

RESEARCH ARTICLE

Decentralized Federated Learning Over Slotted ALOHA Wireless Mesh Networking

ABDELAZIZ SALAMA¹, (Member, IEEE), ACHILLEAS STERGIOLIS¹, (Member, IEEE),
ALI M. HAYAJNEH², (Member, IEEE), SYED ALI RAZA ZAIDI¹, (Senior Member, IEEE),
DES MCLERNON¹, (Member, IEEE), AND IAN ROBERTSON¹, (Fellow, IEEE)

¹Department of Electrical and Electronic Engineering, University of Leeds, LS2 9JT Leeds, U.K.

²Department of Electrical Engineering, Faculty of Engineering, The Hashemite University, Zarqa 13133, Jordan

Corresponding author: Abdelaziz Salama (elamasa@leeds.ac.uk)

This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC), U.K., under Grant EP/S016813/1 and Grant EP/N010523/1; in part by the Royal Academy of Engineering, U.K., under Grant 122040; in part by the Transforming Systems through Partnership under Grant TSP1040; and in part by the Royal Academy through the Distinguished International Associates under Grant DIA-2021-18.

ABSTRACT Federated Learning (FL) presents a mechanism to allow decentralized training for machine learning (ML) models inherently enabling privacy preservation. The classical FL is implemented as a client-server system, which is known as Centralised Federated Learning (CFL). There are challenges inherent in CFL since all participants need to interact with a central server resulting in a potential communication bottleneck and a single point of failure. In addition, it is difficult to have a central server in some scenarios due to the implementation cost and complexity. This study aims to use Decentralized Federated learning (DFL) without a central server through one-hop neighbours. Such collaboration depends on the dynamics of communication networks, e.g., the topology of the network, the MAC protocol, and both large-scale and small-scale fading on links. In this paper, we employ stochastic geometry to model these dynamics explicitly, allowing us to quantify the performance of the DFL. The core objective is to achieve better classification without sacrificing privacy while accommodating for networking dynamics. In this paper, we are interested in how such topologies impact the performance of ML when deployed in practice. The proposed system is trained on a well-known MNIST dataset for benchmarking, which contains labelled data samples of 60K images each with a size 28×28 pixels, and 1000 random samples of this MNIST dataset are assigned for each participant's device. The participants' devices implement a CNN model as a classifier model. To evaluate the performance of the model, a number of participants are randomly selected from the network. Due to randomness in the communication process, these participants interact with the random number of nodes in the neighbourhood to exchange model parameters which are subsequently used to update the participants' individual models. These participants connected successfully with a varying number of neighbours to exchange parameters and update their global models. The results show that the classification prediction system was able to achieve higher than 95% accuracy using the three different model optimizers in the training settings (i.e., SGD, ADAM, and RMSprop optimizers). Consequently, the DFL over mesh networking shows more flexibility in IoT systems, which reduces the communication cost and increases the convergence speed which can outperform CFL.

INDEX TERMS Simplicity, privacy, federated learning, decentralization learning.

The associate editor coordinating the review of this manuscript and approving it for publication was Jose Saldana¹.

I. INTRODUCTION

In recent years, the number of Internet of Things (IoT) and smart wearable devices have witnessed an increased proliferation in several vertical domains (e.g., wearables,

home automation systems, smart glasses, health monitors, health-fitness trackers, smart grids, etc). These devices use a variety of wireless technologies and thus manifest different topological properties. For instance, LoRa deployment support one hop connection to the gateway with a distance-dependent spreading factor manifesting star topology. In contrast, Thread, Zigbee, and BLE Mesh manifest the mesh topology. The Mesh deployment for local connectivity is gaining extra momentum with the rapid evolution of standards.

The huge amount of data that sensors and edge devices collect is critical. It can be gathered and analyzed using modern ML techniques to build a classification model that can learn, predict, and meet the end user's requirements optimally. ML algorithms can enable various applications. These include controlling and monitoring the home appliance, controlling autonomous vehicles, and monitoring various health parameters of the elderly such as heart rate, fall detection, etc. However, the data from an edge device may carry very personal information. Thus, data privacy and security are significant challenges as users usually do not allow sharing of sensitive information and data by putting it all in one central location.

Now, Federated Learning (FL) [1] has recently emerged to address this issue and solve participants' essential requirements and concerns to preserve privacy and data security. FL has developed as a new paradigm in building models from distributed ML setups that can offer the opportunity to learn a model from multiple disjoint sensitive local datasets while keeping the user data private through distributed training [1] and [2].

FL has attracted much attention because of its ability to preserve the privacy of the client's data by sharing only the locally trained model parameters instead of the local data itself. Figure 1 shows a reference deployment scenario for a healthcare application. This example shows how the Community Hospital, the Research Medical Centre, and the Cancer Treatment Centre all train their local models using their local private data and share only the model parameters to create a global model in order to improve the system performance.

Thus, FL performs the aggregation and analysis of the local models and updates the models on the participants' devices or servers without sharing any devices' data with others, thus keeping the private data protected. So, each participating device will train a model exploiting its local data usually using classification or regression algorithms (i.e., employing a Deep Neural Network (DNN)), and only share the model parameters with a central server. Afterwards, the FL central server will aggregate the parameters of these local models to process and create a global model. The FL server broadcasts the global model to update the local models. The system will iterate the same procedure until the system achieves a convergence state [3]. This paradigm of FL is also known as Centralized Federated Learning (CFL).

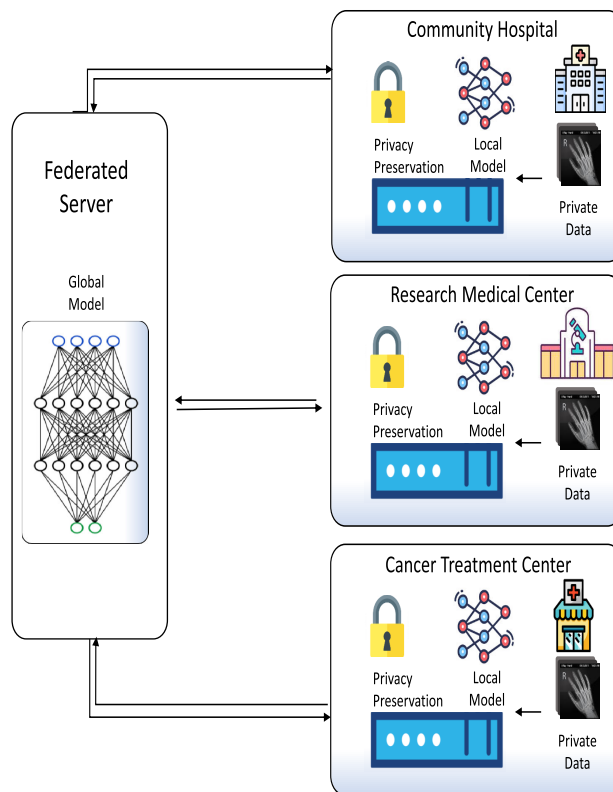


FIGURE 1. A Federated learning approach.

A. MOTIVATION

The data generated by IoT devices and smartphone terminals has become essential for driving intelligent services through applications of Machine Learning (ML). Various applications ranging from healthcare to autonomous vehicles are rapidly deploying IoT solutions. It is thought that FL can offer more secure and shared security services for a wider range of applications, helping to support the steady growth of distributed ML applications [4]. Although the CFL systems are indeed promising, they face several limitations and challenges as they require a central FL server. Efficient communication is a critical challenge that needs to be addressed in FL to ensure that all participating devices are connected and that performance is not compromised [5]. Furthermore, in particular, it is hard to implement in some scenarios where a reliable and robust central server is difficult to find (e.g., a fully self-driving and autonomous network). Moreover, the CFL network faces the limitation of a single point of failure (communication bottleneck) at the central server. Therefore, this research will study a Decentralized Federated Learning (DFL) approach to overcome these challenges. In the DFL model, all neighbouring devices share their model parameters directly in a peer-to-peer manner. In this case, each device acts as a server by aggregating the parameters from neighbours and then averaging them to update a sub-global model that can be shared with others within one hop of a communication link.

The aim of this paper is to evaluate the feasibility of the implementation of the DFL approach under spatio-temporal dynamics. DFL algorithms are implemented using clusters of mesh networking groups located in different environments. Our system model can be translated into two practical deployment modalities: 1) Autonomous and Decentralized IoT networks: these are the networks formed by limited capability IoT and edge devices where decentralized FL is required intrinsically. For instance, networks formed by wearable IoT devices in battlefields or remote hospitals. 2) Edge/Fog-Assisted IoT networks: these are the hierarchical networks where the data from IoT devices is collected and processed by the edge/fog gateways [6]. These gateways then form a mesh network where DFL must be implemented. A typical example of such deployment is in agricultural IoT setup [7] or other smart city IoT applications. Both cases can be translated into the system model considered in this paper. In addition, this study will implement an FL network based on the slotted ALOHA protocol as a sub-optimal MAC protocol to evaluate the performance of the network and optimal configuration under the proposed setup.

The combination of the DFL with the slotted ALOHA mesh networking protocol is proposed to satisfy the users' privacy preservation, increase the protection for confidential data, increase the prediction accuracy of the implemented algorithms and build a robust system. This system can help us exploit a greater percentage of the users' data that will be trained to create a global model that can improve the local models in each participating device without sharing any data with other participants except the local model parameters.

B. KEY CONTRIBUTIONS

Limited studies were conducted using FL with a central server to achieve better results. The objectives of this study are to introduce and simulate the Centralized and Decentralized FL's wireless communication stage between the devices in the learning process based on qualitative examinations of the CFL approach with a central server and the DFL approach over a Wireless Mesh Networking (WMN) without a central server. Furthermore, this study aims to simulate the communication network for the CFL and DFL models to design a robust wireless communication network during the training process, which helps to evaluate how the model can perform in different network conditions, such as congestion and interference.

C. PAPER ORGANIZATION

The rest of the paper is organized as follows. In Section II, the related work on the FL approach is provided. Background and challenges for DFL over WMN are presented in Section III. Section IV, introduces the system model in terms of theoretical analysis of device communication in the network and learning metrics for DFL over WMN. The learning criteria for DFL are demonstrated in Section V. The proposed framework simulation and results are summarized in Section VI. Finally,

a summary of this research and future work is presented in Section VII.

