TOWARDS TRAJECTORY PREDICTION-BASED

UAV DEPLOYMENT IN SMART TRANSPORTATION SYSTEMS

Institute for Homeland Security

Sam Houston State University

Fan Liang

Xing Liu

Nuri Alperen Kose

Kubra Gundogan

Wei Yu

Towards Trajectory Prediction-based UAV Deployment in Smart Transportation Systems

Fan Liang*, Xing Liu*, Nuri Alperen Kose*, Kubra Gundogan*, and Wei Yu †
* Dept. of Computer Science, Sam Houston State University, USA
Emails: {fxl027, xxl020, nxk022, kxg067}@shsu.edu
† Dept. of Computer and Information Sciences, Towson University, USA
Emails: wyu@towson.edu

Abstract—A smart transportation system (i.e., intelligent trans- portation system) refers to a transportation critical infrastructure system that integrates advanced technologies (e.g., networking, distributed computing, big data analytics, etc.) to improve the efficiency, safety, and sustainability of the transportation system. However, the rapid increase in the number of vehicles on roads and significant fluctuations in the flow of traffic can cause the cov- erage holes of Road Side Units (RSUs) and local traffic overload in smart transportation systems, which can negatively affect the performance of systems and causes accidents. To address these issues, deploying Unmanned Aerial Vehicles (UAVs) as mobile RSUs is a viable approach. Nonetheless, how to deploy UAVs to the optimal position in the smart transportation system remains an unsolved issue. This paper proposes a Vehicle Trajectory-based Dynamic UAV Deployment Algorithm (VTUDA). The VTUDA utilizes vehicle trajectory prediction information to improve the efficiency of UAV deployment. First, we deploy a distributed Seq2Seq-GRU model to the UAVs and train the model. We leverage the well-trained model to predict vehicle trajectory. VTUDA then uses the predicted information to make informed decisions on the optimal location to position the UAVs. Further- more, VTUDA considers both the condition of communication channels and energy consumption during the deployment process to ensure that UAVs are deployed to optimal positions. Our experimental results confirm that the proposed VTUDA can effectively improve the deployment of UAVs. The experimental results also demonstrate that VTUDA can significantly enhance vehicle access and communication quality between vehicles and UAVs.

Index Terms—Smart Transportation Systems, Edge Comput- ing, UAV Deployment, Machine Learning

I. Introduction

Smart transportation is a rapidly evolving field that aims to improve the efficiency, safety, and sustainability of transportation systems [1]. Smart transportation is a integration of advanced technologies such as the Internet of Things (IoT), big data, and distributed computing, which plays a crucial role in addressing the challenges faced by modern transportation systems, such as urban congestion and safe driving [2], [3]. To support communication, Vehicle Ad hoc Networks (VANETs) are adopted as the key to supporting Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I), and Vehicle to everything (V2X) communication [4]. However, the complexities of urban environments pose significant challenges to the development of VANETs.

For example, with the growing number of vehicles on the road, urban vehicle networks face the challenge of local traffic overload. These issues can negatively impact communication quality and accessibility for vehicles, reducing the overall performance of smart transportation systems. In addition, the complex environment of cities, including urban constructions, obstacles, and inaccessible areas, can cause problems such as base station coverage holes and poor communication link quality. Additionally, urban road congestion and local traffic hotspots can pose a serious threat to the low latency and high-reliability requirements of VANETs.

In response to the above problems, the deployment of UAVs become a growing trend to support communication in areas that are without communication coverage or the base station is overloaded (critical areas). There are many existing studies that have proposed using UAVs to carry the mini-base station as aerial nodes to assist ground vehicle communication [5], [6], [7]. The deployment of UAVs aims to achieve full- area coverage through aerial nodes. UAVs are widely used in agriculture, security inspection, communication, disaster rescue, and other fields because of their affordability [8], fast response, and versatility. Not only can it quickly adapt to various environments, but as an aerial node, it can also achieve large-scale coverage of complex environments within the city, which is an important part of the future communication network. In addition, although base stations and roadside units (RSUs) have provided communication guarantees for the communication of smart transportation, considering the high price and not flexible, they cannot deal with the overloading and coverage holes problems in a timely manner. Therefore, deploying UAVs to support smart transportation has obvious advantages in solving coverage holes and emergency communications.

In order to deploy UAVs to critical areas, identifying critical areas is important. Generally, existing studies focus on adopting effective network access and network traffic as the key factors in identifying critical areas. Since most of the services in the VANETs are periodic broadcast services, the number of vehicles in this area directly impacts the network traffic which means there is a strong correlation between network traffic and road traffic. However, road traffic is constantly changing, and the network traffic also follows. The complexity of the dynamic environment brings uncertainties. How to efficiently deploy UAVs to support communication is still a challenge. Focus on the challenge, in this study, we consider road traffic, wireless communication channels, and UAV energy consumption to propose a co-design of a UAV deployment algorithm, which is Vehicle Trajectory-based Dynamic UAV Deployment Algorithm (VTUDA). In detail, we first leverage vehicle trajectory, location information, and moving trends to predict road traffic. Based on the road traffic prediction, we can describe the network traffic since the strong correlation between network traffic and road traffic in VANETs. Then, we also consider the wireless communication channel qualities. We design a wireless communication model between UAVs and vehicles according to the Air-to-Ground Propagation Channel Model [9]. Based on the communication model, we can evaluate the communication channel quality to optimize the UAV deployment. Furthermore, we also consider the energy consumption of UAV, specifically, we consider the energy consumption of UAV hover and fly, in order to ensure the flight of UAVs.

To summarize, in our study we make the following contributions.

- **Vehicle Trajectory Prediction:** We introduce a novel approach for vehicle trajectory prediction, leveraging distributed federated learning. To achieve this, we first employ a Seq2Seq-GRU framework to develop a multi- input multi-output model that takes in historical vehicle trajectory data as input and generates a predicted trajectory for a specified time in the future. Our proposed approach offers an effective solution for vehicle trajectory prediction that can be applied in a range of real- world scenarios. In addition, to enhance performance and decrease latency, we deployed the proposed model in a distributed manner, taking into account the distributed structure of the smart transportation system. Specifically, we distributed the model across the RSUs, allowing for efficient processing and faster prediction times. By doing so, our approach offers a significant improvement in the overall performance of the smart transportation system.

