

FAIR: Towards Impartial Resource Allocation for Intelligent Vehicles With Automotive Edge Computing

Haoxin Wang , *Member, IEEE*, Jiang Xie , *Fellow, IEEE*, and Muhana Magboul Ali Muslam , *Member, IEEE*

Abstract—The emerging vehicular connected applications, such as cooperative automated driving and intersection collision warning, show great potentials to improve the driving safety, where vehicles can share the data collected by a variety of on-board sensors with surrounding vehicles and roadside infrastructures. Transmitting and processing this huge amount of sensory data introduces new challenges for automotive edge computing with traditional wireless communication networks. In this work, we address the problem of traditional asymmetrical network resource allocation for uplink and downlink connections that can significantly degrade the performance of vehicular connected applications. An end-to-end automotive edge networking system, *FAIR*, is proposed to provide fast, scalable, and impartial connected services for intelligent vehicles with edge computing, which can be applied to any traffic scenes and road topology. The core of *FAIR* is our proposed symmetrical network resource allocation algorithm deployed at edge servers and service adaptation algorithm equipped on intelligent vehicles. Extensive simulations are conducted to validate our proposed *FAIR* by leveraging real-world traffic dataset. Simulation results demonstrate that *FAIR* outperforms existing solutions in a variety of traffic scenes and road topology.

Index Terms—Connected and automated vehicles, edge computing, intelligent driving.

I. INTRODUCTION

LESS than half a decade, the expeditious evolution of wireless communications, Artificial Intelligence (AI), and high performance computing (HPC), has been reinventing the mobility concept and systems. For example, in 2019, Toyota announced a profound transformation from being an automaker to

become a mobility company, with an emphasis on Connectivity, Autonomous driving, Shared mobility, and Electrification of vehicles (CASE) [1]. Meanwhile, other major automotive original equipment manufacturers (OEMs), including Volkswagen, Audi, BMW, etc., are investing heavily in future mobility solutions to enhance their core competencies and to ingratiate themselves with their customers. In addition, a variety of cutting-edge technologies and applications have been proposed to facilitate CASE and next-generation intelligent vehicles, such as mobility digital twins (MDT) [2], automotive edge computing [3], [4], [5], and vehicle-to-everything (V2X) communications with 5G [6], [7], [8], [9].

The mobility digital twin is an emerging implementation of Digital Twins (DT) in the transportation domain and is one of the killer applications of the CASE. The MDT aims to create a perpetual digital replica of a human driver or a connected and automated vehicle (CAV) in the digital world based on the data acquired in the physical world. It has been attracting tremendous attention from both industry and academia. According to a market research report, the global DT market size was valued at USD 7.48 billion in 2021 and is projected to grow at a compound annual growth rate of 39.1% from 2022 to 2030, where the automotive and transportation industry accounted for the largest revenue share of more than 19.0% of the overall revenue [10]. Very recently, a few MDT research have investigated and demonstrated how the MDT can facilitate next-generation CAVs and potential connected services, including cooperative automated driving [11], [12], [13], [14], personalized adaptive cruise control (P-ACC) [15], [16], driving behavior modelling and prediction [17], [18], [19], and 3D virtual environment for autonomous driving testing [20]. One of the shared attributes among all these MDT-assisted connected services is the requirement of continuous physical sensory data collection, modelling, and data processing. For example, 4 terabytes of sensory data will be gathered in two hours from a CAV's on-board camera, LiDAR, and radar sensors [21], [22]. The volume of the data collected by perception sensors on CAVs would be extremely huge even in a scale of ten vehicles, which requires a large amount of computation resource. Therefore, it is imperative to investigate how to efficiently and fast collect and process these sensory data.

One of the promising solutions to this challenge is automotive edge computing, which brings the computing and

Manuscript received 3 December 2022; revised 20 December 2022; accepted 24 December 2022. Date of publication 6 January 2023; date of current version 20 March 2023. This work was supported in part by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University under Research Partnership Program under Grant No. RP-21-07-05 and in part by the U.S. National Science Foundation under Grants No. 1910667, 1910891, and 2025284. (Corresponding author: Haoxin Wang.)

Haoxin Wang is with the Department of Computer Science, Georgia State University, Atlanta, GA 30303 USA (e-mail: haoxinwang@gsu.edu).

Jiang Xie is with the Department of Electrical and Computer Engineering, The University of North Carolina at Charlotte, Charlotte, NC 28223 USA (e-mail: linda.xie@uncc.edu).

Muhana Magboul Ali Muslam is with the Department of Information Technology, Imam Mohammad Ibn Saud Islamic University, Riyadh 11432, Saudi Arabia (e-mail: mmmuslam@imamu.edu.sa).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TIV.2023.3234888>.

Digital Object Identifier 10.1109/TIV.2023.3234888

storage resources at the location where the data is generated. Instead of processing these huge amount of sensory data on vehicles or sending to a remote cloud server, the data can be transmitted to and processed on an edge server that is in close proximity to vehicles. For example, an edge server can be deployed with a roadside unit (RSU), and vehicles can offload their sensory data directly to the edge server in real time through wireless communication technologies. In addition, the edge server can provide the capability of real-time data processing and data caching for those emerging MDT-assisted connected services.

Motivation: Although compared to processing sensory data locally on a vehicle or transmitting the data to a remote cloud server, edge-assisted solutions can enhance the efficiency and reduce the end-to-end latency (i.e., starting from the data collection to receiving the processing result), it still remains challenges of providing reliable, scalable, and resilient edge computing networks for vehicles with dynamics in data offloadings. Furthermore, the quality of service (QoS) of vehicular connected services in edge computing networks cannot always be guaranteed due to the legacy design of radio resource allocation in existing wireless networks.

Traditionally, wireless networks allocate more radio resources to downlink, since downlink traffic volume is usually much higher than uplink. For example, downloading an entertainment video from the YouTube server consumes much more bandwidth than uploading a vehicle's GPS information for the navigation purpose. Therefore, downlink acquires higher throughput, peak data rate, and spectral efficiency than uplink. However, MDT-assisted connected services may require a higher throughput than the vehicular downlink traffic. For instance, a CAV with cooperative automated driving needs to continuously offload its on-board camera captured image frames to an edge server for further processing in real time, while the edge server will send back the processing results (downlink traffic) to the CAV. The uplink traffic will be massive and latency-sensitive compared to the downlink traffic, and thus requires much more network resource than the downlink traffic. *Such new traffic distribution creates unique challenges for MDT-assisted connected services in automotive edge computing and requires the radio resource allocation issue in wireless networks to be revisited and re-designed.*

Our contributions: In this paper, we study these research challenges in automotive edge computing networks and design impartial resource allocation algorithms for MDT-assisted connected services. The novel contributions of this work are summarized as follows:

- 1) This work is, to the best of our knowledge, the first to systematically address the asymmetrical network resource allocation for uplink and downlink connections in automotive edge computing networks.
- 2) We implement a physical network testbed to conduct preliminary experimental study and to demonstrate the resource allocation issues in traditional wireless communication networks.
- 3) We propose and design an end-to-end automotive edge networking system, *FAIR*, which provides fast, scalable,

and impartial connected services for intelligent vehicles with edge computing in any traffic scenes.

- 4) We conduct extensive data-driven simulations to validate our proposed method based on the real world traffic data with different road topology and traffic scenes.

The remainder of this paper is organized as follows. Section II introduces our preliminary study and the observations from the experiments. Section III presents the analytical factors in edge-assisted intelligent vehicle systems and the details of the proposed *FAIR* that consists of a symmetrical resource allocation algorithm and a service adaption algorithm. Section IV evaluates the performance of *FAIR* through extensive simulations with real-world traffic data. Finally, the conclusion is drawn in Section V.

II. PRELIMINARY EXPERIMENTS

Conventional wireless communication networks, for example, the downlink in Wi-Fi and Long-Term Evolution (LTE) networks acquire a higher bandwidth, peak data rate, throughput, and spectral efficiency than their uplink [23], [24]. However, most current edge-assisted applications and services, such as high definition (HD) map generation [25] and mobile augmented reality [26] are no longer downlink traffic dominant only. Future MDT-assisted connected services are prone to have more uplink traffic. Therefore, traditional network resource allocation designs that favor the downlink traffic are no longer suitable for supporting QoS/QoE in automotive edge computing networks. In this section, we conduct experimental measurement studies to explore the wireless network performance changes while heavy uplink and downlink traffic co-exist, and demonstrate the unfairness in traditional IEEE 802.11 wireless networks.

