This article has been accepted for publication in IEEE Transactions on Vehicular Technology. 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/TVT.2022.3209539

Towards Predictive Forwarding Strategy in Vehicular Named Data Networking

Junxia Wang, *Student Member, IEEE,* Jiangtao Luo*, *Senior Member, IEEE,* Yongyi Ran*, *Member, IEEE,* Junchao Yang, Kai Liu, *Member, IEEE,* and Song Guo, *Fellow, IEEE*

***Abstract*—Vehicular Named Data Networking (V-NDN) is promising to improve the content delivery efficiency in Vehicular Ad Hoc Networks (VANETs). However, the potential broadcast storm caused by Interest packet flooding and return path failures caused by vehicle mobility can significantly degrade the content delivery performance. Existing forwarding strategies based on outdated position information cannot address these issues well. In this paper, we propose a novel *predictive forwarding strategy* (PRFS) for V-NDN. In PRFS, Long Short-Term Memory (LSTM) is employed to amend the NeighBor Table (NBT) for preciser neighboring vehicles' positions. In addition, the next-hop forwarder is selected among the neighbors, taking into account the link reliability and the Distance along the Road (DR) in both directions. Furthermore, a new mechanism is designed to notify the selected next-hop forwarder by embedding the forwarder identity in the Interest packet header, so as to accelerate the forwarding process. Finally, extensive simulations are carried out, and experimental results demonstrate that PRFS can reduce the number of forwarded Interest packets and data packets by 21.29% and 25.75%, respectively, and improve the success ratio of satisfied Interest packets by 35.1% compared to the existing baseline algorithms.**

***Index Terms*—Vehicular ad hoc network, named data networking, data forwarding, LSTM**

I. INTRODUCTION

IN the past decade, vehicular ad hoc networks (VANETs) have gained increasing attention in academia and industry for enabling emerging Intelligent Transportation Systems (ITSs) to improve traffic efficiency, driving safety, and the comfort of passengers [1]. Typical application scenarios include map, safety information, commercial advertisement, and video, etc [2]. Nevertheless, traditional host-centric addressing and connection-oriented content transmission require end-to-end connections to be established prior to data transmission [3], which is less efficient in the case of highly dynamic topologies in VANETs, with short duration and interruption-prone links [4, 5]. Therefore, the communication paradigm based on end-to-end connectivity is not proper for content delivery in VANETs, and new architecture and mechanisms are called for [6].

Vehicular Named Data Networking (V-NDN), which integrates Named Data Networking (NDN) with VANETs, was proposed to tackle these issues [7]. NDN features with name-based addressing and ubiquitous in-network caching. In the NDN framework, each data packet has a unique name. When a consumer requests a specific content (e.g., *map, safety information, video*), an Interest packet with the content name rather than a host address will be sent to the network, trying to pull back the corresponding data packet. The request may be fulfilled by either the original server (called *Producer*) or intermediate routers with cached duplicates. This name- and pull-based paradigm [8, 9] does not need a long-lived connection and can effectively bypass the problems of intermittent link connectivity and content provider mobility. In addition, NDN utilizes a stateful forwarding mechanism. Each NDN node maintains a *Pending Interest Table* (PIT), *Forwarding Information Base* (FIB) and *Content Store* (CS) [10]. CS plays the role of in-network caching as mentioned, while PIT records the status of all incoming unsatisfied Interests. When receiving an Interest packet, the NDN node first looks up its CS to find whether it has the requested content or not. If yes, a duplicate data packet will be returned to the consumer immediately, thus reducing the content delivery delay and cost [11]. Otherwise, the node will check the PIT to find whether identical Interests have been received. If yes, it will discard this Interest and update the corresponding PIT entry by appending the incoming interface information; otherwise, it will forward this Interest according to a specific strategy contained in FIB. As a result, the hop-by-hop forwarding of Interest packets creates a path, named return path, recorded by PIT entries [12]. Finally, the requested content (if found) will be sent back to the consumer along the return path [13].

Everything has two sides. Despite the above benefits it brings, V-NDN still has two issues to address. The first is broadcast storm due to the Interest packet flooding [14]. The most straightforward forwarding strategy of V-NDN is broadcast; That is, each vehicle continues to broadcast the Interest packets it has received. This approach is simple, fast, and can increase the success ratio of forwarding. Nevertheless, wireless signal interference will significantly increase due to the avalanche effect, resulting in collision blocking. The

This work is jointly supported by National Natural Science Foundation of China (No. 62171072, 62172064, 62003067) (Corresponding authors: Jiangtao Luo and Yongyi Ran.)

J. Wang is with School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: D170101010@stu.cqupt.edu.cn).

J. Luo and Y. Ran are with Electronic Information and Networking Research Institute, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: Luojt@cqupt.edu.cn, ranyy@cqupt.edu.cn).

J. Yang is with School of Artificial Intelligence, Chongqing Technology and Business University, Chongqing 400067, China (e-mail: yangjc@ctbu.edu.cn).

K. Liu is with the Key Laboratory of Dependable Service Computing in Cyber Physical Society (Chongqing University), Ministry of Education, and also with the College of Computer Science, Chongqing University, Chongqing 400030, China (e-mail: liukai0807@gmail.com).

Song Guo is with the Department of Computing, and Research Institute for Sustainable Urban Development, Hong Kong Polytechnic University, Hong Kong (e-mail: song.guo@polyu.edu.hk).

second is that the return path may fail before the packet is successfully received due to the high mobility of vehicles when selecting one optimal path. The entries in the PIT for building the return path may be outdated because one of the in-path vehicles moves out of the coverage of its previous-hop vehicle. Therefore, a good forwarding strategy needs to take into account both. To begin with, efficient forwarding allows Interest packets to reach providers as soon as possible. Then, as much as possible, ensure that the return path remains valid.

Many efforts have been devoted to designing forwarding strategies for addressing the above issues. In order to alleviate the broadcast storm, location-aware forwarding strategies were designed to determine the path of Interest packets based on the location of content providers [15–17]. However, these approaches are generally only suitable for applications with a specific location. Later, distance-aware forwarding approaches were proposed to select the vehicle farthest from the current forwarder as the next-hop in order to make Interest packets reach the content provider faster [18, 19]. However, a further next-hop will lead to a shorter link duration as well as a more vulnerable return path. To avoid that, link-stability-based forwarding methods were presented in [20, 21] to choose the next-hop forwarders with stable links, which involved more hops resulting in inefficient content delivery. Furthermore, NeighBor Table (NBT)-based forwarding approaches were proposed by comprehensively considering the link duration and distance [22, 23], which outperformed the previous methods because of achieving a better tradeoff between efficiency and reliability. However, such NBT-based approaches select the next-hop forwarder according to the mobility information stored in NBT, which may become outdated and eventually degrade content delivery performance.

In this paper, we extend our previous work in [24] and propose *predictive forwarding strategy* (PRFS). To begin with, it estimates the current positions of all neighbors based on the NBT when making forwarding decisions; in other words, a predicted NBT is generated. After comparing with Kalman filtering, we select LSTM as the prediction method. Furthermore, the optimal forwarder is selected from the neighbors considering both Link Expired Time (LET) and Distance along Road (DR) based on the predicted NBT. Specifically, we select the neighbors whose LET are greater than a predefined threshold as candidate next-hop nodes. To further improve the efficiency and reliability, we filter out the optimal next-hop node with the farthest DR in both road directions respectively for transferring Interest packets while creating the return path. In addition, the native Interest packet format is modified to record the information of next-hop nodes and prevent indiscriminately flooding.

The main contributions of this paper are summarized as follows.

- We revise the V-NDN system architecture by adding a prediction module to address the outdated NBT and improve the efficiency and success ratio of content delivery.
- We present a predictive forwarding strategy for V-NDN based on a predicted NBT, which adopts LSTM to estimate the current positions of all neighbors.
- We propose a hybrid forwarder selection method combining the LET and DR based on the predicted NBT, which determines the optimal next-hop node with the farthest DR and LET more than the predefined threshold.
- PRFS is implemented in ndnSIM, and extensive simulations are carried out. Experiment results demonstrate that our proposed method can achieve a better performance than the existing baseline algorithms in terms of the number of Interest packets and data packets, the success ratio of Interest packets, and transmission delay.

The remainder of this paper is structured as follows. Section II reviews the related work. Section III describes the system architecture and problem statement. Section IV presents the proposed predictive forwarding strategy in detail. The experiment results are illustrated and analyzed in Section V. Finally, section VI concludes this paper.

II. Related Work

In order to prevent the broadcast storm of Interest packets and guarantee the return path being valid during the communication period, many research efforts have been devoted to designing forwarding strategies for V-NDN. According to the forwarding mechanism, we classify the existing methods into five categories: flooding-based forwarding, location-aware forwarding, link-stability-based forwarding, distance-aware forwarding, and NBT-based forwarding strategies.

1) Flooding-Based Forwarding: The original V-NDN adopted a flooding strategy, where each node would flood every received Interest packet, resulting in Interest and data packet broadcast storm. Especially, because the size of a data packet is much larger than that of an Interest packet, data packet broadcast storm degrades the overall network performance more severely than Interest packet broadcast storm. Hence, Ahmed *et al.* designed the CODIE scheme to limit the forwarding hops of the data packet by adding a hop counter in the Interest packet, and a DDL field in the data packet [25]. Nevertheless, this method only relieves the data packet broadcast storm outside the area bordered by the content consumer and provider, while the broadcast storm still exists in the area.

