SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I5P117
Volume 12 Issue 5, 200-209, May 2025

Original Article

Improved Knowledge Mapping in Heterogeneous Network Using Enhanced Federated Learning

Yelithoti Sravana Kumar[1]*, Tapaswini Samant[2], Swati Swayamsiddha[2]

[1,2]School of Electronics, KIIT Deemed to be University, Bhubaneswar, India.

**Corresponding Author : 1981083@kiit.ac.in*

Received: 12 March 2025 Revised: 14 April 2025 Accepted: 15 May 2025 Published: 27 May 2025

Abstract - *Edge networks consist of devices that analyze different data types and models. A knowledge map can be developed in device scheduling using homogeneous or heterogeneous models. In this study, a Federated learning algorithm is used to minimize communication overhead by distributing device information in batches, ensuring that all devices have an equal opportunity. The federated learning method is utilized in this study to allocate spectrum between primary and secondary users by categorizing them using standard parameters such as bandwidth utilization, energy, and a novel metric known as signal-to-noise ratio. Here, the spectrum is allocated using the deep learning technique. However, in other algorithms, global loss minimization was not considered for the model analysis. Fortunately, in this study, model analysis was carried out using deep learning architectures such as Convolution Neural Networks for feature extraction, pooling layers for downsampling, and accessing the performance using evaluation metrics. The findings indicated that knowledge mapping could also be improved by improving the model.*

Keywords - *Heterogeneous network, 5G, Deep learning, Knowledge mapping, Cognitive radio network, Spectrum allocation, Federated Learning.*

1. Introduction

Many different devices can now communicate with one another and share data through a network; this data is then saved on a central server, which is known as an edge server. Multiple devices comprise the edge network, each conducting analysis using a unique combination of models and data. Each device should have a suitable schedule for using the edge network's resources in such situations. A knowledge map can be used to achieve the goal of accurately scheduling the devices. This knowledge map can be formatted in a way that works for either homogeneous or heterogeneous models.

Homogenous models will keep the data and machine learning the same across all devices. In circumstances like this, the knowledge map can be readily framed. While the heterogeneous models allow for unique data and machine learning models on each device. When this happens, the process of forming the knowledge map will be challenging. Federation learning can overcome this obstacle, which is the construction of a knowledge map in a heterogeneous network.

In [1, 2], a strategy for heterogeneous networks based on load balancing and using a traditional scheduling approach was suggested. In [3], the scheduling was improved even further by using the network behaviour gleaned by the double deep Q learning technique. While in [4], the difficult handoff that occurs in a heterogeneous network is reduced by applying user experience and load-balancing approaches. The Q learning is improved in [5] using the user experience during scheduling. In [6], along with scheduling, minimizing bandwidth consumption also involved employing caching within the base stations. In [7], the optimization method contributes to an even higher level of improvement in device-to-device communication. In [8, 9, 10], a concise assessment of the functions that machines and deep learning algorithms play in wireless networks was presented.

The algorithms described above assessed the properties of the device networks primarily for the purpose of scheduling. However, in federated learning, in addition to the node features, the purpose of the device is also studied in order to ensure resource allocation and usage most effectively. In light of what was found in [11], a combination of game theory and federated learning was used to schedule devices. On the other hand, [12] scheduling events using the network service and its characteristics. The cache was used by [13] in the same way as [6] did, but it integrated with user properties and made use of a separate policy for sharing the information. For wireless networks, [14, 15] conducted a cursory investigation into the federated learning strategy. In

[16], a hierarchical game-theoretic framework for optimizing edge association and resource allocation in Hierarchical Federated Learning (HFL) systems. In [17], a two-layer Federated Learning (FL) framework is tailored for 6G-enabled Internet of Vehicles (IoV) environments. The proposed model leverages a distributed end-edge-cloud architecture to enhance learning efficiency and accuracy while preserving data privacy and minimizing communication overhead.

The deployment of Federated Learning (FL) in energy-harvesting wireless networks, where base stations equipped with massive Multiple-Input Multiple-Output (MIMO) systems serve users powered by independent energy harvesting sources [18]. In [19], it provides an overview of Federated Learning (FL), focusing on its types, architectures, challenges, and potential applications. It discusses various FL paradigms, including horizontal, vertical, and transfer learning, and examines client-server and peer-to-peer architectures.

Explores the concept of Knowledge-Defined Networking (KDN), an architectural paradigm that integrates Software-Defined Networking (SDN), network telemetry, and Machine Learning (ML) to achieve autonomous and intelligent network management, particularly in the context of future 6G wireless networks [20]. In [21], it introduces a Knowledge-Aided Federated Learning (KFL) framework tailored for energy-constrained wireless networks. Unlike traditional federated learning, which requires devices to share entire model parameters, KFL enables devices to independently design their machine learning models and share only high-level data features, termed "knowledge".

