SDN-Based Resource Management for Autonomous Vehicular Networks: A Multi-Access Edge Computing Approach

Haixia Peng, Qiang Ye, and Xuemin (Sherman) Shen

Abstract

Enabling HD-map-assisted cooperative driving among CAVs to improve navigation safety faces technical challenges due to increased communication traffic volume for data dissemination and an increased number of computing/storing tasks on CAVs. In this article, a new architecture that combines MEC and SDN is proposed to address these challenges. With MEC, the interworking of multiple wireless access technologies can be realized to exploit the diversity gain over a wide range of radio spectrum, and at the same time, computing/storing tasks of a CAV are collaboratively processed by servers and other CAVs. By enabling NFV in MEC, different functions can be programmed on the server to support diversified AV applications, thus enhancing the server's flexibility. Moreover, by using SDN concepts in MEC, a unified control plane interface and global information can be provided, and by subsequently using this information, intelligent traffic steering and efficient resource management can be achieved. A case study is presented to demonstrate the effectiveness of the proposed architecture.

Introduction

Autonomous vehicles (AVs) have the potential to deal with traffic related problems, including avoiding traffic accidents caused by human mistakes and reducing traffic congestion, energy consumption, and exhaust pollution [1]. However, the market perspective of AVs continues to face significant challenges. The autonomous navigation accuracy of an AV depends on the timeliness and granularity of its perceived and predicted road environment [2], which is constrained by the AV's sensing capability. Due to the deficiency in sensing accuracy, especially in road surroundings with bad weather, confusing traffic signals, and faded lanes, it is difficult to guarantee complete AV safety without human intervention. Existing studies have indicated that, through enabling cooperative driving among and providing high-definition (HD) maps to AVs, the autonomous vehicular network (AVNET) emerges as a complementary technology to compensate the sensing deficiency and improve AV safety [3]. Connected and autonomous vehicles (CAVs) with the same driving direction can be grouped into platoons/convoys for cooperative driving. On the other hand, HD maps can assist CAVs for accurate accelerating or decelerating prior to obtaining the sensing information.

Forming platooning/convoying can improve CAV safe navigation in some highways or urban major avenues where vehicle density is relatively stable, whereas it may incur extra cost (e.g., time waste and fuel consumption) in urban areas due to constantly changing memberships in each platoon/convoy. Therefore, a promising AVNET scenario is the co-existence of HD-map-assisted cooperative driving vehicles and free driving vehicles interconnected via wireless networking, which can better adapt to different road environments. Here, vehicles in a free driving pattern can be either AVs or manual vehicles. However, the following characteristics pose technical challenges over current AVNET architecture.

Increased Communication Data Traffic Volume and Computing/Storing Tasks: Information from both onboard sensors and vehicle-to-vehicle/vehicle-to-infrastructure (V2V/V2I) interactions, such as sharing velocity and acceleration information between CAVs, is required for cooperative driving, which substantially increases the data traffic volume. Also, building/updating the HD maps of a road environment, which are then shared with other CAVs, leads to a large amount of data traffic in AVNETs. Moreover, to support cooperative driving, the number of computing/storing tasks performed on some CAVs increases to collect and process the information from both onboard sensors and other CAVs. However, the computing and storing capabilities on each CAV are also limited to afford the increasing demands [4];

Heterogeneous Quality-of-Service (QoS) Requirements: Stringent delay requirements with guaranteed reliability is important to cooperative driving and safety-related applications. For example, a short response-delay from the vehicle traffic management system is required to maintain efficient cooperative driving. On the other hand, high throughput is required for data transmission of some non-safety applications, including HD map information and infotainment services. To guarantee the heterogeneous application QoS requirements, efficient QoS-oriented resource management is essential.

To accommodate the surge of data traffic volume, the integration of different types of wireless access technologies has been proposed, such as the interworking between dedicated short range

Digital Object Identifier: 10.1109/MWC.2019.1800371

The authors are with the University of Waterloo. Qiang Ye is the corresponding author.

communications (DSRC) and cellular networks [5]. To address the challenge caused by the increased number of computing/storing tasks, edge computing that moves computing and storing capabilities to the edge of the core network is one of the widely accepted technologies. For example, a two-level edge computing architecture has been proposed in [6] for coordinated content delivery in AVNETs. Another promising method is enabling collaborative computing among CAVs. For example, a collaborative computing scheme has been proposed in [7] to allow CAVs to share idle computing power with neighboring CAVs. To simultaneously address the challenges in communication and computing/storing, multi-access edge computing (MEC) has attracted lots of attention from both academia and industry. An industry specification group has been established by the European Telecommunication Standards Institute (ETSI) to identify the MEC features to support vehicle-to-everything (V2X) applications [8]. However, most existing works are either dedicated to supporting some non-computation-intensive vehicular applications with only bandwidth resources demanded, or focused on managing computing or bandwidth resources for CAV applications with homogeneous QoS requirements. Due to the increasing demand for multiple types of resources and the strict delay requirement to support some newly emerging computing-intensive CAV applications, it is difficult to apply the existing architecture to maximize the multi-resource utilization with different levels of QoS guarantee for diversified applications. Thus, a more comprehensive AVNET architecture is required.

