Parallel Computing 120 (2024) 103083

Contents lists available at ScienceDirect

Parallel Computing

journal homepage: www.elsevier.com/locate/parco

Federated learning based modulation classification for multipath channels

Sanjay Bhardwaj, Da-Hye Kim, Dong-Seong Kim [*,1]

ICT Convergence Research Center, Department of IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea

ARTICLE INFO

Keywords:
Federated learning (FL)
Convolutional neural networks (CNN)
Automatic modulation classification (AMC)
Multipath channels
Class imbalance
Classification performance

ABSTRACT

Deep learning (DL)-based automatic modulation classification (AMC) is a primary research field for identifying modulation types. However, traditional DL-based AMC approaches rely on hand-crafted features, which can be time-consuming and may not capture all relevant information in the signal. Additionally, they are centralized solutions that are trained on large amounts of data acquired from local clients and stored on a server, leading to weak performance in terms of correct classification probability. To address these issues, a federated learning (FL)-based AMC approach is proposed, called FL-MP-CNN-AMC, which takes into account the effects of multipath channels (reflected and scattered paths) and considers the use of a modified loss function for solving the class imbalance problem caused by these channels. In addition, hyperparameter tuning and optimization of the loss function are discussed and analyzed to improve the performance of the proposed approach. The classification performance is investigated by considering the effects of interference level, delay spread, scattered and reflected paths, phase offset, and frequency offset. The simulation results show that the proposed approach provides excellent performance in terms of correct classification probability, confusion matrix, convergence and communication overhead when compared to contemporary methods.

1. Introduction

The number of wireless devices is rapidly increasing, and the types of modulation schemes are diversifying as a result of the astounding improvements in wireless communication technology, producing an increasingly complex communication environment. As a result, it becomes increasingly important to be able to identify and categorize communication signals rapidly and automatically. In order to improve spectrum use and management, automatic modulation classification (AMC) has lately been considered a solution to communication problems such as spectrum monitoring and link adaption [1].

According to [2–4], likelihood-based classifiers (LBC) and feature-based classifiers (FBC) make up the majority of traditional modulation classifiers. LBC first calculates the likelihood of the received signals under various modulation assumptions, and then uses the maximum likelihood (ML) to confirm the modulation scheme [5]. Although the LBC methods theoretically ensure that the classification outcome satisfies the Bayes minimum misclassification cost criterion, these AMC methods are challenging to implement quickly and with high performance due to several drawbacks: lack of prior knowledge in non-cooperative communication, sensitivity of model adaptation, and the complexity of the likelihood ratio function with unknown parameters. As opposed to this, FBC extracts and compares properties of the input signals, such as higher-order statistics, to decide the categorization, [6,7].

1.1. Federated learning (FL) key advantages

• The federated learning (FL) framework divides model training among clients, and each client group independently updates the gradient to reflect the learning model based on their local training data. • FL provides data privacy and security, which is another amazing feature due to the fact that only the model parameters are updated during the training process and no original data is transmitted. • FL could be implemented with the use of edge computing tools, given the ongoing development of big data and edge computing. This makes it possible to utilize different types of data at the edge fully without the requirement for a centralized and efficient data center. • Additionally, because the model is trained on the user terminal, no local data is transferred from the local server, which lowers communication latency and communication costs related to initial data transfer.

These factors make FL a popular topic, and several scholars have authored numerous articles about it. Some of them, including but not limited to those specifically related to wireless communication, are discussed. In [8], the FL-based minimization of energy and time consumption for job computation and transmission in mobile-edge computing-enabled balloon networks is examined. For 5G millimeter-wave (mm-wave) and 6G terahertz (THz) scenarios, [9] proposed a

* Corresponding author.
E-mail addresses: sanjay.b1976@gmail.com (S. Bhardwaj), dahem91@kumoh.ac.kr (D.-H. Kim), dskim@kumoh.ac.kr (D.-S. Kim).
[1] Senior Member, IEEE.

https://doi.org/10.1016/j.parco.2024.103083
Received 15 October 2023; Received in revised form 25 February 2024; Accepted 11 March 2024
Available online 16 March 2024

FL-aided decentralized resource allocation approach to maximize the sum capacity and reduce overall power consumption while guaranteeing the quality of service (QoS) requirements of both cellular users and device-to-device (D2D) users. [10] proposed the FL framework for resource allocation in wireless networks with multiple systems. In order to balance the learning accuracy of FL and the energy consumption of the IoT device, a resource allocation challenge for privacy-preserving edge computing-based Internet of Things (EdgeIoT) was put up by [11]. A federated deep reinforcement learning algorithm to coordinate multiple independent x-applications (xAPPs) in an open radio access network (O-RAN) for network slicing was proposed by [12].

GNU Radio comes equipped with a comprehensive set of tools essential for constructing a robust dataset in this domain. It provides a range of modulators, encoders, and demodulators, along with an extensive collection of channel simulation modules specifically crafted for applying simulated channel propagation models to signals generated synthetically. In essence, we seamlessly combine these logical modules within GNU Radio to formulate our dataset [13,14].

Class imbalance refers to a situation where the distribution of samples in the different classes of a dataset is uneven or imbalanced [15]. In the context of modulation classification, class imbalance occurs when the dataset used to train a machine learning model contains an uneven distribution of samples across different modulation schemes. This occurs due to a variety of reasons, such as variations in the signal caused by multipath channels, the use of certain modulation schemes more frequently than others, or errors in the data collection process. Class imbalance can pose a challenge for machine learning algorithms as they may be biased towards the more prevalent class, resulting in poor performance in the minority class. In the context of modulation classification, this can result in inaccurate identification of the modulation scheme used to transmit a signal, which can lead to errors in demodulation and decoding [16]. Thus, class imbalance refers to the situation where the number of samples in different classes is not balanced, thereby leading to biased performance measures and classification results.

1.2. Motivation and contribution

Signal modulation classification is an essential task in wireless communications, which involves identifying the modulation scheme used to transmit a signal. However, in the presence of multipath channels, where a signal takes multiple paths to reach the receiver, the received signal can suffer from class imbalance due to variations in the signal caused by scattered and reflected paths. This can lead to poor classification accuracy and suboptimal performance of machine learning algorithms trained on the dataset. Thus, FL based approach is proposed, called FL-MP-CNN-AMC, that addresses the challenges posed by class imbalance in modulation classification caused by multipath channels. Along with that, instead of a fixed weighting scheme, the assigned weights self-adapt their scale based on the prediction difficulty of the data instance. This approach allows the model to assign greater importance to samples that are more difficult to classify, which can improve the performance of the model on imbalanced datasets.

Moreover, traditional machine learning methods for modulation classification are limited in their ability to handle class imbalance caused by multipath channels. This is because the distribution of samples in the classes is often uneven due to the varying strengths of the different paths taken by the signal. This class imbalance can result in biased models that perform poorly in the minority class. To address this limitation, a FL based approach is proposed that allows the data and deep learning models to be stored and trained locally on clients, while only sharing knowledge learned from the data with the server.

The proposed approach, FL-MP-CNN-AMC, leverages FL in conjunction with a modified focal loss function to automatically learn to extract features from a multipath channel. A modified loss function is formulated to address class imbalance by assigning higher weights to the minority class samples. By doing so, the loss function gives more importance to correctly classifying the minority class, which improves the overall performance of the classifier on imbalanced datasets. Thus it addresses both class imbalance and reliability in terms of the predicted probabilities accurately reflecting the true probabilities of the classes in the dataset. For in-depth analysis of the proposed approach 8 scenarios are considered, with each scenario have different combination of modulation signals percentage wise, thus creating further class imbalance. Along with it, FL-MP-CNN-AMC is also evaluated against different fading models, [17–19] in terms of correct classification probability (CCP).

The proposed approach is extensively evaluated in comparison to traditional and contemporary AMC methods. The results show that the proposed FL-MP-CNN-AMC approach outperforms traditional and contemporary methods in terms of class imbalance and noise robustness. Specifically, the proposed approach not only shows improved performance in terms of interference level, confusion matrices, CCP, convergence and communication overheads, but also when compared with asynchronous FL (AFL) for the metrics: CCP, misclassified samples, convergence and communication overhead, proposed approach out performs it. This highlights the effectiveness of the proposed approach in addressing the challenges posed by class imbalance in modulation classification caused by multipath.

Along with it, the proposed approach is evaluated in terms of CCP against a standard synthetic signal dataset of 12 different modulations, [20], which have the multipath characteristics. In along with it, the proposed approach for in depth evaluation is also evaluation against another standard data set, ML2018.01 A [21] which consists of a wider range of modulation signal with 25 in numbers.

1.3. Overview of contemporary AMC methods used for comparison

(1) **TD-AMC** [22]: This approach analyzes the performance of typical modulations in feature-based approaches to determine which modulation results in higher performance for a specific feature. In these tests, they utilized a support sector machine (SVM) classifier in combination with cyclostationary features and an S-transform. But SVMs are sensitive to the choice of kernel function, computational expense, sensitivity to outliers, and limited applicability to binary classification.

(2) **FL-CNN-AMC** [23]: A CNN-based FL approach that consisted of only 4 clients and a class imbalance training data set ranging from 1000 to 6000 samples was considered. Along with this, balanced class entropy (BCE) is introduced to address the class imbalance. Moreover, a test dataset of 20,000 IQ samples for the SNR range of −10 dB to 10 dB was also prepared. The major drawbacks of their proposed approach are loss of information, sensitivity to class distribution, overfitting, reduced discriminatory power, and limited applicability to multiclass problems.

(3) **FL-DP-CNN-AMC** [24]: A FL-based CNN approach is used for the MC, wherein the number of training devices considered was 10, and the data trained by each device includes all modulation types. But their approach had increased communication overhead, heterogeneous data, limited control over training data, limited availability of devices.

(4) **CNN-DC-AMC** [25]: A decentralized, separable CNN-based approach was considered for AMC, trained on a collection of four sub-datasets, with no class imbalance problem. Moreover, it also has some potential disadvantages, such as dependence on edge devices and complexity of model consolidation.

(5) **CNN+LSTM-AMC** [26]: A dual-stream structure based on the combination of CNN and LSTM was proposed, in which one stream (CNN) is to extract the local raw temporal features from raw signals and the other stream (LSTM) is to learn the knowledge from amplitude and phase information. DL-based

approaches, such as CNN-LSTM-based dual-stream structures, have shown promise for AMC, they also have some potential disadvantages, such as dependence on large amounts of data, computational expense, sensitivity to hyperparameters, and limited applicability to known modulation schemes.