Experimental study: In this paper, we mainly focus on IEEE 802.11 wireless technology, since compared with LTE, IEEE 802.11 is a better communication approach for data transmission between connected vehicles and edge servers, due to its higher data rate and lower monetary cost. In addition, since both heavy uplink traffic (e.g., offloaded sensing data) and downlink traffic (e.g., downloaded entertainment videos) may co-exist in connected vehicle networks, we conduct an experimental measurement study to explore the network performance changes while heavy uplink and downlink traffic co-existing. Our testbed is deployed as shown in Fig. 1. There are one edge server and two emulated connected vehicles in our testbed. The edge server consists of a computation unit and a communication unit. The computation unit is implemented on a HP Z820 workstation that is equipped with a NVIDIA QUADRO P2000 GPU. The communication unit is a LINKSYS EA4500 access point (AP), which is directly connected to the computation unit through a Ethernet cable. In order to eliminate the impact of different wireless network cards on the experiment results, two emulated connected vehicles are equipped with the same TPLINK-WN781ND wireless network cards. In our experiments, connected vehicle A downloads entertainment videos from the edge server, while connected vehicle B offloads its real-time camera captured image frames to the edge server for conducting



Fig. 1. Overview of the testbed implementation for our preliminary experiments. It consists of three components: an edge server and two connected vehicles. The edge server is built upon an HP Z820 workstation equipped with a NVIDIA QUADRO P2000 GPU and a LINKSYS EA4500 AP which is directly connected to the workstation through an Ethernet cable. To eliminate the impact of different wireless network cards on the experiment results, two emulated connected vehicles are deployed with the same TPLINK-WN781ND wireless network cards. Connected vehicle A downloads entertainment video frames from the edge server, while connected vehicle B offloads its camera image frames to the edge server for conducting object detection.

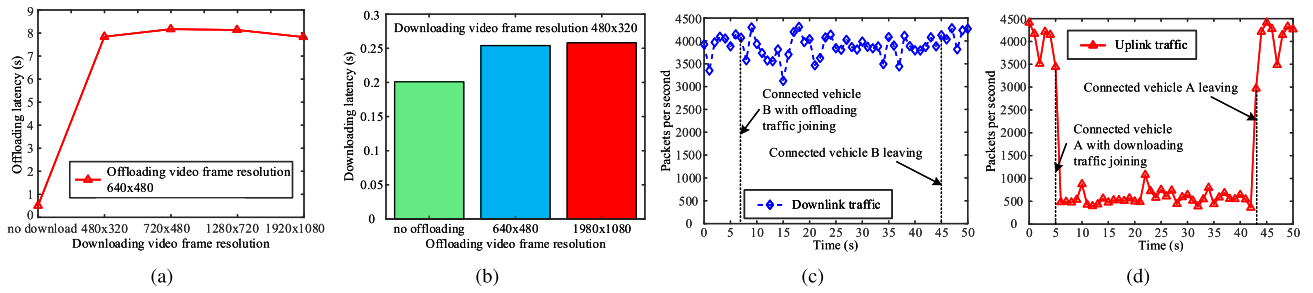


Fig. 2. Experimental measurement results. (a) Average offloading latency per frame; (b) Average downloading latency per frame; (c) The number of downlink packets; and (d) The number of uplink packets.

object detection, which is one of the most important steps of edge-assisted intelligent driving [27]. The object analytics module in the edge server is designed based on the YOLOv3 framework [28].

Our observations: We measure the latency of downloading/offloading each video frames, and calculate the average latency. Our observations are described as follows:

- As shown in Fig. 2(a), the average offloading latency per frame is dramatically increased, approximately 1600%, after connected vehicle A with downloading traffic joining the network.
- As shown in Fig. 2(b), the average downloading latency per frame is only increased to approximately 125% after connected vehicle B with offloading traffic joining the network, which is much less than the increase of the average offloading latency.
- As shown in Fig. 2(a), when the downloading video frame resolution increases (i.e., increasing the total downloading data size), the increment of the average offloading latency per frame does not change much. The same observation is obtained in the downlink scenario, as shown in Fig. 2(b).

Then, we conduct another test to explore the reasons of the aforementioned observations. We measure the traffic of downlink and uplink, as depicted in Fig. 2(c) and (d), respectively. In Fig. 2(d), we find that a big uplink throughput drop occurs from 5 sec to 43 sec, when connected vehicle A with downloading traffic joins the network. However, there is no obvious downlink throughput degrade in Fig. 2(c), when connected vehicle B with

uplink traffic joins the network. This is because usually, AP is set up with a shorter Arbitration Inter-Frame Space Number (AIFSN) value for voice and video traffic (e.g., AIFSN = 1) than its associated stations (e.g., AIFSN = 2) [29], [30], [31]. Thus, an AP obtains a higher priority for channel contention than its associated stations (i.e., the downlink traffic has a higher priority than the uplink traffic during contention), which significantly impacts the transmission efficiency of connected vehicle B offloading its camera captured video frames. Although, we may set the same AIFS value for both AP and stations in order to eliminate above unfairness, the more complicated and fierce channel contention may still badly influence on the throughput of both downlink and uplink.

Summary: MDT-assisted connected services with edge computing will suffer from a significant performance degradation due to the network resource allocation unfairness between downlink and uplink in traditional wireless communication networks. It is imperative to systematically investigate the impartial network resource allocation in automotive edge computing networks towards fast and reliable MDT-assisted connected services.

III. PROPOSED FAIR AUTOMOTIVE EDGE NETWORKING SYSTEM

Based on the above observations, in this section, we propose FAIR, an end-to-end automotive edge networking system, which can provide fast, scalable, and impartial connected services for

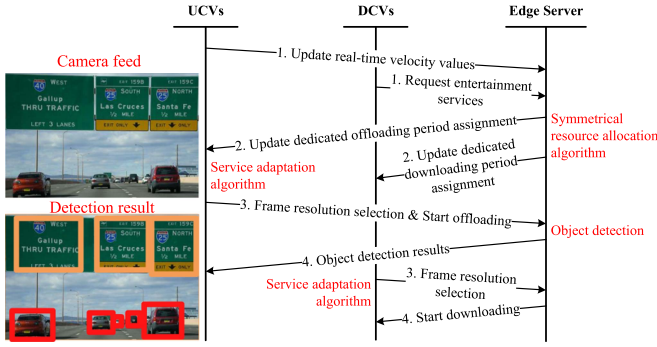


Fig. 3. Overview of the proposed *FAIR*. The core of *FAIR* is our proposed symmetrical network resource allocation algorithm deployed at the edge server and proposed service adaptation algorithm equipped on intelligent vehicles. The symmetrical network resource allocation algorithm can mitigate the service performance degradation incurred by the asymmetrical radio resource allocation for uplink and downlink connections. The service adaptation algorithm can dynamically adjust configurations of environmental sensing and frame resolution according to the reserved service period and user preference.

TABLE I
TABLE OF NOTATION

Variable	Description
V	Speed of a connected vehicle (m/s)
fps	Frame rate (frame/second)
ΔS	The area of camera captured different surroundings (m^2)
$k \cdot s$	Image frame resolution (pixels)
γ	The number of bits carried by one pixel (bits)
R	Average wireless data rate
LU	Data offloading latency of an image frame (second)
LD	Data offloading latency of a video frame (second)
EU	Per frame energy consumption for offloading (J)
ED	Per frame energy consumption for downloading (J)
UU	Utilization of the assigned <i>DOP</i>
UD	Utilization of the assigned <i>DDP</i>
EL	Life-time energy consumption (J)
P_{tr}	Average power consumption of transmission phase (W)
P_{rev}	Average power consumption of receiving data (W)
\mathcal{N}	The set of all intelligent vehicles
\mathcal{N}_d	The set of all DCVs
\mathcal{N}_u	The set of all UCVs
θ	Horizontal field of view of a camera ($^\circ$)
l	3-D measurement range of a camera (m)
σ	Ratio of the areas of UCVs' captured different surroundings
ω	Positive weight parameter customizing user preference

intelligent vehicles in a variety of traffic scenes. Fig. 3 illustrates the overview of the proposed *FAIR*. To mitigate the service performance degradation incurred by the asymmetrical radio resource allocation for uplink and downlink connections, we design a new symmetrical network resource allocation algorithm deployed at edge servers to proactively and impartially reserve service periods for individual intelligent vehicles that request dissimilar connected services. In addition, a service adaptation algorithm deployed on intelligent vehicles is proposed to dynamically adjust configurations of environmental sensing and frame resolution according to the reserved service period and user preference.

A. Key Factors in Edge-Assisted Intelligent Vehicle

We consider an end-to-end automotive edge computing system consists of a roadside edge server a and N intelligent vehicles that request connected services. Denote N_d as the

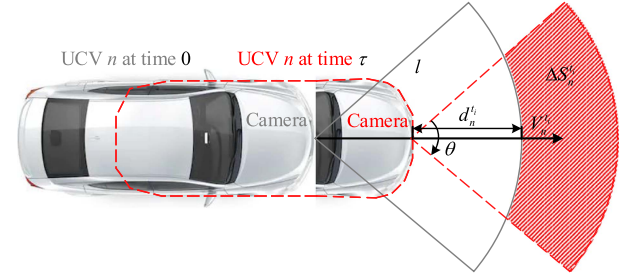


Fig. 4. The area change of camera captured image frames while moving.

number of intelligent vehicles requesting connected services that downloading traffic are dominant, e.g., streaming entertainment videos. While N_u denotes the number of intelligent vehicles requesting connected services that uploading traffic are dominant, e.g., offloading on-board camera captured images to the edge server for further processing). In this paper, the intelligent vehicles with downloading and uploading traffic are named as DCVs and UCVs, respectively. Furthermore, to make our system practical, vehicles are capable of requesting both downloading traffic dominant and uploading traffic dominant services concurrently. Thus, $N_d + N_u \geq N$. Denote \mathcal{N} , \mathcal{N}_d , and \mathcal{N}_u as the set of all intelligent vehicles served by the edge server, DCVs, and UCVs, respectively.