In [22], it addresses the challenges posed by client heterogeneity in Federated Learning (FL) systems, which can lead to increased training latency and straggling during server aggregation. In [23], it introduces a Federated Learning (FL) framework that incorporates a lightweight differential privacy mechanism to enhance data security while maintaining model performance. A comprehensive reflection on the current state and future directions of Federated Learning (FL) as applied in practical scenarios [24].

By modelling worker behaviour through evolutionary game theory and leveraging a Stackelberg differential game for incentive mechanisms, the study enables dynamic, decentralized coordination among workers, edge servers, and the model owner.

The proposed federated learning distinguishes itself from other schemes and has the following impacts. The federated learning approach combines both resource utilization and network energy efficiency for scheduling the devices. However, it is mostly applied to homogeneous models. Only a few works were analyzed for heterogeneous models with two or three layers.

- It is also mostly used to combine the model parameters for the classification process.
- The SNR was also not included in the analysis.
- The federation algorithm is also not applied for device scheduling.

As a result, an optimal federated learning technique for device scheduling in a whole heterogeneous network employing a deep learning approach is proposed. In addition to that, it incorporated SNR and Model error for the federated learning that was performed in device scheduling.

The following outline should help you understand how the paper is structured: The mechanisms for scheduling the operation of the devices are discussed in Section 2. In sections 3 and 4, a concise explanation of the procedures involved in the suggested method and the outcomes of using it has been shown. In section 5, a summary of the suggested method's advantages is provided, and in part 6, the method's potential applications are discussed.

2. Literature Review

Fletscher et al. (2018) utilized a predictive controller for energy utilization to operate the heterogeneous wireless sensor network. Here, the heterogeneous network was operated using grid and renewable energy sources. In that, the author proposed a model predictive controller to estimate the network load and utilize energy from renewable instead of the grid. This approach helps minimize grid utilization but can be further enhanced by using the routing and scheduling of data.

Huang et al. (2018) also proposed a load-balancing concept for the femtocells of the heterogeneous networks. However, they utilized energy efficiency and load balancing by gathering the device information from the base station. Then, the base station employed dynamic switching off and on of femtocells using the incoming load and its own node parameters. Its performance can be enhanced further by using optimization and learning algorithms.

Zhao et al. (2018) employed multiple algorithms to schedule the nodes in the heterogeneous network. Here, they employed multi-agents to gather information from the nodes. Then, it utilized a double deep Q reinforcement algorithm to learn the network behavior for optimal scheduling. This performance is good compared to the existing one, but its computational time is high because it uses multiple algorithms.

Kobayashi et al. (2018) proposed a new approach for efficient transmission in heterogeneous networks. Here, they employed user experience, load balancing, and network properties to select the network's access points. Because the access points should possess higher energy and processing speed for scheduling, they proposed user experience-based access point selection to minimize the hard handoff.

Wang et al. (2019) utilized the distributed Q-learning network instead of double Q-learning for scheduling. However, they employed network service as one of the parameters for node scheduling. This approach is designed only for the two-tier heterogeneous network.

Haw et al. (2019) proposed a new approach for saving the bandwidth along with the node's energy. They employed the cache process in the base station to save the most popular information. This helps to reduce the multiple searches for data in the network. Overall, it helps the network's node energy and bandwidth.

Chen et al. (2019) proposed an optimization algorithm for efficient spectrum utilisation by the devices in heterogeneous networks. Because in a heterogeneous network, the communication between the devices is highly difficult as it utilizes most of the higher spectrum. Hence, this problem was overcome using an optimization algorithm with different propagation conditions and switching mechanisms between micro and millimeter wavebands.

Lim et al. (2020) proposed a federated learning approach for data transmission in heterogeneous networks. In this, they employed learning to load the model parameters from different devices without saving the data.

This helps to preserve the original information. However, the game theory approach is used to describe the user nature. By using federated learning and game theory, the information is shared securely.

Zhang et al. (2020) employed the deep learning algorithm with the Lagrange decomposition technique to allocate the power for the devices in the heterogeneous network. Here, the allocation is based on node energy efficiency and then network service. Using these two pieces of information, power and bandwidth will be allocated to the devices in the network.

Li et al. (2020) also employed the cache process to preserve the resource bandwidth. However, they employed a smart way of caching by using the user experience and the device properties.

Lim et al. (2021) enhanced their federated learning by using two layered architectures in a hierarchical manner for a vehicle-based heterogeneous network. In this case, the device selection is also based on the game theory for both layers.

Zhou et al. (2021) also proposed a two-layered federated network for vehicle networks. However, they employed a distributed learning pattern for device scheduling and reducing the communication overhead.

Hamdi et al. (2021) also employed the federated learning approach to minimize grid power utilization by harvesting more energy from renewable energy sources. This minimizes the grid power utilization and enhances the network size by employing more nodes for efficient communication. Singh et al. (2022) and Ashtari et al. (2022) analyzed the federated learning and knowledge mapping in wireless networks.

