Postprint: Dynamic Cooperative Coverage Optimization Algorithm for UAVs Based on Trajectory Prediction in the Internet of Vehicles

Authors: Wu Zhuang, Tang Lun, Pu Hao, Wang Zhiping, Chen Qianbin

Date: 2022-04-07T15:01:57+00:00

Abstract

To address issues such as base station coverage holes and local traffic overload in urban vehicular networks, this paper proposes a dynamic pre-deployment scheme based on vehicle trajectory prediction information. First, to train a unified Seq2Seq-GRU trajectory prediction model, multiple UAVs equipped with edge computing servers operate under a distributed federated learning and blockchain architecture, eliminating the central aggregation node and adopting an improved Raft algorithm. In each training round, nodes are elected based on the volume of contributed data to complete parameter aggregation and model update tasks. Second, based on the model prediction results, an improved virtual force-guided deployment algorithm is proposed, which utilizes various virtual forces to guide UAVs in dynamic deployment, thereby enhancing vehicle access rates and communication quality. Simulation results demonstrate that the proposed training architecture can accelerate model training, while the deployment algorithm enhances vehicle access rates and improves communication quality between vehicles and UAVs.

Full Text

UAV Dynamic Collaborative Optimization Coverage Algorithm Based on Trajectory Prediction in Internet of Vehicles

Wu Zhuanga,b†, Tang Luna,b, Pu Haoa,b, Wang Zhipinga,b, Chen Qianbina,b

aSchool of Communication & Information Engineering, bKey Lab of Mobile Communication Technology, Chongqing University of Posts & Telecommunications, Chongqing 400065, China

Abstract: To address issues such as base station coverage voids and local traffic overload in urban vehicular networking environments, this paper proposes a dynamic pre-deployment scheme based on vehicle trajectory prediction information. First, to train a unified Seq2Seq-GRU trajectory prediction model, multiple UAVs equipped with edge computing servers eliminate the central aggregation node under a distributed federated learning and blockchain architecture. An improved Raft algorithm is adopted, where nodes are elected in each training round based on their contributed data volume to complete parameter aggregation and model updating tasks. Second, based on the model' s prediction results, an improved virtual force-guided deployment algorithm is proposed. Various virtual forces guide UAVs to deploy dynamically to improve vehicle access rates and communication quality. Simulation results demonstrate that the proposed training architecture accelerates model training, while the deployment algorithm enhances vehicle access rates and improves communication quality between vehicles and UAVs.

Keywords: unmanned aerial vehicle; Internet of vehicles; federated learning; blockchain; virtual force

0 Introduction

In recent years, with China' s economic development and improved urban living standards, traffic volume in cities has increased rapidly. The Internet of Vehicles (IoV), as a new paradigm integrating automotive and electronic information technologies, aims to address urban congestion and safe driving issues through artificial intelligence and information communication technologies. Future modern Intelligent Transportation Systems (ITS) urgently require intelligent, convenient, safe, and reliable IoV environments. However, in complex urban environments, various factors such as urban construction, obstacles, and inaccessible areas can cause base station coverage voids and poor communication link quality. Additionally, urban road congestion caused by rush hour traffic, accidents, and road construction, along with local traffic hotspot issues, pose serious threats to the low-latency and high-reliability requirements of IoV.

To address these problems, scholars have proposed using aerial nodes such as Unmanned Aerial Vehicles (UAVs) to assist ground vehicle communications, which has become a trend. China will commercialize 6G networks by 2030, with space-air-ground integrated networks as a crucial component aiming to achieve ubiquitous coverage through aerial nodes. UAVs, due to their low cost and agility, are widely used in agriculture, security inspection, communications, disaster relief, and other fields. As aerial nodes, they can quickly adapt to various environments and achieve large-scale coverage in complex urban environments, representing an important part of future 6G communication networks. Although base stations and Road Side Units (RSUs) have provided communication guarantees for IoV, they cannot promptly handle the aforementioned issues. UAVs' high mobility and flexible deployment offer significant advantages in solving coverage voids and emergency communications, enabling better service for

ITS when assisting base stations.