II. RELATED WORK

Many recent papers have investigated the fundamentals of CFL algorithms [8], [9], [10], [11], and [12]. CFL and its central server algorithm are called the Federated averaging (FedAvg) algorithm were first proposed and implemented in [1]. The FedAvg algorithm is implemented to create a global model by averaging the aggregated parameters from the participants [1].

In [8], comprehensive research of CFL for mobile-edge networks was presented. The authors examined the critical implementation issues with existing solutions and potential applications of CFL in IoT and mobile edge networks. In addition, some existing limitations and challenges in CFL are highlighted, such as the difficulty of aggregating sufficient data, real applications' heterogeneous data distribution, and theoretical analysis of device communication and convergence. The work [9] reviewed the challenges in implementing CFL, future research directions and the existing CFL approaches. In [10] and [12] the authors surveyed the CFL implementations, devised a taxonomy, and overviewed the currently proposed solutions and their challenges in the CFL framework. They presented the essentials of preserving privacy and checking fairness in CFL.

The study in [13] conducted a thorough and comprehensive examination of the architecture, design, and deployment of FL, comparing it to centralized and distributed on-site ML-based systems. Furthermore, the challenges and potential future directions for research in FL were discussed, where some classification problems of FL topics and research fields were also presented, based on a thorough literature review, including taxonomies for its important technical and emerging aspects, such as the core system model and design, application areas, privacy and security, and resource management.

The authors in [11] examined the "In-Edge-AI" model for edge networks to allow for efficient collaboration between terminal devices and terminal servers to exchange learning model parameter updates. They explored two use scenarios: edge caching and compute offloading. To efficiently support these scenarios, they trained a double deep Q-learning (DDQN) model via CFL. Lastly, the authors in [8], [9], [10], [11], and [12] addressed several existing issues in CFL for actual applications, such as the ability of mobile devices to handle a high computation process and the power consumption and battery life to keep connected to a central server. Furthermore, CFL raises concerns about flexibility because it may cease to function due to the aggregation server's failure (i.e., due to a malicious assault or physical flaw). Moreover, training CFL models via IoT networks necessitates many communication resources to allow participants to communicate with a central server [14].

Most existing DFL systems are based on gossiping schemes, and the number of neighbours in the learning process are chosen regardless of communication challenges,

end-user capability and network capacity. For instance, the works in [15] and [16] implement a classic DFL algorithm that allows a user to aggregate the model parameters from an estimated number of multiple neighbouring devices, and Ramanan and Nakayama [17] propose an alternative approach that uses a blockchain-based FL scheme to aggregate updates for the participants' devices. However, these approaches suffer from several limitations related to the communication constraints in the real environment applications, the data size and the terminal capability (i.e., energy consumption and computation cost) of blockchain-based transactions. In summary, we list some existing works on FL-related topics with our paper's contribution in Table 1.

III. BACKGROUND AND CHALLENGES FOR DFL OVER WMN

The implementation of the DFL approach over a WMN is supposed to reduce communication costs, cope with the single point of failure issue in CFL, and provide innovative capabilities in a range of aspects, including healthcare systems (e.g., monitoring physiological data like heart-rate variability [18] to classify various cardiac pathologies), industry, and smart homes [19], [20], [21].

In addition, the combination of DFL and WMN using mesh protocols will likely be helpful in preserving privacy, guaranteeing a robust network, and improving the battery life of the devices by reducing communication costs. Instead of transferring data from the device to the central server provider, which will need large bandwidth and consume high power on the edge device, the decentralized approach can be applied to minimize these back-and-forth journeys of data. This decentralized fashion is implemented by processing the data into the edge device and communicating with the other neighbours using the mesh networking links to exchange and update the model parameters.

This paper will simulate and implement the DFL model over WMN system protocols. The model and the system performance will be evaluated by training the model using a dataset divided into training data, validation data, and test data. The performance metrics of the algorithms will be prediction accuracy, communication cost, and latency. Both DFL and WMN protocols are implemented in some applications separately and individually.

This research proposes integrating DFL and WMN into one intelligent system to optimize for robust communication networks that could be applied in many IoT applications to ensure participant privacy preservation and data security. Although DFL and WMN have impressive characteristics and features, they reveal several challenges and problems faced by engineers that can influence the model's accuracy. The following is a short summary of those limitations:

I. The network will be designed for low-power IoT devices under IEEE 802.15.4-2006 and WMN would need to be designed to coexist in IoT systems with other technologies, and not to replace them [32].

- II. Convergence speed: DFL algorithms usually adopt a peer-to-peer one-layer architecture. Each participant collects and aggregates all the local model updates of one-hop neighbours in a CFL architecture. With such multi-hop architecture, the wireless routing paths between participants can be easily saturated, resulting in a slower convergence speed [33].
- III. Unbalance: the amount of data varies at each participant resulting in different local training data quality.
- IV. Lack of stability and flexibility in communication networks of a massive number of devices in real-time applications.

The communication stage is one of four main steps in the learning process that cannot be neglected, and most researchers do not consider it analytically in their research. In this paper, this stage will be addressed in detail. We propose using mesh networking to maximize the communication stage's flexibility and the channel's capacity during the learning process. The motivation for this is the fact that the DFL algorithms can efficiently update the terminal edge with the parameters through the Thread protocol or any other mesh networking protocols (e.g., ZigBee or Bluetooth). This will allow us to design and develop a global model that can precisely analyze the end-users data without sharing the data with a central server or any other devices within the network. In other words, the data stays protected locally and never leaves the device itself and this will achieve personalization and guarantee high Quality of Service (QoS) as well as enhance the performance of devices in IoT applications.

To the best of our knowledge, this research is the first work that combines DFL and WMN using the slotted ALOHA protocol. The model results verify the intuition, showing that implementing DFL over mesh networks can offer more flexibility as no central server is required and promises more communication channels available to communicate. More participants can be involved in the learning process in the form of neighbour groups. The rest of this paper will introduce the wireless communication characteristics for the mesh networking and DFL criteria. The wireless communication constraints will be considered and CFL and DFL models will be implemented by simulations.

A. TRADITIONAL MACHINE LEARNING (ML) ON EDGE DEVICES

There are different kinds of ML and deep learning algorithms used in various proposals. For instance, Convolution Neural Network (CNN) algorithms are powerful tools widely used in image classification processing and other classification problems [34] since CNN has a proven ability to achieve higher accuracy, and can efficiently learn from thousands of image datasets. To implement CFL and DFL algorithms, the local ML algorithms are required to be embedded in the terminal devices (participants) to train the algorithm on the local data before sharing the parameters with the server in a CFL approach or with the neighbours in a DFL approach.

TABLE 1. Existing works on FL-related topics with our paper’s contribution.

Related works	Key topic	Evaluate recent advances design of the network in FL approaches					
		Resource management and system cost	Security and privacy	Users’ distribution analysis	Communication in the network	FL Network intensity and performance	Central Server-free
[22]	DFL concept	x	✓	x	✓	x	✓
[23]	FL concept	x	✓	x	✓	x	✓
[24]	Distributed ML	✓	x	x	✓	x	x
[25]	Security and privacy in FL	x	✓	x	x	x	✓
[8]	FL in edge networks	✓	✓	x	x	x	x
[26]	FL for IoT	✓	✓	x	✓	x	x
[27]	FL for IIoT	✓	✓	x	x	x	x
[28]	FL for health informatics	✓	✓	x	✓	x	x
[29]	Decentralized Wireless FL	x	✓	x	✓	x	✓
[30]	DFL framework	✓	✓	x	x	x	✓
[31]	Blockchain-based FL	✓	✓	x	x	x	✓
This paper	DFL over WMN	✓	✓	✓	✓	✓	✓

Details on ML-enabled edge devices challenges and opportunities have been addressed in many research papers in the recent past [35], [36] and they are out of the scope of this paper. In this research, the CFL and DFL models will be implemented for a classification problem, and the CNN algorithm will be the main algorithm that is used to train the local models on the proposed system.

B. CENTRALIZED FEDERATED LEARNING (CFL)

The main objective of CFL systems is to train, in coordination with a central server for model aggregation, a shared global model from participating devices that act as local learners. Figure 2 shows the fundamental CFL architecture and the main four steps to train a CFL network, where these steps are iterated until reaching the convergence status [12]:

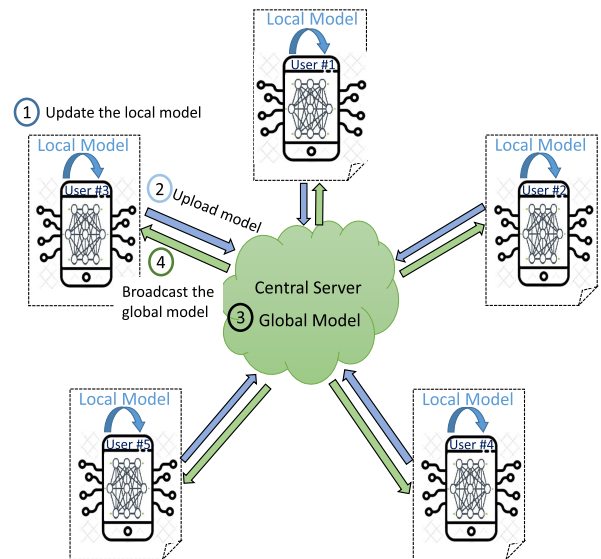


FIGURE 2. CFL framework over an IoT network.

- I. Local learning where each edge device uses its local dataset to train the model locally and update the parameters of the ML model (e.g., neural biases and weights).
- II. Upload model parameters to the central server: participants upload (transmit) their parameters to the central server via communication channels.
- III. Global aggregation: the central server aggregates the local models’ parameters from those successfully received to update a new version of the global model on the server.
- IV. Download (broadcast) and synchronize the devices with the latest global model updates [14], and then back to step 1 and repeat until system convergences.