2.1. Research Gap

Based on the above analysis, the following points were observed

- The federated learning approach combines resource utilization and network energy efficiency to schedule the devices.
- But it is mostly applied to homogeneous models. Only a few works were analyzed for heterogeneous models with two or three layers.
- It is also mostly used to combine the model parameters for the classification process.
- The SNR was also not included in the analysis.
- The federation algorithm is also not applied for device scheduling.

3. Proposed Method

In this study, federated learning was used to distribute spectrum to secondary users in homogenous and heterogeneous networks. This allowed the researchers to overcome obstacles such as bandwidth utilization, energy consumption, and signal-to-noise ratio. The allocation of communication channels is accomplished through the use of device scheduling. In this scenario, the different devices each make use of a deep learning algorithm in order to complete the recognition process.

3.1. System Model

The proposed algorithm is tested on an edge network consisting of N devices connected to an edge server, as shown in Figure 1. It illustrates the system architecture used in the proposed federated learning framework for spectrum allocation. The model consists of an edge server and multiple edge devices (e.g., Device 1, Device 2, ..., Device N).

3.1.1. Edge Server

It acts as the central coordinating unit that collects local model updates from scheduled devices, aggregates the updates to improve a global model, and broadcasts the updated model back to the devices.

3.1.2. Edge Devices

These are the participating edge nodes (e.g., IoT devices, mobile devices) that possess local data and perform individual deep-learning tasks. Each device trains a model locally and may be scheduled or unscheduled in each training round.

3.1.3. Scheduled Devices

Actively participate in federated learning during the current round by sharing their local updates with the edge server.

3.1.4. Unscheduled Devices

Do not participate in that particular round due to constraints like low battery, poor network conditions, or system scheduling policies.

3.1.5. Local Knowledge

It Represents the individual model updates or insights generated by each device based on its local dataset

3.1.6. Device Selection and Global Broadcasting

This denotes the communication flow from the edge server, where selected devices are chosen to contribute to the global model, and the aggregated global model is broadcast back to them.

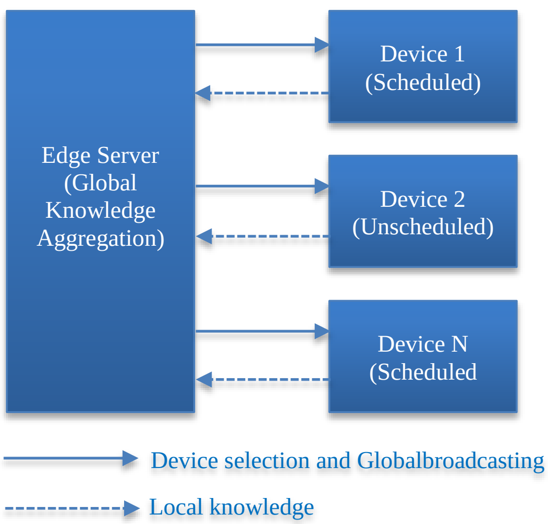

Fig. 1 Heterogeneous network model

Each individual device is denoted as 1, 2, 3 and N devices. The device notation is as follows:

$$\text{Device notation}=1,2,3...N \tag{1}$$

Each device has different datasets and classification processes. The datasets in the devices are noted as follows:

$$\text{Dataset}=D1, D2, D3...DN \tag{2}$$

The corresponding classification process is as follows:

$$\text{Task}=T1, T2, T3...TN \tag{3}$$

There is no overlapping between the datasets in the devices, and the dataset samples in the tasks are as follows.

$$Dk, T=|Dk, T| \tag{4}$$

With this information, the federated learning process allocates the bandwidth for a device to share its information based on its energy and signal-to-noise ratio values.

3.2. Federated Learning

This paper aims to minimise the overhead communication in the edge server while updating the device information. Hence, the communication overhead is reduced by sharing their information in batches.

This helps reduce the communication overhead and gives all communication devices equal chances. The steps in federated learning are as follows:

3.2.1. Selection

In this algorithm, the devices communicate circularly, as indicated by the parameter

$S_{n,r}$. The selection of a device is as follows:

$$S_{n,r} = \begin{cases} 1 \text{ if device n is selected} \\ 0; \text{otherwise} \end{cases} \tag{5}$$

In this, the r indicates the rounds 1, 2, 3… R rounds, and n indicates the devices.

3.2.2. Broadcast

Once the devices are selected, the edge server will start to broadcast the information between the selected devices. This process is called the global knowledge update process and is denoted using G_r.
This global update in terms of tasks is given as follows:

$$G_r = G_{1,r}, G_{2,r}, \dots, G_{t,r} \tag{6}$$

Here, the t denotes the tasks in each device; the individual tasks of each device will be updated in all the scheduled devices in each round for training.