Effective network access and traffic prediction in IoV are key to UAV deployment and coordinated networking. However, traffic data in IoV is difficult to obtain. Literature [5] provides an insight: since most vehicular terminal services in IoV are periodic broadcast services with relatively stable network traffic, there is a strong correlation between network traffic and vehicle data within a certain area. Therefore, this paper uses urban vehicle migration trends and vehicle location information to describe network access and traffic prediction in IoV.

Current research on UAV deployment as aerial base stations mainly focuses on three-dimensional spatial deployment and UAV flight trajectory optimization under energy constraints. Literature [6] studied UAV-assisted edge users in ground base stations, maximizing the minimum throughput for all users in a cell by optimizing UAV trajectory, bandwidth allocation between UAVs and ground base stations, and user partitioning. Literature [7] investigated communication scheduling and user association optimization problems for multi-UAV systems supporting multiple users, accelerating algorithm convergence and improving throughput through problem decomposition. However, these scenarios do not consider that user locations change over time, which affects overall system communication throughput. Research on vehicle trajectory prediction in IoV primarily uses neural networks to learn vehicle mobility characteristics to predict possible future locations. Zhang et al. [8] employed Spatio-Temporal Residual Networks (ST-ResNet) combined with external factors such as weather and weekdays/weekends on Beijing taxi and New York bike-sharing trajectory datasets, demonstrating the network' s capability to predict citywide vehicle trajectories effectively. Zhang et al. [9] first utilized CNN to extract spatial correlations of traffic flow across urban regions, studying a Spatio-Temporal prediction model (DeepST) based on partial and global components, validating the network model' s advantages in capturing spatio-temporal characteristics on datasets. In research on spatio-temporal distribution and prediction of cellular network traffic in cities, Zhang et al. [10] designed a Spatio-Temporal Densely Connected Convolutional Neural Network for urban cellular traffic prediction. Validation using Telecom Italia datasets revealed significant temporal periodicity in various services and differences in the same service across different urban areas. Zhang et al. [11] designed a Spatio-Temporal Cross-domain Neural Network (STCNet). By collecting base station information, POI distribution, social activities, and related SMS, call, and Internet cellular service data, they conducted correlation analysis, studied stepwise quantization of spatial correlations between cross-regional datasets and cellular traffic, calculated Pearson correlation coefficients, found certain similarities among three types of cellular service data, and determined that cellular services are also affected by base station numbers and cell POIs. Based on this, the designed STCNet network can capture spatio-temporal dependencies and external influencing factors of cellular traffic, demonstrating good prediction performance. For vehicle trajectory prediction and service deployment research, Dalgkitsis [12] designed a four-layer network architecture: data center, regional server, edge computing server, and vehicles.

By using CNN for vehicle mobility trajectory prediction and genetic algorithms to dynamically migrate IoV services to the nearest edge server, they met user quality-of-service requirements. However, they did not consider communication overhead and data privacy leakage issues caused by centralized training.

Addressing the aforementioned issues of training data privacy protection, UAV deployment without considering spatial distribution changes of vehicle users, and multi-UAV collaboration, the main contributions of this paper are summarized as follows:

1) A training framework based on distributed federated learning and blockchain is designed. In this framework, multiple UAVs with edge computing capabilities use local vehicle trajectory data for training while eliminating the central aggregation node in traditional federated learning. UAVs adopt an improved Raft algorithm to compete for election as aggregation servers to complete parameter aggregation tasks, with updated parameters stored in a uniformly maintained blockchain. Each UAV node downloads parameters to continue training until completion.
2) An improved virtual force-guided algorithm is proposed. This algorithm models UAVs as charges with mutual attraction and repulsion forces and ground vehicle users as charges distributed at various locations that also exert attractive forces on UAVs. In the design of attraction between UAVs, the UAV' s energy and the number of vehicles in its coverage area affect the Coulomb force coefficient. In the repulsion design between UAVs, the safety distance serves as the coefficient to prevent collisions. In the attraction design from ground vehicles to UAVs, considering vehicle distribution locations, the force direction is from vehicles to UAVs, ensuring UAVs deploy at the center of vehicle clusters. Under the 牵引 of various forces, each UAV deploys at positions where the net force is zero.

