Adaptive Federated Deep Learning-based Semantic Communication in the Social Internet of Things

Weixing Tan, Yusen Wang, Lei Liu*, Xiaoding Wang, Tong Ding

***Abstract*—The introduction of semantic communication offers an effective solution for achieving efficient and reliable information transmission in the Social Internet of Things (SIoT). SIoT combines social networks with the Internet of Things (IoT) to create a "social network of smart objects," utilizing analytical and statistical models to provide efficient and scalable services. However, ensuring high-quality and reliable data transmission within the SIoT remains a significant challenge. Semantic communication methods can effectively address this issue. Semantic communication represents an advanced paradigm aimed at achieving reliable transmission through semantic-level data compression. In this paper, we propose a semantic communication framework based on adaptive federated deep learning. This framework combines source-channel joint coding with channel bandwidth adaptation techniques to enhance transmission efficiency and promote natural and effective information exchange. Specifically, deep reinforcement learning is employed to manage dynamic bandwidth allocation, enabling the selection of optimal bandwidth under varying signal-to-noise ratios and data conditions, thereby improving transmission quality and bandwidth utilization. Additionally, we introduce a training method based on federated learning to enhance the model's generalization ability under different channel conditions. Simulation results demonstrate that our proposed method outperforms traditional models, exhibiting excellent adaptability to low signal-to-noise ratios and low bandwidth environments, as well as higher stability. This positions our method as a valuable approach for ensuring reliable data communication in the SIoT.**

Keywords: social internet of things, semantic communication, deep learning, deep reinforcement learning, federated learning

I. Introduction

The Social Internet of Things (SIoT) [1][12] combines social networks with the Internet of Things to form a "social network of smart objects" in which these objects can establish social relationships. By leveraging analytical and statistical models from social networks, SIoT ensures efficient and scalable network navigability to facilitate the discovery of services and objects. Further research is needed to achieve high-quality and high-stability social communication in the SIoT system. Semantic communication, an innovative form of intelligent communication, offers a new approach to achieving reliable communication through semantic-level data compression [11]. Therefore, it is promising to deploy a specific semantic communication model in SIoT to improve the efficiency, accuracy, and robustness of the system.

The transmission quality of semantic communication largely depends on the quality of the transmission channel and the available bandwidth. Under conditions of low SNR and limited bandwidth, the effectiveness of semantic communication is greatly restricted. Considering the low-latency and reliability requirements of many real-world applications, it is essential to explore more feasible methods to design communication systems that can adapt to various wireless settings. Addressing this need, this paper introduces an effective semantic communication system for SIoT [10] based on joint source-channel coding using deep learning. This system leverages the synergy of deep neural networks with channel-bandwidth adaptation technologies to enable natural, high-speed communication of various types of information, thereby significantly improving the user experience. By applying deep reinforcement learning for dynamic bandwidth allocation, we can determine the optimal bandwidth allocation strategy under different SNR conditions, thereby significantly improving transmission quality across various channel conditions.

Given the importance of data security and privacy protection [4] in information transmission, we utilize federated learning to protect privacy and enhance model generalization during training [28]. Federated Learning is a privacy-by-design distributed deep learning paradigm, where clients collaboratively train a model via the coordination of a server without sharing their private data. Training the semantic communication model using federated learning strengthens its generalization and privacy protection capabilities [31].

In the context of the Social Internet of Things (SIoT), federated learning enhances privacy by ensuring that data remains localized on individual devices, thus minimizing the risk of data breaches. It also employs techniques such as differential privacy and encrypted model updates to further protect sensitive information. Regarding generalization, federated learning leverages the diverse, heterogeneous data generated by various devices to improve the model's adaptability and performance across different scenarios. This dynamic and decentralized training approach allows the model to better generalize and respond to new data patterns and conditions, making it more robust and effective in SIoT environments.

The main contributions of this paper can be summarized as the followings:

- We integrate channel adaptation into semantic communication: By considering channel conditions during data encoding and decoding, the transmission model can effec-

This work was partially supported by Natural Science Foundation of Shandong (Shandong NSF no. ZR2021LZH006), National Natural Science Foundation of China (NSFC no. 62220106004), and Taishan Scholars Program. *(Corresponding author: Lei Liu.)*

Weixing Tan, Lei Liu and Tong Ding are with the School of Software, Shandong University, Jinan, 250101, China. Email: 202220799@mail.sdu.edu.cn; l.liu@sdu.edu.cn; tongding@mail.sdu.edu.cn. Yusen Wang and Xiaoding Wang are with the College of Computer and Cyber Security, Fujian Normal University, Fuzhou, 350117, China. Email: wangyusen1104@gmail.com; wangdin1982@fjnu.edu.cn

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3484230

tively adapt to low SNR, thereby enhancing transmission performance.
- We apply deep reinforcement learning to the bandwidth allocation module: By integrating bandwidth adaptation into semantic communication, the model can adapt not only to low SNR ratios but also to environments with limited bandwidth, significantly improving transmission performance under challenging communication conditions.
- We train the model in a federated learning environment: Utilizing the distributed training mode of federated learning enhances the model's generalization ability and ensures data privacy protection, thereby improving its applicability and security.
- We evaluate the developed adaptive federated deep learning model for its capability to enhance the reliability and stability of data transmission. Simulation experiments demonstrate that the model achieves effective communication under conditions of low bandwidth and low SNR, thereby enhancing user experience and providing reliable services.

The rest of the paper is organized as follows. Section II reviews the related work. Section III presents the theoretical basis and system model. Sections IV introduces the implementation details of the proposed framework. Section V shows the simulation results and discussions. Section VI concludes the paper.

II. Related work

Semantic communication is a research endeavor aimed at achieving intelligent interaction between humans and machines. Its goal is to enable computers to understand and respond to human language expressions, thereby facilitating more natural and efficient communication. The following provides an overview of recent developments in various aspects of semantic communication.

