A Fog Computing Paradigm for Efficient Information Services in VANET

Ke Xiao*, Kai Liu*, Junhua Wang*, Yanning Yang*, Liang Feng*, Jingjing Cao†, Victor Lee‡
*Department of Computer Science, Chongqing University, Chongqing, China
Email: {xiaoke317, liukai0807, jhua, ynyang, liangf}@cqu.edu.cn
†School of Logistics Engineering, Wuhan University of Technology, Wuhan, China
Email: bettycao@whut.edu.cn
‡Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong
Email: csvlee@cityu.edu.hk

***Abstract*—With recent advances in wireless communications, vehicular networks have attracted great interests in both industry and academia. This work aims at proposing a novel vehicular fog computing paradigm including both the system architecture and the scheduling algorithm. Specifically, we present a hierarchical architecture, which integrates the paradigm of both fog computing and the software defined networking (SDN). Then, we formulate a novel problem called Cooperative Service in Vehicular Fog Computing (CS-VFC), which aims at maximizing the bandwidth efficiency by coordinating the service in both the fog layer and the cloud layer. We prove that CS-VFC is NP-hard. On this basis, we propose an on-line scheduling algorithm, which incorporates with the network coding and makes scheduling decisions at SDN controller. In particular, it will determine the coding policy for each cloud node, and then it will implement both the intra and inter cooperation strategies at the fog layer. Finally, we build the simulation model by implementing NS3 simulator and SUMO. A comprehensive simulation is carried out to demonstrate the superiority of the proposed system architecture and the solution.**

I. Introduction

Vehicular ad-hoc networks (VANETs) is a promising approach to integrating wireless technologies with vehicles to meet the overwhelming data service demand by both the transportation administrators and the roadway users. Transportation administrators wish to collect and monitor real-time status of vehicles so as to enable advanced intelligent transportation systems (ITSs); while the drivers and passengers expect to access a variety of information services for various purposes, such as assisted driving, infotainment services and mobile commerce. However, unique characteristics of VANETs such as sparse distribution of roadside units (RSUs), diverse vehicle densities, and heterogeneous network interfaces, etc., make it challenging to optimize the utilization of resources to provide efficient information services.

Currently, vehicles mainly make use of cellular networks (e.g., 3G/4G) to access data services, which may suffer from high data traffic charges, long service delay or insufficient performance for enabling emerging ITSs such as path plan [1], intersection control [2], road reservation [3], etc. Some research has incorporated roadside infrastructures such as RSUs to enrich data services in VANETs [4]–[6]. However, it suffers from intermittent connections due to the short communication range and the sparse deployment of RSUs. Hence, a few studies have considered information services in heterogeneous vehicular communication environments by incorporating the concept of software defined networking (SDN) in VANETs [7], [8]. These studies mainly focused on the architecture design for managing heterogeneous wireless communication interfaces, in which the control plane is able to obtain global knowledge of the network and implement logically centralized control. On this basis, different algorithms have been proposed in SDN-based VANETs for data scheduling and resource allocation [9], [10]. Nevertheless, existing solutions cannot well coordinate the synergistic effect of different wireless communication interfaces based on their respective features so as to maximize overall system performance.

This work is dedicated to proposing a novel vehicular fog computing paradigm including both the system architecture and the scheduling algorithm. In particular, due to the heterogeneous wireless communication interfaces in VANETs, we abstract those roadside infrastructures with shorter radio coverage, such as RSUs, WiFi access points (APs), etc., as the fog node in VANETs, since they are typically closer to the end users (i.e., vehicles) in the sense of both physical deployment and communication delay. On the other hand, for those infrastructures with wider coverage, which are generally far away from the end users, such as cellular network base stations (BSs), are abstracted as the cloud node. To facilitate description in the rest of this paper, we call the communication between vehicles and the fog nodes as V2F, the communication between vehicles as V2V, and the communication between vehicles and the cloud nodes as V2C. On the other hand, network coding has been demonstrated as a promising approach to enhance bandwidth efficiency in VANETs [9], [10]. In this work, the bitwise exclusive-or (XOR) coding operation is considered for information services, due to its trivial implementation overhead. Consequently, with such a hierarchical architecture, this work further proposes a network coding based data scheduling algorithm to enhance the overall

978-1-5386-7646-2/19/$31.00 ©2019 IEEE

system performance by best coordinating V2C, V2F and V2V communications.

The main contributions of this work are outlined as follows.