- Vehicle Trajectory-based Dynamic UAV Deployment: We propose a co-design of trajectory prediction, wireless communication channels, and UAV energy consumption which is the Vehicle Trajectory-based Dynamic UAV Deployment Algorithm. In detail, to deploy the UAV at the position that obtains the optimal coverage rate, we consider the impacts of the number of vehicles in the UVA coverage area based on the predicted road traffic, meanwhile the safety distance between each UAV to formalize the system as a force field and deploy the UAV at the position where the resultant force is zero.

II. RELATED WORKS

This section introduces the related work of UAV deployment algorithms and task-offloading schemes in smart transportation systems. In the smart transportation system, RSU generally acts as an edge node server to provide computing and commu- nication services for moving vehicles [6], [16]. Nonetheless, due to the fixed position of the RSU, considering the cost of large-scale deployment, it is impossible to achieve the full coverage of RSUs in some road sections in practical applications. To ensure the communication quality and user experience of the smart transportation, UAVs are considered promising solutions for temporary and dynamic edge server nodes [17]. For instance, Yu et al. [18] proposed a UAV- enabled mobile edge computing system, which is optimized by minimizing the UAV energy consumption and communication latency. Likewise, Seid et al. [19] leveraged UAVs as aerial base stations to assist the edge network in enhancing ground network performance (e.g., extending network coverage).

Due to the distributed and dynamic nature of VANETs, purely theoretical models have limitations in finding optimal task offloading schemes [20]. As a machine learning algorithm that can find the optimal strategy through trial and error, deep reinforcement learning has been widely used in task offloading of edge computing in scenarios [21], such as mobile edge computing [22], smart home [23], and smart healthcare [24], among others. In terms of intelligent transportation, deep rein- forcement learning has been applied to the task offloading op- timization of smart transportation systems [25]. For instance, He et al. [26] proposed deep reinforcement learning model with prioritized experience replay and stochastic weight aver- aging for enhancing the satisfaction of quality of experience (QoE). Likewise, Ning et al. [27] adopted distributed deep Q- learning to minimize the offloading cost while satisfying the latency constraint of users.

The task-offloading for ensuring the quality of experience is critical due to the diverse task requirements and dynamic wireless communication environment in smart transportation systems [28], [29]. The task allocation schemes that minimize the task processing latency has been researched recently. For instance, Xu et al. [30] proposed a task offloading scheme based on game theory and fuzzy neural networks. Furthermore, Raza et al. [31] analyzed the computation efficiency for limited battery Electrical Vehicles (EVs) while considering the trade- off between computing time and energy consumption.

Table I

Symbols	Descriptions
$P_{i,j}^{LOS}$	Probability of having a LOS link between u_i and v_j
$L_{i,j}^{LOS}$	Path loss for LoS link between u_i and v_j
U	UAVs
V	Vehicles
B	Buildings
h_{u_i}	Height of u_i
h_{v_i}	Height of v_i
$d_{i,j}$	Distance between u_i and v_j
μ	Ratio of built-up land to the total area
ς	Average number of buildings per unit area
η	Free space loss for LOS and NLOS
λ	Wavelength of the radio wave
γ	Path loss exponent
z_t	Reset gate
r_t	Update gate
h_t	Hidden state
l_t	Location information at time t
a_t	Acceleration at time t
$p_{i,j}$	Transmitting power from u_i to v_j
$g_{i,j}$	Power gain
E^f	Total flying power consumption
E^h	Total hovering power consumption

In addition, there are a number of studies that focus on the network resource allocation field to avoid network congestion and improve performance. For example, in [32], a wireless sensor network was utilized for the deployment and simulation analysis of sensor nodes within a particular environment. Also, Akram et al. [33] made an attempt to create the maximum capacity space for objects by using the network adaptive learning approach for trajectory prediction.
Similarly, Tossa et al. [34] proposed a scheme for finding the optimal location for sensor nodes in wireless sensor networks with genetic algorithms, while considering coverage and connectivity. Xue *et al.* [35] suggested reducing position inaccuracies caused by sensitivity when computing the coordi- nates of beacons, unknown nodes, and devices. The proposed scheme focuses on the least-squares location in order to design a solution, which is both faster and more accurate than the non-optimized scheme concerning location. Likewise, Metaaf and Wu [36] designed schemes to improve the deployment optimization, minimizing energy consumption and extending the life of the network.

III. SYSTEM MODEL

In this section, we begin by exploring the problem space of deploying UAVs to smart transportation, Then, we present our design rationale. Afterward, we introduce our proposed system model.

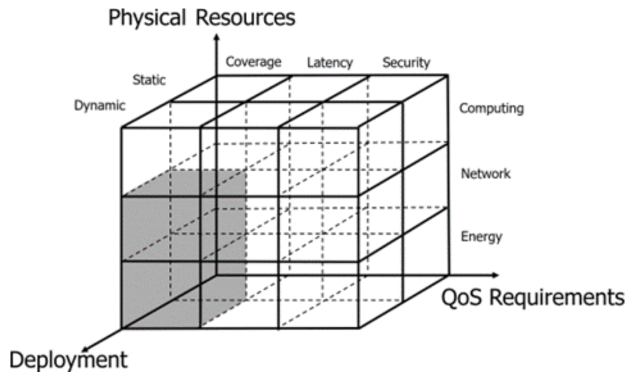

Fig. 1. Problem space of deploying UAVs in smart transportation

A. **Design Rationale**

We identify the problem space of deploying UAVs in smart transportation which is shown in Fig. 1. Here, we define a three-dimensional space that includes physical resources, QoS requirements, and deployment. As shown with the shadow in the figure, in this study, we focus on improving the coverage rate considering network conditions and energy consumption. Furthermore, we propose a dynamic deployment algorithm to handle real dynamic road traffics. In detail, in a typical smart transportation scenario, RSUs are deployed beside roads as distributed computing nodes to response to the demands of vehicles. Due to considerations of reliability, availability, and cost, it is unfeasible to deploy a significant amount of RSUs to meet the highest demand. Thus, in a large city, there are many communication coverage holes. In addition, due to the dynamic of road traffic, the traffic density in- creases significantly during peak time. The RSUs don't have any flexibilities to handle this situation. Therefore, deploying UAVs as mobile RSUs to assist smart transportation systems to mitigate the harms of the aforementioned issues is becoming a popular approach. However, road traffic is highly dynamic as we discussed, and employing the current road information to deploy UAVs is inadequate in responding to sudden changes in traffic. Thus, road traffic prediction is necessary to be involved to improve the performance of the UAV deployment algorithm. To this end, based on the vehicle trajectory prediction, we consider the wireless communication channels and energy consumption and format the problem as a co-design of a multifeature optimization problem.

