




Article

Wireless Network Optimization for Federated Learning with Model Compression in Hybrid VLC/RF Systems [†]

Wuwei Huang ¹, Yang Yang ^{1,*}, Mingzhe Chen ² , Chuanhong Liu ¹ , Chunyan Feng ¹ and H. Vincent Poor ² 

¹ Beijing Key Laboratory of Network System Architecture and Convergence, School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China; wuweihuang@bupt.edu.cn (W.H.); 2016_liuchuanhong@bupt.edu.cn (C.L.); cyfeng@bupt.edu.cn (C.F.)

² Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ 08544, USA; mingzhec@princeton.edu (M.C.); poor@princeton.edu (H.V.P.)

* Correspondence: yangyang01@bupt.edu.cn

[†] This paper is an extended version of our paper published in the Proceedings of the IEEE Wireless Communications and Networking Conference, Nanjing, China, 29 March–1 April.

Abstract: In this paper, the optimization of network performance to support the deployment of federated learning (FL) is investigated. In particular, in the considered model, each user owns a machine learning (ML) model by training through its own dataset, and then transmits its ML parameters to a base station (BS) which aggregates the ML parameters to obtain a global ML model and transmits it to each user. Due to limited radio frequency (RF) resources, the number of users that participate in FL is restricted. Meanwhile, each user uploading and downloading the FL parameters may increase communication costs thus reducing the number of participating users. To this end, we propose to introduce visible light communication (VLC) as a supplement to RF and use compression methods to reduce the resources needed to transmit FL parameters over wireless links so as to further improve the communication efficiency and simultaneously optimize wireless network through user selection and resource allocation. This user selection and bandwidth allocation problem is formulated as an optimization problem whose goal is to minimize the training loss of FL. We first use a model compression method to reduce the size of FL model parameters that are transmitted over wireless links. Then, the optimization problem is separated into two subproblems. The first subproblem is a user selection problem with a given bandwidth allocation, which is solved by a traversal algorithm. The second subproblem is a bandwidth allocation problem with a given user selection, which is solved by a numerical method. The ultimate user selection and bandwidth allocation are obtained by iteratively compressing the model and solving these two subproblems. Simulation results show that the proposed FL algorithm can improve the accuracy of object recognition by up to 16.7% and improve the number of selected users by up to 68.7%, compared to a conventional FL algorithm using only RF.

Keywords: federated learning; model compression; visible light communication



Citation: Huang, W.; Yang, Y.; Chen, M.; Liu, C.; Feng, C.; Poor, H.V. Wireless Network Optimization for Federated Learning with Model Compression in Hybrid VLC/RF Systems. *Entropy* **2021**, *23*, 1413. <https://doi.org/10.3390/e23111413>

Academic Editor: Vaneet Aggarwal

Received: 27 August 2021

Accepted: 17 October 2021

Published: 27 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Federated learning (FL), which allows edge devices to cooperatively train a shared machine learning model without transmitting private data, is an emerging distributed machine learning technique [1,2]. The FL training process needs to iteratively transmit machine learning parameters over wireless links. However, due to dynamic wireless channels and imperfect wireless transmission, the performance of FL will be significantly affected by wireless communication. In addition, due to limited communication resources, the number of users that can participate in FL is limited.

1.1. Related Work

A number of prior studies in [3–16] have investigated important problems related to wireless network optimization of FL. The works in [3–6] provided a comprehensive survey of existing studies and summarized open problems in FL. One key challenge is the contradiction between the huge communication costs required by FL parameter transmission and the limited available communication resources [5]. Therefore, on one hand, the existing studies in [7–12] proposed to compress the FL model parameters to reduce the communication cost. In particular, the authors of [7] proposed a sparsification and quantization method that compresses the trained FL model. In addition, they also proposed a low rank method and a random mask method, which directly learns a model from a restricted space. In [8], the authors combined quantization and sparsification to implement sketched updates with low sparsity rate. The work in [9] proposed a sparse ternary compression (STC) method that incorporates gradient sparsification, ternary quantization, and lossless encoding, which further improves the compression gains. The authors of [10] proposed to introduce the STC in [9] into structured updates, and thus they focused on compression during the training phase, instead of compressing the trained FL model. Overall, as demonstrated in [7,10], FL model compression can significantly reduce communication costs with minor impact on training accuracy.

On the other hand, the studies in [13–16] proposed to optimize resource allocation to improve communication efficiency in FL. In [13], the authors studied a joint learning, wireless resource allocation, and user selection optimization problem to improve FL performance. In [14], the trade-off of time and energy consumption, and the trade-off of computational and communication delay have been studied. Meanwhile, the authors of [15] optimized a joint computation and transmission problem, whose goal is to minimize the total energy consumption with communication constraints such as limited computational resources and transmission energy. In addition, the authors of [16] formulated an optimization problem of resource allocation and introduced the use of artificial neural networks (ANNs) to predict the unselected users' FL model parameters to improve the FL convergence speed and training loss.

Based on the above research, we can observe that the existing studies focus on improving the communication efficiency either by reducing the transmission data amount or optimizing wireless resource allocation to improve the data rate of each user. As a complement of RF, visible light communication (VLC) has advantages including having large, license-free bandwidth, high energy efficiency, and being free of interference to the RF systems. Introducing VLC into FL can significantly supplement the communication resources for FL training. In addition, model compression can be introduced to further reduce the resources needed to transmit FL model parameters and increase the number of users participating in FL training.

1.2. Contribution

The main contribution of this paper is a novel hybrid VLC/RF FL algorithm, that jointly optimizes user selection, bandwidth allocation, and model compression (USBA-MC). To the best of our knowledge, this is the first work that introduces the use of VLC techniques for FL performance optimization. The contributions are summarized as follows.

- We propose a USBA-MC algorithm over a hybrid VLC/RF system. In the USBA-MC algorithm, each user obtains a local FL model by training through its own dataset and transmits the model parameters to a base station (BS). The BS aggregates the received local models to generate a global FL model and transmits it back to each user. For the considered FL model, the performance is significantly affected by wireless factors such as available bandwidth and users' channel state information. This formulates a joint user selection and bandwidth allocation problem, whose goal is to minimize the FL training loss.
- To solve this problem, we first introduce a model compression method to reduce the size of FL model parameters that are transmitted over wireless links. To this end, we

first sort the model parameters and design a threshold selection mechanism according to the sparsity rate. Then, we cut off the redundant model parameters based on the threshold and, thus, compress an FL model of each user.

- Following the model compression, we separate the joint user selection and bandwidth allocation problem into two subproblems. The first subproblem is a user selection problem with a given bandwidth allocation, which is solved by a traversal algorithm. The second subproblem is a bandwidth allocation problem with a given user selection, which is solved by a numerical method. The ultimate user selection and bandwidth allocation are obtained by iteratively compressing the model and solving these two subproblems.

Simulation results show that the proposed FL algorithm can improve the accuracy of object recognition by up to 16.7% and improve the number of selected users by up to 68.7%, compared to a conventional FL algorithm using only RF.

The remainder of this paper is organized as follows. In Section 2, we introduce the hybrid VLC/RF system model. Section 3 introduces a model compression method. The joint user selection, bandwidth allocation, and model compression algorithm is described in Section 4. Simulation results are presented and discussed in Section 5. Finally, Section 6 draws some important conclusions.

2. System Model and Problem Formulation

In this section, we first introduce a hybrid VLC/RF system for FL. Then, we introduce the computational model and the communication models of RF and VLC systems. Finally, based on the established model, we introduce a user selection and bandwidth allocation problem.

2.1. FL Model

In this model, each user n stores a local dataset \mathcal{D}_n with D_n being the number of training data samples. Therefore, the total number of training data samples of all users is $D = \sum_{n=1}^N D_n$. We assume that the training data samples of user n can be expressed by $\{\mathbf{x}_n, \mathbf{y}_n\}$ with $\mathbf{x}_n = [\mathbf{x}_{n1}, \dots, \mathbf{x}_{nD_n}]$ and $\mathbf{y}_n = [\mathbf{y}_{n1}, \dots, \mathbf{y}_{nD_n}]$, where each \mathbf{x}_{ni} is an input vector of the FL algorithm and \mathbf{y}_{ni} is the output of \mathbf{x}_{ni} .

For each user, the FL training purpose is to find the model parameter ω that minimizes the loss function:

$$J_n(\omega) := \frac{1}{D_n} \sum_{i \in \mathcal{D}_n} f_i(\omega), \quad (1)$$

where $f_i(\omega)$ is a loss function that captures the performance of the FL algorithm. For example, for a linear regression FL, the loss function is $f_i(\omega) = \frac{1}{2}(\mathbf{x}_i^T \omega - \mathbf{y}_i)^2$ [14].

All users aim to minimize the following global loss function:

$$\min_{\omega \in \mathbf{R}^d} J(\omega) := \sum_{n=1}^N \frac{D_n}{D} J_n(\omega). \quad (2)$$

To solve (2), the BS will first transmit the global FL model parameters to its users, and users will use the received global FL model parameters to train their local FL models. Then, the users will transmit their local FL model parameters to the BS to update the global FL model. For strongly convex objective $J(\omega)$, the maximum number of global iterations that an FL algorithm needs to converge is [17]

$$K(\varepsilon, \theta) = \frac{o(\log(1/\varepsilon))}{1 - \theta}, \quad (3)$$

where ε is the accuracy of global model and θ is the accuracy of local model. We consider a fixed global accuracy ε .