- We present a novel hierarchical architecture in heterogeneous VANETs which consists of fog nodes and cloud nodes. Vehicles within the coverage of a fog node can either retrieve information locally via V2F communication or retrieve information from cloud nodes via V2C communication, depending on information availability as well as scheduling policies.
- With the presented fog computing paradigm in VANETs, we formulate a novel problem called Cooperative Service in Vehicular Fog Computing (CS-VFC). In CS-VFC, vehicles request for different services and requested data items will be broadcast by cloud nodes incorporating with bitwise exclusive-or (XOR) coding strategy. Vehicles are allowed to cache unrequested data items or the packets which are not immediately decodable. With cached information of different vehicles, cooperative services can be scheduled via V2V and V2F communications. The objective of CS-VFC is to maximize the bandwidth efficiency of V2C communication by devising an optimal coding strategy so as to best coordinate the services between the cloud layer and the fog layer.
- We prove the NP-hardness of CS-VFC problem by constructing a polynomial-time reduction from the Minimum Clique Cover (MCC) problem, which is a classic NP-hard problem [11].
- We propose a heuristic algorithm for coding determination at the SDN controller, as well as corresponding intra and inter fog cooperation schemes.
- We build the simulation model by NS3 simulator and SUMO, and give a comprehensive performance evaluation, which demonstrates the superiority of the proposed solution.

The rest of this paper is organized as follows. Section II reviews related works. Section III presents the fog computing paradigm in VANETs. Section IV formulates the CS-VFC problem and proves that it is NP-hard. Section V proposes an on-line scheduling algorithm as well as the service protocols at the fog layer. Section VI builds the simulation model and gives performance evaluation. Finally, Section VII concludes this work.

II. Related Work

Great efforts have been devoted to information services in VANETs. A large number of studies focused on improving vehicular communication quality and reliability at the MAC layer. J. Zhang *et al*. [4] proposed a Vehicular Cooperative Media Access Control protocol (VC-MAC), which jointly exploits the cooperation of V2I and V2V communications to enhance spatial reusability. The proposed VC-MAC protocol enables vehicles to cooperatively share their cached information via V2V communication when they are driving out of the RSUs coverage. Y. Bi *et al*. [12] proposed a Multi-Channel Token Ring MAC Protocol (MCTRP) for V2V communication. The CSMA/CA mechanism is applied for delivering emergency messages with low delay. In addition, a token-based data exchange protocol is designed to improve bandwidth efficiency for non-safety applications. Y. Kim *et al*. [13] proposed a Coordinated multichannel MAC protocol (C-MAC) for VANETs. Vehicles can send channel reservation request to the RSU within their allocated time slot. RSU will coordinate the broadcast sequence of safety related messages to avoid the collision with control messages. Also, the protocol will allocate the service channel based on throughput and delay constraints to enhance overall service quality.

Fog computing is envisioned as a promising paradigm in supporting low latency, high mobility and location dependent applications. I. Stojmenovic *et al*. [14] firstly claimed that applying fog computing together with the SDN concept would resolve the main issues in vehicular networks, such as intermittent connectivity, collisions and high packet loss rate, by augmenting V2V communications with V2I communications and centralized control. X. Hou *et al*. [15] proposed a vehicular fog computing (VFC) paradigm, in which some slow moving or parked vehicles are considered as infrastructures for communication and computation. By analyzing different scenarios in real vehicular mobility environments, it demonstrated that the proposed VFC paradigm can exploit tremendous computational potentialities based on better utilization of individual communication and computational resources of each vehicle. Y. Xiao *et al*. [16] proposed a visional concept on vehicular fog computing that turned connected vehicles into mobile fog nodes and utilized mobility of vehicles for providing cost-effective and on-demand fog computing for vehicular applications. In addition, open issues and technical challenges on enabling vehicular fog computing has been discussed. C. Huang *et al*. [17] formalized the vehicular fog computing architecture which comprises three layers, namely smart vehicles as the data generation layer, roadside units/fog nodes as the fog layer, and cloud servers as the cloud layer. A fog-assisted traffic control system is discussed as an example to show the benefit of vehicular fog computing, and also the forensic challenges and potential solutions are discussed.

Recently, network coding has been incorporated into data dissemination to improve the bandwidth efficiency. H. Ji *et al*. [18] proposed a network coding based solution to enable mobile hosts (MHs) which are not neighbors to cooperate indirectly. Specifically, In the NCB scheme, the mobile support station (MSS) examines the cache information of a group of MHs rather than individual MHs, and makes encoding decisions for broadcasting. The requesting MH can retrieve a data item from its peers to decode the encoded packet, and obtain the desired data item sequentially. Meanwhile, many studies have incorporated the idea of SDN into VANETs. Z. He *et al*. [7] proposed an SDN-based architecture to enable rapid network innovation for vehicular communications, in which vehicles and roadside units are