To predict the vehicle trajectory, we deploy the Seq2Seq- GRU model to RSUs and operate a distributed federated learn- ing. The Gated Recurrent Unit network (GRU) is an improved version of the Recurrent Neural Network (RNN). The GRU has better performance on time series data predictions and has a simpler structure than the Long Short Term Memory network (LSTM). GRU only consists of two gates which are update gate and reset gate. The update gate determines how much of the previous hidden state to retain and how much of the new information to incorporate. The reset gate determines how much of the previous hidden state to forget. By con- trolling these gates, the GRU is able to selectively preserve or discard information, allowing it to better handle long-term dependencies in sequential data. Since the smart transportation system is a distributed system, we deploy the Seq2Seq-GRU model to distributed RSUs and operate a distributed federated learning.

Finally, according to the result of vehicle trajectory pre- diction, we design a wireless communication channel model based on the number of connections and channel interfer- ences. In addition, to make sure reliable power supply, we consider the energy consumption for UAVs. By incorporating the above-mentioned features, we formalize the system as a comprehensive force field. The optimal position for the UAV can be determined by finding the spot where the net force is equal to zero.

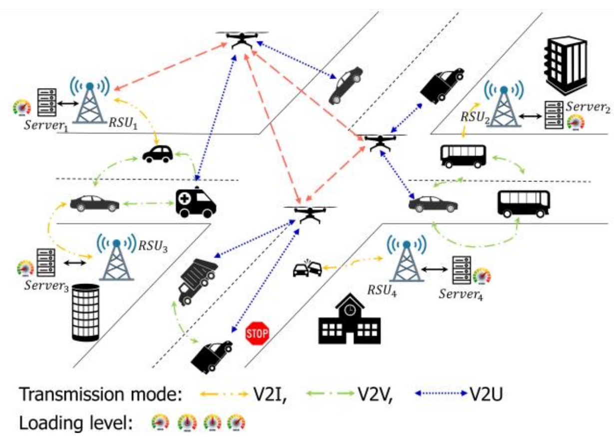

Fig. 2. System structure of deploying UAVs in smart transportation

B. System Model

We define a typical smart transportation scenario which is shown in Fig. 2. In this complex road traffic scenario, some RSUs are fully loaded. Thus, we deploy UAVs as mobile RSUs to response to the demands of vehicles. We assume a GPS device equips each vehicle and sends location information at each time slot sequentially by 5G wireless communication channel to RSUs. In addition, in our study, we assume all UAVs have the same coverage radius and UAVs can be mobile RSU and communicate with other RSUs by VANET.

We denote UAVs as $U \in \{u_1, u_2, u_3, \cdots, u_n\}$ and vehicles as $V \in \{v_1, v_2, v_3, \cdots, v_n\}$. In addition, the buildings in this area also can be represented by $B \in \{b_1, b_2, b_3, \cdots, b_n\}$ Since this is an urban area, the probability of having a line-of-sight (LOS) link between one of the UAVs u_i and one of the vehicles v_j is defined as equation (1). Here, h_{u_i} denotes the height of the u_i , h_{u_i} denotes the height of the v_j and h_B denotes the height of the building. We use a probability distribution function $F(h_B)$ to represent the height distribution of buildings. Furthermore, we have $k = \lfloor d_{i,j} \cdot (\mu \cdot \varsigma)^{-2} - 1 \rfloor$. We denote $d_{i,j}$ as the distance between u_i and v_j, μ as the ratio of built-up land to the total area, and ς is the average number of buildings per unit area.

$$P_{i,j}^{LOS} = \prod_{n=0}^{k}\left[1-\exp\left(-\frac{\left[h_{u_i}-\frac{(n+1/2)\left(h_{u_i}-h_{v_j}\right)}{k+1}\right]^2}{2\cdot F(h_B)^2}\right)\right]. \qquad (1)$$

In general speaking, the height of vehicles can be approximately zero, hence, we update the connectivity probability of the LOS link as equation (2). Based on the connectivity probability of the LOS link we also can get the connectivity probability of the non-LOS (NLOS) link as $P_{i,j}^{NLOS} = 1 - P_{i,j}^{LOS}$ We consider the case as a mixture path loss model, the average pass loss $L_{i,j}^{AVG}$ can be represented by equation (3).

$$P_{i,j}^{LOS} = \prod_{n=0}^{k_{i,j}}\left[1-\exp\left(-\frac{\left[h_{u_i}-\frac{(n+1/2)\cdot h_{u_i}}{k+1}\right]^2}{2\cdot F(h_b)^2}\right)\right]. \qquad (2)$$

$$L_{i,j}^{AVG} = P_{i,j}^{LOS}\cdot L_{i,j}^{LOS} + P_{i,j}^{NLOS}\cdot L_{i,j}^{NLOS}. \qquad (3)$$

The $L_{i,j}^{LOS}$ and $L_{i,j}^{NLOS}$ denote path loss of LOS and NLOS link and can be represented by equation (4).

$$\begin{cases} L_{i,j}^{LOS}(d) = L_0^{LOS} + 10\gamma\log(d/d_0) + \eta_{LOS} \\ L_{i,j}^{NLOS}(d) = L_0^{NLOS} + 10\gamma\log(d/d_0) + \eta_{NLOS} \end{cases} \qquad (4)$$

The $L_{i,j}^{LOS}$ and $L_{i,j}^{NLOS}$ denote path loss at reference distance d_0. $10\log\left[\left(\frac{4\pi d_0}{\lambda}\right)^2\right]\cdot\gamma$ is the path loss exponent, which is obtained by using minimum mean square error, and λ is the wavelength of the radio wave. Finally, η_{LOS} and η_{NLOS} represent free space loss for LOS and NLOS link respectively.