In the CFL process, each device’s local algorithm has an optimization technique to update the model iteratively, such

as Stochastic Gradient Descent (SGD). Afterwards, the global model emerges from aggregating the local parameters from participants’ devices, which can then be weighted according to the perceived quality of the updates of the devices [34]. One crucial property of CFL is that the participant user data never transfers between devices and the server, which reduces communication costs and data sharing privacy concerns. However, due to the central server node, the CFL system will run into scalability issues. Even if the server node’s hardware and software capabilities have been optimized, the

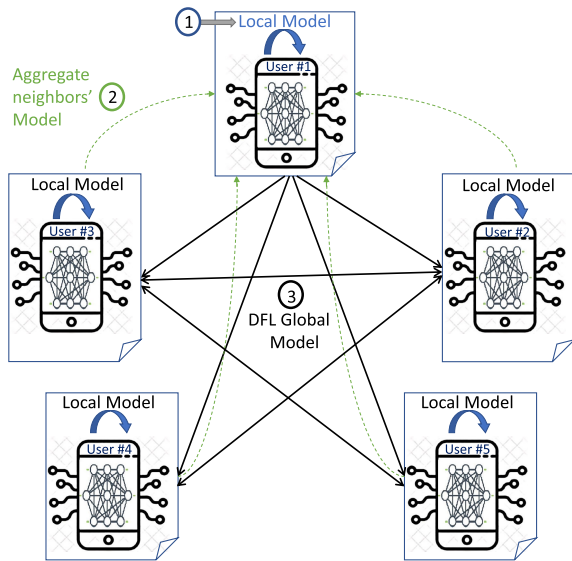


FIGURE 3. DFL framework over an IoT WMN.

server node's performance will not improve when thousands of client nodes join [37]. Communication bottlenecks may appear due to the amount of traffic that is increasing exponentially, and the system becomes overburdened. Furthermore, accessing a central node cannot be possible in some scenarios, for instance, self-driving vehicles and high mobility sensor systems. Decentralized architectures have recently been proposed to avoid communication bottlenecks and protect data privacy [38], [39]. Remove the centralized server, and each participant only communicates with its one-hop neighbours in its local area and exchanges their local models and updates parameters [40].

C. DECENTRALIZED FEDERATED LEARNING (DFL)

As shown in Figure 3, the DFL framework does not need a central server to coordinate the training tasks and contains only terminal participants (nodes) [37]. The idea is that each participant exchanges parameter updates with neighbours in a peer-to-peer manner. Besides the local model algorithm, the FedAvg algorithm is employed on each terminal participant to create its global models in the DFL approach with no central server. Each participant trains on its local data and averages it within the aggregated models' parameters from the selected neighbours using the FedAvg algorithm to broadcast an updated global model to the neighbours again at each iteration. Afterwards, the same procedure is applied to all other participants until the system converges.

In [41], the Combo algorithm proposed an approach in which the participants send and average a segment of the models' parameters to reduce the required communication bandwidth without affecting the system performance and convergence rate. Even though the proposed DFL algorithms overcome some of the problems associated with general FL systems that require a central server, they use model averages

to fuse models at the local clients, which is not always very efficient in heterogeneous data scenarios. For each, local model parameters are updated toward the local optimum, and averaging the model parameters from different clients leads to the averaged outcome of each client's local optimum being used. The optimum of each participant's loss function may be quite far from the others, which is also far from the global optimum due to different participants owning different sets of training data. Thus, those datasets typically have different distributions or even no overlap, which is defined as data heterogeneity [42], [43], [44].

Furthermore, most prior works do not consider the wireless environment in the communication network, so they do not account for wireless impairments caused by channel fading, link blockages, and wireless interference. Most researchers choose the number of participants by estimation without concerning the communication medium constraints, which is not always very efficient in terms of reliability and flexibility for real-time applications (e.g., AV and UAV networks).

Therefore, this study will focus on designing a model based on the gossip method [45] that can deal with decentralized approaches and heterogeneous datasets, and it will precisely analyze the communication process between the participants in the network, considering the interference and the noise in the transmission medium to simulate a real scenario of the communication connection between the terminal devices through the learning process.

D. WIRELESS MESH NETWORKING (WMN)

Nowadays, many access points have overlapping areas, and almost each traditional wireless network has to be connected to the wired network. In this scenario, the cost of installing IoT devices is costly and extremely difficult. Thus, a WMN will benefit from its flexibility to connect the devices within the network and offer a different perspective than non-mesh networks. The connectivity needs in wireless mesh networks are reduced mainly because the devices within the network have a multi-route capability to send and receive packets. In addition, the WMN also has a range of advantages such as self-healing and self-organizing, attracting a vast number of investigations and research developments [46]. Furthermore, WMN can reduce the networking cost for innovative home applications using low-profile hardware.

The routing protocols significantly contribute to WMNs as they help find the best path between multi-hop networks in unreliable wireless media. The WMN protocols have been widely investigated to achieve higher throughput, low latency and low power consumption [46], [47]. According to some related research, ZigBee, Bluetooth Low Energy BLE, Z-wave and Thread protocols are the most common protocols used in many wireless mesh-networking applications. Each protocol has its own unique specification for particular implementation depending on the user requirement. For instance, the Z-wave protocol is advantageous for long-range coverage because of its low-frequency band (900 MHz)

compared to others with a 2.4 GHz band (i.e., Bluetooth and Thread). According to [48], all protocols achieve similar performance (i.e., latency and throughput) for small networks and small payloads. By contrast, for large mesh networks with multi-hop nodes between the transmitter and the receiver, the Thread protocol achieves better performance metrics in terms of latency and efficiency.

IV. SYSTEM MODEL

A. COMMUNICATION IN WMN

In this paper, a typical receiver is considered that is connected to a corresponding desired transmitter. A Rayleigh fading channel is adopted for the small-scale path-loss model and complemented with a single slope large-scale path-loss. Hence, the received power at the typical receiver from the desired transmitter is $(P_k h_{k0} d_{k0}^{-\alpha})$ [49], where P_k is the signal transmit power from the desired transmitter device k , h_{k0} is the fading coefficient for the channel between device k to the target destination, d_{k0} is the distance between the desired transmitter k and the corresponding receiver and $(\alpha \geq 2)$ is the path-loss exponent.

The signal can be correctly decoded at the typical receiver if the corresponding SINR (Signal to Interference plus Noise Ratio) is higher than a certain threshold T_k . Therefore, the probability that $SINR \geq T_k$ is defined as:

$$P(SINR \geq T_k) = P\left(\frac{P_k h_{k0} d_{k0}^{-\alpha}}{\sum_{i \in \varphi} I_i + N_0} \geq T_k\right) \quad (1)$$

where I_i is the interference from the device i in the network ($I_i = P_i h_{i0} d_{i0}^{-\alpha} a_i$), with $i = 1, 2, \dots, M$, M is the total number of active devices within the desired receiver coverage area, and $i \in \varphi$ where φ represents the number of all participant devices within the whole target area. Now a_i is a binary random variable that defines the state of the device, whether in a transmitting state or ready to receive such that $P_A = \text{Probability}\{a_i = 1\}$ represents the device is in transmitting state and $1 - P_A = \text{Probability}\{a_i = 0\}$ represents the device is ready to receive and is not transmitting.

The proposed IoT network is assumed to have a small thermal noise power variance N_0 in comparison with the cumulative interference power (i.e., interference-limited network). Therefore, the noise effect in the network is negligible, and so (1) can be re-written as:

$$P(SINR \geq T_k) \cong P\left(\frac{P_k h_{k0} d_{k0}^{-\alpha}}{\sum_{i \in \varphi} I_i} \geq T_k\right) \quad (2)$$

$$\cong P\left(h_{k0} \geq \frac{T_k d_{k0}^\alpha \sum_{i \in \varphi} I_i}{P_k}\right). \quad (3)$$

Since the proposed IoT device network has Rayleigh fading channels, and for the sake of simplicity, the h_{k0} term is assumed to be an exponentially distributed random variable with a unit mean. The probability distribution function (PDF) of an exponential random variable X with unit mean can be expressed in the form of $f_X(x) = \exp(-x)$, and thus, (3) can

be reformulated as:

$$P(SINR \geq T_k) \cong \mathbb{E}_{a_i} \left(\mathbb{E}_{h_{i0}} \left(\int_{\frac{T_k d_{k0}^\alpha \sum_{i \in \varphi} I_i}{P_k}}^{+\infty} (e^{-x} dx) \right) \right) \quad (4)$$

$$\cong \mathbb{E}_{a_i} \left(\mathbb{E}_{h_{i0}} \left(\exp\left(-\frac{T_k d_{k0}^\alpha \sum_{i \in \varphi} I_i}{P_k}\right) \right) \right) \quad (5)$$

$$\cong \mathbb{E}_{a_i} \left(\mathbb{E}_{h_{i0}} \left(\prod_{i \in \varphi} \exp\left(\frac{T_k d_{k0}^\alpha I_i}{P_k}\right) \right) \right) \quad (6)$$

$$\cong \mathbb{E}_{a_i} \left(\mathbb{E}_{h_{i0}} \left(\prod_{i \in \varphi} \exp\left(\frac{-T_k d_{k0}^\alpha P_i h_{i0} d_{k0}^{-\alpha} a_i}{P_k}\right) \right) \right) \quad (7)$$

where \mathbb{E}_{a_i} is the expectation with respect to the random variable a_i and $\mathbb{E}_{h_{i0}}$ denotes the expectation of the fading coefficients from the devices i to the desired receiver.

The Bernoulli distribution property can be used to simplify (7), $\mathbb{E}_{a_i} \exp(a_i x) = (1 - P_A + P_A \times \exp(x))$, and according to [49] $\mathbb{E}_{h_{i0}} (\exp(-h_{i0} \times y)) \cong 1/(1 + y)$.

Consequently, the probability of successful transmission for the participant devices within the network area can be explicitly obtained as:

$$P(SINR \geq T_k) \cong \prod_{i \in \varphi} \left(1 - P_A + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) (\gamma_{ki}))} \right) \quad (8)$$

where $\gamma_{ki} = P_i/P_k$ is the power ratio.

B. ACHIEVABLE TRANSMISSION CAPACITY OVER SLOTTED-ALOHA

The ALOHA protocol is a class of fully decentralized MAC protocols [50] that does not perform carrier sensing and attempts to avoid packet collision. The slotted-ALOHA protocol was introduced to enhance the utilization of the shared communication medium and reduce the chances of collisions for multiple transmitting devices by synchronizing the transmission of devices at the beginning of discrete timeslots.

