and averaged at the server of each iteration phase. Circled numbers correspond to the basic operations.

B. Evaluation metrics

The evaluation metrics that are used to score the importance of each node and then select a representative of each group of workers are F1-score, consumed energy, and energy budget. The evaluation measures used for the definition and evaluation of the EaMC-FL algorithm are explained hereafter.

- **Sum of energy consumed per round:** We calculate the aggregation of energy consumed by all workers per round. In order to measure the energy consumption in the FL-baseline we sum the energy consumed of all workers that have participated in training of the shared model at each round, while in EaMC-FL, the sum of energy consumed by all representative workers is calculated.
- **Average of energy budget per round:** The energy budget or battery lifetime in the FL-baseline is calculated by the average energy budget of all workers in each round until a global model is converged. Similarly, the energy budget in EaMC-FL is calculated by averaging the energy budget of all representative workers in each round.
- **F1-score:** F-measure is the harmonic mean of precision and recall values for each class. The scikit-learn of the F-measure (*micro-average F₁*) has been used to evaluate the accuracy of EaMC-FL and FedAvg models.
- **Number of rounds to converge:** In FL context, the training process consists of iterative rounds. It is repeated until a stopping criteria is met. In our experimental scenarios, the training process stops when the difference between the accuracy scores produced by two consecutive training rounds is less than 0.001 or when number of iterations exceeds 100.
- **Number of representatives per round:** The initial number of representatives (edge nodes) is equal to the optimal number of clusters identified for each dataset. At each training round this number can change due to the clustering update.

C. Experimental setting

In order to simulate a distributed scenario each dataset is used to generate 10 experimental datasets by randomly separating the data points into a number of groups (workers). Each group is supposed to represent data supplied by one worker. In addition, we have studied two different experimental data distribution scenarios: IID (Independent and Identically Distributed) and non-IID. In that way for each studied number of workers (20, 30, 40, 50 and 60) and each used dataset (mHealth and Pamap2) we have built 20 experimental datasets, i.e. 200 experimental datasets in total.

We benchmark the EaMC-FL algorithm against the FL-baseline [2] on the built experimental datasets under six different use cases (see Table I). Each use case presents a different multi-criteria evaluation scenario. For example, in the first use case the three evaluation criteria (model accuracy, energy consumption and battery lifetime) are equally important. In the second use case, the battery lifetime is prioritized ($w_2 = 0.5$), i.e. we care more about the consumed energy and battery lifetime than accuracy. While the accuracy is more important in use case 3 ($w_1 = 0.5$). Use cases 4, 5 and 6 present more extreme scenarios in which one of the three evaluation criteria is neglected, i.e. the corresponding weight is zero.

TABLE I
MULTI-CRITERIA EVALUATION SCENARIOS

Use case	w_1	w_2	w_3
1	0.33	0.33	0.33
2	0.2	0.5	0.3
3	0.5	0.3	0.2
4	0	0.5	0.5
5	0.9	0.1	0
6	0	0.9	0.1

The FL-baseline and EaMC-FL algorithms are implemented in Python using Scikit-learn library. Stochastic Gradient Descent (SGD) classifier is used as a training model in the two compared FL algorithms. The machine learning task is a multi-class classification problem of human activity recognition. Each worker trains its local model for a number of epochs e on its local dataset using sklearn as the ML library. The updated local model is then sent back to the coordinator, which updates the global model and sends it again to the workers. The process continues and conducts until the model is converged.

In addition, in our experimental scenarios, the size of local models the edge nodes send to the server at each iteration round is constant, 2.447 kB and 6.575 kB for mHealth and Pamap2 datasets, respectively. Furthermore, the energy consumed by the edge node of a model transmission on average is 1.17888 (J), whereby a random value between 43 ms and 54 ms is assigned for the latency per worker during transmission. Finally, we assume the simulated edge devices have homogeneous battery power.