Due to the presence of multiple UAVs offering services to vehicles, the communication channel can be disrupted by overlapping signals. To mitigate the effects of signal interference, vehicles opt for a channel with the highest Signal-to- Interference-plus-Noise Ratio (SINR) to ensure optimal communication. We define SINR between the i^{th} UAV to the j^{th} vehicle as equation (5). The $p_{i,j}$ denotes sending power from u_i to v_j and $g_{i,j}$ denotes the power gain. $\sum_{n=1,n\neq i}^{n} p_{n,j}\cdot g_{n,j}$ is the sum of the interference for all other UAVs except u_i and N is white Gaussian noise.

$$SINR_{i,j} = \frac{p_{i,j}\cdot g_{i,j}}{\sum_{n-1,n\neq i}^{n} p_{n,j}\cdot g_{n,j}+N}. \qquad (5)$$

Now we present the energy consumption model for UAVs. Recall the energy consumption for UAVs, which includes flying energy consumption and hovering energy consumption. We define constant e^f to represent the flying energy consumption per time unit and e^h to represent the hovering energy consumption per time unit. Therefore, the total energy consumption E^f for flying can be calculated by equation (6). Similarly, the total energy consumption E^h for hovering can be calculated by equation (7).

$$E^f = e^f \Delta t = e^f \cdot \frac{[(x'-x)(y'-y)]^{-2}}{v_i^f}. \quad (6)$$

$$E^h = e^h \Delta t. \quad (7)$$

IV. Our Approach

In this section, we introduce our approach to deploying UAVs in smart transportation. First, we present the vehicle trajectory prediction. Specifically, we define our distributed Seq2Seq-GRU model and loss function. Then, we introduce our proposed UAV deployment algorithm in detail.

A. Vehicle Trajectory Prediction

Recall the key factor in deploying UAVs to the demand position is to obtain vehicle trajectory information. However, the high dynamic traffic causes difficulty to use traditional mathematical models to represent the accurate vehicle movement tendency. To conduct an accurate model that represents the vehicle movement patterns in a certain area, it is necessary to adopt deep learning methods to learn the related historical information, such as velocity, acceleration, location, etc. However, limited by the computing power for RSUs, training the model in one RSU costs a long time and doesn't satisfy the response time requirement for dynamic road traffic. To overcome these limitations, a distributed learning framework is required to coordinate the training of multiple RSU nodes (including RSUs and mobile RSUs), leading to a faster and more comprehensive understanding of traffic movement pat- terns. We now present the proposed federated learning model in the following.

1. Seq2Seq-GRU:

GRU is a type of recurrent neural net- work (RNN) architecture that is well-suited for processing sequential data. One of the major advantages of GRU over traditional RNNs is that it can overcome the vanishing gradient problem that can occur when training RNNs on long sequences of data. Given that vehicle trajectory can be organized as a time series data, it is possible to leverage the power of GRU to predict vehicle trajectory. By processing the historical positions and velocities of the vehicle as a sequence of data, a GRU model can learn to predict the future positions and velocities of the vehicle.

In our study, we let $x_t = l_t$. Here, x_t is one input of the node in GRU. Also, $l_t = \{(x_t, y_t), a_t\}$, (x_t, y_t) is the current location and a_t is acceleration data. Hence, we define the reset gate and update gate as follows.

$$\begin{cases} z_t = \sigma(W_z \odot \{h_{t-1}, l_t\}) \\ r_t = \sigma(W_r \odot \{h_{t-1}, l_t\}) \end{cases} \tag{8}$$

The hidden state h_t can be represented by $h_t = (1 - z_t) \odot h_{t-1} + z_t \odot h'_t$ and $h'_t = \tanh(W \odot \{r_t \odot h_{t-1}, l_t\})$.We organize vehicle trajectory data as time series data that is $\{l_t, l_{t+1}, \cdots l_{t+k}\}$ and we can model the movement of a vehicle as a sequence of states, where each state represents the location of vehicle at a specific point in time. The goal is to find a function that maps the previous sequence of states to the future sequence of states. The vehicle trajectory prediction problem can be represented by equation (9).

$$l_{t+1} = z_t \odot \{h_t \odot r_t, l_t\} + (1 - z_t) \odot h_t \tag{9}$$

In the smart transportation context, improving the accuracy of predicting the trajectory of a vehicle requires providing a sequence of historical trajectories as input to the model. By doing so, the model can learn to identify patterns and dependencies in the vehicle's motion, and use this information to generate a more accurate prediction of the next location in the future. In addition, in our study, we need to obtain the future vehicle trajectory which is a time series sequential location set. However, the typical GRU cannot output a time series sequential location set.

Seq2Seq (Sequence-to-Sequence) models are a type of neural network architecture, those models consist of an en- coder and decoder. One of the key advantages of Seq2Seq models is their ability to capture context and dependencies within the input sequence. This is accomplished through the use of an encoder-decoder architecture, where the encoder processes the input sequence and produces a fixed-length vector representation, which is then used by the decoder to generate the output sequence. This allows Seq2Seq models to generate more accurate and coherent output sequences than other types of models. Specifically, we input k past moments to the Seq2Seq model, the model generates a prediction of the vehicle's trajectory for k' future moments. Generally, the encoder and decoder of Seq2Seq are RNNs. In order to improve the performance, in our study, we leverage GRU as the encoder and decoder.

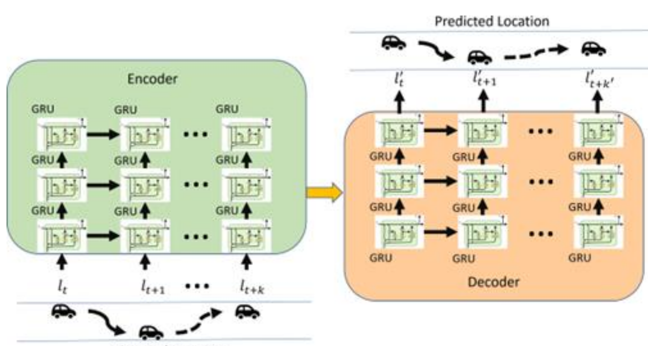

Fig. 3. Proposed Seq2Seq-GRU model

Fig. 3 shows the structure of our proposed Seq2Seq-GRU model. Specifically, the encoder network of the model consists of three GRU layers and each layer has k nodes. We group k sequential historical location information as input to each GRU node in the first layer, and the resulting output state information is then fed as input to the next layer, along with the next moment in the trajectory sequence. This process is repeated for all three layers, allowing the encoder to capture the temporal dependencies in the input trajectory.