2) Location-Aware Forwarding: The location-aware forwarding strategies were designed to determine the path of Interest packets based on the location of content providers [15–17]. In [15], Bian *et al.* made the Interest reach the position of content provider indicated in the data name by presenting a geo-based forwarding strategy. In this work, the vehicle in the intersection or farthest was selected to forward Interest packets preferentially. A similar approach in [16] named GeoZone got the position of content provider according to a geo-referenced naming scheme and limited the data transmission zone with source position. In order to adapt to different applications, Deng *et al.* proposed HVNDN, which first divided the request types into location-related and location-independent, and then adopted opportunistic forwarding strategy and probabilistic forwarding strategy, respectively. Unlike getting the content provider location from the data name, LoICen prioritized the forwarding of the Interest packet by vehicles closer to the destination by opportunistically getting the locations of content providers [17]. But such methods require prior knowledge of the location of content providers and are generally only suitable for applications with a specific location.

3) Distance-Aware Forwarding: The distance-aware forwarding approaches selected the farthest vehicle as the next-hop forwarder [18, 19]. Yu *et al.* proposed OIFP to make vehicles farther away from the current sender preferentially forward packets by a defer timer method [18]. In order to further improve the efficiency, Rondon *et al.* considered the road characteristics and prioritized the vehicle farther from the current sender in a hot area as the next-hop forwarder [19]. Although these strategies can make Interest packets reach the content producer fast, a farther next-hop regularly leads to a shorter link duration and a more vulnerable return path.

4) Link-Stability-Based Forwarding: In order to prevent the return path from becoming invalid before completing the data packet transmission, the link-stability-based forwarding strategies chose the next-hop forwarders with long link durations [20, 21]. Boukerche *et al.* implemented LISIC, which employed a defer transmission timer to preferentially make the vehicles with longer link expired time forward Interest packets [20]. Similarly, Sousa *et al.* proposed LSIF, where only the vehicles with predicted link durations greater than a predefined threshold were regarded as next-hop forwarders [21]. Although these forwarding policies alleviate the return path failure, they are inefficient because more hops are involved.

5) NBT-Based Forwarding: The NBT-based forwarding strategies selected the optimal next-hop forwarder by comprehensively considering multiple attributes, according to NBT [22, 23]. Ahmed *et al.* proposed RUFS to select only one relay node at each hop according to its NBT [22]. However, the optimal next-hop forwarder and the content producer may be located in different road directions of the consumer in VANETs. To address this problem, Ahmed *et al.* presented DIFS by disseminating Interest packets in both road directions of the consumer respectively to find content providers [23]. Because of the limitation of the wireless communication mechanism, the NBT cannot be updated frequently and may become outdated in a short period due to the high mobility of vehicles. Although a better tradeoff between efficiency and reliability is balanced than the previous methods, the outdated NBT usually decreases content delivery performance.

III. System Architecture and Problem Statement

A. System Architecture

Fig. 1 shows the system architecture in V-NDN. The following assumptions are made.

- Each vehicle is equipped with either a Global Positioning System (GPS) or Beidou device to obtain its real-time mobility information (i.e., location, speed, and direction) [26].
- Each vehicle communicates with other vehicles via a wireless interface: Dedicated Short Range Communication (DSRC) (e.g., IEEE 802.11p) or C-V2X [27]. The maximum communication range of each vehicle is R.

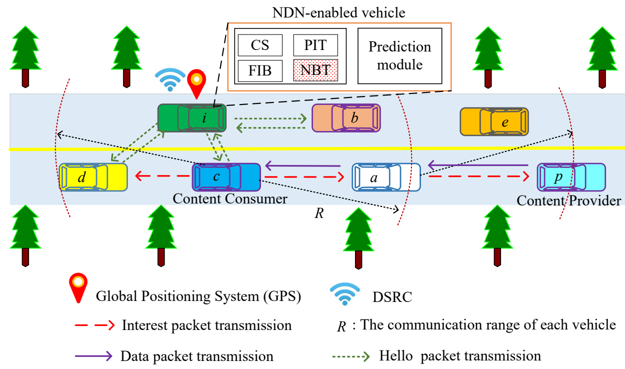


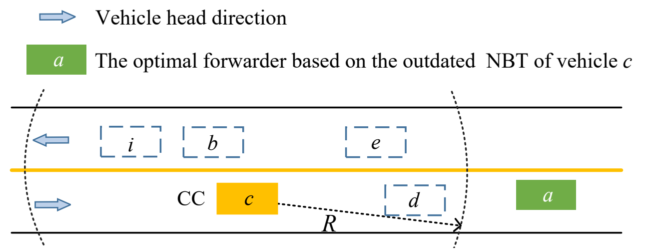
Fig. 1. Illustration of system architecture in V-NDN.

- Each vehicle can obtain the road directions of a straight or turning road segment by a digital map.
- Each vehicle adopts NDN protocol and maintains a NBT and a LSTM prediction module in addition to the original PIT, FIB, and CS. Three types of packets: Interest, data, and hello packets, are exchanged in the network.
- NBT is created to record the neighbors' mobility information, maintained by periodically exchanging hello packets. The period time is denoted by T. The time for exchanging hello packets is expressed by its period index, $t = 1, 2, \cdots, j-1, j, j+1, \cdots$. The position of vehicle i at $t = j$ is denoted by (x_{ij}, y_{ij}). Accordingly, its velocities in the direction of x and y are denoted by v_{ij}^{x} and v_{ij}^{y}, respectively. The consumer c receives the hello packet containing the mobility information at $t = j$ from vehicle i, and updates its NBT entry as $\mathrm{NBT}_c^j(i) = (x_{ij}, y_{ij}, v_{ij}^x, v_{ij}^y, j)$. $\mathrm{NBT}_c(i)$ is not updated until vehicle c receives next hello packet of vehicle i. The greater the interval between the time vehicle c needs to make a forwarding decision and the time it last updated its NBT, the greater the probability of the vehicle c's NBT becoming outdated. Hence, a prediction module is employed to estimate the mobility information in NBT when planning a path for each Interest packet.
- A vehicle can play any of the following three roles: content consumer (CC), content provider (CP), and data forwarder. Due to each vehicle's limited communication range, multi-hop forwarding is usually required to help CC retrieve content from the CP beyond its range. The CC requests content (e.g., *map of Beijing, China*) by sending an Interest packet with the name (e.g., */map/cn/beijing/*) and pulls back the corresponding data packet along the return path. How NDN nodes process Interest and data packets is described in detail in the second paragraph of Section I (Introduction).

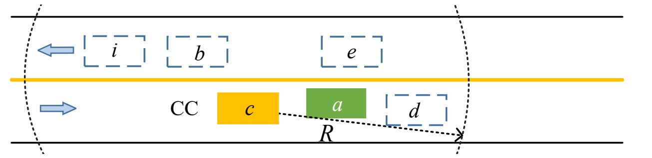
B. Problem Statement

Our objective is to work out a novel forwarding strategy that selects an optimal path for Interest packets based on predicted neighbors' positions by considering the duration and DR of wireless links.

Outdated NBT in general NBT-based strategies may lead to content delivery failure or inefficiency, and two possible



(a) Scene 1: Content delivery fails, where vehicle a is beyond the reach of c.



(b) Scene 2: Content delivery is inefficient, where vehicle a is no longer the optimal forwarder.

Fig. 2. Two scenarios in which the outdated NBT leads to content delivery failure or inefficiency in VANETs.

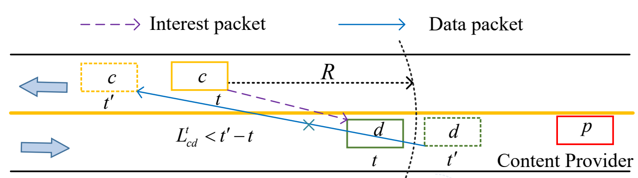


Fig. 3. The selected forwarder (vehicle d) is beyond the reach of previous-hop vehicle (vehicle c) when the content returns after a period of time $t' - t$.