1.4. Organization

The rest of the paper is organized as follows: related works are discussed in Section 2, Section 3 presents signal portraiture and the proposed approach, Section 4 provides prelude to the FL and AMC, Section 5 explains the simulation results, and finally, the conclusion is presented in Section 6.

2. Related works

Some aspects of AMC have previously been investigated but not limited to, such as, higher order cumulants [27], cyclostationarity-based [28], instantaneous features [29], wavelet transforms [30]. Unfortunately, these techniques have shown to take a lot of time, due to the fact that they typically rely on the extraction of handmade features that demand in-depth engineering and domain understanding. Given that such classifiers' assumed signal model may be inaccurate in the novel situation, the above classifiers confront the difficulty of having to be adjusted to another channel, a different noise state, or a different environment. Along with that, even though these classifiers might be able to handle a more complicated signal model, increasing the number of unknown parameters makes parameter estimation more computationally complex. In addition, data-driven modulation classification (MC): random forests [31], K nearest neighbor (KNN) [32], support vector machines (SVM) [33], easily adaptable to new situations provided that signals identified with their modulation schemes can be gathered. These methods are based on traditional machine learning, which has the advantages of simple technological ideas, strong algorithmic feasibility, and excellent performance but necessitates challenging feature extraction.

One of the current most promising approaches for classification and regression applications is deep learning (DL). It has shown significant potential in various domains including resource allocation [34], joint antenna selection and beam forming [35], and mmWave-MIMO systems [36]. Additionally, it has been demonstrated that DL-based AMC is highly effective, largely as a result of its potent automatic feature extraction capacity [37,38]. Recently, several modulation classifiers have been developed using deep neural networks (DNN), which have a more complicated architecture and more advanced models like recurrent neural networks (RNN) and convolutional neural networks (CNN), which are used to automatically extract features from raw data [39–41]. The authors of [42] proposed a neural architecture search (NAS)-based automatic modulation recognition (AMR) method to automatically adjust the structure and parameters of a DNN and find the optimal structure under the combination of training and constraints. A multisignal frequency domain detection and recognition method was proposed by [43] in which the frequency spectrum of the time-domain overlapping signal is obtained through the fast Fourier transform (FFT) and the frequency spectrum is segmented based on the signal energy detection method and thus a complex CNN is constructed for the identification of signal spectrum information.

However, the vast majority of DL-based algorithms, including that for AMC, rely on massive amounts of data saved on a centralized server that are gathered from a variety of local clients (devices) and are known as centralized learning (CL). For instance, in [44] for improved performance and increased resilience, a large amount of modulation signals uploaded from diverse clients is used to train the DL-based CL-AMC. But this upload can lead to the streaming of the user's information that is stored in modulation signals. Thus, this leads to a major challenge in terms of leakages, i.e., data leaks associated with CL, and it faces a significant hurdle in terms of privacy breach. In addition, AMC at the level of local devices leads to inadequate classification robustness and accuracy.

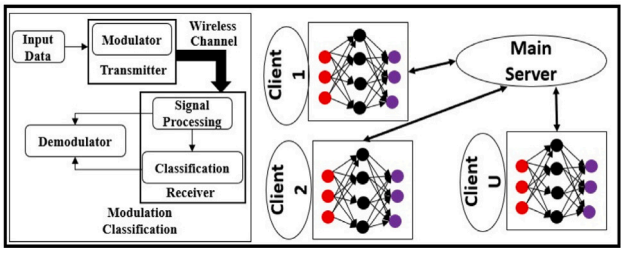


Fig. 1. Elucidation of system model.

A CNN-based AMC was first proposed by [45], in which analog and digital modulation signals were classified with high accuracy in comparison to a traditional-based AMC. A hybrid model was proposed by [46], in which residual neural networks (RNN) and long short term memory (LSTM) are combined for AMC, wherein they demonstrated the RNN + LSTM approach having greater performance than the feature-based AMC approach. Furthermore, it was demonstrated by [47] that CNN-based AMC approaches have higher inference speeds than likelihood-based AMC techniques. [48] presented a lightweight AMC approach in which asymmetric convolution structures are used in the proposed model to reduce computational complexity. The vanishing gradient problem is resolved using the skip connection method, which also boosts calculation speed while improving classification accuracy. Although, as mentioned above, AMC provided good accuracy and faster computation, the noise environment considered by them is additive white Gaussian noise (AWGN) along with in-phase quadrature (IQ) training samples. A more realistic scenario with white non-Gaussian noise is considered by [49] for AMC using CNN, but also with IQ samples. In [50], they used CNN-based models: AlexNet and GoogLeNet, thereby converting IQ samples into constellation diagrams for AMC and achieving excellent performance in terms of classification. The above-mentioned AMC approaches use a single information modality.

In order to implement AMC based on deep residual networks (Resnet) for the classification of higher order digital modulation types, [51] made use of the waveform spectrum multimodal fusion (WSMF) method. The classification results, particularly for higher-order digital modulation types, were excellent. The visualization of different types of modulation signals was proposed by [52], by extracting radio features, thereby indicating that short radio samples lead to misclassifications. Since the majority of DL AMC approaches have been studied for single-input and single-output (SISO) systems, [53] suggested a CNN-based zero-forcing (ZF) equalization AMC (CNN/ZF-AMC) method for multiple-input and multiple-output (MIMO) systems, based on ZF equalization AMC. These methods discussed CNN-based AMCs with admirable classification accuracy, but they are centralized approaches, and they fail to consider decentralized (distributed) scenarios. A federated learning (FL)-based AMC approach, called as FL-DP-CNN-AMC, was proposed by [24], where data privacy and security are maintained while the recognition rate is acceptable. But they only considered AWGN as a noise environment. The FL based AMC along with the noise imbalance approach was proposed by [23], called as FL-CNN-AMC, in which good accuracy of classification is obtained with low risk of data leakage, but used only 4 clients. They also introduced balanced cross entropy (BCE) as a loss function for solving the noise imbalance problem, but the fundamental flaw with BCE, however, is that it solely takes into account class label errors.

3. Signal portraiture and proposed approach

A basic proposed system model is shown in Fig. 1, which consists of a transmitter which mainly constitutes a modulator and the receiver is

equipped with signal processing and classifier (AMC) for demodulation. The main server, M_s, is connected with local clients, U, which process the data at the local level and only share outcomes (biases, weights, and gradients) with M_s. The modulation scheme can be chosen from a consolidated listing: BPSK, QPSK, 8PSK, 16QAM, 64QAM, PAM4, GFSK, CPFSK, [54–56].

The channel impulse response is a complex-valued function that describes the multipath fading and the frequency-selective nature of the wireless channel. One commonly used model for the channel impulse response is the Saleh–Valenzuela model [57], which can be expressed as:

$$h_i(t) = \sum_{j=1}^{L} a_{i_j} e^{-j2\pi f_{d_j} t} e^{-j\phi_{i,j}} \delta(t - \tau_{i_j}), \tag{1}$$

where i is the index of the path, j is the index of the delay element, L is the number of delay elements, a_{i_j} is the amplitude of the jth delay element for the ith path, f_{d_j} is the Doppler frequency for the jth delay element, ϕ_{i_j} is the phase shift for the jth delay element of the ith path, and τ_{i_j} is the delay for the jth delay element of the ith path.

The received signal with N paths can be expressed as:

$$r(t) = \sum_{i=1}^{N} h_i(t) s_i(t - \tau_i) + n(t), \tag{2}$$

where $h_i(t)$ is the complex-valued channel impulse response for the ith path, $s_i(t)$ is the complex-valued baseband signal for the ith path, τ_i is the delay of the ith path, and $n(t)$ is the additive white Gaussian noise. The complex-valued baseband signal $s_i(t)$ is expressed as $s_i(t) = A_i e^{j(\phi_c + \Delta\phi_i(t))}$, where A_i is the amplitude of the ith path, ϕ_c is the carrier phase, and $\Delta\phi_i(t)$ is the phase offset.

The received signal $r(t)$ can be used to extract features for modulation classification. One common approach is to use the time–frequency representation of the received signal, such as the spectrogram or the continuous wavelet transform, as the input to a deep learning model.

The basic focal loss equation is given as [58]

$$\mathrm{FL}(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t), \tag{3}$$

where p_t is the predicted probability of the true class label at time t, α_t is the weight assigned to the sample at time t, and γ is the focusing parameter that controls the degree of down-weighting for well-classified signals. The term $(1-p_t)^{\gamma}$ is the focal factor, which reduces the weight of easy signals and emphasizes the weight of hard signals. The parameter γ controls the degree of down-weighting for well-classified signals. The weight α_t is a scalar that can be used to assign different weights to different samples in the training data. It is typically defined as:

$$\alpha_t = \begin{cases} \alpha & \text{if } y_t = 1 \\ 1 - \alpha & \text{otherwise,} \end{cases} \tag{4}$$

where y_t is the true class label at time t, and α is a hyperparameter that controls the degree of weighting. The α parameter is a single scalar value that is used to assign a somewhat higher weight to the minority class. However, this uniform weighting does not differentiate between hard signals and easy signals within the minority class. All minority class signals receive the same weight, regardless of their hardness. Hardness in this context refers to how difficult or easy the signal is for the model to classify correctly.

Therefore, modified focal loss (L_{CFL}) is proposed to address two main challenges that arise in the context of modulation classification:

1. **Class Imbalance in Modulation Classification:** AMC is indispensable in contemporary wireless communication systems, tasked with identifying the modulation scheme employed by incoming signals. However, a practical challenge known as the class imbalance problem complicates AMC due to the dynamic and diverse nature of real-world communication scenarios.

Wireless networks strategically employ various modulation schemes based on factors like signal strength, channel conditions, and application requirements. This strategic selection results in an uneven distribution of modulation classes, contributing to class imbalance. Furthermore, the impact of channel conditions, particularly in multipath channels and fading environments, exacerbates class imbalance, affecting the prevalence of certain modulation schemes.

Addressing class imbalance in AMC is crucial for optimizing system performance. A balanced model is essential for robust performance across all modulation classes. An imbalanced model, biased towards prevalent classes, can compromise accuracy, particularly for less common modulation types, impacting communication performance and QoS standards.

Moreover, the adaptability of AMC systems to dynamic environments is hindered by class imbalance. A robust AMC system should dynamically adjust to the prevalence of modulation types in real-time, aligning with variations in the communication environment. In essence, class imbalance is not just a theoretical concern but a practical challenge deeply embedded in the dynamic nature of wireless communication scenarios, necessitating solutions for accurate, adaptable, and reliable AMC systems in real-world applications.