In the WMN, the probability for each device sending a packet to neighbours is P_A and the probability of being ready to receive a packet is $(1 - P_A)$, and therefore, the probability of every mesh node getting a packet from one of its neighbours is $P_A(1 - P_A)P(SINR \geq T_k)$ where T_k is the predefined SINR threshold and $P(SINR \geq T_k)$ represents the successful transmission probability. Based on Shannon's Theorem, the achievable transmission capacity increases when the transmit power increases, and the packet can carry up to $\log(1 + SIR)$ (bits/second/hertz) data [49]. Thus, the capacity C_{ALOHA} for the mesh network can be written as:

$$C_{ALOHA} = P_A(1 - P_A)\log(1 + SIR)P(SINR \geq T_k). \quad (9)$$

From (8), the outage probability (θ_k) constraints concerning P_A , T_K and γ_{ki} can be obtained as follows:

$$1 - \prod_{i \in \varphi} \left(1 - P_A + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) (\frac{P_i}{P_k}))} \right) \leq \theta_k. \quad (10)$$

To simplify (9), a natural logarithm is applied to compute the maximum achievable transmission capacity with respect to P_A . Hence, we use an auxiliary function $f(P_A) = \ln(C_{ALOHA}(P_A))$ to simplify the search of the optimal value P_A that maximizes C_{ALOHA} . Then, the problem formulation can be written as:

$$\begin{aligned} \arg \max_{P_A} f(P_A) = & \arg \max \left(\ln(P_A) + \ln(1 - P_A) \right. \\ & + \ln(\log(1 + T_k)) \\ & \left. + \sum_{(i \in \varphi)} \ln \left(1 - P_A + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})} \right) \right) \end{aligned} \quad (11)$$

$$\text{s.t. } \varepsilon_k \leq \sum_{(i \in \varphi)} \ln \left(1 - P_A + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})} \right). \quad (12)$$

With a Taylor series expansion, which is $\ln(1 - x) = -x - x^2/2 - x^3/3 - \dots \cong -x$, ($|x| < 1$), so (12) can be simplified to:

$$\varepsilon_k \leq \sum_{(i \in \varphi)} \left(-P_A \left(1 - \frac{1}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})} \right) \right). \quad (13)$$

Thus, (11) can be reformulated as:

$$\begin{aligned} \arg \max_{P_A} f(P_A) = & \arg \max_{P_A} (\ln(P_A) + \ln(1 - P_A) \\ & + \ln(\log(1 + T_k))) \\ & + \sum_{(i \in \varphi)} \left(-P_A + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})} \right). \end{aligned} \quad (14)$$

Hence, (14) can be written as:

$$\begin{aligned} \arg \max_{P_A} f(P_A) = & \arg \max_{P_A} (\ln(P_A) + \ln(1 - P_A) \\ & + \ln(\log(1 + T_k)) - P_A f_2) \end{aligned} \quad (15)$$

where $f_2 = \sum_{(i \in \varphi)} [1 - \frac{1}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})}]$, and therefore a partial derivative of $f(P_A)$ will be taken to find P_A that will maximize the function $f(P_A)$ in (15):

$$\frac{\partial f(P_A)}{\partial P_A} = \left(\frac{1}{P_A} - \frac{1}{(1 - P_A)} - f_2 \right).$$

The probability of being in transmitting mode using the slotted ALOHA protocol at $\frac{\partial f(P_A)}{\partial P_A} = 0$ is defined as $P_A(0)$ and obtained as:

$$P_A(0) = \frac{f_2 + 2 - \sqrt{f_2^2 + 4}}{2f_2}$$

so, $\frac{\partial f(P_A)}{\partial P_A} > 0$, when $0 \leq P_A < P_A(0)$ and $\frac{\partial f(P_A)}{\partial P_A} < 0$ when $P_A(0) < P_A < 1$. Consequently, if the system is assumed to have a constant threshold T_k and power ratio γ_{ki} , then $f(P_A)$ will increase when P_A is increased within the

range of 0 to $P_A(0)$, and by contrast, it will decrease when $P_A > P_A(0)$.

A special case has been adopted by setting the system's outage probabilities equal to its SINR threshold values to find the maximum transmission capacity based on the slotted ALOHA protocol for the wireless mesh networks. Then (9) can be reformulated to restate the maximum achievable transmission capacity is as follows:

$$\begin{aligned} \max(C_{ALOHA}) = & P_A(1 - P_A)\log(1 + T_k)(1 - \theta_k), \\ = & \left(P_A(1 - P_A)\log(1 + T_k) \right. \\ & \times \prod_{(i \in \varphi)} \left(1 - P_A \right. \\ & \left. \left. + \frac{P_A}{(1 + T_k (\frac{d_{k0}^\alpha}{d_{i0}^\alpha}) \gamma_{ki})} \right) \right). \end{aligned} \quad (16)$$

From this analysis, the intuition is provided as to what extent the network parameters T_k , P_A and γ_{ki} effect the maximum achievable transmission capacity of the slotted-ALOHA mesh network.

C. LATENCY

The required time to achieve a convergence state in any FL model plays a key role in evaluating the system performance. Therefore, the average learning latency for the participants will be one of the performance metrics. Latency in the proposed centralized and decentralized FL will be defined as the expected time duration (in seconds) required for the model to complete learning in a typical one-hop mesh communication network. Let R denotes the smallest number of iterations to meet the convergence criterion. The expected learning latency (T_{total}) is a function in the computation time ($T_{computation}$) which is the required time for the device to run and update the local model, the server to devices (participants) broadcast communication time ($T_{broadcast}$) and the device to server communication time (T_{cmm}) over a number of iterations R for the number of successful transmit participants A_i in the learning process:

$$T_{total} = R \sum_{(s_i=1)}^{A_i} (T_{cmm}^{(s_i)}) + R(T_{computation} + T_{broadcast}). \quad (17)$$

The summation of the communication process time T_{cmm} in real applications is slightly variable depending on the number of participants in each iteration. In order to simplify the calculation, it will be assumed that the network has a fixed average number of successful transmission participants A_i for the whole process in centralized and decentralized FL.

D. ACCURACY AND LOSS

Accuracy and loss functions are the two main model metrics that are mainly applied to adjust the model weights during the training process and to measure the system performance in order to optimize a model (e.g., a convolution neural network model) and solve for example, a face recognition problem.

Accuracy is calculated as follows:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}. \quad (18)$$

Loss is a measure of the difference between the actual output value and the predicted output value by the implemented model. In the classification models whose output values are an array of probability values between 0 and 1, the most common loss function applied is the cross-entropy loss function. The cross-entropy loss function is also known as logistic loss, log loss or logarithmic loss. The probability of each predicted value is weighed against the actual desired output 0 or 1, and a loss is measured based on how far it is from the actual expected value for each sample. A larger loss for significant differences close to 1 and more, and a slight loss for minor differences tending to 0, and therefore, the overall cross-entropy loss of 0 means the model is perfect. The cross-entropy loss function is defined as:

$$\text{Loss} = - \sum_{j=1}^m \sum_{i=1}^n y_{(i,j)} \log(p_{(i,j)}), \quad (19)$$

for n classes and m samples

where $y_{(i,j)}$ is the actual output and $p_{(i,j)}$ is the softmax probability of the model output for the i^{th} class in the classification problems and j^{th} instance sample.

Therefore, the objective is almost always to increase the accuracy and minimize the loss of the FL models or any other implemented models.

V. THE LEARNING CRITERION FOR DFL

The designed system considers a group of φ individual participants nodes (i.e., edge devices) in which set $\varphi = 1, \dots, K$ is randomly located as a spatial point process following a stationary Poisson Point Process PPP [51] with intensity (λ) in disk cells with uniformly distributed over a large-scale network. In supervised learning, each node $i \in \varphi$ has access to a dataset $\mathcal{D}_{(i)}$ consisting of n instance-label pairs of samples (X_i^n, Y_i^n) , where X_i^n and Y_i^n represent the labelled sample input and output for the device i , respectively and $n = 1, \dots, k$. The samples are a subset of heterogeneous or homogeneous datasets that follow an unknown probability distribution $p(x, y)$ and possibly have non-empty interaction for different sets (i.e., $\mathcal{D}_{(i)}$ and $\mathcal{D}_{(j)}$ where $i \neq j$). Each instance $X_i^n \in X_i \subseteq X$, where X_i denotes the local instance space of node i and X denotes a global instance space, which satisfies $X \subseteq \bigcup_{i=1}^{\varphi} X_i$.

Similarly, let y denote the set of all possible labels over all the nodes. Some examples include $y = 0, 1$ for binary classification learning and $y = \mathbb{R}$ for regression learning.

There are $X_i^{(1)}, X_i^{(2)}, \dots, X_i^{(k)}$ i.i.d samples at each participant node, and the total number of examples k is a variable depending on the size of the local datasets for each one.

All participants' devices have to have the same machine-learning model (e.g., a CNN), which have a common weight parameters matrix (W). The main aim of the designed model is to minimize the cross-entropy loss function between the expected (actual) output and the predicted

output:

$$\arg \min_W F(W) \triangleq \frac{1}{A_i} \sum_{i \in \varphi} f_i(W) \quad (20)$$

where $F(W)$ denote the global loss function, $f_i(W)$ represents the local loss for the device i , s_i represents the set of active devices that succeed in transmitting to the i^{th} participant at each iteration in the training process and A_i is the length of the s_i set as:

$$s_i = 1, 2, \dots, A_i, \quad \forall \text{ SINR} \geq T_k.$$

The local loss function for the i^{th} device is calculated by the model cross-entropy loss for the local training dataset $\mathcal{D}_{(i)}$ as follows:

$$f_i(W) = \frac{1}{M} \sum_{(m=0)}^M l(h_W(X_i^m), y_i^m),$$

$$m = 0, 1, 2, \dots, M \quad (21)$$

where $m \subseteq n$ is the subset of the total number of local training datasets and M is equivalent to the length of the participant datasets divided by the batch size as follows ($\frac{|\mathcal{D}_{(i)}|}{B_i}$), and $l(h_W(X_i^m), y_i^m)$ denotes the cost function for the weights matrix W evaluated on a hypothesis $h_W(X_i^m)$ With data samples X_i^m . For instance, the hypothesis for the simple linear regression is defined as $h_W(X) = W_0 + W_1 X$.

At the i^{th} iteration of the DFL process, each device $i \in \varphi$ has a local parameters weights matrix $W_i^{(t)}$ that is updated to maximize the model accuracy and to find an optimum solution to the target problem.

In the CFL models, there will be two optimizers' levels. First is the local model level; the local optimizer in each device to update the local parameters based on the local dataset can use a common machine learning optimizer called Stochastic Gradient Descent algorithm SGD. Second is the global model level; a global model optimiser uses the aggregated parameters from neighbours (i.e., Federated Averaging algorithm (FedAvg)) to create and update the global model.