The decoder network has the same structure as the encoder network, with the last updated state of the final GRU unit in the encoder network being used as the initial state of the decoder network. Additionally, the last trajectory data is fed as input into the first GRU unit of the decoder, and the prediction result from the previous step is fed as input into the first layer of the GRU unit in the next step. This approach allows the model to generate accurate predictions for each step of the trajectory sequence, while also taking into account the previous predictions to improve the accuracy of future predictions.

1. Distributed Federated Learning Framework:

In the traditional centralized training framework for smart transportation systems, each computing node (RSU) holds local trajectory data, which needs to be uploaded to a cloud server for training a common neural network predictive model. Then, the well-trained model is deployed to RSUs to perform the prediction. However, this approach has several challenges. Firstly, data upload can consume a significant amount of network resources, leading to congestion and affecting realtime applications. Secondly, uploading data to a centralized server poses a risk of data leakage, compromising user privacy and security. Federated learning is a distributed learning architecture that can address the challenges of small data, data islands, and slow training speed while protecting data security and user privacy. The smart transportation system is a typical distributed system, that generates a large amount of data that can be used for model training. Therefore, we leverage the horizontal federated learning framework to implement the proposed Seq2Seq-GRU model. We define the global loss function as equation (10), where D_i is the size of training data on u_i, $\mathrm{loss}_i\ (w)$ is the local loss function for u_i and D is the size of total training data. In our proposed model, we use Mean-Squared Loss as the loss function, and equation (11) represents the loss function. $\hat{L}_i$ is the predicted location and L_i is the real location.

$$\mathrm{Loss}(w) = \frac{\sum_{1-1}^{n} D_i \cdot \mathrm{loss}_{\mathrm{v}}(w)}{D}. \quad (10)$$

$$\mathrm{loss}\ (w) = \frac{\sum_{i-1}^{n} (\hat{L}_x - L_i)^2}{n}. \quad (11)$$

B. Vehicle Trajectory-based Dynamic UAV Deployment Algorithm

In this subsection, we present the proposed VTUDA in detail. After leveraging the Seq2Seq-GRU model, we obtain the prediction of the vehicle trajectory in near future (T time steps). By doing so, the model facilitates pre-deployment of UAVs in strategic locations, which can alleviate communication pressure in hotspot areas. Furthermore, we also consider the communication channels, neighbor UAVs, and energy consumption to conduct a co-design to deploy UAVs to the optimal position.

To accomplish this, we utilize a force-based model that treats the communication channels, energy conditions, and number of users as component forces. We update several virtual forces to lead the UAVs to perform accurate flight and position dynamic updates. Equation (12) represents the attractive force between two UAVs u_i and u_i. Here u_i^t and u_{max} represent the number of connected vehicles and the maximal capacity of UAV to connect with vehicles. If the number of connected vehicles increases, the attractive force increases which means if the number of connected vehicles is reaching the maximal capacity of UAV, another UAV will have high opportunity to come close to help. E_i^t and E_{min} represent the current energy level for the UAV and the minimal energy level that allows UAV back to the station. If the energy level is low, the attractive force increases, in order to call another UAV to replace the low-energy level UAV. The $\frac{1}{n}\sum_{k=1}^{n} L_{i,k}^{AVG}$ is the average path loss for all the neighbor UAVs and $d_{i,k}$ is the distance between two UAVs.

$$f_{i,k}^{+} = \frac{\exp\left(\frac{u_i^t}{u_{max}}\right) \cdot E_{min} \cdot \frac{1}{n}\sum_{k=1}^{n} L_{i,k}^{AVG}}{E_t^t \cdot \log\, d_{i,k}}. \tag{12}$$

By using a Seq2Seq-GRU network model, each UAV is capable of predicting the movement trajectory and future position of vehicles within a specific area. This allows the UAVs to gather data and generate a two-dimensional spatial distribution of the vehicles in the area, providing a detailed overview of their positions. Equation (13) represents the attractive force between vehicles and UAVs.

$$f_{i,j}^{+} = \frac{\frac{1}{n}\sum_{j-1}^{n} L_{i,j}^{AVG}}{\log d_{i,j}}. \tag{13}$$

Moreover, considering the safe distance between UAVs, we set ω as the minimum safety distance between two UAVs. We define equation (14) to represent the repulsion force between two UAVs.

$$f_{i,k}^{-} = 10 \cdot \exp\left(-\frac{d_{i,k}}{\omega}\right). \tag{14}$$

Taking into account the effects of various forces, total force $\mathbb{F}$ for u_i can be represented by equation (15). The $\mathbb{F}_{i,x}$ and $\mathbb{F}_{i,y}$ are the vector sum of gravitational and repulsive forces for u_i at a certain time step on x and y dimension respectively. The total force $\mathbb{F}$ can be represented by the vector sum of $\mathbb{F}_{i,x}$ and $\mathbb{F}_{i,y}$, which is $\mathbb{F} = \mathbb{F}_{i,x} + \mathbb{F}_{i,y}$. We set up a time interval Δt between updating the new position for a UAV, in order to prevent the unstable total force due to the small-scale movements of vehicles. If all UAVs are at the $\mathbb{F} = 0$ position at a time step, which indicates it completes the deployment.

In the cross-coverage area, the vehicles calculate the SINR of all the available UAVs, and vehicles select the UAV with the largest SINR to make the connection. Similarly, if one UAV flies close to another UAV to assist. Vehicles in the cross-coverage area calculate the SINA for the new UAN, if it is larger than the original UAV, vehicles will switch the wireless communication channel to the new UAV, otherwise, they continue to connect with the original UAV.

$$\begin{cases} \mathbb{F}_{i,x} = \sum f_{i,k}^{+}(x) + \sum f_{i,j}^{+}(x) + \sum f_{i,k}^{-}(x) \\ \mathbb{F}_{i,y} = \sum f_{i,k}^{+}(y) + \sum f_{i,j}^{+}(y) + \sum f_{i,k}^{-}(y) \end{cases} \tag{15}$$

The proposed VTUDA is shown as algorithm 1. There are three factors that impact the time complexity of the algorithm. The number of UAVs, the number of vehicles that in the coverage area, and the calculation time of the Equation (12), (13), (14). We assume the calculation time is T and the time complexity of the algorithm can be represented by $O(k_u \cdot k_v \cdot T) \approx O(n^2 \cdot T)$. Here, k_u and k_v are the number of UAVs and the number of vehicles that in the coverage area.