In this article, we introduce an AVNET architecture which integrates the software-defined networking (SDN) and network function virtualization (NFV) concepts in MEC, to address the preceding challenges in both communication and computing while guaranteeing heterogeneous application QoS requirements. Specifically, the proposed architecture has the following advantages:

- Via MEC, multiple wireless access networks can interwork to support the increased data traffic volume, and quick computing responses can be provided by uploading the computing/storing tasks (e.g., cooperative driving) to the network edge to avoid extra delay for data transmission between MEC servers and the cloud-computing server, and hot content (e.g., HD maps) can be cached in and processed by MEC servers.
- By integrating SDN control with cloud-computing/MEC servers, unified control plane interfaces are provided by decoupling the control plane from the data plane without placing new infrastructures. With global network control, intelligent traffic steering and efficient resource management can be achieved to improve the overall resource utilization.
- By enabling NFV in could-computing and MEC servers, different network functions supporting AV applications can be programmed on servers flexibly with reduced provisioning cost.
- A joint multi-resource management scheme is designed under the proposed AVNET architecture to maximize overall resource utilization with heterogeneous QoS guarantee through computing task migration and radio bandwidth slicing.

The remainder of this article is organized as follows. First, an MEC-based AVNET architecture which incorporates both SDN and NFV technologies is proposed. Then, a joint multi-resource management scheme is presented, and a case study is provided to demonstrate the effectiveness of our designed resource management scheme under the proposed architecture. Finally, we discuss future research issues and draw concluding remarks.

An MEC-Based AVNET Architecture

In the following, we first present the problem statement and motivations to consider MEC, SDN, and NFV in AVNETs, and then propose an MEC-based AVNET architecture.

Problem Statement and Motivations

Existing studies have indicated that enabling cooperative driving among CAVs and providing real-time HD maps to CAVs are important to improve CAV safe navigation [9]. In cooperative driving, such as platooning, neighboring CAVs on one lane move at a steady speed and keep a small steady inter-vehicle space [1]. A leader vehicle is chosen to lead all the other CAVs (referred to as member vehicles) within a platoon to maintain the string stability [10]. To do so, some delay-sensitive information (e.g., speed and acceleration, braking, joining/leaving [11]) need to be shared among cooperative driving CAVs. Although providing static HD maps to show the road environment to CAVs can timely compensate the inaccurate sensing information to improve CAV safe navigation, the large amount of traffic volume is generated due to data exchanges among CAVs, which results in an increasing demand on bandwidth resources.

Moreover, new computing-intensive vehicular applications have sprung up with increasing demands for multiple types of resources. For example, large amounts of computing, storing, and bandwidth resources are demanded for virtual reality (VR). However, the increasing amount of computing/storing resources at each CAV can be cost-ineffective, and uploading tasks with high computing/storing requirements to the cloud-computing server increases task response delay due to extra data transmission, and also results in high traffic load onto the core network. Therefore, we leverage the MEC technology to form an MEC-based AVNET architecture. By enabling computing and storing capabilities at the edge of the core network, CAVs' tasks with high computing/storing requirements can be offloaded to MEC servers when short response delay is demanded for delay-sensitive applications. At the same time, CAVs are allowed to access the computing/storing resources at MEC servers via different wireless access technologies to accommodate an increasing communication demand. To further improve the resource management efficiency, we integrate SDN control into the MEC-based AVNET architecture, to enable programmable network configuration for flexible resource orchestration.

By activating SDN control functions on the edge servers, a hierarchical SDN control is deployed to realize radio resource pooling and resource slicing [12] among the wireless access infrastructures to improve the overall resource utilization and achieve differentiated QoS guarantees. Note that

By activating SDN control functions on the edge servers, a hierarchical SDN control is deployed to realize radio resource pooling and resource slicing among the wireless access infrastructures to improve the overall resource utilization and achieve differentiated QoS guarantees.

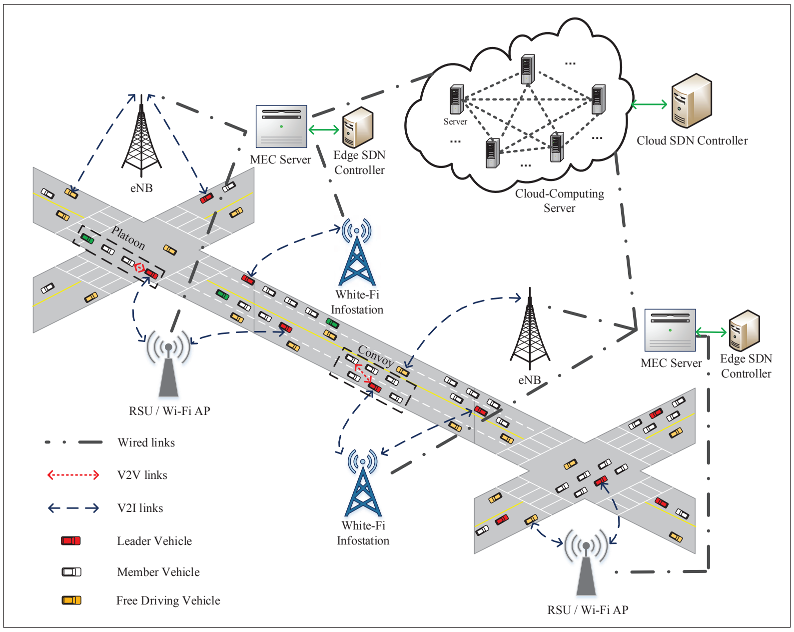

FIGURE 1. An MEC-based AVNET architecture.