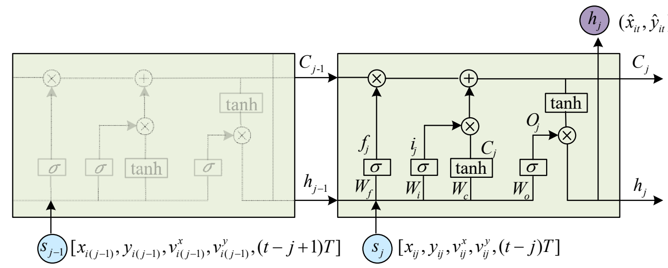


Fig. 4. LSTM is employed to predict the position of neighboring vehicle i at time t based on NBT.

scenarios are shown in Fig. 2. We assume that the CP is beyond the communication range of the CC (vehicle c). So the Interest packet sent by vehicle c requires multi-hop forwarding to reach CP. Moreover, the first key issue is that vehicle c selects the next-hop forwarder from its neighbors. We suppose that Fig. 1 shows the positions of the vehicles in the NBT of vehicle c. And vehicle c regards vehicle a as the next-hop forwarder, according to its NBT. The real positions of vehicles at that time are depicted in Fig. 2. Owing to the outdated NBT, there is a difference between the mobility information stored in the NBT and the real-time information, which affects the data transmission performance. In the case shown in Fig. 2 (a), the regarded forwarder (vehicle a) is outside the vehicle c's communication range, so content delivery fails. As illustrated in Fig. 2 (b) that the considered forwarder is not the optimal one, so content delivery is inefficient, because more hop counts (usually mean more delays) are required to transmit the Interest to the CP. Hence, to prevent the outdated NBT from degrading the performance of the forwarding strategy, vehicle c should know as much as possible about the exact current locations of all its neighbors when selecting the optimal next-hop forwarder. Two past mobility information of vehicle i at $t = j-1, t = j$ (namely, $\mathrm{NBT}_c^{j-1}(i)$, $\mathrm{NBT}_c^{j}(i)$) are employed to predict the position of vehicle i at time t ($j < t < j+1$), referred to as (x_{it},y_{it}). Generally, the position prediction is related to the time difference. Therefore, the prediction objective is to use the two mobility information $[x_{ij}, y_{ij}, v^x_{ij}, v^y_{ij}, (t-j)T]$, $[x_{i(j-1)}, y_{i(j-1)}, v^x_{i(j-1)}, v^y_{i(j-1)}, (t-j+1)T]$ to predict the position of the vehicle $(\hat{x}_{it}, \hat{y}_{it})$. Therefore, we first use the LSTM algorithm to predict the position of the vehicles in the NBT and then select the next-hop forwarder based on the predicted NBT. Thus, the strategy can prevent frequent exchanges of hello packets and ensure the accuracy of the vehicle location in predicted NBT, thereby improving network performance.

Then, a good forwarding strategy should ensure that the return path remains valid and Interest packets are spread to the content provider as soon as possible. Vehicle c is in charge of selecting the optimal next-hop forwarders according to the predicted NBT. On one hand, the selected forwarder may be out of the previous-hop vehicle when the content returns after a period of time, as shown in Fig. 3. LET can be estimated based on the relative position, speed, driving direction of two neighboring vehicles, and the wireless communication range [21]. The LET between vehicle c and d at time t is represented as L^t_{cd}. And $L^t_{cd} < t' - t$. So when the data packet arrives at d at time t', the return path between c and d becomes invalid. Hence, we select the neighbors whose LET are greater than a predefined threshold μ (e.g., $\mu > t' - t$) as candidate next-hop nodes to prevent the return path from becoming invalid. μ is determined by the historical experience of the maximum data transmission delay. On the other hand, we furthermore filter out the optimal next-hop node with the farthest DR in both road directions respectively for transferring Interest packets while creating the return path.

IV. Proposed Predictive Forwarding Strategy

This section presents the predictive forwarding strategy and explains it in detail. The objective of the proposed PRFS is to plan an optimal path for each Interest packet to improve content delivery efficiency and reliability in VANETs. It includes mobility prediction by using LSTM, packet header extensions, LET and DR computation, and the forwarding process of the Interest packet and data packet.

A. The Mobility Prediction Based on LSTM

To prevent the outdated NBT from degrading network performance, it is necessary to accurately and timely predict the mobility information of vehicles in the NBT [28]. As a special kind of Recurrent Neural Network (RNN), LSTM has been proven to perform well in training sequence data and solve the vanishing and exploding gradient problem [29]. Therefore, we leverage the LSTM method to predict the positions of the neighboring vehicles at the moment t ($j < t < j+1$) by using their mobility state at the past two moments (i.e., $j-1$, j), as shown in Fig. 4. The current input layer s_j, the previous input layer s_{j-1}, and the corresponding output value h_j are represented by (1), respectively,

$$\begin{cases} s_{j-1} = [x_{i(j-1)}, y_{i(j-1)}, v^x_{i(j-1)}, v^y_{i(j-1)}, (t-j+1)T] \\ s_j \ = [x_{ij}, y_{ij}, v^x_{ij}, v^y_{ij}, (t-j)T] \\ h_j \ = (\hat{x}_{it}, \hat{y}_{it}) \end{cases}, \tag{1}$$

where the subscript $j-1$ and j represent consecutive time index numbers.

A single LSTM cell consists of a cell state (C_j) and three gates: forget gate (f_j), input gate (g_j), and output gate (O_j) [30]. The cell state controlled by these three gates stores historical information. Gate is a notable feature of the LSTM method, which can selectively decide which information to pass. The forget gate, input gate, and output gate are introduced as follows.

First, the forget gate (f_j) controls what information to pass from the previous state (C_{j-1}) and is computed by (2),

$$f_j = \sigma\left(W_f \cdot [h_{j-1}, s_j] + b_f\right), \tag{2}$$

where $\sigma(\cdot)$ is the logistic *sigmoid* function, and the output is between 0 and 1. 1 and 0 represent previous cell state C_{j-1} completely retained or discarded, respectively. $[h_{j-1}, s_j]$ represents a longer vector obtained by concatenating two vectors, such as previous output h_{j-1} and current input s_j. W_f and b_f indicate the weight matrix and the offset vector, respectively. These two parameters above (i.e., W_f, b_f), the following three weight matrices (i.e., W_g, W_C, W_o), and three offset vectors (i.e., b_g, b_C, b_o) in (3), (4), and (6) are the objects to train in LSTM.

Next, the input gate (g_j) and input node decide what new information to be stored in the cell state (C_j). The input gate (g_j) is a *sigmoid* layer to determine the ratio of input update. The input node generates a new candidate vector ($\tilde{C}_j$), which is added to the cell state. The above two outputs are calculated by (3) and (4), respectively,

$$g_j = \sigma\left(W_g \cdot [h_{j-1}, s_j] + b_g\right), \tag{3}$$

$$\tilde{C}_j = \tanh\left(W_C \cdot [h_{j-1}, s_j] + b_C\right). \tag{4}$$

By combining (2), (3), and (4), the current cell state is updated by (5).

$$C_j = f_j \times C_{j-1} + g_j \times \tilde{C}_j \tag{5}$$

Finally, the output gate (O_j) determines what information to output from the cell state, which is calculated by (6). The output value (h_j) is the cell state processed by the *tanh* function multiplied by the output gate and is computed by (7).

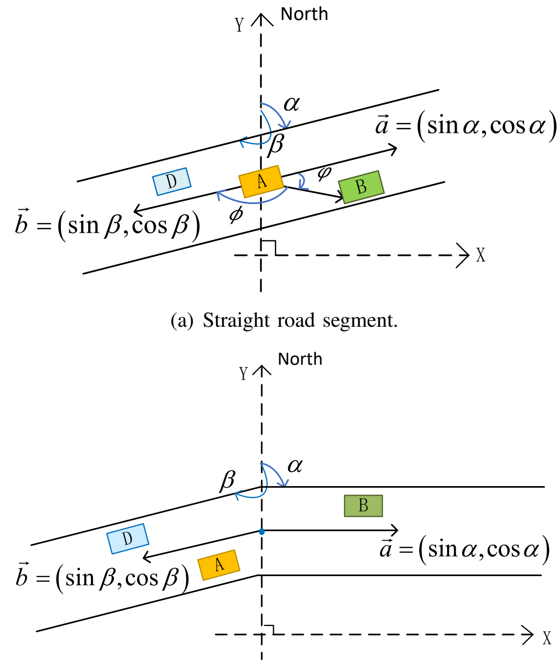
$$O_j = \sigma\left(W_o \cdot [h_{j-1}, s_j] + b_o\right) \tag{6}$$

$$h_j = O_j \times \tanh\left(C_j\right) \tag{7}$$

Original field | Additional field

Content Name	Nonce	Selectors	RRD	FIRRD	RD	FIRD

Fig. 5. Modified Interest packet format.



(a) Straight road segment.

(b) Turning road segment.

Fig. 6. The RD and RRD of road segment in two cases.

B. Packet Header Extensions

Four fields, including `RRD` (Reverse Road Direction), `FIRRD` (Forwarder ID in Reverse Road Direction), `RD` (Road Direction), and `FIRD` (Forwarder ID in Road Direction) are appended to the original fields of the Interest packet (e.g., content name, nonce), as shown in Fig. 5. The following two reasons explain this.

1) `FIRD` and `FIRRD` specify the next-hop forwarder in road direction and reverse road direction, respectively. When an Interest packet is received, only the specified nodes forward it further, while other nodes discard it, thus reducing the number of forwarded packets and alleviating broadcast storm.
2) `RD` and `RRD` are employed to specify the direction to select the next-hop forwarder. Among them, `RD` is defined as the relatively small angle elapsed from the northward clockwise rotation to the road direction, denoted by α, whereas `RRD` is defined as the relatively big angle, denoted by β.

