EdgeCooper: Network-Aware Cooperative LiDAR Perception for Enhanced Vehicular Awareness

Guiyang Luo, Chongzhang Shao, Nan Cheng, *Senior Member, IEEE*, Haibo Zhou, *Senior Member, IEEE*, Hui Zhang, *Member, IEEE*, Quan Yuan, *Member, IEEE*, and Jinglin Li

***Abstract*— Autonomous driving vehicle (ADV) that is ready to transform our society and economy, is in desperate need of precise positioning over itself as well as surrounding environments. However, it is still a challenging issue for ADV to retrieve real-time positioning knowledge over road participants and dynamic surrounding environments, due to unsatisfied perception accuracy caused by sparse observations and limited perception range. Cooperative perception, which advocates cooperatively disseminating perception data among vehicles, has the potential to overcome the above limitations. To this end, this article proposes a novel edge-assisted multi-vehicle perception system to enhance vehicles' awareness over surrounding environments, which is termed as EdgeCooper. EdgeCooper first schedules vehicles to share complementarity-enhanced and redundancy-minimized raw sensor data with an edge server, using multi-hop cooperative 5G V2X communications. Then, EdgeCooper merges vehicles' individual views to form a holistic view with a higher resolution, thus enhancing perception robustness and enlarging perception range. We formulate multi-vehicle multi-hop cooperative data sharing as a minimum cost flow problem with conflict, and further prove that there exists no polynomial-time approximation algorithm with a constant performance ratio unless P = NP. Furthermore, a two-dimension graph coloring algorithm with guaranteed performance is proposed to eliminate conflict. We evaluate EdgeCooper by building a comprehensive simulation platform through a joint manipulation of SUMO, CARLA, NS3, and PyTorch. The experiment results show that, compared to a single vehicle's perception, EdgeCooper performs effective and efficient in enhancing vehicular awareness, e.g., extending up to 3.6 times detection range and improving perception accuracy by 20%.**

***Index Terms*— Cooperative communication, collaborative perception, network-aware, edge computing, positioning.**

Manuscript received 8 February 2023; revised 27 June 2023; accepted 13 August 2023. Date of publication 9 October 2023; date of current version 19 December 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB4300400, in part by the Natural Science Foundation of China under Grant 62102041 and Grant 62272053, and in part by the Young Elite Scientists Sponsorship Program by China Association for Science and Technology (CAST) under Grant 2022QNRC001. *(Corresponding author: Nan Cheng.)*

Guiyang Luo and Quan Yuan are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the Key State Laboratory of ISN, Xidian University, Xi'an 710071, China (e-mail: luoguiyang@bupt.edu.cn; yuanquan@bupt.edu.cn).

Chongzhang Shao and Jinglin Li are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: shaochongzhang@bupt.edu.cn; jlli@bupt.edu.cn).

Nan Cheng is with the Key State Laboratory of ISN and the School of Telecommunications Engineering, Xidian University, Xi'an 710071, China (e-mail: nancheng@xidian.edu.cn).

Haibo Zhou is with the School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China (e-mail: haibozhou@nju.edu.cn).

Hui Zhang is with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100876, China (e-mail: huizhang1@bjtu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2023.3322764.

Digital Object Identifier 10.1109/JSAC.2023.3322764

I. Introduction

AUTONOMOUS driving vehicle (ADV) is anticipated to transform transportation systems by providing better road safety and traffic efficiency. The deployment of it in our roads within the next future requires highly accurate positioning information, i.e., decimeter-level positioning for highway operation and near-centimeter level for operation on local and residential streets [1]. Such high-accuracy positioning depends on a sophisticated manipulation over a combination of positioning technologies, e.g., global navigation satellite system (GNSS), cellular positioning, and simultaneous localization and mapping using onboard sensors such as LiDAR, radar, and cameras [2]. On top of self-positioning, it is also crucial for ADVs to real-timely extract the position and orientation of adjacent vehicles, road participants (e.g., pedestrians, cyclists), as well as dynamic surrounding environments [3]. Collaborative awareness messages (CAM) [4], which are short messages exchanged by vehicles for mutual awareness of each other's location, have already been standardized by Telecommunication Standard Institute (ETSI, the standard EN 302 637-2) [5]. However, CAM only provide position information of adjacent vehicles, and are not able to locate pedestrians without an end device (smartphone). Furthermore, it is challenging for ADV to retrieve the extremely dynamic components in surrounding environments, e.g., congestion, accidents, construction, bikes, and road surface irregularities (pits, bumps, holes, speed breakers, etc.) [6].

Perception, which detects and understands the environments through onboard sensors, can percept neighboring pedestrians as well as dynamic environments by exploiting advanced computer vision techniques [7]. However, these sensors suffer from two fundamental limitations [8]. First, they are vulnerable to occlusion, i.e., these sensors cannot perceive objects occluded by non-transparent objects. Second, similar to human eyes, the farther an object is, the fewer details the sensors can capture, i.e., the observations are very sparse or non-existent for a farther object [9]. Consequently, even equipped with redundant and rich sensors, the perception accuracy is still unsatisfactory.

For example, Tesla's vehicle failed to recognize the truck in 2016, the highway divider in 2018, and the semitrailer in 2019, all leading to fatalities [10]. A similar fatality happened in 2018 because Uber's autonomous driving system failed to recognize that pedestrians jaywalk [11].

To this end, we advocate exploiting collaborative perception (CP) [12], [13], [14] to enhance vehicles' awareness over adjacent vehicles, road participants, as well as surrounding environments. Building upon 5G vehicle-to-everything (V2X) communication [15], [16], CP cooperatively shares local perception data to construct a complete and robust perception of its environment [17], which is very promising to overcome the above limitations. CP could not only maximize the line of sight and field of view but also reduce the uncertainty in local object detection results and increase perception accuracy, compensating for sensor/perception deficits, and removing blind spots [18]. For example, as shown in Fig.1, vehicles at an intersection cooperatively share their perceived information to see-through occlusion. Vehicles C and D are occluded, which can also be illustrated by the point clouds shown in View C and View D, where the point clouds in the corresponding areas are empty. Powered with CP, C and D can perceive each other if vehicle A or B could share its perceived information. Therefore, CP could enhance perception accuracy and robustness, as well as enlarge perception range [19]. Considering the advantages of CP, recent years have witnessed standardization efforts for collaborative perception services from the ETSI [20].

To fully release the power of CP and vehicular edge computing [21], this paper develops a system called EdgeCooper where ADVs cooperatively upload their perceived LiDAR data to the edge using 5G V2X communications, creating a global and complete view and enhancing vehicles' awareness over surrounding environments. The edge first exploits the computing resources to perform object detection over a merged view and then broadcasts the detection results to all participants for providing better cooperative awareness. EdgeCooper advocates cooperative perception by aggregating ADVs' sensor data to the edge, which enjoys the following advantages compared with cooperative perception between adjacent ADVs: 1) Better detection accuracy. ADV may not have enough computational resources to process others' data at a line rate and may resort to a lightweight model with sacrificed accuracy [22]. However, the edge can exploit the abundant computing resources in vehicular edge computing to run a heavyweight deep learning detection model for improved detection performance. 2) Reduced bandwidth usage. In EdgeCooper, the edge server aggregates point clouds from vehicular and roadside LiDARs and then performs object detection over the merge view. EdgeCooper schedules vehicles to share partial point clouds with the edge server, thereby necessitating transmission of the point cloud only once for each vehicle. Consequently, EdgeCooper effectively minimizes bandwidth utilization. However, if each vehicle retrieves the sensor data by cooperatively sharing point clouds among adjacent vehicles, then, each vehicle has to disseminate its point clouds to every neighbor, consuming massive communication resources. 3) A wider range of beneficiaries. End users with little computing resources, e.g., bicyclists, and pedestrians, can benefit from EdgeCooper since the computation task is performed at the edge. However, implementing EdgeCooper meets the following challenges.

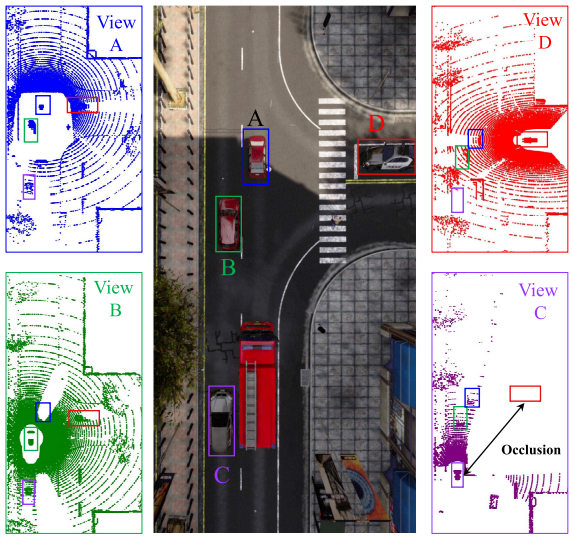


Fig. 1. Collaborative perception for autonomous driving at an intersection, which could enhance perception accuracy and robustness as well as enlarge perception range.

A. Challenges

1) How to Select Complementarity-Enhanced and Redundancy-Minimized Data for Sharing?: The shared data could either be raw sensor data or intermediate features. Several studies advocate sharing of processed data, i.e., intermediate features of object detection deep learning model. However, there will be a loss of information during data processing, and sharing processed data lacks generality. In contrast, raw sensor data has a simple, fundamental, and universal data format to flexibly support a wide range of CAV applications. As for sharing of raw sensor data, Arnold et al. [23] directly transmit individual perceived raw sensor data, which is bandwidth-inefficient since the sensor data are redundant for containing overlapping areas. Therefore, Zhang et al. and Aoki et al. [24] adopt a disjoint spatial partition of the environment and cooperatively transmit the partitioned sensor data, i.e., select the best view for each area. However, it would deteriorate the performance since multiple views are complementary. These methods perform toward two extremes. Therefore, we should properly select the shared data that is less redundant and most complementary, to reduce the transmitted data while minimizing information loss.

2) How to Design a Network-Aware and Scalable Transmission Strategy?: In CP, each area can be perceived by multiple ADVs, and a subset of ADVs is scheduled to transmit the data for each area considering network status. Considering the bandwidth requirements of CP, multi-hop cooperative transmission between the device to device and cellular links is exploited for improved bandwidth performance [15]. Therefore, EdgeCooper has to jointly consider

link scheduling, channel assignment, packet scheduling, and relay selection to develop a scalable transmission strategy. Link scheduling selects a subset of links under a predefined utility, which is usually modeled as a maximum weighted independent set problem and is known to be NP-Hard [25]. Channel assignment allocates each link with a channel considering interference and half-duplex constraints, which involves solving a graph coloring problem and is also a known NP-Hard problem [26]. On top of these two NP-Hard problems, EdgeCooper has to additionally consider packet scheduling and relay selection, which further complicates and mystifies the transmission strategy.

3) How to Conduct Co-Simulation to Evaluate EdgeCooper?: EdgeCooper involves joint manipulation over transportation traffic, sensors of ADVs, edge computing and wireless communications, and deep learning detection models. Conducting co-simulation among transportation simulators, autonomous driving simulators, wireless communication simulators, and artificial intelligence frameworks is a requisite for evaluating the performance of EdgeCooper.

B. Solutions and Contributions

1) Voxelization-Based Strategy for Selecting Complementarity-Enhanced and Redundancy-Minimized Data: Instead of cooperatively sharing a disjoint spatial partition or entire perceived data of the environment, EdgeCooper adopts a novel voxelization-based sharing strategy. It is inspired by voxel-based LiDAR detection model that samples and groups point in voxel. EdgeCooper schedules sharing of data in each voxel considering the network situations. The cumulated number of points in each voxel is urged to provide enough representations for the objects. Consequently, in each voxel, the edge can retrieve data from multiple ADVs to enhance complementarity. Meanwhile, the retrieved total number of points is restricted to minimize redundancy.