MEC runs on top of the virtualization infrastructure [13] to support application programmability with reduced provisioning cost. With the decoupling of applications from its underlying hardware in NFV, each MEC server can host functions based on flexible application/service demands.

Proposed MEC-Based AVNET Architecture

The proposed MEC-based AVNET architecture is illustrated in Fig. 1, which is a two-tier server structure [14], that is, a cloud-computing server in the first tier and some MEC servers in the second tier. Based on the resource availability (including computing, storing, and bandwidth resources) and application QoS, computing tasks can be processed on CAVs, or be uploaded to and processed by MEC servers or the cloud-computing server. For example, delay-sensitive applications, such as platooning management, dynamic HD map management, and other safety-related applications, are prioritized to be processed on MEC servers for less processing latency.

To improve both resource utilization and network scalability, the MEC servers are placed at the edge of the core network to be close to and maintain very few communication hops with vehicles. Thus, an MEC server can control the computing task offloading for a large number of vehicles under the coverages of several base stations (BSs), and the enhanced service area of each MEC server can better support the fluctuating service demand caused by high vehicle mobility, where the service area of an MEC server is defined as the coverage areas of all BSs connected to the server. Via integrating the NFV concept into each MEC server, computing and storing resources on each MEC server are virtualized to host functions for different applications and services. Thus, different MEC applications, represented by virtual network functions (VNFs), can share the same set of virtualized resources which are under the NFV infrastructure and can be jointly orchestrated for different multi-access edge hosts for heterogeneous QoS guarantee. Moreover, an SDN control module is also running on the NFV-enabled MEC server. With the assistance of the edge SDN controller's partial-global information, the local computing/storing resources, including computing/storing resources placed in CAVs and in the MEC server, can be efficiently allocated among different CAV tasks with heterogenous QoS guarantee. Bandwidth resources on different access networks are orchestrated through resource pooling and resource slicing to improve the overall radio resource utilization.

On the cloud server placed at the core network, computing and storing resources are also virtualized, and the processing of integrating the NFV concept into the cloud computing server is similar to that of MEC servers. SDN control functions are programmed in cloud virtual machines (VMs) to decouple the control plane from the data plane, and to manage global traffic steering among the cloud servers, upon which the resource availability and usage can be significantly enhanced. On the other hand, each MEC server forwards its state information, including the amount of idle computing/storing resources and QoS demands from different CAV applications, to the cloud SDN controller after pre-analysis and pre-processing (e.g, quantization), and the controller then makes decisions for task migration among MEC servers based on the received information.

SDN-Enabled Resource Management

Based on the proposed MEC-based AVNET architecture, the increased amount of data traffic and computing/storing tasks can be supported with QoS guarantee. In this section, we investigate how to optimize the management of computing/storing resources among MEC servers and the slicing of bandwidth resources among BSs.

Resource Management Scheme

For MEC servers with pre-allocated computing and storing resources, inter-MEC resource sharing is of paramount importance. Through migrating computing/storing tasks, the distribution of task requests can be balanced among MEC servers according to their resource usages, thus enhancing the computing/storing resource utilization. Processing results of the migrated tasks will be returned to the original MEC server to respond to the request. In addition, the task migration decision can also be made based on its requester's moving direction, and the task processing results can be directly delivered to the requester once the requester moves into the service area of the new MEC server, to reduce the response delay. Consider a scenario with one cloud-computing server, M MEC servers denoted by M_i, $i = 1, ..., M$, and N CAVs distributed over the entire AVNET. Each BS is connected to one of the M MEC servers, and WiFi/DSRC, White-Fi, and cellular technologies are applied to support CAV applications. In the following, resource management schemes for computing, storing, and bandwidth are investigated for the considered scenario.

Computing and Storing Resource Management: Let $C_i^k(t)$ and $S_i^k(t)$ denote the amounts of computing and storing resources that MEC M_i allocates to CAV k at time slot t. Due to the fixed amount of computing/storing resources at each MEC server and the varying amount of computing/

storing tasks generated by the regionally distributed moving CAVs, each MEC only processing the computing/storing tasks of CAVs within its service area can lead to overloaded or underloaded task processing. To mitigate the imbalanced task requests, computing/storing tasks can be migrated among MEC servers to increase the computing/storing resource utilization, which, on the other hand, incurs migration cost in terms of extra bandwidth consumption and extra response delay.