3.2.3. Training

Once the devices receive the global knowledge, they start to update their local feature and its corresponding predictor using the following equations:

$$f_{n,r,l+1} = f_{n,r,l} - \eta_u(\nabla_u F_n\left(f_{n,r,l}, p_{n,r,l}\right) + \lambda \nabla L_n\left(f_{n,r,l}\right)) \tag{7}$$

In this, ∇ is the gradient operator, Ln is the local knowledge of own device, η_u is the feature extractor learning rate, (f, p) is the loss of the feature extractor. The corresponding predictor is as follows:

$$p_{n,r,l+1} = p_{n,r,l} - \eta_v(\nabla_v F_n(f_{n,r,l}, p_{n,r,l}) + \lambda \nabla L_n(f_{n,r,l})) \quad (8)$$

In this, ∇ is the gradient operator, Ln is the local knowledge of own device, ηv is the feature extractor learning rate, (f, p) is the loss of the feature extractor. To balance the predictor and the feature training process, the λ is used.

3.2.4. Mapping

After the completion of training in scheduled devices, each device predicts the class with the gained knowledge and data. Again, the knowledge will be updated and broadcast between the devices. The mapped knowledge is given in the following equation:

$$G_{n,t,r+1} = \frac{1}{D_{n,t}} \sum_{x,y \in D_n, r} h_n(u_{n,r+1}; x) \quad (9)$$

The overall knowledge mapping for all tasks is denoted as follows:

$$G_{n,r+1} = (G_{n,1,r+1}, \dots . G_{n,t,r+1}) \quad (10)$$

3.2.5. Aggregation

Once the round is completed, the knowledge gathered from the devices will be updated in the edge server using the following formula:

$$G_{t,r+1} = \frac{\sum_{n \in s_r} D_{n,r} G_{n,t,r+1}}{\sum_{k \in s_r} D_{n,r}} \quad (11)$$

With these steps, the edge network updates the information for all users. While training, the loss factor is important for efficient training and final knowledge mapping. This feature and predictor loss calculation are explained below

3.3. FL Training Metric

In this, the federated Learning algorithm optimizes its training process by minimising its loss function. The model error will be the only loss function for a normal model. But in this, the different types of models were combined; hence, the FL also optimize its training by minimizing two losses called

- Model error loss
- Knowledge aided loss

3.3.1. Model Error Loss

The term model error loss indicates the individual model error by the individual device with the collected feature from the edge network. Because the device applies its model using the features from the FL process. Hence, this loss should be minimized for efficient knowledge mapping. This formula is denoted in equation 7.

3.3.2. Knowledge Aided Loss

This loss function denotes the overall knowledge model loss in equation 9. Knowledge-aided loss minimization also helps enhance the final knowledge mapping.

With these loss parameters, the federated learning will improve its learning process for device scheduling.

3.3.3. Problem Formulation

The proposed federated learning will improve its learning using the model and knowledge loss function optimization for individual devices. However, this loss function will not be sufficient for the overall network. Hence, in addition to this loss function, the following parameters are also considered for the learning process:

- Energy consumption
- Bandwidth
- SNR of device
- Round time
- Device selection

Based on these above factors, the device can be scheduled for knowledge mapping in the federated learning process. The notation and boundary of the values is as follows:

Table 1. Notations of problem formulation

Term	Notation	Definition	Value
Energy	E	Energy consumed by devices for knowledge sharing	$E_n < E\_max$ (12)
Band width	B	Bandwidth occupied by all devices and individual devices	$B_n \& B_N < B$ (13)
SNR	SR	The signal-to-noise ratio of the overall device and individual device	$SR_n \& SR_N > SR\_min$ (14)
Time	T	The total time for one round by each device	$T_n < T\_max$ (15)
Device Selecti on	DS	The same device cannot be selected for all rounds in a continuous manner	$DS_n < DS\_max$ (16)

The edge network also updates these metrics for device scheduling in the federated learning process. These metrics help to utilize all device knowledge in edge networks to create a well-equipped network for information sharing.

3.4. Device Scheduling

As the network consists of N number of devices, the proposed problem cannot be solved using single convex problem optimization. Hence, in this, the problem is formulated as non-convex optimization and solved using the Lyapunov optimization process in21. Using those steps, the devices are scheduled. The device will be scheduled for deep learning using the following process.

Step : 1 Begin
Step : 2 Load network and device properties for federated learning.
Step : 3 Perform the first round of federated learning with initialized parameters.
Step : 4 Compute global loss and other problem formulation metrics.
Step : 5 Analyze the bandwidth allocation using Lyapunov optimization.
Step : 6 Allocate bandwidth utilization of individual devices and the overall network.
Step : 7 Compute global loss.
Step : 8 If equation 13 is satisfied, the energy scheduling process will be takes place.
Step : 9 Like bandwidth, the energy and SNR are also calculated using Lyapunov optimization.
Step : 10 if equations 12 and 14 are satisfied along with equation 16, the device can be selected for the next round.
Step : 11 Otherwise, the device will not be included for learning.
Step : 12 Once all the rounds are completed, the global loss will be calculated, and the knowledge map will be updated.
Step : 13 Stop.