2) Network-Aware and Scalable Transmission Strategy Based on Maximum Flow Theory: We first formulate a collaborative transmission strategy involving link scheduling, channel assignment, packet scheduling, and relay selection as an integer optimization problem, and prove that it is NP-Hard. Then, we convert it into a minimum cost flow problem with conflict, and theoretically prove that there is no polynomial time approximation algorithm with a constant performance ratio unless P = NP. After that, we divide it into two subproblems, minimum cost flow, and conflict elimination problem. A two-dimension graph coloring algorithm with guaranteed performance is proposed to address the conflict elimination problem.

3) Co-Simulation Among SUMO, CARLA, NS3, and PyTorch: We build a comprehensive simulation platform that is capable of conducting large-scale co-simulation by a joint manipulation over transportation traffic simulator SUMO [27], autonomous driving simulator CARLA, communication simulator NS3, and deep-learning framework PyTorch. Exploiting the co-simulation platform, EdgeCooper is evaluated at 28 intersections involving up to 50 vehicles as compared with several baseline methods. Experimental results demonstrate that EdgeCooper can reconstruct a robust and complete perception over an enlarged area with a network-aware and scalable transmission strategy, performing effectively and efficiently in enhancing vehicular awareness.

The object detection can achieve real-time processing at 50 frame per second (FPS) and EdgeCooper has end-to-end latency of $\approx$ 100 ms. Furthermore, compared to a single vehicle's perception, EdgeCooper can extend up to 3.6 times the detection range and improve perception accuracy by 20%.

The remainder of this paper is organized as follows. Section II reviews the related works. Section III describes the preliminaries. Section IV introduces the system model. We investigate the complexity and the proposed solution in Section V and Section VI, respectively. Section VII is focused on the experiments. Finally, Section VIII concludes the paper.

II. Related Works

EdgeCooper enables multi-agent collaborative perception [28] through multi-hop cooperative communication, where interested vehicles upload their individual point clouds to the edge server to construct a holistic and extended view. This section reviews the related works.

The multi-agent collaborative perception [29] approaches mainly include output-based late collaboration, feature-based intermediate collaboration, and raw-measurement-based early collaboration [30]. Late collaboration only shares perception results among multiple agents. Volk et al. [31] implement a track-to-track fusion of the cooperatively perceived objects to the local tracks of the ego vehicle, to enhance object tracking performance. Arnold et al. [23] apply a post-processing algorithm non-maximum suppression for box fusion. Liu et al. [32] adopt a self-adaptive topology merging algorithm based on a bipartite graph to determine if detected vehicles from two views are the same, thus creating an accurate merged topology map. In late collaboration, each individual perception output could be noisy and incomplete, which may cause unsatisfying fusion results.

Intermediate collaboration involves sharing neural network features among agents, which has been investigated in [19], [33], [34], [35], [36], and [37]. Chen et al. [19] propose a point cloud feature based cooperative perception framework for connected autonomous vehicles to achieve better object detection precision. Guo et al. [33] differentiate weights among feature maps for a more guided fusion, based on how much new semantic information is provided by the received feature maps. It enhances the inconspicuous features corresponding to far/occluded objects to improve their detection precision. Liu et al. [34] support multi-agent collaborative perception in a distributed and bandwidth-efficient manner, by learning how to construct communication groups as well as decide when to communicate. Liu et al. [35] propose a multi-stage handshake communication mechanism where the neural network can learn to compress relevant information needed for each stage. Wang et al. [36] propose a vehicle-to-vehicle approach that transmits compressed intermediate representations, which can

intelligently aggregate the information received from multiple nearby vehicles using a spatially aware graph neural network. Xu et al. [37] build a holistic attention model to effectively fuse information across on-road agents. It includes both heterogeneous multi-agent self-attention and multi-scale window self-attention, which can capture inter-agent interaction and per-agent spatial relationships. Intermediate collaboration is bandwidth-efficient, since we can squeeze representative information into compact features. However, there will be a loss of information during data processing, and sharing processed data lacks generality. For example, different agents use heterogeneous neural networks, making it hard to share intermediate features since they have different latent spaces.

Early collaboration first aggregates sensor data collected by multi-agent, resulting in a holistic view. Then, object detection is performed using the holistic view. Early collaboration can fundamentally solve the occlusion and long-range issues occurring in the single-agent perception. Furthermore, raw sensor data has a simple, fundamental, and universal data format to flexibly support a wide range of CAV applications [38]. Aoki et al. [24] propose a cooperative perception scheme with deep reinforcement learning [39] to select the data to transmit, which mitigates the network load in vehicular communication networks and enhances the communication reliability. Zhang et al. [40] merge vehicle individual views to form a more complete view with a higher resolution, by transmitting the raw sensor data to an edge server. Arnold et al. [23] also combine point clouds from multiple spatially diverse sensing points of view before detection. To the best of our knowledge, no one has considered multi-hop cooperative communication for supporting early collaboration. The raw sensor data are of large volume, which urgently needs cooperative communication for higher spectrum efficiency. Yu et al. [41] explores the performance gap between distributed and centralized C-V2X scheduling in terms of achievable throughput and communication efficiency in CP. However, they ignore the sensor data characteristics and apply dummy perception data for simulation. Furthermore, most existing methods advocate sharing of all sensor data among vehicles, which is quite inefficient. For instance, in uncertainty-aware localization, only features that contribute to object location inference are needed, and features of objects with high uncertainties among vehicles can be prioritized for transmission to improve real-time localization accuracy. To the best of our knowledge, our paper is the first to catenate communication and perception, which is achieved by a novel voxelization-based strategy. The proposed strategy establishes a connection between metrics in communications (e.g., throughput, efficiency) and metrics in object detection (e.g., accuracy). Furthermore, a network-aware and scalable transmission strategy is proposed to cooperatively disseminate point clouds.

Most existing vehicular communication studies focus on the transmission of CAMs [4], which are short messages exchanged by vehicles for mutual awareness of each other's location in tasks such as collision warning or traffic estimation. The difference between CAMs and CP is that CAMs contain only metadata about the vehicle itself (location, velocity, orientation, etc.), while CP contains sensory data or data representations obtained by the vehicle. Compared with CAMs, CP makes it possible to detect pedestrians and things (e.g., holes in the road) that are not equipped with any wireless communication devices or sensors. Both CAMs and CP need to be transmitted and received periodically in real-time, but CP is commonly much more bandwidth-hungry than CAMs.

III. Preliminaries

A. Point Clouds and Objection Detection Model PointPillars

This paper is focused on the perception task of LiDAR-based object detection because the unifying 3D space naturally allows the aggregation of multiple LiDAR scans [42]. Each LiDAR N_i can periodically generate point cloud $\mathcal{P}_i = \{p_i^1, p_i^2, \cdots\}$, where $p_i^j \triangleq [x, y, z, r]$, (x, y, z) is the 3D coordinates, and r is the reflectance value.

Inspired by the significant success of voxel-based object detection models, e.g., PointPillars [43], which divides point clouds into equally spaced voxels. Specifically, we apply the state-of-the-art PointPillars [43] for object detection. It is computationally efficient and is adopted by existing industry-level autonomous driving platforms such as Baidu Apollo and Autoware. PointPillars advocates point cloud voxelization in birds-eye view and organizes point clouds in vertical columns (pillars), i.e., the point clouds are discretized into an evenly spaced grid in the x-y plane, and each grid represents a pillar.[1] Each pillar o contains several data points and can be represented as $(x, y, \mathcal{P}_i^o)$, where x, y denote the pillar center location, and $\mathcal{P}_i^o$ is the set of points located in pillar o, $\mathcal{P}_i^o \subset \mathcal{P}_i$.

Definition 1: ***Sampling strategy in PointPillars.*** *Different pillars contain different numbers of points. If a pillar o holds too many points and only P_{max} points are kept by random sampling, i.e., $|\mathcal{P}_i^o| > P_{max}$. Otherwise, zero padding is applied.*

Furthermore, a *sampling strategy* is applied in PointPillars for ensuring the same number of points in each pillar, which is defined in *Definition 1*. Consequently, the pillars can create a pseudo-image that is compatible with standard 2D convolutions. In the experiments, P_{max} is set to 32 and the pillar size is set to 0.4×0.4 meters according to [43]. Then, a 2D CNN backbone network with high computational efficiency can be applied to extract features, followed by a detection head for classification and regression.

B. Relationship Between Accuracy and Point Cloud Density

LiDAR point clouds are sparse and have highly variable point density, due to factors such as non-uniform sampling of the 3D space, effective range of the sensors, occlusion, and the relative pose. It is significantly challenging to detect objects with sparse observations, i.e., areas with low point cloud density. Therefore, considering the sampling strategy of PointPillars, the pillar satisfaction degree is defined as:

[1]Pillars are special voxels where there is only one voxel along the height (z) dimension, which eliminates the need to tune binning of the vertical direction by hand and achieves faster speed by removing expensive 3D convolutional layers.

TABLE I
SUMMARY OF NOTATIONS

Notations	Descriptions
$\mathcal{N}, \mathcal{M}$	The set of vehicular and roadside LiDARs, respectively.
$\mathcal{O}$	The set of pillars.
$\mathcal{P}_i$	Point cloud observed by N_i, $\mathcal{P}_i = \{p_i^1, p_i^2, \cdots\}$.
$\mathcal{P}_i^o$	The set of points observed by LiDAR N_i and located in pillar o, $\mathcal{P}_i^o \subset \mathcal{P}_i$.
P_{max}	The maximum number of points for the sampling strategy.
$\mathcal{T}, \mathcal{C}$	The set of time slots and subchannels, respectively.
$u_{j,k,t}^{i,o}$	The number of points that participant i uploads regarding pillar $o \in \mathcal{O}$, using the link $j \to k$ at t-th time slot.
$m_{j,k,t}$	The maximum number of points can be transmitted on link $j \to k$ at t-th time slot.
$x_{i,j,t}$	Link activation status.
$\eta_{j,k,t}$	Channel allocation strategy.
$E(o)$	The aggregated points for pillar $o \in \mathcal{O}$ from all participants.
$i \to j$	Transmission link from i to j.
$\mathcal{G}_t^p(i \to j, k \to l)$	Conflict of primary interference between $i \to j$ and $k \to l$.
$\mathcal{G}_t^s(i \to j, k \to l)$	Conflict of secondary interference between $i \to j$ and $k \to l$.
S,T	The virtual source and sink node, respectively.
n_i^t	Participant i in the t-th time slot.
$\beta_t^p(n_j^t \to n_k^t)$	The set of neighbors to $n_j^t \to n_k^t$ that is conflict by the primary conflict.
$\beta_t^s(n_j^t \to n_k^t)$	The set of neighbors to $n_j^t \to n_k^t$ that is conflict by the secondary conflict.

Definition 2: ***Pillar satisfaction degree.*** *Inspired by the sampling strategy of PointPillars, the satisfaction degree over a pillar o is defined as $g_o = min(P_{max}, |\mathcal{P}_i^o|)$.*

IV. System Model and Formulation

A. Collaborative Perception

We consider an urban intersection covered and serviced by an edge server, as shown in Fig.2. There exist two kinds of sensor devices, N vehicular LiDARs (a subset of vehicles that is installed with LiDARs and is willing to join the collaborative perception task), and a set $\mathcal{M}$ of M roadside LiDARs. We denote the edge server together with the set of vehicular LiDARs as $\mathcal{N} = \{0, 1, \cdots, N\}$, where 0 stands for the edge server. All these LiDARs collaborate to enhance the perception performance. Powered with high-definition map [44], infrastructure-based positioning, and vehicle cooperative positioning, it is possible to achieve high-precision positioning at the centimeter, or even millimeter level [45]. Such positioning accuracy is quite enough to align voxel grids collected from different participants (note that the pillar size is 0.4×0.4 meters according to [43]). Therefore, we reasonably assume that each agent is provided with an accurate position and that the perceived measurements are well synchronized.