Two cases for `RD` and `RRD` are shown in Fig. 6. In Fig. 6(a), the road segment is straight and obviously, $\beta = \alpha + 180°$. In Fig. 6(b), there is a turn in the road segment and then $\beta = \alpha + \gamma$, where γ is the turning angle of the road. `RD` and `RRD` can also be represented by vectors $\vec{a}$ and $\vec{b}$, and computed by

$$\begin{cases} \vec{a} = (\sin\alpha, \cos\alpha) \\ \vec{b} = (\sin\beta, \cos\beta). \end{cases} \tag{8}$$

Next, we introduce how to determine whether a node is located in the RD or RRD of another node, in the case that two nodes A and B are located on the straight road segment (i.e., in Fig. 6(a)). Their positions are denoted by (x_A, y_A) and (x_B, y_B), respectively. So vector $\overrightarrow{AB}$ is represented by

$$\overrightarrow{AB} = (x_B - x_A, y_B - y_A). \tag{9}$$

Compute $\cos\varphi$ according to (10), where φ is the angle between $\overrightarrow{AB}$ and $\vec{a}$; if $\cos\varphi \geq 0$, i.e., $(x_B - x_A)\cdot\sin\alpha + (y_B - y_A)\cdot\cos\alpha \geq 0$, then we can tell that B is located in the RD of A; Otherwise, B is said to be in the RRD of A.

$$\cos(\varphi) = \frac{\vec{a}\cdot\overrightarrow{AB}}{\|\vec{a}\|\left\|\overrightarrow{AB}\right\|} \tag{10}$$

C. LET and DR computation

To establish a reliable path to get the requested content faster, we select relay nodes based on two properties between the current forwarder and its neighboring nodes, namely LET and DR. They are calculated based on the predicted NBT rather than the original NBT when making the decision. According to [31], the LET between vehicle i and k at time t, L_{ik}^t, is calculated by

$$L_{ik}^t = \frac{-(ab+cd)+\sqrt{(a^2+c^2)R^2-(ad-bc)^2}}{a^2+c^2}, \tag{11}$$

where $a = v_{it}^x - v_{kt}^x, b = x_{it} - x_{kt}, c = v_{it}^y - v_{kt}^y, d = y_{it} - y_{kt}$. As depicted in Fig. 7(a), if node i and k are located at the same straight road segment, in which the RD is α, the DR between these two nodes, D_{ik}^t, is computed by

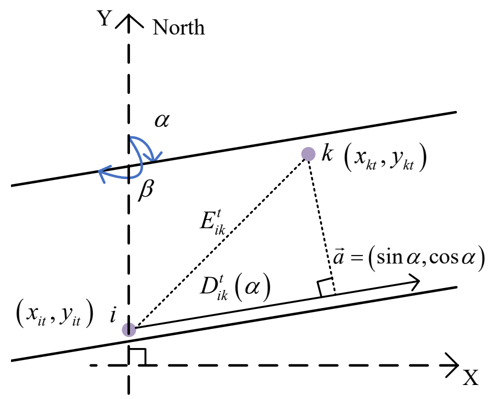
$$D_{ik}^t(\alpha) = E_{ik}^t \cdot \cos\left(\alpha - \arctan\left|\frac{x_{kt} - x_{it}}{y_{kt} - y_{it}}\right|\right), \tag{12}$$

where E_{ik}^t is the Euclidean distance between vehicle i and k at time t, $E_{ik}^t = \sqrt{(x_{it} - x_{kt})^2 + (y_{it} - y_{kt})^2}$. In the case that node i and k are located at different road segments of a turning road, depicted in Fig. 7(b), DR can still be computed by (12).

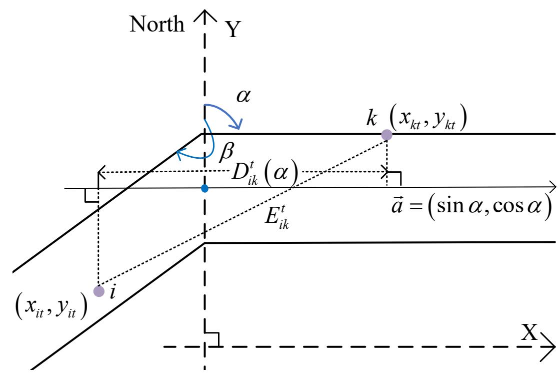
Fig. 8 is to explain why we select the vehicle with the largest DR instead of the Euclidean distance as a relay node. Two paths, $c \to a \to b \to d \to e$ and $c \to f \to g \to h \to i$, are created by considering the largest DR and Euclidean distance, respectively. It is observed that the former can reach vehicle father. Therefore, DR instead of Euclidean distance is regarded as an attribute, when selecting the next-hop.

D. Proposed Forwarding Strategy

Our proposed scheme PRFS takes into account both LET and DR when selecting the next-hop relay node. Specifically, the node with longer LET and the farthest DR away from the current forwarding node is preferred to be the relay node for next-hop. First, a longer LET means a larger probability that the reverse path will still be present when the Data packet is sent back. Second, a larger DR from the current forwarder means fewer hop counts along the road to the content provider, meaning less response delay (Fig. 8). More importantly, LSTM



(a) Two nodes located at the same road segment.



(b) Two nodes located at different road segments.

Fig. 7. Diagram of the DR between two nodes.

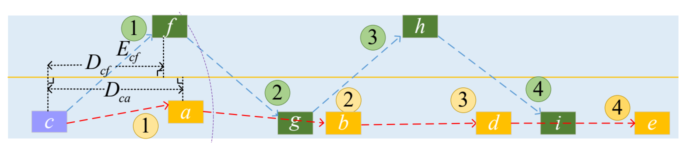


Fig. 8. Different paths selected based on the largest DR and Euclidean distance.

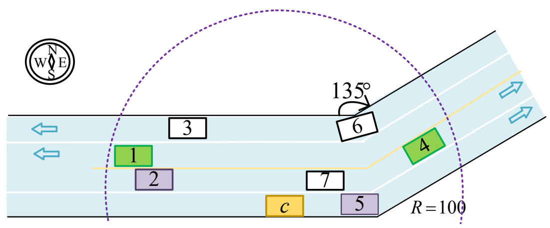
is introduced to estimate the exact positions of all neighboring vehicles during the interval between two hello packets, forming a predicted NBT to avoid the information in the NBT being obsoleted due to the rapid movements of vehicles. After that, DR and LET are calculated based on the predicted NBT.

Next, the procedure of PRFS will be presented in detail. Since a consumer will send Interest packets in opposite directions on both sides, which is different from other intermediate forwarders, we describe the procedure in the following two parts.

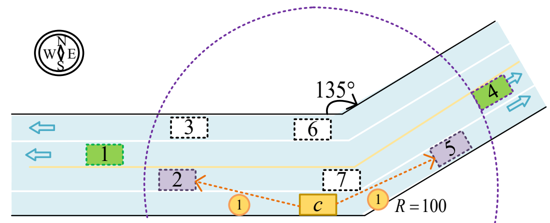
1) Forwarding procedure in a consumer: When requesting content at time t, the consumer needs to select one relay node in each direction among its neighboring nodes, as shown in Fig. 9. The steps are in detail described as follows.

Step 1. Predict the NBT. The consumer, denoted by c, uses LSTM to predict the positions in its NBT at time t $(j < t < j+1)$, based on the location and velocity information of its neighbors at the past two time points (snapshots), NBT_c^{j-1} and NBT_c^j, creating the predicted NBT_c^t.

Step 2. Group neighbors by direction. Get the road direction



(a) The positions of vehicles in the NBT of consumer (vehicle c), where vehicle 1 and 4 are regarded as the best relay nodes.



(b) The actual positions of vehicles in the NBT of consumer (vehicle c), where vehicle 2 and 5 are the best relay nodes.

ID	X	Y	V^x	V^y	t	X	Y	V^x	V^y	t
c	950	0	7	0	5	962	0	5	0	7
1	890	8	-8	0	5	877	8	-10	0	7
2	873	4	4	0	5	882	4	6	0	7
3	943	12	-12	0	5	922	12	-10	0	7
4	1025	29	10	10	5	1045	49	11	11	7
5	1000	0	9	9	5	1015	15	8	8	7
6	1010	22	-5	-5	5	1000	12	-7	0	7
7	956	8	12	0	5	979	8	11	0	7

1)

ID	X	Y	V^x	V^y	t
c	973	0	5	0	9
✗1	850	8	-10	0	9
2	895	4	6	0	9
3	898	12	-10	0	9
✗4	1068	72	11	11	9
5	1030	30	8	8	9
6	986	12	-7	0	9
7	999	8	11	0	9

2)

3)

$DL_{RRD}(c)$

ID	LET	DR
2	177.9	78
3	8.3	25

$DL_{RD}(c)$

ID	LET	DR
5	5	61.5
6	9.4	13
7	12.3	26

(c) An example illustrating forwarding procedure in a consumer.

1) Consumer predicts the positions of vehicles in its NBT using LSTM.
2) Divide the vehicles in its NBT except for vehicles beyond the reach (e.g., vehicle 1 and 4) to $\mathrm{DL_{RRD}}(c)$ or $\mathrm{DL_{RD}}(c)$ according to (13) and (14), respectively.
3) Select the vehicle 2 and 5 as `FIRRD` and `FIRD`, respectively.

Fig. 9. Consumer selects one relay node in each direction among its neighboring nodes, when sending an Interest packet.