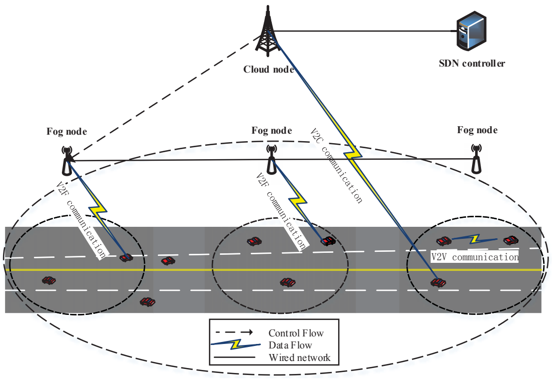

Fig. 1. System architecture

abstracted as SDN switches. In addition, network resources such as wireless bandwidth can be allocated and assigned by the logically centralized control plane, which provides a more agile configuration capability. Y. Liu *et al.* [8] presented an SDN architecture for GeoBroadcast in vehicular networks. The SDN controller helps the source vehicles find the path towards the destination vehicles with knowing the topological and geographical information. K. Liu *et al.* [19] investigated cooperative data dissemination via hybrid I2V/V2V communications within an RSU. The proposed model and solution, which are based on the centralized scheduler at the RSU, represent the first known vehicular network implementation of SDN concept. Y. He *et al.* [20] proposed an integrated framework that can enable dynamic orchestration of networking, caching and computing resources to improve the performance of next generation vehicular networks. The resource allocation strategy in this framework is formulated as a joint optimization problem. In addition, a novel deep reinforcement learning approach is proposed to solve this problem. Further, K. Liu *et al.* [10] extended the SDN-based architecture to heterogeneous vehicular communication environments, in which cellular BSs and RSUs are cooperated for providing information services, and proposed a memetic algorithm (MA) to solve the coding-assisted broadcast scheduling (CBS) problem which aims at maximizing the broadcast efficiency of BS and enhancing overall system performance.

Distinguishing from previous efforts, this paper embarks the first study on integrating both SDN and fog computing paradigm in heterogeneous vehicular communication environments by best exploiting the benefit of those roadside infrastructures which are closer to end users. In particular, this work presents a novel fog computing based architecture in VANETs. On this basis, this work proposes a novel solution to enhance overall system performance by maximizing the broadcast efficiency of cloud nodes and enabling cooperative data dissemination at the fog layer.

III. System Architecture

In this section, we present a novel system architecture, which forms the basis of fog computing paradigm in VANETs. As shown in Fig. 1, it is a hierarchical architecture in heterogeneous vehicular communication environments. For those roadside infrastructures with relatively smaller radio coverage, such as Wi-Fi APs, RSUs, 5G small/micro cells, etc., they are abstracted as the fog nodes to form the fog layer. Vehicles within the coverage of a fog node will inform the fog node with their physical (e.g., location, velocity, direction, etc.) and cyber (e.g., requested data, cached data, etc.) information via the heartbeat message.

On the other hand, for those roadside infrastructures with relatively larger radio coverage such as 3G/4G BSs, they are abstracted as cloud nodes to form the cloud layer. Cloud nodes are connected to the SDN controller via the backbone network, and fog nodes can upload the collected information to the cloud node via the wired connection. For vehicles which are not in the coverage of any fog node, they can report their status to the cloud node via V2C communication, so that the SDN controller is able to obtain global view of the network and exercise logically centralized scheduling. Note that it is critical to make use of the scare bandwidth resource of cloud node. In view of this, network coding is adopted at the cloud node to enhance broadcast bandwidth efficiency. The cloud node can retrieve data items from the database via the backbone network. Different data items can be encoded into a packet for broadcasting via V2C communication. In the rest of this paper, the term packet is used to refer to either encoded or non-encoded data items without causing ambiguity. Vehicles are able to cache unrequested data items or those packets which are not decodable immediately.

Given the broadcast packet and the vehicular physical information and cached information, there are two types of cooperative services: *intra fog* cooperation and *inter fog* cooperation. Specifically, for intra fog cooperation, each fog node can coordinate vehicles in its coverage to share data items via V2V or V2F communications. For instance, in a fog node, vehicles can share the cached packets to their neighbors via V2V communications directly or upload the cached packet to the fog node so that other vehicles in the same fog node can retrieve it via V2F communication. For inter fog cooperation, different fog nodes can share their local information with each other based on the scheduling of the cloud node, so that vehicles are able to be cooperatively served even if they are within different fog nodes. Obviously, with dedicated scheduling decisions for the cloud node and fog nodes, information services can be provided more efficiently via both intra fog and inter fog cooperation based on V2F and V2V communications.