In order to obtain optimal computing and storing resource allocation while balancing the tradeoff between increasing the computing/storing resource utilization and reducing the task migration cost, an optimization problem is described as shown in Fig. 2. The objective is to maximize the network utility which is defined as the summation of utilities of each individual MEC server. The utility of an MEC server allocating computing/storing resources to CAVs is defined with the consideration of computing/storing resource utilization and task migration cost, where the computing (or storing) resource utilization of MEC server M_i is defined as the ratio of the amount of occupied resources over its total amount of computing (or storing) resources. The input of the formulated problem includes:

- The computing, storing, and bandwidth resources placed at MEC server M_i, denoted as C_i^{max}, S_i^{max}, and B_i^{max}, respectively, where B_i^{max} is the total available bandwidth resources of the multiple radio access technologies for AVNETs.
- The total amount of resources required by CAV k at time slot t, denoted as $D^k(t) = \{C^k(t), S^k(t), B^k(t)\}$, including the required amounts of computing resources $C^k(t)$ and storing resources $S^k(t)$ for processing its application requests, and bandwidth resources $B^k(t)$ for downlink transmissions, where we have $C^k(t) \geq 0$, $S^k(t) \geq 0$, and $B^k(t) > 0$.
- A downlink response delay threshold, T_{th}^k, used to guarantee that CAV k (either generating delay-sensitive requests or delay-tolerant requests) receives the response before it moves out of the service area of the MEC server, and a latency threshold, L_{th}^k, used to guarantee the delay requirement of CAV k generating a delay-sensitive request.

In the problem formulation, the following constraints are considered:

- For CAVs that either generate delay-sensitive requests or delay-tolerant requests, the total time cost, that is, the time interval from the time that CAV k's computing/storing task is received by MEC server M_i until the time instant the corresponding response packet generated by MEC server M_i is received by CAV k through a BS, should be less than the downlink response delay threshold, T_{th}^k. Moreover, for CAVs that generate delay-sensitive requests, the total time cost should also be less than the latency threshold, L_{th}^k.
- Computing/storing resource constraints, that is, the total computing (or storing) resources allocated by all the M MEC servers to CAV k should satisfy the computing (or storing) resources required by CAV k, that is, $C^k(t)$ ($S^k(t)$). By solving the formulated maximization problem, the optimal computing and storing resource allocation $C_i^k(t)$ and $S_i^k(t)$ can be obtained.

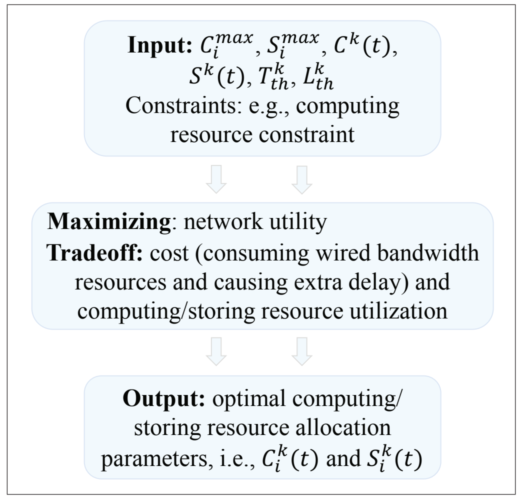

FIGURE 2. Diagramming for formulating optimization problem of computing/storing resources.

Bandwidth Management: Due to the large service area of each MEC server, bandwidth reuse is considered among the BSs connected to the same MEC server. With the consideration of different BS coverages (RSUs/WiFi APs, White-Fi infostations, and eNBs) from different wireless access technologies, CAVs can choose to associate with BSs providing different levels of communication qualities (e.g., transmission rates). To improve bandwidth resource utilization, BSs can reuse bandwidth resources with acceptable inter-cell interference. Therefore, the goal of bandwidth slicing is to determine the set of optimal slicing ratios for different BSs, such that the aggregate network utility is maximized, and the heterogenous application QoS requirements are satisfied.

Taking BS j and CAV k under the service area of MEC server M_i as an example. Let $\gamma_j^k(t)$ denote the achievable downlink transmission rate when CAV k is associated with BS j at time slot t. The utility of CAV k associated with BS j is a concave function of $\gamma_j^k(t)$, for example, a logarithmic function, and the aggregate network utility is the summation of utilities of each individual CAV. Then, a network utility maximization problem is formulated, in which a two-level resource allocation is considered:

- Slicing the total bandwidth resources B_i^{max} into small resource slices, where the set of slicing ratios are denoted by $\{\beta_j | j = 1, 2, ..., I_i(t)\}$, where $\Sigma_{j=1}^{Ii(t)} \beta_j = 1$ and $I_i(t)$ is the number of BSs within the service area of MEC server M_i at time slot t.
- Partitioning the sliced resources to CAVs under the coverage of and associating to each BS.

Constraints under consideration include:

- $D_i^k + R_i^k \leq T_{th}^k$ and $D_i^k + R_i^k \leq L_{th}^k$.
- $\gamma_j^k(t) \geq \hat{\gamma}^k$, where $\hat{\gamma}^k$ is defined as a transmission rate threshold for CAVs that generate delay-tolerant requests
- $\Sigma_k B_i^k(t) \leq \beta_j B_i^{max}$, where $B_i^k(t)$ is the amount of bandwidth resources that BS j allocates to CAV k at time slot t.