With these steps, the devices are scheduled for the federated learning and the knowledge map is built.

4. Results and Discussion

In this, the proposed federated learning was tested on the MNIST dataset using simulation software. The proposed method performance was evaluated in two scenarios as follows:

- Homogeneous model
- Heterogeneous model

In a homogeneous model, the same deep learning algorithm was applied to all the devices in the network. While in the heterogeneous model, a different model architecture was used for analysis. But, in both the models, the deep learning algorithm was used. Both models' performance was analyzed regarding accuracy versus the number of rounds. Because in each round, the device selection will be different and its performance will also be different. The simulation results of the deep learning algorithm in a heterogeneous network model are shown in the following figures.

The device performance is transmitted through channels using federated learning. The channel parameters are shown in the figure. It illustrates the distribution of the H_0 test statistic used for channel selection in the proposed federated learning framework. The figure contains two subplots representing the real and imaginary components of the H_0 statistic.

The real component (top subplot) shows a symmetric, bell-shaped distribution centered around zero, resembling a Gaussian distribution. This reflects low-magnitude noise under the null hypothesis, indicating no primary signal.

The imaginary component (bottom subplot) exhibits a similar pattern, further confirming the presence of complex Gaussian noise in the channel.

These statistical insights help distinguish occupied and unoccupied spectrum bands, enabling efficient and dynamic channel allocation. Federated learning is used for local analysis, allowing devices to share only model updates with the edge server, thereby preserving privacy and reducing communication overhead.

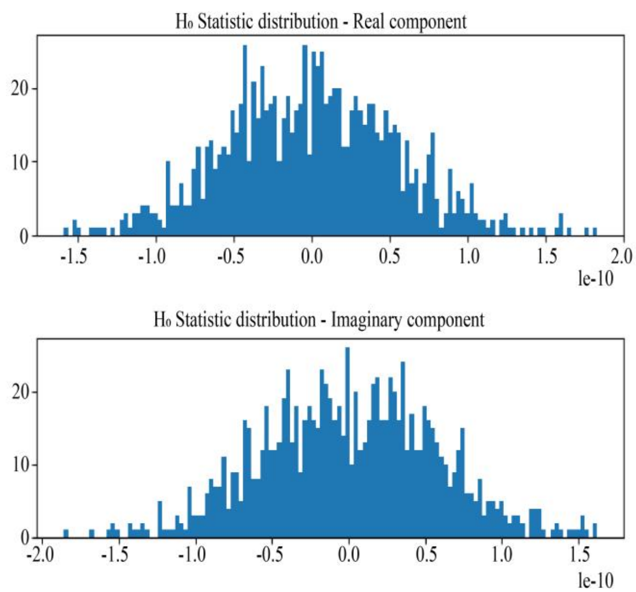

Fig. 2 Channel selection

Using the federated learning, the power control value, the resource allocation and energy threshold are shown in the below figures.

The performance of a cyclostationary detector at different SNR values is displayed in Figure 3. The theoretical false alarm probability computed false alarm probability P_{FA}, and estimated probability of detection P_D are plotted. While P_{FA} is rather stable and closely resembles the theoretical P_{FA}, P_D greatly improves as SNR rises, approaching 1.

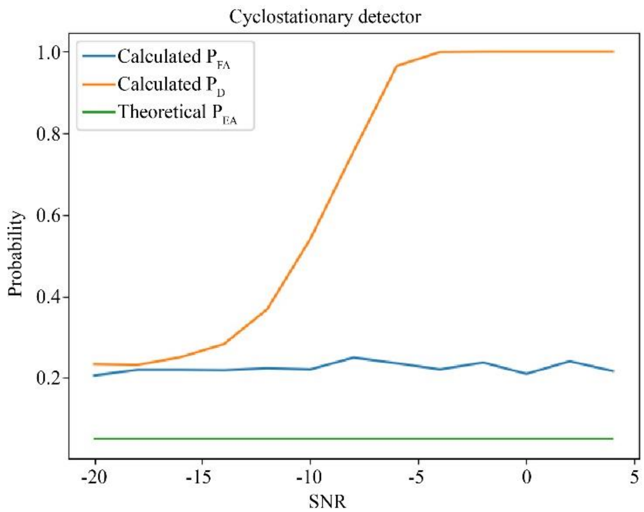

Fig. 3 Cyclostationary detection at power controlvalues

The performance of an energy detector at different SNR levels is shown in Figure 4. It displays the theoretical and computed odds of false alarm P_{FA} and detection P_D. P_{FA} is steady and around the expected value the entire time, whereas P_D rises dramatically and matches the theoretical curve as SNR increases.