IV. Problem Formulation

Without loss of generality, we consider that the system consists of one cloud node connected to the backbone network and it can retrieve data items from the database $D = \{d_1, d_2, ..., d_{|D|}\}$ for information services. Note that it can be straightforwardly extended to the scenario where multiple cloud nodes are connected to the SDN controller. In the concerned scenario, there is a set of fog nodes $N = \{n_1, n_2, ..., n_{|N|}\}$. The set of vehicles in the system is denoted by $K = \{k_1, k_2, ..., k_{|K|}\}$. Each vehicle k_m $(1 \leq m \leq |K|)$ may request a set of data items, which is denoted by $Q_m = \{d'_1, d'_2, ..., d'_{|Q_m|}\}$ $(Q_m \subseteq D)$. For the set of vehicles which are in the coverage of the fog node n_j $(1 \leq j \leq |N|)$ is denoted by $K_j = \{k^j_1, k^j_2, ..., k^j_{|K_j|}\}$. The bitwise XOR coding strategy is applied at the cloud node, and hence it needs to schedule a set of encoded packets for broadcasting, which is denoted by $P = \{p_1, p_2, ..., p_{|P|}\}$. The cached content for vehicle k_m is denoted by C_m.

Therefore, the CS-VFC problem is formulated as follows. Given the database D, the vehicle set K, the request set Q, the cache set C and a set of fog nodes N, the problem is to find a set of encoded packets P at the cloud node for broadcasting, as well as to determine corresponding data dissemination via V2V and V2F communications for infra fog cooperation and inter fog cooperation, so that Q can be satisfied with the minimum number of packets in P.

The CS-VFC problem can be proved as NP-hard by constructing a polynomial-time reduction from a well-known NP-hard problem, namely, Minimum Clique Cover (MCC) [11]. In the following, we give a sketch of the idea for the proof, while the detailed mapping procedures from CS-VFC to MCC will be introduced in Section V.

- First, the definition of the MCC problem is recapitulated as follows. Given an undirected graph $G = (V, E)$, MCC is to find the minimum set of cliques in *G* such that each vertex in *V* is in at least one of the cliques.
- Second, the general rules for constructing a graph *G* from CS-VFC problem is described as follows. For each vehicle k_i which requests for a data item d_j, we create a vertex v_{ij} in *V*. Then, for two vertices $v_{i_1j_1}$ and $v_{i_2j_2}$, if broadcasting a coded packet $d_{j_1} \oplus d_{j_2}$ could serve both the vehicles (i.e. k_{i_1} and k_{i_2}), then we create an edge between $v_{i_1j_1}$ and $v_{i_2j_2}$. Note that $d_{j_1} \oplus d_{j_2}$ may not be necessarily to serve k_{i_1} and k_{i_2} directly. To be elaborated in the next section, we will design intra fog and inter fog cooperation strategies to further explore the potentiality of network coding.
- Third, we give a brief analysis about the rationality of the problem reduction from MCC to CS-VFC. With the constructed graph *G*, clearly, given any clique in *G*, for example, suppose $v_{i_1j_1}, v_{i_2j_2}, .., v_{i_sj_s}$ form a clique, then the coded packet $d_{j_1} \oplus d_{j_2} \oplus ... \oplus d_{j_s}$ can serve $k_{i_1}, k_{i_2}, .., k_{i_s}$ simultaneously. Note that the union of the vertices in *G* represents all the requested data items of all the vehicles. Therefore, if we could find a set of cliques which cover all the vertices, then the corresponding encoded packets could be determined to serve all the outstanding requests. Recall that the objective of CS-VFC is to find the minimum number of encoded packets in *P* to satisfy all the requests in *Q*. With above analysis, clearly, the minimum number of encoded packets can be determined if and only if the minimum set of cliques which cover *G* is found.

The above proves the NP-hardness of the CS-VFC problem.

V. Proposed Solution

In this section, we propose a cooperative scheduling algorithm (CS) to solve the CS-VFC problem. The general idea of the CS is outlined as follows. First, in each scheduling period, it constructs a graph G based on current status of vehicles. Second, it determines a set of encoded packets based on an on-line scheduling algorithm. Third, it performs intra and inter fog cooperation via V2V and V2F communications. Detailed steps are elaborated below.

A. Graph construction

To facilitate the intra and inter fog cooperation, from the view point of a cloud node, the requested and the cached data items of vehicles in each fog node will be handled as a whole. Specifically, for vehicle set $K_j = \{k^j_1, k^j_2, ..., k^j_{|K_j|}\}$ in fog node n_j $(1 \leq j \leq |N|)$, we denote UQ_j as the union of requested data by each k^j_i, and denote UC_j as the union of cached data by each k^j_i $(k^j_i \in K_j)$. Then, the rules for constructing the graph G are presented as follows.

First, it traverses every requested data in each UQ_j $(1 \leq j \leq |N|)$, if d_i is found in UQ_j, a vertex v_{ji} is created. Then, for any two vertices v_{ji} and v_{mn} $(1 \leq j, m \leq |N|$ and $1 \leq i, n \leq |D|)$, if they satisfied any of the conditions below, then an edge would be added between them.