Each participant i makes M training passes over its local datasets $\mathcal{D}_{(i)}$ to update its local model weights simultaneously via SGD with learning rate (η) and batch B_i which is shown as follows:

$$\nabla f_i(W_i^t) = \frac{1}{M} \sum_{(m=0)}^M \nabla l(h_{(W_i^t)}(X_i^m), y_i^m), \quad (22)$$

$$\forall m = 1, \dots, M \text{ and } i \in \varphi$$

$$W_i^t := W_i^t - \eta_i \nabla f_i(W_i^t). \quad (23)$$

Here $\nabla f_i(W_i^t)$ denotes the gradient matrix obtained from the batch B_i of the device i 's local samples after M local training on the local data at each iteration t . The gradient $\nabla f_i(W_i^t)$ expresses the rate of change of f_i with respect to the model parameters W at the iteration t .

The total number of local training M is equivalent to the participant datasets length divided by the batch size ($M = |\mathcal{D}_{(i)}|/B_i$) in order to train the model locally on its datasets

before sharing the parameters with neighbours. After a certain number of iterations for each participant device trained a model on its local dataset with sufficient hyper-parameters, the models' parameters are shared with other neighbours via a one-hop communication process following the mesh topology.

For all participants, the updates are simultaneously done where each participant receives the other neighbours' weights and gradients and averages them with the local weights and gradients.

In the next step, each participant in the network successfully aggregates the local parameters W_i^t from the trusted neighbours who satisfied the wireless communication constraints to execute the Federated Averaging (FedAvg) algorithm in order to obtain a new global model at each t^{th} iteration. Then, each authorized participant sends a broadcast containing the latest global model weights matrix to all participants located within a one-hop mesh network and starts the next iteration. Figure 3 illustrates the proposal DFL network architecture and the learning process steps. The global model which can be denoted as \hat{W}_i^t is executed by averaging the local model with the aggregated model weights from the neighbours s_i (the set of active devices whose received their weights successfully) at the iteration t .

These weights are then applied on the local model using the device dataset $\mathcal{D}_{(i)}$ and batch size B_i to create an updated weights matrix $W_i^{(t+1)}$ and broadcast it to the neighbours in the next iteration as follow:

$$\hat{W}_i^t = \frac{1}{A_i + 1} \left(W_i^t + \sum_{(s_i=1)}^{A_i} \hat{W}_{s_i}^t \right) \quad (24)$$

$$\hat{\nabla} f(\hat{W}_i^t) = \frac{1}{M} \sum_{(m=1)}^M \hat{\nabla} l(h_{(\hat{W}_i^t)}(X_i^m, Y_i^m)); \quad (25)$$

$$\hat{W}_i^t = W_i^{(t+1)}. \quad (26)$$

Then these new parameters $W_i^{(t+1)}$ at the device i will be used to train the local model in the next iteration and also share $W_i^{(t+1)}$ with all participants within one-hop communication range to repeat the same procedure at each device until achieving the predefined convergence bound ε_k of the implemented model. The convergence is determined by measuring the average loss function of all active device ($s_i = 1, \dots, A_i$) whose successful transmission to the i^{th} device at each iteration as follows:

$$\arg \min_W \nabla F_i(W) \triangleq \left[\left(\frac{1}{(A_i + 1)} (\hat{\nabla} f_i(\hat{W}_i^t) + \sum_{s_i=1}^{A_i} \hat{\nabla} f_{s_i}(\hat{W}_{s_i}^t)) \Big|_{A_i \geq 1} \right) \right] \leq \varepsilon_k. \quad (27)$$

The participants will communicate and exchange the parameters and the system loss function's values over a wireless mesh network via a peer-to-peer manner. The number of successful transmit devices (participants) is subject to wireless communication constraints. Consequently, it is possible

to show that every device i in the network after receiving the latest global model update can train its model using the latest update and then evaluate how the model performance improved in terms of the loss functions and accuracy.

To sum up, the DFL process is divided into t iterations. At each iteration, each i^{th} participant performs local learning and exchanges their parameters \hat{W}_i^t in parallel with other cooperative neighbours s_i to create global models at each device in a central server-free manner. The R symbol in pseudocode represents the total number of iterations the system needs to achieve the consensus and meet the gradient divergence condition in (27). Finally, the DFL learning steps are presented in Algorithm 1.

VI. SIMULATIONS AND RESULTS

A. THE SIMULATION OF THE NETWORK COMMUNICATION

The proposal network for the CFL and DFL approaches will have many participants were distributed on a two-dimensional bounded space. A large-scale circular area with radius $R > 0$ is proposed for notational simplicity, but a hexagonal one can also be applied, as the outcomes will differ slightly by a very small constant. The number of participants is distributed according to the Poisson distribution with network intensity λ and area A . In other words, the number of participants within A can be defined as a Poisson random variable with mean λA , where $\lambda > 0$ is the network intensity, and A is the network circular area ($A = 2\pi R^2$).

On the one hand, the central point in a two-dimensional bounded space will be the position for the central server of the network in the CFL model as illustrated in Figure 4 (a), and the participants will be randomly distributed within the target area. However, the successful transmission and participation in the learning process will be subject to wireless communication constraints (see section IV) to approximate the real situation applications.

On the other hand, in DFL, each point (participant) i will define each neighbour based on a one-hop communication link within the device signal coverage that will be a smaller circle with a radius $r_i < R$. For instance, the wireless mesh connections for three DFL participants with a Rayleigh fading are illustrated in Figure 4 (b).

The positions of the participants are independently and randomly distributed within the network area (circular area), where the distance for each participant i to the centre point (central server) is $d_{k0} \leq R$. In the simulation, the relationship between the SINR threshold T_k and the probability of successful transmission $P(\text{SINR} \geq T_k)$ of the network, participants will be found based on this study analysis as in (1) from IV and the total number of participants is a random variable in the network space radius $R = 1000 \text{ m}$ and the intensity $\lambda = 0.1$.

The participants are assumed to have equal transmit power (0.8 Watts). Only a finite number of participants can simultaneously exchange and update their parameters based on the slotted ALOHA protocol. When the given parameters

Algorithm 1 Decentralized Federated Learning

```

1:
2) All participants have initial weights with  $W(0)$ 
3) for each iteration  $t = 1, 2, 3, \dots, R$  do
4)   for each device  $i = 1, 2, 3, \dots, K$  do {in parallel}
5)     for  $m = 1, 2, 3, \dots, M$ , where  $M = \frac{|\mathcal{D}_{(i)}|}{B_i}$  (Local training steps)
6)        $\nabla f_i(W_i^t) = \frac{1}{M} \sum_{m=1}^M \nabla l(h_{(W_i^t)}(X_i^m, y_i^m))$ 
7)        $W_{(i)}^t \leftarrow W_{(i)}^t - \eta_i^t \nabla f_i(W_i^t; B_i)$ 
8)     end
9)     from the  $i^{\text{th}}$  device's neighbours  $s_i = 1, 2, \dots, A_i$  do {in parallel}
10)      receive  $\hat{W}_{s_i}^t, \hat{\nabla} f_{s_i}(\hat{W}_{s_i}^t)$ ,
11)       $\hat{W}_i^t \leftarrow \frac{1}{(A_i+1)}(W_i^t + \sum_{s_i=1}^{A_i}(\hat{W}_{s_i}^t))$ 
12)       $\hat{\nabla} f_i(\hat{W}_i^t) = \frac{1}{M} \sum_{m=1}^M \hat{\nabla} l(h_{(\hat{W}_i^t)}(X_i^m, y_i^m))$ 
13)      If  $\frac{1}{(k+1)}(\hat{\nabla} f_i(\hat{W}_i^t) + \sum_{s_i=1}^{A_i} \hat{\nabla} f_{s_i}(\hat{W}_{s_i}^t)) |A_i \geq 1| \leq \varepsilon_k$ 
14)        Yes: end process (Gradient Convergence)
15)        No: continue
16)       $W_i^{(t+1)} \leftarrow \hat{W}_i^t$ 
17)    end
18) end

```

(i.e., signal power, distance, and interference) were applied in the SINR equation 8, the theoretical results matched the simulation results, and both proved that the number of successful transmitter devices decreased when the SINR threshold increased for different intensity of users in the network, as shown in Figure 5.

In contrast, reducing the SINR threshold allows more participants to involve in the learning process, but the required bandwidth and the system's latency will also increase. Therefore, there is a need to examine the trade-off between the probability of success and the SINR threshold to achieve higher throughput and capacity and acceptable latency.

Consequently, the proposed wireless communication model outcomes in Figure 5 confirm the conclusions of the theoretical analysis in (8) and (16) where a trade-off between the probability of success transmit and the SINR threshold is required to satisfy the FL network target in term of the capacity and the number of users (participants) within the network during the learning process.

B. CFL SETUP AND SIMULATION

The simulation settings are proposed for implementing an FL with a central server deployed in the centre of the target disk area with a radius R . It has been assumed that the network has a large-scale massive IoT and edge devices network following the PPP with intensity λ and the total number of participants is N , the distance from any participant i to the server is $r_i < R$.

To simplify the simulation, all participants are assumed to have the same transmit power and are all randomly distributed around the centre within the network area. The system will be trained using Python and Tensorflow APIs frameworks, the well-known MNIST dataset and a local algorithm on the edge devices to perform digit number recognition from

handwritten images. And while this is a simple problem and well-understood problem, it is being used here to illustrate the principle. The MNIST dataset contains labelled data samples of 60K images, each with a size of 28×28 pixels. This dataset was shuffled to distribute randomly for the participants, where each participant had 200 random samples. The edge devices will implement a CNN algorithm as a classifier model to evaluate the outcome results and the system performance metrics. Large-scale massive IoT networks are used to validate the algorithm, and thus the network has two cases:

- I. Low mobility scenarios where the number of participants' success transmitted in the system can be the same for all iterations until the system convergences.
- II. High mobility scenarios where participants move in a wide geometric area and many participants are IN and OUT of the network during the learning processes, such as autonomous vehicles and Unmanned Aerial Vehicle (UAV) networks. Thus, the number of successful transmits is usually different in each learning iteration.

This simulation will implement the first case, low mobility scenarios, and we will leave the second case for subsequent works. In both cases, the number of successful transmit participants in the system is subject to the communication network constraints (described in section IV) and the effect of network parameters in terms of the devices' intensity and participants' distribution.