The latency constraints for CAV k reflect the coupling relation between the two formulated problems for computing/storing resource and bandwidth resource management. Thus, these two problems have to be jointly solved, and the obtained optimal computing/storing resource

For MEC servers with pre-allocated computing and storing resources, inter-MEC resource sharing is of paramount importance. Through migrating computing/storing tasks, the distribution of task requests can be balanced among MEC servers according to their resource usages, thus enhancing the computing/storing resource utilization.

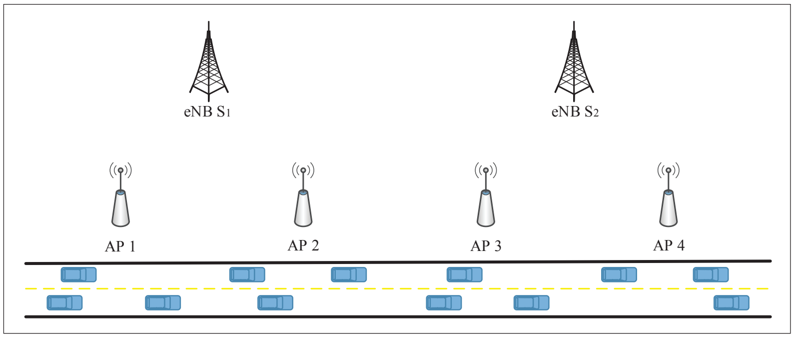

FIGURE 3. A network scenario under consideration.

allocation results and bandwidth resource allocation results maximize the network utility while collaboratively satisfying the delay requirements for delay-sensitive applications.

Case Study

In this subsection, a case study is presented to demonstrate the effectiveness of our bandwidth management scheme over a simple network scenario, where cellular and WiFi communication technologies are employed to support CAV applications. As shown in Fig. 3, two eNBs (eNB S1 and eNB S2) and four WiFi APs (AP 1 to AP 4) are distributed over one side of a one-way straight road with two lanes. Locations of the eNBs and WiFi APs are fixed. Transmit power of each eNB is set to be 40 dBm (i.e., 10 watts) to ensure a maximum communication range of 600 m, which fully covers the CAVs on the road, whereas the transmit power of a WiFi AP is adjusted and limited to 2.5 watts, and the coverage rate of WiFi APs is less than or equal to 1. The coverage rate of a type of BS is defined as the probability that a CAV is within the coverage of the BS. The CAV density over one lane, that is, the number of CAVs in each lane per meter, is assumed to vary within the range of [0.04, 0.20]CAV/m.

To balance the bandwidth resource utilization and inter-cell interference, bandwidth reuse is considered between two WiFi APs and between one eNB and one WiFi AP (only the WiFi APs having no overlapping coverage area with the eNB can reuse the bandwidth allocated to the eNBs). Thus, the total available bandwidth is sliced into three slices allocated to eNB S_1, eNB S_2, and WiFi APs with the slicing ratios β_1, β_2, and β_w, respectively. For example, CAVs associating to AP 3 and AP 4 can reuse the bandwidth resources $\beta_w B_i^{max} + \beta_1 B_i^{max}$, where the first part is the amount of bandwidth sliced to WiFi APs and the second part indicates reusing the bandwidth sliced to eNB S_1 since there is less interference between these two APs and eNB S_1. For each CAV, only one type of delay-sensitive applications or delay-tolerant applications is considered in each time slot, where the probability that the CAV generates a delay-sensitive application request is p. Here, we take platooning/convoying as a delay-sensitive application example with delay bound 10 ms and delay bound violation probability 10^{-3}, and take downloading HD maps as a delay-tolerant application example. The aggregate bandwidth resources from BSs and WiFi APs are 20 MHz.

As discussed previously, an optimization problem that maximizes the network utility is formulated and certain approximation methods can be applied to solve this problem to obtain the optimal bandwidth slicing ratios $\{\beta_1^*, \beta_2^*, \beta_w^*\}$ and BS-CAV association patterns. To demonstrate the efficiency of the designed bandwidth slicing scheme, we compare our designed scheme with the max-SINR scheme proposed in [15] and the max-utility scheme proposed in [12]. Since transmit power controlling is not enabled by both the max-SINR and max-utility schemes, the transmit power of WiFi AP is set as 1 watt for both schemes. For the max-SINR scheme, no bandwidth slicing is enabled and CAVs associate with the BS providing the highest SINR level.

Network throughput, that is, the summation of individual CAV's achievable transmission rate, is used to demonstrate the performance of our designed scheme. Figures 4 and 5 show the achievable network throughputs of the designed scheme and two other benchmark schemes under different p and CAV densities. Figure 4 shows that when CAV density is set as 0.05 CAV/m, our designed scheme achieves higher network throughput than the max-utility scheme, and the max-SINR scheme when p is small. A large p indicates that with high probability, a CAV generates delay-tolerant application, and thus a small amount of spectrum bandwidth resources are required to satisfy the applications' QoS requirements. For the max-SINR scheme, more bandwidth resources are allocated to the CAVs with higher SINR, thus achieving higher network throughput when $p > 0.6$. Figure 5 shows that our designed scheme is more robust to the CAV density changing than the two other schemes. Due to equal spectrum allocation among CAVs associating to the same BS for both the max-SINR and the max-utility schemes, the scenarios with small CAV densities can be supported by the two schemes, while all scenarios with CAV densities from 0.08 to 0.40 can be accommodated by our designed scheme. Moreover, about 50 percent and 40 percent higher network throughput can be achieved by the designed scheme than the max-utility scheme and the max-SINR scheme, respectively. That is because our designed scheme allocates on-demand bandwidth among CAVs instead of equal bandwidth partitioning. The transmit power control is also considered in our designed scheme. For the designed scheme, the effects of the applied approximation methods for solving the formulated optimization problem are also evaluated. It is obvious that the network throughput achieved with approximations closely matches with the exact one without approximations.