In the considered urban intersection, EdgeCooper advocates that vehicles cooperatively upload their individual point clouds to the edge server, releasing the power of collaborative perception. EdgeCooper is divided into the following three stages for supporting collaborative perception service: 1)

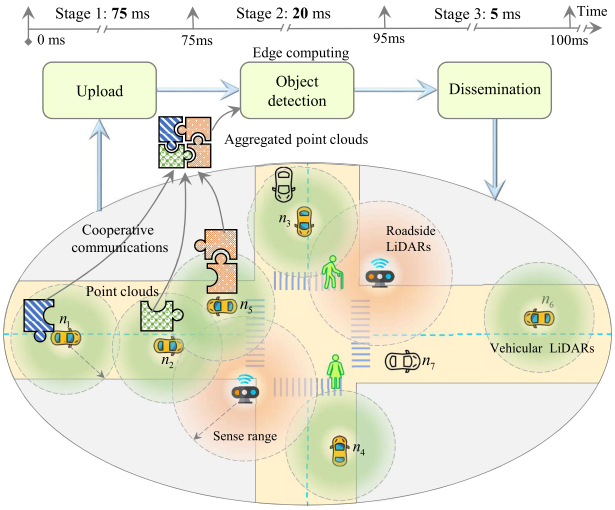


Fig. 2. Framework of EdgeCooper: collaborative perception at a complex intersection assisted with vehicular edge computing.

Upload. Participants (i.e., all interested vehicles and roadside LiDARs) first cooperatively upload point clouds to the edge server, through multi-hop wireless communications (including Sidelink for V2V and V2I communication as well as Uplink for vehicle to base station communication) or wire-line links (roadside LiDARs). Therefore, roadside LiDARs can upload all their point clouds while vehicular LiDARs disseminate partial raw point clouds rather than the whole sensory data to reduce the redundancy caused by overlapping perspectives, improving communication efficiency. 2) *Object detection*. The edge server performs object detection based on aggregated point clouds and provides improved perception results for better driving decisions. 3) *Results broadcast*. The edge server disseminates the perception results to all interested participants through a broadcast channel, since the detection results are of small size.

Definition 3: ***Grid.*** *The covered area of the edge server is horizontally (x-y plane) partitioned into grids of equal size.*

In order to merge points from point clouds generated by different devices, the size and area of pillars generated by each participant should be unified. Therefore, the covered area is divided into grids according to *Definition 3*, as shown in Fig.3. Here, the grid size can be greater or equivalent to the pillar. For simplicity, we assume that each grid is exactly a pillar, and the grid is used interchangeably as the pillar hereafter. Furthermore, this pillar setting is agreed upon and shared by all the participants and each participant would process its point clouds based on *Definition 3*. Each participant observes the point clouds based on different coordinate systems since they have very different perspectives of the world depending on the location and orientation of its sensors. Therefore, each participant can exploit perspective transformation based on the accurate position to unify the coordinate system. For simplicity, the set of grids is denoted as $\mathcal{O}$. Each participant i can observe a subset of pillars $\mathcal{O}_i \subseteq \mathcal{O}$, and the set of points in pillar $o \in \mathcal{O}_i$ is $\mathcal{P}_i^o$. Participants cooperatively disseminate points to the edge server, for broadening the perception range and enhancing detection accuracy and robustness. Furthermore,

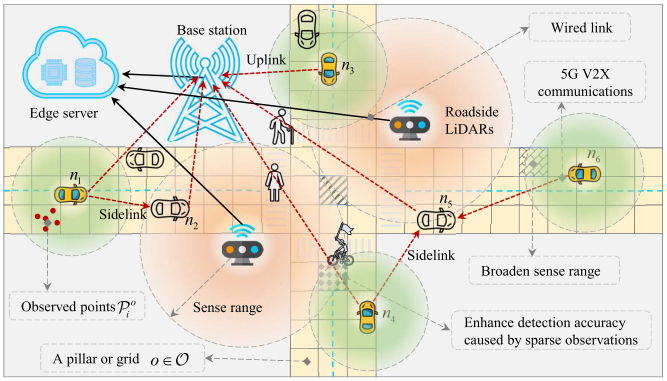


Fig. 3. The upload stage of EdgeCooper, where participants cooperatively share point clouds to the edge server through multi-hop wireless communication for higher spectrum efficiency.

the dissemination should be complementarity-enhanced and redundancy-minimized as defined in *Definition 4* to consume less wireless resources and support more users.

Definition 4: ***Complementarity-enhanced and redundancy-minimized point cloud sharing.*** *All the vehicular and roadside LiDARs transmit point clouds to the edge server in the* Upload *stage and each pillar may contain points uploaded by several LiDARs. According to the sampling strategy in PointPillars, it is redundancy for the transmission if a pillar holds too many points, i.e.,* $|\mathcal{P}_i^o| > P_{max}$*, since only* P_{max} *points are kept by random sampling. It is complementarity if a pillar holds less than or equal to* P_{max}*. Otherwise, zero padding is applied, which would harm the performance.*

Collaborative perception can provide better cooperative awareness, requiring cooperatively sharing of safety messages (e.g., positions, speed) at a frequency of 10 Hz, which enables higher road safety and cooperative traffic efficiency. Therefore, the edge server should update the perception data to vehicles every 100 ms, and the collaborative perception service should be executed every 100 ms. In *object detection* stage, the computation time expends approximately 20 ms since PointPillars has the detection speed of 50 Hz. The *results broadcast* stage usually takes several milliseconds. Consequently, there exists approximately 75ms for the *upload* stage, as shown in Fig.2. We further assume the considered system is time-slotted and the *upload* stage contains several time slots, which allows *multihop cooperative dissemination* of the partial point clouds to the edge server. Compared with single-hop transmission, multi-hop transmission enjoys higher data rates, more efficient use of the wireless medium, and better energy efficiency.

The goal of the *upload* stage is to maximize the pillar satisfaction degree based on optimized multi-hop cooperative dissemination, i.e., retrieve P_{max} points for each pillar $o \in \mathcal{O}$, as shown in Fig.3. EdgeCooper is periodically executed every 100 ms, and such a short duration makes it possible to ignore the movement of vehicles.

B. System Model

According to Rel-14 to Rel-17, 5G-V2X use 10 ms sidelink frames in the time domain divided into 1 ms subframes (the same as LTE-V2X) [46]. Each subframe is defined as one transmission time interval. We assume a 20 MHz channel in this paper as this is the most common bandwidth and it is supported in both 5G- and LTE-V2X. The 5G-V2X channel is divided into several equal-width logical subchannels. Each subchannel consists of 10 resource blocks (RB).[2] Depending on payload size, a transmission may require one or more RBs. Considering the large-volume data transmission, we further pack an equal number of contiguous RBs within a subchannel as resource chunks [47]. Each resource chunk spans T_0 ms and is applied to support transmission. Consequently, our system is slotted based on T_0 and the set available time slots for the *upload* stage is denoted as $\mathcal{T} = \{T_0, 2 * T_0, \cdots, T * T_0\}$. We further assume that the 5G-V2X channel is divided into C subchannels, and the set of all subchannels is denoted as a set $\mathcal{C}$, which is shared among the V2X links [47], [48], where each link can be allocated with only a subchannel. EdgeCooper applies sidelink as well as uplink for disseminating point clouds to the edge server through multi-hop communication. We further assume that the sidelink and uplink share the C subchannels. The 5G-V2X works in mode 1, which is similar to mode 3 in LTE-V2X, where the base station assigns and manages the wireless resources for V2X communications in a centralized scheduling manner.

Denote $u_{j,k,t}^{i,o}$ as the number of points that participant i uploads regarding pillar $o \in \mathcal{O}$, using the link $j \rightarrow k$ at t-th time slot. The maximum number of points can be transmitted on link $j \rightarrow k$ at t-th time slot is known, which is denoted as $m_{j,k,t}$. Specifically, we assume there exists a self-loop link $i \rightarrow i$ with infinite capacity, i.e., participant i stores the data for further transmission. The multi-hop cooperative dissemination strategy has to consider the following constraints:

- **Link capacity:** The transmitted number of points could not exceed the link capacity, i.e.,

$$0 \leq u_{j,k,t}^{i,o} \leq |\mathcal{P}_i^o|, \quad u_{j,k,t}^{i,o} \in \mathbb{N}^+, \\ \sum_{i \in \mathcal{N}} \sum_{o \in \mathcal{O}} u_{j,k,t}^{i,o} \leq m_{j,k,t}. \tag{1}$$

- **Flow conservation:** The aggregated outgoing link flow rate at each participant equals the incoming link flow rate for the relay node.

$$\sum_{i \in \mathcal{N}} \sum_{o \in \mathcal{O}} \sum_{j \in \mathcal{N}} u_{j,k,t-1}^{i,o} = \sum_{i \in \mathcal{N}} \sum_{o \in \mathcal{O}} \sum_{j \in \mathcal{N}} u_{k,j,t}^{i,o}. \tag{2}$$

- **Channel allocation:** Each activated link should be allocated with a subchannel and a binary variable $x_{i,j,t}$ can be applied to represent the link activation status, i.e., $x_{i,j,t}$ is one if activated else zero, which can be denoted as:

$$x_{j,k,t} = sign(\sum_{i \in \mathcal{N}} \sum_{o \in \mathcal{O}} u_{j,k,t}^{i,o}), \tag{3}$$

where $sign(\cdot)$ is Signum function.
We further define $\eta_{j,k,t} \in \{0, 1, \cdots, |\mathcal{C}|\}$ as the subchannel allocation strategy, where $\eta_{j,k,t} = 0$ if link $j \rightarrow k$ does not apply any subchannel at t-th time slot, otherwise,

[2]RBs are used for the physical sidelink control channel and the physical sidelink shared channel.

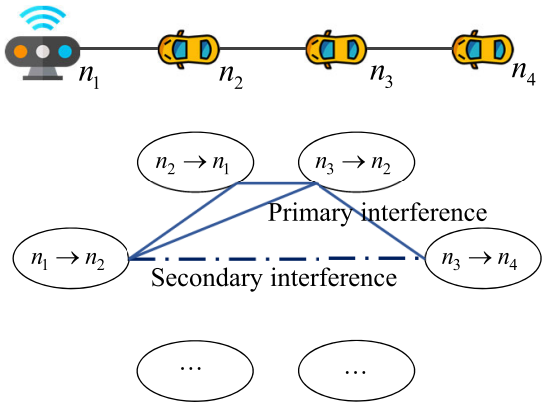


Fig. 4. Conflcit graph with primary and secondary interference.

$\eta_{j,k,t}$ is the index of the used subchannel. Each activated link can be allocated with only a subchannel, which outputs the following subchannel allocation constraint:

$$\eta_{j,k,t} \geq 1 \text{ if } x_{j,k,t} = 1. \tag{4}$$

Considering the wireless communication interferences between 5G V2X links, there exist the following two types of interferences.

1) *Primary* interference: A participant cannot act as the transmitter or receiver simultaneously;
2) *Secondary* interference: If the receiver is within the interference range of another transmitter, the two transmitters cannot use the same subchannel.

Note that secondary interference can be removed by properly allocating subchannel. Therefore, inspired by [47] and [49], at $t \in \mathcal{T}$ time slot, we define a conflict graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t^p, \mathcal{E}_t^s)$, in which each node in $\mathcal{V}_t$ is a link and each edge in $\mathcal{E}_t^p \cup \mathcal{E}_t^s$ indicates conflict between the link pairs, as shown in Fig.4. $\mathcal{E}_t^p$ and $\mathcal{E}_t^s$ represent the conflicts of the primary and secondary interference, respectively.