Based on the proposed wireless communication model in VI-A, the system will be allocated the target SINR threshold (-10 dB) in order to achieve a higher transmission capacity, the probability of the participant to be in a transmit mode $P_A = 0.5$ and the intensity of users' devices $\lambda = 0.1$ to increase the number of participants. This results in

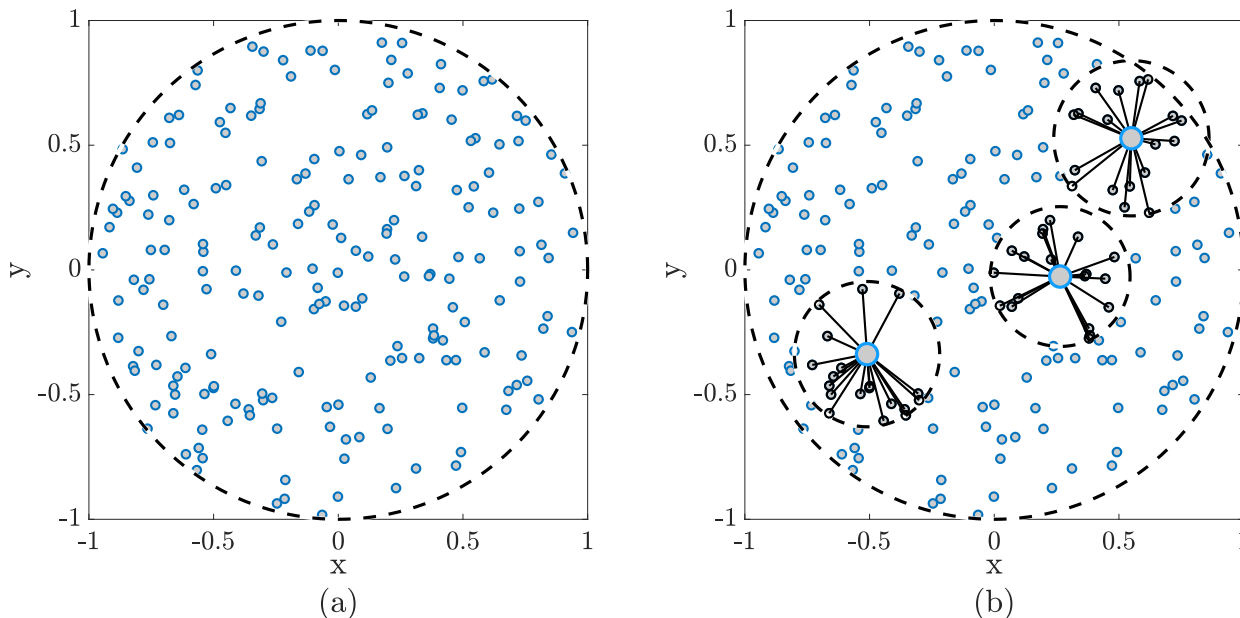


FIGURE 4. (a) The distribution for random participants around the CFL centre server. (b) Example of three participants communicating with neighbours in DFL approach.

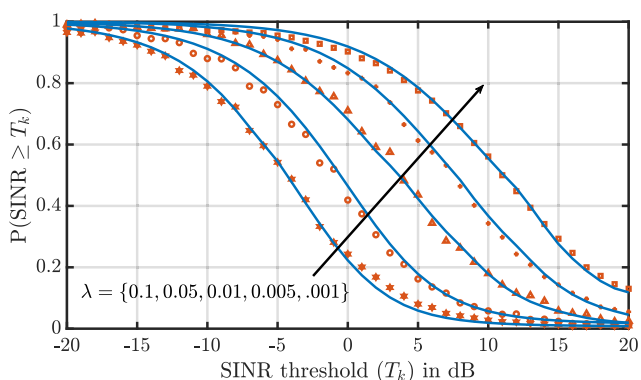


FIGURE 5. The simulation (markers) and theoretical results (solid lines) of the relationship between the SINR threshold and the probability of success for the participants within different network intensities.

having around 80% probability of successful transmit devices ($P(\text{SINR} \geq T_k)$) in the network as shown in Figure 5.

Consequently, the total number of participants within the target area is 80 devices, but the number of successful transmit participants in the CFL process was 65 participants.

In this study, the CFL with a central server is designed by using the FedAvg algorithm as a global model optimizer. The main function of the server algorithm is to aggregate and average the participants' parameters to update the new global model at each iteration and then measure the accuracy and the loss of the model outcomes.

The system evaluation for the CFL network optimization regarding the accuracy and loss used the cost function Cross-Entropy. The simulated system in subsection VI-A has 65 successful transmit participants as regards the communication constraints in subsection IV.

It can be noted from Figure 6 that the procedure moved progressively towards the global minimum dramatically in the first 50 epochs, where the accuracy increased from 17% to 90%. After 200 iterations, the model moved progressively and achieved the convergence state with the accuracy and loss of 98.1% and 0.15, respectively. The latency of the model was 1850 seconds, which was the required time to achieve the predefined convergence bound. Despite having applied constraints for the proposed CFL model to build a robust CFL network by considering the communication constraints and real environment scenarios in the simulation, the model was capable of converging faster and achieving slightly higher accuracy and lower loss than the FL baselines that have not considered the communication constraints in practice. In other words, the baselines estimate a fixed number of participants in the learning process without considering the network challenges, which cannot be reliable in real applications. In contrast, our proposed CFL model defines the participants based on the communication model and considers the real applications environment's constraints to obtain a trustworthy and robust network while achieving high accuracy and low loss.

Although the system achieved high accuracy and reached convergence after around 200 iterations in the designed simulation, there is potentially a single point of failure at the central server. The system, in reality, could be completely down if any failure occurs in the central server or the link to the server is blocked (communication bottleneck).

C. DFL SETUP AND SIMULATION

The set-up simulation uses Python and Tensorflow APIs frameworks to implement and evaluate the design of the DFL

system over a WMN for IoT devices. In order to make an accurate comparison, we concentrate on the same previous CFL design in terms of the communication links constraints and the parameters model optimizers, low mobility (fixed) and the same total number of participants in the learning process but without a central server. The system was evaluated based on performance metrics; the validation accuracy, the latency and the convergence speed rate. The simulation settings are listed as follows:

- **Training settings**

The participants trained a classification CNN model on the MNIST datasets. As in CFL, each participant will have 1000 random samples to train the local models, but the parameters in this DFL approach will be shared directly with neighbours to create and update the global models at each device without needing a central server.

The local models of each participant were trained and updated using a stochastic gradient descent (SGD) optimizer with the same learning rate of 0.01 and batch size 32 (hyper-parameters) in CFL simulation.

Furthermore, the designed system also used another two different local models optimizers' algorithms, the Adam algorithm [52] and Root Mean Square Propagation (RMSProp) optimizer [53], to observe how the performance of the DFL model gets affected by implementing different optimizers' methods.

- **Network Settings**

In the simulation, the network is designed to simulate the communication stage between the participants' devices in a peer-to-peer manner over wireless mesh networking, avoiding the communication bottleneck challenge in the central server in the CFL scenario. Each participant will successfully communicate and exchange the parameters with neighbours if the desired transmitters have an SINR over the target threshold and no collision occurs. The number of successful transmits participants is configured to be variable between the participants in an asymmetric manner. For instance, there are two participants A and B within the network, and participant A can receive parameter updates from participant B. In contrast, participant B does not have to successfully obtain the parameters update from participant A unless he satisfies the network communication constraints.

- **Comparison Setting**

The DFL system outcomes will be compared with the CFL outcomes in terms of system performance. Both DFL and CFL were implemented to train their models on the same factors (i.e., the dataset, the total number of participants within the network and the network geographic area) in order to make them comparable. Thus, the accuracy, loss and latency and convergence speed were measured.

- **Simulation Results**

Based on the successful transmit conditions and the network capacity, every device within the network created its one-hop neighbours' group to exchange parameters and performed the CFL model training. The model results have been

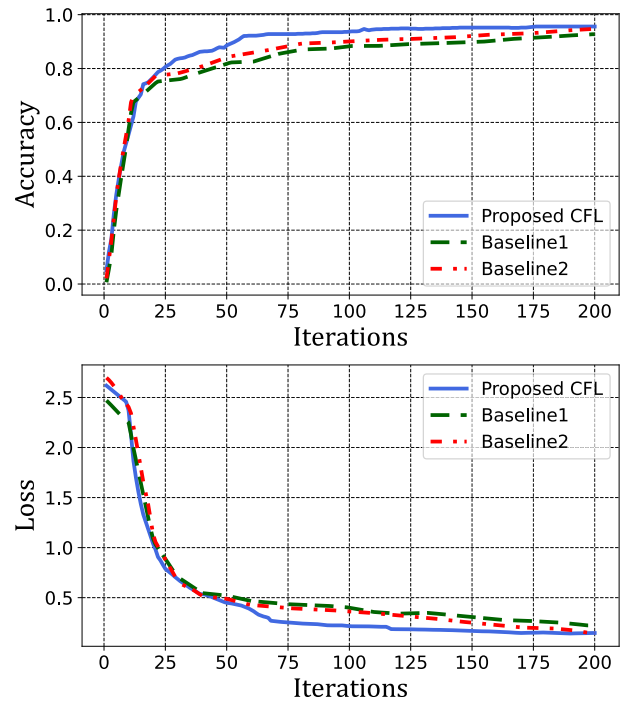


FIGURE 6. The accuracy and loss for the CFL network.

recorded for some random participants in the DFL simulation as an example to evaluate the system behaviour.

The designed network shows that these random recorded participants are connected successfully with a variant number of neighbours (who meet the communication constraints and they are able to exchange parameters and update their global models).

The DFL's simulated system has shown that the classification prediction achieved about 90% accuracy in the first 20 iterations for all these recorded participants with any of the three different model optimizers mentioned in the training settings (SGD, ADAM and RMSprop optimizers).

The results and statistics of some participants within the network are illustrated in Figure 7, and Table 2. The results for the DFL approach without a central server show high accuracy and low cross-entropy loss in predicting the digit number from the handwritten samples after 135 iterations.

In this study, the wireless mesh networking model in the section (VI-A) is integrated with the DFL model to verify the number of successful participants during the learning process. The system shows that each participant will be able to communicate with particular neighbours depending on the neighbours' locations, the desired participant transmit power and other devices' power interference.