D. Results and analysis

We initially compare FL-baseline (FedAvg) and EaMC-FL algorithms with respect to the average energy budget under

Fig. 3. Average energy with the total number of rounds to achieve the convergence on Pamap2 dataset.

the six studied evaluation scenarios (see Table I) by applying the algorithms on the experimental datasets of Pamap2 [30] and mHealth [29] for 20 numbers of workers under two data distribution scenarios: IID and non-IID. The obtained results are plotted in Figures 3 and 4, respectively. One can notice that for both datasets EaMC-FL outperforms FedAvg in all studied experimental scenarios. It is interesting to observe that EaMC-FL demonstrates different behaviour with respect to the two data distribution scenarios (IID versus non-IID) in two datasets. Namely, EaMC-FL IID has generated higher values than EaMC-FL Non IID in 2/3 of the use cases of Pamap2 dataset (see Figure 3), while EaMC-FL Non IID outperforms EaMC-FL IID on mHealth dataset again in 2/3 of the use cases (see Figure 4).

We have also compared the FedAvg and EaMC-FL algo-

rithms under the same experimental scenarios as ones studied in Figures 3 and 4, but with respect to the sum of energy consumed of the workers over a number of rounds until the model is converged. The results produced by EaMC-FL for both datasets in all studied experimental scenarios are significantly better than the ones of FedAvg. In addition, in comparison with the FedAvg algorithm the EaMC-FL algorithm in most of the evaluated use cases needs less or at most equal number of rounds for training the shared model. This is expected to reduce further the total energy consumed for the ML model training. Due to the space constraint we present in Figure 5 only the results generated on the Pamap2 experimental datasets. The results obtained on mHealth data are similar.

Table II presents the accuracy scores (Accuracy) of the

Fig. 4. Average energy budget with the total number of rounds to achieve the convergence on mHealth dataset.

FedAvg and EaMC-FL models, the average of energy consumed over the total number of rounds (Energy) and the corresponding number of rounds (Rounds) needed for the model convergence generated in the six evaluation scenarios on Pama2 experimental datasets for IID and Non-IID data distribution settings, respectively. Note that in case of FedAvg, the coordinator randomly chooses 20 workers out of 40 workers for each training round. In both data distribution settings, the EaMC-FL model produces higher accuracy scores on the whole six evaluation scenarios than the FedAvg and in most of the use cases for less number of rounds. It is also interesting to analyse and benchmark the results (in Table II) generated in evaluation scenarios 2, 4 and 6 against those produced by use cases 3 and 5. The former ones assign higher importance to energy consumption and battery lifetime than the latter ones,

which prioritise the model performance. Logically, the highest accuracy scores generated by the EaMC-FL model in both data distribution settings are for use cases 3 and 5, i.e. the ones that assign higher importance to the model accuracy. It is also interesting to notice that in both data distribution settings the lowest amount of consumed energy is in use cases 3 and 4. The former use case balances between the model accuracy and energy consumption and budget while the latter one neglects the model accuracy. Curiously, the smallest number of rounds is needed in two use cases that are again the same for both data distribution scenarios, namely 1 and 6. Remind that in use case 1 the three evaluation criteria are equally important while use case 6 prioritises the battery lifetime and neglects the model accuracy. Note that different trends are observed on mHealth experimental datasets in the different multi-criteria

Fig. 5. Sum of energy consumed with the total number of rounds to achieve the convergence on Pamap2 dataset.

evaluation scenarios (use cases in Table I). Hence this is worth to be further studied maybe in a larger scale real-world data context, which is part of our future work.

Figure 6 shows the average of total energy consumed over the needed training rounds with respect to different number of workers for the Pamap2 dataset. The results produced on the mHealth experimental dataset are similar. One can clearly see that for the FL-baseline algorithm (FedAvg), the total energy consumed increases with the number of workers that participating in federated learning, which is not as much as that

of the EaMC-FL algorithm. This is due to the fact that more energy must be used by the workers for model parameters transmission. In summary, EaMC-FL outperforms FL-baseline in terms of total energy consumption with respect to different number of workers in both data distribution settings.