V. PERFORMANCE EVALUATION

In this section, we introduce our implementation to validate the efficacy of our proposed algorithm. We first compare the performance of different prediction models, then, we evaluate the coverage rate, energy consumption, and SINR for the proposed algorithm.

A. Prediction Evaluation

In order to evaluate performance of the proposed Seq2Seq- GRU, we leverage a well-known open dataset that is NYC Taxi Trips Dataset [31]. The dataset includes 697,622,444 trips of Taxis with GPS tracking data in New York City. The data is organized into 12 categories, and we use 70% as training data, 15% as testing data, and 15% as validate data. We adopt Python 3.10 and Pytorch framework to implement the Seq2Seq-GRU model.

To evaluate the effectiveness of the Seq2Seq-GRU model for predicting trajectories, we are comparing its performance against several traditional machine learning techniques, including RNN, LSTM, and GRU models that are commonly used in deep learning. We use RNN as a baseline model to establish a benchmark for the Seq2Seq-GRU model's predictive accuracy. In Fig. 4, we can observe the comparison results. The RNN baseline model has a mean square error of 3.531 on the testing dataset. The LSTM model outperforms the RNN model with a mean square error of 1.023, while the GRU model has a slightly lower mean square error of 0.8537. Lastly, we evaluated the proposed Seq2Seq-GRU model, which achieved a testing mean square error of 0.4835, indicating that it outperforms all the other models evaluated in this study.

We utilized a distributed federated learning structure for our proposed Seq2Seq-GRU model and compared the training loss of both centralized and distributed learning structures. Fig. 5 demonstrates the comparative results between centralized and distributed learning. In the figure, the dotted line with circle marks represents the training loss of the dataset using a centralized learning structure, while the solid line with star marks represents the training loss of the dataset using a distributed learning structure. The results indicate that the distributed learning structure has a faster convergence speed than the centralized learning structure. Moreover, the training loss of the distributed learning structure, which is 0.2659, is better than that of the centralized learning structure, which is 0.7678, in our case.

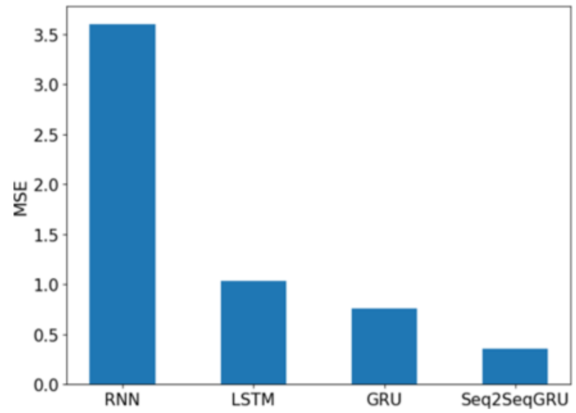

Fig. 4. Training loss comparison between different models

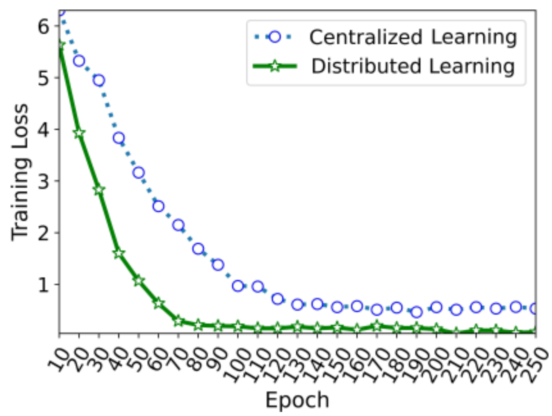

Fig. 5. Training loss comparison between centralized and distributed models

B. VTUDA Performance Evaluation

In this subsection, we detail the evaluation results of the proposed VTUDA. Table I lists the parameter setting of our evaluation.

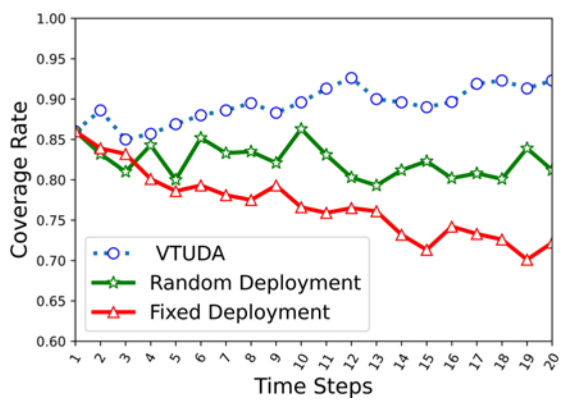

Fig. 6. Coverage rate comparison

Based on the parameter settings, we first evaluate the cover- age rate of the proposed VTUDA algorithm. The coverage rate is determined by calculating the ratio of the total connected vehicles and total vehicles. Therefore, the coverage rate τ is a number less than 1. We compare the coverage rate of the proposed VTUDA, random deployment approach, and fixed deployment approach, the result is shown in Fig. 6. We first randomly deploy UAVs, then, based on the different algorithms, UAVs move to the next position. In the figure, the x-axis shows the time steps and the y-axis shows the coverage rate. The dotted line with circle marks illustrates the coverage rate of the proposed VTUDA, the solid line with star marks represents the random deployment and the solid line with triangle marks represents the fixed deployment. Among the three UAV deployment approaches compared, VTUDA achieves the highest coverage rate. This is attributed to the fact that VTUDA employs accurate prediction results to deploy UAVs, resulting in superior performance compared to the other approaches.