A. Semantic Communication and Natural Language Processing

In the field of semantic communication research, natural language processing (NLP) and machine learning technologies play a crucial role. Researchers are dedicated to developing semantic understanding and generation models to parse and comprehend the meaning of human language. These models can extract meaningful information from sentences or texts through techniques such as semantic analysis, intent recognition, and context inference, and respond accordingly. For example, Eliyahu Kiperwasser and Yoav Goldberg [15] proposed a simple yet effective dependency parsing scheme based on bidirectional LSTMs (BiLSTM). Each token in a sentence is associated with a BiLSTM vector that represents the token's context within the sentence. Feature vectors are constructed by concatenating several BiLSTM vectors. The resulting parser has a very straightforward architecture and can match or exceed state-of-the-art accuracy for both English and Chinese.

Similarly, Farsad et al. [8] proposed a deep learning-based encoder and decoder for scenarios involving limited-length documents and constrained encoding lengths, aiming to achieve lower word error rates. Unlike traditional separate source and channel coding methods, this approach first embeds sentences into a semantic space and then performs joint source and channel coding on these embeddings, thereby preserving the semantic information of the sentences.

B. Adaptive semantic communication

Adaptive semantic communication is a method designed to address the challenges of dynamic environments and changing data distributions. Traditional communication systems typically assume static channels and fixed data distributions, whereas real-world communication environments and data distributions are often dynamic. Adaptive semantic communication aims to tackle this challenge by dynamically adjusting the parameters and models of the communication system to adapt to environmental changes, ensuring communication reliability and performance. Guangyi Zhang [32] proposed a deep learning-based semantic communication system (DeepSC-MIMO), which leverages channel state information and noise variance for model design. The system includes a performance evaluator to predict the reconstruction quality of each image, enabling intelligent image allocation. This significantly improves the reliability of image transmission while greatly reducing feedback overhead.

Jincheng Dai [6] introduced a novel online learning method for joint source and channel coding, utilizing the overfitting characteristics of deep learning models. This method updates pre-trained models in a lightweight online manner post-deployment to adapt to changes in the distribution of source data and environmental domains. By pushing the concept of overfitting to the extreme, a series of implementation-friendly methods were proposed, adapting the encoder-decoder models or representations to individual data or channel state instances. This approach achieves significant gains in end-to-end rate-distortion performance. The system design is formulated as a joint optimization problem aiming to minimize a loss function that balances data stream bandwidth cost, model stream bandwidth cost, and end-to-end distortion. The proposed method enables all parameters in the network to achieve communication-efficient adaptability without compromising decoding speed. Extensive experiments on dynamically changing target source data and wireless channels, including user studies, validate the effectiveness of this approach.

C. Application of Federated Learning in Model Training

Federated learning is a distributed machine learning approach that allows models to be trained without centralizing data. Participants retain their local data and train models locally, then aggregate the model parameters on a central server to form a global model. This method protects data privacy while allowing the use of global data statistics for model updates. Federated learning has many advantages and is particularly suited for privacy-sensitive domains such as healthcare and financial services [16][25]. It also enhances model generalization and performance, addresses data fragmentation and ownership issues, and facilitates model sharing among multiple organizations.

Jianrui Chen [5] proposed a trustworthy semantic communication system for the metaverse based on federated learning, designed to handle large volumes of multimodal data in immersive environments while safeguarding data security and privacy. By leveraging distributed decision-making and privacy-preserving capabilities, this system reliably manages confidential data exchanges.

III. Theoretical basis and system model

A. Semantic communication theoretical basis

The following uses text semantic communication as an example to introduce the process of semantic communication. The transmitter maps a sentence s into a complex stream of symbols x, which is then passed through a physical channel with transmission impairments such as distortion and noise [26]. The receiver decodes the received y to estimate the original sentence s. The sender consists of a semantic encoder and a channel encoder, which are used to extract semantic information and ensure the successful transmission of semantic information on the physical channel. The encoded symbol stream can be expressed as:

$$x = C_\alpha(S_\beta(s)). \tag{1}$$

Here, $S_\beta(\cdot)$ is a semantic encoder network with parameter set β, and $C_\alpha(\cdot)$ is a channel encoder network with parameter set α.

After being encoded by the encoder, the encoded symbol x is transmitted through a physically noisy channel with transmission impairments, such as distortion and noise. We will mainly consider using AWGN channels [21][23]. The chosen channel must allow backpropagation. Physical channels can be represented by neural networks. For example, simple neural networks can be used to simulate AWGN channels, multiplicative Gaussian noise channels [9] and erasure channels [2]. The transmission process of data in the channel can be expressed as:

$$y = \eta(x) = x + \omega. \tag{2}$$

Here, the vector $\omega \in C_k$ represents the channel interference coefficient.

We also consider to conduct experiments with this system on other channels, in which the transmission process of the channel can be represented as:

$$y = hx + \omega. \tag{3}$$

Here, $h \in C$ is the channel gain.

On the other hand, the receiver includes a channel decoder and a semantic decoder to recover the transmitted symbols and transmitted sentences respectively. The decoded signal can be expressed as:

$$\hat{s} = S_\nu^{-1}(C_\mu^{-1}(y)). \tag{4}$$

Here, $\hat{s}$ is the recovered sentence, $C_\mu^{-1}(\cdot)$ is the channel decoder with parameter set μ, and $S_\nu^{-1}(\cdot)$ is the semantic decoder network with parameter set ν.

Note that the goal of this system is to minimize semantic errors while reducing the number of symbols to be transmitted. Using cross entropy (CE) [7] as the loss function to measure the difference between s and $\hat{s}$. For example, consider a scenario where both the sender and receiver understand the concept of a "sunny day." In traditional communication, the sender would transmit each letter of the phrase "sunny day," and the receiver would reconstruct and decode these letters to retrieve the original message. In contrast, semantic communication allows the sender to convey the core meaning of "sunny day" directly. The receiver can then recover the information by understanding this meaning, which reduces the amount of data transmitted and enhances communication efficiency.

B. System model

As shown in Figure 1, it is the basic framework of the entire adaptive semantic communication model. The entire framework is divided into two parts: encoder and decoder. The encoder and decoder have their corresponding neural networks to encode and decode text data. The main contribution of this paper is to add channel adaptation and bandwidth adaptation on top of the basic semantic communication framework. The green part at the bottom of the figure is the channel adaptation part, which extracts the signal-to-noise ratio information of the channel and fuses the features with the encoder and decoder to adapt to different channel qualities. The blue part above the encoder is the Deep Reinforcement Learning bandwidth decision module, which considers the transmission data and the channel SNR to decide the output bandwidth of the encoder. In this way, data can be transmitted with bandwidth adaptation in different channels.