1.1 System Model

As shown in Figure 1, considering base station coverage void areas and local traffic overload regions in cities, such as road traffic congestion areas, multiple UAVs are required to assist base stations in completing coverage optimization for these regions. The hotspot area is divided into regions, with the set of divided sub-regions denoted as . Each initialized sub-region contains a certain number of vehicles.

1) **Underlying IoV Vehicles:** Assume all vehicles in the city are equipped with GPS devices to obtain their current positions. Each vehicle has 5G transceiver equipment for accessing UAV base stations and can transmit its location information. Using a time-slot division approach, each vehicle records one position in each time slot . After every time slots, each vehicle aggregates the positions from the past time slots, i.e., , and uploads the past time slots' location information to the UAV for processing.

2) **UAVs:** Currently, UAVs are commonly used as aerial base stations for wireless access and coverage tasks due to their high mobility and agility. Edge computing, as an emerging technology, offloads computing tasks to the user equipment edge, significantly reducing task processing latency. Considering future UAVs as aerial base stations, to meet the low-latency computing requirements of IoV, they also integrate edge computing capabilities. Each region has one UAV for local coverage. Literature [5] indicates a strong correlation between network traffic and vehicle numbers within a certain IoV area. This paper uses the number of vehicles accessed by a UAV to represent its coverage capability, assuming each UAV has identical coverage capability, i.e., . UAVs in the city form a self-organizing network through networking and can inform each other of their positions via routing and forwarding.

Fig. 1 System architecture

1.2 Problem Description

Considering that vehicle users in the region are not static and vehicle distribution locations in IoV change over time, traditional mathematical models have significant limitations in characterizing vehicle trajectories. Therefore, to learn the vehicle mobility characteristics in the region, deep learning methods are required to learn vehicle trajectory change information. Single UAV node training has limitations, as it cannot interact with other UAVs or learn vehicle information across the entire region. Thus, a distributed training framework is needed to coordinate UAV training and accelerate model inference.

Traditional UAV deployment addresses single-moment deployment and energy consumption issues. UAVs originally fixed in each divided region may experience base station overload due to more vehicles moving in than out, or a sudden surge in local accessed vehicles, affecting communication quality in IoV. Conversely, more vehicles moving out than in may lead to UAV idleness due to decreased accessed vehicles, causing communication resource waste. When vehicle users' spatio-temporal characteristics change, UAV deployment energy optimization must recalculate positions as ground vehicle user locations change. Therefore, this paper designs an intelligent pre-deployment algorithm for iterative UAV position updates. To obtain mobile vehicle user distribution characteristics, this paper proposes a Seq2Seq-GRU training architecture based on federated learning and blockchain. The trained model can predict and obtain vehicle user distribution features. To optimize UAV deployment positions, an improved virtual force-guided deployment algorithm is designed for dynamic updates.

2.1 Seq2Seq-GRU

Gated Recurrent Unit (GRU) is a neural unit improved based on Recurrent Neural Network (RNN) [13]. Its network architecture is shown in Figure 2. It demonstrates superior performance over traditional RNNs when processing and predicting time series data and has a simpler structure than Long Short-Term Memory (LSTM) networks. The GRU unit structure includes two gates: an update gate and a reset gate. The calculation formulas for the update gate, reset gate, and next moment output gate are:

Vehicle trajectory data is time series data, . By knowing the location sequence, the next time location sequence can be predicted. The trajectory prediction problem can be formulated as follows:

Sequence-to-Sequence (Seq2Seq) is a structure specifically designed for time series learning and prediction. It can map input sequences of arbitrary length to variable-length output sequences. Its network model is shown in Figure 3. The Seq2Seq framework consists of two different neural networks: an encoder network and a decoder network [14]. First, the encoder network reads the input sequence and converts it into a fixed-length vector as its overall representation. When the basic neural unit of Seq2Seq is GRU, the overall representation includes the final output state vector and candidate output state vector from the encoder network. Then, the decoder uses this overall representation to initialize its internal state and gradually estimates the correct output sequence in subsequent iterations. Each step' s output represents the prediction result at that time. Typically, the decoder is designed as an autoregressive model, where the previous step' s output is used as the next step' s input.