We denote the conflict relationship in terms of primary and secondary interference as $\mathcal{G}_t^p(i \to j, k \to l)$ and $\mathcal{G}_t^s(i \to j, k \to l)$, respectively, where it is 1 if link $i \to j$ conflicts with $k \to l$ and 0 otherwise. Therefore, the following constraint can be applied to avoid conflict between activated links:

$$\begin{cases} \eta_{i,j,t} \neq \eta_{k,l,t}, & \text{if } \begin{array}{l} x_{i,j,t} + x_{k,l,t} = 2 \text{ and} \\ \mathcal{G}_t^s(i \to j, k \to l) = 1 \end{array} \\ x_{i,j,t} + x_{k,l,t} \leq 1, & \text{if } \mathcal{G}_t^p(i \to j, k \to l) = 1. \end{cases} \tag{5}$$

With the subchannel allocation strategy $\boldsymbol{\eta} = \cup_{i\in\mathcal{N}} \cup_{j\in\mathcal{N}} \cup_{t\in\mathcal{T}} \eta_{j,k,t}$ and multi-hop transmission strategy $\boldsymbol{u} = \cup_{i\in\mathcal{N}} \cup_{o\in\mathcal{O}} \cup_{j\in\mathcal{N}} \cup_{k\in\mathcal{N}} \cup_{t\in\mathcal{T}} u_{j,k,t}^{i,o}$, the edge server can receive points for each pillar $o \in \mathcal{O}$ from all participants, which is denoted as $E(o)$ and can be denoted as:

$$E(o) = \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} \sum_{t\in\mathcal{T}} u_{j,0,t}^{i,o} + \sum_{i\in\mathcal{M}} |\mathcal{P}_i^o|. \tag{6}$$

Considering the pillar satisfaction degree defined by *Definition 2*, the edge server hopes to retrieve at least P_{max} points for each pillar. The pillar satisfaction degree over the pillar $o \in \mathcal{O}$ is defined as $g_o(\boldsymbol{u}, \boldsymbol{\eta}) = min(P_{max}, E(o))$. The cumulated satisfaction degree is defined as:

$$g(\boldsymbol{u}, \boldsymbol{\eta}) = \sum_{o\in\mathcal{O}} g_o(\boldsymbol{u}, \boldsymbol{\eta}) = \sum_{o\in\mathcal{O}} min(P_{max}, E(o)). \tag{7}$$

C. Collaborative Perception Scheduling Formulation

By optimizing the subchannel allocation strategy $\boldsymbol{\eta}$ and multi-hop transmission strategy $\boldsymbol{u}$, the collaborative perception scheduling (CPS) task requires maximizing the cumulated pillar satisfaction degree, which is defined as:

$$\begin{aligned} \textbf{P1:} \quad & \max_{\boldsymbol{u},\boldsymbol{\eta}} \ g(\boldsymbol{u}, \boldsymbol{\eta}) \\ & s.t. \ (1), (2), (3), (4), \text{and } (5). \end{aligned} \tag{8}$$

Theorem 1: CPS problem is NP-Hard.

Proof: The CPS problem contains channel allocation, link scheduling, packet scheduling, and relay selection. The link scheduling on a conflict graph in a wireless network has been proven to be NP-Hard in [47], which demonstrates that the sub-problem of the CPS problem is NP-Hard. Consequently, the CPS problem is NP-Hard. □

V. Maximum Cost Flow Reformulation and Complexity

A. CPS Reformulation

The CPS problem **P1** formulated by (8) is converted into **P2**, which is denoted as:

$$\textbf{P2:} \quad \max_{\boldsymbol{u},\boldsymbol{\eta}} \sum_{o\in\mathcal{O}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} \sum_{t\in\mathcal{T}} u_{j,0,t}^{i,o} \tag{9}$$

$$\begin{aligned} s.t. \quad & (1), (2), (3), (4), \text{and } (5), \\ & \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}} \sum_{t\in\mathcal{T}} u_{j,0,t}^{i,o} \\ & \leq max(P_{max} - \sum_{i\in\mathcal{M}} |\mathcal{P}_i^o|, 0). \end{aligned} \tag{9a}$$

*Theorem 2: Optimization **P1** and **P2** lead to the same cumulated pillar satisfaction degree.*

Proof: Please refer to Appendix A. □

Maximizing the cumulated pillar satisfaction degree has been formulated as an integer programming, which is of a large number of variables and is challenging to solve since it is NP-Hard according to *Theorem 1*. The formulated problem can also be converted into a minimum cost flow problem (MCFP) with conflict constraints by a polynomial algorithm.

B. Minimum Cost Flow Formulation of P2

We first introduce MCFP and minimum cost flow with conflict constraints (MCFPCC) problem.

*Definition 5: **MCFP.** Consider a graph network $\mathcal{F} = (\mathcal{N}_f, \mathcal{E}_f)$ with a source S and a sink T, where each edge $e_{i,j} \in \mathcal{E}_f$ is associated with a cost. The MCFP problem is to find a flow with the least total cost while satisfying the following constraints*:

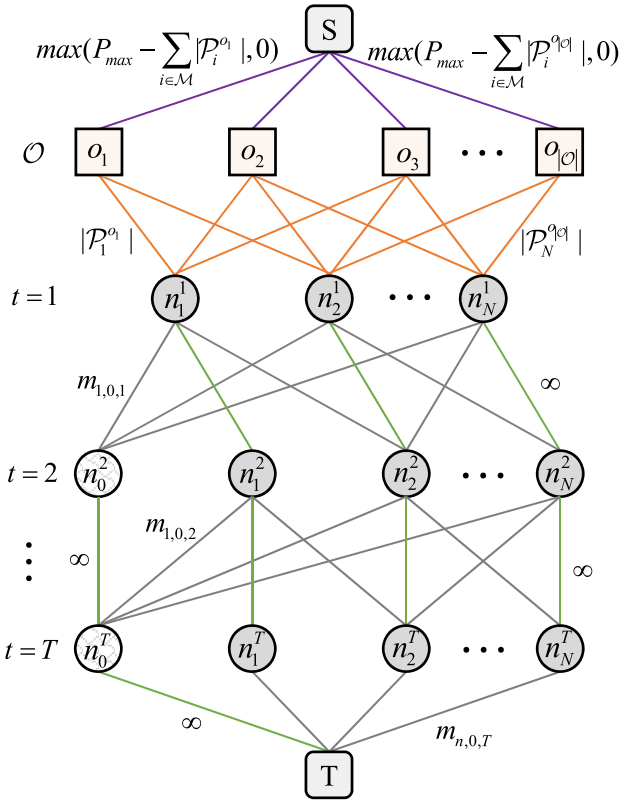


Fig. 5. Minimum cost flow formulation of CPS.

- ***Capacity constraint:*** *the flow of an edge cannot exceed its capacity;*
- ***Conservation constraint:*** *the sum of the flows entering a node must equal the sum of the flows exiting that node, except for the source and the sink.*

Definition 6: ***MCFPCC.*** *Building upon the MCFP problem, the edge pairs in the networks are in conflict, i.e., only one of the conflicting pairs can be selected to route flows. The conflicting relationship can be represented as a conflict graph.*

Theorem 3: Optimization ***P2*** *can be transformed into MCFPCC within polynomial time.*

Proof:

We construct a single source and single sink flow network denoted by $\mathcal{F}=(\mathcal{N}_f,\mathcal{E}_f)$, as shown in Fig.5, with vertex set:

$$\mathcal{N}_f = \{S,T\}\cup\mathcal{O}\cup\bigcup_{t\in\mathcal{T}}\{n_0^t,n_1^t,\cdots,n_N^t\}, \tag{10}$$

where:

- S: a virtual source node.
- T: a virtual sink node.
- $o\in\mathcal{O}$: a pillar $o\in\mathcal{O}$.
- n_i^t: participant i in the t-th time slot.

The edge set, $\mathcal{E}_f$, consists of directed edges and is constructed as follows:

- **Step 1:** For each $o\in\mathcal{O}$, there exists an edge of $max(P_{max}-\sum_{i\in\mathcal{M}}|\mathcal{P}_i^o|,0)$ capacity from S to o.
- **Step 2:** For each $o\in\mathcal{O}$ and participant $i\in\{\mathcal{N}-0\}$ at the first time slot, there exists an edge of $|\mathcal{P}_i^o|$ capacity from o to n_i^1.
- **Step 3:** For each pair (n_i^t,n_j^{t+1}), $i,j\in\mathcal{N},i\neq j$, there exists an edge of $m_{i,j,t}$ capacity from n_i^t to n_j^{t+1}.
- **Step 4:** For each pair (n_i^t,n_i^{t+1}), $i\in\mathcal{N}$, there exists an edge of infinite capacity from n_i^t to n_i^{t+1}.
- **Step 5:** For each participant $i\in\{\mathcal{N}-0\}$ at the last time slot, there exists an edge of $m_{i,0,T}$ capacity from n_i^T to T. There exists an edge of infinite capacity from n_0^T to T.

We then narrate how to map optimization **P2** into MCFPCC on the constructed graph $\mathcal{F}=(\mathcal{N}_f,\mathcal{E}_f)$. (1) is considered by MCFPCC as the capacity constraint between the edge (n_i^t,n_j^{t+1}). (2) is exactly the conservation constraint. As for (3), (4), and (5), they are mapped into conflict constraints. (9a) is mapped as the capacity constraint of edges between S and $o\in\mathcal{O}$. The objective $\max_{\boldsymbol{u},\boldsymbol{\eta}}\ \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}}u_{j,0,t}^{i,o}$ is the minimum cost flow on $\mathcal{F}=(\mathcal{N}_f,\mathcal{E}_f)$ between the source S and the sink T. □

We then analyze the complexity of transformation optimization **P2** into MCFPCC. The number of iterations for step 1 to step 5 are $|\mathcal{O}|$, $|\mathcal{O}||\mathcal{N}|$, $|\mathcal{N}|^2$, $|\mathcal{N}|^2$, and $|\mathcal{N}|$, respectively. Therefore, the complexity of the algorithm is $O(T|\mathcal{O}||\mathcal{N}|+T|\mathcal{N}|^2)$.

C. Hardness of α-Approximate Algorithms

The CPS problem is significantly complex to solve, since it is NP-Hard. Furthermore, we prove that even a polynomial time approximation algorithm with a constant performance ratio ($\alpha>1$) does not exist unless P=NP. Based on the constructed graph $\mathcal{F}=(\mathcal{N}_f,\mathcal{E}_f)$, we have *Theorem 4*.

Theorem 4: There is no polynomial time approximation algorithm for the CPS problem with a constant performance ratio unless P = NP.

Proof: Please refer to Appendix B. □

VI. Solutions

According to *Theorem 4*, it is impossible to obtain a polynomial time approximation algorithm for the CPS problem with a constant performance ratio unless P = NP. Therefore, this section is concentrated on a heuristic algorithm. We divide optimization **P2** into a minimum cost flow problem and a graph coloring problem, which are formulated as **P3** and **P4**, respectively.

$$\textbf{P3:}\quad \overline{\boldsymbol{u}}=\arg\max_{\boldsymbol{u}}\ \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}}u_{j,0,t}^{i,o}$$
$$s.t.\ (1),(2),\text{and }(9a). \tag{11}$$

$$\textbf{P4:}\quad \max_{\boldsymbol{u}\in\overline{\boldsymbol{u}},\boldsymbol{\eta}}\ \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}}u_{j,0,t}^{i,o}$$
$$s.t.\ (3),(4),\text{and }(5). \tag{12}$$

Optimization **P3** ignores the conflict relationship between links, which purely computes the minimum cost flow from the source S to the sink T. Therefore, **P3** can be converted

into a linear programming problem as follows:

$$\overline{\boldsymbol{u}} = \arg\max_{\boldsymbol{u}} \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} \tag{13}$$

$$s.t.\ 0 \le u^{i,o}_{j,k,t} \le |\mathcal{P}^o_i|, \tag{14}$$

$$\sum_{i\in\mathcal{N}}\sum_{o\in\mathcal{O}} u^{i,o}_{j,k,t} \le m_{j,k,t}. \tag{15}$$

$$\sum_{i\in\mathcal{N}}\sum_{o\in\mathcal{O}}\sum_{j\in\mathcal{N}} u^{i,o}_{j,k,t} = \sum_{i\in\mathcal{N}}\sum_{o\in\mathcal{O}}\sum_{j\in\mathcal{N}} u^{i,o}_{k,j,t}, \tag{16}$$

$$\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} \le max(P_{max} - \sum_{i\in\mathcal{M}} |\mathcal{P}^o_i|, 0). \tag{17}$$

Exploiting linear programming (13) results $\overline{\boldsymbol{u}}$, we can obtain the set of scheduled links $\mathcal{V}_t$ at $t \in \mathcal{T}$ time slot, i.e.,

$$\mathcal{V}_t = \{n^t_j \to n^t_k \Big| \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}} \overline{u}^{i,o}_{j,k,t} > 0\}. \tag{18}$$

We then selected a partial of links considering the subchannel allocation constraints at each time step. A two-dimensional graph coloring algorithm (TDGC) is proposed to address this problem.