IV. The Implementation Details of the Proposed Framework

A. Channel Adaptation

We propose a channel-adaptive method based on the article [29]. This method can operate under different SNR levels during transmission, dynamically adjusting the compression ratio of source coding and the encoding rate of channel coding according to the SNR. This is achieved through an attention mechanism.

We use the Attention Fusion (AF) module to process the SNR feedback from the channel, and then embed it into the encoder by alternating with the Feature Learning (FL) module, as shown in Figure 2. In general, the AF module processes the features extracted by the FL module using global average pooling. The pooled features are then combined with the SNR to form contextual information. This contextual information is fed into a fully connected neural network to produce scaling factors. The scaled features are derived by applying these scaling factors, thereby adapting the features according to the SNR conditions.

We then explain the channel adaption in details. Recall that our encoder and decoder consist of the FL and AF modules

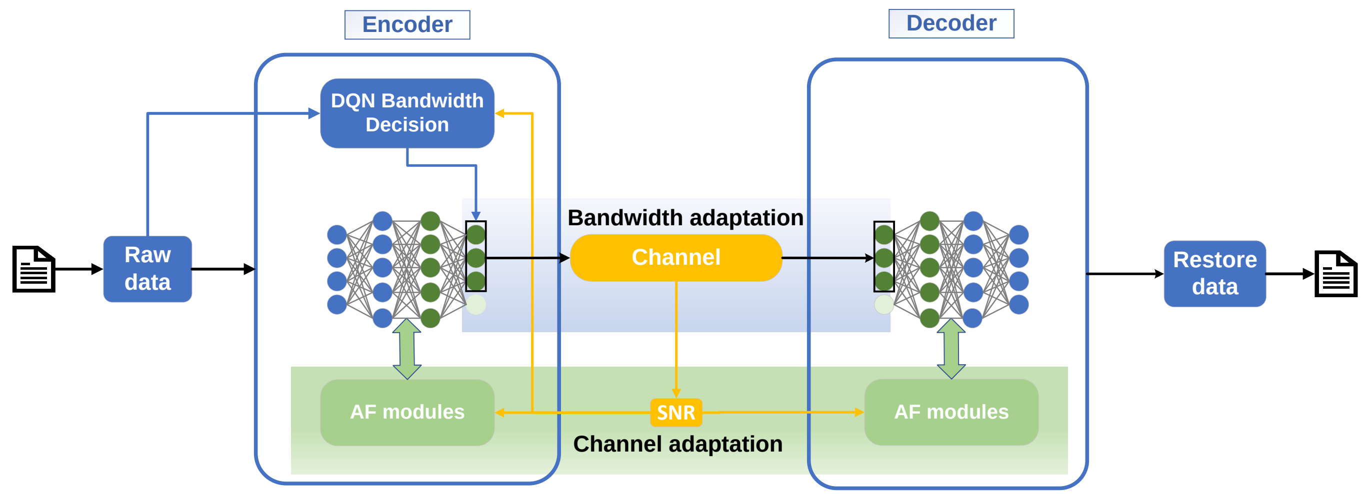


Fig. 1: Deep learning-based adaptive semantic communication system model

respectively. In the encoder-decoder system, the encoder integrates both the FL and AF modules to encode the input data into a format suitable for transmission. The FL module extracts features from the data and combines them with SNR feedback to create context information. The AF module then applies weighting to these features to enhance data representation. The decoder performs the reverse process, decoding the encoded data to recover the original information, incorporating the reverse FL and AF modules to ensure effective data recovery.

The FL module first extracts information from the data and fuses it with the SNR to form contextual information, which is then provided to the next AF module. Then, given the contextual information as input, an attention mask for the FL feature is generated in the factor prediction network, and the FL feature is scaled according to the attention weights. The output of each AF module is then fed back to the next FL module, and this process is repeated four times to obtain the final scaled data.

Training this structure under a certain range of SNR enables it to adaptively encode data under different channel environments, improving the quality of data transmission, especially under poor channel quality and low SNR. In the following experiments, we compare the channel-adaptive model with three models trained under fixed SNR to demonstrate its superior performance, especially under poor channel quality. The pseudo code of channel adaptation is shown in Algorithm 1.

B. Bandwidth Adaptation

In traditional semantic communication, the encoder encodes all data and transmits it in its entirety. This traditional approach, known as full bandwidth transmission, aims to faithfully reproduce the original data. However, in scenarios with poor channel quality, this method transmits all data, even when not all of it is essential. Consequently, it fails to effectively utilize the limited bandwidth resources to transmit the most critical data. To address this, we propose the use of deep reinforcement learning [30] to create a bandwidth decision

---

Algorithm 1 Channel adaptive algorithm

Input: Data x, Signal-to-noise ratio SNR
Output: Eigenvalues A_f

1: The FL module extracts features ω_L from data x and passes ω_L to the AF module.
2: The AF module performs average pooling on it to obtain I. $I = A(\omega_L)$
3: Connect I with SNR. $I_s = (I, SNR)$
4: After a simple neural network. $I_N = N_\epsilon(I_s)$
5: Finally, it is checked with ω_L to obtain the attention feature A_f. $A_f = I_N \cdot \omega_L$
6: **return** A_f

---

module. This module, by learning the current channel's SNR and optimizing data transmission within the constraints of limited bandwidth resources, adapts bandwidth allocation to the given environment. The pseudo code of channel adaptation is shown in Algorithm 2.

We treat the allocation of available bandwidth for each data set as a decision process and utilize deep reinforcement learning to find the optimal bandwidth allocation strategy. The decision module is defined by a tuple (S, A, r), where S is the set of states, A is the set of actions, and $r: S \times A \to R$ is the reward function. At each time step n, we observe the state $s_n \in S$ and choose an action $a_n \in A$ based on its policy $\pi: S \to A$. Subsequently, the state transitions to s_{n+1} based on feedback regarding data transmission quality, and a reward rn is received. The objective of this reinforcement learning network is to maximize the expected cumulative reward. In the context of dynamic bandwidth allocation problems, we define the state at time step n as:

$$S_n = \{Data_n, SNR_n\}. \tag{5}$$

Here, $Data_n$ represents text data, and SNR_n represents the current signal-to-noise ratio.