Parameters	**Value**
Number of UAVs	75
The max coverage area of UAV	100 m
The max connection ability of UAV	120
The full battery package energy of UAV	5000 KJ
The energy limit of UAV	500 KJ
The height of UAV	250 m
The sending power of UAV	10 w
Flying energy consumption	200 w
Hovering energy consumption	130 w

Table 1: Parameter Settings

Regarding energy consumption, we also evaluate three different UAV deployment approaches. Fig. 7 shows the comparison result. The fixed deploy approach has the best performance in this evaluation since UAVs only hover in a fixed position. As we discussed before, the energy consumption for hovering is lower than flying. Hence, the fixed deployment consumes lower energy. However, the fixed deploy approach has the worst coverage rate. Compared with the proposed VTUDA and random deployment, the random deployment has a slightly better energy consumption performance. But the coverage rate for random deployment is also worse than VTUDA.

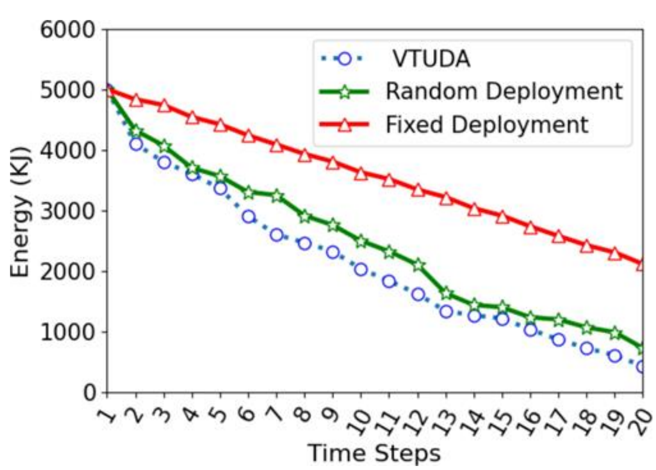

Fig. 7. Energy consumption

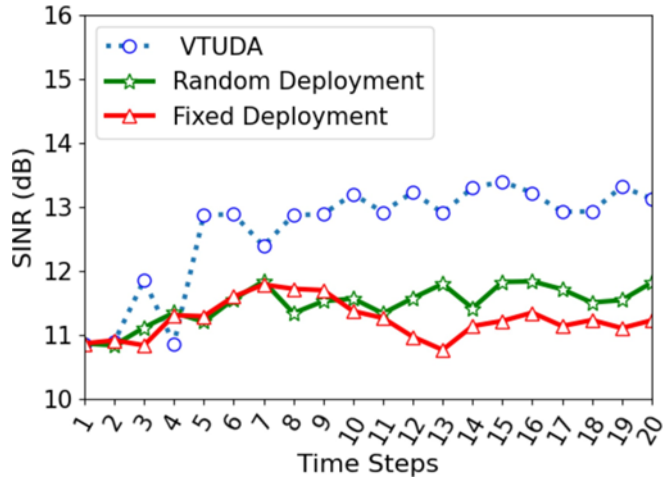

Fig. 8. SINR comparison

Finally, we compare the SINR for three different UAV deployment approaches in Fig. 8. The proposed VTUDA has the highest SINR than the other two deployment approaches which indicates that the proposed VTUDA has the highest probability to connect with vehicles that are in the communication range. In summary, based on the vehicle trajectory prediction result, the proposed VTUDA algorithm has the best coverage rate for moving vehicles compared with random deployment and fixed deployment approaches.

VI. Final Remarks

In this paper, we proposed a VTUDA algorithm to address the connectivity issues between vehicles and RSUs in dynamic traffic scenarios of smart transportation systems. Specifically, we dynamically deploy UAVs as mobile RSUs to provide connection coverage for moving road traffic. We considered the wireless communication channels, energy consumption, and predicted vehicle trajectory, in order to deploy UAVs to optimal positions. Since smart transportation systems are dis- tributed systems, we designed a distributed federated learning model and deployed the model to the RSUs. In order to predict the vehicle trajectory, we adopted the Seq2Seq-GRU model which supports receiving a sequence input and generating a sequence output. Based on the real datasets, the evaluation results showed the proposed VTUDA algorithm can improve the coverage rate and did not increase the energy consumption compare with the random deployment scheme.

VII. References

[1] H. F. Azgomi and M. Jamshidi, “A brief survey on smart community and smart transportation,” in 2018 IEEE 30th international conference on tools with artificial intelligence (ICTAI). IEEE, 2018, pp. 932–939.

[2] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, “A survey on the edge computing for the Internet of things,” IEEE access, vol. 6, pp. 6900–6919, 2018.
[3] F. Liang, W. Yu, D. An, Q. Yang, X. Fu, and W. Zhao, “A survey on big data market: Pricing, trading and protection,” IEEE Access, vol. 6,
pp. 15 132–15 154, 2018.
[4] Y. Toor, P. Muhlethaler, A. Laouiti, and A. De La Fortelle, “Vehicle ad hoc networks: Applications and related technical issues,” IEEE communications surveys & tutorials, vol. 10, no. 3, pp. 74–88, 2008.
[5] A. Giyenko and Y. Im Cho, “Intelligent uav in smart cities using iot,” in 2016 16th International Conference on Control, Automation and Systems (ICCAS). IEEE, 2016, pp. 207–210.
[6] S. Garg, A. Singh, S. Batra, N. Kumar, and L. T. Yang, “Uav- empowered edge computing environment for cyber-threat detection in smart vehicles,” IEEE network, vol. 32, no. 3, pp. 42–51, 2018.
[7] S. Evtiukov, E. Kurakina, and S. Evtiukov, “Smart transport in road transport infrastructure,” in IOP Conference Series: Materials Science and Engineering, vol. 832, no. 1. IOP Publishing, 2020, p. 012094.
[8] F. Liang, F. Yuan, and X. Liu, “User position-based wireless sensor net- work deployment algorithm,” in 2022 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing Commu- nications (GreenCom) and IEEE Cyber, Physical Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), 2022, pp. 70–76.
[9] W. Khawaja, I. Guvenc, D. W. Matolak, U.-C. Fiebig, and N. Schneck- enburger, “A survey of air-to-ground propagation channel modeling for unmanned aerial vehicles,” IEEE Communications Surveys Tutorials, vol. 21, no. 3, pp. 2361–2391, 2019.
[10] A. Guerna, S. Bitam, and C. T. Calafate, “Roadside unit deployment in internet of vehicles systems: A survey,” Sensors, vol. 22, no. 9, p. 3190, 2022.
[11] S. Danba, J. Bao, G. Han, S. Guleng, and C. Wu, “Toward collaborative intelligence in iov systems: recent advances and open issues,” Sensors, vol. 22, no. 18, p. 6995, 2022.
[12] Z. Yu, Y. Gong, S. Gong, and Y. Guo, “Joint task offloading and resource allocation in uav-enabled mobile edge computing,” IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3147–3159, 2020.
[13] A. M. Seid, G. O. Boateng, B. Mareri, G. Sun, and W. Jiang, “Multi- agent drl for task offloading and resource allocation in multi-uav enabled iot edge network,” IEEE Transactions on Network and Service Management, vol. 18, no. 4, pp. 4531–4547, 2021.
[14] A. Islam, A. Debnath, M. Ghose, and S. Chakraborty, “A survey on task offloading in multi-access edge computing,” Journal of Systems Architecture, vol. 118, p. 102225, 2021.
[15] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, “Applications of deep reinforcement learning in communications and networking: A survey,” IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3133–3174, 2019.
[16] T. Alfakih, M. M. Hassan, A. Gumaei, C. Savaglio, and G. Fortino, “Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on sarsa,” IEEE Access, vol. 8, pp. 54 074–54 084, 2020.
[17] T. Li, Y. Xiao, and L. Song, “Integrating future smart home operation platform with demand side management via deep reinforcement learn- ing,” IEEE Transactions on Green Communications and Networking, vol. 5, no. 2, pp. 921–933, 2021.