2. Channel Effects in Modulation Classification: Modulation classification often occurs in wireless communication signals, influenced by the quality of the wireless channel. The channel-aware weight function in the modified focal loss considers the wireless channel's quality, assigning varying weights to samples based on their channel conditions. This approach aims to diminish the impact of samples with poor channel conditions on the loss function, ultimately enhancing the model's overall performance in the modulation classification task. The consideration of channel effects is crucial in ensuring the robustness and reliability of the AMC model, especially when operating in diverse and challenging wireless communication environments.

Therefore, balancing parameter, a class-aware weight factor, α in the modified focal loss helps to mitigate this problem by adjusting the weights assigned to the samples in the minority and majority classes, thereby reducing the impact of the majority class on the loss function and improving the overall performance of the model on the minority class, i.e. it uses a vector of α values, with a separate α value assigned to each signal based on its class and difficulty. The α values are determined based on both the class imbalance and the individual hardness. Thus higher α values are assigned to signals from the minority class to compensate for the lower number of samples. Within the minority class, higher α values are given to hard signals that are likely to be misclassified. Lower α values are assigned to signals from the majority class and to easy signals that are likely to be classified correctly. Therefore, modified focal loss uses a vector of α values, individually tuned for each signal based on class and difficulty.

Thus, the modified focal loss L_{CFL} is proposed for addressing the challenges of class imbalance and multipath channels in the context of modulation classification and is expressed as

$$L_{CFL}(y, v) = -\sum_{i=1}^{C} \left[\alpha_i (1 - y_i)^{\gamma} \log(y_i) + (1 - \alpha_i) y_i^{\gamma} \log(1 - y_i)\right], \tag{5}$$

where y is the predicted probability vector for the input sample, v is the true class label vector for the input sample, C is the number of classes, α_i is the class-aware weight factor for class i, and γ is the focusing parameter of the loss function. The class-aware weight factor α_i serves as a scalar value representing the weight assigned to class i within the loss function. The computation of α_i involves the utilization of the weight function, expressed as: $\alpha_i = \frac{1}{\sum_{j=1}^{\varpi} w(r(t)_j, t_i)}$. Here, ϖ denotes the total number of samples in the dataset, r_j signifies the received signal for sample j, and t_i represents the time instant corresponding to class i. The weight function $w(r_j, t_i)$ intricately assigns a weight to

each sample, taking into account its received signal and the time instant corresponding to its true class label and is given as

$$w(r(t),t) = \exp\left(-\frac{(t-\text{delay}(r(t)))^2}{2\sigma_t^2}\right) \exp\left(-\frac{|r(t)|^2}{2\sigma_r^2}\right) w_c(t), \quad (6)$$

In Eq. (6), the tuning parameters σ_t and σ_r play a crucial role in shaping the behavior of the weight function. These parameters control the contribution of the delay and amplitude components, respectively, influencing the assignment of weights to samples. The term delay$(r(t))$ represents the delay associated with the received signal $r(t)$. The careful tuning of σ_t and σ_r allows for the customization of the weight function, tailoring it to the specific characteristics of the dataset. Adjusting these parameters enables helps in the emphasis or de-emphasis of the impact of temporal delay and signal amplitude, providing a flexible mechanism to adapt the model's sensitivity to different aspects of the input data. The inclusion of the term $w_c(t)$ further adds a temporal weight component, enhancing the adaptability of the weight function across different temporal instances. Thus, the weight function in Eq. (6), guided by the tuning parameters σ_t and σ_r, empowers the model to dynamically adjust the importance assigned to individual samples based on their temporal characteristics and received signal properties. The tunability of these parameters ensures versatility in addressing diverse datasets and varying temporal dynamics.

The delay factor delay$(r(t))$ is given as delay$(r(t)) = \arg\max_t |r(t)|^2$, which represents the time delay associated with the received signal power P_r given as $P_r = \frac{1}{T}\int_0^T \left|\sum_{i=1}^N h_i(t)s_i(t-\tau_i) + n(t)\right|^2 dt$ $= \frac{1}{T}\int_0^T \left|\sum_{i=1}^N h_i(t)A_i e^{j(\phi_c+\Delta\phi_i(t))}\delta(t-\tau_i) + n(t)\right|^2 dt$ $= \frac{1}{T}\int_0^T \left|\sum_{i=1}^N h_i(\tau_i)A_i e^{j(\phi_c+\Delta\phi_i(\tau_i))} + n(\tau_i)\right|^2 d\tau_i$. Assuming delay τ_i is contained in the delta function, expanding the square and taking the expectation over the noise $n(t)$, we get, $E[|r(t)|^2] = E\left[\left|\sum_{i=1}^N h_i(\tau_i)A_i e^{j(\phi_c+\Delta\phi_i(\tau_i))} + n(\tau_i)\right|^2\right] = \sum_{i=1}^N |h_i(\tau_i)|^2|A_i|^2 + E[|n(\tau_i)|^2] + 2\,\text{Re}\left\{\sum_{i=1}^N h_i(\tau_i)A_i e^{j\phi_c}\,\text{E}\left[e^{j\Delta\phi_i(\tau_i)}\right]\right\}$, and $w_c(t)$ is the class-aware weight factor, given as $w_c(t) = \frac{N_{\text{majority}}}{N_{\text{minority}}} \times \mathbb{1}_{\text{minority}}(t) + \mathbb{1}_{\text{majority}}(t)$, where N_{majority} and N_{minority} are the number of samples in the majority and minority classes, respectively, $\mathbb{1}_{\text{minority}}(t)$ and $\mathbb{1}_{\text{majority}}(t)$ are indicator functions that return 1 if the sample at time t belongs to the minority or majority class, respectively.

Moreover, $\text{Re}\cdot$ and $\text{Im}\cdot$ denote the real and imaginary parts, referred to as the in-phase (I) and quadrature (Q) component: IQ component, and are merged with the sample for training and testing of the proposed approach thus $r(t)$ is written as $\text{I}(t) = \text{Re}\,r(t) = \sum_{i=1}^N \text{Re}\,h_i(t)s_i(t-\tau_i) + \text{Re}\,n(t)$ and $\text{Q}(t) = \text{Im}\,r(t) = \sum_{i=1}^N \text{Im}\,h_i(t)s_i(t-\tau_i) + \text{Im}\,n(t)$.

4. Prelude: Federated learning

A wireless multi-user system with just one primary server (M_s) is taken into consideration and U number of users (clients), where in each participating client u stores its data with size d_u, locally, such that the total data size D is given as $D = \sum_{u=1}^U d_u$, where a set of input–output pairs can be used to define as a collection of data samples d_u, as $\{x_j, y_j\}_{j=1}^{d_u}$, where $x_j \in \mathbb{R}^d$ is a sample input vector with d characteristics, and $y_j \in \mathbb{R}$ is the labeled output value for the sample x_j.

For a sampled data pairs of $\{x_j, y_j\}$, for x_j the input, the task is to find the global model parameter m that is able to characterize output y_j. This characterization of the output with a loss function $l(f_j(m))$ can be represented in the forms: for linear regression: $l(f_j(m)) = \frac{1}{2}\left(x_j^T m - y_j\right)^2, y_j \in \mathbb{R}$ and for support vector machine: $l(f_j(m)) = \left\{0, 1 - y_j x_j^T m\right\}, y_j \in \{-1, 1\}$, where, $x_j^T m$ is linear mapping function. Then, the global optimization problem can be represented for the model parameter m, as

$$\min_{m\in\mathbb{R}^d} J(m) = \sum_{u=1}^U \frac{d_u}{D} l(J_u(m)), \quad (7)$$

Thus, for the user u the local loss function is given as $l(J_u(m)) = \frac{1}{d_u}\sum_{j=d_u} l(f_j(m))$, and the loss function $l(f_j(m))$ captures the error of the model parameter m on the input–output pair $\{x_j, y_j\}$.

4.1. FL based modulation classification explication

Iterative based approach is used by FL for solution of Eq. (7) and each iterative round is indexed by t. Therefore, in iteration t: the server selects clients S_{d_t} and distributes the current global model m_{t-1} and each client u computes local gradient updates to minimize, $l(J_u(m)) = \frac{1}{d_u}\sum_{j=d_u} L_{CFL}(y_j, v_j)$. The server aggregates local gradients to obtain a new global model m_t, the process repeats with m_t distributed for the next iteration $t+1$ and it consists of three steps: **(1) Scheduling of participants:** The main server M_s selects the initial model parameters m_p, and schedule S_d for its training, and then broadcast these details to each client participant $U_{p_i}, i \in 1, 2..N_c$ where N_c are the number of clients for an iterative round t. For each round of training, the server M_s selects a subset of clients $S_{d_t} \subseteq U$ to participate in the training process. During the scheduling process, the server M_s also broadcasts the current global model parameter, $m_{p_{t-1}}^{S_{d_{t-1}}}$, to each selected client U_{p_i}. This model parameter is the result of the previous round of training, and it serves as the starting point for the current round. In the modified focal loss approach, each client U_{p_i} has its own local dataset, and uses the gradient descent algorithm to update its local model parameter, $\tilde{m}_p \in m_p$ for a learning rate of μ. Once the clients have completed their local updates, they upload their parameters to the server M_s for aggregation. The server M_s then aggregates the local parameters using a weighted average to obtain a new global model parameter, which is then used as the starting point for the next round of training. **(2) Schedule sharing:** Each U_{p_i} has its own local dataset, and uses the gradient descent algorithm to update its local model parameter, $\tilde{m}_p \in m_p$ for a learning rate of μ, using the modified focal loss. For a mini-batch $d_{U_p} \subseteq D_{U_p}$ of $U_p \in u$ data samples, the estimate of the gradient $\nabla f_j\left(m_p(t)\right)$ is given as $\tilde{\nabla} f_j\left(m_p(t)\right) = \frac{1}{|d_{U_p}|}\sum_{j\in d_{U_p}} \nabla L_{CFL}(y_j, v_j)$, where y_j is the predicted probability distribution for a given class, v_j is a one-hot encoded vector representing the true class label at time t for the jth data sample. The modified focal loss function L_{CFL} computes the loss for each data sample, which incorporates the channel-aware weight function $w(r_j(t), t)$ to handle the effects of multipath in terms of the received power P_r. The exchange of the local model parameter is scheduled and repeated ϱ times, represented as $f_j(m_p(\varrho))$ is uploaded and shared to M_s. The server M_s then aggregates the local model parameters from all the selected clients using a weighted average to obtain a new global model parameter, $m_p(t+1)$. The evaluation is performed on the global model parameter $m_p(t+1)$ after the completion of each round of training. The training process continues for a fixed number of rounds or until convergence is achieved. The convergence is determined by monitoring the change in the global model parameter after each round of training. If the change is below a certain threshold, the training process is terminated. **(3) Update global model**: Weighted average combination of the local parameters received is carried out by the server M_s, which aggregates them to obtain a new global model parameter, $m_p(t+1)$, given as $m_p(t+1)^{S_{d_t}} = \frac{\sum_{U_{p_i}\in S_{d_t}} w(r_i(t),t)\tilde{m}_p(t+1)^{S_{d_t}}}{\sum_{U_{p_i}\in S_{d_t}} w(r_i(t),t)}$, where $\tilde{m}_p(t+1)^{S_{d_t}}$ is the local model parameter for client U_{p_i} after the completion of the training process, $w(r_i(t), t)$ is the channel-aware weight function that incorporates the delay and amplitude components of the received signal for client U_{p_i}, and S_{d_t} is the set of selected clients for round t. The resulting global model parameter, $m_p(t+1)$, is used as the starting point for the next round of training. Thus, the server

M_s aggregates the local model parameters received from the selected clients by using a weighted average with the modified focal loss and the channel-aware weight function. The resulting global model parameter is used as the starting point for the next round of training.