The encoder network includes three stacked GRU layers. Each trajectory input enters each GRU unit in the first layer, with output state information fed to the next layer and the next time step' s GRU unit. The decoder adopts the same network structure as the encoder. The final updated GRU unit state of the encoder network serves as input to the decoder network. Simultaneously, the last trajectory data is input to the first GRU unit of the decoder, and so on, with each previous step' s prediction result input to the next step' s first GRU unit.

Fig. 2 GRU structure

Fig. 3 Sequence-to-sequence GRU network model

2.2 Seq2Seq-GRU Training Under Federated Learning and Blockchain

In traditional centralized training frameworks, each UAV node possesses local trajectory data. This framework requires each node to upload local data to a unified cloud server, which undergoes data cleaning and format unification before training a general neural network prediction model. After neural network model training, each UAV node downloads the network model from the cloud server. Data upload consumes substantial network resources and struggles to meet real-time application requirements. Moreover, uploading data to cloud servers risks data leakage, making it difficult to protect the security and privacy of IoV user data.

Federated learning is a distributed machine learning architecture aimed at protecting data security and user privacy [15]. It addresses issues such as small data, data silos, and slow model training. Region-divided IoV is a typical distributed network where vehicles in each region generate massive data. Uploading data to a unified center for model learning faces three challenges: first, massive data upload causes communication uplink congestion; second, data upload involves vehicle privacy issues; third, centralized training leads to slow model training.

Under a single federated learning framework, distributed model training individuals rely on a unified central node to complete parameter aggregation and model updates. However, when the central node is attacked, training stalls, model parameters cannot be updated timely, affecting the entire system model training process. Attacks can also lead to Seq2Seq-GRU model information leakage. Literature [16] points out that blockchain's decentralization and distributed ledger characteristics can solve central node vulnerability and information security issues. Therefore, this paper designs a Seq2Seq-GRU training framework based on federated learning and blockchain, characterizing each UAV node as a blockchain miner node for model learning.

UAVs in divided regions are distributed across sub-regions, representing a typical distributed structure. Each federated learning individual uses local data for training, aiming to minimize the global loss function, defined as follows:

In equation (3), is the local loss function of UAV , is the size of UAV ' s dataset, and is the size of the entire multi-UAV network dataset.

Blockchain Leader: The initial leader of the UAV group is randomly elected from the entire UAV group. Its task is to create the blockchain' s genesis block. All UAVs apply to this UAV for registration to obtain public and private keys. Subsequent leaders are competitively elected by each UAV, tasked with creating and maintaining new blocks and completing parameter aggregation and model updates.

Public and Private Keys: Each UAV uses public keys to encrypt parameter information when uploading model parameters and can use public keys to verify whether the model parameters have been attacked. Private keys are local keys kept by each UAV, providing verification and decryption of model parameter information.

Genesis Block: The first block in the blockchain, containing the initial model parameters of the Seq2Seq-GRU to be trained.

Miner: Miners are UAV nodes. Each miner can use collected vehicle trajectory data for local model training and provide maintenance and verification of blocks in the blockchain. The entire miner group can be divided into candidate groups and voter groups.

The learning and training process for multiple UAVs is shown in Figure 4. The detailed process is as follows:

Step 1: The UAV group randomly elects an initial leader, who creates the blockchain' s genesis block. Other UAVs in the group download initial model parameters, public keys, and their respective private keys. At this point, each UAV is a miner participating in the entire training process.

Step 2: Miners use local IoV vehicle datasets to perform local updates according to the stochastic gradient descent algorithm based on their local loss functions, i.e.:

where is the learning rate. Each miner encrypts model parameter information using the public key.

Step 3: To reward miners for their training contributions, miners who complete more training tasks gain opportunities to create new blocks and perform global model parameter aggregation and updates locally. The miner group elects a leader according to Algorithm 1.

Step 4: The leader creates the next block using the public key.

Step 5: Each miner digitally signs their encrypted model parameters with their private key and uploads them to the leader.

Step 6: The leader decrypts and verifies each miner' s uploaded model parameters using private keys and updates global model parameters using the federated averaging algorithm as follows:

The leader encrypts the updated model using the public key, combines it with digital signatures as transactions, and adds them to the block.

Step 7: Each terminal vehicle downloads the updated blockchain, decrypts to obtain the updated global model parameters, and then repeats the process from Step 2 based on these parameters until the global model converges or meets other termination conditions.