Meanwhile, we define action set A as follows, where k

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3484230

represents the upper limit of allocable bandwidth.

$$A_n = \{1, 2, 3 \ldots k\}. \tag{6}$$

After data transmission, we get the feedback transmission quality and MSE. The higher the transmission quality, the smaller the MSE. The average MSE for N data is defined as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N} d(x^{(i)} - x^{(i)}). \tag{7}$$

The smaller MSE, the greater the reward. Thereby, we define reward as follows, where x is a value greater than and close to MSE, which is used to convert MSE into a positive number

$$r = x - MSE. \tag{8}$$

To learn the optimal allocation strategy, we employ the deep Q-learning [20] , where the network q_ψ seeks to approximate the Q-function $Q: \;\; S \times A \rightarrow R$. The purpose of the Q-function is to map each state-action pair to a Q-value, where the Q-value represents the total discounted reward for a given state-action pair (s_n, a_n) at time step n, as shown below:

$$Q(\mathbf{s}^n, \mathbf{a}^n) = E\left[\sum_{i=n}^{\infty} \gamma^{i-n} r^i \middle| \mathbf{s}^n, \mathbf{a}^n\right], \forall(\mathbf{s}^n, \mathbf{a}^n) \in \mathcal{S} \times \mathcal{A}. \tag{9}$$

In this paper, we employ a typical DQN method, using a replay buffer, a target network, and an ε-greedy policy to enhance the learning of the Q-function. The replay buffer stores experiences (s_n, a_n, r_n, s_{n+1}) and uniformly samples them to update the parameters ψ. We use the following loss function to compute the loss of the DQN.

$$L_{\text{DQN}}(\boldsymbol{\psi}) = \left(r^n + \gamma \max_{\mathbf{a}} \left\{q_{\boldsymbol{\psi}^-}\left(\mathbf{s}^{n+1}, \mathbf{a}\right)\right\} - q_{\boldsymbol{\psi}}\left(\mathbf{s}^n, \mathbf{a}^n\right)\right)^2. \tag{10}$$

Although using DQN for bandwidth allocation is innovative, but its computational complexity and practical feasibility in the social Internet of Things (SIoT) must be carefully evaluated. The computational complexity of DRL primarily arises from the size of the state space S and the action space A. The state space must represent various channel conditions and data characteristics, while the action space encompasses different bandwidth allocation strategies. This can result in high-dimensional state and action spaces, increasing computational complexity. Additionally, training DQN models typically requires substantial computing resources and time, as the model needs to optimize its strategy through numerous interactions, which may pose a bottleneck for real-time applications.In the context of real-time SIoT applications, while DQN offers significant advantages, it also presents challenges. DQN can dynamically adjust bandwidth allocation strategies based on real-time channel conditions, greatly enhancing data transmission efficiency. This dynamic adaptability is vital for optimizing bandwidth utilization. However, the high computational resource demands of DQN may strain real-time applications. Therefore, balancing optimization algorithms with computing resources is essential to ensure the system operates efficiently in real-time environments.

Algorithm 2 Bandwidth Adaptive Algorithm

Initialization:
Initialize the memory for revisiting, denoted as D, with a capacity to store N data records
Initialize the action-value function Q using the weight $\theta = \gamma, r$;
Set $\theta^- = \theta$ to initialize the target action-behavior network Q^-;
The first state $S_1 = \{Text_1, Image_1, Speech_1, SNR_1\}$ of the event is initialized, and the characteristic input $\phi_1 = \phi(s_1)$ corresponding to the state is obtained through preprocessing
for $t = 1, T$ **do**
    Select random action with probability ε, i.e., $A_t = a_1^t$
    In the absence of the occurrence of the low-probability event, the strategy resorts to the greedy approach, choosing the set of actions with the highest current value function.
    $a^t = argmax_a Q(\phi(s_t), \theta; a)$
    The selected action group A_t is used for the output bandwidth of the semantic communication encoder
    Get the output data loss value: MSE_t
    Then r_t is awarded by MSE_t
    A new state S_{t+1} is obtained, and $\phi_{t+1} = \phi(S_{t+1})$ is obtained after preprocessing
    Store $(\phi_t, A_t, r_t, \phi_{t+1})$ in playback memory D
    Sample a transformation sample data uniformly at random from playback memory D: $(\phi_j, A_j, r_j, \phi_{j+1})$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ
    Every C steps reset $Q^- = Q$
end for

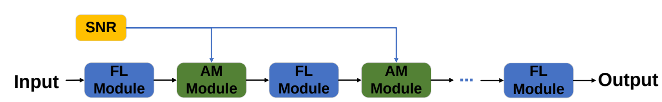


Fig. 2: Attention Fusion module flow chart

C. Training in a Federated Learning Environment

Federated learning [17] is an emerging distributed machine learning method that allows multiple participants to collaboratively train a model without the need to centrally collect and store data. By keeping data on local devices and only sharing model parameters or gradients, federated learning protects user privacy. Since the data and model remain on local servers and do not need to be transmitted to a central server, the risk of data breaches is significantly reduced. This method effectively prevents data from being intercepted or attacked during transmission. Additionally, traditional centralized training methods require uploading large amounts of data to a central server, whereas federated learning only needs to transmit updated model parameters or gradients, significantly reducing bandwidth consumption. More importantly, by integrating data from different sources, federated learning can generate models with stronger generalization capabilities. This training

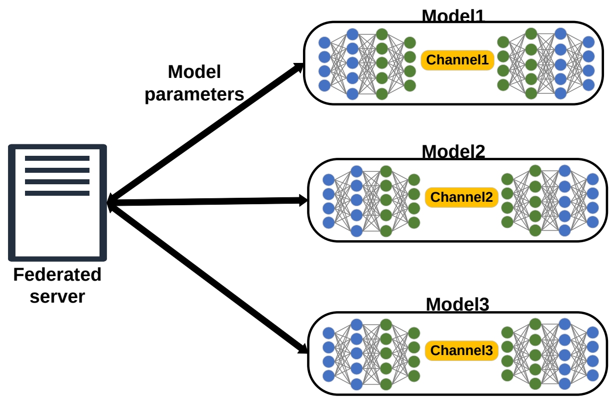


Fig. 3: Federated learning-based training of semantic communication model

approach helps models better adapt to different scenarios and applications, enhancing their overall generalization ability [3].