Therefor, proposed approach exhibits the potential for enhanced accuracy compared to centralized approaches. In AMC, diverse modulation schemes are prevalent, and FL leverages the decentralized nature of training across devices to capture a broader spectrum of modulation patterns. By preserving local insights on devices without sharing raw data, FL ensures that the model benefits from the collective intelligence of the distributed network. This approach is particularly advantageous in scenarios with class imbalance and varying channel conditions, allowing the model to adapt more effectively to the intricacies of wireless communication signals. Thus, the collaborative learning framework of FL positions it favorably for achieving improved accuracy in AMC tasks compared to traditional centralized approaches.

4.2. Problem formulation and objective function

In FL, communication overheads refer to the amount of data that needs to be exchanged among the clients and the server during the training process. This includes the data used to update the model parameters, as well as the control information used to coordinate the training process. The accumulated communication rounds (ACR) refer to the total number of times that the clients and the server need to exchange data during the training process i.e. defined as the total number of communication rounds made by the local clients in t iterations. The number of communication rounds is closely related to the communication overheads, as each communication round involves the exchange of data and control information, which contributes to the overall communication overheads. In general, the more communication rounds required, the higher the communication overheads will be. As the number of communication rounds increases, the amount of data exchanged among the clients and the server also increases, which results in higher communication overheads. This can have a significant impact on the overall performance of the FL system, as the communication overheads can consume a significant portion of the available resources. As already discussed FL clients upload data to the main servers, i.e., client updates throughout the entire training process.

Thus, specifically for FL, the communication overhead can be juxtaposed to the number of times the model parameters are transmitted between the local clients and the server. Lowering the communication overhead means minimizing the ACR. The ACR for t iterations can be written as:

$$\varsigma_t = \sum_{i=1}^{CR_t} \left| S_{t,i} \right|, \tag{8}$$

where $S_{t,i}$ is the set of local clients selected for the ith communication round (CR) in iteration t.

To minimize the communication overhead while optimizing the model with the updated focal loss, the optimization problem is formulated as

$$\begin{aligned} &\min_{\varsigma_t} \lambda L_{CFL}(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t; \alpha_t, \zeta) + \mu \text{fairness}(\mathbf{z}_t) \\ &+\zeta \left| \mathbf{z}_t - \mathbf{z}_{t-1} \right|^2 + \beta \left| f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) - f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t) \right|^2, \end{aligned} \tag{9}$$

where $\mathbf{z}_t$ represents the local model parameters at iteration t, $\mathbf{z}_t^*$ is the optimal local model parameter at iteration t, $\mathbf{X}_t$ is the input dataset at iteration t, $\mathbf{y}_t$ is the corresponding output labels, and λ, μ, β, and ζ are weighting factors that balances the trade-off between minimizing the communication cost, optimizing the model, promoting fairness, and minimizing the difference between the local model parameters and the optimal local model parameter. The term $\left| f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) - f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t) \right|^2$ measures the distance between the local model parameters at iteration t and the optimal local model parameter at iteration t in terms of the loss function, encouraging the local model parameters to be closer to the optimal local model parameter. The term $\left| \mathbf{z}_t - \mathbf{z}_{t-1} \right|^2$ represents the communication cost of transmitting the local model parameters from the previous iteration to the current iteration. The term $|z_t - z_{t-1}|^2$ in the optimization problem acts as a surrogate for the communication cost associated with transmitting local model parameters from the previous iteration $(t-1)$ to the current iteration (t) in FL. Motivated by the Euclidean distance concept, where $|z_t - z_{t-1}|^2$ quantifies the squared distance between the local model parameters at the current and previous iterations, it serves as a measure of change in the model. In the context of FL, transmitting the entire model incurs a communication cost, and this choice captures the cost by considering the squared Euclidean distance between local model parameters in consecutive iterations. A smaller squared distance indicates less significant changes, minimizing communication cost. Therefore, utilizing $|z_t - z_{t-1}|^2$ aims to balance accurate model updates with the objective of minimizing communication overhead during FL.

The optimization problem is subject to constraints on the values of the variables, which ensure that the solution is non-negative and bounded by upper limits u_i for each variable z_i. Specifically, the constraints are:

$$0 \le z_i \le u_i \quad \text{for all } i. \tag{10}$$

The optimization problem is solved using the Karush–Kuhn–Tucker (KKT) conditions [59], which provide a set of necessary conditions for optimality. The optimization problem, along with its constraints, is written as

$$\begin{aligned} &\mathcal{L}(\varsigma_t, \lambda, \boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \\ &\lambda L_{CFL}(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t; \alpha_t, \zeta) + \mu \text{fairness}(\mathbf{z}_t) + \zeta \left| \mathbf{z}_t - \mathbf{z}_{t-1} \right|^2 \\ &+\beta \left| f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) - f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t) \right|^2 + \sum_{i=1}^{n} \alpha_i (z_i - u_i) \\ &+ \sum_{i=1}^{n} \beta_i (-z_i) + \lambda^T (\mathbf{z}_t - \mathbf{z}_{t-1}) + \boldsymbol{\mu}^T \mathbf{z}_t + \boldsymbol{\nu}^T f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) + \boldsymbol{\gamma}^T \mathbf{z}_t, \end{aligned} \tag{11}$$

where $\lambda, \boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{\gamma}, \boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the Lagrange multipliers (LM) associated with the non-negativity and upper bound constraints, respectively.

The optimization problem subject to these constraints using the KKT conditions, the KKT conditions are modified as

$$\begin{aligned} &\frac{\partial \mathcal{L}}{\partial \varsigma_t} = \lambda \frac{\partial L_{CFL}}{\partial \varsigma_t} + \mu \frac{\partial \text{fairness}}{\partial \varsigma_t} + 2\zeta(\mathbf{z}_t - \mathbf{z}_{t-1}) \\ &+2\beta(f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) - f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t)) \frac{\partial f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t)}{\partial \varsigma_t} \\ &+\lambda + \boldsymbol{\mu} + \boldsymbol{\nu} \frac{\partial f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t)}{\partial \varsigma_t} + \boldsymbol{\gamma} + \boldsymbol{\alpha} - \boldsymbol{\beta} = 0, \end{aligned} \tag{12}$$

The modified KKT condition for the partial derivative of the Lagrangian ($\mathcal{L}$) with respect to ς_t. The Eq. (12) combines the derivatives of the objective function L_{CFL} and the fairness term, along with other terms related to the optimization process and the LM.

$$\frac{\partial \mathcal{L}}{\partial \lambda} = \mathbf{z}_t - \mathbf{z}_{t-1} = 0, \tag{13}$$

where, the partial derivative of $\mathcal{L}$ with respect to LM, λ encapsulates the essence of continuity between current iteration's variable values $\mathbf{z}_t$ and those of preceding iteration $\mathbf{z}_{t-1}$.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}} = \mathbf{z}_t = 0, \tag{14}$$

where, it sets a stringent mandate by equating the partial derivative of the $\mathcal{L}$ with respect to $\boldsymbol{\mu}$ to the current iteration's variable values $\mathbf{z}_t$. This condition dictates that the variable values at the present iteration align precisely with the expectations encapsulated by the LM, $\boldsymbol{\mu}$. The enforcement of equality, Eqs. (13) $\sim$ (14), with zero implies a strict adherence to the optimization process and underscores the importance of maintaining equilibrium between the variables and the LM.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\nu}} = f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t) - f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t) = 0, \tag{15}$$

where, the partial derivative of $\mathcal{L}$ with respect to the LM, ν is linked intricately to the discrepancy between the function evaluations at the current iteration and the optimal scenario by ensuring that the difference in the function values, as dictated by the ν, converges to zero. Thus, it ensures the equality between the function evaluations $f(\mathbf{z}_t, \mathbf{X}_t, \mathbf{y}_t)$ and $f(\mathbf{z}_t^*, \mathbf{X}_t, \mathbf{y}_t)$.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\gamma}} = \mathbf{z}_t = 0, \tag{16}$$

where, this condition asserts a stringent equality constraint, compelling the variable values at the current iteration $\mathbf{z}_t$ to be precisely zero. This imposition of zero is critical in the optimization landscape, signifying that the influence exerted by the LM, $\boldsymbol{\gamma}$ aligns the variable values with the condition of nullity. Thus, it imposes the equality constraint between the current iteration's variable values $\mathbf{z}_t$ and zero.

$$\alpha_i(z_i - u_i) = 0 \quad \text{for all } i, \tag{17}$$

where, this encapsulates the complementary slackness condition for the LM, α_i associated with non-negativity constraints. Thus, it reflects that when a variable is non-negative ($z_i \geq 0$), the corresponding LM can be zero, indicating that the non-negativity constraint is not actively influencing the optimization process, and ensures that either the variable value z_i is zero or α_i is zero for all i.

$$\beta_i z_i = 0 \quad \text{for all } i, \tag{18}$$

Eq. (18), represents the complementary slackness condition for the LM, β_i associated with upper-bound constraints. Similar to the previous condition, it ensures that for each variable z_i, either the variable value z_i is zero or β_i is zero. This condition underscores the duality between the upper-bound constraints and their respective LM, emphasizing that the constraint is either satisfied or its associated LM becomes inactive.

$$z_i \geq 0 \quad \text{for all } i, \tag{19}$$

$$z_i \leq u_i \quad \text{for all } i, \tag{20}$$

$$\alpha_i \geq 0 \quad \text{for all } i, \tag{21}$$

$$\beta_i \geq 0 \quad \text{for all } i. \tag{22}$$

where, Eq. (19) ensures that each variable z_i must be non-negative, Eq. (20) imposes an upper bound constraint on each variable z_i, ensuring that z_i does not exceed its prescribed limit u_i, Eq. (21) enforces non-negativity for the LM α_i, Eq. (22) similarly imposes non-negativity for the LM β_i. Collectively, these conditions define the boundaries and relationships that the solution must satisfy, contributing to the robustness and feasibility of the optimization process. They strike a balance between variable values and LM, ensuring adherence to constraints while optimizing the objective function.