2.2. FL Based on Hybrid VLC/RF System

Due to the limited wireless bandwidth, only a subset of users can be selected for FL training, which can seriously degrade the training accuracy. To enable more users to join the FL training process, we design a hybrid VLC/RF system. The system structure is shown in Figure 1.

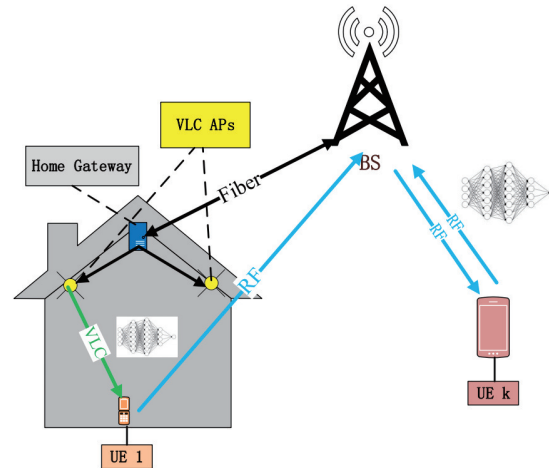


Figure 1. Illustration of FL based on a hybrid VLC/RF system.

The considered system consists of one BS, home gateways, and users cooperatively performing an FL algorithm for data analysis and inference. Denote the total users by a set \mathcal{N} of N users. Denote the indoor users by a set \mathcal{N}_1 of N_1 users and the outdoor users by a set \mathcal{N}_2 of N_2 users. In this model, the BS will send the global FL model parameters to outdoor users by RF. Meanwhile, the BS transmits the global model parameters to the home gateways which are connected to the indoor VLC access points (APs). Then, the VLC APs transmit the global FL model parameters to indoor users through the visible light signal. Assuming that the BS and home gateways are connected by fiber on which bit errors can be negligible.

In indoor scenarios, each VLC AP consists of an LED lamp. Each user is served by the AP that provides the strongest signal. In addition, we assume that all indoor users can be covered by visible lights. We also assume that there is a central unit (CU) which controls both VLC and RF systems. Note that there is no interference between the RF and VLC systems, which is a key benefit of introducing VLC for the deployment of FL over wireless networks.

2.3. Computational Model

Let c_n be the number of CPU cycles for user n to process one sample of data. As the data size of each training data sample is equal, the number of CPU cycles required for user n to execute one local iteration is $c_n D_n$. Denote the CPU-cycle frequency of user n by f_n . Then, the energy consumption of user n updating its local FL model in one global iteration can be expressed as follows:

$$E_n^P = \frac{v \alpha_n c_n D_n}{2} f_n^2 \log(1/\theta), \tag{4}$$

where $n = 1, 2, \dots, N$, $\frac{\alpha_n}{2}$ is the effective capacitance coefficient of the computing chipset of user n , and v is a positive constant that depends on the data size of training data sample and the number of conditions in the local problem [14].

Furthermore, the computational time per local iteration of user n can be denoted as $\frac{c_n D_n}{f_n}$, $n = 1, 2, \dots, N$. The computational time, however, depends on the number of local

iterations, which is upper bounded by $o(\log(1/\theta))$. Therefore, the required computational time of user n for data processing is

$$t_n^P = \frac{vc_n D_n \log(1/\theta)}{f_n}. \tag{5}$$

2.4. RF Transmission Model

We use the orthogonal frequency division multiple access (OFDMA) technique for both uplink and downlink RF transmissions. The uplink rate of user n is given by

$$r_n^U = \sum_{i=1}^{R^U} r_{n,i}^U B^U \log_2 \left(1 + \frac{P_n h_n}{\sum_{i' \in \mathcal{U}'_n} P_{i'} h_{i'} + B^U N_0^{RF}} \right), \tag{6}$$

where $\mathbf{r}_n^U = [r_{n,1}^U, \dots, r_{n,R^U}^U]$ is a resource block allocation vector and R^U is the total number of RBs that the BS can allocate to the users. $r_{n,i}^U \in \{0, 1\}$ and $\sum_{i=1}^{R^U} r_{n,i}^U = 1$; $r_{n,i}^U = 1$ implies that RB i is allocated to user n ; otherwise, we have $r_{n,i}^U = 0$; \mathcal{U}'_n represents the set of users that are located at the other service areas and transmit data over RB i ; B^U is the bandwidth of each RB and P_n is the transmit power of user n ; h_n is the channel gain between user n and the BS; N_0^{RF} is the noise power spectral density; $\sum_{i' \in \mathcal{U}'_n} P_{i'} h_{i'}$ is the interference caused by the users that are located in other service areas and use the same RB.

On the other hand, the downlink data rate of the BS transmitting global FL model parameters to each user n is given by

$$r_n^D = \sum_{i=1}^{R^D} r_{n,i}^D B^D \log_2 \left(1 + \frac{P_B h_n}{\sum_{j \in \mathcal{B}'} P_B h_{nj} + B^D N_0^{RF}} \right), \tag{7}$$

where B^D is the bandwidth of each RB that the BS used to transmit the global FL model to each user n ; $\mathbf{r}_n^D = [r_{n,1}^D, \dots, r_{n,R^D}^D]$ is a RB allocation vector with R^D being the total number of RBs that the BS can be used for FL parameter transmission. $r_{n,i}^D \in \{0, 1\}$ and $\sum_{i=1}^{R^D} r_{n,i}^D = 1$; $r_{n,i}^D = 1$ indicates that RB i is allocated to user n ; otherwise, we have $r_{n,i}^D = 0$; P_B is the transmit power of the BS; \mathcal{B}' is the set of other BSs that cause interference to the BS that performs the FL algorithm; h_{nj} is the channel gain between user n and BS j . Let B_R be the total RF bandwidth, and we have $R^U \times B^U + R^D \times B^D \leq B_R$. For simplicity, we assume $B^U = B^D$ which means the bandwidth of an uplink resource block is equal to that of a downlink RB.

Denote the data size in bit of an FL model that each user needs to upload by s^L . To upload the local FL model within transmission delay requirement t_n^U , we have $t_n^U r_n^U \geq s^L$. Meanwhile, the required energy of user n transmitting FL parameters is $E_n^M = t_n^U P_n$. Similarly, we assume that the data size in bit of the global parameters which are transmitted to users is s^G . To download the global FL model within transmission delay t_n^D , we have $t_n^D r_n^D \geq s^G$.

2.5. VLC Transmission Model

The optical channel gain of a line-of-sight (LoS) channel can be expressed as [18]

$$u = \begin{cases} \frac{(m+1)A_p}{2\pi d^2} T_s(\theta) g(\theta) \cos^m(\varphi) \cos(\theta), & 0 < \theta \leq \Theta_F, \\ 0, & \theta > \Theta_F, \end{cases} \tag{8}$$

where $m = -\frac{1}{\log_2(\cos(\theta_{1/2}))}$ is the Lambertian index which is a function of the half-intensity radiation angle $\theta_{1/2}$, A_p is the receiver’s physical area of the photo-diode, d is the distance from the VLC AP to the optical receiver, φ is the angle of irradiation and θ is the angle of incidence, Θ_F is the half angle of the receiver’s file of view (FoV), $T_s(\theta)$ is the gain of the optical filter, and the concentrator gain $g(\theta)$ can be written as

$$g(\theta) = \begin{cases} \frac{n_0^2}{\sin^2\Theta_F}, & 0 < \theta \leq \Theta_F, \\ 0, & \theta > \Theta_F, \end{cases} \tag{9}$$

where n_0 is the refractive index. For a given user n connected to a VLC AP k , the signal-to-interference-plus-noise ratio (SINR) can be written as

$$s_{nk} = \frac{(\gamma u_{nk} P_v)^2}{N_0^{VLC} B + \sum_{l \neq k} (\gamma u_{nl} P_v)^2}, \tag{10}$$

where γ is the optical to electric conversion efficiency, P_v is the transmitted optical power of a VLC AP, N_0^{VLC} is the noise power spectral density, u_{nk} is the channel gain between user n and the VLC AP k , u_{nl} is the channel gain between user n and the interfering VLC AP l , and B is the bandwidth of each VLC RB. Each user is served by a single VLC AP which has the largest SINR for the user. In the VLC systems, optical OFDMA is employed. It is known that the input signal of the LEDs is amplitude constrained. Therefore, the classical Shannon capacity formula for complex and average power constrained signal is not applicable in VLC. Therefore, the lower bound of achievable data rate is used, which can be expressed as [19]

$$r_n = \sum_{i=1}^{R^V} r_{n,i}^V \frac{B}{2} \log_2\left(1 + \frac{2}{\pi e} s_n\right), \tag{11}$$

where s_n is the largest SINR which is evaluated as $s_n = \max\{s_{n1}, \dots, s_{nK}\}$, where K is the total number of VLC APs; $\mathbf{r}_n^V = [r_{n,1}^V, \dots, r_{n,R^V}^V]$ is a RB allocation vector with R^V being the total number of VLC RBs, $r_{n,i}^V \in \{0, 1\}$ and $\sum_{i=1}^{R^V} r_{n,i}^V = 1$; $r_{n,i}^V = 1$ indicates that RB i is allocated to user n ; otherwise, we have $r_{n,i}^V = 0$. Similarly, we have: $R^V \times B \leq B_V$, where B_V is the total bandwidth of VLC.