To thoroughly study the correlations between the performance of connected services and the state of vehicles, such as speed and user preference, we define four analytic factors for intelligent vehicles with automotive edge computing.

1) *Velocity of Intelligent Vehicles*: Denote V_n^t as the real-time velocity of a UCV at time t , where $n \in \mathcal{N}_u$. Assume each intelligent vehicle is equipped with cameras with l meter 3-D measurement range and θ° horizontal field of view. The area of the surrounding environment that the UCV's camera can capture will change while the UCV is moving a time period τ . As illustrated in Fig. 4, we model the area of the changed surrounding at time t as $\Delta S_n^t = (2l - \theta \cdot d_n^t) \cdot V_n^t$, which also describes the area of new captured surroundings by the on-board camera is determined by the real-time velocity of the vehicle. Therefore, based on the above analysis, we define that

$$\sigma_{(n,m)}^t = \frac{\Delta S_n^t}{\Delta S_m^t} = \frac{(2l - \theta \cdot d_n^t) \cdot V_n^t}{(2l - \theta \cdot d_m^t) \cdot V_m^t}, n, m \in \mathcal{N}_u, \quad (1)$$

where, $d_n^t = V_n^t \cdot \tau$, $d_m^t = V_m^t \cdot \tau$.

2) *Frame Rate*: In this paper, the frame rate, fps , is defined as the number of image frames that a DCV downloads for in-vehicle entertainment or a UCV offloads for further processing such as object detection per second. **For UCVs**. When the velocity of a UCV is high, the UCV has to offload more camera captured frames to the edge server for a reliable and safe driving assistance since it has a larger ΔS_n^t . Assume the minimum acceptable offloading frame rates for UCVs n and m , $n, m \in \mathcal{N}_u$, are fps_n^t and fps_m^t , respectively, we can define the correlation between fps_n^t and fps_m^t as

$$fps_m^t = \frac{1}{\sigma_{(n,m)}^t} \cdot fps_n^t. \quad (2)$$

For DCVs: A more smooth entertainment video experience requires a higher downloading frame rate, which may also requires a higher data rate.

3) *Frame Resolution:* Frame resolution is crucial for the performance of both entertainment service and edge-assisted driving assistance. **For UCVs.** In this paper, we use $k_n^{f_u} \cdot s_n^{f_u}$ (pixels) to represent the frame resolution of the f_u th image captured by the vehicle's on-board camera. Denote γ as the color depth of a frame, i.e., the number of bits required to represent the information carried by one pixel. The data size of the camera captured image f_u at time t is calculated as $k_n^{f_u} \cdot s_n^{f_u} \cdot \gamma$ bits. We model the offloading latency of the frame f_u as

$$LU_n^{f_u} = \frac{k_n^{f_u} \cdot s_n^{f_u} \cdot \gamma}{R_n^t}, \quad (3)$$

where R_n^t is the average transmission data rate of UCV n at time t . Thus, the image offloading latency is a function of wireless data rate and image resolution. The image resolution can be adapted to meet the latency requirement under wireless network dynamics. For example, the UCV may proactively degrade its camera image resolution when the wireless data rate is low. However, the connected service performance, e.g., object detection accuracy, may be affected by the image resolution as well. A lower image resolution usually leads to a worse mean average precision of an object detection [32]. Therefore, it is critical for UCVs with high velocity to keep a high camera image resolution for a better service performance and a safer driving. Furthermore, as the data size of object detection results is usually small, we do not consider the latency caused by returning the results in this work.

For DCVs: Consider a DCV $g \in \mathcal{N}_d$ that request downloading entertainment videos while moving. Let the frame resolution of the requested video at time t is $k_g^{f_d} \cdot s_g^{f_d}$ pixels. We model the downloading latency as

$$LD_g^{f_d} = \frac{k_g^{f_d} \cdot s_g^{f_d} \cdot \gamma}{R_g^t}, \quad (4)$$

where R_g^t is the wireless data rate of DCV g at time t .

4) *Energy Consumption of Connected Vehicles:* **For UCVs:** To estimate the energy efficiency of performing the edge-assisted intelligent driving, in particular, camera sensing and image offloading, we propose *per frame energy consumption*, a new performance metric that appraises the total amount of energy consumed in a connected vehicle by performing the edge-assisted intelligent driving on one image frame. In this work, the per frame energy consumption of image frame f_u in UCV $n \in \mathcal{N}_u$ can be defined as

$$EU_n^{f_u} = EU_{cam}^{(f_u, n)} + EU_{tr}^{(f_u, n)}, \quad (5)$$

where EU_{cam} and EU_{tr} denote the energy consumed by sampling an image and transmitting the image frame to a edge server, respectively. Image sampling is the process of transferring the vehicle's camera sensed continuous light signal to a processable digital image frame. EU_{cam} is a function of image resolution $k \cdot s$, e.g., a larger $k \cdot s$ will result in a higher EU_{cam} . Furthermore, since the energy consumed by image sampling contributes

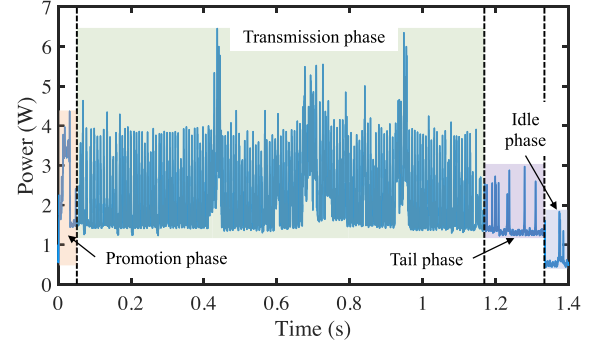


Fig. 5. Power consumption of promotion, transmission, tail, and idle phases.

the largest portion of the per frame energy consumption [33], the camera sampling frequency should be adaptable to improve the energy efficiency. Therefore, we propose to co-adapt the image sampling and offloading frequency.

During the transmission of a single image frame in wireless communication networks, e.g., Wi-Fi, the vehicle's wireless interface undergoes four phases: promotion, data transmission, tail, and idle. As depicted in Fig. 5 (power consumption measured in our testbed), the wireless interface first steps into the promotion phase once a transmission request is initiated. Then, the image data is sent to the edge server when the data transmission phase is invoked. The completion of transmission triggers the tail phase, where the wireless interface is compelled to stay active for a fixed duration and to wait for either a new transmission request or service feedback from the edge server. The wireless interface finally enters the idle phase to save energy if there is no new request initiated or feedback received. We model the energy consumption of transmitting image frame f_u in UCV $n \in \mathcal{N}_u$ as

$$EU_{tr}^{(f_u, n)} = e_{pro} + P_{tr}^{(n, f_u)} \cdot LU_n^{f_u} + e_{tail}, \quad (6)$$

where $P_{tr}^{(n, f_u)}$ is the average power consumption of the data transmission phase; e_{pro} and e_{tail} are the energy consumed in promotion and tail phases, respectively. e_{pro} and e_{tail} are constant.

For DCVs: The per frame energy consumption of downloading video frame f_d in DCV $g \in \mathcal{N}_d$ can be modeled as

$$ED_g^{f_d} = P_{rev}^{(g, f_d)} \cdot LD_g^{f_d}, \quad (7)$$

where $P_{rev}^{(g, f_d)}$ is the average power consumption of receiving data.

In addition to the per frame energy consumption, we propose a new performance metric, *life-time energy consumption*, to estimate the total amount of energy consumed by performing uploading and downloading connected services for a certain amount of time ΔT , which is defined as

$$EL(\Delta T) = \sum f_u EU + \sum f_d ED. \quad (8)$$

B. Symmetrical Resource Allocation Algorithm

As discussed in Section III-A, to assure a reliable and safe edge-assisted driving assistance, the amount of offloading camera image data should be proportional to the velocity of the UCV. In this work, the core of our proposed end-to-end automotive edge networking system, *FAIR*, is that *the edge server proactively and periodically reserves a dedicated offloading period (DOP) for each served UCV based on its real-time velocity*. In particular, the value of the reserved DOP at time t for UCV n is with respect to the real-time velocity V_n^t . Thus, we define DOP as $DOP_n = \psi(V_n^t)$, where a larger DOP will be assigned to the UCV with a higher velocity for supporting an edge-assisted driving assistance with high frame rate and resolution. Furthermore, to make the reserved DOPs can be adapted to the environment dynamics, including the varying velocity and position of intelligent vehicles, the edge server will update the DOP reservation once every T s. After completing the DOP reservation for all UCVs, the edge server will allocate a dedicated downloading period (DDP) for each DCVs. Since the required data volume of the in-vehicle entertainment is usually irrelevant to vehicle's velocity, we design that the length of the DDP assigned for each DCV is the same, where

$$DDP_g = \frac{T - \sum_{n \in \mathcal{N}_u} (DOP_n)}{N_d}, g \in \mathcal{N}_d. \quad (9)$$