Open Research Issues

Combining SDN and NFV with MEC architecture to support the increasing data traffic volume while guaranteeing heterogeneous QoS requirements for different services in AVNETs, is still in its infancy. In this section, some open research issues are discussed.

MEC Deployment: In our designed resource management scheme, we consider that MEC servers are placed at the edge of the core network to maintain two-hop wireless transmissions between a CAV and an MEC server. Placing MEC servers close to BSs reduces the computing task response delay, but increases the computing server deployment cost. Therefore, how to place MEC servers and how much computing and storing resources should be placed on each MEC server need to be investigated for the MEC deployment problem. A simple method to deploy MEC servers is based on local service requirements to balance the placement cost with CAVs' application QoS requirements. Moreover, considering service demand fluctuations due to the high CAV mobility, vehicular traffic variations, and increasingly diversified applications, pre-placed MEC deployment results in inefficient resource utilization. Unmanned aerial vehicle (UAV)-assisted MEC deployment, that is, mounting MEC servers in UAVs, is a promising method to this problem. Via the decentralized controlling by each edge SDN controller, moving paths for MEC-mounted UAVs and multiple resources can be scheduled to satisfy the QoS requirements even in service demand fluctuation.

Task Offloading: Since computing/storing resources on each MEC server are limited and task migration from one MEC server to another incurs extra cost, the amount of tasks allowed to be registered and processed in MEC servers should be constrained. Designing a proper task offloading criterion is necessary to maximize the computing/storing resource utilization under the constraints of task migration costs. For the proposed architecture, we determine where to register CAV applications based on the application types, that is, only safety-related applications are registered in the MEC servers. However, other criteria, such as delay requirements for each type of computing task, should also be taken into consideration to optimize the offloading decisions among MEC servers. Given the amount of resources on each MEC server, how to design appropriate criteria for task offloading among MEC servers to balance the trade-off between QoS satisfaction and minimizing the offloading cost needs more investigation. From the perspective of CAVs, each CAV can process the computing task by itself, offload it to neighboring CAVs when cooperative computing is enabled, or upload it to the MEC server. Due to the limited available computing/storing resources in each CAV and the fluctuations of service demands and available bandwidth resources, task offloading decisions made by any CAV can affect the resource utilization of the whole network and other tasks' QoS satisfaction, and therefore, how to make the task offloading decision for each CAV with consideration of the distribution of service requests and available bandwidth is important. However, making task offloading decisions for CAVs requires a central controller, and the interactions between the controller and CAVs can increase the cost for signaling exchange and time complexity. To address this issue, a decentralized reinforcement learning (DRL)-based offloading decision making scheme can be designed for each individual CAV.

QoS-Guaranteed Multiple Resource Management: Bandwidth and at least one type of the computing and storing resources are required by most tasks in AVNETs, such as offloaded computing task and content caching task, and even all three types of resources are demanded by some tasks, such as virtual reality. However, due to the coupled relation among them to guarantee QoS requirements for different tasks or balance the fairness among CAVs, it is challenging to simultaneously manage the three types of resources among different tasks from both MEC servers and CAVs perspectives.

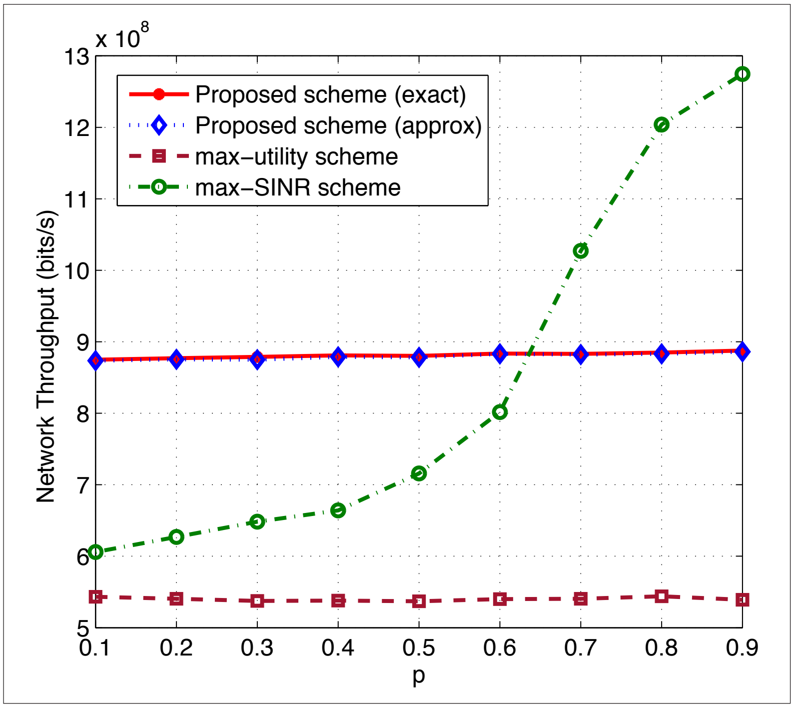

FIGURE 4. Network throughput vs. p with CAV density 0.05 CAV/m.