As the data size of global parameters is s^G , the downlink transmission delay of indoor user n in each global iteration will be $t_{dn} = \frac{s^G}{r_n}$.

2.6. Problem Formulation

Next, we introduce the optimization problem. Our goal is to minimize the global loss function under time, energy, and bandwidth allocation constraints. The minimization problem is given by

$$\min_{B, B^D, B^U, \mathcal{S}} J(\omega) \tag{12}$$

$$\text{s. t. } R^U \times B^U + R^D \times B^D \leq B_R, \tag{12a}$$

$$R^V \times B \leq B_V, \tag{12b}$$

$$t_{dn} + t_n^U + t_n^P + t_d \leq T_{round}, \forall n \in \mathcal{S}_1, \tag{12c}$$

$$t_n^D + t_n^U + t_n^P \leq T_{round}, \forall n \in \mathcal{S}_2, \tag{12d}$$

$$\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}, \tag{12e}$$

$$E_n^M + E_n^P \leq \gamma_n E, \forall n \in \mathcal{N}, \tag{12f}$$

$$R^U = |\mathcal{S}|, R^D = |\mathcal{S}_2|, R^V = |\mathcal{S}_1|, \tag{12g}$$

where \mathcal{S} denotes the set of selected users participating in FL, \mathcal{S}_1 denotes the set of selected indoor users, \mathcal{S}_2 denotes the set of selected outdoor users, and $|\cdot|$ denotes the cardinality of a set. In addition, T_{round} is the time threshold for each round and t_d denotes the delay between BS and the home gateway. In addition, γ_{nE} is the energy constraint of user n . (12a) and (12b) are the bandwidth constraints of RF link and VLC link, respectively. Constraint (12c) is the delay constraint of each round for all selected indoor users while (12d) is the delay constraint of each round for all selected outdoor users. (12e) denotes the set of selected users. In addition, (12f) is the energy consumption requirement of performing an FL algorithm.

3. Model Compression

In this section, we first analyze the optimization problem (12) so as to figure out how the communication factors affect the FL performance. Then, we introduce a model compression method to reduce the size of FL model parameters that are transmitted over wireless links so as to increase the number of users that participate in FL.

3.1. Problem Analysis

To simplify problem (12), we first provide the following lemma:

Lemma 1. *Given the transmit power of each user, the optimization problem (12) can be transformed into an optimization problem aiming to maximize the total size of data samples of the selected users, which can be denoted as*

$$\max_{B, B^D, B^U, \mathcal{S}} \sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n r_{n,i}^V + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n r_{n,i}^D \tag{13}$$

$$\text{s. t. } R^U \times B^U + R^D \times B^D \leq B_R, \tag{13a}$$

$$R^V \times B \leq B_V, \tag{13b}$$

$$t_{dn} + t_n^U + t_n^P + t_d \leq T_{round}, \forall n \in \mathcal{S}_1, \tag{13c}$$

$$t_n^D + t_n^U + t_n^P \leq T_{round}, \forall n \in \mathcal{S}_2, \tag{13d}$$

$$\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}, \tag{13e}$$

$$E_n^M + E_n^P \leq \gamma_{nE}, \forall n \in \mathcal{N}, \tag{13f}$$

$$R^U = |\mathcal{S}|, R^D = |\mathcal{S}_2|, R^V = |\mathcal{S}_1|, \tag{13g}$$

Proof. Minimizing the global loss function is equivalent to minimizing the gap between the global loss function $J(\omega_t)$ at time t and the optimal global loss function $J(\omega^*)$. According to the Theorem 1 in [13], the gap is caused by the packet error rate (PER) and the number of selected users. Here, we do not consider the packet errors and hence, we have $q_i = 0$. Using the same simplification method in [13], the optimization problem can be transformed to problem (13). This ends the proof. \square

3.2. Model Weights Compression

From problem (13), we observe that as the number of users that implement FL increases, the gap decreases, and the performance is improved. This is coincide with the experimental conclusions in [20]. To maximize the number of users in FL, we introduce a model compression method to reduce the transmission delay, energy, and bandwidth, so as to increase the number of users that participate in FL. In particular, the FL model has data redundancy during training and, thus, we prune the connections with small weight updates to reduce the size of transmission model parameters. Meanwhile, although the model compression will lose a part of model information, the experiments in [9,10,21] have proved that appropriate compression methods do not significantly affect the convergence speed and accuracy under proper sparsity rates. In this section, we first introduce a com-

pression method with non-fixed thresholds. Then, we analyze the impact of the model compression on the optimization problem (13).

An FL model needs to be carefully compressed without affecting the global model training. The change of weights in a model can be used to evaluate their importance [22]. Therefore, an appropriate pruning threshold is the key for FL model compression. To ensure that the gradients of an FL model are in the same order, we first normalize the gradients in each layer [23]. In particular, the gradients of an FL model can be given by

$$\mathbf{G}_n^\tau = \text{Train}(\mathbf{W}_n^\tau, D_n) - \mathbf{W}_n^\tau, \quad (14)$$

where $\mathbf{G}_n^\tau \in \mathcal{R}^{d_1 \times d_2}$ is the gradients of user n at iteration τ , $\mathbf{W}_n^\tau \in \mathcal{R}^{d_1 \times d_2}$ is the trained local model weights, and $\tau \in \{1, \dots, T\}$ is a global iteration; d_1 and d_2 represent the output and input dimensions, respectively; and $\text{Train}(\mathbf{W}_n^\tau, D_n)$ refers to the trained model weights of user n . For a given sparsity rate, we obtain a threshold according to the sorted gradients. In particular, the weights less than the threshold are set to 0, while those larger than the threshold are set to 1. This process can be expressed by a sparsifying filter mask $\mathbf{M}_n \in \mathcal{R}^{d_1 \times d_2}$ for user n . Therefore, the compressed local model weights can be written as

$$\mathbf{W}_{n,C} = \mathbf{W}_n \otimes \mathbf{M}_n, \quad (15)$$

where $\mathbf{W}_n \in \mathcal{R}^{d_1 \times d_2}$ is the local model weights of user n and \otimes is the Hadamard product. Similarly, the compressed global model weights can be expressed as

$$\mathbf{W}_C = \mathbf{W} \otimes \mathbf{M}, \quad (16)$$

where $\mathbf{W} \in \mathcal{R}^{d_1 \times d_2}$ is the global model weights and $\mathbf{M} \in \mathcal{R}^{d_1 \times d_2}$ is the sparsifying filter mask for the BS. From (16), we observe that each user receives the same sparse global model.

The gradients that are not transmitted to the BS or the users are called residuals [24], which will be used for the local model training and the global FL model generation. Therefore, residuals can be used to mitigate the errors caused by the sparsification and accelerate the FL convergence speed [21]. In particular, the residuals of user n can be defined by

$$\mathbf{R}_n^T = \sum_{\tau=1}^T (\mathbf{G}_n^\tau - \mathbf{G}_{n,C}^\tau) = \mathbf{R}_n^{T-1} + \mathbf{G}_n^T - \mathbf{G}_{n,C}^T, \quad (17)$$

where $\mathbf{R}_n^T \in \mathcal{R}^{d_1 \times d_2}$ is the accumulation of the residuals at iteration T , and $\mathbf{G}_{n,C}^\tau \in \mathcal{R}^{d_1 \times d_2}$ denotes compressed \mathbf{G}_n^τ . Similarly, residuals of the BS can be defined by

$$\mathbf{R}^T = \sum_{\tau=1}^T (\mathbf{G}^\tau - \mathbf{G}_C^\tau) = \mathbf{R}^{T-1} + \mathbf{G}^T - \mathbf{G}_C^T, \quad (18)$$

where $\mathbf{G}^\tau \in \mathcal{R}^{d_1 \times d_2}$ is the model gradients at the BS, and $\mathbf{G}_C^\tau \in \mathcal{R}^{d_1 \times d_2}$ is the compressed \mathbf{G}^τ .

During transmissions, users only need to transmit the positions of non-zero parameters and their values. Through receiving these information, the BS can recover the model, and get the sparsifying filter masks. We assume the initial model weights is $\mathbf{W}^I \in \mathcal{R}^{d_1 \times d_2}$, the final output is $\mathbf{W}^F \in \mathcal{R}^{d_1 \times d_2}$ and the matrix with all elements of 1 is $\mathbf{1}$. The overall process of model compression is shown in Algorithm 1.

Let p_n and p be the sparsity rate corresponding to \mathbf{M}_n and \mathbf{M} . Then, the size of the compressed local FL model and the compressed global FL model can be expressed as $s_C^L = s^L \cdot p_n$ and $s_C^G = s^G \cdot p$, respectively. Using the proposed compression scheme, the optimization problem (13) can be rewritten by

$$\begin{aligned} & \max_{B, B^D, B^U, \mathcal{S}} \sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n r_{n,i}^V + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n r_{n,i}^D & (19) \\ \text{s. t. } & R^U \times B^U + R^D \times B^D \leq B_R, & (19a) \\ & R^V \times B \leq B_V, & (19b) \\ & t_{dn,C} + t_{n,C}^U + t_n^P + t_{d,C} \leq T_{round}, \forall n \in \mathcal{S}_1, & (19c) \\ & t_{n,C}^D + t_{n,C}^U + t_n^P \leq T_{round}, \forall n \in \mathcal{S}_2, & (19d) \\ & \mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}, & (19e) \\ & E_{n,C}^M + E_n^P \leq \gamma_{nE}, \forall n \in \mathcal{N}, & (19f) \\ & R^U = |\mathcal{S}|, R^D = |\mathcal{S}_2|, R^V = |\mathcal{S}_1|, & (19g) \end{aligned}$$

where $t_{dn,C} = \frac{s_C^G}{r_n}$, $t_{n,C}^U = \frac{s_C^L}{r_n^U}$, $t_{n,C}^D = \frac{s_C^G}{r_n^D}$ and $E_{n,C}^M = t_{n,C}^U \cdot P_n$. Accordingly, we denote the compression algorithm that reduces the communication costs in each iteration by MC (s^L, s^G).