We assume the velocity of each intelligent vehicle is constant in T s, and define the velocity vector of UCVs as $\vec{v} = \{v_1^x, \dots, v_{N_u}^y\}$, a N_u -tuple containing the real-time velocity of each UCV within T s. UCVs in the velocity vector are sorted in descending order in terms of the value of velocity. Denote \vec{DOP} as the vector of edge reserved DOPs for UCVs in \mathcal{N}_u within T s. The edge server will reserve the DOP for each UCV every T s by leveraging the updated \vec{v} . In our proposed symmetrical wireless resource allocation algorithm, the UCV with the highest velocity will be always reserved with the highest priority. In particular, the UCV with the highest velocity in T obtains the best edge networking resource including the largest reserved DOP, where $DOP = DOP_{\max}$; the highest camera frame resolution, where $k \cdot s = k_{\max} \cdot s_{\max}$, with an acceptable frame rate fps_0 . The DOP_{\max} is defined as

$$DOP_{\max} = DOP_x = \frac{k_{\max} \cdot s_{\max} \cdot \gamma}{R_n^t} \cdot fps_0 \cdot T. \quad (10)$$

Furthermore, DOP of the rest of UCVs will be assigned in sequence according to \vec{v} . The reserved DOP of UCV n is defined as

$$DOP_n = \psi(V_n) = \frac{1}{\sigma_{(x,n)}^t} \cdot DOP_{\max}, n \in \mathcal{N}_u. \quad (11)$$

Based on the designed principle of service allocation in edge networking systems, we can see that a UCV with a higher real-time velocity can always be assured of a more reliable connected service by reserving a larger DOP. In addition, we classify three cases to describe special traffic scenarios, defined as follows.

- 1) Highway. We set $V_{highway}$ as the trigger to determine if the traffic scene is driving on a highway. For example, in the United States of America, the value of $V_{highway}$ is

Algorithm 1: FAIR Symmetrical Resource Allocation Algorithm.

Input: Vector of UCV speed: \vec{v} ; the number of UCVs in cases 1, 2, 3: $\kappa_1, \kappa_2, \kappa_3$; data rate: R ; total number of UCVs and DCVs: N_u, N_d ; period of resource allocation: T ; acceptable frame rate: fps_0 ; highest camera frame resolution: $k_{\max} \cdot s_{\max}$.

Output: Sets of RSU server allocated DOPs and DDPs: \vec{DOP} and \vec{DDP} .

```

1  $T_0 \leftarrow T$ ;
2 if  $\kappa_1 > 0$  then
3   Find  $R_{min}$ ; ▷ Case 1 triggered
4    $DOP_{\max} \leftarrow$  calculate via (12);
5    $T_0 \leftarrow T_0 - \sum DOP_{\max}$ ;
6 else
7    $DOP_{\max} \leftarrow$  calculate via (10);
8    $T_0 \leftarrow T_0 - DOP_{\max}$ ;
9 if  $\kappa_2 > 0$  then
10   $DOP_{n \in \mathcal{K}_2} \leftarrow$  calculate via (11); ▷ Case 2 triggered
11   $T_0 \leftarrow T_0 - \sum_{n \in \mathcal{K}_2} DOP_{n \in \mathcal{K}_2}$ ;
12 for  $n \in \mathcal{N}_u$  do
13    $DOP_n \leftarrow$  calculate via (11);
14   if  $\kappa_3 > 0$  then
15      $DOP_{n \in \mathcal{K}_3} \leftarrow$  calculate via (13); ▷ Case 3 triggered
16      $T_0 \leftarrow T_0 - \sum_{n \in \mathcal{K}_3} DOP_{n \in \mathcal{K}_3}$ ;
17   if  $T_0 \leq DOP_n$  then
18      $DOP_n \leftarrow T_0$ ;
19      $T_0 = 0$ ;
20     break;
21    $T_0 \leftarrow T_0 - DOP_n$ ;
22 if  $T_0 > 0$  then
23   for  $g = 1$  to  $N_d$  do
24      $DDP_g \leftarrow$  calculate via (9); ▷ Allocate DDP for DCVs
25  $\vec{DOP} \leftarrow DOP_n$ ;  $\vec{DDP} \leftarrow DDP_g$ ;
26 return  $\vec{DOP}$  and  $\vec{DDP}$ .
```

usually over 60 miles per hour or 26.8 m/s. To ensure the reliability of the service connection and the safety of driving, our proposed symmetrical service allocation algorithm assigns a DOP_{\max} for every UCV n that is classified as driving on highway (i.e., $n \in \mathcal{K}_1$). In this case, DOP_{\max} is defined as

$$DOP_{\max}^{n \in \mathcal{K}_1} = \frac{k_{\max} \cdot s_{\max} \cdot \gamma}{R_n^t} \cdot fps_0 \cdot T. \quad (12)$$

The assigned DOP_{\max} aims to assure each UCV is capable of offloading camera captured images with the highest frame resolution $k_{\max} \cdot s_{\max}$ and the acceptable frame rate fps_0 .

- 2) Continuous unallocated. UCVs that are not assigned for a DOP in the last $\beta \cdot T$ s. κ_2 and \mathcal{K}_2 represent the number of UCVs and the set of UCVs in case 2, respectively. UCVs in this case are provided with the highest DOP allocation priority except the UCVs in case 1.
- 3) Temporary stop. This case includes UCVs with a temporary stop incurred by a red traffic light, traffic congestion, or temporary parking. κ_3 and \mathcal{K}_3 represent the number of UCVs and the set of UCVs in case 3, respectively. UCVs with a temporary stop only need to opportunistically sense and transmit a camera image to substantiate whether the

preceding car's status has changed or not, or if the traffic signal light turns green. Therefore, the lowest priority is assigned to the UCVs in this case, where

$$DOP_{n \in \mathcal{K}_2} = \frac{k_{\min} \cdot s_{\min} \cdot \gamma}{R_n^t}. \quad (13)$$

In addition, DDP reserved for each DCV is defined by (9). Denote \vec{DDP} as the vector of edge reserved $DDPs$ for DCVs in \mathcal{N}_d within T s. Algorithm 1 presents the proposed symmetrical resource allocation algorithm in detail.

C. Service Adaptation Algorithm

Our proposed symmetrical network resource allocation algorithm proactively and impartially allocates a DOP or a DDP for each connected vehicle that initiates either uploading or downloading traffic. Then, connected vehicles will perform their request connected services through exploiting these allocated $DOPs$ and $DDPs$. Therefore, it is imperative to deploy a mechanism on connected vehicles that can dynamically adapt image sensing and offloading (e.g., image frame resolution) for UCVs or downloaded video quality for DCVs (e.g., video frame resolution). To achieve effective adaptations, we propose a service adaptation algorithm to coordinate with the symmetrical resource allocation algorithm.

As we proposed in the symmetrical resource allocation algorithm, *FAIR* ensures that the UCV with the highest instantaneous speed or UCVs with instantaneous speed over $V_{highway}$ are reserved with $DOPs$ that allow the camera image sampling and transmission with the highest frame resolution $k_{\max} \cdot s_{\max}$. While the image frame resolution of other UCVs are selected in the range of $k_{\min} \cdot s_{\min}$ to $k_{\max} \cdot s_{\max}$ based on the optimization problem \mathcal{P}_0 , which aims to achieve an optimal image frame resolution $k^{fu} \cdot s^{fu}$ to balance the offloading energy consumption and the utilization of the assigned DOP . \mathcal{P}_0 is formulated as follows.

$$\begin{aligned} \mathcal{P}_0: \quad & \min_{\{k^{fu}, s^{fu}\}} Q_u = \omega_1 \cdot EU^{fu} - \omega_2 \cdot UU^{fu} \\ \text{s.t.} \quad & C_1: k^{fu} \cdot s^{fu} \in \{k_{\min} \cdot s_{\min}, \dots, k_{\max} \cdot s_{\max}\}; \\ & C_2: UU^{fu} \leq 1; \end{aligned} \quad (14)$$

where $UU^{fu} = \frac{k^{fu} \cdot s^{fu} \cdot \gamma}{DOP \cdot R^t}$ describes the utilization of the assigned DOP ; and ω_1 and ω_2 are two positive weight parameters that are introduced to customize the user preference or to discriminate the configuration of different connected services. For example, given a larger ω_1 and a smaller ω_2 , the frame resolution selection strategy will tend to be energy saving, and vice versa. ω_1 and ω_2 can be specified by either the user or connected service provider. Similarly, frame resolution determination in DCVs is formulated and resolved by

$$\begin{aligned} \mathcal{P}_1: \quad & \min_{\{k^{fd}, s^{fd}\}} Q_d = \bar{\omega}_1 \cdot ED^{fd} - \bar{\omega}_2 \cdot UD^{fd} \\ \text{s.t.} \quad & C_1: k^{fd} \cdot s^{fd} \in \{k_{\min}^- \cdot s_{\min}^-, \dots, k_{\max}^- \cdot s_{\max}^-\}; \\ & C_2: UD^{fd} \leq 1; \end{aligned} \quad (15)$$

Algorithm 2: FAIR Service Adaptation Algorithm.