At each time step, the weight of a link is defined as the number of points transmitted on the link, i.e.,

$$W(n^t_j \to n^t_k) = \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}} \overline{u}^{i,o}_{j,k,t}. \tag{19}$$

Then, considering the *primary* and *secondary* interference, we select a subset of non-conflict links. Based on $\mathcal{V}_t$, the corresponding conflict graph is $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}^p_t, \mathcal{E}^s_t)$, where each vertex in $\mathcal{V}_t$ is assigned with a weight. At first, we have an empty set ϖ_t, and the node with the highest weight $n^t_j \to n^t_k = argmax_{\delta\in\mathcal{V}_t} W(\delta)$ is selected and added into ϖ_t. Then, delete all its neighbors $\beta^p_t(n^t_j \to n^t_k)$ that are conflict by the primary conflict. As for the set of neighbors $\beta^s_t(n^t_j \to n^t_k)$ that is conflict by the secondary conflict, delete the neighbors with the smallest weight until only $C-1$ neighbors are left. After that, loop over the above steps until $\mathcal{V}_t$ is empty. Finally, for each path from S to T, if an edge in this path is deleted, then, this path should be deleted, i.e., if link $n^t_j \to n^t_k$ is deleted, $\overline{u}^{i,o}_{j,k,t} = 0, \forall i \in \mathcal{N}, \forall o \in \mathcal{O}$.

The TDGC algorithm sequentially selects the non-conflict links for each time step.

Theorem 5: TDGC algorithm outputs a subset of links at least of $\frac{\sum_{\delta_0\in\mathcal{V}_t} W(\delta_0)}{2-C+\gamma_p+\gamma_s}$, *where* $\gamma_p = \max_{\delta_0} |\beta^p_t(\delta_0)|$, $\gamma_s = \max_{\delta_0} |\beta^s_t(\delta_0)|$, *is the maximum number of neighbor that have primary and secondary interfere, respectively. We assume* $C < \gamma_s + 1$.

Proof: Please refer to Appendix C. □

VII. Implementation and Experiment

In this section, we implement and evaluate EdgeCooper by conducting large-scale co-simulation exploiting deep-learning framework PyTorch, autonomous driving simulator CARLA [50], transportation traffic simulator SUMO and communication simulator NS3 [51].

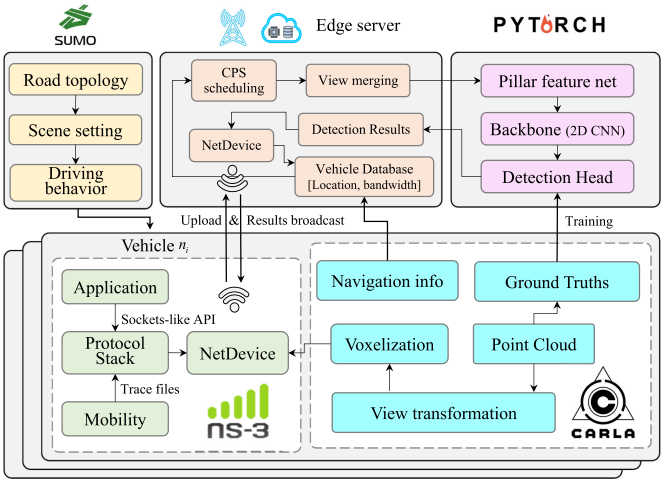


Fig. 6. Evaluation of collaborative perception in vehicular edge computing using co-simulation among CARLA, SUMO, NS3, and PyTorch.

A. Dataset and Simulation Architecture

Our data are collected from eight default town provided by CARLA, and consists of 28 scenes, where each scene contains traffic flow at a certain intersection, originating from OpenV2V [52]. As shown in Fig.6, in each scene, SUMO is firstly used to produce numerically-realistic traffic flow, for its capability of handling large-scale and realistic traffic flows. Then, CARLA is employed to get realistic 64-channel LiDAR streams, from multiple vehicles located in the same geographical area. The simulated LiDAR is streamed at 20 Hz and recorded at 10 Hz. The LiDAR points are coped within the region of $[-70, 70] \times [-40, 40] \times [-3, 1]$ meters defined in the ego-vehicle XYZ coordinate. We set the width/length (X/Y) of each pillar as 0.4 meter, and the height (Z) as 4 meter following [43]; After that, NS3 is applied to simulate vehicular communications, implementing the CPS via the NS3 applications module. We implement the CPS scheduling algorithm based on NS3 simulations, i.e., the edge server collects points data from all participants. The maximum number of points in each grid P_{max} is 32 according to [43]. Finally, the edge server adopts the PyTorch framework for implementing PointPillar and conducts object detection based on aggregated point clouds. We further adopt Average Precision (AP) at different Intersection-over-Union (IoU) thresholds to assess different models.

The real-time 3D detection adopts PointPillars, a state-of-the-art open-source 3D object detection framework. We adopt the same training and testing parameters of PointPillars as in [43]. The experiments are conducted on a server equipped with an Intel(R) Xeon(R) CPU E5-2698 v4 at 2.20GHz, an NVIDIA Tesla V100 GPU, and 512GB of DDR4 RAM. The object detection can achieve real-time processing at 50 FPS and Edgecooper achieves end-to-end latency of 100 ms.

B. Object Detection Model of EdgeCooper

EdgeCooper eliminates the expensive 3D convolutions in [53] and advocates 2D convolutional architecture for higher speed and better computational efficiency, as shown in Fig.7.

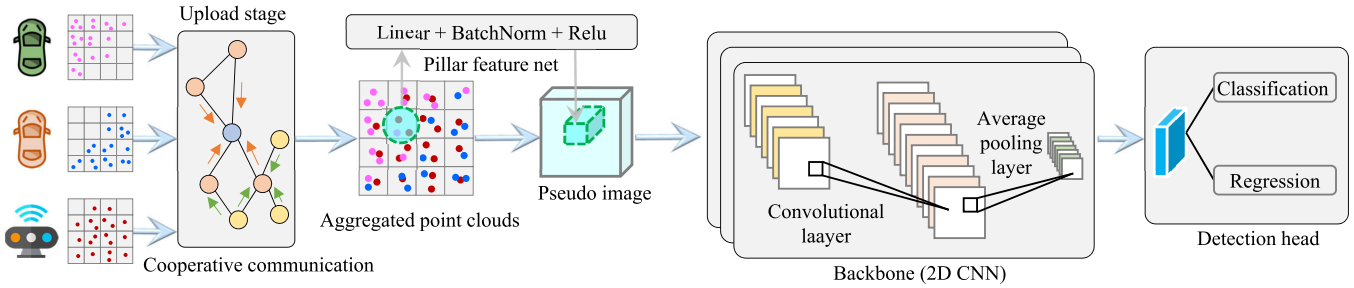


Fig. 7. The neural network architecture of EdgeCooper.

Therefore, we first convert the point cloud to a pseudo-image. In the upload stage, a network-aware and scalable transmission strategy based on maximum flow theory is proposed to advocate the cooperative sharing of point clouds among participants. This stage gives rise to aggregated point clouds at the edge server. The aggregated point clouds consist of pillars and a pillar is a voxel with unlimited spatial extent in the z direction and hence there is no need for a hyperparameter to control the binning in the z dimension. If a pillar holds too much data, e.g., larger than P_{max}, the data is randomly sampled. Conversely, if a pillar has too few point clouds, zero padding is applied. The points in each pillar are then decorated (augmented) to a tensor with 9 dimensional according to [43].

For each point, a linear layer is applied followed by BatchNorm and ReLU to process each point, resulting in a $(S_{dimension}, S_{pillars}, P_{max})$ sized tensor, where $S_{dimension}$ is the new dimension of each point, and $S_{pillars}$ is the number of pillars. This tensor is followed by a max operation over the channels to create an output tensor of size $(S_{dimension}, S_{pillars})$. Note that the linear layer can be formulated as a 1×1 convolution across the tensor resulting in very efficient computations. Once encoded, the tensor of size $(S_{dimension}, S_{pillars})$ are scattered back to the original pillar locations to create a pseudo-image of size $(S_{dimension}, H, W)$ where H and W indicate the height and width of the interested areas. The pseudo-image is followed by a 2D convolutional network backbone network for extracting efficient features. After that, a detection head is applied for classification and regression, as shown in Fig.7.

C. Communication Settings

In each scene, there exist $7-50$ vehicles, in which only a small part of vehicles (with high-level autonomous driving) are installed with LiDARs, and the rest vehicles (without high-level autonomous driving while having communication capability) cannot percept the environment but are willing to join the collaboration perception services, i.e., relay the point cloud data. The reason behind it is that all vehicles could benefit from this service for cooperative safety. All vehicles are within the coverage of the base station. The *upload* stage has a duration of 75ms. We set up the simulation following the evaluation methodology defined in 3GPP TR36.885 [2], which describes the detailed channel models for V2X links. Vehicular communications are simulated according to the PHY and MAC layers of the 5G C-V2X protocols: NR-V2X, based on 3GPP Rel-16/17 [46]. Our simulation scenarios are designed to represent a 20 MHz 5G V2X system, with 5.9 GHz carrier frequency. Following [46], 5G V2X channel is divided into ten 2 MHz subchannels. The major network simulation parameters are listed in Tab.II.

TABLE II
SUMMARY OF SIMULATION PARAMETERS

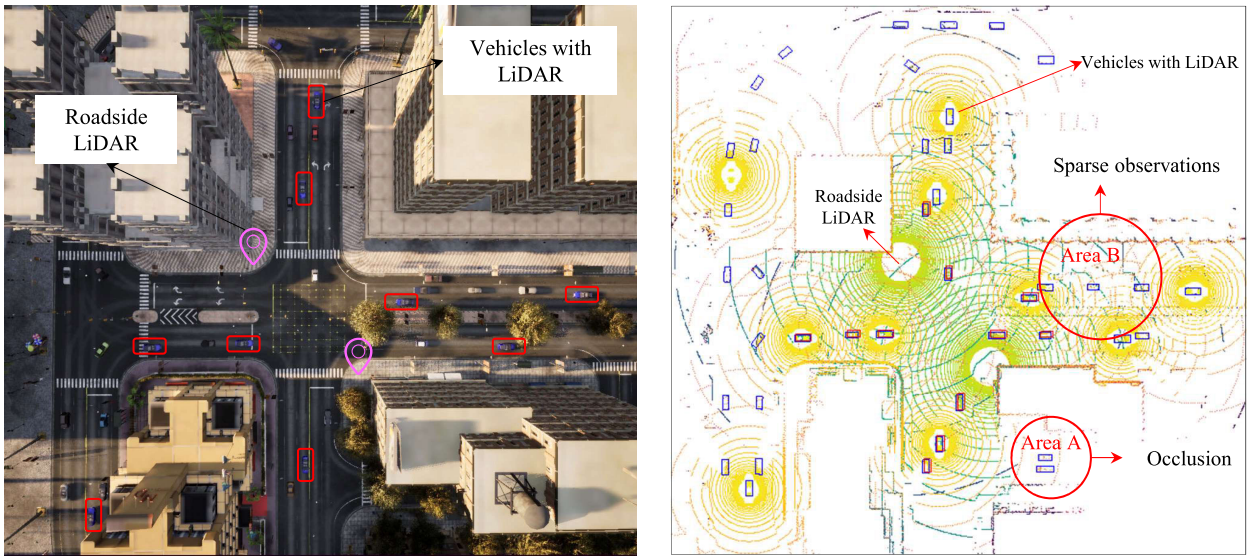
Parameter	Value
Frequency band	5.9 GHz
Bandwidth	20 MHz
Number of subchannels C	10
Transmission power	23 dbm
Number of vehicles	7-50
Travel velocity	0 - 40 km/h
T_0	5 ms, 15 ms, 25 ms
Path loss model	$128.1 + 37.6log_{10}(d)$
Shadowing distribution	Log-normal
Shadowing standard deviation	8 dB
Fast fading	Rayleigh fading
Noise power	-114 dbm

D. Comparison Algorithms

In the experiments, we compare EdgeCooper with the following algorithms: 1) Edge-assisted multi-vehicle perception (EMP) [40]. EMP exploits a disjoint spatial partition of the environment and transmits the partitioned sensor data to the edge, thus creating a global view and achieving a balance between bandwidth consumption and data quality. 2) Maximum capacity induced transmission (MC). In each time step, the feasible link with maximum link capacity is scheduled for transmission. 3) Without relay (WR). Each participant transmits data into the edge directly without multi-hop cooperative communication.