Applying federated learning to the training of semantic communication models can enhance data privacy [13] and improve the model's generalization ability across different environments. As shown in Figure 3, this is the training mode of a semantic communication model in a federated learning environment. This mode provides models for three different transmission environments, and these models undergo text semantic transmission training for 100 epochs. After each training epoch, the parameters of the three models are saved and uploaded to the federated server, where they are averaged. The averaged parameters are then returned to each model. The three models receive the same averaged parameters, load them into their respective neural networks, and continue with semantic training, completing one federated training cycle. A total of 100 federated training cycles are conducted. The final averaged parameters represent the training result.

In comparison, our centralized training method involves training three models across the same three scenarios and focusing on text semantic transmission for 10,000 epochs. This is contrasted with the federated learning training mode. In the subsequent verification experiments, we will evaluate and compare the performance of models trained using these two different approaches.

V. Numerical Results

A. Experiment Setup

Our experiments were conducted on a laboratory computer with the following hardware configuration: CPU: Intel® Core™ i5-8300H @ 2.30 GHz; RAM: 16.0 GB; Operating System: 64-bit, x64-based processor. The software configuration was as follows: the compiler used was Visual Studio Code (VSCode), and the deep learning framework employed was PyTorch.

After training in the federated learning environment, experiments were set up on the model to examine the impact of adaptive strategies and federated learning training on model performance. We trained two sets of models for this purpose, one was trained using an adaptive strategy; the other was trained with a fixed bandwidth. We used fixed bandwidth to train the full bandwidth model; 3/4 bandwidth model; 1/2 bandwidth model and 1/4 bandwidth model. They are trained on the AWGN channel with a SNR [14] ranging from 0 to 20.

The adaptive part of the bandwidth decision is based on deep reinforcement learning. Set the learning rate: 0.001; the discount factor for future rewards (default is 0.99); the parameters of the target network soft update (default is 0.005); the exploration rate of the ϵ-greedy strategy (default is 1.0); the maximum size of the replay buffer (default 1000000); batch size for experience replay (default 256). Pytorch [22] was used to train these models, using the Adam optimizer [34] with a learning rate of 0.0001.

In this paper, we utilized the European Parliament [27] Proceedings dataset, which contains approximately 2 million sentences and 53 million words. The preprocessing procedure involved several key steps: First, the text was tokenized into individual words to facilitate further analysis. Next, sentences were standardized to fall within a length range of 4 to 30 words, ensuring that only sentences of appropriate length were retained. Sentences shorter than 4 words or longer than 30 words were discarded to maintain the quality and consistency of the training data. Finally, the preprocessed dataset was split into training and test sets. The training set was used for model training, while the test set was reserved for evaluating model performance.

The trained model was tested for bandwidth. According to the test signal-to-noise ratio, it was divided into four groups: SNR=0; SNR=6; SNR=12; SNR=18. Each group of experiments adjusted the bandwidth ratio during transmission as a variable to observe their transmission quality. , the experimental results are shown in Figure 4. In addition, the trained model was tested for signal-to-noise ratio. According to the test bandwidth ratio, it was divided into four groups: bandwidth=0.25; bandwidth=0.5; bandwidth=0.75; bandwidth=1. Each group of experiments adjusted the signal-to-noise ratio during transmission as a variable to observe them. The transmission quality, and the experimental results are shown in Figure 5. The transmission quality of the text is represented by the Bilingual Evaluation Understudy (BLEU) score [24]. For a sent sentence of length l and a decoded sentence of length l, BLEU can be expressed as:

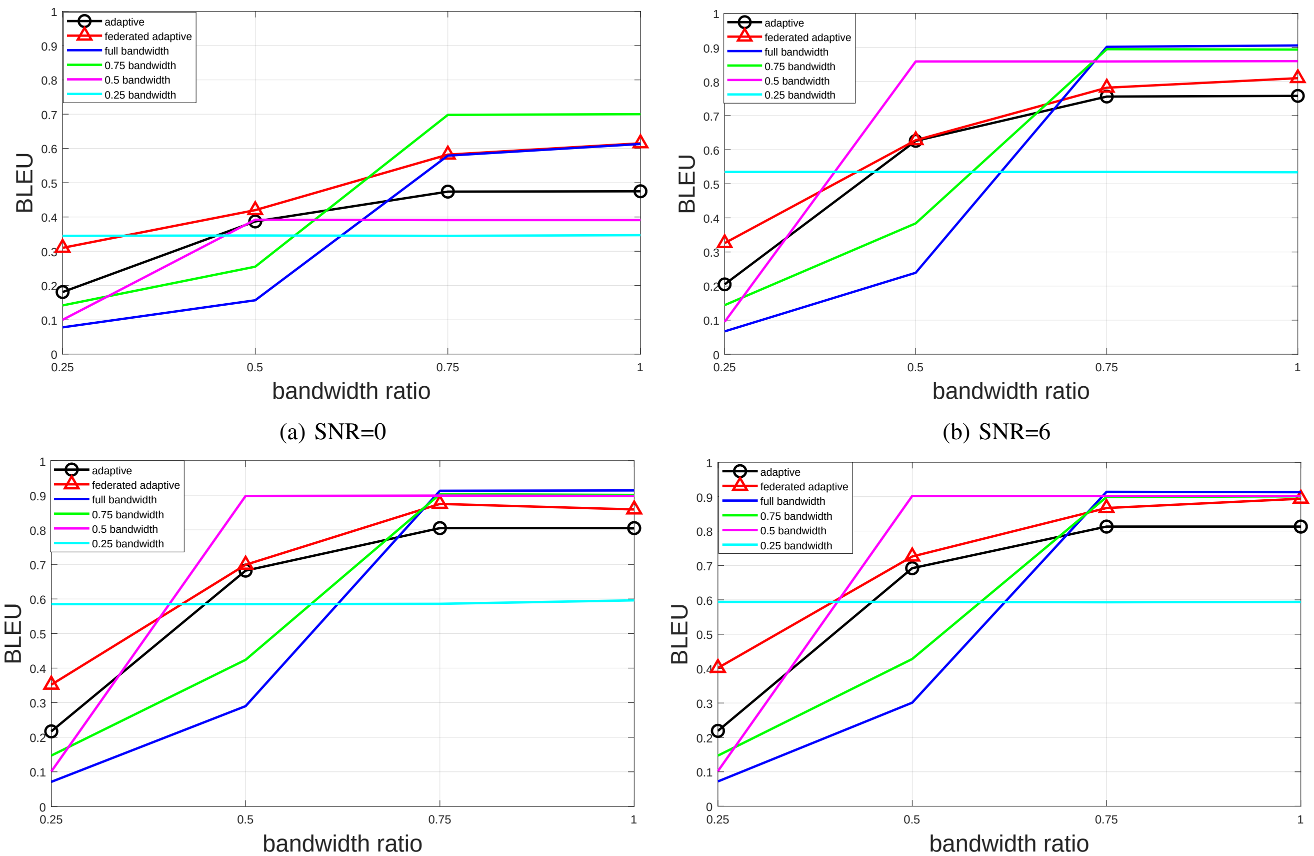
$$\log \mathrm{BLEU} = \min\left(1 - \frac{l\hat{\mathbf{s}}}{l_{\mathbf{s}}}, 0\right) + \sum_{n=1}^{N} u_n \log p_n. \qquad (11)$$