α and β by looking up the digital map, and then group all neighbors into two sets based on the predicted NBT_c^t and according to (13) and (14). Specifically, all vehicles satisfying (13) are put into the road direction decision list $\mathrm{DL_{RD}}(c)$, while all vehicles satisfying (14) are placed into the reverse road direction decision list $\mathrm{DL_{RRD}}(c)$. Refer to the analysis in Section IV-B for the basis for such grouping.

$$\begin{aligned} &\mathrm{DL_{RD}}(c) = \{i \in \mathrm{NBT}_c^t\} \\ &s.t. \quad \begin{cases} (x_{it} - x_{ct}) \cdot \sin\alpha + (y_{it} - y_{ct}) \cdot \cos\alpha \geq 0 \\ d_{ic}^t \leq R \end{cases} \end{aligned} \tag{13}$$

$$\begin{aligned} &\mathrm{DL_{RRD}}(c) = \{i \in \mathrm{NBT}_c^t\} \\ &s.t. \quad \begin{cases} (x_{it} - x_{ct}) \cdot \sin\beta + (y_{it} - y_{ct}) \cdot \cos\beta \geq 0 \\ d_{ic}^t \leq R \end{cases} \end{aligned} \tag{14}$$

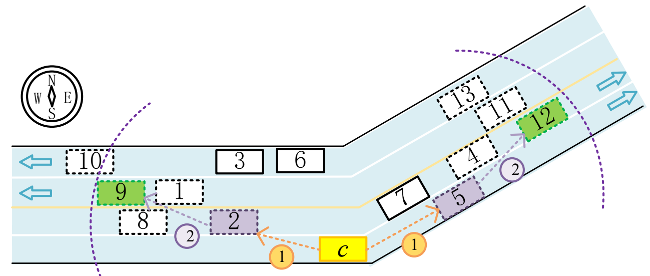


Fig. 10. Diagram of the specified relay node selects a next-hop relay node away from the consumer. i.e., vehicle 5 (vehicle 2) selects vehicle 12 (vehicle 9) as `FIRD` (`FIRRD`).

Step 3. Calculate LETs and DRs. According to (11) and (12), compute the LET and DR for each neighbouring vehicle in the two decision lists $\mathrm{DL_{RD}}(c)$ and $\mathrm{DL_{RRD}}(c)$. For vehicle i, the LET between itself and the consumer c at time t is denoted by L_{ci}^t; DR is denoted by $D_{ci}^t(\alpha)$ or $D_{ci}^t(\beta)$, depending on which decision list vehicle i lies in. It is worth noting that there may be two RDs or two RRDs for consumers close to the inflection point of the road, depending on the actual location of the neighboring vehicle.

Step 4. Select the first pair of forwarders. The consumer needs to select the first-hop two forwarders on both sides, each for a decision list. Given a threshold for LET, denoted by μ, if there are vehicles with LETs larger than the threshold, we select the one with the largest DR among them; otherwise, we select the one with the largest LET. The policy is described by (15) and (16). So far, the first pair of forwarders are selected. The `FIRD` chosen from $\mathrm{DL_{RD}}(c)$ is denoted by ξ, while the `FIRRD` chosen from $\mathrm{DL_{RRD}}(c)$ is denoted by ψ.

$$\xi = \begin{cases} \underset{i \in \mathrm{DL_{RD}}(c)}{\arg\max} \left[D_{ci}^t(\alpha) \,\middle|\, L_{ci}^t > \mu \right], \ \exists\, L_{ci}^t > \mu \\ \underset{i \in \mathrm{DL_{RD}}(c)}{\arg\max} \ L_{ci}^t, \ \nexists\, L_{ci}^t > \mu \end{cases} \tag{15}$$

$$\psi = \begin{cases} \underset{i \in \mathrm{DL_{RRD}}(c)}{\arg\max} \left[D_{ci}^t(\beta) \,\middle|\, L_{ci}^t > \mu \right], \ \exists\, L_{ci}^t > \mu \\ \underset{i \in \mathrm{DL_{RRD}}(c)}{\arg\max} \ L_{ci}^t, \ \nexists\, L_{ci}^t > \mu \end{cases} \tag{16}$$

Step 5. Transmit the Interest packet. The consumer transmits the Interest packet and designates the next-hop forwarders, ψ and ξ, on certain road directions, β and α, with the extended packet header fields, as illustrated in Fig. 5.

2) Forwarding procedure in relay nodes: When a vehicle receives an Interest packet from its neighbor, on one hand, it processes the packet as an ordinary NDN node; at the same time, it has to check whether it has been assigned as a forwarder. The procedure is described in Algorithm 1.

Specifically, the node first checks whether there exists an entry with the same name in CS or not. If yes, it generates a corresponding data packet and transmits it out immediately; otherwise, it checks the PIT whether there has been a request with the same name before. If yes, it drops the Interest packet; otherwise, it needs to check whether it is the chosen one, by comparing its identity (namely r) with the `FIRRD` and `FIRD`

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. 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/TVT.2022.3209539

Algorithm 1 The forwarding procedure in a relay node

Input: Current vehicle r receives Interest packet [Name, Selector(s), Nonce, β, ψ, α, ξ] at time t.

if CS does not has the content with Name **then**
  if PIT does not has a request with Name **then**
    if $r = \xi$ **then**
      Add [Name, Nonce, Face] in PIT.
      Predict its NBT at time t using the LSTM algorithm.
      Considering `RD` α, put the vehicles satisfying (13) into $\mathrm{DL}_{\mathrm{RD}}\,(r)$.
      Compute LET L_{ri}^{t} and DR $D_{ri}^{t}(\alpha)$ between vehicle r and all the vehicles i in $\mathrm{DL}_{\mathrm{RD}}\,(r)$, according to (11) and (12).
      ξ' is selected as `FIRD` using (15).
      Broadcast the Interest packet [Name, Selector(s), Nonce, $\alpha + 180°$, $\varnothing$, α, ξ'].
    else if $r = \psi$ **then**
      Add [Name, Nonce, Face] in PIT.
      Predict its NBT at time t using the LSTM algorithm.
      Considering `RRD` β, place the vehicles fulfilling (14) into $\mathrm{DL}_{\mathrm{RRD}}\,(r)$.
      Calculate LET L_{ri}^{t} and DR $D_{ri}^{t}(\beta)$ between vehicle r and all the vehicles i in $\mathrm{DL}_{\mathrm{RRD}}\,(r)$, according to (11) and (12).
      ψ' is selected as `FIRRD` using (16).
      Broadcast the Interest packet [Name, Selector(s), Nonce, β, ψ', $\beta - 180°$, $\varnothing$].
    else
      Discard Interest packet.
    end if
  else
    Discard Interest packet.
  end if
else
  Return data packet.
end if

carried in the Interest packet header. Only if it is the chosen one, i.e., $r = \xi$ or $r = \psi$, it is allowed to forward the packet, and it is called a *Forwarder*.

The forwarder is obligated to select a next-hop forwarder before it transmits the packet out. It first updates the PIT entry and then leverages LSTM to evaluate its neighboring vehicles' current positions based on its NBT, as did the consumer. The difference is the forwarder only needs to select its successor in one direction, the same direction as it is relative to its predecessor. That is, if this forwarder is a `FIRD`, it will search its successor in its $\mathrm{DL}_{\mathrm{RD}}$; and vice versa. The policy of selecting the forwarder in the decision list is done in the same way as the consumer did, following (15) or (16). The procedure is exemplified in Fig. 10, showing the dissemination of Interest packets with the consumer at the center. In this example, vehicle 5 selects vehicle 12 as the next `FIRD`; in the other direction, vehicle 2 selects vehicle 9 as the next `FIRRD`.

Now then, this forwarder will transmit the Interest packet after updating its header fields: `RRD` and `FIRRD`, or the `RD` and `FIRD`, bearing the identity of the new chosen one along the same direction.

TABLE I
THE PARAMETERS IN LSTM

Parameters	values
Number of Input Hidden Layer	5
Time step	2
Hidden size	64
Number of hidden layer unit	2
Learning rate	0.003
Optimizer	Adam
Loss Type	MSE

V. PERFORMANCE EVALUATION

A. Data Set and Methodology

We consider vehicular network topology in a highway scenario composed of two-way four-lane and a model-driven trace generated by SUMO [32]. Random trips are generated, where the maximum speed is 32 m/s, and the maximum acceleration is $4.5\ \mathrm{m/s^2}$. The data preprocessing process mainly consists of two parts: using the above mobility trace to generate data set in the style of (1), and then normalizing it. Afterward, LSTM is empolyed to train the normalization data set, generating the trained well model. Finally, the trained model is utilized for location prediction of NBT.