E. Qualitative Evaluation

To understand how EdgeCooper can enhance perception accuracy and robustness, as well as enlarge perception range, we visualize a complex intersection in a city with 10 vehicles equipped with LiDARs, 2 roadside LiDARs, and other 50 vehicles, as shown in Fig.8. Fig.8(a) shows the complex intersection in a city, where the red box means vehicle installed with LiDARs. In such an intersection, the roadside LiDARs have limited detection range and are not able to detect areas with occlusion (occluded by trees, and trucks). Therefore, it depends on EdgeCooper for seeking a larger



(a) A complex intersection in CARLA town 10, where red box means vehicles equipped with LiDARs.

(b) Collaborative perception at EdgeCooper, where red and blue box can be detected by Non-Cooper and EdgeCooper, respectively.

Fig. 8. Visualization of EdgeCooper in enhancing perception accuracy and robustness, as well as enlarging perception range.

perception range and see-through perception, which results in higher cooperative awareness for related participants. The edge computing node could aggregate the point clouds of both roadside LiDARs as well as vehicular LiDARs, which is shown in Fig.8(b). The point clouds perceived by roadside LiDARs and vehicular LiDARs are shown in green and yellow colors, respectively. The red and blue box means the vehicle detected by EdgeCooper and Non-Cooper algorithms, respectively.

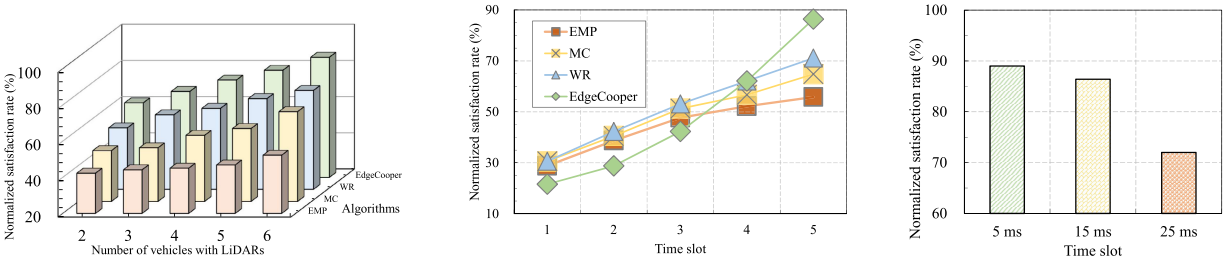
We draw special attention to marked areas. In area A in Fig.8(b), the roadside LiDARs have extremely sparse observation, since it has a few point clouds. Therefore, depending on the ability of roadside LiDARs alone is hard to perform accurate perceptions over this area. In area B in Fig.8(b), the roadside LiDARs cannot percept it at all, since it does not have any point cloud. Therefore, depending on the ability of the roadside LiDARs alone could not perceive this area. However, powered with EdgeCooper, the edge server can successfully percept both area A and area B by cooperatively uploading individual point clouds to the edge to build a complete and holistic view. The detection results are shown in Fig.8(b). The results illustrate that EdgeCooper can see-through occlusion (area A in Fig.8(b)), as well as enhance perception accuracy and robustness(area B in Fig.8(b)).

F. Quantitative Evaluation

EdgeCooper is able to provide real-time enhanced perception under various processing workloads and network conditions. We first evaluate the pillar satisfaction degree under different time slots and a different number of vehicles. Then, we evaluate the extended perception range of EdgeCooper under a different number of vehicles. Finally, the detection performance with different time slot duration is evaluated.

Fig.9 shows the normalized satisfaction rate, which is the cumulated satisfaction degree of a certain algorithm divided by the optimized value of (9). Fig.9(a) shows the normalized satisfaction rate of the four schemes when there exist $2-6$ vehicles with LiDARs. The results demonstrate that more vehicles lead to a higher satisfaction rate. The reasons behind this are 1) each pillar can be perceived by more vehicles and therefore, each pillar can have a higher satisfaction degree; 2) more vehicles participate in relaying the data and could upload more data into the edge. Meanwhile, EdgeCooper achieves the higher satisfaction rate and EMP achieves the lowest. EMP shares non-overlapping data to the edge server, which is the least redundant while destroying complementarity. Specifically, each vehicle N_i can observe set of points $\mathcal{P}_i^o$ for pillar o and $|\mathcal{P}_i^o| < P_{max}$. In EMP, the vehicle with the largest $|\mathcal{P}_i^o|$ is scheduled to transmit. However, it would not fully satisfy the perception requirements. As in EdgeCooper, each vehicle is scheduled to share partial data, and therefore, the total transmitted data can exceed P_{max}. Therefore, EMP achieves the lowest satisfaction rate. Comparing EdgeCooper with WR, the satisfaction rate gain comes from the multi-hop cooperative communications, which could potentially bring about higher data rates, and more efficient use of the wireless medium. Comparing EdgeCooper with MC, the satisfaction rate gain comes from minimum cost flow and our proposed TDGC algorithm. Since in MC, the scheduling results in different time slots are absent of coordination, and a good scheduling strategy might become poor in the sequential time slot.

Fig.9(b) shows the normalized satisfaction rate at different time slots. In the beginning, EdgeCooper has the lowest rate, however, it has the highest growth rate at the end. EdgeCooper schedules multi-hop cooperative transmission, which means that data might be relayed to other nodes and therefore leads to a lower satisfaction rate in the beginning. Fig.9(c) shows

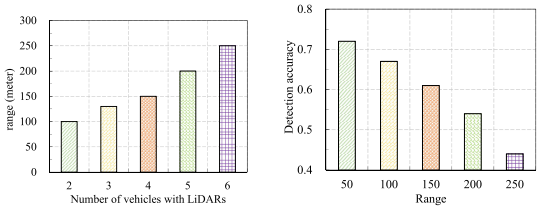


(a) Different number of vehicles with LiDARs (b) Different time slots (c) Different time slot duration

Fig. 9. Normalized satisfaction rate of EdgeCooper.

TABLE III
DETECTION ACCURACY OF EDGECOOPER AS COMPARED WITH EMP AND NON-COOPER

Metrics	IoU=0.3	IoU=0.5	IoU=0.7
Non-Cooper	0.74	0.69	0.56
EMP	0.80	0.74	0.60
EdgeCooper	0.89	0.84	0.69



(a) Detection range (b) Accuracy with range

Fig. 10. EdgeCooper performance.

the normalized satisfaction rate at different time slot duration. Note that the duration for the upload stage is 75 ms and therefore, a larger time slot duration means less number of slots. The results show that a smaller time slot duration leads to a higher satisfaction rate. The reason is that a small time slot duration means more sophisticated exploitation of available bandwidth.

To measure the detection accuracy, we calculate IoU between the detection results (locations and dimensions of detected object bounding boxes) and the ground truth. Detection accuracy at different IoU thresholds is adopted to evaluate the performance. Tab.III shows the detection accuracy of EdgeCooper as compared with EMP and Non-Cooper (perform detection using the roadside LiDARs without any vehicles' LiDARs). The results show that EMP performs better than Non-Cooper, i.e., EMP achieves a 0.6 improvement than Non-Cooper in terms of IoU=0.3, and 0.4 improvement in terms of IoU=0.7. Compared with Non-Cooper, EdgeCooper improves $(0.89-0.74)/0.74 = 20.27\%$, $(0.84-0.69)/0.69 = 21.74\%$, and $(0.69 - 0.56)/0.56 = 23.32\%$ in terms of IoU=0.3, IoU=0.5, and IoU=0.7, respectively.

EdgeCooper could not only enhance perception accuracy and robustness but also enlarge perception range. The perception range is defined using the furthest object that can be detected. Fig.10(a) shows the detection range of EdgeCooper when there exists a different number of vehicles equipped with LiDARs. The results show that more vehicles would lead to a larger detection range, e.g., the communication range is extended to 250 m when there exist 6 vehicles that are equipped with LiDARs, which is nearly $250/70 = 3.57$ times of single vehicle's perception range. However, the objects located at different distance enjoys a distinct detection accuracy, as shown in Fig.10(b), i.e., an object at a near distance can be detected at a higher accuracy.

Tab.IV provides a thorough comparison between the proposed EdgeCooper and the existing approaches, e.g., Cooper [54] and DiscoNet [42], specifically focusing on the trade-off

TABLE IV
DETECTION ACCURACY OF EDGECOOPER AS COMPARED WITH EMP AND NON-COOPER

	Bandwidth (Mb/s)	Accuracy
Cooper [24]: Complete raw sensor	76.8	0.69
DiscoNet [43]: Intermediate features	11.2	0.60
EdgeCooper: Partial sensor data	6.7	0.69

between detection performance and communication bandwidth. Cooper advocates for the sharing of complete raw sensor data, whereas DiscoNet prefers the sharing of intermediate features extracted using a backbone network. The comparison reveals the following insights: i) Cooper consumes the highest bandwidth due to the large volume of raw sensor data, resulting in the highest detection accuracy; ii) DiscoNet, utilizing a 16-times feature compression by autoencoder, achieves a significant reduction in communication volume. However, this reduction in bandwidth comes at the cost of lower accuracy, as the intermediate features lose information, leading to reduced performance; and iii) Our proposed EdgeCooper transmits partial raw sensor data, employing a strategy that enhances complementarity and minimizes redundancy in the shared point clouds. Consequently, EdgeCooper achieves the smallest communication volume while maintaining higher accuracy compared to DiscoNet. These results clearly demonstrate that EdgeCooper effectively utilizes fewer communication resources while achieving superior detection accuracy.

VIII. CONCLUSION

This paper proposed EdgeCooper, a novel edge-assisted multi-vehicle perception system based on vehicular communications and vehicular edge computing, enabling collaborative perception and enhancing vehicles' awareness over surrounding environments. EdgeCooper advocates a novel voxelization-based strategy, which establishes a connection between

metrics in communications (e.g., throughput, efficiency) and metrics in object detection (e.g., accuracy). It efficiently schedules the sharing of complementarity-enhanced and redundancy-minimized raw sensor data with an edge server, using multi-hop cooperative communications. EdgeCooper is demonstrated to enhance perception accuracy and robustness, as well as enlarge perception range, enabling or boosting a wide range of cooperative sensing applications. By conducting a joint simulation of SUMO, CARLA, NS3, and PyTorch, results have demonstrated that, compared to a single vehicle's perception, EdgeCooper can extend up to 3.6 times detection range and improve perception accuracy by 20%.

EdgeCooper endeavor sits at the confluence of three transformational technologies, intelligent transportation, wireless communications, and artificial intelligence. In the future, EdgeCooper is expected to expand its support for a wider range of vehicular sensors, including cameras, further augmenting its perception capabilities. This integration of cameras into the EdgeCooper system will enable vehicles to capture visual information, fostering a more comprehensive and detailed understanding of the surrounding environment. Additionally, EdgeCooper will take into account factors like mobility over topology and address the challenges posed by link interference. EdgeCooper will employ advanced algorithms and protocols to ensure uninterrupted communication and robust connectivity.

APPENDIX A
PROOF OF *Theorem 2*

Proof: We first show constraint (9a) for any pillar $o \in \mathcal{O}$ would not deteriorate the optimal objective value of **P1**. Suppose $\bar{\boldsymbol{u}}$ is the optimal solution to **P1**. If there exists a pillar $o \in \mathcal{O}$ whose $E(o)$ is greater than P_{max}, i.e., $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} + \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i| > P_{max}$, then, there alway exists another solution $\hat{\boldsymbol{u}}$ that satisfies 1) the difference between $\bar{\boldsymbol{u}}$ and $\hat{\boldsymbol{u}}$ only lies in $\hat{u}^{i,o}_{j,k,t}$; and 2) $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} + \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i| = P_{max}$. $\hat{u}^{i,o}_{j,k,t}$ can be easily computed by gradually decreasing $\bar{u}^{i,o}_{j,k,t}$, i.e., strategy $\hat{\boldsymbol{u}}$ transmits only a subset data of strategy $\bar{\boldsymbol{u}}$. Meanwhile, $\hat{\boldsymbol{u}}$ also satisfies (1), (2), (3), (4), and (5). In this case, both $\bar{\boldsymbol{u}}$ and $\hat{\boldsymbol{u}}$ achieves the same objective value, i.e., $g(\bar{\boldsymbol{u}}, \boldsymbol{\eta}) = g(\hat{\boldsymbol{u}}, \boldsymbol{\eta})$.