VII. CONCLUSION AND FUTURE WORK

In this paper, we have investigated the problem of energy consumed and energy budget of edge nodes of FL over wireless networks. The presented work is an approach to

Fig. 6. Average of energy consumed with the total number of rounds to achieve the convergence on Pamap2 dataset.

control and optimize communication of FL in the wireless network. We have proposed an Energy-aware Multi-Criteria Federated Learning (EaMC-FL) algorithm for edge computing. The EaMC-FL algorithm has been benchmarked against the FL-baseline (FedAvg) algorithm on two publicly available physical activity monitoring datasets under different experimental scenarios. The obtained experimental results have shown that EaMC-FL outperforms FedAvg in terms of total energy consumption, energy budget, and model accuracy.

Our future aim is to carry out further evaluation of the proposed EaMC-FL algorithm by studying its potential on much larger scale real-world datasets and other applied FL scenarios. In addition, we plan to compare the proposed EaMC-FL algorithm with some other state-of-the-art energy-aware FL models.

ACKNOWLEDGMENTS

The first author is supported by an Iraq Ministry of Higher Education and Scientific Research PhD Scholarship. The work of E.Casalicchio is partially supported by the SmartDefense Project (Ricerca Ateneo 2019). The work of V. Boeva is partially supported by the MIRAI ITEA3 (19034) 2020 Project.

REFERENCES

- [1] I. J. Goodfellow, Y. Bengio, and A. C. Courville, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
- [2] H. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *AISTATS*, 2017.
- [3] J. Konecný, H. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *CoRR*, vol. 1610.02527, 2016.

TABLE II
EAMC-FL AND FEDAVG ACCURACY SCORES, AVERAGE OF ENERGY
CONSUMED WITH THE TOTAL NUMBER OF ROUNDS AND NUMBER OF
ROUNDS TO ACHIEVE THE CONVERGENCE IN THE SIX EVALUATION
SCENARIOS ON PAMAP2 EXPERIMENTAL DATASETS

IID Setting						
FedAvg				EaMC-FL		
Case	Accuracy	Energy	Rounds	Accuracy	Energy	Rounds
1	0.950	22.633	9	0.965	5.419	9
2	0.949	22.516	11	0.963	5.205	10
3	0.946	22.769	13	0.968	4.413	10
4	0.948	22.737	15	0.964	4.723	14
5	0.952	22.732	11	0.966	4.886	10
6	0.944	22.550	11	0.963	6.375	9

Non-IID Setting						
FedAvg				EaMC-FL		
Case	Accuracy	Energy	Rounds	Accuracy	Energy	Rounds
1	0.940	22.640	9	0.964	5.588	8
2	0.942	22.736	14	0.963	5.307	11
3	0.923	22.724	16	0.967	4.379	12
4	0.941	22.815	16	0.964	4.419	12
5	0.935	22.787	15	0.965	4.974	10
6	0.950	22.574	10	0.963	4.505	8