1) Condition 1: $i = n$;

2) Condition 2: $i \neq n$, $d_i \in UC_m \wedge d_n \in UC_j$;

3) Condition 3: $i \neq n$, $(d_i \in UC_m \wedge d_n \in UQ_j) \vee (d_i \in UQ_m \wedge d_n \in UC_j)$.

We give an example to better illustrate the idea of graph construction. Suppose the status (e.g., fog node information, requested and cached data items) of each vehicle is shown in Table I. Then, according to the union operation, the view from the cloud node is shown in Table II.

In Fig. 2, the black solid edge is added based on condition 1, such as the edge between v_{12} and v_{32}. Clearly, in this case, broadcasting d_2 can serve corresponding vehicles in the fog nodes n_1 and n_3 simultaneously.

TABLE I
Vehicle Information

Fog Node	Vehicle	Cached data	Requested data
n_1	k_1	d_1, d_6	d_4
	k_2	d_3, d_6	d_5
	k_3	d_3, d_7	d_2
n_2	k_4	d_1, d_2	d_5
	k_5	d_1, d_4	d_6
	k_6	d_2, d_3	d_7
n_3	k_7	d_3, d_5	d_7
	k_8	d_1, d_3	d_4
	k_9	d_5, d_6	d_2

TABLE II
The view from the cloud node

Fog Node	Cache set UC	Request set UQ
n_1	d_1, d_3, d_6, d_7	d_2, d_4, d_5
n_2	d_1, d_2, d_3, d_4	d_5, d_6, d_7
n_3	d_1, d_3, d_5, d_6	d_2, d_4, d_7

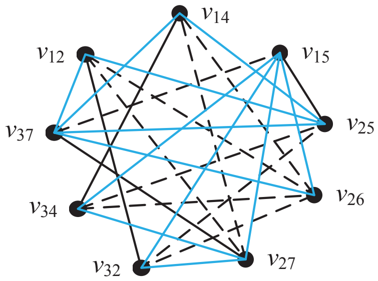

Fig. 2. Constructed graph G

The black dashed edges are created based on condition 2, such as the edge between v_{12} and v_{26}. In this case, broadcasting $d_2 \oplus d_6$ can serve corresponding vehicles in the fog nodes n_1 and n_2 simultaneously. Note that it may require intra fog cooperation to decode out data items. For example, k_3 in n_1 requests d_2 but it does not cache d_6. However, k_1 in n_1 has cached d_6. Therefore, k_1 needs to share d_6 to k_3 via either V2F or V2V communications.

The blue solid edges are created based on condition 3, such as v_{12} and v_{37}. In this case, it is not that straightforward to serve corresponding vehicles by simply broadcasting $d_2 \oplus d_7$, because none of the vehicles in n_3 has cached either d_2 or d_7. However, since d_2 is requested by the vehicles in both n_1 and n_3, and meanwhile d_7 is cached in n_1, it would be promising to have inter fog cooperation in such a case. Specifically, d_2 can be decoded out in n_1 by broadcasting $d_2 \oplus d_7$. Then, n_1 can transmit either d_2 or d_7 to n_3 via V2F communication. Finally, both d_2 and d_7 can be decoded out in n_3.

With above analysis, we have the following claim. Suppose $v_{i_1j_1}, v_{i_2j_2}, .., v_{i_sj_s}$ form a clique in the constructed G, then the coded packet $d_{j_1} \oplus d_{j_2} \oplus ... \oplus d_{j_s}$ can serve corresponding vehicles in fog nodes $n_{i_1}, n_{i_2}, .., n_{i_s}$ simultaneously via intra and inter fog cooperation. Recall that the objective of CS-VFC is to find the minimum number of encoded packets in P to satisfy all the requests in Q. Therefore, with the designed graph construction policy, CS-VFC can be mapped to the MCC problem, which is to find the minimum set of cliques in G such that each vertex in V is in at least one of the cliques.

B. An on-line scheduling algorithm

In this subsection, we propose a heuristic algorithm to enable on-line scheduling for packet encoding and broadcasting. Specifically, given scheduling period T, the algorithm needs to iteratively find K cliques from G ($K \leq T$) in each scheduling period. The following greedy method is adopted to search one clique in G.

First, it constructs a complement graph of G, denoted by G'. Second, it selects the vertex v^* with the maximum value of $1/(d(v)+1)$ for $\forall v \in G'$, where $d(v)$ represents the degree of vertex v. Third, it updates G' by removing v^* and its adjacent vertices. Fourth, it repeats the above steps until there is no vertex remaining in G'. In this way, the set of selected vertices forms an independent set of G', which is also a clique of G.

After finding a clique, the graph G is updated by removing the selected vertices and their connecting edges. Then, the above greedy method is repeated until either all the vertices have been removed from G (i.e. $V(G) = \emptyset$) or the number of selected cliques is greater than T (i.e. $K > T$).