Therefore, if $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} + \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i| \leq P_{max}$ for any pillar $o \in \mathcal{O}$, there exists a solution to achieve the optimum, which produces the following inequality:

$$0 \leq \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} \leq P_{max} - \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i|. \tag{20}$$

Therefore, we can derive $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} \leq max(P_{max} - \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i|, 0)$, which is exactly constraint (9a).

If $\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} + \sum_{i\in\mathcal{M}}|\mathcal{P}^o_i| \leq P_{max}$, the objective of **P1** can also be denoted as:

$$\max_{\boldsymbol{u},\boldsymbol{\eta}} \sum_{o\in\mathcal{O}} min(P_{max}, E(o)) = \max_{\boldsymbol{u},\boldsymbol{\eta}} \sum_{o\in\mathcal{O}} E(o) \tag{21}$$

$$= \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t} + \sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{M}}|\mathcal{P}^o_i|. \tag{22}$$

Note that $\mathcal{P}^o_i, i \in \mathcal{M}$ is a constant since the edge server could retrieve all points generated by roadside LiDARs. Therefore, the objective becomes $\sum_{o\in\mathcal{O}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{t\in\mathcal{T}} u^{i,o}_{j,0,t}$. □

APPENDIX B
PROOF OF *Theorem 4*

Proof: CPS problem can be converted into a MCFPCC within polynomial time and vice versa. Therefore, this theorem holds if there is no polynomial time approximation algorithm for MCFPCC with a constant performance ratio unless P = NP. We achieve it by prove that α-approximated MCFPCC algorithm is equivalent to path avoiding forbidden pairs (PAFP), which is a known NP-complete problem [55].

Definition 7: ***Path avoiding forbidden pairs (PAFP).*** *Given a graph $\mathcal{F} = (\mathcal{N}_\mathcal{F}, \mathcal{E}_\mathcal{F})$ with two fixed vertices $S, T \in \mathcal{N}_\mathcal{F}$ and a set of pairs of vertices $\mathcal{E}_c \subset (\mathcal{N}_\mathcal{F} \times \mathcal{N}_\mathcal{F}), \mathcal{E}_c \subset \mathcal{E}_\mathcal{F}$, PAFP aims to find a path from S to T that contains at most one vertex from each pair in $\mathcal{E}_c$, or to recognize that such path does not exist. The pairs in the set $\mathcal{E}_c$ are called forbidden pairs.*

A formal definition to PAFP is introduced in *Definition 7*. PAFP is a NP-complete problem [55]. The proof of *Theorem 4* is achieved by reduction from the arc variant of PWFP. An instance of MCFPCC can be constructed from any instance of PAFP such that a yes/no answer to the question of whether the directed graph $\mathcal{F}$ has a directed S,T-path that contains at most one arc from each pair in $\mathcal{E}_c$ can be obtained from any conflict-free flow on the constructed MCFPCC instance, whose total flow cost is within α times of the minimum conflict-free total flow cost.

We then introduce how to construct an instance of MCFPCC from any instance of PAFP. Given PAFP instance $\mathcal{F} = (\mathcal{N}_\mathcal{F}, \mathcal{E}_\mathcal{F})$, with S, T, and forbidden pairs $\mathcal{E}_c$, we construct a complete network $\mathcal{H} = (\mathcal{N}_\mathcal{H}, \mathcal{E}_\mathcal{H})$. The constructed graph $\mathcal{H}$ has the same vertex set, i.e., $\mathcal{N}_\mathcal{H} = \mathcal{N}_\mathcal{F}$. Furthermore, each vertex $n_i \in \mathcal{N}_\mathcal{H}$ has the supply or demand values as:

$$b(n_i) = \begin{cases} 1, & \text{if } n_i = S, \\ -1, & \text{if } n_i = T, \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

Each edge $e_{i,j} \in \mathcal{E}_\mathcal{H}$ has unit capacity. The cost $C(e_{i,j})$ of each edge $e_{i,j}$ is defined as:

$$C(e_{i,j}) = \begin{cases} 1, & \text{if } e_{i,j} \in \mathcal{E}_c, \\ |\mathcal{E}_\mathcal{F}|(1+\alpha), & \text{if } e_{i,j} \in \mathcal{E}_\mathcal{H} - \mathcal{E}_c. \end{cases} \tag{24}$$

A conflict graph can be applied to represent conflicting (forbidden) edge pairs in $\mathcal{E}_c$.

In the constructed complete network $\mathcal{H} = (\mathcal{N}_\mathcal{H}, \mathcal{E}_\mathcal{H})$, each edge has integer capacity since each edge has unit capacity. Therefore, according to the Integrality Theorem in network flow theory [56], the feasible flow of any local optimal solution should be integer, since network flow problem posed with integer capacities on edges always has a maximum flow in which the flow on every edge is an integer. Furthermore, both the source node and the sink has unit supplying/demand value, the feasible flow should be 1, which forms the S,T-path in $\mathcal{H}$.

PAFP aims to find a path from S to T that contains at most one vertex from each pair in $\mathcal{E}_c$, and such path could also be treated as the conflict-free flow in $\mathcal{H} = (\mathcal{N}_\mathcal{H}, \mathcal{E}_\mathcal{H})$. On the other hand, in the constructed complete network $\mathcal{H} = (\mathcal{N}_\mathcal{H}, \mathcal{E}_\mathcal{H})$, a α-approximated MCFPCC algorithm aims to find a S,T-path without any conflicts, i.e., the pairs in the set $\mathcal{E}_c$ are not selected simultaneously. However, such path could be consisted of edges in $\mathcal{E}_\mathcal{H} - \mathcal{E}_c$.

We assume that there exists a α-approximated MCFPCC algorithm. We could use it to compute a conflict-free feasible flow f^* in the constructed complete network $\mathcal{H} = (\mathcal{N}_\mathcal{H}, \mathcal{E}_\mathcal{H})$. The total cost $C(f^*)$ of the computed feasible flow is at most α times of the optimal value (minimum cost). If $C(f^*)$ is larger than or equal to $|\mathcal{E}_\mathcal{F}|(1+\alpha)$, this implies that the optimal value is at least:

$$\frac{|\mathcal{E}_\mathcal{F}|(1+\alpha)}{\alpha} = |\mathcal{E}_\mathcal{F}| + \frac{|\mathcal{E}_\mathcal{F}|}{\alpha}.$$

It implies that the minimum cost conflict-free flow is a S,T-path in $\mathcal{H}$ with at least one edge not in $\mathcal{E}_\mathcal{F}$. Therefore, it implies $\mathcal{F}$ does not have a conflict-free directed S,T-path. On the other hand, if $C(f^*)$ is less than $|\mathcal{E}_\mathcal{F}|(1+\alpha)$, the conflict-free directed S,T-path f^* is a subgraph of $\mathcal{F} = (\mathcal{N}_\mathcal{F}, \mathcal{E}_\mathcal{F})$. This case occurs for sure when the approximation algorithm guarantees an α-approximate feasible flow.

□

APPENDIX C
PROOF OF *Theorem 5*

Proof: Considering the conflict graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t^p, \mathcal{E}_t^s)$, if a node $\delta_0 \in \mathcal{V}_t$ with maximum weight is selected, all its neighbors $\beta_t^p(\delta_0)$ that is conflict by primary interference is deleted, therefore,

$$W(\delta_0) \geq \frac{1}{|\beta_t^p(\delta_0)|} \sum_{\delta_1 \in \beta_t^p(\delta_0)} W(\delta_1). \tag{25}$$

As for the neighbors $\beta_t^s(\delta_0)$ that is conflict by secondary interference, $C-1$ neighbors are kept. Denote the set of deleted neighbors as $\beta_t^{s-}(\delta_0)$. Therefore, we have

$$W(\delta_0) \geq \frac{1}{|\beta_t^{s-}(\delta_0)|} \sum_{\delta_1 \in \beta_t^{s-}(\delta_0)} W(\delta_1). \tag{26}$$

$$\begin{aligned}
\sum_{\delta_0 \in \mathcal{V}_t} W(\delta_0) &= \sum_{\delta_0 \in \varpi_t} \Big(W(\delta_0) + \sum_{\delta_1 \in \beta_t^p(\delta_0)} W(\delta_1) \\
&\quad + \sum_{\delta_1 \in \beta_t^{s-}(\delta_0)} W(\delta_1) \Big) \\
&\leq \sum_{\delta_0 \in \varpi_t} \Big(W(\delta_0) + W(\delta_0) \cdot |\beta_t^p(\delta_0)| \\
&\quad + W(\delta_0) \cdot |\beta_t^{s-}(\delta_0)| \Big) \\
&\leq \sum_{\delta_0 \in \varpi_t} W(\delta_0) + \max_{\delta_0} |\beta_t^p(\delta_0)| \cdot \sum_{\delta_0 \in \varpi_t} W(\delta_0) + \\
&\max_{\delta_0} |\beta_t^{s-}(\delta_0)| \cdot \sum_{\delta_0 \in \varpi_t} W(\delta_0) \\
&\leq (1 + \max_{\delta_0} |\beta_t^p(\delta_0)| \\
&\quad + \max_{\delta_0} |\beta_t^{s-}(\delta_0)|) \sum_{\delta_0 \in \varpi_t} W(\delta_0) \\
&= (2 - C + \max_{\delta_0} |\beta_t^p(\delta_0)| \\
&\quad + \max_{\delta_0} |\beta_t^s(\delta_0)|) \sum_{\delta_0 \in \varpi_t} W(\delta_0).
\end{aligned} \tag{27}$$

Here $\gamma_p = \max_{\delta_0} |\beta_t^p(\delta_0)|$, and $\gamma_s = \max_{\delta_0} |\beta_t^s(\delta_0)|$ is the maximum number of neighbor while have primary and secondary interfere, respectively. Furthermore,

$$\max_{\delta_0} |\beta_t^s(\delta_0)| = \max_{\delta_0} |\beta_t^{s-}(\delta_0)| + C - 1, \tag{28}$$

Therefore, (29) can be rewritten as

$$\sum_{\delta_0 \in \mathcal{V}_t} W(\delta_0) \leq (2 - C + \gamma_p + \gamma_s) \sum_{\delta_0 \in \varpi_t} W(\delta_0). \tag{29}$$

□

REFERENCES

[1] T. G. R. Reid et al., "Localization requirements for autonomous vehicles," 2019, *arXiv:1906.01061*.

[2] Z. Wang, Z. Liu, Y. Shen, A. Conti, and M. Z. Win, "Location awareness in beyond 5G networks via reconfigurable intelligent surfaces," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 7, pp. 2011–2025, Jul. 2022.

[3] D. D. Yoon, B. Ayalew, and G. G. M. N. Ali, "Performance of decentralized cooperative perception in V2V connected traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6850–6863, Jul. 2022.

[4] F. Raviglione et al., "From collaborative awareness to collaborative information enhancement in vehicular networks," *Veh. Commun.*, vol. 36, Aug. 2022, Art. no. 100497.

[5] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, document ETSI EN 302 637-2 V1.3.2, 2014.

[6] R. Liu, J. Wang, and B. Zhang, "High definition map for automated driving: Overview and analysis," *J. Navigat.*, vol. 73, no. 2, pp. 324–341, Mar. 2020.

[7] X. Zhou et al., "Edge-guided recurrent positioning network for salient object detection in optical remote sensing images," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 539–552, Jan. 2023.

[8] H. Qiu, F. Ahmad, F. Bai, M. Gruteser, and R. Govindan, "AVR: Augmented vehicular reality," in *Proc. 16th Annu. Int. Conf. Mobile Syst., Appl., Services*, Jun. 2018, pp. 81–95.