Algorithm 1 Improved Raft Algorithm

Input: Number of miners , effective dataset size for training by each miner, number of miner candidates , previous miner leader' s dataset size

Output: Miner leader , miner candidate group , miner voters , training round

1: if 2: for 3: if , elect candidate group, candidates gain priority for election in the next round 4: else if , divide miner group, 5: end for 6: else 7: if , previous election participants return to miner group 8: end if 9: for , elect this round' s leader from candidate group 10: end for 11: for , miner with largest training data contribution in this round enters candidate group for next round training 12: end for 13: end if

After training, each UAV can decrypt the model parameter information in the final block using its private key and use the trained Seq2Seq-GRU to predict the spatio-temporal distribution characteristics of IoV vehicles in the divided region. This serves as the basis for subsequent dynamic pre-deployment of the UAV group.

3.1 Channel Model Between UAVs and Ground IoV Vehicle Users

For simplified analysis, referring to the UAV air-ground channel model [17], define the line-of-sight transmission probability between UAVs and ground IoV vehicles as:

where is the angle between the UAV and ground IoV vehicle; and are geographical environment parameters determined by geographical factors. Meanwhile, the non-line-of-sight transmission probability is:

The average path loss for line-of-sight and non-line-of-sight transmission between UAVs and ground vehicles is:

where is the straight-line distance between UAV and ground vehicle, is the carrier frequency of the UAV' s selected channel, is electromagnetic wave propagation speed, and and are additional free-space losses in line-of-sight and non-line-of-sight cases, respectively.

The path loss between UAV and ground vehicle can be expressed as:

When multiple UAVs serve ground vehicles, ground vehicle users receive signals from various UAVs, creating superimposed interference. To ensure communication quality, ground vehicles select UAVs with SINR above their threshold. The SINR for the th ground vehicle user receiving from the th UAV is:

where is the sum of interference power from other UAVs except UAV , is Gaussian white noise power, and is the power gain between UAV and ground vehicle user, determined by . Assuming UAVs allocate equal power to each vehicle user . To ensure vehicle user communication quality, the vehicle user' s SINR must satisfy the minimum SINR constraint in equation (10).

3.2 UAV Energy Consumption Model

UAVs are energy-constrained mobile terminals. To ensure normal flight and recovery, UAV energy consumption must be modeled to update remaining energy. For simplified analysis, UAV transmission energy as a base station is ignored, focusing primarily on UAV flight energy consumption.

UAV flight behavior can be divided into two states: hovering and straight-line flight. For model simplification, UAV power in straight-line flight is set as a constant, denoted as . After flying for a period, UAV flight energy consumption is:

where is UAV flight speed, is the departure position, and is the position after flying.

In hovering state, UAV power is also set as a constant, denoted as . After hovering at the original position for a period, UAV hovering energy consumption is:

where is the time UAV stays at the original position. The average power during flight should be greater than during stationary states, i.e., .

3.3 Virtual Force Design

The Seq2Seq-GRU model learned by UAVs can predict and perceive vehicle users' next positions or positions after multiple steps, enabling pre-deployment and providing a strong basis for alleviating communication pressure in hotspot areas. To enable UAVs to pre-deploy at appropriate positions, UAV groups and vehicle users are modeled as charges with mutual virtual Coulomb forces. Several virtual forces are designed and improved to guide UAVs in accurate flight and dynamic position updates.

In divided regions, each UAV covers a different number of vehicles. When vehicle numbers in a region are excessive, the UAV requests assistance from nearby UAVs for coverage, and the UAV' s remaining energy is a limiting condition for serving the region, as shown in Figure 5(a). To characterize this relationship, virtual attraction between UAVs is modeled according to Coulomb' s law as:

where is a constant representing the ratio of minimum energy to maximum access capability; is the number of vehicles in UAV ' s coverage area at time ; is UAV ' s remaining energy at time ; is the unit vector from UAV to UAV ; and is the straight-line distance between UAV and UAV . Equation (13) shows that this attraction increases when vehicle users in a UAV' s region grow or when its remaining energy is insufficient, thereby requesting assistance from nearby UAVs.