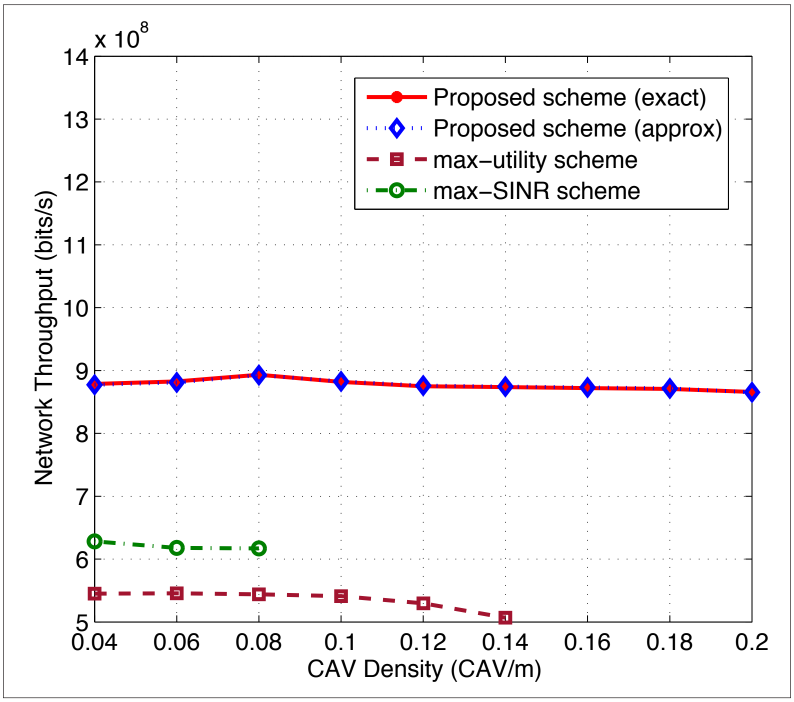

FIGURE 5. Network throughput vs. CAV density with p = 0.2.

Moreover, task offloading among MEC servers and CAVs impacts the amount of resource demanding in each MEC server and CAV, which makes the multiple resource management in our proposed architecture more challenging. To address these challenges, designing a DRL-based algorithm for each MEC server and CAV to jointly make task offloading decisions and manage the three types of resources is a potential solution.

Security and Privacy: How to ensure a secure communication among CAVs is a key research issue. Since the accelerating or braking decisions from communication-assisted CAVs are based on

Via integrating NFV into MEC, functions supporting different applications can be hosted on servers flexibly with reduced function provisioning cost. To achieve intelligent traffic steering and efficient multiple resource management, SDN control functionality is applied in the proposed architecture.

the collected information via V2V and V2I communications, security attacks on communication channels and sensor tampering may result in driving safety issues. Due to the MEC controllers, the privacy of applications registered in MEC servers can be improved through local communications. However, the MEC servers or cloud-computing servers can become the major targets of hackers, from which the attacks indirectly cause driving safety issues and result in serious privacy invasion. Moreover, exchanging individual vehicle information is required to support cooperative driving among CAVs. How to ensure identity privacy, data privacy, and location privacy is essential to stimulate cooperative driving among CAVs. To deal with these security and privacy issues, potential solutions include identity authentication for communications, access control at MEC and cloud-computing servers, and trust management from CAVs and servers.

Conclusion

In this article, we have proposed a new MEC-based architecture considering both SDN and NFV technologies to support the increasingly intensified computing and communication requirements in AVNETs. By applying MEC in AVNETs, computing and storing resources are moved to the edge of the core network and CAVs access the network via different wireless access technologies. Via integrating NFV into MEC, functions supporting different applications can be hosted on servers flexibly with reduced function provisioning cost. To achieve intelligent traffic steering and efficient multiple resource management, SDN control functionality is applied in the proposed architecture.

To further improve the overall resource utilization with heterogeneous QoS guarantee, a joint multi-resource management scheme has been designed, where computing/storing resource utilization are enhanced through task migration and bandwidth resource slicing is conducted among heterogeneous BSs. A case study has been conducted to demonstrate the effectiveness of the designed resource management scheme under the proposed AVNET architecture over the other two benchmark schemes. Some important open research issues related to the proposed architecture are also discussed.

References

[1] S. Sabau *et al.*, "Optimal Distributed Control for Platooning via Sparse Coprime Factorizations," *IEEE Trans. Auto. Control*, vol. 62, no. 1, Jan. 2017, pp. 305–20.

[2] X. Cheng, R. Zhang, and L. Yang, "Wireless Towards the Era of Intelligent Vehicles," *IEEE Internet Things J.*, vol. 6, no. 1, Feb. 2019, pp. 188–202.