Algorithm 1 : FL Model Compression

```

1: Input:  $\mathbf{W}^I$ 
2: for  $\tau \in \{1, \dots, T\}$  do
3:   for  $n \in \{1, \dots, N\}$  do
4:     Client n does:
5:      $(\mathbf{W}_C^{t-1}, \mathbf{M}) \leftarrow \text{Download}_{BS \rightarrow n}(\mathbf{W}_C^{t-1})$ 
6:      $\mathbf{W}_n^t \leftarrow \mathbf{W}_C^{t-1} + (\mathbf{1} - \mathbf{M}) \otimes \mathbf{W}_n^{t-1}$ 
7:      $\mathbf{G}_n^t \leftarrow \text{Train}(\mathbf{W}_n^t, D_n) + \mathbf{R}_n^{t-1} - \mathbf{W}_n^t$ 
8:      $\mathbf{M}_n \leftarrow \text{Compress}(\mathbf{G}_n^t)$ 
9:      $\mathbf{G}_{n,C}^t \leftarrow \mathbf{G}_n^t \otimes \mathbf{M}_n$ 
10:     $\mathbf{W}_{n,C}^t \leftarrow \mathbf{W}_n^t \otimes \mathbf{M}_n$ 
11:     $\mathbf{R}_n^t \leftarrow \mathbf{G}_n^t - \mathbf{G}_{n,C}^t$ 
12:     $\text{Save}(\mathbf{R}_n^t, \mathbf{W}_n^t)$ 
13:     $\text{Upload}_{n \rightarrow BS}(\mathbf{W}_{n,C}^t)$ 
14:   end for
15:   BS does:
16:   for  $n \in \mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$  do
17:      $(\mathbf{W}_{n,C}^t, \mathbf{M}_n) \leftarrow \mathbf{W}_{n,C}^t$ 
18:   end for
19:    $\mathbf{W}_C^t \leftarrow \text{Aggregate}(\frac{1}{|\mathcal{S}|} \sum_{n \in \mathcal{S}} \mathbf{W}_{n,C}^t) + \mathbf{R}^{t-1}$ 
20:    $\mathbf{W}^t \leftarrow \mathbf{W}_C^t + (\mathbf{1} - \mathbf{M}_1) \otimes (\mathbf{1} - \mathbf{M}_2) \cdots \otimes (\mathbf{1} - \mathbf{M}_n) \otimes \mathbf{W}^{t-1}$ 
21:    $\mathbf{G}^t \leftarrow \mathbf{W}^t - \mathbf{W}^{t-1}$ 
22:    $\mathbf{M} \leftarrow \text{Compress}(\mathbf{G}^t)$ 
23:    $\mathbf{G}_C^t \leftarrow \mathbf{G}^t \otimes \mathbf{M}$ 
24:    $\mathbf{W}_C^t \leftarrow \mathbf{W}^t \otimes \mathbf{M}$ 
25:    $\mathbf{R}^t \leftarrow \mathbf{G}^t - \mathbf{G}_C^t$ 
26:    $\text{Save}(\mathbf{R}^t, \mathbf{W}^t)$ 
27:    $\text{Transmits}_{BS \rightarrow n}(\mathbf{W}_C^t)$ 
28: end for
29: Return  $\mathbf{W}^F$ 

```

4. The Proposed Algorithm

To solve (19), in this section, we propose a joint user selection, bandwidth allocation, and model compression algorithm, USBA-MC, which divides problem (19) into two sub-problems and solve them iteratively. In particular, we first fix the bandwidth allocation and optimize user selection. Then, the problem of bandwidth allocation is formulated and

solved with the obtained subset of the selected users. The model compression and these subproblems are iteratively solved until a convergent solution of (19) is obtained.

4.1. Optimal User Selection

Given the bandwidth of each RB, (19) can be further simplified as

$$\max_{\mathcal{S}} \sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n r_{n,i}^V + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n r_{n,i}^D \tag{20}$$

$$\text{s. t. } t_{dn,C} + t_{n,C}^U + t_n^P + t_{d,C} \leq T_{round}, \forall n \in \mathcal{S}_1, \tag{20a}$$

$$t_{n,C}^D + t_{n,C}^U + t_n^P \leq T_{round}, \forall n \in \mathcal{S}_2, \tag{20b}$$

$$\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}, \tag{20c}$$

$$E_{n,C}^M + E_n^P \leq \gamma_{nE}, \forall n \in \mathcal{N}, \tag{20d}$$

$$R^U = |\mathcal{S}|, R^D = |\mathcal{S}_2|, R^V = |\mathcal{S}_1|. \tag{20e}$$

We can observe from (20) that if the bandwidth of each RB is fixed, the subset of selected users is determined by the users' computing power and channel condition. We denote the algorithm that optimizes user selection under fixed bandwidth allocation by Algorithm 2.

Algorithm 2 : User Selection Algorithm $\text{GetS}(B^U, B^D, B)$

```

1: Input:  $\mathcal{N}_1, \mathcal{N}_2$ 
2: for  $n \in \mathcal{N}_1$  do
3:   if  $t_{dn,C} + t_{n,C}^U + t_n^P + t_{d,C} \leq T_{round}$ , and  $E_{n,C}^M + E_n^P \leq \gamma_{nE}$  then
4:      $\mathcal{S}_1 \leftarrow n$ 
5:   end if
6: end for
7: for  $n \in \mathcal{N}_2$  do
8:   if  $t_{n,C}^D + t_{n,C}^U + t_n^P \leq T_{round}$ , and  $E_{n,C}^M + E_n^P \leq \gamma_{nE}$  then
9:      $\mathcal{S}_2 \leftarrow n$ 
10:  end if
11: end for

```

4.2. Optimal RB Bandwidth

With an obtained subset of users, we need to find the optimal B , B^U , and B^D that can further optimize the capability of the hybrid VLC/RF systems. Note that the larger the bandwidth of a RB is, the smaller the delay can be, implying more users can be potentially selected. Based on this observation, the optimal RB bandwidth allocation problems are

$$\max B^U \tag{21}$$

$$\text{s. t. } R^U \times B^U + R^D \times B^D \leq B_R, \tag{21a}$$

$$B^U = B^D, \tag{21b}$$

$$R^U = |\mathcal{S}|, R^D = |\mathcal{S}_2|, \tag{21c}$$

and

$$\max B \tag{22}$$

$$\text{s.t. } R^V \times B \leq B_V, \tag{22a}$$

$$R^V = |\mathcal{S}_1|. \tag{22b}$$

Lemma 2. *The maximum bandwidth of a RB can be obtained when $R^U \times B^U + R^D \times B^D = B_R$ and $R^V \times B = B_V$.*

Proof. We use the contradiction method to prove *Lemma 2*. First, we assume that maximum B_0^U , B_0^D , and B_0 exist when (21a) and (22a) are not equal. Therefore, we have

$$B_0^U = B_0^D < \frac{B_R}{|S| + |S_2|}, \quad (23)$$

and

$$B_0 < \frac{B_V}{|S_1|}. \quad (24)$$

However, when (21a) and (22a) are equal, B_1^U , B_1^D , and B_1 satisfy the following equations:

$$B_1^U = B_1^D = \frac{B_R}{|S| + |S_2|}, \quad (25)$$

and

$$B_1 = \frac{B_V}{|S_1|}. \quad (26)$$

Obviously, $B_1^U = B_1^D > B_0^U = B_0^D$ and $B_1 > B_0$, which contradicts the assumption. This ends the proof. \square

Therefore, we have

$$B^U = B^D = \frac{B_R}{R^U + R^D} = \frac{B_R}{|S| + |S_2|}, \quad (27)$$

and

$$B = \frac{B_V}{R^V} = \frac{B_V}{|S_1|}. \quad (28)$$

4.3. Iterative Solution

In each iteration, we first use the proposed model compression method to reduce the transmission delay and energy. Then, we update the selected users based on the constraints, using $\mathbf{GetS}(B^U, B^D, B)$. Finally, the bandwidth allocation is obtained by the given selected users, which is denoted by $\mathbf{GetB}(S)$. The iteration ends when both the user selection and bandwidth allocation remain fixed. Obviously, the algorithm can always reach convergence after a certain number of iterations. We summarize the proposed USBA-MC algorithm in Algorithm 3.