Here, BLEU is a number between 0 and 1 that indicates how similar the decoded text is to the transmitted text, with 1 being the most similar.

B. Experiment Results

As shown in Figure 4, when the bandwidth ratio is below 0.5, the performance of the adaptive bandwidth model surpasses that of the full bandwidth model, the 1/4 bandwidth model, and the 3/4 bandwidth model, but is inferior to the 1/2 bandwidth model. However, at lower bandwidth ratios,

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3484230



(a) SNR=0 (b) SNR=6 (c) SNR=12 (d) SNR=18

Fig. 4: Comparison experiments between bandwidth-adaptive model and fixed bandwidth model under different signal-to-noise ratios

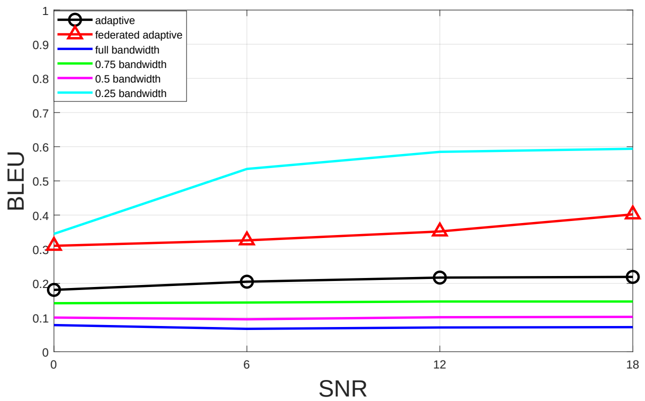
the adaptive model outperforms most of the fixed bandwidth models. This is because fixed bandwidth models trained in high-bandwidth environments struggle to adapt to sudden bandwidth reductions, resulting in a cliff effect. In contrast, our adaptive bandwidth model, which is trained under various bandwidth conditions, exhibits better generalization and bandwidth adaptability compared to fixed bandwidth models. Nonetheless, this averaging adaptive mechanism comes at the cost of performance in high-bandwidth scenarios.

As shown in Figure 5, four experiments were conducted with six models across four different bandwidth ratio conditions. From a broad perspective, the transmission performance of all models improved as the SNR increased. A detailed analysis revealed that the performance differences between models became more pronounced as the bandwidth ratio decreased. Specifically, the adaptive model and the federated adaptive model outperformed most fixed bandwidth models, with the federated adaptive model showing superior performance compared to the standard adaptive model. It is evident that model performance improves with better channel quality. Additionally, the stability and generalization ability of both adaptive models were significantly enhanced due to the adaptive averaging effect, and the inclusion of federated learning further enhanced the adaptability of the model to varying conditions.

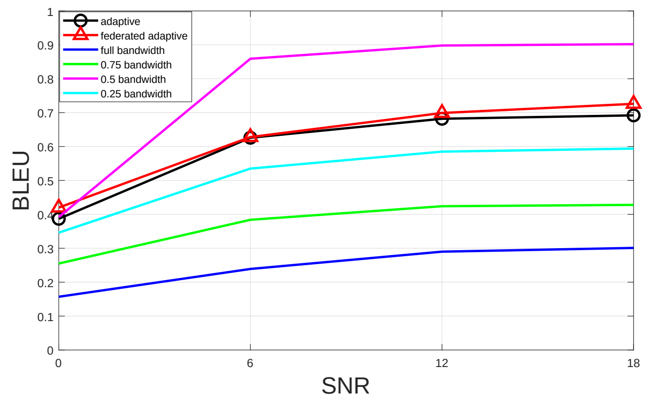
Result Analysis. From these two figures, it is clear that the performance of different models under varying bandwidth ratios and SNRs shows that the 1/4 bandwidth model remains largely unchanged. In contrast, the full bandwidth model, 3/4 bandwidth model, and 1/2 bandwidth model exhibit a significant performance decline as the bandwidth ratio and SNR decrease. This means that when the test bandwidth ratio is lower than the training bandwidth ratio, fixed bandwidth models fail to adapt to low-bandwidth environments, resulting in a "cliff-edge" effect.

Table I is a performance comparison between the federated adaptive model and the standard adaptive model based on the full-bandwidth basic model. By observing the performance of the federated adaptive model and the standard adaptive model in Table I, it can be seen that the federated adaptive model trained by the federated learning method shows more stable performance and stronger generalization ability under different SNR and bandwidth ratio conditions. From Table II, it can be seen that the adaptive model with federated learning has improved performance in various situations, especially in low bandwidth.