The complementary slackness condition, combines the complementary slackness conditions for non-negativity and upper bound constraints. It states that either the variable value z_i is zero or LM, α_i and β_i are zero for all i and is given as

$$\alpha_i(z_i - u_i) = \beta_i z_i = 0, \tag{23}$$

Sequential quadratic programming (SQP) [60] is used to find the optimal solution for the decision variables ς_t that satisfy the constraints on the variable values, where in the solution obtained satisfies the complementary slackness conditions for the non-negativity and upper bound constraints. If the complementary slackness conditions are not satisfied, the LM are adjusted until they are satisfied. The resulting solution will be the optimal solution to the optimization problem subject to the constraints.

4.3. Proposed federated CNN description

In this study, CNN is used as the DL model for AMC because it has demonstrated strong classification performance even for PSK and QAM signals, [1,6,61].

Table 1
CNN Description.

Structure	Number of Layers
Input (samples, categories)	–
Conv1D(128,16)+ ReLU+ BN + Dropout (0.1)	2
AveragePool	1
Conv1D(64,8)+ ReLU+ BN + Dropout (0.1)	1
Fully Connected	1
Dense (1024) + Softmax +Dropout (0.5)	1

The CNN consists of total 6 layers, in Table 1, including 3 convolution layers (Conv1D),1 average pool layer, 1 fully connected (FC) and 1 Dense layer. In each Conv1D, batch normalization (BN) and rectified linear unit (ReLU) is applied along with Dropout which avoids overfitting problem. For last layer (Dense) Softmax activation function is used along with Dropout. Additionally, the output is a probability distribution that contains the probability of each type of modulation, with the input being the dataset (samples and categories).

The real and imaginary components of a complex vector y are separated to convert it into actual data. For instance, at the first Conv1D, $c_l = 1$, the input vectors are presented as $x^{c_l} = y \in \mathbb{R}^{\left[N_{c_l-1} \times V_{c_l-1}\right]}$, where N and V are the dimensions and vector number, respectively. Convolution layer c_l with 1 dimension, have c_{l-1} input and c_l output vectors, thereby giving $c_l \times c_{l-1}$ convolution kernels and c_l biases, for a kernel size of k_{c_l}. The current layer's output vector for the results of the previous layer $x_i^{c_l-1}, i \in \left\{1, \ldots, N_{c_{l-1}}\right\}$ is written as $x_i^{c_l} = f\left(\sum_{j=1}^{N_{c_{l-1}}} x_j^{c_l-1} * k_{i,j}^{c_l} + b_i^{c_l}\right)$, where ReLU is the non-linear activation function: $f(x) = max(0, x)$ and the convolution operation is represented as $*$. Average pooling layer reduces the dimensionality by aggregating the average value of the elements present in the vector, thus for the ith output vector, $x_i^{c_l+1} = down(x_i^{c_l})$. Then the output vectors are packed and restricted into a vector by the fully connected layer. After that, a dense layer d_l encodes the input vector $x^{d_{l-1}}$ into a output vector (extracted high level information) given as $x^{d_l} = f\left(W^{d_l} x^{d_{l-1}} + b^{d_l}\right)$, where $W^{d_l} \in \mathbb{R}^{N_{d_l} \times N_{d_{l-1}}}$ and $b^{d_l} \in \mathbb{R}^{N_{d_l}}$. After that, the dense layer with Softmax activation function gives normalized likelihood vector $\hat{\varpi}^N = \left[\hat{\varpi}_1^N, \ldots \hat{\varpi}_l^N\right]$, where $\hat{\varpi}_i^N = \frac{e^{x_i}}{\sum_{j=1}^{l} e^{x_j}}$. Thus providing a final output as $\hat{\varpi}^N = \mathbb{F}(y, \varrho)$, where ϱ is the network parameter and $\mathbb{F}(\cdot)$ is overall CNN-based AMC. Thus, the final classification of the modulation $\hat{C}_m$ is based on the predicted probability distribution $\mathscr{P}(W; IQ)$ and is given as $\hat{C}_m = \underset{C_m \in M}{arg\ max}\ \mathscr{P}(W; IQ)$, where the projected modulation type is the modulation type with the highest likelihood. By doing this, a one-hot vector will be generated automatically from each category found in the dataset. A probability that an signal corresponds to any of the categories will be output by the model by learning to predict each one-hot vector, which can be thought of as a probability distribution.

5. Simulation results

The specified IQ sample distribution across local clients is shown Table 2, which are applied for training purpose. Additionally, an independent testing dataset was prepared for evaluating the performance of the modulation classification model. The testing dataset consists of 25,000 IQ samples for each modulation type, ensuring a comprehensive evaluation across the different modulation schemes as shown in Table 3, with SNR range from −12 dB to 12 dB.

The training parameters: batch size = 128, maximum iterations (epochs) = 1000 and learning rate = 0.001. The raw IQ domain of the 8 modulation signals that are used for the analysis and comparison are shown in Fig. 2.

The analysis and comparison is carried out under two broad categories:

Table 2
IQ training samples across local clients.

Modulation Scheme	BPSK	QPSK	8PSK	16QAM	64QAM	PAM4	GFSK	CPFSK
Local Clients (LC)	IQ Training Samples							
LC1	10000	5000	5000	2000	2000	2000	10000	2000
LC2	5000	10000	5000	2000	5000	10000	2000	2000
LC3	5000	5000	5000	10000	2000	2000	10000	2000
LC4	2000	5000	5000	2000	10000	2000	5000	10 000
LC5	2000	10000	10000	5000	2000	5000	2000	5000
LC6	5000	2000	2000	10000	5000	10000	5000	2000
LC7	10000	2000	2000	5000	10000	2000	5000	5000
LC8	2000	5000	5000	10000	5000	2000	2000	10 000

Table 3
Data set description.

Number of Modulations	8
Modulation Type	BPSK, QPSK, 8PSK, 16QAM, 64QAM, PAM4, GFSK, CPFSK
SNR (E_s/N_o) Range	−12:2:12 dB
Channel Specifications	Multi reflected and scattered paths, Carrier frequency offset, Phase offset, Doppler shift, AWGN

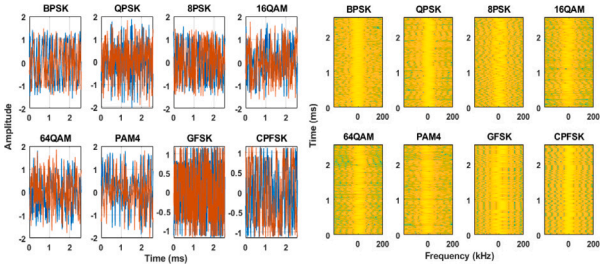


Fig. 2. IQ time domain for 8 modulation signals.

1. Parametric influence analysis, in which proposed approach is rationalized in terms of: different number of samples, delay spread, phase and frequency offset, classification error.

2. Comparative analysis, in which comparison is made between the proposed and contemporary approaches in terms of: confusion metrics, effect of level of interference, correct classification probability and communication overheads. For analysis and comparison multipath channel is taken into consideration unless other wise specified.

The correct classification probability (CCP) is used to mark the classification performance which is given as $CCP^i = \frac{N_c^i}{N_s}$ at $SNR = i\ dB, i \in \{-12, -10, \ldots, 12\}$, where N_c^i and N_s are number of correct classification at respective $i\ dB$ and total number of test samples, respectively.

1. Parametric Influence Analysis: Proposed approach is evaluated for different number of samples (l), presented in Fig. 3(a), in which with the increase in the number of samples, CCP increases with improvement in the value of SNR.

Delay spread (DS) also relates to the different levels of multipath channels and its effect, illustrated in Fig. 3(b), along with AWGN. As SNR (E_s/N_o) improves, CCP increases consistently and as the value of DS increase there is only marginal drop in the CCP, i.e. the proposed model is able to maintain high value of CCP ≈ 0.93 for DS = 2.0 at SNR = 12 dB.

Normal and acceptable range of the phase is between 0° and 90° and where as the frequency offset is only few Hertz [62]. They causes rotation of the signal constellation, which can degrade CCP. Fig. 3(c), shows the behavior of CCP for the proposed scheme for frequency offset varying from 0 to 0.001 (Hz) and corresponding phase offset, from 0° to maximum rotation of 90°. The performance of the proposed method is acceptable at the frequency offset value <0.0075 (Hz) and phase offset <75°. But for higher phase and frequency offset CCP decreases.

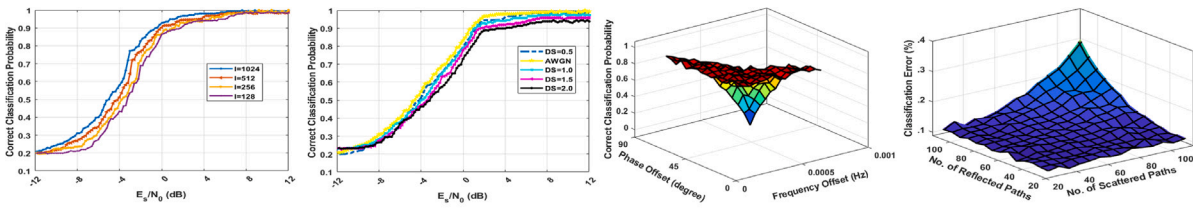
Classification error (CE) for different numbers of S and R is shown in Fig. 3(d), in which the increase in CE can only be seen at higher value (≈ 90) of R and S. Herein, CE is the measure of the performance of a modulation classifier and is calculated by dividing the number of incorrectly classified modulations by the total number of modulations. The robust performance of FL-MP-CNN-AMC, against phase and frequency offset, S and R can be attributed to CNNs compactness along with extending the modified loss function which helps in eliminating the singularities thus providing more stable and convergent network, thus it avoids exploration of more features space thereby making it less vulnerable to perturbations.