The number of samples in data set for test is N. The n-th sample's real-time location is represented by $(x_{\langle n\rangle}, y_{\langle n\rangle})$. Meanwhile, $(\hat{x}_{\langle n\rangle}, \hat{y}_{\langle n\rangle})$ denotes the n-th sample's location predicted by implementing LSTM in Python programming languages. The following performance metrics are introduced to evaluate prediction accuracy between the real-time and prediction positions of the vehicle. Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 are calculated by (17), (18), and (19), respectively.

$$\mathrm{MAE} = \frac{1}{2N}\sum_{n=1}^{N}\left(\left|x_{\langle n\rangle} - \hat{x}_{\langle n\rangle}\right| + \left|y_{\langle n\rangle} - \hat{y}_{\langle n\rangle}\right|\right) \tag{17}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{2N}\sum_{n=1}^{N}\left[\left(x_{\langle n\rangle} - \hat{x}_{\langle n\rangle}\right)^2 + \left(y_{\langle n\rangle} - \hat{y}_{\langle n\rangle}\right)^2\right]} \tag{18}$$

$$\begin{aligned}\mathrm{R}^2 &= 1 - \frac{\sum_{n=1}^{N}\left[\left(\hat{x}_{\langle n\rangle} - x_{\langle n\rangle}\right)^2 + \left(\hat{y}_{\langle n\rangle} - y_{\langle n\rangle}\right)^2\right]}{\sum_{n=1}^{N}\left[\left(\bar{x}_{\langle n\rangle} - x_{\langle n\rangle}\right)^2 + \left(\bar{y}_{\langle n\rangle} - y_{\langle n\rangle}\right)^2\right]} \\ &= 1 - \frac{\mathrm{MSE}\left[\left(x_{\langle n\rangle}, y_{\langle n\rangle}\right), \left(\hat{x}_{\langle n\rangle}, \hat{y}_{\langle n\rangle}\right)\right]}{\mathrm{Var}\left(x_{\langle n\rangle}, y_{\langle n\rangle}\right)}\end{aligned} \tag{19}$$

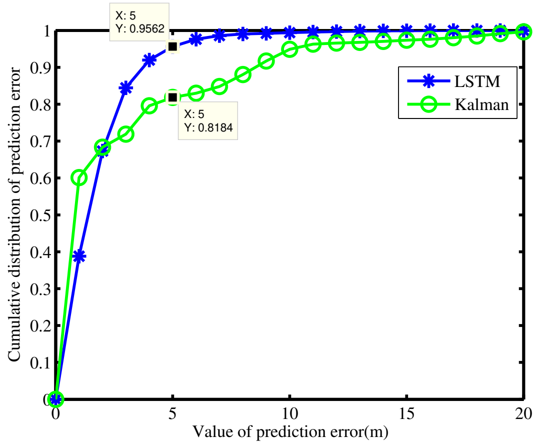
In addition, the predicted location error of the n-th sample is denoted by (20).

$$D_{\mathrm{error}}^{\langle n\rangle} = \sqrt{\left(x_{\langle n\rangle} - \hat{x}_{\langle n\rangle}\right)^2 + \left(y_{\langle n\rangle} - \hat{y}_{\langle n\rangle}\right)^2} \tag{20}$$

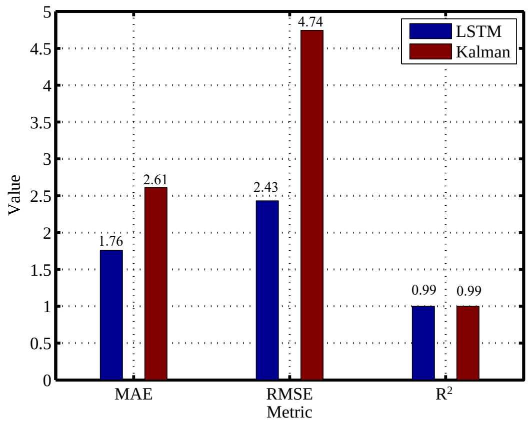
The cumulative distribution of prediction error is defined as,

$$F\left(D_{\mathrm{error}}\right) = \frac{\sum_{n=1}^{N}\mathbb{1}\left(D_{\mathrm{error}}^{\langle n\rangle} \le D_{\mathrm{error}}\right)}{N}, \tag{21}$$

where $\mathbb{1}(\cdot)$ is the indicator function.



(a) The cumulative distribution of prediction error.



(b) The value of each metric corresponding to the prediction result.

Fig. 11. The comparison of prediction results using LSTM and Kalman model.

The parameters in LSTM are shown in Table I. The cumulative distribution of prediction error using LSTM and Kalman [28] is shown in Fig. 11, where the exchange period of hello packets is 2 s. It can be inferred from Fig. 11 that LSTM has a better prediction performance. Compared with the Kalman model, the LSTM has 16.84% more vehicles with a distance error of less than 5 m. According to Fig. 11(b), the LSTM has less 32.6% MAE, less 48.76% RMSE, and similar R-squared, compared with the Kalman model. Therefore, LSTM is adopted to predict the positions of the vehicles in NBT. To achieve this, ndnSIM first uses the socket to reports the NBT to Pycharm (Python) software, which calls the trained LSTM model, and then collects the corresponding predicted NBT, when a vehicle selects the next-hop.

B. Experimental Setup and Metrics

We implement PRFS, DIFS [23], and flooding strategy in ndnSIM, which is NS-3 based NDN simulator [33]. In addition to the traditional data structures in NDN, we add a NBT, which records the mobility information of neighbors. For the sake of fairness, when comparing forwarding strategies, we use the same mobility traces and network parameters, as

TABLE II
SIMULATION PARAMETERS

Parameters	values
Simulation scene	Highway with 10 km
Number of vehicles	50, 100, 150, 200 and 250
Number of Interest packet requested	200, 400, 600 and 800
Maximum speed	32 m/s
Maximum acceleration	4.5 m/s^2
payload size of data packet	1 kb
Max transmission range	500 m
Transmission power	40 mW
Bit rate	6 Mbps
Carrier frequency	5.9 GHz
Communication technology	IEEE 802.11p
Additional data structures	NBT
Simulation time	20 s
Simulation times	10

shown in Table II. We consider a highway with 10 km, with different numbers of vehicles (e.g., 50, 100, 150, 200, 260). The communication range of each node is 500 meters. The communication technology adopts the IEEE 802.11p protocol, and the transmission rate is 6 Mbps. Like a real scenario, we randomly select content consumers and producers. Moreover, without knowing the producer's location, the consumers request content by sending Interest packets randomly. The Interest re-transmission time for consumers is set to 3 seconds. The period of exchanging hello packets for each vehicle is 2 s. The simulation result is an average of 10 runs.

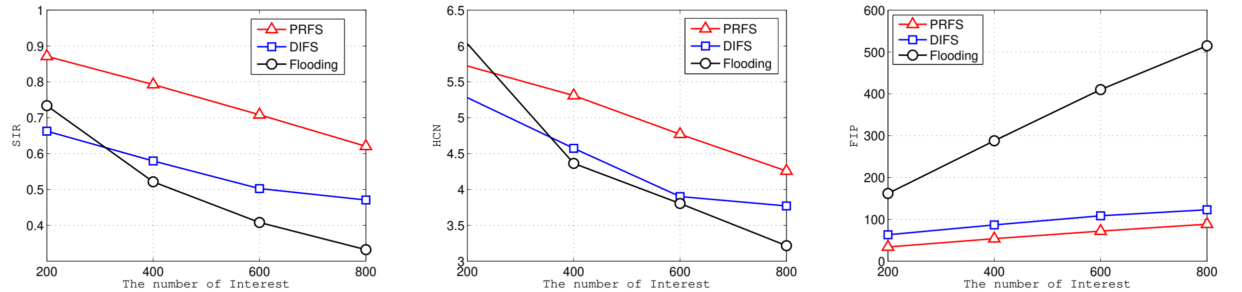
The performance metrics for the forwarding strategy are shown as follows.

- SIR: The ratio of satisfied numbers of Interest packets to the total ones sent by consumers.
- HCN: The average hop counts for content retrieval (last re-transmitted hop count if needed).
- FIP: The average copies of forwarded Interest packets per vehicle, including hello packets if required.
- FDP: The average copies of forwarded data packets per vehicle.
- ISD: The average Interest satisfaction delay, from the first time the consumer sends an Interest packet to the receipt of the corresponding data packet, including re-transmission delay.

C. Results Analysis

1) Performance Comparison under Different Network Traffic: During simulations, when the total number of vehicles in the network is 200, by changing the number of requests, the performance on SIR, ISD, HCN, FIP, FDP of PRFS, DIFS, and flooding strategy is evaluated, as shown in Fig. 12. With the increase in network traffic, SIR and HCN of these three strategies decrease while FIP and FDP rise. This is because more packets in the network inevitably increase wireless transmission collision, network congestion, and delay, making it impossible to retrieve content from further away content providers.

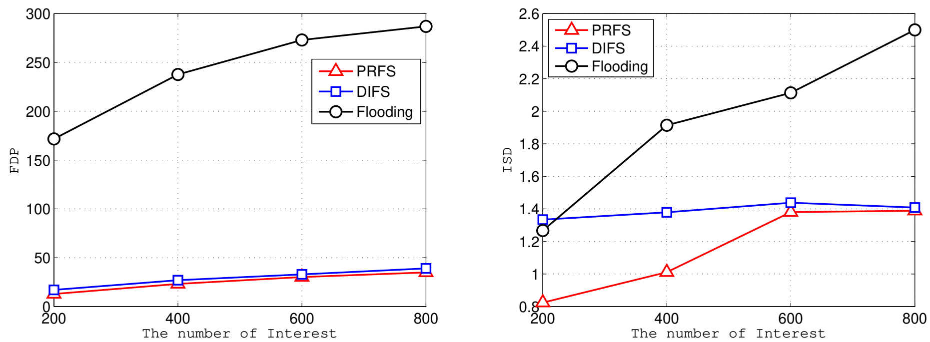
The SIR and HCN of the flooding strategy decrease the most, while the FIP and FDP increase the fastest. Meanwhile, PRFS has more 49.99% SIR, less 40.92% ISR, less 81.93%



(a) The satisfaction ratio of Interest packets (SIR).