The latency for each participant varies slightly depending mainly on the number of iterations that each participant is required to achieve the system convergence status. In this study, the results have shown that the average latency for the participants was 420 seconds with the assumption the

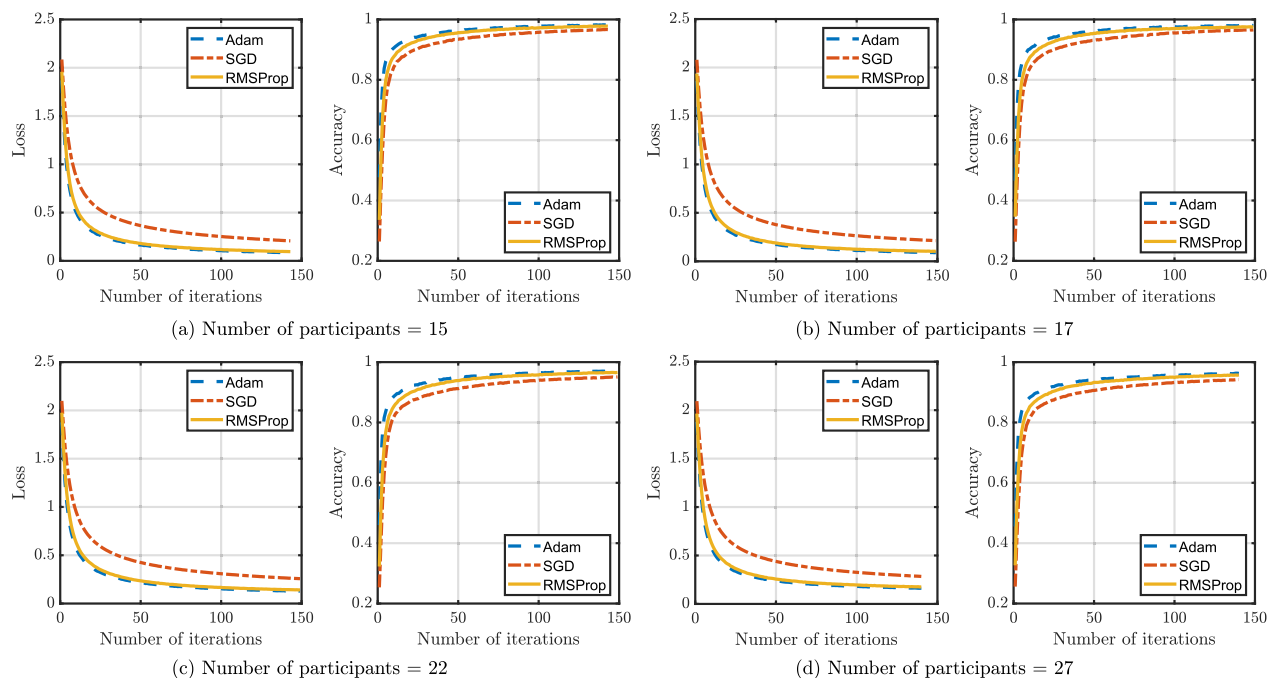


FIGURE 7. The DFL model outcomes (Accuracy and Loss) for four random participants.

TABLE 2. Simulation results for some random participants of the designed DFL model.

Optimiser Algorithm	No. of Succ. Transmit Neighbours	Iterations	Accuracy			Cross-Entropy Loss		
			ADAM	SGD	RMSprop	ADAM	SGD	RMSprop
Participant 1	15	135	0.982	0.966	0.969	0.088	0.214	0.149
Participant 2	17	135	0.979	0.964	0.974	0.094	0.200	0.138
Participant 3	22	135	0.970	0.948	0.965	0.133	0.270	0.237
Participant 4	27	135	0.962	0.951	0.957	0.164	0.218	0.178
Participant 5	13	135	0.979	0.957	0.974	0.131	0.2673	0.2346
Participant 6	24	135	0.981	0.959	0.976	0.132	0.266	0.2341
Participant 7	22	135	0.986	0.971	0.981	0.092	0.197	0.136
Participant 8	19	135	0.974	0.961	0.964	0.125	0.268	0.231
Participant 9	10	135	0.977	0.956	0.963	0.093	0.187	0.211
Participant 10	24	135	0.983	0.951	0.972	0.161	0.256	0.236
Participant 11	18	135	0.975	0.946	0.969	0.171	0.218	0.198

computation and broadcast time are equal for all participants. The expected system convergence in the DFL model is around 130 iterations on average.

As shown in Figure 7, the system achieved sufficiently high accuracy and low loss in the data predictions. The random chosen participants’ records show that the participants can learn in parallel and follow similar progress toward convergence in terms of accuracy and loss. The designed system offers the benefits of utilizing the DFL framework, where the data never leaves the participants’ devices, and privacy restrictions exist. In comparison with the centralized model, the decentralized model achieved competitive results

without needing a central server. For instance, participant 1 reached 98.2% accuracy and 0.088 cross-entropy loss after only 135 iterations.

In contrast, after more than 200 iterations, the centralized model achieved the convergence state with the accuracy and loss of 98.1% and 0.15, respectively. Based on the latency criteria in the subsection IV-C, CFL has higher latency than DFL as the system outcomes find that the communication cost and the number of iterations of CFL were higher than the DFL approach.

Thus, DFL can reduce the latency and loss and increase the convergence speed in which they can outperform CFL.

From these results, we can conclude that the DFL models over WMN produced significant developments in classification prediction using sufficient datasets. In this study, the DFL framework is combined with the wireless mesh networking using the Slotted-ALOHA protocol to improve communication between the participants during the learning process.

The DFL approach over WMN using the slotted-ALOHA protocol could be very competitive to the CFL. The simulated models prove that the designed DFL system can achieve better latency, more flexibility, and similar accuracy without installing a central server in the network.

VII. CONCLUSION

To sum up, DFL reduced the communication cost compared to CFL as the participants' devices communicate directly and send the packets of their parameters and updates with only one-hop neighbours using slotted-ALOHA as the devices' MAC protocol. The network topology was a mesh network topology. In this study, the network communication model simulated the real scenario of the mesh networking topology considering the frequency interference in the network environment and then combined it with the DFL model to train the network efficiently and reliably. The effectiveness of combining the DFL framework and the mesh networking protocol results in a comprehensive improvement in the model performance, and the DFL approach is becoming very competitive compared to CFL.

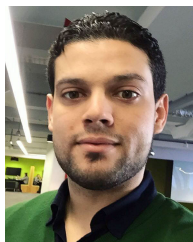
The following is a summary of this research and future work:

- 1) Analysis of the wireless communication stage between the participants during the learning process in order to simulate the real scenarios of interaction between the IoT devices, reducing the communication resources and increasing the system flexibility.
- 2) Implementing the DFL system using the FedAvg algorithm to enforce consensus techniques by sharing local model updates; established gossip methods are also extended by consensus.
- 3) Emphasis on an experimental IoT setup, considering convergence speed, complexity, communication cost, and average prediction accuracy on the DFL embedded devices.
- 4) The CFL and DFL algorithms were implemented by considering the communication stage challenges in the simulation to calibrate the real environment scenarios and the real applications.
- 5) In the future, DFL models need to be designed for high-mobility sensors and devices in the wireless mesh networking system to build a robust system, increase the system flexibility and scalability and enhance the performance for some applications, such as a Driverless Transport System.

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 *Proc. 20th Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1273–1282.
- [2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.
- [3] A. Lalitha, O. C. Kilinc, T. Javidi, and F. Koushanfar, "Peer-to-peer federated learning on graphs," 2019, *arXiv:1901.11173*.
- [4] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl.-Based Syst.*, vol. 216, Mar. 2021, Art. no. 106775. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950705121000381>
- [5] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140699–140725, 2020.
- [6] Z. Eghbali and M. Z. Lighvan, "A hierarchical approach for accelerating IoT data management process based on SDN principles," *J. Netw. Comput. Appl.*, vol. 181, May 2021, Art. no. 103027.
- [7] U. Shafi, R. Mumtaz, J. Garcia-Nieto, S. A. Hassan, S. A. R. Zaidi, and N. Iqbal, "Precision agriculture techniques and practices: From considerations to applications," *Sensors*, vol. 19, no. 17, p. 3796, Sep. 2019.
- [8] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 2031–2063, 3rd Quart., 2020.
- [9] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.
- [10] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, and B. He, "A survey on federated learning systems: Vision, hype and reality for data privacy and protection," *IEEE Trans. Knowl. Data Eng.*, early access, Nov. 2, 2021, doi: [10.1109/TKDE.2021.3124599](https://doi.org/10.1109/TKDE.2021.3124599).
- [11] 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 Netw.*, vol. 33, no. 5, pp. 156–165, Sep. 2019.
- [12] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," 2020, *arXiv:2003.02133*.
- [13] S. A. Rahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A survey on federated learning: The journey from centralized to distributed on-site learning and beyond," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5476–5497, Apr. 2020.
- [14] L. U. Khan, W. Saad, Z. Han, and C. S. Hong, "Dispersed federated learning: Vision, taxonomy, and future directions," *IEEE Wireless Commun.*, vol. 28, no. 5, pp. 192–198, Oct. 2021.
- [15] A. Koloskova, S. Stich, and M. Jaggi, "Decentralized stochastic optimization and gossip algorithms with compressed communication," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3478–3487.
- [16] A. G. Roy, S. Siddiqui, S. Pölsterl, N. Navab, and C. Wachinger, "Brain-Torrent: A peer-to-peer environment for decentralized federated learning," 2019, *arXiv:1905.06731*.
- [17] P. Ramanan and K. Nakayama, "BAFFLE: Blockchain based aggregator free federated learning," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Nov. 2020, pp. 72–81.
- [18] N. J. Dabanloo, S. Moharreri, S. Parvaneh, and A. Nasrabadi, "Application of novel mapping for heart rate phase space and its role in cardiac arrhythmia diagnosis," in *Proc. Comput. Cardiol.*, 2010, pp. 209–212.
- [19] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018.
- [20] S. Savazzi, S. Sigg, F. Vicentini, S. Kianoush, and R. Findling, "On the use of stray wireless signals for sensing: A look beyond 5G for the next generation of industry," *Computer*, vol. 52, no. 7, pp. 25–36, Jul. 2019.
- [21] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Federated learning for ultra-reliable low-latency V2V communications," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7.
- [22] H. Ye, L. Liang, and G. Y. Li, "Decentralized federated learning with unreliable communications," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 3, pp. 487–500, Apr. 2022.
- [23] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.
- [24] Y. Guo, R. Zhao, S. Lai, L. Fan, X. Lei, and G. K. Karagiannidis, "Distributed machine learning for multiuser mobile edge computing systems," *IEEE J. Sel. Topics Signal Process.*, vol. 16, no. 3, pp. 460–473, Apr. 2022.