2. Comparative Analysis: The confusion matrix (CM) of the proposed approach is compared with contemporary methods, as in Fig. 4, at the SNR=10 dB. Among the FL based approaches: FL-MP-CNN-AMC, FL-CNN-AMC, and FL-DP-AMC, the proposed approach exhibits consistent performance in the identification of the modulation signal even for higher orders such as 16QAM and 8PSK. In contrast, the performance of the other two FL based methods degrades. Similarly, in the case of the two AMC methods: CNN-DC-AMC and CNN-LSTM-AMC, the performance is low even for the low-order modulation signal BPSK, and for 16QAM and 8PSK, it is significantly lower. The performance further deteriorates to the lowest level for TD-AMC, indicating that feature extraction using traditional means is inadequate. This significant difference in performance between FL-MP-CNN-AMC and other AMC approaches underscores the compelling results achieved through the formulation of the modified loss function, which mitigates the impact of R and S. It is worth noting that for 16QAM and 64QAM, the observed challenge is expected, as each frame carries only 128 symbols and 16-QAM is a subset of 64-QAM, as illustrated in Fig. 5. This partially explains the network's difficulty in correctly classifying these two types.

The performance of the proposed model is evaluated for different level of interference presented in Fig. 6, in which proposed approach not only starts with highest value of CCP, but also is able to sustain consistent CCP value as the interference level increases. Even at the maximum interference level, CCP for FL-MP-CNN-AMC is still highest as compared to other algorithms, this performance can be accredited to better extraction of the features which are robust to interference.

The CCP of FL-MP-CNN-AMC is best even in the presence of the multiple reflected and scattered paths in comparison to other FL and CNN based approaches, as shown in Fig. 7. The traditional approach, TD-AMC, shows the lowest CCP because of the feature extraction using traditional (manual) method. Moreover, other approaches also in the presence of multipath channels are not able to maintain a consistent CCP.

To conduct a comprehensive evaluation of the proposed approach, it was tested on a standard dataset. The results shown in Fig. 8(a) indicates that the proposed approach exhibits classification accuracy, which increases exponentially and approaches 1 as E_s/N_0 improves. For a comprehensive assessment of the proposed approach, it is also



(a) Number of Samples (b) Delay Spread and AWGN (c) Phase and Frequency Offset (d) Scattered and Reflected Paths

Fig. 3. Effect of parametric variations on FL-MP-CNN-AMC.

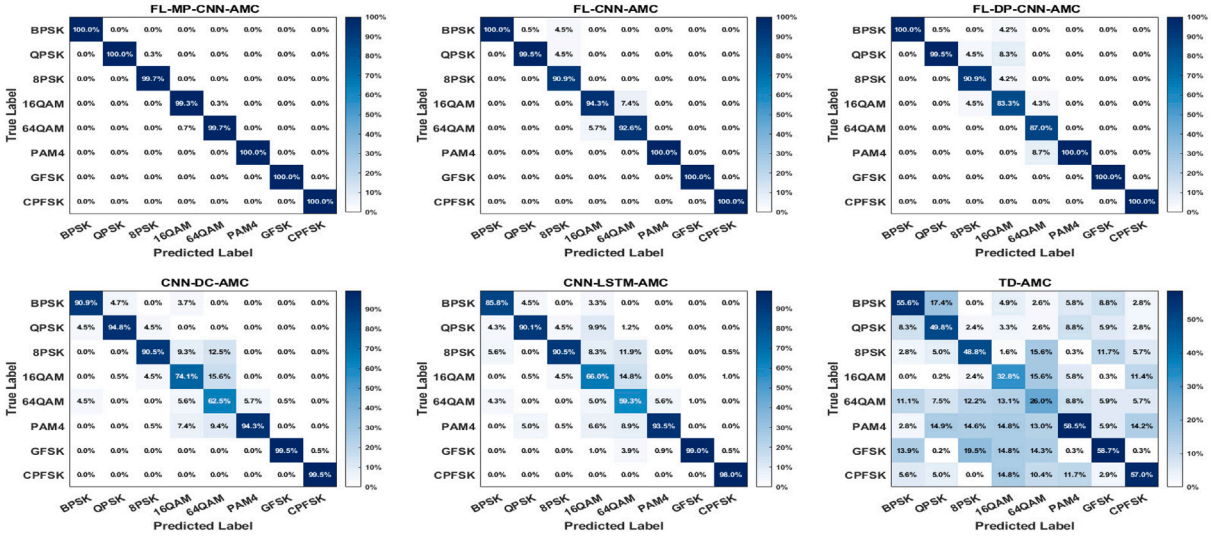


Fig. 4. Confusion matrix at SNR = 10 dB.

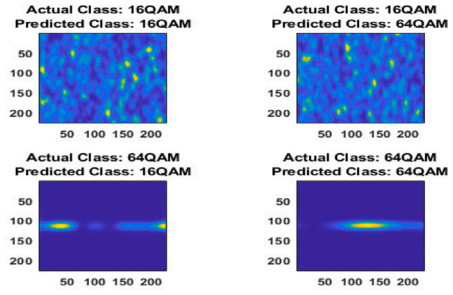


Fig. 5. Erudition of 16QAM and 64QAM.

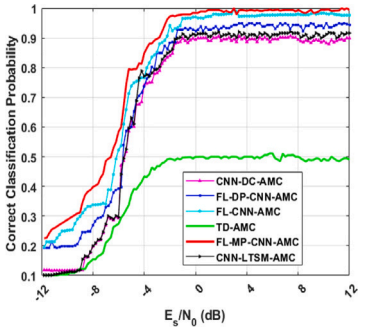


Fig. 7. CCP comparison.

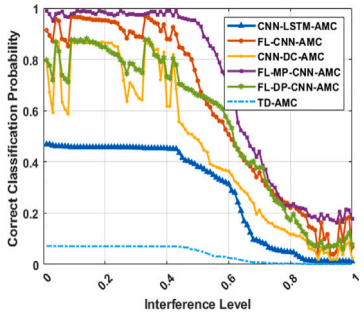


Fig. 6. Comparison of interference level effect.

subjected against a set encompassing 25 diverse modulation signals. Across this extensive range of signals, the CCP exhibits a noteworthy performance, displaying a substantial improvement with increasing E_s/N_0, presented in Fig. 8(b). The observed exponential enhancement in CCP reinforces the robustness and efficacy of the proposed approach. These evaluation demonstrates the robustness and ability of the proposed approach to effectively handle the broad spectrum of modulation signals.

Comparison of the convergence of the proposed approach is made with the contemporary approaches, in Fig. 9. Each iterative round in the proposed approach consists of 1000 iterations. In this study, convergence is defined as the functional convergence of the defined optimization problem, where the objective function approaches a minimum value, indicating an optimal solution. The proposed approach not only starts at the lowest value but also reaches the optimal solution faster compared to other contemporary approaches. This can be attributed to the federated structure, along with the optimized solution provided by the KKT transformation for the objective function.

For further in-depth analysis of the proposed approach, 8 scenarios are considered. Herein, scenarios are defined as the combination within

Table 4
Distribution of the modulation types for different scenarios.

Scenarios	Balanced	Imbalanced	Skewed	Random	Overrepresented	Underrepresented	Partially	Sequential
Modulation Types								
BPSK	12.50%	40%	5%	15%	30%	5%	25%	0.39%
QPSK	12.50%	10%	5%	10%	10%	15%	25%	0.78%
8PSK	12.50%	10%	5%	20%	10%	15%	25%	1.56%
16QAM	12.50%	10%	5%	8%	10%	15%	25%	3.13%
64QAM	12.50%	10%	5%	12%	10%	15%	15%	6.27%
PAM4	12.50%	5%	5%	5%	10%	15%	15%	12.54%
GFSK	12.50%	5%	5%	18%	10%	15%	35%	25.09%
CPFSK	12.50%	10%	65%	12%	10%	5%	35%	50.18%

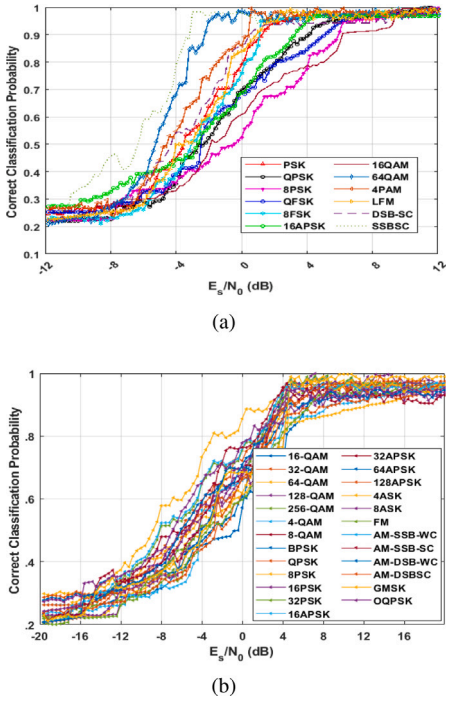


Fig. 8. CCP for standard data set (a) [20] and (b) [21].

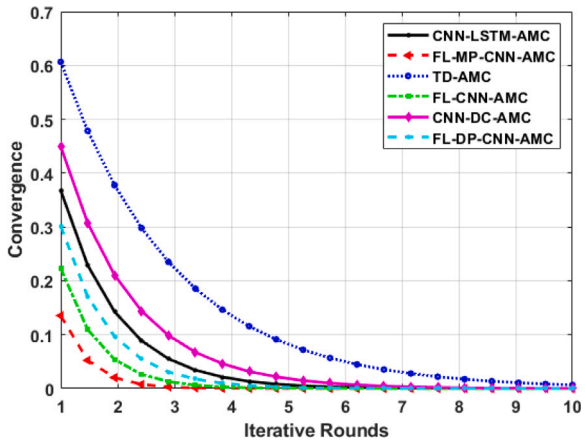


Fig. 9. Convergence comparison.

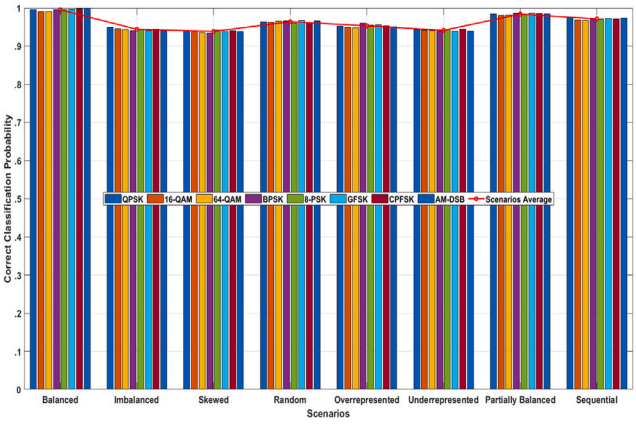


Fig. 10. Performance analysis of proposed approach against different scenarios.