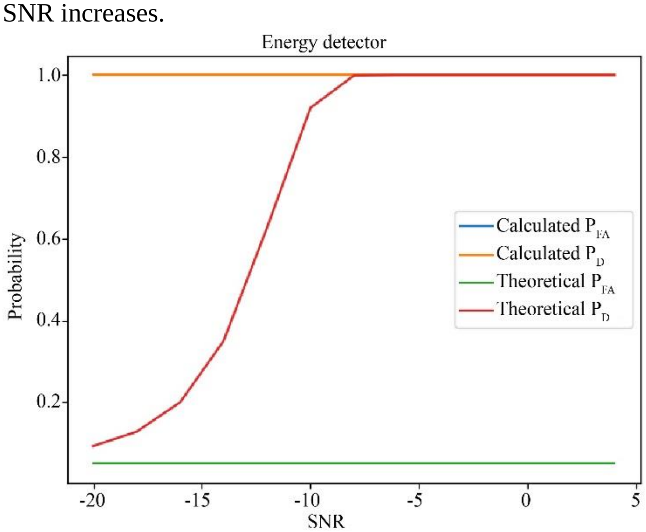

Fig. 4 Energy detection at power control values

Figure 5 shows the average time and energy usage for the three algorithms ANN-ROF, SaROF, and CNN-FR over various uplink spectrum resources for bandwidth allocation. CNN-FIT consistently demonstrates lower time and energy consumption, but SARCF typically displays the highest results, particularly at lower resource allocations.

Figure 6 shows a comparison of energy detector thresholds versus SNR for two methods: CNN_CR Threshold and Bayes Threshold. Both thresholds decrease as SNR increases, with the CNN_CR threshold slightly higher than the Bayes threshold at lower SNR values.

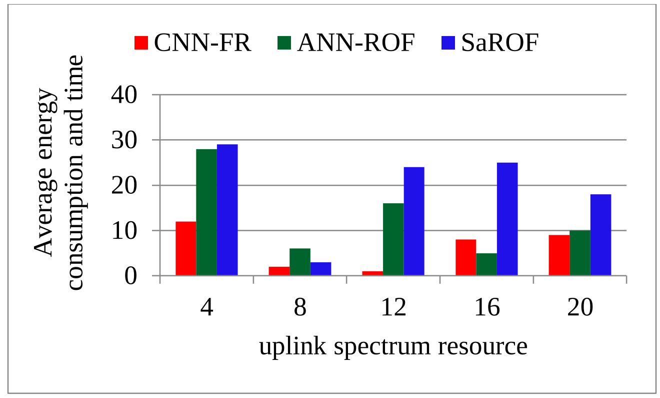

Fig. 5 Bandwidth allocation

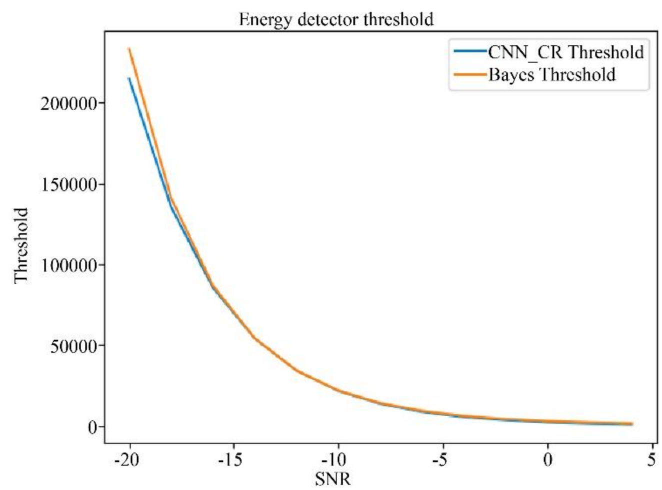

Fig. 6 Energy allocation

4.1. Homogeneous Model

In homogeneous model, all the devices in the edge network utilize the same model. As it utilized the same model, the proposed device scheduling reaches its convergence earlier and its accuracy remains the same for all the scheduling patterns.