[9] Q. Xu, Y. Zhong, and U. Neumann, "Behind the curtain: Learning occluded shapes for 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 3, pp. 2893–2901.

[10] Wikipedia. (2009). *List of Self-Driving Car Fatalities*. [Online]. Available: http://en.wikipedia.org/wiki/List_of_self-driving_car_fatalities

[11] L. Liu et al., "Computing systems for autonomous driving: State of the art and challenges," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6469–6486, Apr. 2021.

[12] G. Luo, H. Zhang, Q. Yuan, and J. Li, "Complementarity-enhanced and redundancy-minimized collaboration network for multi-agent perception," in *Proc. 30th ACM Int. Conf. Multimedia*, Oct. 2022, pp. 3578–3586.

[13] H. Zhang, G. Luo, Y. Cao, Y. Jin, and Y. Li, "Multi-modal virtual-real fusion based transformer for collaborative perception," in *Proc. IEEE 13th Int. Symp. Parallel Architectures, Algorithms Program. (PAAP)*, Nov. 2022, pp. 1–6.

[14] J. Wang et al., "F-transformer: Point cloud fusion transformer for cooperative 3D object detection," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2022, pp. 171–182. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-031-15919-0_15

[15] H. Zhou, N. Cheng, Q. Yu, X. S. Shen, D. Shan, and F. Bai, "Toward multi-radio vehicular data piping for dynamic DSRC/TVWS spectrum sharing," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 10, pp. 2575–2588, Oct. 2016.

[16] H. Zhou et al., "Spatial coordinated medium sharing: Optimal access control management in drive-thru Internet," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2673–2686, Oct. 2015.

[17] S. V. Balkus, H. Wang, B. D. Cornet, C. Mahabal, H. Ngo, and H. Fang, "A survey of collaborative machine learning using 5G vehicular communications," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 1280–1303, 2nd Quart., 2022.

[18] K.-L. Wright, P. Sakulkar, B. Krishnamachari, and F. Bai, "VESPER: A real-time processing framework for vehicle perception augmentation," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 1–6.

[19] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds," in *Proc. 4th ACM/IEEE Symp. Edge Comput.*, Nov. 2019, pp. 88–100.

[20] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2*, document. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/103300_103399/103324/02.01.01_60/ts_103324v020101p.pdf

[21] G. Luo, Q. Yuan, J. Li, S. Wang, and F. Yang, "Artificial intelligence powered mobile networks: From cognition to decision," *IEEE Netw.*, vol. 36, no. 3, pp. 136–144, May 2022.

[22] Y. Cai et al., "YOLObile: Real-time object detection on mobile devices via compression-compilation co-design," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 2, pp. 955–963.

[23] E. Arnold, M. Dianati, R. de Temple, and S. Fallah, "Cooperative perception for 3D object detection in driving scenarios using infrastructure sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 1852–1864, Mar. 2022.

[24] S. Aoki, T. Higuchi, and O. Altintas, "Cooperative perception with deep reinforcement learning for connected vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 328–334.

[25] B. L. Nguyen, D. T. Ngo, M. N. Dao, V. N. Q. Bao, and H. L. Vu, "Scheduling and power control for connectivity enhancement in multi-hop I2V/V2V networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 10322–10332, Aug. 2022.

[26] G. Luo et al., "Cooperative vehicular content distribution in edge computing assisted 5G-VANET," *China Commun.*, vol. 15, no. 7, pp. 1–17, Jul. 2018.

[27] F. D. Rango, P. Raimondo, and D. Amendola, "Extending SUMO and PLEXE simulator modules to consider energy consumption in platooning management in VANET," in *Proc. IEEE/ACM 23rd Int. Symp. Distrib. Simulation Real Time Appl. (DS-RT)*, Oct. 2019, pp. 1–9.

[28] H. Zhang, G. Luo, Y. Li, and F.-Y. Wang, "Parallel vision for intelligent transportation systems in metaverse: Challenges, solutions, and potential applications," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 53, no. 6, pp. 3400–3413, Jun. 2023.

[29] A. Caillot, S. Ouerghi, P. Vasseur, R. Boutteau, and Y. Dupuis, "Survey on cooperative perception in an automotive context," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 14204–14223, Sep. 2022.

[30] Q. Yang, S. Fu, H. Wang, and H. Fang, "Machine-learning-enabled cooperative perception for connected autonomous vehicles: Challenges and opportunities," *IEEE Netw.*, vol. 35, no. 3, pp. 96–101, May 2021.

[31] G. Volk, A. von Bernuth, and O. Bringmann, "Environment-aware development of robust vision-based cooperative perception systems," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 126–133.

[32] H. Liu, P. Ren, S. Jain, M. Murad, M. Gruteser, and F. Bai, "FusionEye: Perception sharing for connected vehicles and its bandwidth-accuracy trade-offs," in *Proc. 16th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2019, pp. 1–9.

[33] J. Guo et al., "CoFF: Cooperative spatial feature fusion for 3-D object detection on autonomous vehicles," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11078–11087, Jul. 2021.

[34] Y.-C. Liu, J. Tian, N. Glaser, and Z. Kira, "When2com: Multi-agent perception via communication graph grouping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4105–4114.

[35] Y.-C. Liu, J. Tian, C.-Y. Ma, N. Glaser, C.-W. Kuo, and Z. Kira, "who2com: Collaborative perception via learnable handshake communication," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 6876–6883.

[36] T. H. Wang et al., "V2VNet: Vehicle-to-vehicle communication for joint perception and prediction," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16* Springer, 2020, pp. 605–621. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-58536-5_36

[37] R. Xu, H. Xiang, Z. Tu, X. Xia, M.-H. Yang, and J. Ma, "V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer," 2022, *arXiv:2203.10638*.

[38] R. Molina-Masegosa, M. Sepulcre, J. Gozalvez, F. Berens, and V. Martinez, "Empirical models for the realistic generation of cooperative awareness messages in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5713–5717, May 2020.

[39] G. Luo, H. Zhang, H. He, J. Li, and F.-Y. Wang, "Multiagent adversarial collaborative learning via mean-field theory," *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 4994–5007, Oct. 2021.

[40] X. Zhang et al., "EMP: Edge-assisted multi-vehicle perception," in *Proc. 27th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2021, pp. 545–558.

[41] R. Yu, D. Yang, and H. Zhang, "Edge-assisted collaborative perception in autonomous driving: A reflection on communication design," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Dec. 2021, pp. 371–375.

[42] Y. Li, S. Ren, P. Wu, S. Chen, C. Feng, and W. Zhang, "Learning distilled collaboration graph for multi-agent perception," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 29541–29552.

[43] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast encoders for object detection from point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12689–12697.

[44] Z. Bao, S. Hossain, H. Lang, and X. Lin, "High-definition map generation technologies for autonomous driving," 2022, *arXiv:2206.05400*.

[45] X. Ren and Y. Yang, "A commentary of hyper-accurate position in MIT technology review 2021," *Fundam. Res.*, vol. 1, no. 6, pp. 840–841, Nov. 2021.

[46] G. Twardokus and H. Rahbari, "Towards protecting 5G sidelink scheduling in C-V2X against intelligent DoS attacks," *IEEE Trans. Wireless Commun.*, early access, Mar. 3, 2023, doi: 10.1109/TWC.2023.3249665.

[47] Z. Naghsh and S. Valaee, "Conflict-free scheduling in cellular V2X communications," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 106–119, Feb. 2021.

[48] F. Abbas, X. Yuan, M. S. Bute, and P. Fan, "Performance analysis using full duplex discovery mechanism in 5G-V2X communication networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11453–11464, Aug. 2022.

[49] G. Luo et al., "Software-defined cooperative data sharing in edge computing assisted 5G-VANET," *IEEE Trans. Mobile Comput.*, vol. 20, no. 3, pp. 1212–1229, Mar. 2021.

[50] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. Conf. Robot Learn.*, 2017, pp. 1–16.

[51] G. Carneiro, "Ns-3: Network simulator 3," in *Proc. UTM Lab Meeting April*, vol. 20, 2010, pp. 4–5.

[52] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "OPV2V: An open benchmark dataset and fusion pipeline for perception with Vehicle-to-Vehicle communication," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 2583–2589.

[53] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499.

[54] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3D point clouds," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 514–524.

[55] P. Kolman and O. Pangrác, "On the complexity of paths avoiding forbidden pairs," *Discrete Appl. Math.*, vol. 157, no. 13, pp. 2871–2876, Jul. 2009.

[56] D. S. Altner, Ö. Ergun, and N. A. Uhan, "The maximum flow network interdiction problem: Valid inequalities, integrality gaps, and approximability," *Oper. Res. Lett.*, vol. 38, no. 1, pp. 33–38, Jan. 2010.

Guiyang Luo was a Post-Doctoral Fellow with the Computer Science Department, Beijing University of Posts and Telecommunications (BUPT), from 2020 to 2022. He is currently an Assistant Professor with the Computer Science Department, BUPT. His current research interests include multi-agent systems and machine-type communications.

Chongzhang Shao is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications. His research interests include intelligent transportation systems, multi-vehicle cooperative perception, and swarm intelligence.

Nan Cheng (Senior Member, IEEE) received the B.E. and M.S. degrees from the Department of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009 and 2012, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, in 2016. He was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Toronto, from 2017 to 2019. He is currently a Professor with the State Key Laboratory of ISN and with the School of Telecommunications Engineering, Xidian University, Shaanxi, China. He has published over 90 journal articles in IEEE TRANSACTIONS and other top journals. His current research interests include B5G/6G, AI-driven future networks, and space-air-ground integrated networks. He serves as an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, and *Peer-to-Peer Networking and Applications*. He serves/served as the guest editor for several journals.

Haibo Zhou (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Shanghai Jiao Tong University, Shanghai, China, in 2014. From 2014 to 2017, he was a Post-Doctoral Fellow with the Broadband Communications Research Group, Department of Electrical and Computer Engineering, University of Waterloo. He is currently a Full Professor with the School of Electronic Science and Engineering, Nanjing University, Nanjing, China. His research interests include resource management and protocol design in B5G/6G networks, vehicular ad hoc networks, and space-air-ground integrated networks. He was a recipient of the 2019 IEEE ComSoc Asia Pacific Outstanding Young Researcher Award, the 2023–2024 IEEE ComSoc Distinguished Lecturer, and the 2023–2025 IEEE VTS Distinguished Lecturer. He served as the Track/Symposium Co-Chair for IEEE/CIC ICCC 2019, IEEE VTC-Fall 2020, IEEE VTC-Fall 2021, WCSP 2022, IEEE GLOBECOM 2022, and IEEE ICC 2024. He is an Associate Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE INTERNET OF THINGS JOURNAL, *IEEE Network*, and *Journal of Communications and Information Networks*.

Hui Zhang (Member, IEEE) received the B.S. degree in automation from Beijing Jiaotong University, Beijing, China, in 2015, and the Ph.D. degree in control theory and control engineering from the University of Chinese Academy of Sciences (UCAS), Beijing, in 2020. From August 2018 to October 2019, she was supported by UCAS as a joint-supervision Ph.D. Student with The University of Rhode Island, Kingston, RI, USA. She is currently a Lecturer with the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests include computer vision, pattern recognition, and intelligent transportation systems.

Quan Yuan (Member, IEEE) received the Ph.D. degree in computer science and technology from the Beijing University of Posts and Telecommunications (BUPT), China, in 2018. He is currently a Post-Doctoral Fellow with the State Key Laboratory of Networking and Switching Technology, BUPT. His current research interests include crowdsensing, connected vehicle, mobile internet, and intelligent transportation systems.

Jinglin Li received the Ph.D. degree in computer science and technology from the Beijing University of Posts and Telecommunications in 2004. He is currently a Professor of computer science and technology and the Director of the Switching and Intelligent Control Research Center (SICRC), State Key Laboratory of Networking and Switching Technology, China. His research interests include mobile internet, the Internet of Things, the Internet of Vehicles, convergence networks, and service technologies.