Input: DOP ; DDP ; DOP_{\max} ; user preference of UCVs: ω_1 , ω_2 ; user preference of DCVs: $\bar{\omega}_1$, $\bar{\omega}_2$; $k_{\max} \cdot s_{\max}$, $k_{\min} \cdot s_{\min}$; $k_{\max}^- \cdot s_{\max}^-$, $k_{\min}^- \cdot s_{\min}^-$; and R^t .
Output: Optimal camera frame resolution: $k^{fu} \cdot s^{fu}$ and optimal downloading video resolution: $k^{fd} \cdot s^{fd}$.

```

1 if UCV  $n \in \mathcal{N}_u$  then
2   if  $DOP = DOP_{\max}$  then
3      $k^{fu} \cdot s^{fu} \leftarrow k_{\max} \cdot s_{\max}$ ;
4   else
5      $k^{fu} \cdot s^{fu} \leftarrow$  solving  $\mathcal{P}_0$  with  $DOP$ ,  $\omega_1$ ,  $\omega_2$ ,
6      $k_{\max} \cdot s_{\max}$ ,  $k_{\min} \cdot s_{\min}$ , and  $R^t$ ;
7 else if DCV  $g \in \mathcal{N}_d$  then
8    $k^{fd} \cdot s^{fd} \leftarrow$  solving  $\mathcal{P}_1$  with  $DDP$ ,  $\bar{\omega}_1$ ,  $\bar{\omega}_2$ ,
9    $k_{\max}^- \cdot s_{\max}^-$ ,  $k_{\min}^- \cdot s_{\min}^-$ , and  $R^t$ ;
10 return  $k^{fu} \cdot s^{fu}$  and  $k^{fd} \cdot s^{fd}$ .
```

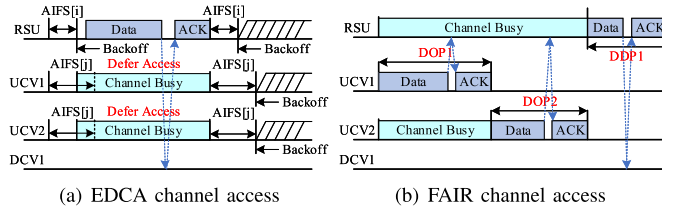


Fig. 6. Comparison of channel access. (a) EDCA channel access. (b) FAIR channel access.

where $UD^{fd} = \frac{k^{fd} \cdot s^{fd} \cdot \gamma}{DDP \cdot R^t}$ describes the utilization of the assigned DDP . Given the above discussion and the formulated problems \mathcal{P}_0 and \mathcal{P}_1 , we design the service adaptation algorithm, where the pseudo code is delineated in Algorithm 2.

In summary, our proposed *FAIR* automotive networking system not only assures the impartial and proactive resource allocation for both UCVs and DCVs within variant environmental conditions, but also enhances both the energy efficiency of performing connected services and network resource utilization compared to traditional wireless networks, IEEE 802.11 Enhanced Distributed Channel Access (EDCA), as illustrated in Fig. 6.

IV. PERFORMANCE EVALUATION

This section presents the performance evaluation of our proposed *FAIR* scheme through vehicle trajectory datasets collected in multiple real-world traffic scenarios and road topology. We first introduce the real-world dataset we used in the simulation. Then, we evaluate the average FPS, per frame energy consumption, channel utilization, and average obtained frame resolution of the proposed *FAIR* under variant configurations through data-driven simulations.

A. Trajectory Dataset

To validate our proposed *FAIR* can achieve network performance improvement for connected vehicles, we conduct extensive end-to-end simulations with real-world vehicle traces collected by drones [34], [35]. The datasets include multiple traffic scenes and road topology, e.g., roundabout, intersection, and

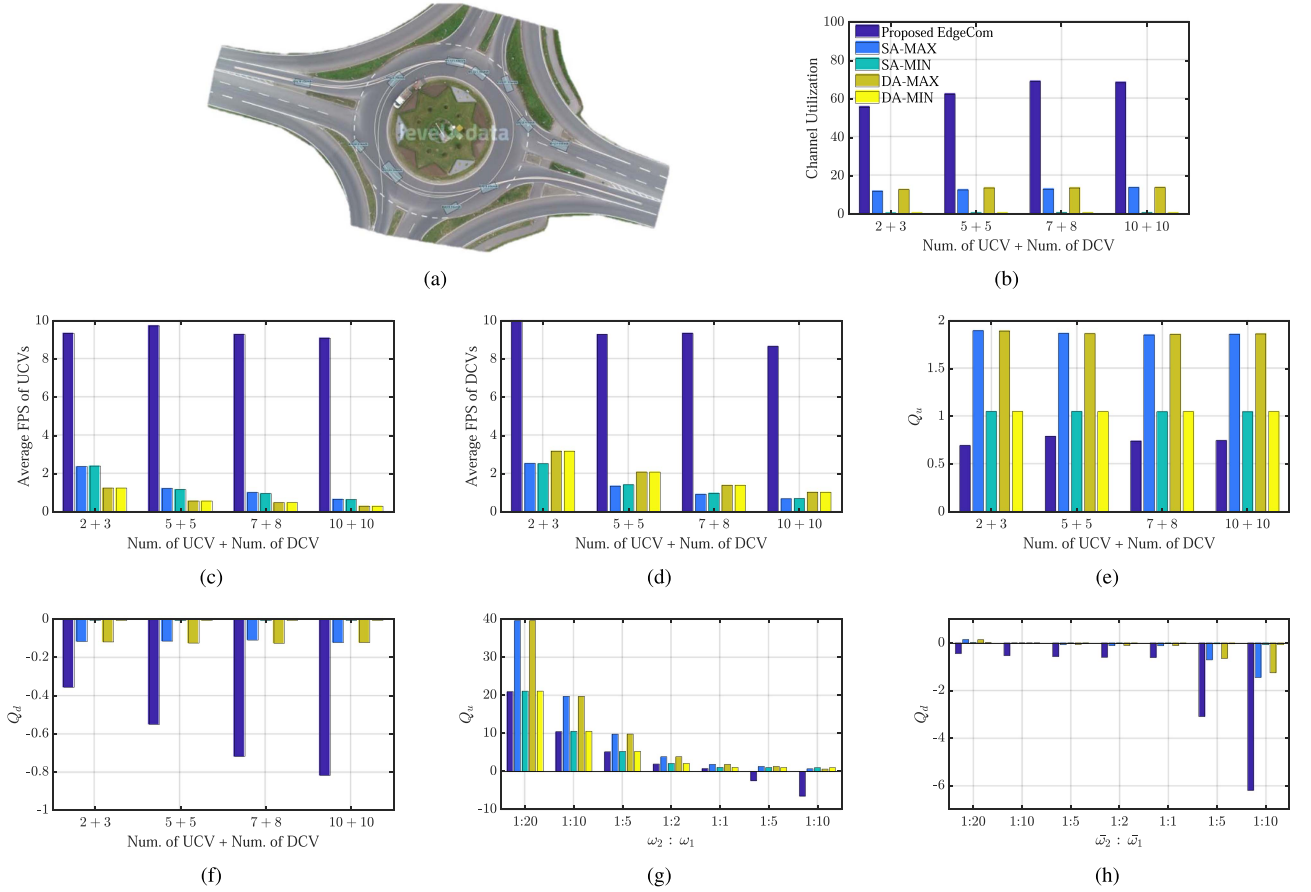


Fig. 7. Performance comparison between FAIR and baseline in the roundabout scenario. (a) Road topology and example vehicle trajectories at the roundabout; (b) Comparison of channel utilization with variant number of UCVs and DCVs; (c) and (d) Comparison of average frame rate with variant number of UCVs and DCVs; (e) and (f) Comparison of optimality with variant number of UCVs and DCVs; (g) and (h) Comparison of optimality with variant user preference.

straight road. The datasets consist of trajectories of over 20,000 road users, including cars, trucks, vans, buses, pedestrians, bicycles and motorcycles, where the trajectories are generated from the videos recorded by drones in 25 Hz at different locations. Since our work is designed for connected vehicles, we filter out pedestrians, bicycles, and motorcycles in the datasets.