- [25] V. Mothukuri, R. M. Parizi, S. Pouriye, Y. Huang, A. Dehghantaha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, pp. 619–640, Feb. 2021.
- [26] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, 3rd Quart., 2021.
- [27] Q.-V. Pham, K. Dev, P. K. R. Maddikunta, T. R. Gadekallu, and T. Huynh-The, "Fusion of federated learning and industrial Internet of Things: A survey," 2021, *arXiv:2101.00798*.
- [28] J. Xu, B. S. Glicksberg, C. Su, P. Walker, and J. Bian, "Federated learning for healthcare informatics," *J. Healthc Inform. Res.*, vol. 5, pp. 1–19, Dec. 2021.
- [29] S. Chen, D. Yu, Y. Zou, J. Yu, and X. Cheng, "Decentralized wireless federated learning with differential privacy," *IEEE Trans. Ind. Inform.*, vol. 18, no. 9, pp. 6273–6282, Sep. 2022.
- [30] C. Che, X. Li, C. Chen, X. He, and Z. Zheng, "A decentralized federated learning framework via committee mechanism with convergence guarantee," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 12, pp. 4783–4800, Dec. 2022.
- [31] J. Kang, D. Ye, J. Nie, J. Xiao, X. Deng, S. Wang, Z. Xiong, R. Yu, and D. Niyato, "Blockchain-based federated learning for industrial metaverses: Incentive scheme with optimal AoI," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Aug. 2022, pp. 71–78.
- [32] Google. (2022). *Openthread Protocol*. [Online]. Available: <https://openthread.io/>
- [33] P. Pinyoanuntapong, P. Janakaraj, P. Wang, M. Lee, and C. Chen, "FedAir: Towards multi-hop federated learning over-the-air," in *Proc. IEEE 21st Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, May 2020, pp. 1–5.
- [34] M. Xin and Y. Wang, "Research on image classification model based on deep convolution neural network," *EURASIP J. Image Video Process.*, vol. 2019, no. 1, pp. 1–11, Dec. 2019.
- [35] S. A. R. Zaidi, A. M. Hayajneh, M. Hafeez, and Q. Z. Ahmed, "Unlocking edge intelligence through tiny machine learning (TinyML)," *IEEE Access*, vol. 10, pp. 100867–100877, 2022.
- [36] P. P. Ray, "A review on TinyML: State-of-the-art and prospects," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1595–1623, Apr. 2021.
- [37] H. Zhang, J. Bosch, and H. H. Olsson, "Federated learning systems: Architecture alternatives," in *Proc. 27th Asia-Pacific Softw. Eng. Conf. (APSEC)*, Dec. 2020, pp. 385–394.
- [38] P. Vanhaesebrouck, A. Bellet, and M. Tommasi, "Decentralized collaborative learning of personalized models over networks," in *Proc. 20th Int. Conf. Artif. Intell. Statist.*, 2017, pp. 509–517.
- [39] A. Bellet, R. Guerraoui, M. Taziki, and M. Tommasi, "Personalized and private peer-to-peer machine learning," in *Proc. 21st Int. Conf. Artif. Intell. Statist.*, 2018, pp. 473–481.
- [40] C. He, C. Tan, H. Tang, S. Qiu, and J. Liu, "Central server free federated learning over single-sided trust social networks," 2019, *arXiv:1910.04956*.
- [41] C. Hu, J. Jiang, and Z. Wang, "Decentralized federated learning: A segmented gossip approach," 2019, *arXiv:1908.07782*.
- [42] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 5132–5143.
- [43] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-iid data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, Sep. 2020.
- [44] N. Shoham, T. Avidor, A. Keren, N. Israel, D. Benditkis, L. Mor-Yosef, and I. Zeitak, "Overcoming forgetting in federated learning on non-IID data," 2019, *arXiv:1910.07796*.
- [45] A. Khosravi and Y. S. Kavian, "Broadcast gossip ratio consensus: Asynchronous distributed averaging in strongly connected networks," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 119–129, Jan. 2016.
- [46] A. Neumann, E. López, and L. Navarro, "Evaluation of mesh routing protocols for wireless community networks," *Comput. Netw.*, vol. 93, pp. 308–323, Dec. 2015.
- [47] M. S. Singh and V. Talasila, "A practical evaluation for routing performance of BATMAN-ADV and HWMN in a wireless mesh network test-bed," in *Proc. Int. Conf. Smart Sensors Syst. (IC-SSS)*, Dec. 2015, pp. 1–6.
- [48] Silicon Labs. (2022). *Webinar: Bluetooth Mesh, Thread, and Zigbee Network Performance Benchmarking*. [Online]. Available: <https://www.silabs.com/wireless/multi-protocol/mesh-performance>
- [49] T. Jing, X. Chen, Y. Huo, and X. Cheng, "Achievable transmission capacity of cognitive mesh networks with different media access control," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 1764–1772.
- [50] R. T. Ma, V. Misra, and D. Rubenstein, "An analysis of generalized slotted-ALOHA protocols," *IEEE/ACM Trans. Netw.*, vol. 17, no. 3, pp. 936–949, Jun. 2008.
- [51] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic geometry and random graphs for the analysis and design of wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 7, pp. 1029–1046, Sep. 2009.
- [52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [53] T. Tieleman and G. Hinton, "Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude," in *Proc. COURSERA, Neural Netw. Mach. Learn.*, vol. 4, 2012, pp. 26–31.



ABDELAZIZ SALAMA (Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from Tripoli University, Tripoli, Libya, in 2009, and the M.Sc. degree in communication, control and digital signal processing from the University of Strathclyde, Glasgow, U.K., in 2017. He is currently pursuing the Ph.D. degree with the University of Leeds, Leeds, U.K. He worked for nine years at local and international firms, in several positions, in the areas of telecommunication engineering, and information technology and management. His research interests include federated learning, and autonomous systems and sensing.



ACHILLEAS STERGIOLIS (Member, IEEE) received the M.Eng. degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2020, and the M.Sc. degree in embedded systems engineering from the School of Electrical and Electronic Engineering, University of Leeds, in 2021, where he focused on applying decentralized federated learning in aid of communication systems. As part of his undergraduate diploma dissertation, he carried out research on causal analysis and inference in the presence of missing data. He is currently working as a digital design engineer in the semiconductor industry.



ALI M. HAYAJNEH (Member, IEEE) received the B.Sc. and M.Sc. degrees from the Jordan University of Science and Technology (JUST), Irbid, Jordan, in 2010 and 2014, respectively, and the Ph.D. degree from the University of Leeds, Leeds, U.K. He is currently with the Department of Electrical Engineering, Faculty of Engineering, The Hashemite University, Zarqa, Jordan. He is also the Director of the Innovation and Entrepreneurial Projects Center, The Hashemite University. His

current research is funded by the Royal Academy of Engineering through two programs: 1) "Transfer Systems Through Partnerships (TSP)" and 2) "Distinguished International Associate (DIA)" in the fields of smart agriculture, drone-assisted micro irrigation, and tiny machine learning on edge IoT devices. His current research interests include drone-assisted wireless communications, public safety communication networks, backscatter communication, deep learning, power harvesting, stochastic geometry, device-to-device (D2D) and machine-to-machine (M2M) communications, modeling of heterogeneous networks, cognitive radio networks, cooperative relay networks, edge computing, and reinforcement learning.



SYED ALI RAZA ZAIDI (Senior Member, IEEE) received the Ph.D. degree from the School of Electronic and Electrical Engineering. From 2011 to 2013, he was associated with the International University of Rabat working as a Research Associate. He was also a Visiting Research Scientist at Qatar Innovations and Mobility Centre, from October 2013 to December 2013, working on QNRF funded project QSON. From 2013 to 2015, he was associated with the

SPCOM Research Group working on U.S. ARL-funded project in the area of network science. He is currently an Associate Professor with the University of Leeds in the broad area of communication and sensing for robotics and autonomous systems. He has published more than 90 papers in leading IEEE conferences and journals. He has been awarded COST IC0902, Royal Academy of Engineering, EPSRC, Horizon EU, and DAAD grants to promote his research outputs. His current research interests include ICT, applied mathematics, mobile computing, and embedded systems implementation. Specifically, his current research is geared toward: 1) design and implementation of communication protocols to enable various applications (rehabilitation, healthcare, manufacturing, and surveillance) of future RAS and 2) design, implementation, and control of RAS for enabling future wireless networks (e.g., autonomous deployment and management and repair of future cellular networks). He was awarded the G. W. and F. W. Carter Prize for best thesis and best research paper. From 2014 to 2015, he was an Editor of the IEEE COMMUNICATION LETTERS and a Lead Guest Editor of *IET Signal Processing* Special Issue on Signal Processing for Large Scale 5G Wireless Networks. He is also an Editor of *IET Access*, *Fronthaul*, and *Backhaul* books. He is also serving as an Associate Technical Editor for *IEEE Communications Magazine*.



DES MCLERNON (Member, IEEE) received the B.Sc. degree in electronic and electrical engineering and the M.Sc. degree in electronics from the Queen's University of Belfast, Ireland, and the Ph.D. degree in signal processing from the Imperial College, University of London, U.K. He was working on radar systems research with Ferranti Ltd., Edinburgh, Scotland. He is currently a Reader of signal processing with the University of Leeds, U.K. His research interests include signal processing

for wireless communications. In these fields, he has around 350 research publications and has also supervised over 50 Ph.D. students.



IAN ROBERTSON (Fellow, IEEE) is a Professor of RF and microwave engineering. He has published over 400 peer-reviewed research papers. He edited the book titled *MMIC Design* (IEEE, 1995) and co-edited the book titled *RFIC & MMIC Design and Technology* (published in English in 2001, and in Chinese in 2007). In 2016, he coauthored the book titled *Microwave and Millimetre-Wave Design for Wireless Communications* (Wiley). The funded projects that he has

recently been involved in include "Pervasive Sensing for Buried Pipes," led by The University of Sheffield, which involves studying advanced robotic swarms that can inspect and repair the underground pipe network. His current research interests include microwave and millimeter-wave design for applications in the Internet of Things (IoT), robotics, wearable electronics, biosensors, wireless power transfer, and 5G/6G communications systems. He was elected as a fellow of the IEEE, in 2012, in recognition of his contributions to MMIC design techniques and millimeter-wave system-in-package technology. He was the General Technical Program Committee Chair of the European Microwave Week, in 2011 and 2016.

...