[3] X. Cheng *et al.*, "5G-Enabled Cooperative Intelligent Vehicular (5GenCiv) Framework: When Benz Meets Marconi," *IEEE Intell. Syst.*, vol. 32, no. 3, May 2017, pp. 53–59.

[4] C. Huang *et al.*, "V2V Data Offloading for Cellular Network Based on the Software Defined Network (SDN) Inside Mobile Edge Computing (MEC) Architecture," *IEEE Access*, vol. 6, Mar. 2018, pp. 17 741–55.

[5] K. Abboud, H. A. Omar, and W. Zhuang, "Interworking of DSRC and Cellular Network Technologies for V2X Communications: A Survey," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, Dec. 2016, pp. 9457–70.

[6] Q. Yuan *et al.*, "Toward Efficient Content Delivery for Automated Driving Services: An Edge Computing Solution," *IEEE Network*, vol. 32, no. 1, Jan. 2018, pp. 80–86.

[7] Z. Su, Y. Hui, and T. Luan, "Distributed Task Allocation to Enable Collaborative Autonomous Driving with Network Softwarization," *IEEE JSAC*, vol. 36, no. 10, Oct. 2018, pp. 2175–89.

[8] "ETSI GR MEC 022 V2.1.1 multi-access edge computing (MEC); study on MEC support for V2X use cases," European Telecommunications Standards Institute, Tech. Rep., Sep. 2018; available: https://www.etsi.org/deliver/etsi gr/MEC/001099/022/02.01.01 60/grMEC022v020101p.pdf.

[9] A. Fortelle *et al.*, "Network of Automated Vehicles: The AutoNet 2030 Vision," *Proc. ITS World Congr.*, 2014; available: https://hal-mines-paristech.archives-ouvertes.fr/hal-01063484/document.

[10] H. Peng *et al.*, "Vehicular Communications: A Network Layer Perspective," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, Feb. 2019, pp. 1064–78.

[11] H. Peng *et al.*, "Resource Allocation for Cellular-Based Inter-Vehicle Communications in Autonomous Multiplatoons," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, Dec. 2017, pp. 11 249–63.

[12] Q. Ye *et al.*, "Dynamic Radio Resource Slicing for a Two-Tier Heterogeneous Wireless Network," *IEEE Trans. Veh. Tech.*, vol. 67, no. 10, Oct. 2018, pp. 9896–9910.

[13] "ETSI GR MEC 017 V1.1.1 mobile edge computing (MEC); deployment of mobile edge computing in an NFV environment," European Telecommunications Standards Institute, Tech. Rep., Feb. 2018; available: https://www.etsi.org/deliver/etsi gr/MEC/001099/017/01.01.01 60/grMEC017v010101p.pdf.

[14] ETSI GS NFV-EVE 005 V1.1.1 (2015-12), European Telecommunications Standards Institute Std. gs nfv-eve 005, Dec. 2015; available: https://www.etsi.org/deliver/etsigs/NFV-EVE/001099/005/01.01.0160/gsNFV-EVE005v010101p.pdf.

[15] C. Liang *et al.*, "Virtual Resource Allocation in Information-Centric Wireless Networks with Virtualization," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, Dec. 2016, pp. 9902–14.

Biographies

Haixia Peng [S'15] received her M.S. and Ph.D. degrees in electronics and communication engineering and Computer Science from Northeastern University, Shenyang, China, in 2013 and 2017, respectively. She is currently a Ph.D. student in the Department of Electrical and Computer Engineering at the University of Waterloo, Canada. Her current research focuses on reinforcement learning and optimization based resource management in autonomous vehicular network. She served as a TPC member for IEEE VTC-Fall 2016 and 2017, IEEE ICCEREC 2018, IEEE Globecom 2016, 2017 and 2018, and IEEE ICC 2017, 2018, and 2019.

Qiang Ye [S'16, M'17] received his Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2016. He is currently a research associate with the Department of Electrical and Computer Engineering, University of Waterloo, where he had been a post-doctoral fellow from December 2016 to November 2018. His current research interests include AI and machine learning for future networking, IoT, SDN and NFV, network slicing for 5G networks, VNF chain embedding and end-to-end performance analysis.

Xuemin (Sherman) Shen [M'97, SM'02, F'09] received the Ph.D. degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. His research focuses on resource management in interconnected wireless/wired networks, wireless network security, social networks, smart grid, and vehicular ad hoc and sensor networks. He is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. He received the R.A. Fessenden Award in 2019 from IEEE Canada; the James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society; and the Joseph LoCicero Award in 2015 and the Education Award in 2017 from the IEEE Communications Society. He has also received the Excellent Graduate Supervision Award in 2006 and the Outstanding Performance Award five times from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/Co-Chair for IEEE Globecom'16, IEEE Infocom'14, IEEE VTC'10 Fall, IEEE Globecom'07, the Symposia Chair for IEEE ICC'10, the Tutorial Chair for IEEE VTC'11 Spring, the Chair for the IEEE Communications Society Technical Committee on Wireless Communications, and P2P Communications and Networking. He is the Editor-in-Chief of the *IEEE Internet of Things Journal* and the Vice President of Publications of the IEEE Communications Society.