B. Performance Evaluation of FAIR

We conduct simulations with one edge server and a number of connected vehicles that have either downlink services (i.e., downloading entertainment videos) or uplink services (i.e., intelligent driving), or both. As we present in Section III, a vehicle can obtain uplink and downlink traffic simultaneously. Furthermore, we utilize a path-loss model to simulate the network dynamics caused by the mobility of vehicles:

$$P_L = 20 \log_{10} f + 10n \log_{10} d - 24(\text{dB}), \quad (16)$$

where d is the distance between the edge server and a vehicle (UCV or DCV); P_L describes the radio frequency (RF) propagation path-loss, which is determined by d , f (i.e., carrier frequency), and n (i.e., path-loss exponent). n and f are set to 6 and 2400 MHz in our simulation environment. In addition,

TABLE II
DATA RATE TABLE OF 802.11N (4 SPATIAL STREAMS)

Min. SNR (dBm)	2	5	9	11	15	18	20	25
Data rate (Mbps)	29	58	87	116	173	231	260	289

the Signal-to-Noise Ratio (SNR) is introduced and calculated to estimate the real-time data transmission rate of each vehicle-to-server connection. The correlation between SNR and data rate is shown in Table II. $\theta = 50^\circ$, $l = 55$ m [36], $f_{ps0} = 10$ Hz [27], [37] are the configurations of the camera sensors equipped on each connected vehicle. The support image frame resolutions for offloading are $k \cdot s = \{640 \cdot 480, 480 \cdot 480, 320 \cdot 480, 320 \cdot 320, 224 \cdot 320, 224 \cdot 224, 128 \cdot 224, 128 \cdot 128\}$ pixels. The support video frame resolution for downloading are $\bar{k} \cdot \bar{s} = \{640 \cdot 480, 480 \cdot 480, 320 \cdot 480, 320 \cdot 320, 224 \cdot 320, 224 \cdot 224, 128 \cdot 224, 128 \cdot 128\}$ pixels. The number of bits carried by one pixel is set as 8. The default user preference are $\omega_1 = 1, \omega_2 = 1$, and $\bar{\omega}_1 = 1, \bar{\omega}_2 = 1$. The edge server is set to update the DOP reservation once every $T = 100$ ms. The speed threshold $V_{highway}$ in case 1 is set as 26.8 m/s. The trigger β of case 2, continuous unallocated, is set as 3. UCVs classified as case 3 will be allowed to offload every $10 \cdot T$.

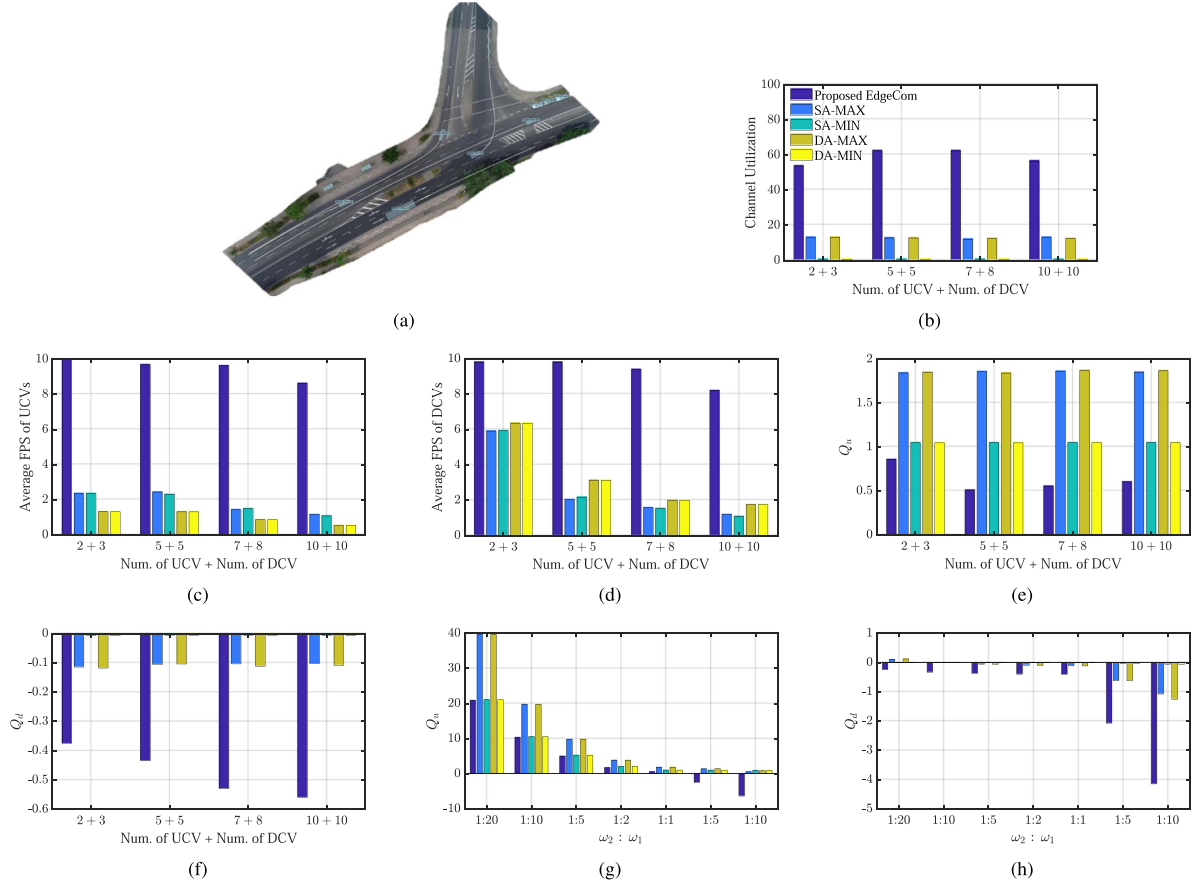


Fig. 8. Performance comparison between *FAIR* and baseline in the suburb intersection scenario. (a) Road topology and example vehicle trajectories at the intersection; (b) Comparison of channel utilization with variant number of UCVs and DCVs; (c) and (d) Comparison of average frame rate with variant number of UCVs and DCVs; (e) and (f) Comparison of optimality with variant number of UCVs and DCVs; (g) and (h) Comparison of optimality with variant user preference.

We compare our propose *FAIR* with the following algorithms:

- *SA + MAX*: The edge server and connected vehicles obtain the same AIFS. Connected vehicles adopt the highest frame resolution for their connected services to achieve maximum accuracy.
- *SA + MIN*: The vehicle adopts the lowest frame resolution for its connected services to achieve the minimum latency.
- *DA + MAX*: The edge server obtains a shorter AIFS than connected vehicles, i.e., different AIFS. The vehicle adopts the highest frame resolution for its connected services.
- *DA + MIN*: The vehicle adopts the lowest frame resolution for its connected services.

From system perspective: We evaluate the performance of *FAIR* at both roundabout and intersection. We aim to study: 1) what's the performance of *FAIR* compared to other solutions; 2) how does *FAIR* perform in different traffic scenes; 3) whether *FAIR* can effectively scale under the different number of connected vehicles. Figs. 7(a) and 8(a) illustrate the road topology of roundabout and intersection adopted in our simulations. In particular, the roundabout in Fig. 7(a) is four-armed and connects a highway with Aachen in Germany, where it obtains the highest traffic volume among all the recorded scenes in the datasets [34]. The average traffic speed is 7.41 m/s, and the speed of vehicles is in the range of [0, 17.72]m/s. The intersection in Fig. 8(a) is

located in suburban area of Aachen and is a T-junction, where the major arterial is straight and has the right of way, and there is a left turn lane into the side road [35]. The average traffic speed is 12.32 m/s, and the speed of vehicles is in the range of [0, 25.13]m/s.

Figs. 7(b) and 8(b) show the channel utilization performance of different algorithms under different number of UCVs and DCVs. We can see that our proposed *FAIR* outperforms other algorithms and always achieves the best channel utilization performance, which demonstrates our analysis in Section III-C. For example, in the roundabout scenario, *FAIR* improves 4.6 times channel utilization in average compared to SA-MAX and DA-MAX, which validates *FAIR* can effectively utilize the network resource and enhance the resource accessibility through proactive scheduling and avoiding reactive channel contention among vehicles.

Figs. 7(c), (d), 8(c), and (d) evaluate the average frame rate achieved by UCVs and DCVs at roundabout and intersection, respectively. We illustrate how *FAIR* performs with different number of UCVs and DCVs. We observe that *FAIR* always assures both UCVs and DCVs of obtaining a high frame rate (i.e., close to 10 Hz). In addition, compared to other algorithms, *FAIR* significantly increases the frame rate of both UCVs and DCVs. For instance, in the intersection scenario, it increases