C. Intra and inter fog cooperation

With the scheduled encoded packets, the cloud node will notify fog nodes and vehicles to perform intra and inter fog cooperation via V2V and V2F communications. To give clear exposition, we follow the above example to describe the cooperation scheme. Suppose the heuristic algorithm searches three cliques from G: $\{v_{12}, v_{32}, v_{26}\}$, $\{v_{34}, v_{15}, v_{25}\}$ and $\{v_{14}, v_{27}, v_{37}\}$, then the three encoded packet will be $d_2 \oplus d_6$, $d_4 \oplus d_5$ and $d_4 \oplus d_7$.

To determine intra and inter fog cooperation, for each encoded packet, the cloud node will back-trace the corresponding clique in G and check each edge of the clique. For instance, given $d_4 \oplus d_5$, it is generated from the clique $\{v_{34}, v_{15}, v_{25}\}$. Then, it checks the three edges $e(v_{15}, v_{25})$, $e(v_{34}, v_{25})$, $e(v_{34}, v_{15})$ and back-trace the rule by which the corresponding edge is generated. Recall that there are three rules for edge construction, each of them corresponds to certain operations. In particular, there are three cases:

- If the edge is constructed by condition 1 (e.g., $e(v_{15}, v_{25})$), then no cooperation is needed as it does not require network coding, and vehicles in both the fog nodes n_1 and n_2 can be served by d_5.
- If the edge is constructed by condition 2 (e.g., $e(v_{34}, v_{25})$), then the cloud node will instruct the fog node to perform intra fog cooperation. In this example, based on Table I, within the fog node n_2, d_4 needs to be shared from k_5 to k_4 via either V2V or V2F communications, depending on real-time locations of these vehicles. Then, k_4 can decode out d_5 by retrieving $d_4 \oplus d_5$. Similarly, within the fog node n_3, d_5 needs to be shared to k_8 via either k_7 or k_9, so that k_8 can decode out d_4.
- If the edge is constructed by condition 3 (e.g., $e(v_{34}, v_{15})$), then the cloud node will instruct the fog node to perform inter fog cooperation. In this example, based on Table I, vehicles in n_1 request both d_4 and d_5, but cached neither of them. On the other hand, vehicles in n_3 request d_4 and have cached d_5. Therefore, it requires the fog node of n_3 to transmit d_5 to the fog node n_1, so that k_2 can be directly served via V2F communication by retrieving d_5 and k_1 can decode out d_4 by retrieving $d_4 \oplus d_5$.

VI. Performance Evaluation

The simulation model is built based on the system architecture presented in Section III. In particular, NS3 simulator [21] and SUMO [22] are implemented to evaluate the performance of system, and the proposed algorithm is implemented by MATLAB. To give a clear view of the developed simulation model, Fig. 3 illustrates the key modules and procedures of the simulation. In this simulation,

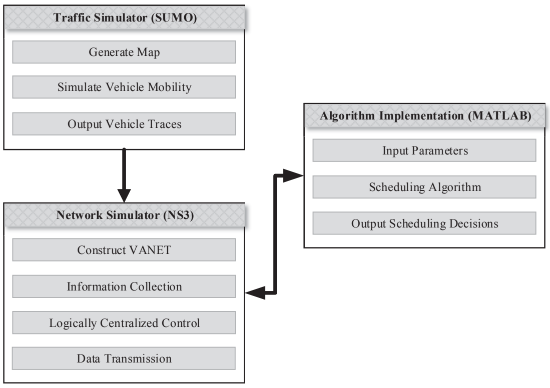

Fig. 3. Simulation setup

the map is exacted from a 2.5*km* x 2.5*km* area of the Jiangbei District, Chongqing, China, in which 9 fog nodes and 1 cloud node are randomly deployed. There are 120 vehicles in this area.

The communication range of each fog node is set to 500*m*. The default number of requested data items of each vehicle is set to 55, and the default number of cached packets of each vechile is uniformly distributed between 15 and 25. For comparison purpose, we have implemented two of the most competitive algorithms in the literature, which are NCB [18] and MA [10]. Details of these two algorithms have been introduced in Section II. To quantitatively evaluate algorithm performance, we design the following three metrics:

- Average service delay (ASD): Suppose there are m requests which are satisfied when vehicles are passing by the service area, and the service time of each request q_i $(1 \leq i \leq m)$ is denoted by t_i, which is the duration from its submission time to the time when the corresponding data item is received. Then, the average service delay of the system is computed by $\sum_{i=1}^{m} t_i/m$.
- Service ratio (SR): Suppose there are in total M requests submitted by passing vehicles and m requests are satisfied when vehicles are passing by the service area. Then, the service ratio of the system is computed by m/M.
- Bandwidth efficiency (BE): Suppose the cloud node broadcasts n packets in total and m requests are satisfied. Then, the bandwidth efficiency is computed by m/n.