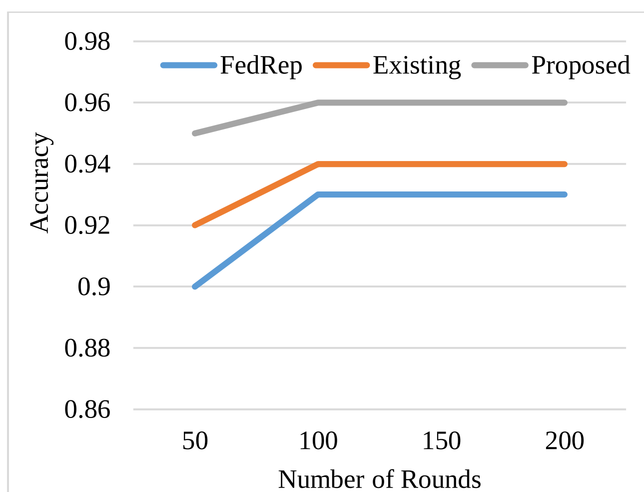

Fig. 7 Homogeneous model performance

In this, the homogeneous model performance was analyzed for fifty devices with the same deep learning algorithm for all devices. As all the devices generate the same loss, the scheduling algorithm performance will remain the same for all the models.

4.2. Heterogeneous Model

Each device uses a different deep learning algorithm for analysis in heterogeneous models. Hence, the model performance will differ, and the scheduling has a significant role in the knowledge mapping. The proposed model performance for the heterogeneous model is shown in the below figure.

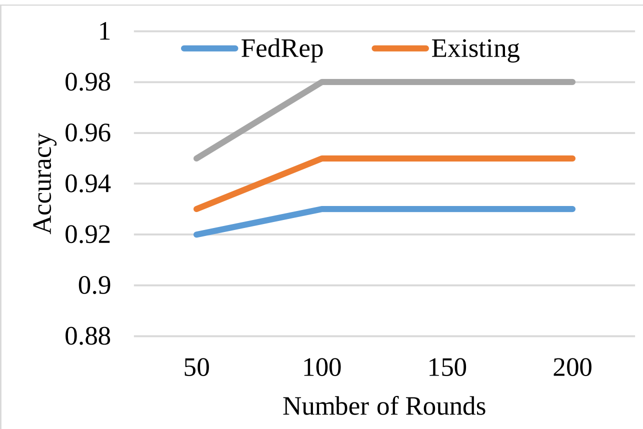

Fig. 8 Heterogeneous model performance

In this, the homogeneous model performance was analyzed for fifty devices with different deep learning algorithms for all devices. As the models differ, the scheduling performance will help enhance the knowledge mapping by minimizing global loss and meeting the device requirements stated in the FL Training Metric section.

Based on the scheduling algorithm, the federated learning performance for fifty devices versus the number of rounds is given in the figure below.

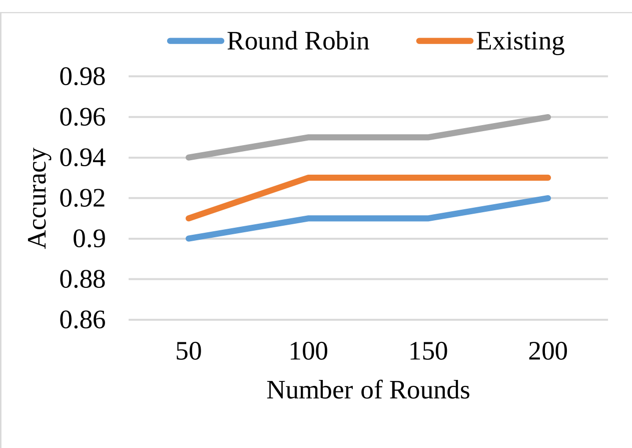

Fig. 9 Scheduling algorithm performance

Based on the research findings presented above, it was determined that the suggested federated learning-based scheduling algorithm and deep learning model are the most effective approaches for device communication in edge networks. In both the heterogeneous and the homogeneous models, the accuracy and scheduling capabilities of the suggested technique are superior to those of the other algorithms.

While several prior works explore machine learning in wireless communication, they typically apply uniform models or restrict usage to basic parameter updates. In contrast, our approach employs deep learning architectures such as Convolutional Neural Networks (CNNs) for feature extraction and downsampling, even in heterogeneous settings where different devices utilize varying models. This flexibility ensures robust local training while enabling more representative global knowledge aggregation.

A dual-loss function combining model error loss and knowledge-aided loss offers a critical innovation. Many previous federated learning models, including those utilizing hierarchical or layered architectures, focus primarily on model aggregation or privacy preservation.

However, they seldom emphasize the optimization of heterogeneous model compatibility. By jointly minimizing both types of losses, the system ensures not only accurate local performance but also improved global knowledge synthesis.

Dynamic scheduling mechanism is formulated as a non-convex optimization problem and solved using Lyapunov techniques, in contrast to static or heuristic-based device scheduling employed. This rigorous mathematical framework ensures real-time adaptability to dynamic network conditions and optimizes device selection across multiple rounds.

While in other algorithms, the model analysis was not taken into account for the global loss minimization. However, in this paper, the model analysis was performed using deep learning architectures, and its results showed that the knowledge mapping can be enhanced by enhancing the model.