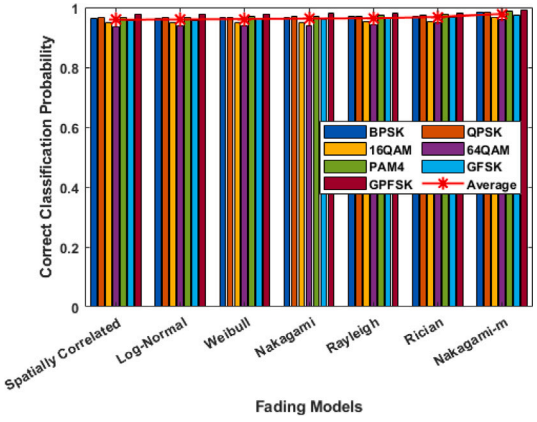
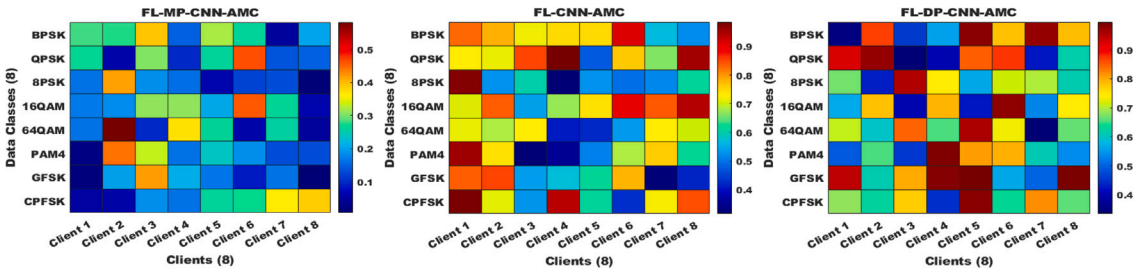


Fig. 11. CCP for different fading models.

which 8 modulation types are having different weight-age percentage wise in Table 4 as:

Balanced distribution: equal percentage are assigned to each modulation scheme. **Partially balanced distribution:** modulation schemes are divided into two groups with a balanced distribution within each group. While there is an imbalance between two groups (group 1: BPSK, QPSK, 8PSK, 16QAM. group 2: 64 QAM, PAM4, GFSK, CPFSK), where group 1 is balanced and group 2 is imbalanced. **Sequential distribution:** higher percentages are assigned to consecutive modulation schemes, introducing a sequential pattern. **Random distribution:** this scenario assigns percentages randomly. **Overrepresented modulation scheme:** one modulation scheme is assigned a higher percentage. **Imbalanced distribution:** significantly different percentages are assigned to modulation schemes, resulting in an imbalanced representation. **Underrepresented modulation scheme:** one modulation scheme is assigned a lower percentage, and **Skewed distribution:** one modulation scheme is assigned a significantly higher percentage.

From Fig. 10, the consistency of CCP remains remarkably robust across all scenarios, as evidenced by the consistent "scenario average" behavior. Despite the diverse range of scenarios, the proposed approach demonstrates exceptional efficiency, adeptly adapting to each scenario and maintaining a consistently high level of performance. This unequivocally establishes the proposed approach as the optimal choice for



(a) FL-MP-CNN-AMC (b) FL-CNN-AMC (c) FL-DP-CNN-AMC

Fig. 12. Communication overhead comparison.

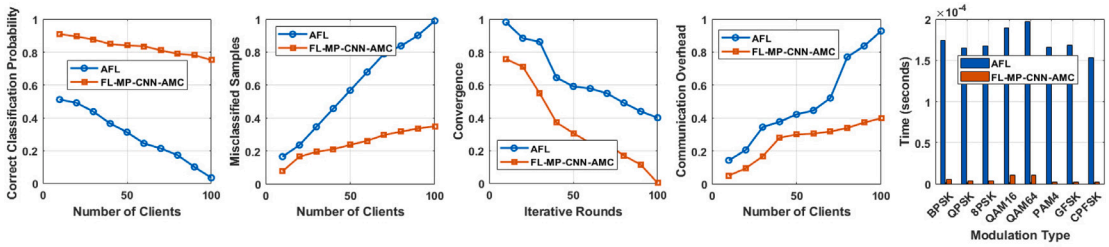


Fig. 13. Comparison between asynchronous FL (AFL) and proposed approach.

attaining reliable and efficient results across a spectrum of challenging scenarios.

In a comprehensive analysis, the proposed approach undergoes scrutiny across a spectrum of channel conditions, encompassing various fading models such as spatially correlated, Log-normal, Weibull, Nakagami, Rayleigh, Rician, and Nakagami-m, as depicted in Fig. 11. Remarkably, the CCP remains consistently high across all these models. Notably, even in the face of challenging channel conditions posed by spatially correlated, Log-normal, and Weibull fading models, the CCP exhibits resilience and consistency. This robust performance underscores the proposed approach's capacity to navigate through adverse channel conditions, substantiating its capability to deliver a consistently high CCP.

The communication overhead in terms of data exchanged (modulation classes) between clients and server for FL approaches is shown in Fig. 12, as a heatmap, between number of clients and modulation classes. The color of each cell in the heatmap represents the communication overhead of each FL approach for average of 20 rounds. For the proposed approach in Fig. 12A, the communication overhead is consistently low for all the clients across all the modulation classes as compared to Fig. 12B (FL-CNN-AMC) and Fig. 12C (FL-DP-CNN-AMC). This clearly substantiate that the communication overheads in the modulation classification problem is improved by utilizing a modified loss function. The modified loss function handles the class imbalance and improve the overall accuracy of the model. Moreover federated approach optimizes the communication rounds between clients and the server. These two techniques worked together to effectively reduce communication overheads and improve the efficiency and accuracy of the proposed approach.

Proposed approach is compared with AFL in Fig. 13. The proposed approach consistently outperforms AFL in terms of CCP, misclassified samples, convergence, and communication overhead. With the increase in the number of clients, the proposed approach demonstrates only marginal reduction in CCP, showcasing its robustness. In stark contrast, AFL experiences an almost exponential reduction in CCP as the client count increases. The proposed approach maintains its efficacy even as the client base expands. Notably, the comparison extends to misclassified samples and communication overhead, revealing a modest increase for the proposed approach with a growing number of clients. AFL, on the other hand, exhibits a substantial and nearly exponential surge in misclassified samples and communication overhead, emphasizing its limitations in scalability. Furthermore, the examination delves into the convergence aspect, encompassing 100 iterative rounds, each comprising 1000 iterations. The proposed approach demonstrates rapid and consistent convergence throughout, underscoring its efficiency. In contrast, AFL grapples with convergence issues, further highlighting the superior performance and adaptability of the proposed approach. The time complexity for the proposed approach is given as $O(1) + O(T) + O(N_c \cdot D \cdot f) + O(N_c \cdot D) + O(N_c \cdot \rho) + O(T \cdot h)$, where, $O(1)$ is broadcasting initial model parameters and schedule details, $O(T)$ is selecting a subset of clients for each training round, $O(N_c \cdot D \cdot f)$ is gradient computation for each data sample, $O(N_c \cdot D)$ is local model parameter update using gradient descent, $O(N_c \cdot \rho)$ is local model parameter upload and sharing, and $O(T \cdot h)$ is aggregating local parameters using weighted average. The time complexity of the proposed approach is consistently low in comparison to AFL. This further substantiates the efficiency of the proposed approach.

6. Conclusion

The proposed method FL-MP-CNN-AMC addresses the issues faced by traditional DL-based AMC approaches by using federated learning and incorporating the effect of multipath channels in the modulation classification process. The use of federated learning reduces the risk of data loss and enables the model to be trained on a diverse set of data from local clients. Additionally, the proposed method addresses the class imbalance problem caused by multipath channels by introducing a modified loss function. Through extensive simulation results, we have demonstrated that the proposed approach provides excellent performance in terms of CCP, convergence and communication overhead, significantly outperforming existing FL-based AMC, CNN-based AMC methods and also AFL. The proposed approach also takes into account various parameters such as interference level, delay spread, scattered and reflected paths, phase and frequency offset, and reflects the true nature of the communication channel.

CRediT authorship contribution statement

Sanjay Bhardwaj: Conceptualization, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Da-Hye Kim:** Data

curation, Formal analysis, Writing – review & editing. **Dong-Seong Kim:** Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported by Priority Research Centers Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (2018R1A6A1A03024003) and the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2022-2020-0-01612) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