Fig. 4 shows the ASD of algorithms under different number of requested data items. As shown, with an increasing number of requested data items, the ASD of all algorithms increases, whereas for CS, the ASD only increases slightly and it is much lower than that of NCB and MA in all cases. This demonstrates the scalability of CS.

Fig. 5 shows the SR of algorithms under different number of requested data items. As noted, NBC achieves similar performance with CS when the workload is very low (i.e., the number of requested data items is 35). However, with an increasing number of requested data items, CS can still serve most of the requests, while the SR of NCB and MA

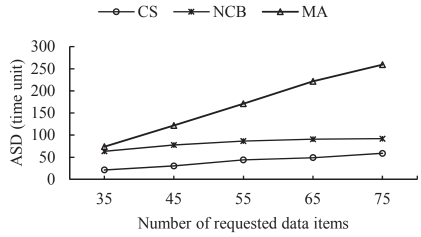

Fig. 4. ASD under different number of requested data items

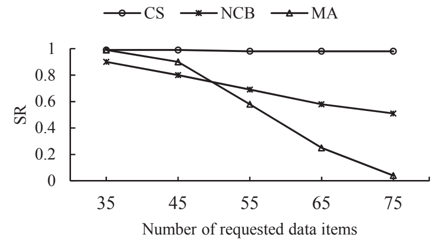

Fig. 5. SR under different number of requested data items

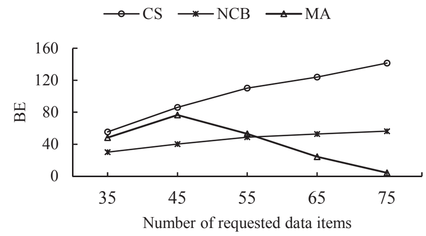

Fig. 6. BE under different number of requested data items

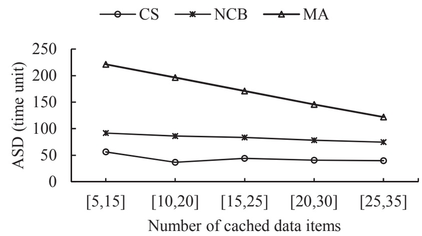

Fig. 7. ASD under different number of cached data items

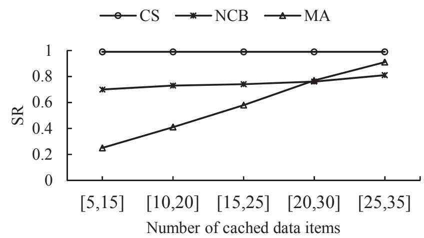

Fig. 8. SR under different number of cached data items

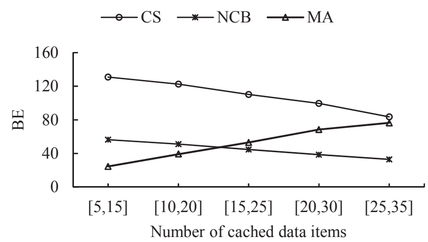

Fig. 9. BE under different number of cached data items

decreases dramatically. On the other hand, as more requests are satisfied by CS, recall that it also achieves much lower service delay for those satisfied requests at the same time, which evidently demonstrates the superiority of CS.

Fig. 6 shows the BE of algorithms under different number of requested data items. As noted, CS outperforms NSB and MA significantly in terms of improving the BE, especially in high workload environments. Due to the feature of MA, the BE of decreases dramatically with an increasing of system workload. This demonstrates that the effectiveness of the proposed algorithm on exploring the benefit of network coding and fog cooperation.

Fig. 7 shows the ASD of algorithms under different number of cached data items. As shown, with an increasing number of cached data items, the ASD of all algorithms decreases since the system workload increases. Nevertheless, CS also achieves the lowest ASD in all scenarios.

Fig. 8 shows the SR of algorithms under different number of cached data items. As noted, all algorithms have higher SR when the cached date items is increasing. Whereas CS constantly outperforms other algorithms in all ranges.

Fig. 9 shows the BE of algorithms under different number of cached data items. As observed, CS outperforms NSB and MA significantly in terms of improving the BE in different scenarios.

VII. Conclusion

In this work, we proposed a novel vehicular fog computing paradigm which enables the cooperation between SDN based services at the cloud layer and the distributed services at the fog layer. We formulated the problem called CS-VFC to maximize the bandwidth efficiency of V2C communication by coordinating the service in the cloud and fog layers. We proved the NP-hardness of CS-VFC by constructing a polynomial-time reduction from the MCC problem. Further, we proposed the CS solution to determine the coding strategy at the cloud layer as well as the corresponding intra and inter fog cooperation schemes to enhance the overall system performance. Finally, we built the simulation model by implementing NS3 simulator and SUMO. A comprehensive performance evaluation was carried out to demonstrate the effectiveness of the proposed solution.