Algorithm 3 : USBA-MC Algorithm

```

1: Input:  $B_0, B_0^D, B_0^U$ 
2:  $(t_{n,C}^U, t_{n,C}^D, t_{dn,C}, E_{n,C}^{EM}) \leftarrow \mathbf{MC}(s^L, s^G)$ 
3:  $S^0 \leftarrow \mathbf{GetS}(B_0^U, B_0^D, B_0)$ 
4: for  $\tau \in \{1, \dots, T\}$  do
5:    $(B_\tau^U, B_\tau^D, B_\tau) \leftarrow \mathbf{GetB}(S^{\tau-1})$ 
6:    $(t_{n,C}^U, t_{n,C}^D, t_{dn,C}, E_{n,C}^{com}) \leftarrow \mathbf{MC}(s^L, s^G)$ 
7:    $S^\tau \leftarrow \mathbf{GetS}(B_\tau^U, B_\tau^D, B_\tau)$ 
8:   if  $S^\tau == S^{\tau-1}$  and  $(B_\tau^U, B_\tau^D, B_\tau) == (B_{\tau-1}^U, B_{\tau-1}^D, B_{\tau-1})$  then
9:     break
10:  end if
11: end for

```

4.4. Convergence, Implementation, and Complexity Analysis

(1) *Convergence Analysis:* We first analyze the convergence of the proposed algorithm. Let the indoor user selection vector be $\mathbf{s}_1 = [s_1^1, \dots, s_1^N]$ and outdoor user selection vector be $\mathbf{s}_2 = [s_2^1, \dots, s_2^N]$, where $s_1^n = 1/s_2^n = 1$ indicates user n performs the FL algorithm; otherwise, we have $s_1^n = 0/s_2^n = 0$. Assume that the gradient $\nabla J(\omega(\mathbf{s}_1, \mathbf{s}_2))$ of $J(\omega(\mathbf{s}_1, \mathbf{s}_2))$ is uniformly Lipschitz continuous with respect to $\omega(\mathbf{s}_1, \mathbf{s}_2)$ [25]. Therefore, we have

$$\|\nabla J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - \nabla J(\omega_t(\mathbf{s}_1, \mathbf{s}_2))\| \leq L\|\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2) - \omega_t(\mathbf{s}_1, \mathbf{s}_2)\|, \tag{29}$$

where $\omega_t(\mathbf{s}_1, \mathbf{s}_2)$ is the global model at step t , L is a positive constant, and $\|\cdot\|$ denotes the norm. Assume that $J(\omega(\mathbf{s}_1, \mathbf{s}_2))$ is strongly convex with positive parameter μ . Therefore, we have

$$J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) \geq J(\omega_t(\mathbf{s}_1, \mathbf{s}_2)) + (\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2) - \omega_t(\mathbf{s}_1, \mathbf{s}_2))^T \nabla J(\omega_t(\mathbf{s}_1, \mathbf{s}_2)) + \frac{\mu}{2}\|\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2) - \omega_t(\mathbf{s}_1, \mathbf{s}_2)\|^2. \tag{30}$$

We also assume that $J(\omega(\mathbf{s}_1, \mathbf{s}_2))$ is twice-continuously differentiable. Moreover, we assume $\|\nabla^2 f_i(\omega_t(\mathbf{s}_1, \mathbf{s}_2))\| \leq \vartheta_1 + \vartheta_2 \nabla \|J(\omega_t(\mathbf{s}_1, \mathbf{s}_2))\|^2$ with $\vartheta_1 \geq 0$ and $\vartheta_2 \geq 0$. The above assumptions are easy to satisfy, such as the loss function that is linear or logistic regression [25]. The expected convergence rate of the proposed algorithm can be obtained by the following lemma:

Lemma 3. *Given the optimal global FL model ω^* . The convergent upper bound of $\mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)]$ applicable to the considered hybrid VLC/RF system satisfies*

$$\begin{aligned} \mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)] &\leq \frac{2\vartheta_1}{LD} \left(\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) \right. \\ &\quad \left. + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D) \right) \frac{1}{1-F}, \end{aligned} \tag{31}$$

where $F = 1 - \frac{\mu}{L} + \frac{4\mu\vartheta_2}{LD} \left(\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D) \right)$.

Proof. As $s_1^n + s_2^n = \begin{cases} 1, & s_1^n = 1, s_2^n = 0 \quad \text{or} \quad s_1^n = 0, s_2^n = 1 \\ 0, & s_1^n = 0, s_2^n = 0 \end{cases}$, we have $1 - (s_1^n + s_2^n) = \begin{cases} 0, & s_1^n = 1, s_2^n = 0 \quad \text{or} \quad s_1^n = 0, s_2^n = 1 \\ 1, & s_1^n = 0, s_2^n = 0 \end{cases}$. Therefore, we have $1 - (s_1^n + s_2^n) \geq 0$ with $n = [1, \dots, N]$. Then, the upper bound of $\mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)]$ can be obtained according to the Theorem 1 in [13] as

$$\begin{aligned} \mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)] &\leq F^t \mathbb{E}[J(\omega_0) - J(\omega^*)] \\ &\quad + \frac{2\vartheta_1}{LD} \sum_{n=1}^N D_n(1 - (s_1^n + s_2^n)) \frac{1-F^t}{1-F}, \end{aligned} \tag{32}$$

where $F = 1 - \frac{\mu}{L} + \frac{4\mu\vartheta_2}{LD} \sum_{n=1}^N D_n(1 - (s_1^n + s_2^n))$. As each resource block will be assigned to a participated user, the upper bound can be further converted into

$$\begin{aligned} \mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)] &\leq F^t \mathbb{E}[J(\omega_0) - J(\omega^*)] \\ &\quad + \frac{2\vartheta_1}{LD} \left(\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D) \right) \frac{1-F^t}{1-F}, \end{aligned} \tag{33}$$

where $F = 1 - \frac{\mu}{L} + \frac{4\mu\theta_2}{LD} (\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D))$. From (33), we can observe that when $F < 1$, F^t approximates to 0 as t increases. Therefore, $\mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2)) - J(\omega^*)] = \frac{1}{D} (\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D)) \frac{1}{1-F}$ and the FL algorithm converges. When $\frac{4\mu\theta_2}{LD} (\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D)) < \frac{\mu}{L}$, $F < 1$. As $\frac{1}{D} (\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D)) \leq 1$, we only need to let $\theta_2 < \frac{1}{4}$. θ_2 can satisfy this condition, as θ_2 can be any value that satisfies $\theta_2 \geq 0$. This completes the proof. \square

From Lemma 3, we can also observe there is a gap $\frac{2\theta_1}{LD} (\sum_{n=1}^{N_1} \sum_{i=1}^{R^V} D_n(1 - r_{n,i}^V) + \sum_{n=N_1+1}^N \sum_{i=1}^{R^D} D_n(1 - r_{n,i}^D))$ between $\mathbb{E}[J(\omega_{t+1}(\mathbf{s}_1, \mathbf{s}_2))]$ and $\mathbb{E}[J(\omega^*)]$. As the number of participated users increases, the gap decreases. Meanwhile, as the number of users increases, the value of F also decreases, which improves the convergence speed of the FL algorithm.

(2) *Implementation Analysis*: Then, we analyze the implementation of the proposed algorithm. To find the optimal user selection set \mathcal{S} , the BS must first calculate the total delay and the energy consumption $E_{n,C}^M + E_n^P$ of each user. In our system, the total delay includes the RF link delay $t_{n,C}^D + t_{n,C}^U + t_n^P$ and the VLC link delay $t_{dn,C} + t_{n,C}^U + t_n^P + t_{d,C}$. In order to calculate the total delay, the BS must know the model size required by FL algorithm and the computational time. The size of the FL model depends on the learning task and the sparsity rate. Before implementing an FL algorithm, the BS must first transmit the task information and model information to each user and set model sparsity rate. Therefore, the BS will know the FL model size and sparsity rate before training. In order to calculate the energy consumption and the computational time, the BS also needs to know the users' device information such as transmit power and CPU. In an FL algorithm, the BS can learn the device information when users initially connect to the BS. Given the total delay $t_{n,C}^D + t_{n,C}^U + t_n^P$, $t_{dn,C} + t_{n,C}^U + t_n^P + t_{d,C}$, and the energy consumption $E_{n,C}^M + E_n^P$, the BS can compute \mathcal{S}_1 and \mathcal{S}_2 using **GetS**(B^U, B^D, B). Given \mathcal{S}_1 and \mathcal{S}_2 , the BS can compute the bandwidth of each RB using **GetB**(\mathcal{S}). As the function in (20) is linear, the USBA-MC algorithm can determine a user selection set \mathcal{S} to improve FL training loss.

(3) *Complexity Analysis*: With regards to the complexity of the USBA-MC algorithm, we first analyze the complexity of the model compression algorithm. In the model compression, the complexity depends on the number of model parameters. Let W_O be the number of model parameters, the complexity of the model compression algorithm is $\mathcal{O}(W_O \log W_O)$ [26]. Then, we analyze the complexity of the traversal algorithm. Since the total number of users is N , the complexity of the traversal algorithm is $\mathcal{O}(N)$ [27]. In addition, the complexity of the numerical method is $\mathcal{O}(1)$ since we only need to allocate RBs according to the user set. Assume that the number of global iterations is T , and the total complexity of the USBA-MC algorithm can be expressed as $\mathcal{O}(TW_O \log W_O)$.

5. Simulation Results and Analysis

Consider a circular network area having a radius $r = 50$ m with one BS at its center. There are $N = 50$ uniformly distributed users, and 80% of the users are in indoors and 20% of them are in outdoors. The system specifications are summarized in Table 1. The following two baselines are considered: (a) the USBA algorithm in a hybrid VLC/RF system [28] and (b) the FL algorithm in RF-only system. To comprehensively evaluate the performance of the proposed USBA-MC algorithm in federated learning systems, we conduct experiments related to three learning tasks: (a) the prediction task of housing price, (b) identification task of identifying the handwritten digits from 0 to 9, and (c) identification task of classifying 10 categories of color images.