[18] M. Aazam, S. Zeadally, and E. F. Flushing, “Task offloading in edge computing for machine learning-based smart healthcare,” Computer Networks, vol. 191, p. 108019, 2021.
[19] K. Tan, D. Bremner, J. Le Kernec, L. Zhang, and M. Imran, “Machine learning in vehicular networking: An overview,” Digital Communica- tions and Networks, vol. 8, no. 1, pp. 18–24, 2022.
[20] X. He, H. Lu, M. Du, Y. Mao, and K. Wang, “Qoe-based task offloading with deep reinforcement learning in edge-enabled internet of vehicles,” IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 4,
pp. 2252–2261, 2020.
[21] Z. Ning, P. Dong, X. Wang, M. S. Obaidat, X. Hu, L. Guo, Y. Guo,
J. Huang, B. Hu, and Y. Li, “When deep reinforcement learning meets 5g-enabled vehicular networks: A distributed offloading framework for traffic big data,” IEEE Transactions on Industrial Informatics, vol. 16, no. 2, pp. 1352–1361, 2019.

[22] H.-S. Lee and J.-W. Lee, “Task offloading in heterogeneous mobile cloud computing: Modeling, analysis, and cloudlet deployment,” Ieee Access, vol. 6, pp. 14 908–14 925, 2018.
[23] L.-M. Ang, K. P. Seng, G. K. Ijemaru, and A. M. Zungeru, “Deployment of iov for smart cities: Applications, architecture, and challenges,” IEEE access, vol. 7, pp. 6473–6492, 2018.
[24] X. Xu, Q. Jiang, P. Zhang, X. Cao, M. R. Khosravi, L. T. Alex, L. Qi, and W. Dou, “Game theory for distributed iov task offloading with fuzzy neural network in edge computing,” IEEE Transactions on Fuzzy Systems, vol. 30, no. 11, pp. 4593–4604, 2022.
[25] G. Qiao, S. Leng, K. Zhang, and Y. He, “Collaborative task offloading in vehicular edge multi-access networks,” IEEE Communications Mag- azine, vol. 56, no. 8, pp. 48–54, 2018.
[26] J. Mao, X. Jiang, and X. Zhang, “Analysis of node deployment in wireless sensor networks in warehouse environment monitoring systems,” EURASIP Journal on Wireless Communications and Networking, vol. 2019, no. 1, p. 288, Dec 2019. [Online]. Available: https://doi.org/10.1186/s13638-019-1615-x
[27] J. Akram, H. S. Munawar, A. Z. Kouzani, and M. A. P. Mahmud, “Using adaptive sensors for optimised target coverage in wireless sensor networks,” Sensors, vol. 22, no. 3, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/3/1083
[28] F. Tossa, W. Abdou, K. Ansari, E. C. Ezin, and P. Gouton, “Area coverage maximization under connectivity constraint in wireless sensor networks,” Feb 2022. [Online]. Available: https://www.ncbi.nlm.nih. gov/pmc/articles/PMC8914776/
[29] D. Xue, “Research on range-free location algorithm for wireless sensor network based on particle swarm optimization,” EURASIP Journal on Wireless Communications and Networking, vol. 2019, no. 1, p. 221, Sep 2019. [Online]. Available: https://doi.org/10.1186/s13638-019-1540-z
[30] A. Metiaf and Q. Wu, “Particle swarm optimization based deployment for wsn with the existence of obstacles,” in 2019 5th International Conference on Control, Automation and Robotics (ICCAR), 2019, pp. 614–618.
[31] B. Donovan and D. Work, “New york city taxi data (2010-2013),”
Dataset, http://dx. doi. org/10.13012/J8PN93H8, 2014.
[32] “Vehicle-to-Vehicle Communication.” NHTSA, www.nhtsa.gov/technology-innovation/vehicle-vehicle-communication. Accessed 7 Aug. 2022.

INSTITUTE FOR HOMELAND SECURITY

Sam Houston State University

The Institute for Homeland Security at Sam Houston State University is focused on building strategic partnerships between public and private organizations through education and applied research ventures in the critical infrastructure sectors of Transportation, Energy, Chemical, Healthcare, and Public Health.
The Institute is a center for strategic thought with the goal of contributing to the security, resilience, and business continuity of these sectors from a Texas Homeland Security perspective. This is accomplished by facilitating collaboration activities, offering education programs, and conducting research to enhance the skills of practitioners specific to natural and human caused Homeland Security events.

Institute for Homeland Security
Sam Houston State University

Liang, F., Liu, X., Kose, N. A., Gundogan, K., Yu, W. (2023) Towards Trajectory Prediction-based UAV Deployment in Smart Transportation Systems. (Report No. IHS/CR-2023-1010). The Sam Houston State University Institute for Homeland Security. https://doi.org/10.17605/OSF.IO/GYFDX