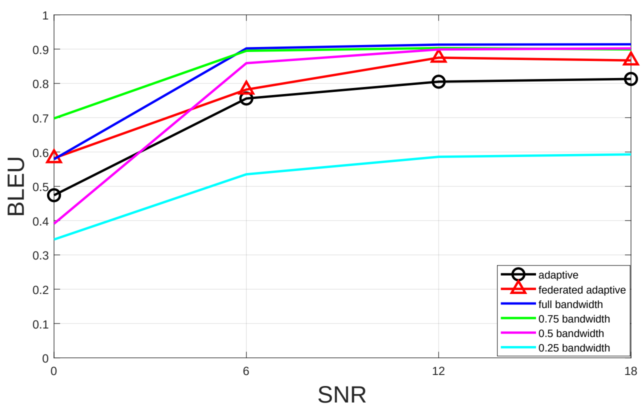
Result Analysis. It can be concluded that the model trained by federated learning can better adapt to low bandwidth and low signal-to-noise ratio environments.



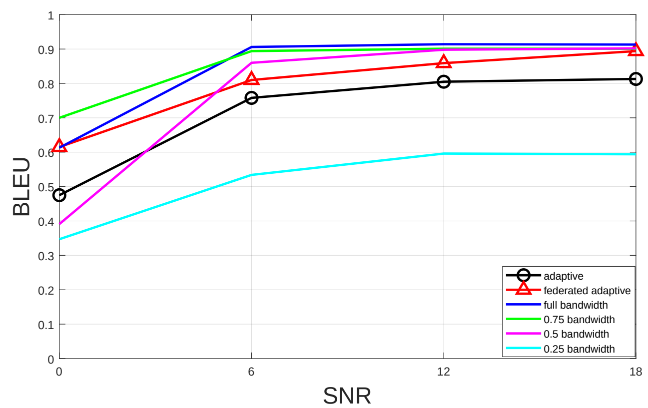
(a) bandwidth=0.25



(b) bandwidth=0.5



(c) bandwidth=0.75



(d) bandwidth=1

Fig. 5: Comparison experiment between bandwidth-adaptive model and fixed bandwidth model under different bandwidths

TABLE I: Performance comparison between adaptive model and full-bandwidth basic model

	Adaptive Systems (%)				Federated Adaptive Systems (%)			
Signal-to-Noise Ratio (dB)	0	6	12	18	0	6	12	18
Bandwidth share 25%	132.0	**206.0**	205.6	200.0	297.4	386.5	395.8	**458.3**
Bandwidth share 50%	143.0	162.0	135.0	129.1	167.5	163.0	141.0	141.2
Bandwidth share 75%	-18.1	-15.6	-10.5	-9.9	0.5	-13.3	-4.2	-5.1
Bandwidth share 100%	-22.3	-15.2	-10.5	-10.0	0.3	-11.9	-6.0	-2.1

TABLE II: Performance comparison between federated adaptive model and basic adaptive model

	Federated Adaptive Systems (%)			
Signal-to-Noise Ratio (dB)	0	6	12	18
Bandwidth share 25%	71.3	59.0	62.2	**83.6**
Bandwidth share 50%	8.5	0.3	2.5	4.9
Bandwidth share 75%	22.8	3.4	8.7	6.6
Bandwidth share 100%	29.5	6.7	6.7	10.0

VI. Conclusion

The Social Internet of Things (SIoT) [33] enhances the intelligence of IoT systems by leveraging social relationships among devices, providing users with a high-quality service experience and promoting efficient resource utilization [18]. In SIoT environments, reliable and efficient communication between devices is essential. Thus, this paper introduces a deep learning-based semantic communication system with joint source-channel coding (DeepJSCC), channel adaptation and bandwidth adaptation. Channel adaptation is achieved through feature fusion technology, which enables the model to adapt to various channel conditions. Using deep reinforcement learning, bandwidth is dynamically allocated under different signal-to-noise ratio conditions. In addition, by utilizing federated learning to train the model in different environments, we enhance the generalization ability of the model while preserving data privacy [19].

Compared with the similar adaptive semantic communication model DeepSC-MIMO [32], it improves communication stability and data security, which are extremely important in social networks. Our approach improves transmission efficiency and enables information to be conveyed more naturally and quickly. Applying it to the SIoT can ensure reliable and high-quality transmission of data in diverse and complex environments, further improving user experience and providing personalized services.

Despite the significant progress made, several limitations and potential directions for future research remain. First, while the model performs well in controlled laboratory settings, ensuring its scalability in large-scale SIoT deployments is still a challenge. Second, practical deployment encounters issues such as computational resource constraints, device diversity, and integration with existing infrastructure. Finally, although federated learning enhances privacy protection, it also increases coordination and communication overhead. Future research should focus on optimizing both federated learning and deep learning processes to reduce communication costs and improve training efficiency, especially in scenarios involving a large number of devices. In addition, regarding the practical application of the system, we believe that based on its excellent performance under harsh conditions, it can be used for long-distance communications and communications in some environments with poor network quality, such as rural network construction or cross-border network construction.

References

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. Siot: Giving a social structure to the internet of things. *IEEE Communications Letters*, 15(11):1193–1195, 2011.

[2] Charles H Bennett, David P DiVincenzo, and John A Smolin. Capacities of quantum erasure channels. *Physical Review Letters*, 78(16):3217, 1997.

[3] Nicolo Cesabianchi, Alex Conconi, and Claudio Gentile. On the generalization ability of on-line learning algorithms. *IEEE Transactions on Information Theory*, 50(9):2050–2057, 2004.

[4] Deyan Chen and Hong Zhao. Data security and privacy protection issues in cloud computing. In *2012 international conference on computer science and electronics engineering*, volume 1, pages 647–651. IEEE, 2012.

[5] Jianrui Chen, Jingjing Wang, Chunxiao Jiang, Yong Ren, and Lajos Hanzo. Trustworthy semantic communications for the metaverse relying on federated learning. *IEEE Wireless Communications*, 30(4):18–25, 2023.

[6] Jincheng Dai, Sixian Wang, Ke Yang, Kailin Tan, Xiaoqi Qin, Zhongwei Si, Kai Niu, and Ping Zhang. Toward adaptive semantic communications: Efficient data transmission via online learned nonlinear transform source-channel coding. *IEEE Journal on Selected Areas in Communications*, 41(8):2609–2627, 2023.