References

[1] S. Peng, S. Sun, Y.-D. Yao, A survey of modulation classification using deep learning: Signal representation and data preprocessing, IEEE Trans. Neural Netw. Learn. Syst. (2021).
[2] H.S. Ghanem, R.M. Al-Makhlasawy, W. El-Shafai, M. Elsabrouty, H.F. Hamed, G.M. Salama, F.E.A. El-Samie, Wireless modulation classification based on Radon transform and convolutional neural networks, J. Ambient Intell. Humaniz. Comput. (2022) 1–10.
[3] Y. Wang, Q. Lu, Y. Jin, H. Zhang, Communication modulation signal recognition based on the deep multi-HoP neural network, J. Franklin Inst. B 358 (12) (2021) 6368–6384.
[4] M. Bkassiny, Y. Li, S.K. Jayaweera, A survey on machine-learning techniques in cognitive radios, IEEE Commun. Surv. Tutor. 15 (3) (2012) 1136–1159.
[5] F. Hameed, O.A. Dobre, D.C. Popescu, On the likelihood-based approach to modulation classification, IEEE Trans. Wirel. Commun. 8 (12) (2009) 5884–5892.
[6] T. Huynh-The, Q.-V. Pham, T.-V. Nguyen, T.T. Nguyen, R. Ruby, M. Zeng, D.-S. Kim, Automatic modulation classification: A deep architecture survey, IEEE Access 9 (2021) 142950–142971.
[7] M.A. Abdel-Moneim, W. El-Shafai, N. Abdel-Salam, E.-S.M. El-Rabaie, F.E. Abd El-Samie, A survey of traditional and advanced automatic modulation classification techniques, challenges, and some novel trends, Int. J. Commun. Syst. 34 (10) (2021) e4762.
[8] S. Wang, M. Chen, C. Yin, W. Saad, C.S. Hong, S. Cui, H.V. Poor, Federated learning for task and resource allocation in wireless high-altitude balloon networks, IEEE Internet Things J. 8 (24) (2021) 17460–17475.
[9] Q. Guo, F. Tang, N. Kato, Federated reinforcement learning-based resource allocation in D2D-enabled 6G, IEEE Netw. (2022).
[10] H.-S. Lee, D.-E. Lee, Resource allocation in wireless networks with federated learning: Network adaptability and learning acceleration, ICT Express 8 (1) (2022) 31–36.
[11] J. Zheng, K. Li, N. Mhaisen, W. Ni, E. Tovar, M. Guizani, Exploring deep reinforcement learning-assisted federated learning for online resource allocation in privacy-preserving EdgeIoT, IEEE Internet Things J. (2022).
[12] H. Zhang, H. Zhou, M. Erol-Kantarci, Federated deep reinforcement learning for resource allocation in O-RAN slicing, 2022, arXiv preprint arXiv:2208.01736.
[13] T.J. O'shea, N. West, Radio machine learning dataset generation with GNU radio, in: Proceedings of the GNU Radio Conference, vol. 1, (no. 1) 2016.
[14] E. Blossom, GNU radio: Tools for exploring the radio frequency spectrum, Linux J. 2004 (122) (2004) 4.
[15] J.M. Johnson, T.M. Khoshgoftaar, Survey on deep learning with class imbalance, J. Big Data 6 (1) (2019) 1–54.
[16] J. Zhang, C. Li, J. Qi, J. He, A survey on class imbalance in federated learning, 2023, arXiv preprint arXiv:2303.11673.
[17] J. Liu, Y. Ma, R. Tafazolli, A spatially non-stationary fading channel model for simulation and (Semi-) analytical study of ELAA-MIMO, IEEE Trans. Wirel. Commun. (2023).
[18] A. Soulimani, M. Benjillali, H. Chergui, D.B. da Costa, On multihop Weibull-fading communications: performance analysis framework and applications, 2016, arXiv preprint arXiv:1610.08535.
[19] H. Sajjad, M. Jamil, Error rate performance of multi-HoP communication systems over nakagami-m fading channel, 2012.
[20] V.-S. Doan, T. Huynh-The, V.-P. Hoang, D.-T. Nguyen, MoDANet: Multi-task deep network for joint automatic modulation classification and direction of arrival estimation, IEEE Commun. Lett. 26 (2) (2021) 335–339.
[21] T.J. O'Shea, T. Roy, T.C. Clancy, Over-the-air deep learning based radio signal classification, IEEE J. Sel. Top. Sign. Proces. 12 (1) (2018) 168–179.
[22] P. Ghasemzadeh, S. Banerjee, M. Hempel, H. Sharif, Performance evaluation of feature-based automatic modulation classification, in: 2018 12th International Conference on Signal Processing and Communication Systems, ICSPCS, IEEE, 2018, pp. 1–5.
[23] Y. Wang, G. Gui, H. Gacanin, B. Adebisi, H. Sari, F. Adachi, Federated learning for automatic modulation classification under class imbalance and varying noise condition, IEEE Trans. Cogn. Commun. Netw. 8 (1) (2021) 86–96.
[24] J. Shi, H. Zhao, M. Wang, Q. Tian, Signal recognition based on federated learning, in: IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, IEEE, 2020, pp. 1105–1110.
[25] X. Fu, G. Gui, Y. Wang, T. Ohtsuki, B. Adebisi, H. Gacanin, F. Adachi, Lightweight automatic modulation classification based on decentralized learning, IEEE Trans. Cogn. Commun. Netw. 8 (1) (2021) 57–70.
[26] Z. Zhang, H. Luo, C. Wang, C. Gan, Y. Xiang, Automatic modulation classification using CNN-LSTM based dual-stream structure, IEEE Trans. Veh. Technol. 69 (11) (2020) 13521–13531.
[27] O.A. Dobre, Y. Bar-Ness, W. Su, Higher-order cyclic cumulants for high order modulation classification, in: IEEE Military Communications Conference, 2003. MILCOM 2003, vol. 1, IEEE, 2003, pp. 112–117.
[28] O.A. Dobre, A. Abdi, Y. Bar-Ness, W. Su, Cyclostationarity-based modulation classification of linear digital modulations in flat fading channels, Wirel. Pers. Commun. 54 (4) (2010) 699–717.
[29] A. Ebrahimzadeh, R. Ghazalian, Blind digital modulation classification in software radio using the optimized classifier and feature subset selection, Eng. Appl. Artif. Intell. 24 (1) (2011) 50–59.
[30] D. Boutte, B. Santhanam, A hybrid ICA-SVM approach to continuous phase modulation recognition, IEEE Signal Process. Lett. 16 (5) (2009) 402–405.
[31] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[32] M.W. Aslam, Z. Zhu, A.K. Nandi, Automatic modulation classification using combination of genetic programming and KNN, IEEE Trans. Wirel. Commun. 11 (8) (2012) 2742–2750.
[33] V. Orlic, M. Dukic, Multipath channel estimation algorithm for automatic modulation classification using sixth-order cumulants, Electron. Lett. 46 (19) (2010) 1.
[34] S. Bhardwaj, R.R. Ginanjar, D.-S. Kim, Deep Q-learning based resource allocation in industrial wireless networks for URLLC, IET Commun. 14 (6) (2020) 1022–1027.
[35] A.M. Elbir, K.V. Mishra, Joint antenna selection and hybrid beamformer design using unquantized and quantized deep learning networks, IEEE Trans. Wireless Commun. 19 (3) (2019) 1677–1688.
[36] A.M. Elbir, CNN-based precoder and combiner design in mmWave MIMO systems, IEEE Commun. Lett. 23 (7) (2019) 1240–1243.
[37] Z. Zhang, C. Wang, C. Gan, S. Sun, M. Wang, Automatic modulation classification using convolutional neural network with features fusion of SPWVD and BJD, IEEE Trans. Signal Inf. Process. Netw. 5 (3) (2019) 469–478.
[38] M. Wang, Y. Fan, S. Fang, T. Cui, D. Cheng, A joint automatic modulation classification scheme in spatial cognitive communication, Sensors 22 (17) (2022) 6500.
[39] A. Khan, A. Sohail, U. Zahoora, A.S. Qureshi, A survey of the recent architectures of deep convolutional neural networks, Artif. Intell. Rev. 53 (8) (2020) 5455–5516.
[40] I.H. Sarker, Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions, SN Comput. Sci. 2 (6) (2021) 1–20.
[41] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436–444.
[42] X. Zhang, H. Zhao, H. Zhu, B. Adebisi, G. Gui, H. Gacanin, F. Adachi, NAS-AMR: Neural architecture search based automatic modulation recognition for integrated sensing and communication systems, IEEE Trans. Cogn. Commun. Netw. (2022).
[43] C. Hou, G. Liu, Q. Tian, Z. Zhou, L. Hua, Y. Lin, Multi-signal modulation classification using sliding window detection and complex convolutional network in frequency domain, IEEE Internet Things J. (2022).
[44] M. Ma, Z. Li, Y. Lin, L. Chen, S. Wang, Modulation classification method based on deep learning under non-Gaussian noise, in: 2020 IEEE 91st Vehicular Technology Conference, VTC2020-Spring, IEEE, 2020, pp. 1–5.
[45] T. O'shea, J. Hoydis, An introduction to deep learning for the physical layer, IEEE Trans. Cogn. Commun. Netw. 3 (4) (2017) 563–575.
[46] M.M. Elsagheer, S.M. Ramzy, A hybrid model for automatic modulation classification based on residual neural networks and long short term memory, Alex. Eng. J. (2022).
[47] F. Meng, P. Chen, L. Wu, X. Wang, Automatic modulation classification: A deep learning enabled approach, IEEE Trans. Veh. Technol. 67 (11) (2018) 10760–10772.

[48] S.-H. Kim, J.-W. Kim, V.-S. Doan, D.-S. Kim, Lightweight deep learning model for automatic modulation classification in cognitive radio networks, IEEE Access 8 (2020) 197532–197541.
[49] Y. Wang, G. Gui, T. Ohtsuki, F. Adachi, Multi-task learning for generalized automatic modulation classification under non-Gaussian noise with varying SNR conditions, IEEE Trans. Wireless Commun. 20 (6) (2021) 3587–3596.
[50] S. Peng, H. Jiang, H. Wang, H. Alwageed, Y. Zhou, M.M. Sebdani, Y.-D. Yao, Modulation classification based on signal constellation diagrams and deep learning, IEEE Trans. Neural Netw. Learn. Syst. 30 (3) (2018) 718–727.
[51] P. Qi, X. Zhou, S. Zheng, Z. Li, Automatic modulation classification based on deep residual networks with multimodal information, IEEE Trans. Cogn. Commun. Netw. 7 (1) (2020) 21–33.
[52] L. Huang, Y. Zhang, W. Pan, J. Chen, L.P. Qian, Y. Wu, Visualizing deep learning-based radio modulation classifier, IEEE Trans. Cogn. Commun. Netw. 7 (1) (2020) 47–58.
[53] Y. Wang, J. Gui, Y. Yin, J. Wang, J. Sun, G. Gui, H. Gacanin, H. Sari, F. Adachi, Automatic modulation classification for MIMO systems via deep learning and zero-forcing equalization, IEEE Trans. Veh. Technol. 69 (5) (2020) 5688–5692.
[54] S. Hu, Y. Pei, P.P. Liang, Y.-C. Liang, Deep neural network for robust modulation classification under uncertain noise conditions, IEEE Trans. Veh. Technol. 69 (1) (2019) 564–577.
[55] Y. Wang, J. Yang, M. Liu, G. Gui, LightAMC: Lightweight automatic modulation classification via deep learning and compressive sensing, IEEE Trans. Veh. Technol. 69 (3) (2020) 3491–3495.
[56] A.P. Hermawan, R.R. Ginanjar, D.-S. Kim, J.-M. Lee, CNN-based automatic modulation classification for beyond 5G communications, IEEE Commun. Lett. 24 (5) (2020) 1038–1041.
[57] A. Meijerink, A.F. Molisch, On the physical interpretation of the Saleh–Valenzuela model and the definition of its power delay profiles, IEEE Trans. Antennas and Propagation 62 (9) (2014) 4780–4793.
[58] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980–2988.
[59] C. Xing, S. Xie, S. Gong, X. Yang, S. Chen, L. Hanzo, A KKT conditions based transceiver optimization framework for RIS-aided multi-user MIMO networks, IEEE Trans. Commun. (2023).
[60] P.T. Boggs, J.W. Tolle, Sequential quadratic programming, Acta Numer. 4 (1995) 1–51.
[61] O.A. Dobre, A. Abdi, Y. Bar-Ness, W. Su, Survey of automatic modulation classification techniques: Classical approaches and new trends, IET Commun. 1 (2) (2007) 137–156.
[62] J.G. Proakis, M. Salehi, Digital Communications, vol. 4, McGraw-hill New York, 2001.