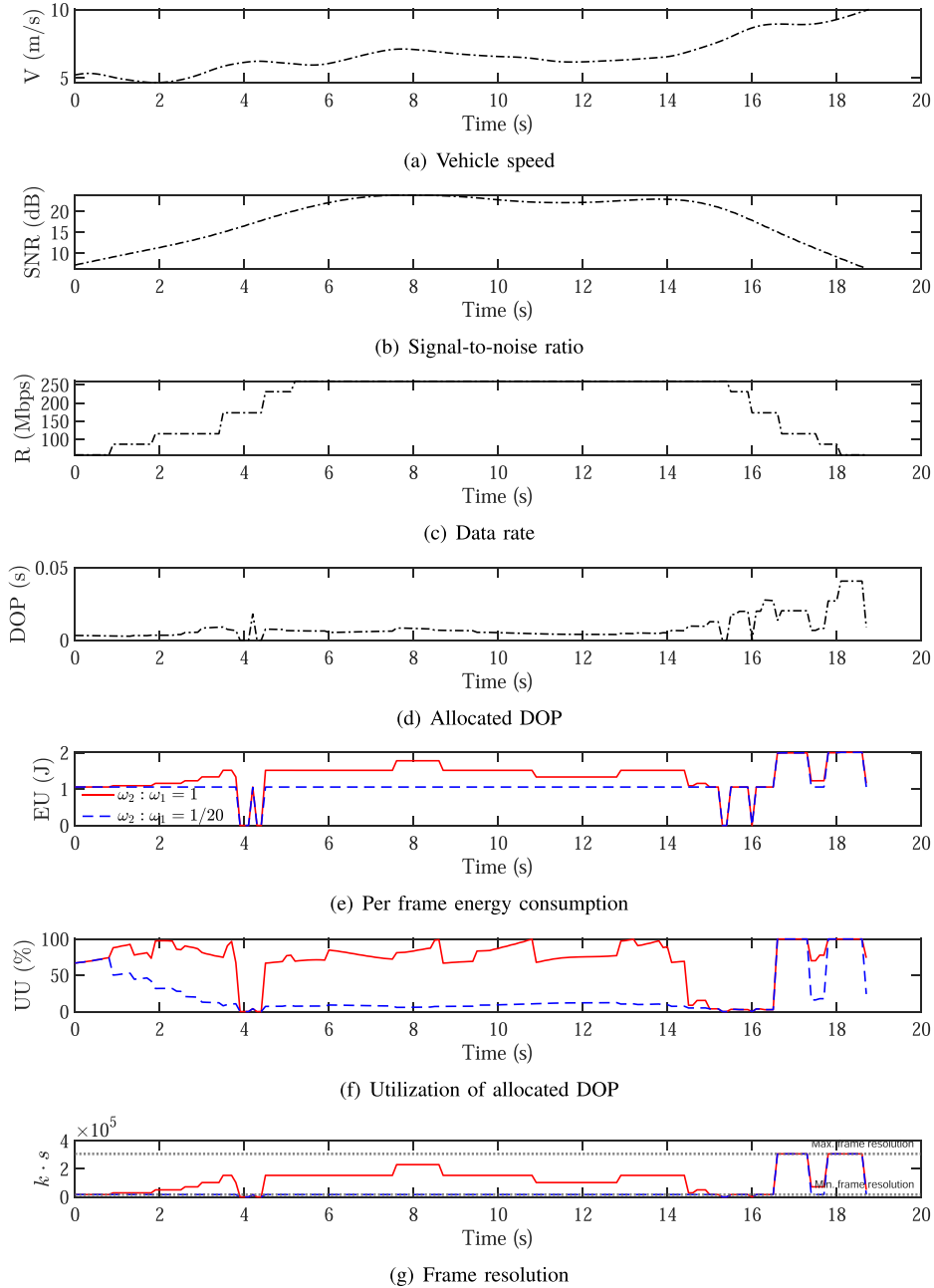


Fig. 9. The performance results of a connected vehicle. (a)–(d) show the selected connected vehicle’s instantaneous speed, signal-to-noise ratio, obtained data rate, and allocated DOP over the time; (e)–(g) illustrate the comparison of two user preference settings (e.g., $\omega_1 = 1$ and $\omega_2 : \omega_1 = 20$) in terms of the per frame energy consumption, utilization of the allocated DOP, and offloaded image frame resolution over the time.

the average frame rate of UCVs by 6.7 and 11.2 times, and the average frame rate of DCVs by 5.9 and 4.7 times, compared to SA and DA, respectively, when the number of UCVs and DCVs is $7 + 8$. In the roundabout scenario, *FAIR* increases the average frame rate of UCVs by 13.7 and 30.2 times, and the average frame rate of DCVs by 12.6 and 8.3 times, compared to SA and DA, respectively, when the number of UCVs and DCVs is $10 + 10$. The larger number of UCVs and DCVs, the higher increment of frame rate *FAIR* can achieve. These results validate that *FAIR* provides an impartial network resource allocation for

uplink and downlink traffic, and is scalable under the different number of connected vehicles and workload.

Figs. 7(e), (f), 8(e), and (f) further validate the optimality of our proposed *FAIR* with the variant number of UCVs and DCVs at roundabout and intersection, respectively. While Figs. 7(g), (h), 8(g), and (h) validate the optimality of *FAIR* with the variant user preference (i.e., ω_1, ω_2) at roundabout and intersection, respectively. We observe that *FAIR* always obtains the minimal Q_u and Q_d compared to other algorithms at both roundabout and intersection.

TABLE III
POWER CONSUMPTION AND DURATION OF PROMOTION & TAIL PHASES IN
DATA TRANSMISSION

P_{pro} (W)	t_{pro} (s)	P_{tail} (W)	t_{tail} (s)
1.97 ± 0.08	0.034 ± 0.004	1.61 ± 0.15	0.21 ± 0.02

From individual connected vehicle perspective: In Fig. 9, we illustrate how FAIR makes the DOP allocation and connected service offloading intelligent under the environment dynamics and varying user preference. We randomly select a UCV in the simulation environment, where the road topology is the roundabout and the number of UCVs and DCVs is 10 and 10, respectively. Fig. 9(a) illustrates the instantaneous speed of the UCV. Fig. 9(b) shows the signal-to-noise (SNR) ratio on the side of the UCV, which also indicates the UCV is in the range of the edge about 19 s. Fig. 9(c) shows the data rate of the UCV and it also demonstrates the correlation between SNR and data rate. Fig. 9(d) illustrates the allocated DOP for the UCV over the time. We can observe that the allocated DOP increases when the UCV's speed is higher (the allocated DOP is also highly affected by other connected vehicles in the environment, which cannot be demonstrated in this figure).

Fig. 9(e), (f), and (g) show the real-time performance of the UCV in terms of per frame energy consumption, EU (i.e., defined in 6), utilization of the allocated DOP, UU (i.e., defined in 15), and the optimal frame resolution determined by Algorithm 2, with two different user preference settings $\omega_2 : \omega_1 = 1 : 1$ or $1 : 20$. The red solid line depicts the real-time performance of the UCV with $\omega_2 : \omega_1 = 1 : 1$; while the blue dashed line depicts the real-time performance of the UCV with $\omega_2 : \omega_1 = 1 : 20$. The energy consumption of transmitting an image frame, EU_{tr} , is calculated based on Table III and the model $P_{tr} = 0.01821 \cdot R + 0.7368$ [26], [38], [39]. The energy consumption of sampling an image frame, EU_{cam} , is calculated by $EU_{cam} = -1.772 \times 10^{-17} \cdot (k \cdot s)^3 + 7.491 \times 10^{-12} \cdot (k \cdot s)^2 + 2.379 \times 10^{-6} \cdot (k \cdot s) + 0.6068$ [26], [38], [39]. We can see that if the user prefers a lower energy consumption (i.e., a larger ω_1), FAIR will be prone to select a lower frame resolution for sampling and offloading, as shown in Fig. 9(g). By lowering the frame resolution, the UCV can save up to 40.6% energy consumption per frame. While if the user prefers a higher frame resolution (i.e., a larger ω_2), the utilization of the allocated DOP will raise significantly, as depicted in Fig. 9(f). These results indicate that our proposed FAIR can intelligently optimize the frame resolution selection based on the user preference.

V. CONCLUSION

In this paper, we proposed FAIR, an end-to-end automotive edge networking system, that can provide fast, scalable, and impartial connected services for intelligent vehicles with edge computing. This is the first work, to the best of our knowledge, that systematically addresses the issue of the asymmetrical resource allocation for uplink and downlink connections in traditional wireless networks. We designed a symmetrical resource allocation algorithm to proactively and periodically reserve dedicated service resource for each

intelligent vehicle under environment dynamics. A service adaptation algorithm was designed to dynamically adjust configurations of environmental sensing and frame resolution according to the reserved service period and user preference. The extensive simulation results demonstrated FAIR can significantly improve the fairness, scalability, and reliability compared to existing solutions in a variety of traffic scenes.