Additionally, a safety distance exists between UAVs, not only to prevent collisions but also to ensure each UAV maintains a minimum coverage range for serving vehicles within it, as shown in Figure 5(b). Virtual repulsion between two UAVs is modeled according to Coulomb' s law as:

where is a constant representing the minimum safety distance between UAVs.

Each UAV can predict and obtain vehicle mobility trajectories and future positions in each region through the Seq2Seq-GRU network model, thereby obtaining the two-dimensional spatial distribution of vehicles in the region, as shown in Figure 6. To enable UAVs to fly above vehicle users, the attraction from user to UAV can also be modeled as Coulomb attraction:

where is a constant representing the attraction constant from ground users to UAVs; is the unit vector from user to UAV ; and is the straight-line distance between UAV and user .

Fig. 5 Gravitation and repulsion diagram between UAVs

Fig. 6 User' s gravitational force to UAV

Using force composition and decomposition from physics while ignoring forces decomposed to the axis, the three resultant forces on UAV can be decomposed into and axis components. The resultant force can be expressed as:

where and are the vector sums of various attraction and repulsion forces on the and axes for UAV .

To prevent UAVs from continuously changing forces due to small-scale vehicle user movements, making the net force non-zero, a UAV prediction deployment time interval is set. On one hand, within single or multiple time intervals, UAVs can use the prediction model to obtain vehicle positions after single or multiple steps. On the other hand, within each time interval, vehicle users are assumed to move to fixed positions and receive service from a UAV. When UAVs in the divided region fly to positions where the resultant force is zero within this time interval, coverage for that moment is completed.

When UAVs assist nearby UAVs, vehicle users in the overlapping coverage area of two UAVs calculate the SINR with the assisting UAV. If higher than the original SINR, they switch to that UAV. Therefore, ground vehicle users in this paper select the maximum SINR for access. The entire assisted deployment is shown in Algorithm 2.

Algorithm 2 UAV Dynamic Assisted IoV Vehicle User Coverage Access

Input: UAVs' Seq2Seq-GRU prediction model, time interval , UAV deployment position update interval , attraction coefficient between UAVs , attraction coefficient from vehicle users to UAVs , repulsion coefficient between UAVs , UAV access capability , UAV minimum energy

Output: UAV flight trajectories and deployment positions at each time interval.

Initialization: UAV initial positions

1: for in do 2: Predict vehicle users' spatial distribution after time and update ; UAV calculates the position where resultant force is zero according to equation (16) 3: if , fly back to base 4: else if , fly to position where ; update according to equations (11) and (12) 5: else if , vehicle users switch access to 6: end if 7: end for

Two key factors affect the deployment algorithm' s complexity: the number of UAVs and the time to calculate zero resultant force. Assuming single UAV calculation time is , the overall deployment algorithm time complexity is .

4.1 Seq2Seq-GRU Training Based on Blockchain Federated Learning

To evaluate the proposed Seq2Seq-GRU algorithm performance, two evaluation metrics are adopted: 1) Mean Squared Error (MSE), measuring the average error between predicted and actual position data for multiple users; 2) Straight-line distance, representing the distance between predicted and actual positions.

MSE is defined as:

where is user ’ s predicted position and is user ’ s actual position.

Straight-line distance is calculated based on latitude and longitude data between two locations using the formula:

where is Earth’ s radius, and are predicted latitude/longitude positions, and and are actual latitude/longitude positions.

This experiment uses a real dataset: GPS coordinates of 500 taxis collected over 30 days in the San Francisco Bay Area, USA. Each taxi sample includes latitude, longitude, occupancy, and timestamp (occupancy is not used in this simulation). The simulation platform uses PyCharm 2020.3 and Anaconda 3, with Python 3.7 as the programming language and TensorFlow as the deep learning framework.

The dataset is divided into training, validation, and test sets at ratios of 80%, 10%, and 10%. To compare the proposed Seq2Seq-GRU trajectory prediction model performance, traditional machine learning methods are used as baselines, including linear regression, support vector machines, and deep learning-related RNN and LSTM. MSE simulation results are shown in Figure 7. Another model evaluation metric is the straight-line distance between single-step predicted positions and actual positions, with simulation results shown in Figure 8.

Fig. 7 MSE of different models

Fig. 8 Distance difference between predicted position and actual position of each model

Figures 7 and 8 show that Seq2Seq-GRU’ s MSE and prediction errors are significantly smaller than traditional regression methods and perform better than RNN and LSTM-related models. Additionally, the attention mechanism (Seq2Seq-Attention) does not outperform the conventional Seq2Seq framework. Since attention mechanisms primarily address word order mismatch issues between input and output sentences in machine translation, while trajectory data is a time series collection of positions with left-to-right sequential relationships, and short-term trajectories have fine-grained characteristics, global information such as speed and direction is more important for trajectory prediction.

When performing multi-step prediction, errors accumulate as prediction steps increase. Table 1 statistics show the error distances between multi-step predicted positions and actual positions for each method.

Table 1 Error distance between multi-step prediction and actual position of each method (m)

From Table 1, although Social-Scene-LSTM has lower single-step prediction error than Seq2Seq-GRU, its cumulative error increases more significantly with prediction step length, resulting in unstable model predictions.

To accelerate model training and protect local vehicle location data privacy, a federated learning and blockchain framework is introduced during training. Figure 9 compares the model training convergence speed between a centralized framework and the proposed framework with 10 nodes. The centralized framework trajectory prediction model begins converging around round 160, while the federated learning and blockchain framework converges by round 20, proving the proposed framework accelerates training.

Fig. 9 Convergence diagram of training model for centralized training and federated learning

4.2 Improved Virtual Force Deployment

Relevant simulation parameters for the pre-deployment algorithm are shown in Table 2.

Table 2 Simulation parameter settings

To validate the dynamic UAV coverage deployment optimization algorithm, four deployment algorithms are compared: UAVs stationary in their regions, UAVs repeatedly moving horizontally between adjacent regions, random movement deployment, and improved virtual force-guided deployment. Coverage access rate over time is compared, where coverage rate is defined as:

Simulation results for the four algorithms at different times are shown in Figure 10.

Fig. 10 Four algorithms cover the access rate at different times

The improved virtual force deployment algorithm maintains a stable coverage access rate around 92% over time, showing significant improvement over the other three algorithms.

Figure 11 compares the remaining energy consumption of the four deployment algorithms at each moment.

Fig. 11 The sum of the remaining energy of each UAV at different times

The improved virtual force deployment algorithm consumes more energy than the other three algorithms because UAVs incur significant flight energy consumption to meet deployment requirements at different times and positions.

Additionally, the average SINR of ground vehicle users under the maximum SINR access algorithm is compared across the four deployment algorithms, as shown in Figure 12.

Fig. 12 Average signal-to-interference-noise ratio of ground vehicle users at different times with four algorithms

The improved virtual force deployment algorithm maintains vehicle user SINR stable around 12.5dB over time, showing significant improvement over the other three algorithms.

Comprehensive analysis of Figures 10-12 reveals that the algorithm sacrifices UAV energy to improve communication performance.

UAVs can perform multi-step prediction and deployment. Simulations compare single-step and multi-step prediction coverage rates versus energy consumption. Average coverage rate for single-step and multi-step prediction is defined as the stable mean coverage rate of the UAV group across multiple experiments when using the improved virtual force pre-deployment algorithm. Average remaining energy rate is defined as the experimental average ratio of total remaining energy to initial total energy after UAVs perform corresponding step predictions and adjust deployment positions. Simulation results are shown in Figure 13.

Fig. 13 The relationship between the average coverage rate and the average remaining energy rate and the adjustment step size

As prediction step size increases, UAV average coverage rate gradually decreases while average remaining energy rate increases, showing a trade-off between the two. To maintain optimal balance, the optimal pre-deployment adjustment step size should be around 2.5 steps.

5 Conclusion

This paper proposes a deep learning training method based on federated learning and blockchain. Additionally, an improved virtual force pre-deployment algorithm is designed for multi-UAV collaboration. To protect privacy of IoV vehicle users in urban traffic hotspot areas and improve distributed UAV network model training efficiency, this paper proposes an edge computing node-coordinated federated learning and blockchain framework. This framework effectively achieves collaborative training and model updates among UAV nodes. Each UAV can predict urban vehicle mobility trajectories and obtain spatio-temporal distribution characteristics of underlying vehicle users using the trained model. The improved virtual force pre-deployment algorithm, combined with the maximum SINR access algorithm for underlying vehicle users, enables UAV nodes to deploy at appropriate positions under forces from neighboring nodes and vehicle users. This algorithm achieves collaborative deployment among nodes and guarantees vehicle user access quality. Compared with four other deployment algorithms, the improved virtual force deployment algorithm significantly improves access rate and average SINR. Simulations also demonstrate the trade-off between energy consumption and UAV coverage. Results show that UAVs performing prediction and adjustment deployment every 2.5 steps achieve a good balance.

References

[1] Zhang Jun, Letaief K B. Mobile edge intelligence and computing for the internet of vehicles [J]. Proceedings of the IEEE, 2019, 108 (2): 246-261.

[2] Lin Yan, Yan Shuai, Zhang Yijin, et al. Flow-of-traffic prediction program based mobile edge computing for internet of vehicles using double auction [J]. Journal on Communications, 2020, 41 (12): 205-214.

[3] Fan Xiying, Huang Chuanhe, Zhu Junyu, et al. Interference-aware node access scheme in UAV-aided VANET [J]. Journal on Communications, 2019, 40 (6): 90-101.

[4] Jawhar I, Mohamed N, Al-Jaroodi J, et al. Communication and networking of UAV-based systems: Classification and associated architectures [J]. Journal of Network and Computer Applications, 2017, 84: 93-108.

[5] Ma Huisheng. Research on wireless transmission technology and optimization methods of 5G-V2X for cooperative autonomous driving [D]. Beijing University of Posts and Telecommunications, 2020.

[6] Lyu Jiangbin, Zeng Yong, Zhang Rui. Spectrum sharing and cyclical multiple access in UAV-aided cellular offloading [C]//GLOBECOM 2017-2017 IEEE Global Communications Conference. IEEE, 2017: 1-6.

[7] Wu Yang, Fan Wenlu, Yang Weiwei, et al. Robust trajectory and communication design for multi-UAV enabled wireless networks in the presence of jammers [J]. IEEE Access, 2019, 8: 2893-2905.

[8] Zhang Junbo, Zheng Yu, Qi Dekang, et al. Predicting citywide crowd flows using deep spatio-temporal residual networks [J]. Artificial Intelligence, 2018, 259: 147-166.

[9] Zhang Junbo, Zheng Yu, Qi Dekang, et al. DNN-based prediction model for spatio-temporal data [C]//Proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems. 2016: 1-4.

[10] Zhang Chuanting, Zhang Haixia, Yuan Dongfeng, et al. Citywide cellular traffic prediction based on densely connected convolutional neural networks [J]. IEEE Communications Letters, 2018, 22 (8): 1656-1659.

[11] Zhang Chuanting, Zhang Haixia, Qiao Jingping, et al. Deep transfer learning for intelligent cellular traffic prediction based on cross-domain big data [J]. IEEE Journal on Selected Areas in Communications, 2019, 37 (6): 1389-1401.

[12] Dalgkitsis A, Mekikis P V, Antonopoulos A, et al. Data driven service orchestration for vehicular networks [J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22 (7): 4100-4109.

[13] Cho K, Merrienboer B V, Bahdanau D, et al. On the properties of neural machine translation: Encoder-Decoder approaches [EB/OL]. (2014-09-03).

https://arxiv.org/abs/1409.1259.

[14] Wang Chujie, Ma Lin, Li Rongpeng, et al. Exploring trajectory prediction through machine learning methods [J]. IEEE Access, 2019, 7: 2017-2017.

[15] Zhao Yu, Yang Jie, Liu Miao, et al. Federated learning based intelligent edge computing technique for video surveillance [J]. Journal on Communications, 2020, 41 (10): 109-115.

[16] Qu Youyang, Pokhrel S R, Garg S, et al. A blockchained federated learning framework for cognitive computing in industry 4.0 networks [J]. IEEE Transactions on Industrial Informatics, 2020, 17 (4): 2964-.

[17] Wu Weiyu, Zhao Haitao, Wang Haijun, et al. Distribute deployment and access selection algorithm for UAV airborne networks [J]. Chinese Journal of Computers, 2019, 42 (02): 121-137.

Note: Figure translations are in progress. See original paper for figures.