Table 1. Simulation parameters.

Parameter	Value
Transmitted optical power per VLC AP, P_v	9 W
Modulation bandwidth for LED lamp, B	40 MHz
The physical area of a PD, A_p	1 cm ²
Half-intensity radiation angle, $\theta_{1/2}$	60 deg.
Gain of optical filter, $T_s(\theta)$	1.0
Receiver FOV semi-angle, Θ_F	90 deg.
Refractive index, n	1.5
Optical to electric conversion efficiency, γ	0.53 A/W
Noise power spectral density, N_0^{VLC}, N_0^{RF}	10^{-21} A ² /Hz
RF total bandwidth, B_R	20 MHz
Transmit power of BS, P_B	1 W
The number of users, N	50
Delay requirement, T_{round}	2.5 s
Energy consumption requirement, γ_{nE}	2 J
Energy consumption coefficient, α	2×10^{-28}
user update size, s	1 Mb

In the housing price prediction task, our goal is to compare the performance of the proposed USBA-MC algorithm under different sparsity rates, and compare the performance of USBA-MC, baselines (a) and (b). The dataset used to train the FL algorithm is Boston house price dataset (<http://lib.stat.cmu.edu/datasets/boston> (accessed on 27 March 2021)) that is randomly allocated to users equally. In this task, each user trains an FNN with one hidden layer composed of 10 neurons.

In the identification task of handwritten digits, we train FNNs using MNIST dataset [29]. The size of neuron weight matrices are 784×200 , 200×200 , and 200×10 . Sixty-thousand handwritten digits are used to train the network and 10,000 handwritten digits are used to test it.

Finally, we train CNNs on CIFAR-10 [30] to investigate the performance of USBA-MC algorithm with different sparsity rates on non-IID data. The size of neuron weight matrices are $5 \times 5 \times 3 \times 64$, $5 \times 5 \times 64 \times 64$, 2304×384 , 384×192 and 192×10 . Fifty-thousand images are used to train the network and 10,000 images are used to test it.

5.1. Performance Over Different Sparsity Rates

Figure 2 shows the performance of the proposed USBA-MC algorithm in two learning tasks under different sparsity rates. We use the coefficient of determination (R^2) to measure the quality of the model in the task of predicting housing price, and use the accuracy of classification to measure the performance in the task of identifying handwritten digits. Moreover, we calculated the average of 10 experiments to ensure the reliability of the experimental results. It shows that the R^2 values first increase and then decrease with the sparsity rate. This is because the model information will be lost with low sparsity rate. In particular, the best sparsity rate is 0.4 for predicting housing price and 0.2 for identifying handwritten digital.

Figure 3 compares the predictive performance of USBA-MC, baselines (a) and (b). The green line is the true values of data samples, and the sparsity rate of USBA-MC is set to 0.4. Before training, we randomly select 18 samples to form a test set for testing. In Figure 3, we can observe the proposed USBA-MC algorithm can achieve better performance than baselines (a) and (b). In particular, the proposed FL algorithm can improve the R^2 by up to 11% and 15%, compared to baselines (a) and (b).

Figure 4 compares the identification performance in the tasks of identifying handwritten digits. It shows USBA-MC is better than baselines (a) and (b) in most global communication rounds, and the final accuracies of these algorithms are 96.52%, 96.45%, and 96.39%, respectively. This is because USBA-MC introduces visible light communication

and reduce the size of transmission model, which can increase the number of selected users, and further improving the FL performance.

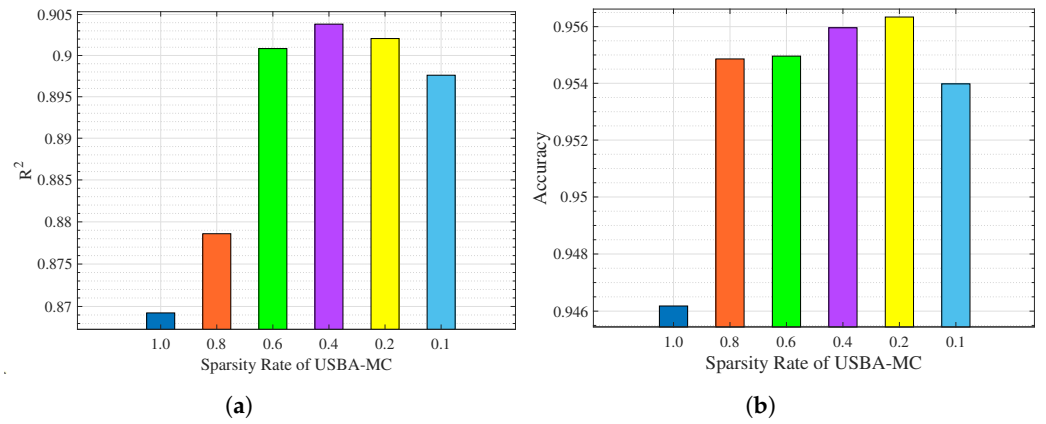


Figure 2. The accuracy achieved by different sparsity rates of USBA-MC in the Boston housing dataset and MNIST dataset. (a) Boston housing dataset. (b) MNIST dataset.

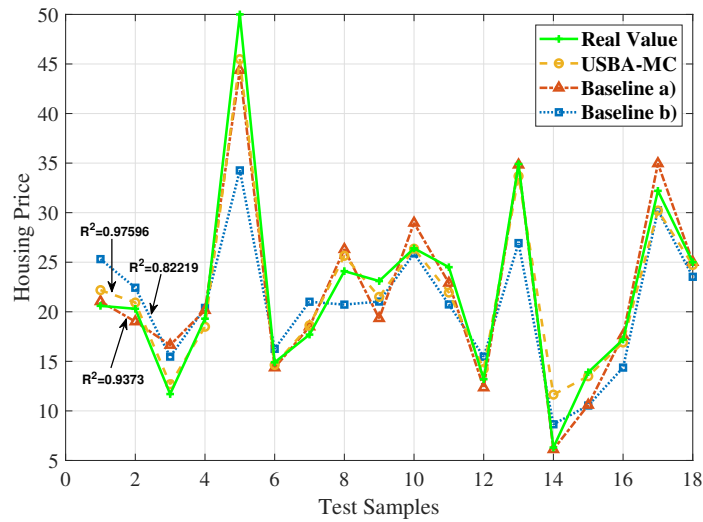


Figure 3. Comparison of accuracy in the Boston housing dataset trained by a BP neural network.

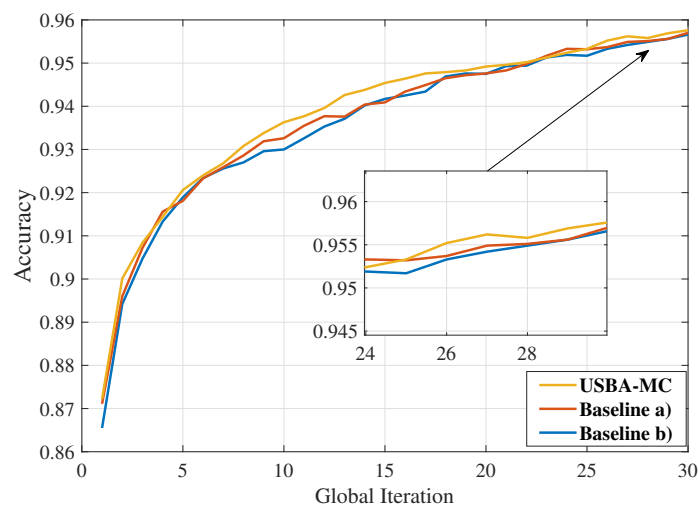


Figure 4. Comparison of accuracy in MNIST dataset trained by BP neural network.

5.2. Number of Selected Users

This subsection evaluates the performance of USBA-MC in user selection. We first compare the number of users selected under different bandwidth conditions and resource constraints.

Figure 5 shows how the number of selected users changes as the total number of users varies in different systems. It can be observed that with the increase of the total users, the selected users in three algorithms will also increase. However, compared with baselines (a) and (b), USBA-MC algorithm enables more users to participate in the training process. This trend is more obvious with the increase of total number. For instance, when the total number is 150, the USBA-MC can improve the number of selected users, by, respectively, up to 37.8% and 68.7% compared to baseline (a) and (b). Table 2 shows the ratio of selected users, we can find that USBA-MC always has the highest ratio in all cases. Figure 5 also compares the user selection under different VLC and RF bandwidths. It can be observed that the proposed USBA-MC algorithm is better than baselines (a) and (b) under all bandwidth settings.

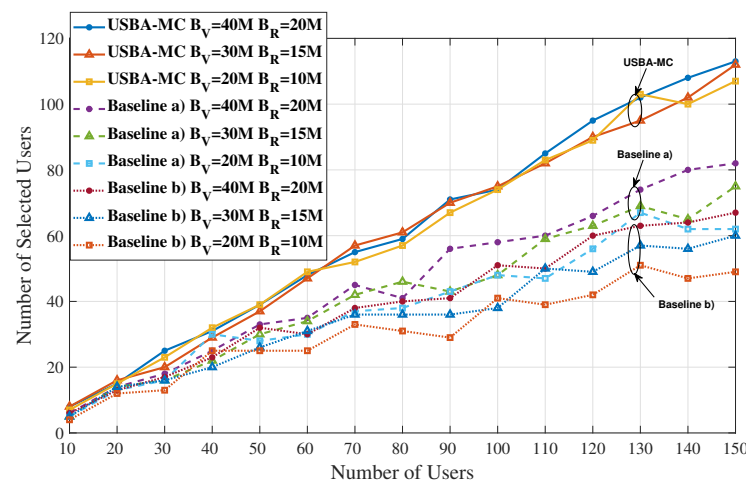


Figure 5. Comparison of user selection under different bandwidth settings with different numbers of users.