(b) Average hop counts for content retrieval (HCN).

(c) Average number of forwarded Interest packets per vehicle (FIP).



(d) Average number of forwarded data packets per vehicle (FDP).

(e) Average Interest satisfaction delay (ISD).

Fig. 12. Content delivery comparison based on network traffic.

FIP, and less 89.55% FDP on average, compared to the flooding strategy, respectively. It is evident that broadcast storm caused by flooding seriously affects network performance, especially in heavy network traffic. In addition, PRFS selects the next-hop vehicle according to the predicted NBT by the LSTM method, while DIFS chooses the next-hop based on outdated NBT. Accordingly, in comparison to the DIFS, 35.1% more SIR, 17.18% less ISD, 14.46% more HCN, 34.83% less FIP, and 12.8% less FDP are experienced by PRFS, respectively. Therefore, the proposed PRFS can achieve better performance with less overhead than DIFS and flooding strategy, whatever the network traffic is.

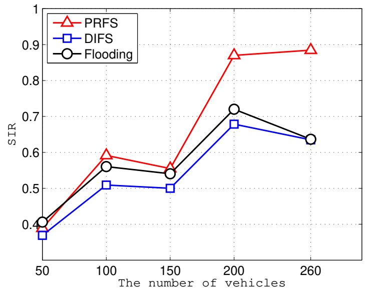
2) Performance Comparison with Different Node Densities: During simulations, by changing the number of nodes, the performance on SIR, ISD, HCN, FIP, and FDP of PRFS, DIFS, and flooding strategy with different node densities is evaluated at low network traffic, where the total number of requests is 200, as shown in Fig.13.

According to Fig.13 (a) and (b), with the growth in node densities, SIR and HCN of these three strategies increase. When the node density is low (i.e., the number of vehicles is 50, 100, and 150), there may be either no link or a short link duration between the current forwarder and its neighboring nodes, so content further from the consumer can not be retrieved, leading to low SIR and HCN. As the density of nodes increases (i.e., the number of vehicles is 200 and 260), more suitable relay nodes can be selected so that data packets can be pulled back to the consumer from farther content providers, thus gaining higher SIR and HCN.

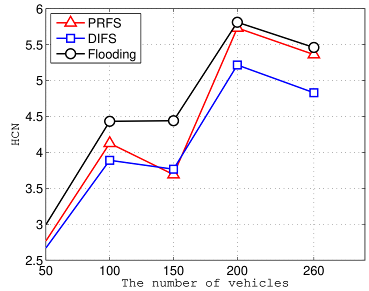
It can be seen from Fig.13 (c) and (d) that the FIP of PRFS is lower than that of Flooding and DIFS by 65.0% and 21.29%, respectively. The FDP of PRFS is lower than that of Flooding and DIFS by 90.6% and 25.72%, respectively. These results show that PRFS has the lowest cost regardless of the node density. For the flooding strategy, every node will forward each received Interest packet, resulting in many redundant Interest packets in the network. For the DIFS strategy, only if the node regarding itself as the most suitable relay node based on its NBT forwards it, there may be no node or multiple nodes for a hop to forward the Interest packet due to the unreliability of the wireless channel. PRFS allows consumers to choose one node in each direction, while the relay node determines only one suitable node at each hop, so its FIP and FDP are less than that of DIFS.

Compared with the flooding strategy, when the node density is low, PRFS has a smaller overhead, fewer hops, similar SIR, and similar delay. This is due to few relay nodes to choose from per hop, resulting in little difference in performance. As node density increases, PRFS performs better than flooding, because more appropriate nodes can be selected. Meanwhile, the implementation of the flooding strategy becomes worse due to the broadcast storm. Consequently, as node density increases, PRFS has a similar number of hop counts, higher SIR by 29.34% with less delay by 41.18%. Compared with the DIFS strategy, when the node density is low, PRFS has similar hops, higher SIR, and similar delay. Compared with the DIFS strategy, PRFS has a higher HCN by 10.42%, higher SIR by 33.57%, and less delay by 31.26%, when the node density

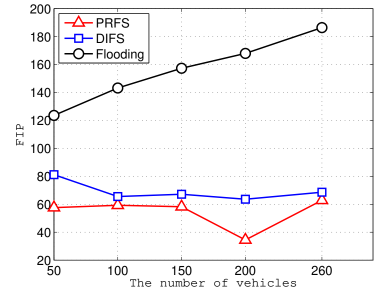
This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2022.3209539



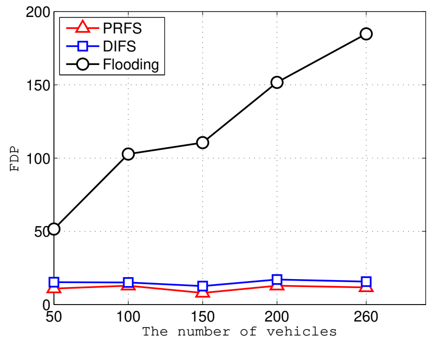
(a) Ratio of satisfied Interest packets (SIR).



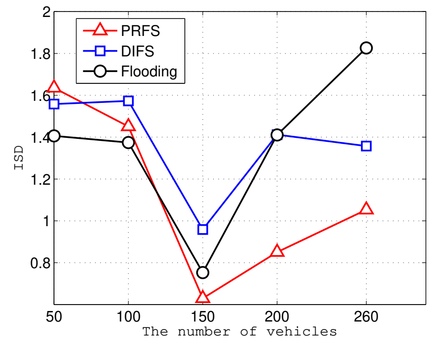
(b) Average hop counts for content retrieval (HCN).



(c) Average number of forwarded Interest packets per vehicle (FIP).



(d) Average number of forwarded data packets per vehicle (FDP).



(e) Average Interest satisfaction delay (ISD).

Fig. 13. Content delivery comparison based on node density.

is high. This phenomenon proves that PRFS has chosen a more suitable relay node for efficient and reliable data delivery because of solving the outdated NBT.

VI. Conclusion

In this paper, we propose a predictive forwarding strategy (PRFS) in V-NDN, in which the consumer or each intermediate forwarder leverages LSTM to estimate the immediate positions of neighbors, forming a predicted neighbor table (NBT), based on which to select the next-hop forwarder for Interest packets. In particular, the selection of the next hop forwarder takes into account both the link expired time (LET) and the distance along the road (DR), calculated from the predicted NBT. Using DR instead of Euclidean distance allows the packets to travel faster on the road and is suitable for turning roads. The simulation results show that the proposed scheme outperforms the existing forwarding strategies based on outdated NBT. It alleviates the broadcast storm significantly and reduces the risk of return link failure at the same time, thus greatly improving the efficiency of content delivery.

In future work, we will further enhance this scheme by considering reliable mechanisms for Data packet delivery and investigate collaborations between VANET and C-V2X.

References

[1] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive survey on machine learning in vehicular network: Technology, applications and challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 2027–2057, Third Quarter 2021.

[2] J. Wang, K. Liu, K. Xiao, X. Wang, Q. Han, and V. C. S. Lee, "Delay-constrained routing via heterogeneous vehicular communications in software defined busnet," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5957–5970, Jun. 2019.

[3] F. Tang, Y. Kawamoto, N. Kato, and J. Liu, "Future intelligent and secure vehicular network toward 6G: Machine-learning approaches," *Proc. IEEE*, vol. 108, no. 2, pp. 292–307, Feb. 2020.

[4] H. Khelifi, S. Luo, B. Nour, H. Moungla, Y. Faheem, R. Hussain, and A. Ksentini, "Named data networking in vehicular ad hoc networks: State-of-the-art and challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 320–351, Jan. 2019.

[5] K. Liu, K. Xiao, P. Dai, V. C. Lee, S. Guo, and J. Cao, "Fog computing empowered data dissemination in software defined heterogeneous VANETs," *IEEE Trans. Mobile Comput.*, vol. 20, no. 11, pp. 3181–3193, Nov. 2021.

[6] X. Wang, Z. Wang, and S. Cai, "Data delivery in vehicular named data networking," *IEEE Netw. Lett.*, vol. 2, no. 3, pp. 120–123, Sep. 2020.

[7] G. Grassi, D. Pesavento, G. Pau, R. Vuyyuru, R. Wakikawa, and L. Zhang, "VANET via named data networking," in *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. Toronto, ON, Canada: IEEE, May 2014, pp. 410–415.

[8] S. H. Bouk, S. H. Ahmed, M. A. Yaqub, D. Kim, and M. Gerla, "DPEL: Dynamic PIT entry lifetime in vehicular named data networks," *IEEE Commun. Lett.*, vol. 20, no. 2, pp. 336–339, Feb. 2015.

[9] S. H. Bouk, S. H. Ahmed, D. Kim, K.-J. Park, Y. Eun, and J. Lloret, "LAPEL: Hop limit based adaptive PIT entry lifetime for vehicular named data networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5546–5557, Jul. 2018.