Acknowledgements

This work was supported in part by the National Science Foundation of China under Grant No.61872049, Grant No. 61572088, Grant No. 61876025, and Grant No. 61502360; the Frontier Interdisciplinary Research Funds for the Central Universities (Project No. 2018CDQYJSJ0034).

References

[1] M. Wang, H. Shan, R. Lu, and R. Zhang, "Real-time path planning based on hybrid-vanet-enhanced transportation system," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 5, pp. 1664–1678, 2015.

[2] P. Dai, K. Liu, Q. Zhuge, H. M. Sha, V. C. S. Lee, and S. H. Son, "Quality-of-experience-oriented autonomous intersection control in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1956–1967, 2016.

[3] P. Wu, A. Che, F. Chu, and Y. Fang, "Exact and heuristic algorithms for rapid and station arrival-time guaranteed bus transportation via lane reservation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 1–16, 2018.

[4] J. Zhang, Q. Zhang, and W. Jia, "Vc-mac: A cooperative mac protocol in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1561–1571, March 2009.

[5] P. Dai, K. Liu, L. Feng, Q. Zhuge, V. C. S. Lee, and S. H. Son, "Adaptive scheduling for real-time and temporal information services in vehicular networks," *Transportation Research Part C Emerging Technologies*, vol. 71, pp. 313–332, 2016.

[6] K. Liu and V. C. S. Lee, "Rsu-based real-time data access in dynamic vehicular networks," in *International IEEE Conference on Intelligent Transportation Systems*, 2010, pp. 1051–1056.

[7] Z. He, J. Cao, and X. Liu, "Sdvn: enabling rapid network innovation for heterogeneous vehicular communication," *IEEE Network*, vol. 30, no. 4, pp. 10–15, 2016.

[8] Y. C. Liu, C. Chen, and S. Chakraborty, "A software defined network architecture for geobroadcast in vanets," in *IEEE International Conference on Communications (ICC)*, 2015, pp. 6559–6564.

[9] K. Liu, J. K. Y. Ng, J. Wang, V. C. S. Lee, W. Wu, and S. H. Son, "Network-coding-assisted data dissemination via cooperative vehicle-to-vehicle/-infrastructure communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1509–1520, 2016.

[10] 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 Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–12, 2017.

[11] P. R. J. Ostergard, "A fast algorithm for the maximum clique problem," *Discrete Applied Mathematics*, vol. 120, no. 1, pp. 197–207, 2002.

[12] Y. Bi, K. H. Liu, L. X. Cai, X. Shen, and H. Zhao, "A multi-channel token ring protocol for qos provisioning in inter-vehicle communications," *IEEE Transactions on Wireless Communications*, vol. 8, no. 11, pp. 5621–5631, 2009.

[13] Y. Kim, M. Lee, and T. J. Lee, "Coordinated multichannel mac protocol for vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6508–6517, 2016.

[14] I. Stojmenovic and S. Wen, "The fog computing paradigm: scenarios and security issues," in *Computer Science and Information Systems*, 2014, pp. 1–8.

[15] X. Hou, Y. Li, M. Chen, D. Wu, D. Jin, and S. Chen, "Vehicular fog computing: a viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, 2016.

[16] Y. Xiao and C. Zhu, "Vehicular fog computing: vision and challenges," in *IEEE International Conference on Pervasive Computing and Communications Workshops*, 2017, pp. 6–9.

[17] C. Huang, R. Lu, and K. K. R. Choo, "Vehicular fog computing: architecture, use case, and security and forensic challenges," *IEEE Communications Magazine*, vol. 55, no. 11, pp. 105–111, 2017.

[18] H. Ji, V. C. Lee, C.-Y. Chow, K. Liu, and G. Wu, "Coding-based cooperative caching in on-demand data broadcast environments," *Information Sciences*, vol. 385, pp. 138–156, 2017.

[19] K. Liu, J. K. Y. Ng, V. C. S. Lee, S. H. Son, and I. Stojmenovic, "Cooperative data scheduling in hybrid vehicular ad hoc networks: Vanet as a software defined network," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1759–1773, 2016.

[20] Y. He, C. Liang, Z. Zhang, F. R. Yu, N. Zhao, H. Yin, and Y. Zhang, "Resource allocation in software-defined and information-centric vehicular networks with mobile edge computing," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017, pp. 1–5.

[21] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena, "Network simulations with the ns-3 simulator," *SIGCOMM demonstration*, vol. 14, no. 14, p. 527, 2008.

[22] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility," in *The Third International Conference on Advances in System Simulation (SIMUL 2011), Barcelona, Spain*, vol. 42, 2011.