Table 2. Selection ratio of users with different total numbers.

Total Number of User	USBA-MC	Baseline (a)	Baseline (b)
50	78%	66%	64%
100	74%	58%	51%
150	75.33%	54.67%	44.67%

Figure 6 compares number of selected users under different transmission data sizes. We can observe that the selected users decrease when the data size increases. Table 3 shows the selection ratio of different algorithms with different data sizes when the total user number is 150. The result shows that USBA-MC can achieve better system performance than the other algorithms. This advantage is important when the model size becomes larger due to the complex neural network.

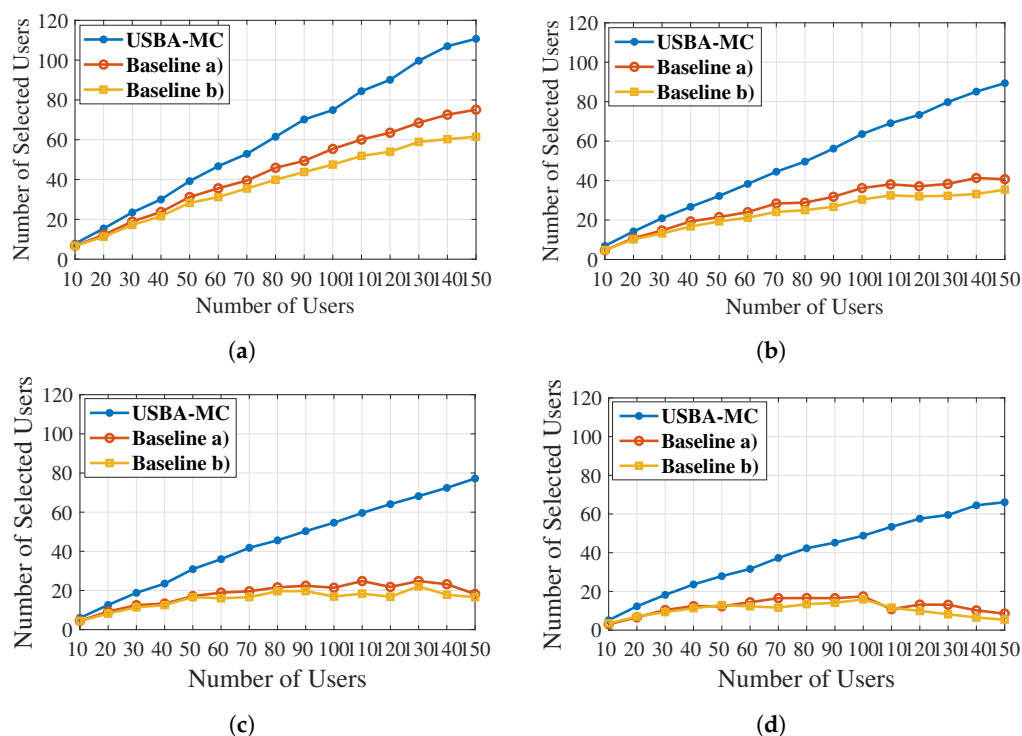


Figure 6. Comparison of user selection under different transmission data sizes with different numbers of users. (a) Data Size of 1 M. (b) Data Size of 3 M. (c) Data Size of 5 M. (d) Data Size of 7 M.

Table 3. Selection ratio of users with different data sizes.

Data Size	USBA-MC	Baseline (a)	Baseline (b)
1 M	73.8%	50.07%	41%
3 M	59.6%	27.13%	23.6%
5 M	51.47%	12.13%	11.07%
7 M	44.07%	5.67%	3.6%

5.3. Non-IID Data

In this subsection, we explore the accuracy of USBA-MC algorithm with non-IID data [31]. To obtain a non-IID dataset, we use the same method as in [31].

As shown in Figure 7, the model is trained on the dataset of non-IID nature. We can clearly observe the advantages of USBA-MC compared with other algorithms. In terms of stability and accuracy, the USBA-MC algorithm achieves the best performance. In USBA-MC, a low sparsity rate will increase the stability of the system and improve the final accuracy. However, when the sparsity rate is 0.2, USBA-MC has lower accuracy and higher stability compared to 0.4. This is because decreasing sparsity rate will increase the loss of model information. Moreover, the model will not converge when the sparsity rate is too low. Therefore, there is a trade-off between the sparsity rate and the model performance.

Figure 8 shows the use of the proposed USBA-MC algorithm for image identification. In this simulation, the BS uses broadcast techniques to transmit the global model and the local models are trained by CIFAR-10. As shown in Figure 8, the proposed USBA-MC algorithm can still achieve the best performance among the three algorithms in terms of both accuracy and stability. In particular, the USBA-MC can improve the accuracy by up to 3.27% and 6.35%, compared to baselines (a) and (b).

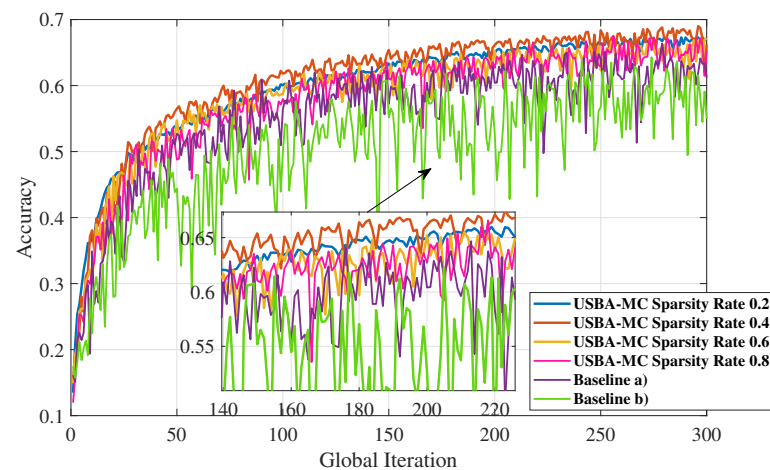


Figure 7. The accuracy of USBA-MC with different sparsity rates, baseline (a) and (b) on CIFAR-10 with non-IID nature.

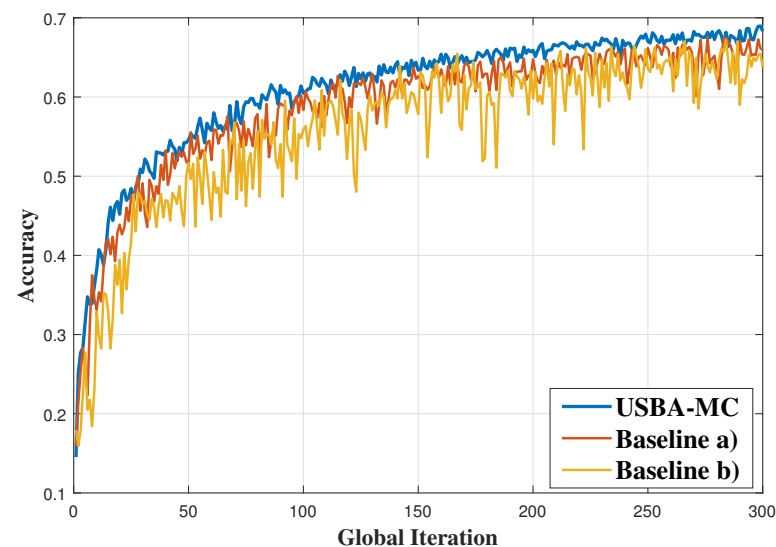


Figure 8. Comparison of accuracy in CIFAR-10 dataset with broadcast channel.

We analyze the gain of USBA-MC when the data set is non-IID. According to Section 3 in [31], the weight divergence will reduce the accuracy in non-IID dataset. The weight divergence is caused by the distance between the data distribution on each user and the population distribution. Such distance can be evaluated with the earth mover's distance (EMD). According to the central limit theorem (CLT) of normal distribution [32], as the number of local models increases, the mean of EMD will be approximated by a normal distribution. Therefore, the weight divergence of the trained global model will be smaller and the model performance will be better with the increase of the number of users. Compared with the accuracy of the transmission model, the system is more sensitive to the number of selected users. Therefore, the increase of users will improve the robustness and accuracy of the global model.

Figure 9 selects the model accuracy of the last 10 global communications to obtain the average and variance of the accuracy. It can be observed that the accuracy first increase and then decrease with the sparsity rate. The USBA-MC can improve the accuracy by up to 7% and 16.7%, compared to baselines (a) and (b) when the sparsity rate is 0.4. We can also observe that as the number of users increases, the model will be more stable until it cannot converge.