[10] A. Tariq, R. A. Rehman, and B.-S. Kim, "Forwarding strategies in NDN-based wireless networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 1, pp. 68–95, Jan. 2020.

[11] X. Wang, X. Wang, and D. Wang, "Cost-efficient data retrieval based on integration of VC and NDN," *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 967–976, Jan. 2021.
[12] F. Wu, W. Yang, J. Ren, F. Lyu, P. Yang, Y. Zhang, and X. Shen, "Named data networking enabled power saving mode design for WLAN," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 901–913, Jan. 2020.
[13] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, Mar. 2017.
[14] R. Hou, S. Zhou, M. Cui, L. Zhou, D. Zeng, J. Luo, and M. Ma, "Data forwarding scheme for vehicle tracking in named data networking," *IEEE Trans. Veh. Technol.*, vol. 70, no. 7, pp. 6684–6695, Jul. 2021.
[15] C. Bian, T. Zhao, X. Li, and W. Yan, "Boosting named data networking for data dissemination in urban VANET scenarios," *Veh. Commun.*, vol. 2, no. 4, pp. 195–207, Apr. 2015.
[16] A. A. Prates, I. V. Bastos, and I. M. Moraes, "GeoZone: An interest-packet forwarding mechanism based on dissemination zone for content-centric vehicular networks," *Comput. Electr. Eng.*, vol. 73, pp. 155–166, Jan. 2019.
[17] A. Boukerche and R. W. Coutinho, "LoICen: A novel location-based and information-centric architecture for content distribution in vehicular networks," *Ad Hoc Netw.*, vol. 93, pp. 1–10, Oct. 2019.
[18] X. Yu, R. W. Coutinho, A. Boukerche, and A. A. Loureiro, "A distance-based interest forwarding protocol for vehicular information-centric networks," in *Proc. IEEE 28th PIMRC*. Montreal, QC, Canada: IEEE, 2017, pp. 1–5.
[19] L. B. Rondon, J. B. da Costa, G. P. Rocha Filho, and L. A. Villas, "A distance and position-based caching discovery protocol for vehicular named-data networks," in *Proc. IEEE LATINCOM*. Salvador, Brazil: IEEE, 2019, pp. 1–6.
[20] A. M. de Sousa, F. R. Araujo, and L. N. Sampaio, "A link-stability-based interest-forwarding strategy for vehicular named data networks," *IEEE Internet Comput.*, vol. 22, no. 3, pp. 16–26, Jun. 2018.
[21] A. Boukerche, R. W. Coutinho, and X. Yu, "LISIC: A link stability-based protocol for vehicular information-centric networks," in *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. Orlando, FL, USA: IEEE, 2017, pp. 233–240.
[22] S. H. Ahmed, S. H. Bouk, and D. Kim, "RUFS: Robust forwarder selection in vehicular content-centric networks," *IEEE Commun. Lett.*, vol. 19, no. 9, pp. 1616–1619, Sep. 2015.
[23] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, and H. Song, "DIFS: Distributed interest forwarder selection in vehicular named data networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 3076–3080, Sep. 2017.
[24] J. Wang, J. Luo, J. Zhou, and Y. Ran, "A mobility-predict-based forwarding strategy in vehicular named data networks," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. Taipei, Taiwan: IEEE, Dec. 2020, pp. 01–06.
[25] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, H. Song, and J. Lloret, "CODIE: Controlled data and interest evaluation in vehicular named data networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3954–3963, Jun. 2016.
[26] K. Liu, L. Feng, P. Dai, V. C. S. Lee, S. H. Son, and J. Cao, "Coding-assisted broadcast scheduling via memetic computing in SDN-based vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2420–2431, Aug. 2018.
[27] R. Lu, L. Zhang, J. Ni, and Y. Fang, "5G vehicle-to-everything services: Gearing up for security and privacy," *Proc. IEEE*, vol. 108, no. 2, pp. 373–389, Feb. 2020.
[28] C. Liu, G. Zhang, W. Guo, and R. He, "Kalman prediction-based neighbor discovery and its effect on routing protocol in vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 159–169, Jan. 2020.
[29] D. T. Le and G. Kaddoum, "LSTM-based channel access scheme for vehicles in cognitive vehicular networks with multi-agent settings," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9132–9143, Sep. 2021.
[30] A. Ip, L. Irio, and R. Oliveira, "Vehicle trajectory prediction based on LSTM recurrent neural networks," in *Proc. 93rd IEEE VTC2021-Spring*. Helsinki, Finland: IEEE, 2021, pp. 1–5.
[31] S. H. Bouk, I. Sasase, S. H. Ahmed, and N. Javaid, "Gateway discovery algorithm based on multiple QoS path parameters between mobile node and gateway node," *J. Commun. Netw.-S. Kor.*, vol. 14, no. 4, pp. 434–442, Aug. 2012.
[32] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO-simulation of urban mobility," *Int. J. Adv. Syst. Meas.*, vol. 5, no. 3&4, Dec. 2012.
[33] S. Mastorakis, A. Afanasyev, and L. Zhang, "On the evolution of ndnSIM: An open-source simulator for NDN experimentation," *SIGCOMM Comput. Commun. Rev.*, vol. 47, no. 3, pp. 19—-33, Jul. 2017.

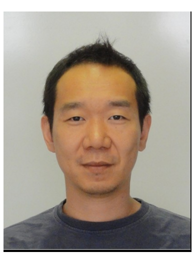
Junxia Wang received the B.S. and M.S. degrees in Henan University of Technology, Zhengzhou, and Chongqing University of Posts and Telecommunications, Chongqing, in 2014 and 2017, respectively. She is currently pursuing Ph.D. degree in Chongqing University of Posts and Telecommunications (CQUPT), Chongqing, China. Her research interests include named data networking, vehicular networks, and blind recognition of linear block code.

Jiangtao Luo (M'11–SM'15) received his B.S. and PhD degrees from Nankai University and the Chinese Academy of Science in 1993 and 1998, respectively. Currently, he is a full professor, PhD supervisor and deputy Dean of the Electronic Information and Networking Research Institute at Chongqing University of Posts and Telecommunications (CQUPT). He has been visiting scholars at the University of Hamburg, Germany and the University of Southern California, USA during 2015 and 2016. His major research interests are network data analysis, visual big data and future Internet architecture. He has published more than 150 papers and owned 30 patents in these fields. He was awarded the Chinese State Award of Scientific and Technological Progress in 2011, the Chongqing Provincial Award of Scientific and Technological Progress twice in 2010 and 2007, respectively, and the Chongqing Science and Technology Award for Youth in 2010. He has been selected as IEEE Senior Member since June 2015.

Yongyi Ran (Member, IEEE) received the B.S. degree and Ph.D. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2008 and 2014, respectively. He is currently a Lecturer at Chongqing University of Posts and Telecommunications, China. He worked as a Research Fellow in the Cloud Computing and Application Platform (CAP) Group at Nanyang Technological University from 2019 to 2020 and worked as a Postdoctoral Research Associate at University of Massachusetts Lowell from 2017 to 2018. His research interests include deep reinforcement learning, green data center, network measurement, and intelligent satellite networks.

Junchao Yang received the B.S. degree in Electronic and Information Engineering and the M.S. and Ph.D. degree in Communication and Information Engineering from Chongqing University of Posts and Telecommunications, Chongqing, in 2012, 2015 and 2019, separately. In 2018, he was a Visiting Scholar with University of Washington, Seattle, USA. He is currently a Lecturer with School of artificial intelligence, Chongqing Technology and Business University, Chongqing, China. His current research interests are in multimedia networking, VR video streaming and convex optimization.

Kai Liu (S'07-M'12-SM'19) received his Ph.D. Degree in Computer Science from the City University of Hong Kong in 2011. From December 2010 to May 2011, he was a Visiting Scholar with the Department of Computer Science, University of Virginia, USA. From 2011 to 2014, he was a Postdoctoral Fellow with Singapore Nanyang Technological University, City University of Hong Kong, and Hong Kong Baptist University. He is currently a Professor with the College of Computer Science, Chongqing University, China. His research interests include Internet of Vehicles, Mobile Computing and Pervasive Computing.



Song Guo is a Full Professor at Department of Computing, The Hong Kong Polytechnic University. He also holds a Changjiang Chair Professorship awarded by the Ministry of Education of China. Prof. Guo is a Fellow of the Canadian Academy of Engineering and a Fellow of the IEEE (Computer Society). His research interests are mainly in big data, edge AI, mobile computing, and distributed systems. He published many papers in top venues with wide impact in these areas and was recognized as a Highly Cited Researcher (Clarivate Web of Science). He is the recipient of over a dozen Best Paper Awards from IEEE/ACM conferences, journals, and technical committees. Prof. Guo is the Editor-in-Chief of IEEE Open Journal of the Computer Society and the Chair of IEEE Communications Society (ComSoc) Space and Satellite Communications Technical Committee. He was an IEEE ComSoc Distinguished Lecturer and a member of IEEE ComSoc Board of Governors. He has served for IEEE Computer Society on Fellow Evaluation Committee, and been named on editorial board of a number of prestigious international journals like IEEE TPDS, IEEE TCC, IEEE TETC, etc. He has also served as chairs of organizing and technical committees of many international conferences.