5. Conclusion

In many network architectures, knowledge mapping on edge networks presents special difficulties. The federated learning strategy is employed as a means of avoiding it. The global loss and device selection are critical aspects of FL knowledge mapping. This study expands upon prior studies using a signal-to-noise ratio to determine which devices should participate in federated learning to boost FL performance. By giving devices access to various learning models, deep learning models enhance the feature aggregation procedure in FL. The enhanced feature contributes to minimizing the loss of both models and information. The proposed deep learning models also contributed to reducing global loss. The signal-to-noise ratio is provided alongside the other criteria for selecting devices. Signal-to-Noise Ratio (SNR) metrics enhance the signal quality that a device receives. Data transmission at the network's edge is improved using the suggested deep learning and federated learning paradigm.

Future works

In the future, the proposed federated learning performance can be enhanced using hybrid deep learning algorithms or the different non-convex optimization algorithms.

References

[1] Luis A. Fletscher et al., "Energy-Aware Resource Management in Heterogeneous Cellular Networks with Hybrid Energy Sources," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 279-293, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[2] Xiaoge Huang et al., "Dynamic Femtocell gNB On/Off Strategies and Seamless Dual Connectivity in 5G Heterogeneous Cellular Networks," *IEEE Access*, vol. 6, pp. 21359-21368, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[3] Nan Zhao et al., "Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Networks," *2018 IEEE Global Communications Conference*, Abu Dhabi, United Arab Emirates, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hideo Kobayashi et al., "Towards Sustainable Heterogeneous Wireless Networks: A Decision Strategy for AP Selection with Dynamic Graphs," *Computer Networks*, vol. 132, pp. 99-107, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[5] Jingjing Wang et al., "Distributed Q-Learning Aided Heterogeneous Network Association for Energy-Efficient IIoT," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2756-2764, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Rim Haw et al., "Cache Aware User Association for Wireless Heterogeneous Networks," *IEEE Access*, vol. 7, pp. 3472-3485, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] Yali Chen et al., "Resource Allocation for Device-to-Device Communications in Multi-Cell Multi-Band Heterogeneous Cellular Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4760-4773, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[8] Yaohua Sun et al., "Application of Machine Learning in Wireless Networks: Key Techniques and Open Issues," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3072-3108, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Chaoyun Zhang, Paul Patras, and Hamed Haddadi, "Deep Learning in Mobile and Wireless Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2224-2287, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mingzhe Chen et al., "Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3039-3071, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] Wei Yang Bryan Lim et al., "Hierarchical Incentive Mechanism Design for Federated Machine Learning in Mobile Networks," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9575-9588, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Haijun Zhang et al., "Deep Learning Based Radio Resource Management in NOMA Networks: User Association, Subchannel and Power Allocation," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2406-2415, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Yun Li et al., "Optimized Content Caching and User Association for Edge Computing in Densely Deployed Heterogeneous Networks," *IEEE Transactions on Mobile Computing*, vol. 21, no. 6, pp. 2130-2142, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Canh T. Dinh et al., "Federated Learning Over Wireless Networks: Convergence Analysis and Resource Allocation," *IEEE/ACM Transactions on Networking*, vol. 29, no. 1, pp. 398-409, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Chen Zhang et al., "A Survey on Federated Learning," *Knowledge-Based Systems*, vol. 216, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Wei Yang Bryan Lim et al., "Dynamic Edge Association and Resource Allocation in Self-Organizing Hierarchical Federated Learning Networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3640-3653, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[17] Xiaokang Zhou et al., "Two-Layer Federated Learning With Heterogeneous Model Aggregation for 6G Supported Internet of Vehicles," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 5308-5317, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Rami Hamdi et al., "Federated Learning Over Energy Harvesting Wireless Networks," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 92-103, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Pushpa Singh et al., *Federated Learning: Challenges, Methods, and Future Directions*, Federated Learning for IoT Applications, Springer, Cham, pp. 199-214, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Sepehr Ashtari et al., "Knowledge-Defined Networking: Applications, Challenges and Future Work," *Array*, vol. 14, pp. 1-48, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[21] Zhixiong Chen et al., "Knowledge-Aided Federated Learning for Energy-Limited Wireless Networks," *IEEE Transactions on Communications*, vol. 71, no. 6, pp. 3368-3386, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Jinglong Shen et al., "Effectively Heterogeneous Federated Learning: A Pairing and Split Learning Based Approach," *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, Kuala Lumpur, Malaysia, pp. 5847-5852, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23] Wenlong Song et al., "A Federated Learning Scheme Based on Lightweight Differential Privacy," *2023 IEEE International Conference on Big Data (BigData),* Sorrento, Italy, pp. 2356-2361, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Katharine Daly et al., "Federated Learning in Practice: Reflections and Projections," *2024 IEEE 6th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications*, Washington, DC, USA, pp. 148-156, 2024. [CrossRef] [Google Scholar] [Publisher Link]