[7] Pieter Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134:19–67, 2005.

[8] Nariman Farsad, Milind Rao, and Andrea Goldsmith. Deep learning for joint source-channel coding of text. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2326–2330. IEEE, 2018.

[9] G David Forney and Gottfried Ungerboeck. Modulation and coding for linear gaussian channels. *IEEE Transactions on Information Theory*, 44(6):2384–2415, 1998.

[10] Murugesan Gurusamy, Maheswara Venkatesh Panchavarnam, and Jayasankar Thangaiyan. Enhancing trust management using locally weighted salp swarm algorithm with deep learning for siot networks. *Brazilian Archives of Biology and Technology*, 67:e24240207, 2024.

[11] HV Jagadish, Jason Madar, and Raymond T Ng. Semantic compression and pattern extraction with fascicles. In *VLDB*, volume 99, pages 186–97, 1999.

[12] Anubha Jain, Priyanka Verma, Usha Badhera, and Pooja Nahar. Design and development of secure mobile social network with iot. In *The Next Generation Innovation in IoT and Cloud Computing with Applications*, pages 74–89. CRC Press, 2025.

[13] Priyank Jain, Manasi Gyanchandani, and Nilay Khare. Big data privacy: a technological perspective and review. *Journal of Big Data*, 3:1–25, 2016.

[14] Don H Johnson. Signal-to-noise ratio. *Scholarpedia*, 1(12):2088, 2006.

[15] Eliyahu Kiperwasser and Yoav Goldberg. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327, 2016.

[16] Yogesh Kumar and Ruchi Singla. Federated learning systems for healthcare: perspective and recent progress. *Federated Learning Systems: Towards Next-Generation AI*, pages 141–156, 2021.

[17] Li Li, Yuxi Fan, Mike Tse, and Kuoxi Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 2020.

[18] Jed Mills, Jia Hu, and Geyong Min. Communication-efficient federated learning for wireless edge intelligence in iot. *IEEE Internet of Things Journal*, 7(7):5986–5994, 2019.

[19] Jed Mills, Jia Hu, and Geyong Min. Multi-task federated learning for personalised deep neural networks in edge computing. *IEEE Transactions on Parallel and Distributed Systems*, 33(3):630–641, 2021.

[20] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[21] Sangwoo Park, Osvaldo Simeone, and Joonhyuk Kang. End-to-end fast training of communication links without a channel model via online meta-learning. In *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2020.

[22] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[23] David R Pauluzzi and Norman C Beaulieu. A comparison of snr estimation techniques for the awgn channel. *IEEE Transactions on communications*, 48(10):1681–1691, 2000.

[24] Ehud Reiter. A structured review of the validity of bleu. *Computational Linguistics*, 44(3):393–401, 2018.

[25] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

[26] Yulin Shao, Qi Cao, and Deniz Gz. A theory of semantic communication. *IEEE Transactions on Mobile Computing*, 2024.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3484230

[27] Astrid Van Aggelen, Laura Hollink, Max Kemman, Martijn Kleppe, and Henri Beunders. The debates of the european parliament as linked open data. *Semantic Web*, 8(2):271–281, 2017.

[28] Jin Wang, Jia Hu, Jed Mills, Geyong Min, Ming Xia, and Nektarios Georgalas. Federated ensemble model-based reinforcement learning in edge computing. *IEEE Transactions on Parallel and Distributed Systems*, 2023.

[29] Jialong Xu, Bo Ai, Wei Chen, Ang Yang, Peng Sun, and Miguel Rodrigues. Wireless image transmission using deep source channel coding with attention modules. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(4):2315–2328, 2021.

[30] Yingjun Ye, Xiaohui Zhang, and Jian Sun. Automated vehicle's behavior decision making using deep reinforcement learning and high-fidelity simulation environment. *Transportation Research Part C: Emerging Technologies*, 107:155–170, 2019.

[31] Zhengxin Yu, Jia Hu, Geyong Min, Zhiwei Zhao, Wang Miao, and M Shamim Hossain. Mobility-aware proactive edge caching for connected vehicles using federated learning. *IEEE Transactions on Intelligent Transportation Systems*, 22(8):5341–5351, 2020.

[32] Guangyi Zhang, Qiyu Hu, Yunlong Cai, and Guanding Yu. Scan: Semantic communication with adaptive channel feedback. *IEEE Transactions on Cognitive Communications and Networking*, pages 1–1, 2024.

[33] Yunfan Zhang, Feihuang Chu, Luliang Jia, Miao Yu, and Wenting Cao. Dynamic anti-jamming strategy in siot: A stackelberg-matching game approach. *IEEE Transactions on Consumer Electronics*, 2024.

[34] Zijun Zhang. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, pages 1–2. Ieee, 2018.

Weixing Tan obtained a Bachelor's degree in Economics from Shandong University in 2000 and a Master's degree in Engineering in 2008. He is currently pursuing a Ph.D. at the School of Software, Shandong University, with a research focus on the application of artificial intelligence in the energy and power industry.

Yusen Wang is currently studying for a master's degree at the School of Computer and Cyberspace Security, Fujian Normal University. He graduated from Henan University of Science and Technology with a bachelor's degree in computer science and technology in 2022. His research interests include semantic communication, network security, and deep learning.

Lei Liu is a full professor in the School of Software, Shandong University, Jinan. He obtained his M.S. degree in software engineering and Ph.D. degree in computer science and technology in 2005 and 2010 from Bradford University, UK, respectively. Dr. Liu has published over 70 research papers on international conferences and journals. His research interests include network performance engineering, 5G technology, quality of service, IoT and UAVs.

Xiaoding Wang received the Ph.D. degree from the College of Mathematics and Informatics, Fu- jian Normal University, Fuzhou, China, in 2016. He is currently an Associate Professor with the College of Computer and Cyber Security, Fujian Normal University. His research interests include network optimization and fault tolerance.

Tong Ding received his B.S. and M.S. degrees in software engineering from Nanjing University of Posts and Telecommunications, Nanjing, and China University of Petroleum, Qingdao, in 2016 and 2020, respectively. Currently, he is working towards his Ph.D. degree with the School of Software at Shandong University, Jinan. His research interests include UAV path processing, reinforcement learning, and federated learning.