- [4] D. Ye, R. Yu, M. Pan, and Z. Han, "Federated learning in vehicular edge computing: A selective model aggregation approach," *IEEE Access*, vol. 8, pp. 23 920–23 935, 2020.
- [5] Z. Chen and X. Huang, "End-to-end learning for lane keeping of self-driving cars," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1856–1860.
- [6] A. Hard, K. Rao, R. Mathews, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *CoRR*, vol. abs/1811.03604, 2018.
- [7] M. Chen, R. Mathews, T. Ouyang, and F. Beaufays, "Federated learning of out-of-vocabulary words," *CoRR*, vol. abs/1903.10635, 2019.
- [8] K. Sozinov, V. Vlassov, and S. Girdzijauskas, "Human activity recognition using federated learning," in *2018 IEEE ISPA/UCC/BDCloud/SocialCom/SustainCom*, 2018, pp. 1103–1111.
- [9] Y. Qian, L. Hu, J. Chen, X. Guan, M. M. Hassan, and A. Alelaiwi, "Privacy-aware service placement for mobile edge computing via federated learning," *Information Sciences*, vol. 505, pp. 562–570, 2019.
- [10] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge ai: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Network*, vol. 33, no. 5, pp. 156–165, 2019.
- [11] Z. Yu, J. Hu, G. Min, H. Lu, Z. Zhao, H. Wang, and N. Georgalas, "Federated learning based proactive content caching in edge computing," in *2018 IEEE Global Communications Conference*, 2018, pp. 1–6.
- [12] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," *IEEE Communications Magazine*, vol. 58, pp. 19–25, 2020.
- [13] R. Yu and P. Li, "Toward resource-efficient federated learning in mobile edge computing," *IEEE Network*, vol. 35, no. 1, pp. 148–155, 2021.
- [14] H. H. Yang, Z. Liu, T. Quek, and H. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Transactions on Communications*, vol. 68, pp. 317–333, 2020.
- [15] Y. Sun, S. Zhou, and D. Gündüz, "Energy-aware analog aggregation for federated learning with redundant data," *2020 IEEE ICC*, pp. 1–7.
- [16] Z. Yang, M. Chen, W. Saad, C. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Trans. on Wireless Communications*, vol. 20, pp. 1935–1949, 2021.
- [17] L. Wang, W. Wang, and B. Li, "Cmfl: Mitigating communication overhead for federated learning," *IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 954–964, 2019.
- [18] K. Hsieh and et al., "Gaia: Geo-distributed machine learning approaching LAN speeds," in *14th USENIX Symposium on Networked Systems Design and Implementation*, 2017.
- [19] M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, pp. 3546–3557, 2020.
- [20] Q. Zeng, Y. Du, K. Leung, and K. Huang, "Energy-efficient radio resource allocation for federated edge learning," *2020 IEEE International Conference on Communications Workshops*, pp. 1–6, 2020.
- [21] W. Wu, L. He, W. Lin, and R. Mao, "Accelerating federated learning over reliability-agnostic clients in mobile edge computing systems," *IEEE TPDS*, vol. 32, no. 7, pp. 1539–1551, 2021.
- [22] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *NIPS Workshop on Private Multi-Party ML*, 2016.
- [23] H. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated learning of deep networks using model averaging," *ArXiv*, vol. 1602.05629, 2016.
- [24] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Lucien M. Le Cam and Jerzy Neyman, editors, Proceedings of the Berkley symposium on mathematical statistics and probability*, vol. 1, pp. 281–297, 1967.
- [25] L. Vendramin, R. J. Campello, and E. R. Hruschka, "Relative clustering validity criteria: A comparative overview," *Statistical analysis and data mining: the ASA data science journal*, vol. 3, no. 4, p. 209–235, 2010.
- [26] P. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.
- [27] X. Fafoutis and et al., "Extending the battery lifetime of wearable sensors with embedded machine learning," *2018 IEEE 4th WF-IoT*, pp. 269–274.
- [28] S. Maheshwari, D. Raychaudhuri, I. Seskar, and F. Bronzino, "Scalability and performance evaluation of edge cloud systems for latency constrained applications," *2018 IEEE/ACM SEC*, pp. 286–299.
- [29] O. Baños, R. García, J. A. H. Terriza, M. Damas, H. Pomares, I. Rojas, A. Saez, and C. Villalonga, "mhealthdroid: A novel framework for agile development of mobile health applications," in *IWAAL*, 2014.
- [30] A. Reiss and D. Stricker, "Introducing a new benchmarked dataset for activity monitoring," *2012 16th ISWC*, pp. 108–109.