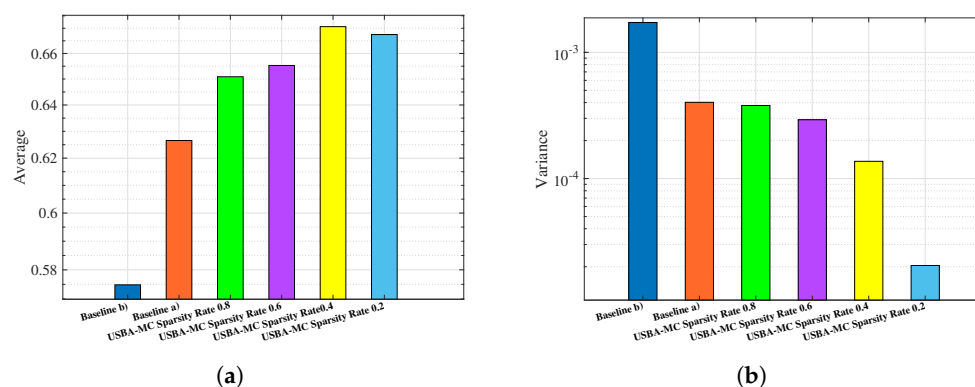


Figure 9. The average and variance of accuracy of USBA-MC with different sparsity rates, baselines. (a) Average of accuracy. (b) Variance of accuracy.

6. Conclusions

This paper has proposed the introduction of VLC into conventional RF systems for supporting FL. We have formulated a joint user selection and bandwidth allocation problem for FL in a hybrid VLC/RF system. To solve this problem, we first used a model compression method to reduce the size of FL model parameters that are transmitted over wireless links, and then we separated the optimization problem into two subproblems. The first subproblem is a user selection problem with a given bandwidth allocation, which is solved by a traversal algorithm. The second subproblem is a bandwidth allocation problem with a given user selection, which is solved by a numerical method. The convergent solution is obtained by iteratively compressing the model and solving these two subproblems. Simulation results have demonstrated that the USBA-MC algorithm outperforms USBA and FL in RF-only systems.

Author Contributions: Conceptualization, W.H., M.C. and Y.Y.; methodology, W.H. and H.V.P.; software, W.H., M.C. and C.L.; validation, W.H., M.C., Y.Y. and C.L.; formal analysis, W.H., H.V.P. and C.L.; investigation, W.H., M.C., Y.Y. and C.L.; resources, H.V.P. and C.F.; data curation, W.H., Y.Y. and C.L.; writing—original draft preparation, W.H.; writing—review and editing, M.C., Y.Y. and H.V.P.; visualization, W.H. and M.C.; supervision, H.V.P. and C.F.; project administration, M.C. and Y.Y.; funding acquisition, H.V.P. and C.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities (2021RC03), National Natural Science Foundation of China (61901047), National Natural Science Foundation of China (61871047), Beijing Natural Science Foundation (4204106), and Proof-of-concept project of Zhongguancun Open Laboratory (202103001).

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found here: <http://lib.stat.cmu.edu/datasets/boston>, <http://yann.lecun.com/exdb/mnist>, and <http://www.cs.toronto.edu/kriz/cifar.html> (accessed on 27 March 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

- McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017.
- Konecny, J.; McMahan, H.B.; Ramage, D.; Richtarik, P. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv* **2016**, arXiv:1610.02527.
- Niknam, S.; Dhillon, H.S.; Reed, J.H. Federated learning for wireless communications: Motivation, opportunities and challenges. *IEEE Commun. Mag.* **2020**, *58*, 46–51. [\[CrossRef\]](#)
- Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063. [\[CrossRef\]](#)

5. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [\[CrossRef\]](#)
6. Chen, M.; Gündüz, D.; Huang, K.; Saad, W.; Bennis, M.; Feljan, A.V.; Poor, H.V. Distributed learning in wireless networks: Recent progress and future challenges. *arXiv* **2021**, arXiv:2104.02151.
7. Konečný, J.; McMahan, H.B.; Yu, F.X.; Richtárik, P.; Suresh, A.T.; Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv* **2016**, arXiv:1610.05492.
8. Tsuzuku, Y.; Imachi, H.; Akiba, T. Variance-based gradient compression for efficient distributed deep learning. *arXiv* **2018**, arXiv:1802.06058.
9. Sattler, F.; Wiedemann, S.; Müller, K.R.; Samek, W. Robust and communication-efficient federated learning from non-iid data. *IEEE Trans. Neural Netw. Learn Syst.* **2019**, *31*, 3400–3413. [\[CrossRef\]](#)
10. Xu, J.; Du, W.; Jin, Y.; He, W.; Cheng, R. Ternary compression for communication-efficient federated learning. *IEEE Trans. Neural Netw. Learn Syst.* **2020**, 1–15. [\[CrossRef\]](#)
11. Shao, R.; Liu, H.; Liu, D. Privacy preserving stochastic channel-based federated learning with neural network pruning. *arXiv* **2019**, arXiv:1910.02115.
12. Chen, M.; Shlezinger, N.; Poor, H.V.; Eldar, Y.C.; Cui, S. Communication-efficient federated learning. *Proc. Nat. Acad. Sci. USA* **2021**, *118*, e2024789118. [\[CrossRef\]](#)
13. Chen, M.; Yang, Z.; Saad, W.; Yin, C.; Poor, H.V.; Cui, S. A joint learning and communications framework for federated learning over wireless networks. *IEEE Trans. Wireless Commun.* **2021**, *20*, 269–283. [\[CrossRef\]](#)
14. Tran, N.H.; Bao, W.; Zomaya, A.; Nguyen, M.N.H.; Hong, C.S. Federated learning over wireless networks: Optimization model design and analysis. In Proceedings of the IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019.
15. Yang, Z.; Chen, M.; Saad, W.; Hong, C.S.; Bahaei, M.S. Energy efficient federated learning over wireless communication networks. *IEEE Trans. Wireless Commun.* **2021**, *20*, 1935–1949. [\[CrossRef\]](#)
16. Chen, M.; Poor, H.V.; Saad, W.; Cui, S. Convergence time optimization for federated learning over wireless networks. *IEEE Trans. Wireless Commun.* **2021**, *20*, 2457–2471. [\[CrossRef\]](#)
17. Ma, C.; Konečný, J.; Jaggi, M.; Smith, V.; Jordan, M.I.; Richtárik, P.; Takáč, M. Distributed optimization with arbitrary local solvers. *Optim. Methods Softw.* **2017**, *32*, 813–848. [\[CrossRef\]](#)
18. Yang, Y.; Zeng, Z.; Cheng, J.; Guo, C.; Feng, C. A relay-assisted OFDM system for VLC uplink transmission. *IEEE Trans. Commun.* **2019**, *67*, 6268–6281. [\[CrossRef\]](#)
19. Pham, T.V.; Pham, A.T. Coordination/cooperation strategies and optimal zero-forcing precoding design for multi-user multi-cell VLC networks. *IEEE Trans. Commun.* **2019**, *67*, 4240–4251. [\[CrossRef\]](#)
20. Hsu, T.H.; Qi, H.; Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv* **2019**, arXiv:1909.06335.
21. Sattler, F.; Wiedemann, S.; Müller, K.R.; Samek, W. Sparse binary compression: Towards distributed deep learning with minimal communication. In Proceedings of the International Joint Conference on Neural Networks, Budapest, Hungary, 14–19 July 2019; pp. 1–8.
22. Xie, H.; Qin, Z. A lite distributed semantic communication system for internet of things. *IEEE J. Sel. Areas Commun.* **2020**, *39*, 142–153. [\[CrossRef\]](#)
23. Aji, A.F.; Heafield, K. Sparse communication for distributed gradient descent. *arXiv* **2017**, arXiv:1704.05021.
24. Lin, Y.; Han, S.; Mao, H.; Wang, Y.; Dally, W.J. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv* **2017**, arXiv:1712.01887.
25. Friedlander, M.P.; Schmidt, M. Hybrid deterministic-stochastic methods for data fitting. *SIAM J. Sci. Comput.* **2012**, *34*, A1380–A1405. [\[CrossRef\]](#)
26. Jiang, Y.; Wang, S.; Valls, V.; Ko, B.J.; Lee, W.H.; Leung, K.K.; Tassiulas, L. Model pruning enables efficient federated learning on edge devices. *arXiv* **2019**, arXiv:1909.12326.
27. Cheung, T.Y. Graph traversal techniques and the maximum flow problem in distributed computation. *IEEE Trans. Softw. Eng.* **1983**, *SE-9*, 504–512. [\[CrossRef\]](#)
28. Liu, C.; Guo, C.; Yang, Y.; Chen, M.; Poor, H.V.; Cui, S. Optimization of user selection and bandwidth allocation for federated learning in VLC/RF systems. In Proceedings of the IEEE Wireless Communications and Networking Conference, Nanjing, China, 29 March–1 April 2021; pp. 1–6.
29. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE Inst. Electr. Electron. Eng.* **1998**, *86*, 2278–2324. [\[CrossRef\]](#)
30. Krizhevsky, A.; Nair, V.; Hinton, G. The CIFAR-10 Dataset. Available online: <http://www.cs.toronto.edu/kriz/cifar.html> (accessed on 27 March 2021).
31. Zhao, Y.; Li, M.; Lai, L.; Suda, N.; Civin, D.; Chandra, V. Federated learning with non-iid data. *arXiv* **2018**, arXiv:1806.00582.
32. Rosenblatt, M. A central limit theorem and a strong mixing condition. *Proc. Nat. Acad. Sci. USA* **1956**, *42*, 43. [\[CrossRef\]](#)