REFERENCES

- [1] Toyota, "Reforming our company to become a "mobility company", Accessed: Sep. 2022, 2019. [Online]. Available: <https://global.toyota/en/company/messages-from-executives/details/>
- [2] Z. Wang et al., "Mobility digital twin: Concept, architecture, case study, and future challenges," *IEEE Internet Things J.*, vol. 9, no. 18, pp. 17452–17467, Sep. 2022.
- [3] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi, "Edge computing for autonomous driving: Opportunities and challenges," *Proc. IEEE*, vol. 107, no. 8, pp. 1697–1716, Aug. 2019.
- [4] S. Lu, N. Ammar, A. Ganlath, H. Wang, and W. Shi, "A comparison of end-to-end architectures for connected vehicles," in *Proc. IEEE 5th Int. Conf. Connected Auton. Driving*, 2022, pp. 72–80.
- [5] P. Lang, D. Tian, X. Duan, J. Zhou, Z. Sheng, and V. C. M. Leung, "Cooperative computation offloading in blockchain-based vehicular edge computing networks," *IEEE Trans. Intell. Veh.*, vol. 7, no. 3, pp. 783–798, Sep. 2022.
- [6] M. H. C. Garcia et al., "A tutorial on 5G NR V2X communications," *IEEE Commun. Surv. Tut.*, vol. 23, no. 3, pp. 1972–2026, Jul.–Sep. 2021.
- [7] S. Xiao, X. Ge, Q.-L. Han, and Y. Zhang, "Resource-efficient platooning control of connected automated vehicles over VANETs," *IEEE Trans. Intell. Veh.*, vol. 7, no. 3, pp. 579–589, Sep. 2022.
- [8] Q. Lu, H. Jung, and K.-D. Kim, "Optimization-based approach for resilient connected and autonomous intersection crossing traffic control under V2X communication," *IEEE Trans. Intell. Veh.*, vol. 7, no. 2, pp. 354–367, Jun. 2022.
- [9] P. Watta, X. Zhang, and Y. L. Murphey, "Vehicle position and context detection using V2V communication," *IEEE Trans. Intell. Veh.*, vol. 6, no. 4, pp. 634–648, Dec. 2021.
- [10] Grand View Research, "Digital twin market size, share & trends analysis report by end use (manufacturing, agriculture), by solution (component, process, system), by region (North America, APAC), and segment forecasts, 2022–2030," 2022. Accessed: Sep. 26, 2022. [Online]. Available: <https://www.grandviewresearch.com/industry-analysis/digital-twin-market>
- [11] H. Wang et al., "Architectural design alternatives based on cloud/edge/fog computing for connected vehicles," *IEEE Commun. Surv. Tut.*, vol. 22, no. 4, pp. 2349–2377, Oct.–Dec. 2020.
- [12] Z. Wang, K. Han, and P. Tiwari, "Digital twin-assisted cooperative driving at non-signalized intersections," *IEEE Trans. Intell. Veh.*, vol. 7, no. 2, pp. 198–209, Jun. 2022.
- [13] Y. Tian, J. Wang, Y. Wang, C. Zhao, F. Yao, and X. Wang, "Federated vehicular transformers and their federations: Privacy-preserving computing and cooperation for autonomous driving," *IEEE Trans. Intell. Veh.*, vol. 7, no. 3, pp. 456–465, Sep. 2022.
- [14] R. van Hoek, J. Ploeg, and H. Nijmeijer, "Cooperative driving of automated vehicles using B-splines for trajectory planning," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 594–604, Sep. 2021.
- [15] Y. Wang, Z. Wang, K. Han, P. Tiwari, and D. B. Work, "Gaussian process-based personalized adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 21178–21189, Nov. 2022.
- [16] Y. Lin, J. McPhee, and N. L. Azad, "Comparison of deep reinforcement learning and model predictive control for adaptive cruise control," *IEEE Trans. Intell. Veh.*, vol. 6, no. 2, pp. 221–231, Jun. 2021.
- [17] Z. Hu, S. Lou, Y. Xing, X. Wang, D. Cao, and C. Lv, "Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles," *IEEE Trans. Intell. Veh.*, vol. 7, no. 3, pp. 417–440, Sep. 2022.
- [18] Y. Liu, Z. Wang, K. Han, Z. Shou, P. Tiwari, and J. H. L. Hansen, "Vision-cloud data fusion for ADAS: A lane change prediction case study," *IEEE Trans. Intell. Veh.*, vol. 7, no. 2, pp. 210–220, Jun. 2022.
- [19] J. E. Hoffmann, H. G. Tosso, M. M. D. Santos, J. F. Justo, A. W. Malik, and A. U. Rahman, "Real-time adaptive object detection and tracking for autonomous vehicles," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 450–459, Sep. 2021.

- [20] dRISK Team, "dRisk demonstrates physically based edge-case testing in NVIDIA DRIVE Sim," 2022. Accessed: Sep. 26, 2022. [Online]. Available: <https://drisk.ai/news/drisk-demonstrates-physically-based-edge-case-testing-in-nvidia-drive-sim/>
- [21] K. Winter, "Intel Editorial: For self-driving cars, there's big meaning behind one big number: 4 terabytes," 2017. Accessed: Sep. 26, 2022. [Online]. Available: <https://www.businesswire.com/news/home/20170414005225/en/Intel-Editorial-For-Self-Driving-Cars-There%E2%80%99s-Big-Meaning-Behind-One-Big-Number-4-Terabytes>
- [22] T. S. J. Darwish and K. A. Bakar, "Fog based intelligent transportation Big Data analytics in the internet of vehicles environment: Motivations, architecture, challenges, and critical issues," *IEEE Access*, vol. 6, pp. 15679–15701, 2018.
- [23] J. Ott and D. Kutscher, "Drive-thru internet: IEEE 802.11b for "automobile" users," in *Proc. IEEE Conf. Comput. Commun. Workshops*, 2004, Art. no. 373.
- [24] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, "A close examination of performance and power characteristics of 4G LTE networks," in *Proc. ACM 10th Int. Conf. Mobile Syst., Appl., Serv.*, 2012, pp. 225–238.
- [25] D. Chen et al., "Love of variety based latency analysis for high definition map updating: Age of information and distributional robust perspectives," *IEEE Trans. Intell. Veh.*, early access Nov. 24, 2022, doi: [10.1109/TIV.2022.3224655](https://doi.org/10.1109/TIV.2022.3224655).
- [26] H. Wang and J. Xie, "User preference based energy-aware mobile AR system with edge computing," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 1379–1388.
- [27] S.-C. Lin et al., "The architectural implications of autonomous driving: Constraints and acceleration," in *Proc. ACM 23th Int. Conf. Architectural Support Program. Lang. Operating Syst.*, 2018, pp. 751–766.
- [28] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
- [29] M. Koprivica, M. Ilić, A. Nešković, N. Nešković, and N. Krajnović, "Experimental evaluation of IEEE 802.11e EDCA QoS mechanism for voice over WLAN," in *Proc. IEEE Int. Conf. Comput. Tool*, 2011, pp. 1–4.
- [30] "NETGEAR," 2022. [Online]. Available: <https://www.netgear.com/>
- [31] H. Wang, J. Xie, and X. Liu, "Rethinking mobile devices' energy efficiency in WLAN management services," in *Proc. IEEE 15th Annu. Int. Conf. Sens., Commun., Netw.*, 2018, pp. 1–9.
- [32] J. Huang et al., "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3296–3297.
- [33] H. Wang, B. Kim, J. Xie, and Z. Han, "Energy drain of the object detection processing pipeline for mobile devices: Analysis and implications," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 1, pp. 41–60, Mar. 2021.
- [34] R. Krajewski, T. Moers, J. Bock, L. Vater, and L. Eckstein, "The round dataset: A drone dataset of road user trajectories at roundabouts in Germany," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst.*, 2020, pp. 1–6.
- [35] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, "The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections," in *Proc. IEEE Intell. Veh. Symp.*, 2019, pp. 1929–1934.
- [36] "BOSCH," 2022. [Online]. Available: <https://www.bosch-mobility-solutions.com>
- [37] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
- [38] H. Wang, B. Kim, J. L. Xie, and Z. Han, "LEAF+AIO: Edge-assisted energy-aware object detection for mobile augmented reality," *IEEE Trans. Mobile Comput.*, early access, Jun. 3, 2022, doi: [10.1109/TMC.2022.3179943](https://doi.org/10.1109/TMC.2022.3179943).
- [39] H. Wang, B. Kim, J. Xie, and Z. Han, "How is energy consumed in smartphone deep learning apps? Executing locally vs. remotely," in *Proc. IEEE Glob. Commun. Conf.*, 2019, pp. 1–6.



Haoxin Wang (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of North Carolina at Charlotte, Charlotte, NC, USA, in 2020, and the B.S. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China in 2015. From 2020 to 2022, he was a Research Scientist with Toyota Motor North America, InfoTech Labs, Mountain View, CA, USA. He is currently an Assistant Professor with the Department of Computer Science, Georgia State University, Atlanta, GA, USA, and Leads the Advanced Mobility and Augmented Intelligence Lab. His research interests include mobile AR/VR, autonomous driving, digital twins, and edge intelligence.



Jiang Xie (Fellow, IEEE) received the B.E. degree in electrical and computer engineering from Tsinghua University, Beijing, China, the M.Phil. degree in electrical and computer engineering from the Hong Kong University of Science and Technology, Hong Kong, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA. In August 2004, she joined the Department of Electrical and Computer Engineering, University of North Carolina at Charlotte, Charlotte, NC, USA, where she is currently a Full Professor. Her research interests include resource and mobility management in wireless networks, mobile computing, Internet of Things, and cloud/edge computing. She was the recipient of the US National Science Foundation Faculty Early Career Development (CAREER) Award in 2010, Best Paper Award from IEEE Global Communications Conference in 2017, Best Paper Award from IEEE/WIC/ACM International Conference on Intelligent Agent Technology in 2010, and Graduate Teaching Excellence Award from the College of Engineering at UNC Charlotte in 2007. She is on the Editorial Boards of the *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, *IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING*, and *Journal of Network and Computer Applications (Elsevier)*. She is a Senior Member of the ACM.



Muhana Magboul Ali Muslam (Member, IEEE) received the B.Sc. degree in computer science from The Future University, Khartoum, Sudan, in 2003, the M.Sc. degree in computer science from the University of Khartoum, Khartoum, in 2006, and the Ph.D. degree in electrical engineering from the University of Cape Town, Cape Town, South Africa, in 2012. He is currently an Assistant Professor with the Department of Information Technology, Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia.