Federated HD Map Updating Through Overlapping Coalition Formation Game

Siping Shi, *Graduate Student Member, IEEE*, Chuang Hu, Dan Wang, *Senior Member, IEEE*, Yifei Zhu, *Member, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*—High Definition (HD) maps have become core supporting components for autonomous driving. To date, their updates heavily depend on the vehicle fleets of the map vendors, which cannot scale and timely reflect the highly dynamic environment. To ensure the HD map quality, it is advocated social vehicles should be used. Nevertheless, there are privacy concerns and a lack of incentives for social vehicles to contribute data. In this paper, we leverage federated analytics (FA), a newly developed collaborative data analytics paradigm, where raw data are kept local and only the insights generated from local analytics are sent to a server for aggregation. We present a new Federated Analytics based HD map Updating model (FAUMap) to protect the privacy of social vehicles. To motivate social vehicles to contribute data and improve the HD map quality, we formulate an overlapping coalition formation game, OCFUMap, and develop an algorithm to find feasible coalitions. Simulations show that our approach can improve the quality of the updated HD map by 1.56 times. To study an end-to-end operation of the FAUMap model and OCFUMap game, we present a case of HD map updates of the Powell street in San Francisco using the autonomous driving simulator CarLA.**

***Index Terms*—Federated analytics, high definition map updating, overlapping coalition game.**

I. Introduction

HIGH definition (HD) maps contain static and dynamic information of the driving environment and are believed to be a core component for future autonomous driving [1], [2]. To reflect the highly dynamic environment, frequent updates of the HD maps are essential for accurate localization and reliable navigation. To date, HD map updates depend on the vehicle fleets of the map vendors [3], [4]. This is not scalable.

To solve this problem, it is advocated that vehicle based crowdsourcing could be leveraged [5], [6]. participating vehicles nowadays are commonly equipped with sensors, such as cameras, Lidar, Radar, and many others [7], [8], which can contribute data the same as the vehicle fleets of the map vendors. HD map vendors are also willing to share profits; for example, DeepMap has given 5,000 USD per kilometer for HD map updating [1], and Japan's Dynamic Map Platform Co. (DMP) Ushr provides 162 million USD for HD map data acquisition [9].

With monetary sharing from the map vendors and the capability of the participating vehicles, there are still challenges preventing the true deployment of crowdsourced HD map updating in practice.

First, participating vehicles have privacy concerns. For example, the video data obtained by the cameras of participating vehicles may well contain privacy sensitive information, i.e., the plate number of front cars. Thus, how to ensure participating vehicles contribute data without raising privacy concerns?

Second, contributing data has a cost. For example, capturing and transmitting data will consume the computing and communication resources of the respective vehicles. participating vehicles are self-centric. Thus, how to motivate a large number of participating vehicles with appropriate rewards so that they are willing to contribute data for HD map updates?

To address the first challenge, we leverage federated analytics (FA), a new computing paradigm proposed by Google [10] for its Gboard application. Gboard is a keyboard app for smartphones providing AI features such as next word prediction, keyboard search suggestion, etc. With privacy concerns, Gboard trains the deep learning models through federated learning. To improve the deep learning models, these models should be evaluated periodically, which again needs to use the user data. Gboard leverages federated analytics to measure the overall quality of the trained machine learning models.

We propose a new *Federated Analytics based HD map Updating model (FAUMap)* to protect the privacy of participating vehicles. In FAUMap, we first define the quality of the HD map, a quantification on which the analytics for HD map updates can be based. We define the HD map quality through two metrics *information freshness* and *spatial completeness*, where the information freshness is characterized by Age of Information (AoI) [11] and the spatial completeness is characterized by

Manuscript received 14 April 2022; revised 8 November 2022; accepted 19 January 2023. Date of publication 31 January 2023; date of current version 8 January 2024. The work of Dan Wang was supported in part by the National Key R&D Program of China under Grant 2020YFE0200500, in part by the General Research Fund (GRF) under Grants 15210119, 15209220, 15200321, and 15201322, in part by the Innovation and Technology Commission (ITC) via project "Smart Railway Technology and Applications" (No. K-BBY1), and in part by the Collaborative Research Fund (CRF) under Grant C5018-20G. The work of Zhu Han was supported in part by the National Science Foundation (NSF) under Grants CNS-2107216, CNS-2128368, and CMMI-2222810, and in part by Toyota and Amazon. Recommended for acceptance by S. K. Das.
(Corresponding author: Chuang Hu.)

Siping Shi and Dan Wang are with the Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: cssshi@comp.polyu.edu.hk; csdwang@comp.polyu.edu.hk).

Chuang Hu is with the Department of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: hchuchuang@gmail.com).

Yifei Zhu is with the University of Michigan-Shanghai Jiao Tong University Joint Institute and with Cooperative Medianet Innovation Center (CMIC), Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: yifei.zhu@sjtu.edu.cn).

Zhu Han is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: hanzhu22@gmail.com).

Digital Object Identifier 10.1109/TMC.2023.3241090

Wasserstein Distance [12] between spatial distributions. We then formally model the local analytics procedure for FA.

To address the second challenge, we adopt a game theoretical approach. We note that an HD map update requires different types of data, e.g., traffic conditions, road construction, etc. Each type of data can be collected from a coalition of participating vehicles and each participating vehicle can join in multiple coalitions simultaneously since it has different types of data. Moreover, Vehicles in a coalition can share their communication resource to gain more benefits. To reflect such a scenario, we formulate an *overlapping coalition formation game (OCFUMap)* under our FAUMap model. We then develop an OCF algorithm that can effectively find feasible coalitions in this game that can maximize the utility of each vehicle and the HD map quality. We analyze our algorithm and prove that convergence is theoretically guaranteed.

We evaluate our OCF algorithm through simulations. We compare the OCF algorithm with the status quo methods. The results show that our algorithm can improve the quality of the HD map up to 1.56 times, and improve the utility of each vehicle up to 1.80 times. We further study an end-to-end operation of the FAUMap model and OCFUMap game with a case of Powell street in San Francisco using the autonomous driving simulator CarLA. Results show that the quality of the updated HD map and the vehicle utility can be improved by 1.33 times and 1.67 times, respectively.

To the best of our knowledge, this is the first work to study privacy-preserving HD map updating with incentive designs on crowdsourced participating vehicles. Technical contributions of this paper are summarized as follows:

- We model the quality of HD Map and develop an HD map updating model FAUMap based on federated analytics to address the privacy concerns of participating vehicles.
- We present a game theoretical approach for incentive design of the participating vehicles and we formulate an overlapping coalition formation game OCFUMap. We develop an algorithm to maximize HD map update quality and the utility of each vehicle.
- We evaluate our model and algorithm through extensive simulations and results demonstrate the superior performance of our proposed algorithm. We further present a case study to show how the proposed model and algorithm operate in a practical scenario.

The rest of this paper is organized as follows. Section II discusses the background and related work. Section III models the privacy-preserving HD Map updating system. Section IV formulates the OCFUMap game and develops the OCF algorithm to solve the game. The evaluation results of OCF algorithm based on extensive experiments are presented in Section V. And a case study of the proposed system is demonstrated in Section VI to show its effectiveness in practice. Finally, a conclusion is drawn in Section VII.

II. Background and Related Work

A. Background on HD Map

An HD map is a planar map with layers of information within a Geographic Information System (GIS). The most typical HD map is the four-layer model [13], see Fig. 1(a): 1) static information layer, which forms the basis of the map (i.e., the location of traffic lights, lanes, crosswalks), 2) semi-static information layer, which is used to support the driving of vehicles (i.e., road speed information and traffic rules, wide-area weather conditions), 3) semi-dynamic information layer, which reports the near real-time driving environment (i.e., accidents, congestion, traffic regulations, road works, and local weather), and 4) dynamic information layer, which reflects the real-time surrounding objects when driving, like nearby vehicles, pedestrians, and traffic signals.

HD maps, in particular, the dynamic layers, need to be frequently updated. HD map updates have three steps: (1) data acquisition, data are collected at regular intervals by vehicles which, nowadays, are equipped with various sensors, such as Lidar, radar, camera, etc; (2) raw data transmission: the collected data is sent to the HD map server; and (3) data analytics and HD map updating: for example, image data can be used by an object detection model to detect lanes. The HD map is updated with changes or new information. To date, map vendors, e.g., HERE, manage the map servers and dispatch vehicle fleet equipped with expensive professional sensors to collect data. Nevertheless, this approach is impossible to scale, i.e., to keep the pace of map updates and reflect real-time situations. As such, HD map updates is projected to leverage participating vehicles, which are also heavily equipped with various sensors nowadays, through a crowdsourced approach [14], [15], [16].

B. Federated Analytics

The objective of federated analytics is to collectively carry out data analytics tasks without disclosing the data belonging to each contributing party. FA differs from federated learning (FL) in the sense that FL emphasizes collaborative model training, in particular, a deep learning model. FA specifically refers to the model inference phase of machine learning, and thus is the sequel of federated learning. For example, a federated video analytics architecture [17] assumes the machine learning model has been established and the participating peer devices collaboratively perform model inference. In FA, data are kept local and only insights generated from local analytics will be sent to the server for result aggregation. It is essential to ensure that the underlying data of peer devices remains private and secure, unlike the traditional collaborative analytics approaches required to collect local data to a central server [18].

In an abstract view, an FA operation framework for HD map updating is shown in Fig. 1(b). There are three steps: 1) data acquisition, each vehicle collects environment data with the equipped sensors while driving; 2) data analytics, the sensor data is analyzed locally in the vehicle and the analytic results are sent to the map server for aggregation; 3) HD map updating, the map server aggregates all received analytic results and updates the HD map. We will detail the FA operations into a formal privacy preserving HD map updating model in Section III. Before this, we discuss the position of our research in the research literature.

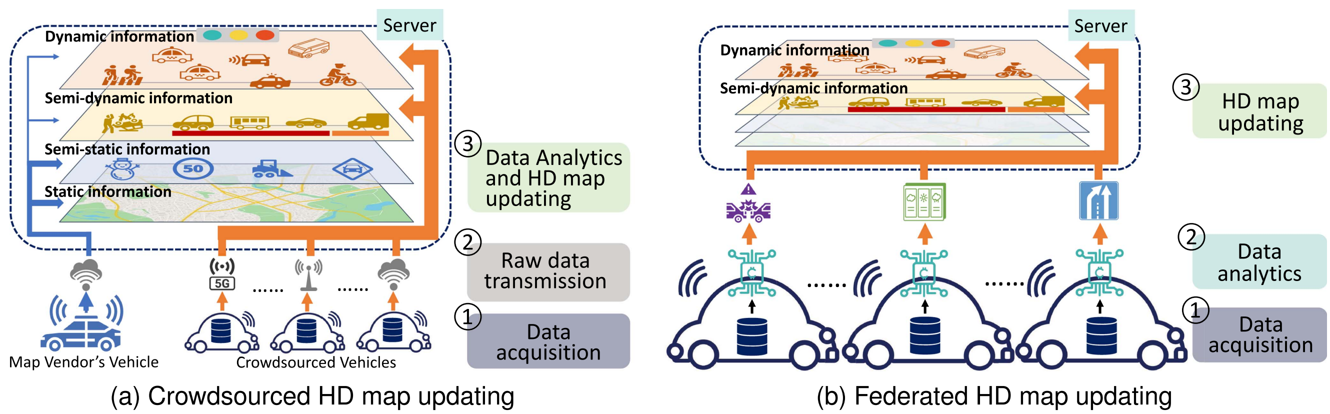


Fig. 1. Overview of the HD map updating model. (a) is the model of crowdsourced HD map updating, and (b) is our proposed model with data analyzed locally.

C. Related Work

In the research literature, our work falls into the *vehicular crowdsourcing* area which applied in *HD map updating* scenario, with *incentive mechanism* design.

Vehicular Crowdsourcing. Vehicular crowdsourcing is a process that a group of vehicles is involved to perform data collection tasks. It has been studied widely for various industrial applications, including environment monitoring [19], [20], [21], traffic management [22], [23], [24], HD map updating [25], [26], etc. In most of works in literature, a vehicular crowdsourcing system consists of two components: a crowdsourcer and a set of vehicles. Each vehicle collects the requested data with their own sensing devices and then sends it to the crowdsourcer. Once received the collected data, the crowdsourcer saves and organizes them for further analytics. For example, vehicles in iLOCuS [19] collect temperature and PM 2.5 data with corresponding sensors and send to the crowdsourcer for analyzing the city air quality. Most of work not consider the privacy problem of each vehicle and all the collected raw data is directly sent to the crowdsourcer. However, most data collected for HD map updating is sensitive (i.e., pedestrian, plate number of front vehicles) and the extra privacy information of vehicle data can be extracted from the collected data. It is not suitable to upload raw data to the crowdsourcer. Some cryptographic technologies like homomorphic encryption [27] is used for protecting the privacy of vehicles in [24], [28]. However, cryptographic methods are not efficient and with high computational costs in practice [29], which can not fulfil the real-time requirements of HD map updating.

HD Map Updating. The architecture of HD map updating in the literature can be mainly divided into two types: centralized and crowdsourced. In the centralized architecture [30], [31], [32], with all the data has been collected by the fleets of map vendors, researchers focus on improving the accuracy and efficiency in HD map updating, such as improving the accuracy of lane detection as in [31]. These can be utilized to update the static and semi-static information layer of the HD map but cannot be applied to update the semi-dynamic and dynamic information layer of the HD map. In the crowdsourced architecture, the HD map is updated with crowdsourced data of participating vehicles, which is an instance of vehicular crowdsourcing [33], [34]. Specifically, LiveMap proposed in [33] use the raw sensor data collected by crowdsourced vehicles to keep the HD map up to update, and FCDMap proposed in [34] also adopted the crowdsourced architecture to update the HD Map. Since all the work in the crowdsourced architecture assume the map server is trusted, the privacy problem of each vehicle is not considered. IFMap proposed in [35] can achieve privacy preserving as it aims to update the local HD map in the vehicle and no data will be shared to the server for global HD map updating. However, this is not suitable in our scenario that a global HD map should be updated. Therefore, in crowdsourced HD map updating, none of the work in the literature has researched the privacy problem. To solve aforementioned issues, we leverage the emerged federated analytics paradigm to protect the privacy of each vehicle in HD map updating and propose FAUMap.

Incentive Mechanism for Vehicular Crowdsourcing. Since generating information in vehicles consumes computation and communication resources, either monetary or nonmonetary incentives [25], [36] are required to motivate vehicles to provide more high quality contributions. There have been many works [37], [38], [39], [40], [41], [42], [43], [44] focusing on incentive mechanism design for vehicular crowdsourcing. For example, auction-based incentive mechanisms are proposed in [38], [39], [40], [41] where vehicles can sell their collected data to the crowdsourcer with their claimed bid prices. And Stackelberg game based incentive mechanism are proposed in [42], [43], [44], in which the bargaining among the participants and the crowdsourcer were modeled. Typically, most of them are designed for single crowdsourcing task, and all vehicles collect one same kind of data at one time. However, in HD map updating, the crowdsourcer is required to aggregate different kind of sensing data to update the HD map and multiple crowdsourcing tasks are distributed to the vehicles. Therefore, the existing incentive mechanisms are not suitable for the HD map updating scenario. Though incentive-aware vehicle recruitment (IVRecruit) proposed in [45] considered multi crowdsourcing tasks, the collaboration between vehicles which can improve vehicle utility is also not modeled in existing incentive mechanism. In our paper, with FAUMap, we design an overlapping coalition formation game based algorithm to find

TABLE I
COMPARISON BETWEEN FAUMAP AND STATUS QUO VEHICULAR CROWDSOURING HD MAP UPDATING METHODS

	Privacy preserving	Crowdsourcing task	Vehicle collaboration
LiveMap [34], FCDMap [35]	No	Single	No
IFMap [36]	Yes	None	No
IVRecruit [46]	No	multiple	No
FAUMap	**Yes**	**Multiple**	**Yes**

feasible coalitions to maximize the utility of each vehicle, and it enables to take both multiple crowdsourcing tasks and vehicles collaboration into consideration.

We summary the comparison between the FAUMap and the status quo vehicular crowdsourcing HD Map updating methods in Table I. To the best of our knowledge, no existing method is designed to achieve privacy preserving HD map updating with an effective incentive mechanism, we are the first to fill this gap.

III. Privacy-Preserving HD Map Updating: A Federated Analytics Model

We develop a model on federated analytics based privacy-preserving HD map updating system. The considered system consists of a map server which contains an HD map and a set of vehicles $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$. The HD map includes a set of information denoted as $\mathcal{M} = \{m_1, m_2, \ldots, m_M\}$, and its quality is determined by the cumulative quality of each information m_i. In each round of federated HD map updating, each vehicle $v_j \in \mathcal{V}$ first generates a set of information locally through analytic functions with the local sensor data, and then send the generated information to the map server, finally, the map server aggregates all the uploaded information and updates the HD map. To incentive the vehicles to contribute more information, the map server provides payoff to each contributed vehicle and the total budget is $\mathcal{B}$. Each vehicle can generate multiple information, and it can determine whether to generate information m_i. Let x_j^i denote the indicator, i.e., $x_j^i = 1$ if vehicle v_j generates information m_i; otherwise, $x_i^j = 0$. We define $\mathbf{x_j} \triangleq (x_j^1, x_j^2, \ldots, x_j^M)$ and the generation strategy matrix $\mathbf{X} \triangleq (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N})$. $\mathbf{X}$ is a decision variable to be optimized. The map server and the vehicle have their own utility function and aim to maximize their utility.

In this section, we first introduce temporal and spatial characteristics of the HD map and present the HD map quality model. Then we model the federated analytic assisted HD map updating. Finally, we analyze the cost of updating and model the utility of the map server and vehicles.

A. HD Map Quality Modeling

A high quality HD Map guarantees the safety and quality of autonomous driving. Intrinsically, an HD map is a set of information, such as road speed information, accidents information, and weather conditions information. Information in HD Map is required to keep updated from the temporal view and keep completeness from the spatial view to assist autonomous driving [34], [46], [47]. To capture these characteristics, we define the temporal and spatial quality of each information with the *information freshness* metric and *spatial completeness* metric, respectively.

For the temporal quality, our goal is to measure the freshness of the dynamic information, which matches the requirement of HD Map where the information should be updated periodically based on the latency requirement of its layer [48] (i.e., in the dynamic and semi-dynamic layer, information should be updated at intervals of less than several seconds and several minutes, respectively.). To this end, we select age of information(AoI) [11], a well developed metric to quantify latency and characterize freshness of collected data in status updating systems and applications. In AoI, information freshness can be quantified through a non-increasing utility function, and the choose of the utility function is based on the information source and the application under consideration [11], [49]. For latency sensitive information in HD Map, the utility is non-negative and will decrease with AoI grows over time. To capture this characteristic, we utilize the logarithmic function to describe the utility function as in [50].

Definition 1. (Information freshness) For the generated information, given the Age of information Δ, the freshness F of the information is defined as

$$F = \log\left(1 + \frac{A}{\Delta}\right), \tag{1}$$

where A is a constant scaling factor to describe sensitivity of information in different HD map layers regarding to delay.

For the spatial quality, we aim to maintain the completeness of the updated region, which matches the spatial requirement of HD Map where the information needs complete coverage of the target area [47]. The data used for information generation is mostly not collected from a determined discrete point, but from a continuous trajectory which can be represented by a distribution [19]. We assume the target and the current spatial distribution of the generated information is z and σ, respectively. Thus, the spatial completeness of the generated information can be described by the distance between z and σ. For the choice of distance metric, KL divergence used in [19] cannot satisfy the symmetric characteristic of HD Map where the distance from target distribution to the current distribution should be same to the distance from current distribution to the target distribution. Even though the Jensen-Shannon divergence is symmetrical, it performs well for joint distributions and can not provide a meaningful value for disjoint distributions [51]. Therefore, Wasserstein Distance [12], [52] is adopted to measure this distance since it is a well-developed symmetric distance metric to measure the similarity between probability distributions on a given metric space and performs well for general probability distributions. We define $Wasserstein(z, \sigma)$ as the Wasserstein distance between z and σ. The smaller of the distance indicates the similar of the target and the current distribution, and thus the more complete of the information. We consider the spatial completeness is inversely proportional to the distance.

Definition 2. (Spatial completeness) For information generated based on the dataset with a spatial distribution σ, given the spatial distribution z required for generating the information, the spatial completeness S of the generated information is defined as

$$S = K \cdot \frac{1}{Wasserstein(z, \sigma)}, \tag{2}$$

where K is a constant scaling factor to describe the sensitivity of different information in HD Map regarding to spatial coverage.

With the temporal and spatial quality of the information are defined with information freshness and spatial completeness, respectively, the quality of the information in HD Map can be represented by the weighted sum of them. We assume the freshness and spatial completeness of information m_i contributed by vehicle v_j are F_j^i and S_j^i, respectively. Thus, the quality q_j^i of information m_i contributed by vehicle v_j is computed as:

$$q_j^i = \alpha^i \cdot F_j^i + \beta^i \cdot S_j^i, \tag{3}$$

where α^i and β^i are weighting parameters indicating the importance of information freshness and spatial completeness to the quality of the information, and their units are unit quality per freshness and unit quality per completeness, respectively. We assume the overall quality of information m_i increases proportionally with the total contribution of the participated vehicles until the total contribution reaches a threshold ρ^i. This assumption makes sense and also adopted in [53]. Suppose the required information in HD Map is the weather conditions of a certain area. When the number of vehicles contributed this information exceeds 100, those additional vehicles make little contribution to the quality of the required information. Thus, given the generation strategy $\mathbf{X}$, the overall quality of information m_i can be computed as

$$Q_i(\mathbf{X}) = \begin{cases} \frac{\zeta^i}{\rho^i} \sum_{\mathbf{x}_j \in \mathbf{X}} x_j^i q_j^i, & \sum_{\mathbf{x}_j \in \mathbf{X}} x_j^i q_j^i \le \rho^i, \\ \zeta^i, & \text{otherwise}, \end{cases} \tag{4}$$

where $\sum_{\mathbf{x}_j \in \mathbf{X}} x_j^i q_j^i$ is the total contribution of vehicles made to information m_i, and ζ^i is the upper bound quality of information m_i regarding the threshold ρ^i.

Since the HD map is constructed through a set of information $\mathcal{M}$, the quality of the HD map is the cumulative quality contribution by each information. Thus, given the generation strategy $\mathbf{X}$, the quality of the HD map is

$$Q_{\mathcal{M}}(\mathbf{X}) = \sum_{m_i \in \mathcal{M}} Q_i(\mathbf{X}). \tag{5}$$

B. FA-Assisted HD Map Updating Modeling

Vehicles equipped with onboard sensors can be used to not only collect data of the surrounding environment but also generate and transmit information with the collected data, which made FA paradigm available for HD Map updating. In each update, the vehicle first obtains data with the equipped sensors. Then it performs local analytics with the collected data to generate information in HD Map, and sends the generated information to the server for updating. Suppose the data set of vehicle v_j is D_j, and the consisted data can be used to generated various information which used to update the HD map, and d_j^i represents the size of data required to generate information m_i.

In order to protect the privacy of each vehicle, generated information instead of collected data is sent to the server for map updating. As the complexity of generating different information is not the same, the computation rate of each information varies. Let u_j^i be the computation rate of vehicle v_j for generating information m_i, then the computation latency can be derived as

$$t_{j,i}^{co} = \frac{d_j^i}{u_j^i}. \tag{6}$$

After local analytics, the generated information will be sent to the map server for updating. We assume the size of generated information is $\widehat{d_j^i}$, and the transmission rate of vehicle v_j is r_j. Then, the transmission latency can be computed as

$$t_{j,i}^{com} = \frac{\widehat{d_j^i}}{r_j}. \tag{7}$$

Particularly, for the calculation of AoI, an updated information with timestamp $u^i(t)$ is said to have age $t - u^i(t)$ at time t when $t \ge u^i(t)$. Here, timestamp $u^i(t)$ denotes the generation time of information and time t denotes the reception timestamp of updating in the map server. Therefore, when at the current update time t, the difference $t - u^i(t)$ describes the transmission latency of updating information m_i, which is the sum of computation latency and communication latency. Let Δ_j^i be the age of information, then we have

$$\Delta_j^i = t_{j,i}^{co} + t_{j,i}^{com}. \tag{8}$$

Each vehicle has its own moving trajectory, thus the spatial distribution of collected data is different from each other. Let z^i be the target spatial trajectory distribution for generating information m_i, and σ_j^i be the spatial trajectory distribution of vehicle v_j for generating information m_i. With Wasserstein Distance metric to measure the distance between σ_j^i and z^i, the spatial distance w_j^i between z^i and σ_j^i can be expressed as:

$$w_j^i = Wasserstein(z^i, \sigma_j^i) = \inf_{\gamma \sim \Pi\left(z^i, \sigma_j^i\right)} \mathbb{E}_{(x,y)\sim\gamma}[\|x - y\|]. \tag{9}$$

With the AoI Δ_j^i and the spatial distance w_j^i, we can get the quality of information m_i generated by vehicle v_j based on (3) as

$$q_j^i = \alpha^i \cdot \log\left(1 + \frac{A^i}{\Delta_j^i}\right) + \beta^i \cdot \frac{K^i}{w_j^i}. \tag{10}$$

C. Utility Modeling

In our proposed federated HD map updating model, both the server and the vehicles can obtain benefits and have their own utility model. For the map server, the utility is determined by the value of the HD map and the cost taken for updating the HD map. For each vehicle, it can get the payoff from generating the updated information at the cost of consuming its own computation and communication resources. We introduce the two kinds of utility models in detail in the following.

Map server utility model: For the server, the utility equals the difference between the map value and the map cost. The map cost comes from the payoff to vehicles for contributing information. We consider the payoff of each updated information is proportional to its quality. As mentioned before, the overall quality of information m_i is represented by $Q_i(\mathbf{X})$, and the cost paid for updating information m_i is defined as :

$$G_i(\mathbf{X}) = \gamma_1 \cdot Q_i(\mathbf{X}), \tag{11}$$

where γ_1 is the incentive factor. As the value of the HD map is related to the quality of the HD map, it increases with the quality improved. We define $U_{\mathcal{M}}$ as the utility of map $\mathcal{M}$, thus the utility of the map server is

$$U_{\mathcal{M}}(\mathbf{X}) = Q_{\mathcal{M}}(\mathbf{X}) - \sum_{m_i \in \mathcal{M}} G_i(\mathbf{X}). \tag{12}$$

Vehicle utility model: For each vehicle, the utility equals the difference between the obtained payoff and the resource cost. Let c_j^i be the sum of communication and computation cost for vehicle v_j to generate information m_i. Since the computation and communication resources of each vehicle are limited, we define C_j^{max} as the total resources of vehicle v_j. Then we have

$$\sum_{i=0}^{M} x_j^i \cdot c_j^i \leq C_j^{max}. \tag{13}$$

Let p_j^i be the payoff for vehicle v_j to generate information m_i. We can compute the utility R_j of vehicle v_j as

$$R_j(\mathbf{x_j}) = \sum_{i=0}^{M} x_j^i \cdot \left(p_j^i - c_j^i\right). \tag{14}$$

IV. An Overlapping Coalition Formation Game for Federated HD Map Updating

In this section, we first formulate an overlapping coalition formation (OCF) game to capture the behaviors of participating vehicles and the map server under our FAUMap model. Then we design an OCF algorithm to maximize the utility of each vehicle and the HD map quality. Finally, theoretical analysis is conduct to show a stable overlapping coalition structure (OCS) will be reached in finite iterations.

A. Overlapping Coalition Formation Game Formulation

In coalition games, there is a set of *players* who seek to form cooperative *coalitions* to strengthen their positions in a given game scenario [54]. Each coalition has a *transferable utility* which can be divided by the player in the coalition. Each player can maximize its utility by performing operations which satisfying the *rules*, then the *strategy* of each player is individually decided. Specifically, the coalitions in OCF can be overlapping, that is a player can participate in multiple coalitions simultaneously.

Players and Coalitions. In our federated HD map updating scenario, any vehicle $v_j \in \mathcal{V}$ is seen as a player and the vehicle set $\mathcal{V}$ is the set of players. Each information m_i is corresponding to a coalition. Let $\mathbf{S}^i$ be the coalition of information m_i, in which the coalition members make contributions to collaboratively update information m_i. Variable x_j^i defined as the indicator to represent whether v_j generate information m_i, can indicate whether vehicle v_j belongs to $\mathbf{S}^i$, thus we have $\mathbf{S}^i = \{v_j \in \mathcal{V} | x_j^i = 1\}$. Members in a coalition can share messages with each other through the vehicle-to-vehicle (V2V) communication method. Thus, they can *collaboratively* make decisions to increase both the utility of coalition and vehicles. For example, when the quality of the coalition-related map information has not reached the upper bound, the utilities can be increased by inviting more vehicles to join the coalition to generate the related map information. The shared messages enable vehicles to allocate their communication and computation resources appropriately and thus earn more payoffs by contributing to multiple coalitions. In contrast, vehicles without shared messages can only *independently* choose the most profitable map information to contribute with fewer payoffs. In other words, when the local data, computation and communication resources are available, a vehicle v_j can join multiple coalitions simultaneously. Therefore, we define an overlapping coalition structure (OCS) as the set of all coalitions, denoted by $\Theta = \{\mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^M\}$.

Transferable Utility. The utility of coalition $\mathbf{S}^i$ is given by a characteristic function $U : [0, 1]^N \rightarrow \mathbb{R}_+$. It is based on the value of information contributed by the members of the coalition which has been explained in (10). Thus, according to the quality model defined in (4) and cost model defined in (11), the utility function of coalition $\mathbf{S}^i$ can be written as

$$U(\mathbf{S}^i) = \begin{cases} \frac{B^i}{\rho^i} \sum_{v_j \in \mathcal{V}} q_j^i \cdot x_j^i, & \sum_{v_j \in \mathcal{V}} q_j^i \cdot x_j^i \leq \rho^i, \\ B^i, & \text{otherwise}, \end{cases} \tag{15}$$

where $B^i = \gamma_i \zeta_i$ is the budget allocated to coalition $\mathbf{S}^i$.

The utility of coalition $\mathbf{S}^i$ is also the payoff that needs to be divided among the vehicles who contribute to coalition $\mathbf{S}^i$, i.e., the members of $\mathbf{S}^i$. Let $\mathbf{p}^i \in \mathbb{R}^N$ be the payoff vector for the members in coalition $\mathbf{S}^i$, and it satisfies $\sum_{v_j \in \mathcal{V}} p_j^i = U(\mathbf{S}^i)$. We assume the payoff of any member in a coalition is based on their contribution level to the coalition, and it is unaffected by those outsiders. Let $\phi_j(\mathbf{S}^i)$ be the rewards for vehicle v_j received from coalition $\mathbf{S}^i$, we have

$$p_j^i = \phi_j(\mathbf{S}^i) = \begin{cases} U(\mathbf{S}^i) \frac{q_j^i}{\sum_{v_l \in \mathbf{S}^i} q_l^i}, & v_j \in \mathbf{S}^i, \\ 0, & v_j \notin \mathbf{S}^i. \end{cases} \tag{16}$$

Rules. A vehicle v_j may obtain negative utility from joining a coalition as the payoff received from a coalition is less than the cost of generating the related information. Therefore, the vehicle has strong motivation to consider transferring from one coalition to another or joining a new coalition to increase its utility or only quit the current coalition if there exits no coalition to increase its utility. The fundamental operations of a vehicle can be classified into transferring operation, joining operation, and quitting operation. Accordingly, we define *transferring rule*, *joining rule* and *quitting rule*.

Let $T_j(\mathbf{S}^p, \mathbf{S}^q)$ denote the operation of vehicle v_j transferring from coalition $\mathbf{S}^p \in \Theta$ to coalition $\mathbf{S}^q \in \Theta$, which is to withdraw all the resources of vehicle v_j from coalition $\mathbf{S}^p$ and invest the required resources into coalition $\mathbf{S}^q$. Transferring operation $T_j(\mathbf{S}^p, \mathbf{S}^q)$ is considered only when the following three conditions can be met: 1) the required resources that vehicle v_j invested to coalition $\mathbf{S}^q$ does not exceed its communication and computation capacity, i.e., (13); 2) the transfer operation is profitable. Specifically, the utility vehicle v_j obtains from joining coalition $\mathbf{S}^q$ should be positive and larger than that from joining coalition $\mathbf{S}^p$. That is

$$\phi_j(\mathbf{S}^q) - c_j^q > \max\{0, \phi_j(\mathbf{S}^p) - c_j^p\}. \tag{17}$$

and 3) after transferring, the utility of any other vehicle which is in the coalition $\mathbf{S}^q$ should not be decreased. That is

$$\phi_k(\mathbf{S}^q \cup \{v_j\}) \geq \phi_k(\mathbf{S}^q), \quad \forall k \in \{\mathbf{S}^q | k \neq j\}. \tag{18}$$

The third condition captures the characteristic of the coalitions that other members in coalition $\mathbf{S}^q$ has the right to decide whether to accept the transfer operation due to their own utility. Specifically, if the utility of other vehicles in coalition $\mathbf{S}^q$ is decreased when vehicle v_j joins this coalition, then vehicle v_j will not be allowed to join in coalition $\mathbf{S}^q$. Thus, we can define the transferring rule as follows:

Rule 1. (Transferring Rule) In the proposed OCF game with the current OCS Θ, the transfer operation $T_j(\mathbf{S}^p, \mathbf{S}^q)$ is feasible and can be executed if it satisfies (13), (17), and (18).

Let $J_j(\mathbf{S}^p)$ denote the operation of vehicle v_j joining a new coalition $\mathbf{S}^p \in \Theta$, which is to invest the required resources into coalition $\mathbf{S}^p$. For its own sake, user v_j only considers joining a new coalition $\mathbf{S}^p$ when it has enough resources to invest, i.e., (13). Besides, it can get positive utility from coalition $\mathbf{S}^p$, and the utility of other members in coalition $\mathbf{S}^p$ should be considered as well. Thus, we have

$$\begin{cases} \phi_j(\mathbf{S}^p \cup \{v_j\}) - c_j^p > 0, \\ \phi_k(\mathbf{S}^p \cup \{v_j\}) \geq \phi_k(\mathbf{S}^p), \quad \forall k \in \{\mathbf{S}^p | k \neq j\}. \end{cases} \tag{19}$$

Thus, we can define the joining rule as follows:

Rule 2. (Joining Rule) In the proposed OCF game with the current OCS Θ, the joining operation $J_j(\mathbf{S}^p)$ is feasible and can be executed if it satisfies (13) and (19).

Let $Q_j(\mathbf{S}^p)$ denote the operation of vehicle v_j quitting from coalition $\mathbf{S}^p \in \Theta$, which is to withdraw all the invested resources from coalition $\mathbf{S}^p$. This operation occurs when the vehicle v_j cannot obtain positive utility either from this coalition or any potential transfer operations. Thus we can define the quitting rule as follows:

Rule 3. (Quitting Rule) In the proposed OCF game with the current OCS Θ, the quitting operation $Q_j(\mathbf{S}^p)$ is feasible and can be executed if it satisfies

$$\begin{cases} \phi_j(\mathbf{S}^p) - c_j^p \leq 0, \\ T_j(\mathbf{S}^p, \mathbf{S}^q) \text{is not profitable}, \forall \mathbf{S}^q \in \Theta. \end{cases} \tag{20}$$

With all the concepts mentioned above, we have

The Overlapping Coalition Formation Game for HD map Updating (OCFUMap):

- *Players:* Each vehicle $v_j \in \mathcal{V}$ is seen as a player. The players of the OCF game is the vehicle set $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$.
- *Coalitions:* Each information m_i corresponds to a coalition $\mathbf{S}^i$. Since the size of the information set is M, there is M coalitions in the OCF game.
- *Transferable Utility:* The utility of coalition $\mathbf{S}^i$ is defined in (15), and can be divided among players in the coalition.
- *Rules:* Transferring rule, joining rule, and quitting rule.
- *Strategy:* Each vehicle decides which coalition to join. The strategy of vehicle v_j is denoted as $\mathbf{x}_j = (x_j^1, x_j^2, \ldots, x_j^M)$, where $x_j^i = 1$ indicates that vehicle v_j choose to contribute to coalition $\mathbf{S}^i$.

B. Overlapping Coalition Formation Algorithm

We design an OCF based algorithm to solve the OCFUMap problem. The overall OCF algorithm is shown in Algorithm 1, which consists of two phases: vehicle strategy initialization phase and overlapping coalition formation phase.

In the vehicle strategy initialization phase, each vehicle chooses to join the coalition with a greedy strategy. In particular, each vehicle initially chooses the most profitable coalition to join. As illustrated in Line 3 - Line 6 of Algorithm 1, all element in strategy matrix are first assigned with 0, and then each vehicle calculates the potential profit when joining each coalition, and finally choose the coalition with the maximum profit to participate in (set the strategy value of the corresponding coalition to 1). With all the vehicles finished choosing, the initial overlapping coalition structure comes out.

In the overlapping coalition formation phase, each vehicle takes different operations to maximize its utility. The procedure may take several iterations until no vehicle having the motivation to move, i.e., no operation can increase the payoff of any vehicle. As we have mentioned, there are three fundamental operations for each vehicle. In each iteration, we first check if each vehicle has the motivation to quit, i.e., obtain non-positive payoff and have no feasible coalition to join. As shown in Line 12 - Line 18, for each coalition the vehicle participated in, if the utility is non-positive and there exists no feasible coalition to transfer, then the vehicle will quit the current coalition. Otherwise, if there exists a set of coalitions to make transfer operation of the vehicle, then choose the most profitable coalition to execute the transfer operation. Once no vehicle wants to quit, we will check whether they have the motivation to transfer or join a new coalition, and the corresponding code is from Line 18 to Line 26. From Line 19 to Line 23, the transfer operation will be performed if it satisfies the Rule 1. Then the vehicle will check if any other new coalition is feasible to join based on the Rule 2. The iterations continue until no vehicle has the motivation to move. With all the vehicles finished all the checking steps in one iteration, we compare the current strategy matrix with the before one, and stop the iterations until the strategy matrix is unchanged (Line 9).

Remark. We have introduced that the resources of each vehicle are limited, and the budget for each coalition is upper bounded. Thus, it is hard for each vehicle to join all coalitions or all vehicles join to one coalition, and the *grand coalition* that includes all vehicles seldom forms.

C. Analysis of the OCF Algorithm

Stability and Convergence. In traditional coalition formation (CF) games, the *core* notion is used to describe stability, which is the set of feasible allocations that cannot be improved upon by a coalition of the players. However, different from the traditional CF games, the OCF game is still an ongoing topic and there exist no general stability concepts. It is more complicated to

Algorithm 1: The OCF Algorithm.

Input: Vehicle set: $\mathcal{V}$; HD Map information set: $\mathcal{M}$.
Output: Vehicle strategy matrix: $\mathbf{X} \in \{0,1\}_{N\times M}$;

1 **Vehicle side**:
2 // Step I: Vehicle strategy initialization
3 $\mathbf{X} \leftarrow \mathbf{0}_{N\times M}$,
4 **for** $v_j \in \mathcal{V}$ **do**
5     $m_i \leftarrow \max_{m_i \in \mathcal{M}}(p_j^i - c_j^i)$,
6     $x_j^i = 1$.
7 // Step II: Overlapping coalition formation
8 $\mathbf{X}' \leftarrow \mathbf{0}_{N\times M}$,
9 **while** $\mathbf{X}' \neq \mathbf{X}$ **do**
10     $\mathbf{X}' \leftarrow \mathbf{X}$,
11   **for** $v_j \in \mathcal{V}$ **do**
12   **for** $\mathbf{S}^p \in \left\{\mathbf{S}^i | x_j^i = 1\right\}$ **do**
13   **if** $\phi_j(\mathbf{S}^p) - c_j^p \leq 0$ **then**
14   **if** *exists a feasible transfer coalitions set* Ω *for* v_j **then**
15                     $\mathbf{S}^q \leftarrow \max_{\mathbf{S}^q \in \Omega}(\phi_j(\mathbf{S}^q) - c_j^q)$,
16                     $T_j(\mathbf{S}^p, \mathbf{S}^q)$;
17   **else**
18                     $Q_j(\mathbf{S}^p)$;
19   **for** $v_j \in \mathcal{V}$ **do**
20   **for** $\mathbf{S}^k \in \left\{\mathbf{S}^i | x_j^i = 1\right\}$ **do**
21   **if** *exists a feasible transfer coalitions set* Ω *for* v_j **then**
22                 $\mathbf{S}^l \leftarrow \max_{\mathbf{S}^l \in \Omega}(\phi_j(\mathbf{S}^l) - c_j^l)$,
23                 $T_j(\mathbf{S}^k, \mathbf{S}^l)$;
24   **for** $m_i \in \mathcal{M}$ **do**
25   **if** $\mathbf{S}^i$ *is a new feasible coalition for* j *to join in* **then**
26                 $J_j(\mathbf{S}^i)$;
27 **return** $\mathbf{X}$.

define notions for stability in OCF games due to the overlapping property. In OCF games, instead of forming a single coalition, the players may form multiple coalitions that overlap one another, thus complicating the computation of the maximal total payoff of such an overlapping coalition structure. Therefore, we incorporate the concept of stability in the traditional cooperative games and extend it in our proposed OCF-algorithm by defining the concept of T-stable OCS.

Definition 3. (T-stable OCS) In the proposed OCF game with the OCS $\Theta = \{\mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^M\}$, for any vehicle $v_j \in \mathcal{V}$, if it cannot make any feasible transfer operations, or quit any coalitions, or join any new coalitions in its current strategy, then we say the OCS Θ here is T-stable.

A T-stable OCS in our proposed OCF game corresponds to an equilibrium state in which no vehicle has the incentive to shift its resource from the currently formed coalitions. Therefore all the vehicles keep to their current strategies and no change happens in the T-stable OCS.

Theorem 1. Starting from the initial OCS, the OCF algorithm converges to a T-stable OCS in finite iterations.

Proof. Each vehicle performs the OCF algorithm iteratively to maximize their utility. Assume the OCS changes from Θ_{k-1} to Θ_k. For each vehicle v_j after k iterations, its generation strategy changes from $\mathbf{x}_j(k-1)$ to $\mathbf{x}_j(k)$ that satisfies $R_j(k-1) < R_j(k)$. For any other vehicle $v_l \in \mathcal{V}$ and $v_l \neq v_j$, the utility satisfies $R_l(k-1) \leq R_l(k)$ according to (17), (18), and (19). Therefore, the utility of any vehicle will not decrease from Θ_{k-1} to Θ_k. Since the utility of each vehicle is bounded by their own limited resources, the final OCS is guaranteed to reach within finite iterations. Thus, no vehicle can make any feasible operations to increase their utility. According to Definition 3, the final OCS is T-stable. □

Complexity. The computational complexity is determined by the number of the iterations and the number of attempts of transfer operations in the coalition formation phase. Note that the total number of iterations cannot be given in the closed-form since we do not know at which moment the vehicles form a T-stable OCS, which is common in the design of most heuristic algorithms. For the number of attempts of transfer operations, we assume the number of coalitions that a vehicle can simultaneously join is $\lceil M/2 \rceil$ at most due to the limited resources equipped in the vehicle. In each iteration of the coalition phase, there are two stages that a vehicle may attempt to transfer. The first stage is to check whether the vehicle has motivation to quit a coalition (from Line 11 - Line 18 in Algorithm 1). Since the vehicle will check feasible transfer coalition from the set of coalitions not joined currently, the vehicle tries at most $\lceil M/2 \rceil (M - \lceil M/2 \rceil)$ attempts of transfer operations in the first stage. The second stage is to check whether the vehicle has the motivation to transfer or join a new coalition (from Line 19 - Line 26 in Algorithm 1). Similarly, the vehicle tries at most $\lceil M/2 \rceil (M - \lceil M/2 \rceil)$ attempts of transfer operations in the second stage. To sum them up, we can obtain that each vehicle tries at most $2\lceil M/2 \rceil (M - \lceil M/2 \rceil)$ attempts of transfer operations in a iteration. The total number of vehicles is N, thus, the total number of transfer attempts in the worst case is $2N\lceil M/2 \rceil (M - \lceil M/2 \rceil)$ in a iteration. Note that the number of attempts is significantly lower than the worst-case as we initialize the strategy of each vehicles with greedy strategy and the time required in an iteration is relatively smaller.

V. Simulation

In this section, we evaluate the overlapping coalition game assisted federated HD map updating scheme with simulated data. We demonstrate the better performance of our method by comparing it with several baseline approaches.

A. Setup

Settings. We use the KITTI360 [55] dataset to generate the HD map, which contains 320 k images and 100 k laser scans in a driving distance of 73.7 km. The data is randomly divided and assigned to 100 vehicles. The total budget for HD map updating is set to 8000 USD. For simplicity, the constant factors (A and K) for the information freshness and spatial completeness of each map information are all set to 1. Each information m_i in different layers of the HD map has different requirements for information freshness and spatial completeness, which leads to a different impact on the quality of the HD map. For example, information in the dynamic layer of an HD map has a stringent

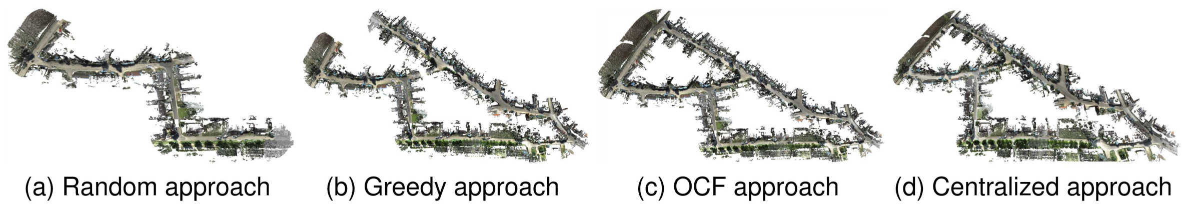


Fig. 2. Visual comparison of HD maps updated by different approaches.

requirement of information freshness, and information in the static layer of an HD map has a stringent requirement of spatial completeness. To this end, for the information in the static layer, semi-static layer, semi-dynamic layer, and dynamic layer, we set the value of α^i to be 1, 1.5, 2.5, and 3, respectively, and the value of β^i is set to be 3, 2.5, 1.5, and 1, respectively. For each map information i, the target distribution z^i follows a certain Gaussian distribution $\mathcal{N}(\mu, \theta^2)$, where μ is randomly selected from a range of $0 \sim 100$ and θ^2 is randomly selected from a range of $0.2 \sim 2.0$. Similarly, the trajectory of each vehicle also follows a Gaussian distribution, but the variance is ranged from $0.5 \sim 1.0$. For each vehicle v_j, the upper bound of computation and communication resource C_j^{max} is uniformly distributed in a range of $500 \sim 1500$. The data size $\hat{d}_j^i$ of the generated information m_i by different vehicles is set to be the same and uniformly distributed in a range of $1 \sim 30$. For vehicle v_j, the required computation rate u_j^i and transmission rate r_j for generate information m_i are randomly sampled in a range of $5 \sim 15$ and $1 \sim 20$, respectively. Since generating different map information consumes a different amount of resources, the value of c_j^i is randomly selected in a range of $5 \sim 16$. The incentive factor γ_1 is set to 0.2 initially.

Benchmark. In order to demonstrate the effectiveness of our FAUMap model, we compare it with the existing crowdsourced approach with data collection as in [33]. To evaluate the performance of our OCF algorithm, we compare it with three baselines: the centralized approach, the random approach, and the greedy approach. The detail of each baseline is introduced as follows:

- *Crowdsourced approach:* Each vehicle sends the collected data to the server directly, and the server performs data analytics with the uploaded data and updates the HD Map with the analytics result.
- *Centralized approach:* The map server has full control of all vehicles. It has all information about the system. The strategy of each vehicle is decided by the map server to maximize the quality of the HD map. It serves as an upper bound of the HD map quality.
- *Random approach:* The map server cannot control any vehicles in the system. The strategy of each vehicle is determined randomly with the computation and communication resource constraints.
- *Greedy approach:* The strategy of each vehicle is independently determined to maximize their own utility within the constrained resources and without considering the choices of other vehicles.

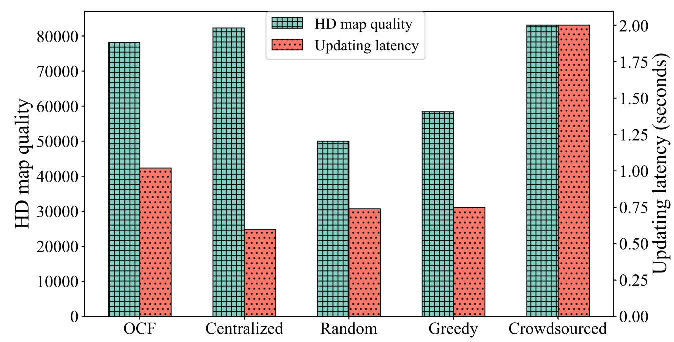


Fig. 3. Comparison on the HD map quality and updating latency.

B. Results

We evaluate our method from two aspects, the first is performance analysis including HD map quality analysis, efficiency improvement vehicle utility analysis, and convergence analysis, and the second aspect is micro analysis including analyzing the impact of incentive intensity and vehicles scale, respectively.

HD Map Quality Improvement. The HD map generated by each approach is shown in Fig. 2. For the map integrity aspect, we can observe that our proposed method outperforms either the random approach or greedy approach, and can reach almost the same integrity as the centralized approach which represents the upper bound. The comparison of numerical results of HD map quality is shown in Fig. 3. Compared to the crowdsourced approach which directly collect sensor data for updating, our proposed method based on FA can achieve similar HD map quality. Besides, the privacy of each vehicle is better considered and protected in our method than the crowdsourced approach. For the performance of the proposed OCF algorithm, though the quality gap of OCF is 5.3% lower than the centralized approach which represents the upper bound of HD map quality, we can see that our proposed OCF approach outperforms the random approach and the greedy approach with an improvement of 56.3% and 33.8% respectively. It reveals that our method can achieve a relatively high quality HD map.

Efficiency Improvement. The HD Map is required to be updated in real time to guarantee the safety and quality of autonomous driving. We evaluate the end-to-end latency of one HD Map updating process with different approaches, and the results are demonstrated in Fig. 3. Among all the evaluated approaches, the crowdsorced approach has the highest latency as it required to upload large amount of sensor data to the server for aggregation.

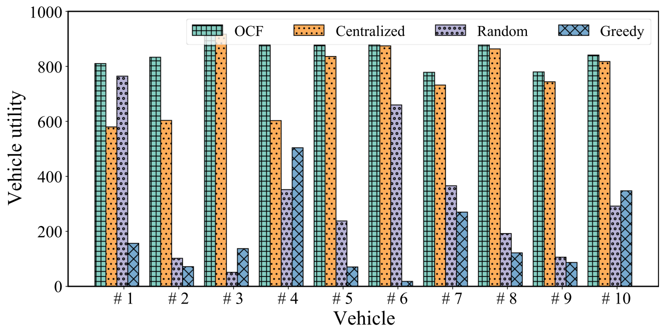


Fig. 4. Comparison on the vehicle utility.

For the proposed OCF approach, even though the updating latency is nearly 36.0% higher than the random and greedy approaches duo to the time cost of coalition formation process, it can finish one updating in about 1 seconed and achieve the real time requirement of HD Map updating. Moreover, compared to the crowdsourced approach, the efficiency of OCF approach is improved more than 96.0%. These results confirms the better efficiency of our proposed approach.

Vehicle Utility Improvement. Fig. 4 shows the utility of 10 vehicles selected randomly from the participated vehicles under different approaches. For each vehicle, the OCF approach presents to reach the highest vehicle utility and is up to 55.2% higher than the centralized approach. The main reason is that the OCF aims to maximize the utility of each vehicle while the centralized approach only considers the quality of the HD map. Moreover, the OCF method also significantly outperforms the greedy approach by at least 80%. The underlying reason is that vehicles in a coalition can collaboratively make decisions through the V2V communication method. Thus each vehicle can appropriately allocate its computation and communication resources and enabling it to make contributions to multiple tasks simultaneously and obtain more payoff. In comparison, each vehicle in the greedy approach makes decisions independently and may even lead to less payoff than the resource cost due to the utility of each information being upper bounded. It reflects that the vehicle can obtain many benefits through collaboration and joining coalitions. The performance of the random approach is also worse than our method as it just determines the generation strategy randomly. These validate that the OCF approach can largely improve the utility of each vehicle.

Convergence Analysis. Our OCF method is proved to reach a stable overlapping coalition structure after finite iterations. The convergence result of OCF is shown in Fig. 5. We can observe the cumulative distribution function (CDF) of the total number of iterations, versus the number of iterations, with a different number of vehicles and a different number of updated HD map information, respectively. It is obvious that OCF can converge within 10 iterations in both scenarios. Moreover, the speed of convergence becomes faster as the number of vehicles increases, and it becomes slower when the number of updated HD map information increases. We see that the computational complexity of our proposed algorithm is rather smaller. For example, when the number of vehicles is 20, the OCF approach requires only 9 iterations to reach convergence.

Impact of Incentive Intensity. The incentive intensity factor γ_1 in (11) determines how much payoff offers to the vehicles for contributing information. We show how the incentive intensity impacts the HD map quality and the vehicle utility in Fig. 6. The base incentive intensity is set to 0.1, and 6X represents 6 times of the base incentive intensity. As shown in Fig. 6, the HD map quality increases greatly when the incentive intensity increased from 1 time to 3 times of the base incentive intensity. However, when the incentive intensity keeps increasing, the increasing speed of the HD map quality becomes slow and turns to 0 when the incentive intensity reaching 4 times. Similarly, when the intensive intensity increasing to 4 times, the vehicle utility turns to saturate as shown in Fig. 6(b). This is because that the information a vehicle can generate is constrained to its computation and communication resources, and the utility stops increasing even the incentive intensity still increasing. This illustrates that a proper incentive intensity should be carefully chosen to obtain a double win.

Impact of Vehicles Scale. We also investigate the impact of the number of vehicles on the HD map quality and the vehicle utility. The experimental results are shown in Fig. 7. We can see that both the HD map quality and the vehicle utility increase steadily at first, and stop increasing when the number of vehicles exceeds 100. This is because that the value of a coalition reached the maximum due to the limited budget, and new vehicles cannot obtain a payoff from the coalition and choose to contribute nothing. Therefore, the budget of the map server should be increased properly when the scale of the vehicle increased to obtain a higher quality HD map.

VI. A Case Study

We develop a case to study the end-to-end operation of our model and algorithm in a real-world environment. We leverage CarLA [56], an open-source autonomous driving simulation platform developed by Intel, which can simulate and integrate three core components: HD map, vehicles, and map server, into holistic cases. We use CarLA to develop a case in San Francisco. First, we construct a base HD map of Powell street of San Francisco using the information from OpenStreetMap, see Fig. 8(a). Second, we simulate vehicles driving through the simulated environments. Each vehicle is equipped with stereo cameras and LiDAR sensors for collecting environmental data, and the driving trajectories are shown in Fig. 8(b) (the red lines). Vehicles can share messages with each other through V2V communication. Third, we simulate the map server to distribute the requirement of updated map information (i.e., the target distribution of the map information) and aggregate updates to construct a global HD map. Since we need to update the lane of Powell street, we set the target distribution to be a Gaussian distribution over the spatial domain, where the probability mass function achieves the maximum at Powell street.

System Integration. We integrate FAUMap into this case by revising the vehicle behaviors of the vehicles and map servers. Specifically, after a certain period of time, each vehicle performs

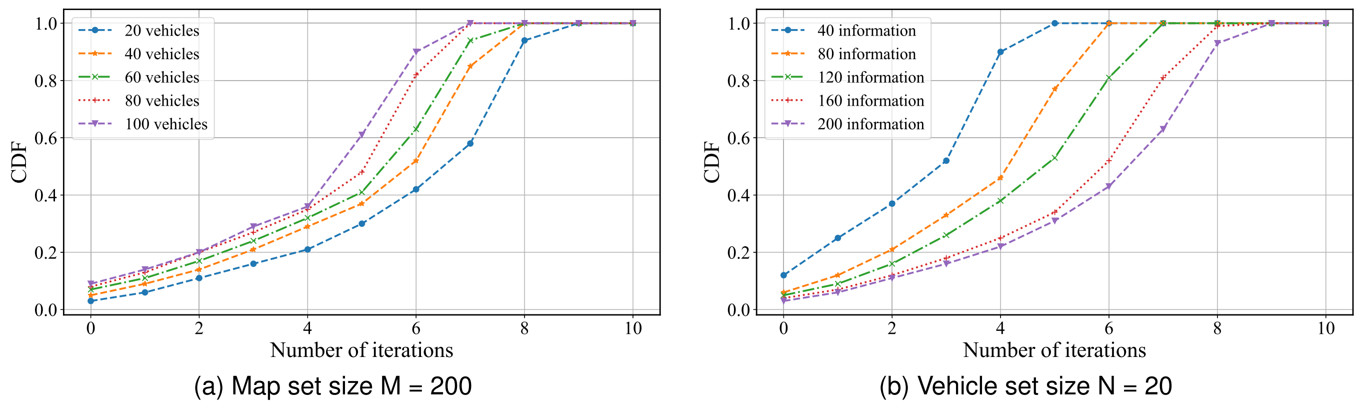


Fig. 5. The CDF of the total number of iterations in the OCF algorithm versus the number of iterations.

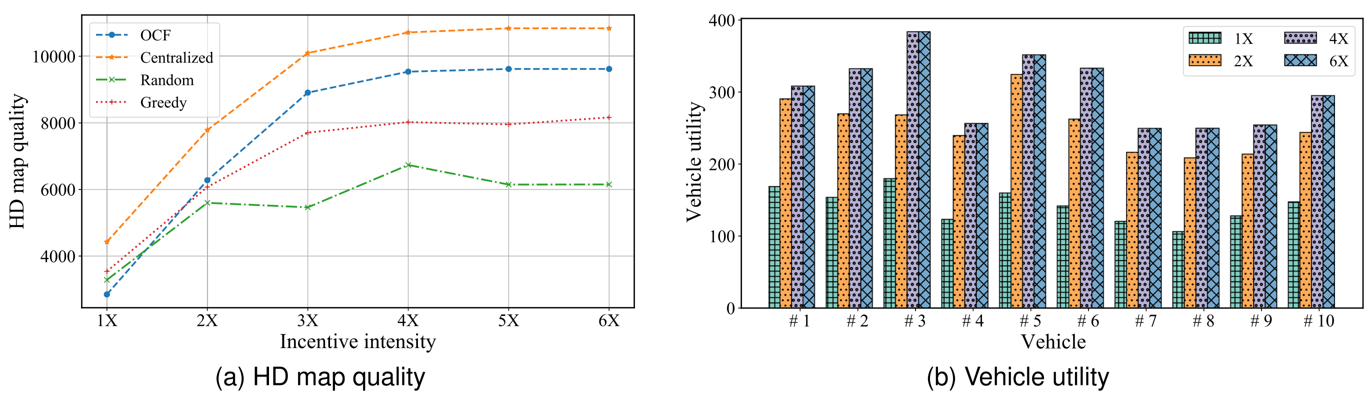


Fig. 6. The impact of incentive intensity on the HD map quality and the vehicle utility.

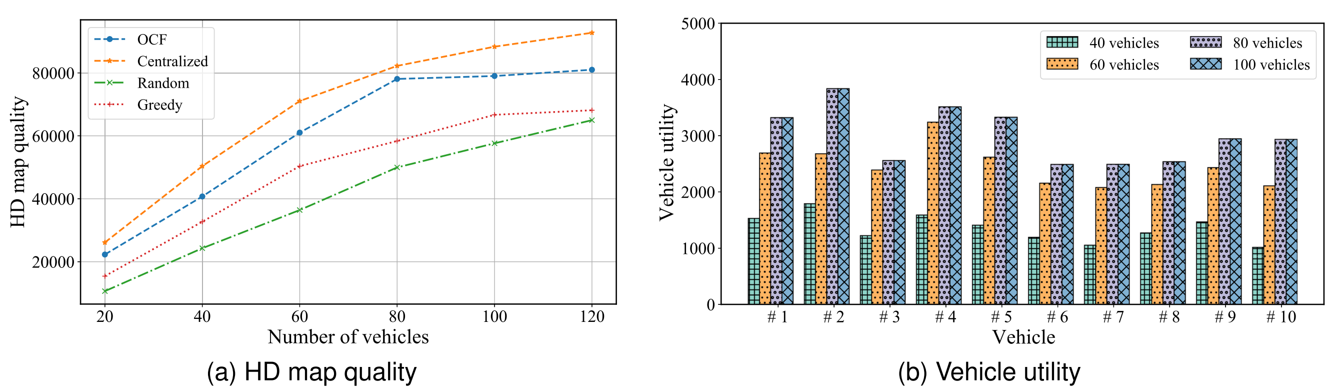
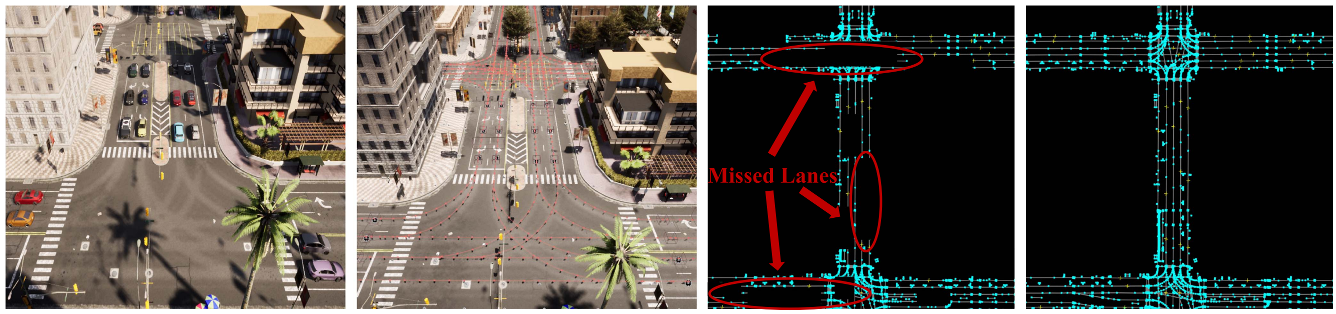


Fig. 7. The impact of vehicle scale on the HD map quality and the vehicle utility.

analytic functions with the local sensor data and generates a set of information related to the HD map. Then these analytic results will be sent to the map server to aggregate the global HD map. We implement the OCF algorithm and deploy it to the vehicles and map server to provide a strategy for them to maximize their individual utility. Specifically, the map server initializes the budget for updating the HD map at first, and then each vehicle decides which map information to generate through the algorithm. We also deploy a random algorithm to the vehicles and map server as a baseline, in which the strategy is randomly decided.

Performance Improvement. The updated HD map in our case study is shown in Fig. 8(c) and (d), respectively. It is obvious that the lane of Powell street is clearly marked in the HD map (the blue points and the grey lines) with our method, while some lanes are missing in the HD map generated by the baseline. For the aspect of vehicle utility, our method has an improvement of 1.67 times compared to the baseline. Our method also improves the HD map quality by 1.33 times.

On Field Operations. Fig. 9 shows the end-to-end operations in the field of FAUMap between 8:00am to 10:00am. The top two graphs show the size of updated map information set and the



(a) Simulated environment and vehicles (b) Trajectory of the simulated vehicles (c) Lane marking result generated by the original method (d) Lane marking result generated by the proposed FAUMap

Fig. 8. Simulation of the lane marking process with the proposed federated HD map updating scheme.

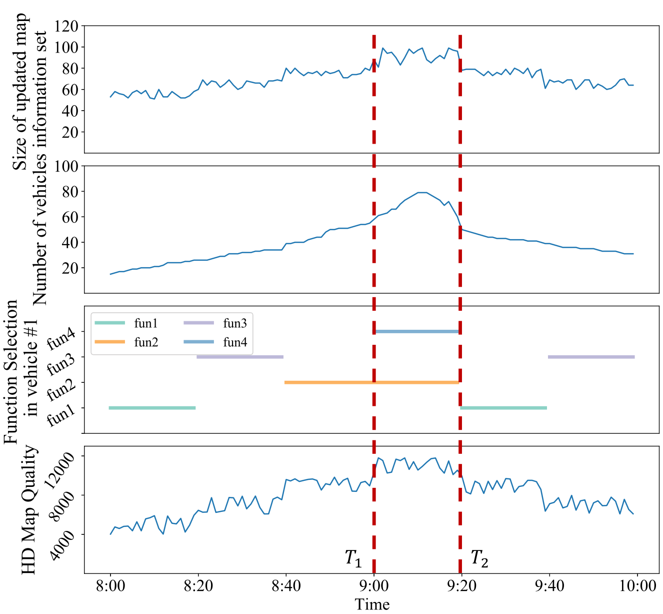


Fig. 9. The end-to-end operations in the field of FAUMap.

number of participated vehicles over a period of time, respectively. The bottom two graphs show the analytic functions of a vehicle and the HD map quality, respectively. We can see that OCF algorithm adjusts the vehicle function, i.e., the strategy, to achieve a high HD map quality in runtime successfully. For example, at time T_1 9:00am, since the number of vehicles increased, a vehicle analyzes the congestion information through function 4 and sends it to the server. The size of the updated map information set is increased and the quality of the HD map also improved with the timely updates. The number of vehicles decreased at time T_2 9:20am, and no congestion information can be generated through function 4. Thus the vehicle shift to perform function 1 to keep a high HD map quality. These validate that our proposed method is feasible and effective in practice.

VII. Conclusion

In this article, we study the privacy preserving and incentive mechanism problem in HD map updating. We first propose FAUMap, a federated analytics based HD map updating model, to protect the privacy of each vehicle. Then we formulate an overlapping coalition formation game, OCFUMap, to capture the behaviors of participating vehicles and the map server under our FAUMap model, and design an OCF algorithm to find feasible coalitions which can maximize the utility of each vehicle and the HD map quality. We evaluate the OCF algorithm with simulated data. Evaluation results show substantial improvements in HD Map quality and vehicle utility. We further develop a case study on HD map updates of a real street to examine the FAUMap end-to-end operations and performance in the field.

References

[1] M. Sarazen, "The golden age of HD mapping for autonomous driving," Aug. 11, 2018. [Online]. Available: https://medium.com/syncedreview/the-golden-age-of-hd-mapping-for-autonomous-driving-b2a2ec4c11d

[2] F. Lyu et al., "Characterizing urban vehicle-to-vehicle communications for reliable safety applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2586–2602, Jun. 2020.

[3] M. Panzarino, "Apple is rebuilding Maps from the ground up," Jun. 29, 2018. [Online]. Available: https://techcrunch.com/2018/06/29/apple-is-rebuilding-maps-from-the-ground-up/

[4] E. Klimczak, "Upgrading uber's 3D fleet," Mar. 05, 2019. [Online]. Available: https://medium.com/uber-design/upgrading-ubers-3d-fleet-4662c3e1081

[5] P. Bonetti, "HERE and Mobileye: Crowd-sourced HD mapping for autonomous cars," Dec. 30, 2016. [Online]. Available: https://360.here.com/2016/12/30/here-and-mobileye-crowd-sourced-hd-mapping-for-autonomous-cars/

[6] B. Li, D. Song, A. Kingery, D. Zheng, Y. Xu, and H. Guo, "Lane marking verification for high definition map maintenance using crowdsourced images," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, NV, 2020, pp. 2324–2329.

[7] E. Ye, P. Spiegel, and M. Althoff, "Cooperative raw sensor data fusion for ground truth generation in autonomous driving," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst.*, Rhodes, Greece, 2020, pp. 1–7.

[8] K. Banerjee, D. Notz, J. Windelen, S. Gavarraju, and M. He, "Online camera LiDAR fusion and object detection on hybrid data for autonomous driving," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Suzhou, China, 2018, pp. 1632–1638.

[9] S. Abuelsamid, "Super cruise HD map developer ushr acquired by dynamic map platform of Japan," Feb. 12, 2019. [Online]. Available: https://bit.ly/3fgEYul

[10] D. Ramage, "Federated analytics: Collaborative data science without data collection," May 27, 2020. [Online]. Available: https://ai.googleblog.com/2020/05/federated-analytics-collaborative-data.html

[11] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1183–1210, May 2021.

[12] L. Chizat, P. Roussillon, F. Léger, F. Vialard, and G. Peyré, "Faster Wasserstein distance estimation with the sinkhorn divergence," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, virtual, 2020, pp. 2257–2269.

[13] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough, and A. Mouzakitis, "A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 829–846, Apr. 2018.

[14] K. Kim, S. Cho, and W. Chung, "HD map update for autonomous driving with crowdsourced data," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 1895–1901, Apr. 2021.

[15] P. Zhang, M. Zhang, and J. Liu, "Real-time HD map change detection for crowdsourcing update based on mid-to-high-end sensors," *Sensors*, vol. 21, no. 7, Apr. 2021, Art. no. 2477.

[16] M. Liebner, D. Jain, J. Schauseil, D. Pannen, and A. Hackelöer, "Crowdsourced HD map patches based on road model inference and graph-based SLAM," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Paris, France, 2019, pp. 1211–1218.

[17] C. Hu, R. Lu, and D. Wang, "FEVA: A federated video analytics architecture for networked smart cameras," *IEEE Netw.*, vol. 35, no. 6, pp. 163–170, Nov./Dec. 2021.

[18] H. Jin, L. Jia, and Z. Zhou, "Boosting edge intelligence with collaborative cross-edge analytics," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2444–2458, Feb. 2021.

[19] S. Xu, X. Chen, X. Pi, C. Joe-Wong, P. Zhang, and H. Y. Noh, "iLOCuS: Incentivizing vehicle mobility to optimize sensing distribution in crowd sensing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 8, pp. 1831–1847, Aug. 2020.

[20] D. Hasenfratz et al., "Deriving high-resolution urban air pollution maps using mobile sensor nodes," *Pervasive Mobile Comput.*, vol. 16, pp. 268–285, Jan. 2015.

[21] J. Huang et al., "A crowdsource-based sensing system for monitoring fine-grained air quality in urban environments," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3240–3247, Apr. 2019.

[22] J. Wan, J. Liu, Z. Shao, A. V. Vasilakos, M. Imran, and K. Zhou, "Mobile crowd sensing for traffic prediction in Internet of Vehicles," *Sensors*, vol. 16, no. 1, Jan. 2016, Art. no. 88.

[23] S. Olariu, "Vehicular crowdsourcing for congestion support in smart cities," *Smart Cities*, vol. 4, no. 2, pp. 662–685, May 2021.

[24] C. Zhang, L. Zhu, C. Xu, X. Du, and M. Guizani, "A privacy-preserving traffic monitoring scheme via vehicular crowdsourcing," *Sensors*, vol. 19, no. 6, Mar. 2019, Art. no. 1274.

[25] X. Cao et al., "Trajectory penetration characterization for efficient vehicle selection in HD map crowdsourcing," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4526–4539, Mar. 2021.

[26] Y. Wang, X. Liu, H. Wei, G. Forman, C. Chen, and Y. Zhu, "CrowdAtlas: Self-updating maps for cloud and personal use," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, New York, NY, 2013, pp. 27–40.

[27] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 1–35, Sep. 2018.

[28] R. Tandon and P. K. Gupta, "A novel encryption scheme based on fully homomorphic encryption and RR-AES along with privacy preservation for vehicular networks," in *Proc. Int. Conf. Adv. Comput. Data Sci.*, M.V. SinghP. K. TyagiJ. GuptaT. FlusserÖren, and V. R. Sonawane et al., Eds., Nashik, India, 2021, pp. 351–360.

[29] L. J. Aslett, P. M. Esperança, and C. C. Holmes, "A review of homomorphic encryption and software tools for encrypted statistical machine learning," 2015, *arXiv:1508.06574*.

[30] D. Pannen, M. Liebner, and W. Burgard, "HD map change detection with a boosted particle filter," in *Proc. IEEE Int. Conf. Robot. Automat.*, Montreal, Canada, 2019, pp. 2561–2567.

[31] C. Yan, C. Zheng, C. Gao, W. Yu, Y. Cai, and C. Ma, "Lane information perception network for HD maps," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst.*, Rhodes, Greece, 2020, pp. 1–6.

[32] K. Massow et al., "Deriving HD maps for highly automated driving from vehicular probe data," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst.*, Rio de Janeiro, Brazil, 2016, pp. 1745–1752.

[33] Q. Liu, T. Han, J. L. Xie, and B. Kim, "LiveMap: Real-time dynamic map in automotive edge computing," in *Proc. IEEE Conf. Comput. Commun.*, virtual, Vancouver, Canada, 2021, pp. 1–10.

[34] D. Pannen, M. Liebner, W. Hempel, and W. Burgard, "How to keep HD maps for automated driving up to date," in *Proc. IEEE Int. Conf. Robot. Automat.*, Paris, France, 2020, pp. 2288–2294.

[35] Y. Liu, M. Song, and Y. Guo, "An incremental fusing method for high-definition map updating," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, Bari, Italy, 2019, pp. 4251–4256.

[36] X. Cao et al., "Online worker selection towards high quality map collection for autonomous driving," in *Proc. IEEE Glob. Commun. Conf.*, Waikoloa, HI, 2019, pp. 1–6.

[37] X. Chen, L. Zhang, B. Lin, and Y. Fang, "Delay-aware incentive mechanism for crowdsourcing with vehicles in smart cities," in *Proc. IEEE Glob. Commun. Conf.*, Waikoloa, HI, 2019, pp. 1–6.

[38] S. Hu, X. Feng, and H. Zheng, "An online incentive mechanism for time-sensitive vehicle-based crowdsourcing," in *Proc. IEEE 4th Int. Conf. Comput. Commun.*, Chengdu, China, 2018, pp. 522–527.

[39] X. Chen, L. Zhang, Y. Pang, B. Lin, and Y. Fang, "Timeliness-aware incentive mechanism for vehicular crowdsourcing in smart cities," *IEEE Trans. Mobile Comput.*, vol. 21, no. 9, pp. 3373–3387, Sep. 2022.

[40] G. Gao, M. Xiao, J. Wu, L. Huang, and C. Hu, "Truthful incentive mechanism for nondeterministic crowdsensing with vehicles," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2982–2997, Dec. 2018.

[41] A. Yassine, M. S. Hossain, G. Muhammad, and M. Guizani, "Cloudlet-based intelligent auctioning agents for truthful autonomous electric vehicles energy crowdsourcing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5457–5466, May 2020.

[42] W. Wu et al., "Incentive mechanism design to meet task criteria in crowdsourcing: How to determine your budget," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 2, pp. 502–516, Feb. 2017.

[43] M. Wu, D. Ye, S. Tang, and R. Yu, "Collaborative vehicle sensing in bus networks: A Stackelberg game approach," in *Proc. IEEE/CIC Int. Conf. Commun. China*, Chengdu, China, 2016, pp. 1–6.

[44] Y. Jiao, P. Wang, D. Niyato, B. Lin, and D. I. Kim, "Mechanism design for wireless powered spatial crowdsourcing networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 920–934, Jan. 2020.

[45] L. Liu, X. Wen, L. Wang, Z. Lu, W. Jing, and Y. Chen, "Incentive-aware recruitment of intelligent vehicles for edge-assisted mobile crowdsensing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12085–12097, Oct. 2020.

[46] M. O. Tas, H. S. Yavuz, and A. Yazici, "High-definition map update framework for intelligent autonomous transfer vehicles," *J. Exp. Theor. Artif. Intell.*, vol. 33, no. 5, pp. 847–865, Jul. 2021.

[47] F. Poggenhans et al., "Lanelet2: A high-definition map framework for the future of automated driving," in *Proc. 21st Int. Conf. Intell. Transp. Syst.*, Maui, HI, 2018, pp. 1672–1679.

[48] A. E. C. C. (AECC), "Operational behavior of a high definition map application white paper," 2020. [Online]. Available: https://aecc.org/wp-content/uploads/2020/07/Operational_Behavior_of_a_High_Definition_Map_Application.pdf

[49] Y. Sun and B. Cyr, "Sampling for data freshness optimization: Non-linear age functions," *J. Commun. Netw.*, vol. 21, no. 3, pp. 204–219, Jun. 2019.

[50] J. Ma, L. Liu, H. Song, and P. Fan, "On the fundamental tradeoffs between video freshness and video quality in real-time applications," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1492–1503, Feb. 2021.

[51] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, Sydney, Australia, 2017, pp. 214–223.

[52] T. Vayer, L. Chapel, R. Flamary, R. Tavenard, and N. Courty, "Fused Gromov-Wasserstein distance for structured objects," *Algorithms*, vol. 13, no. 9, Aug. 2020, Art. no. 212.

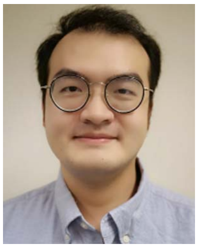
[53] B. Di, T. Wang, L. Song, and Z. Han, "Collaborative smartphone sensing using overlapping coalition formation games," *IEEE Trans. Mobile Comput.*, vol. 16, no. 1, pp. 30–43, Jan. 2017.

[54] T. Rahwan, T. P. Michalak, M. Wooldridge, and N. R. Jennings, "Coalition structure generation: A survey," *Artif. Intell.*, vol. 229, pp. 139–174, Dec. 2015.

[55] J. Xie, M. Kiefel, M.-T. Sun, and A. Geiger, "Semantic instance annotation of street scenes by 3D to 2D label transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, 2016, pp. 3688–3697.

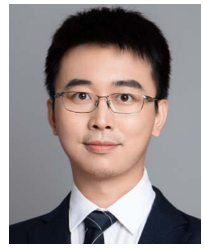
[56] A. Dosovitskiy, G. Ros, F. Codevilla, A. M. López, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. Conf. Robot Learn.*, Mountain View, CA, 2017, pp. 1–16.

Siping Shi (Graduate Student Member, IEEE) received the BS degree in computer science from Sichuan University, in 2014 and the MS degree in computer applied technology from the University of Chinese Academy of Sciences, in 2017. She is currently working toward the PhD degree with the Hong Kong Polytechnic University. Her research interests include edge computing, federated learning, and analytics.



Chuang Hu received the BS and MS degrees from Wuhan University, in 2013 and 2016 and the PhD degree from the Hong Kong Polytechnic University, in 2019. He is currently an associate researcher in the School of Computer Science with Wuhan University. His research interests include edge/federated learning, distributed/parallel computing, and Network.

Dan Wang (Senior Member, IEEE) received the BSc, MSc, PhD degrees in computer science from Peking University, Case Western Reserve University and Simon Fraser University. His research falls in general computer networking and systems, where he published in ACM SIGCOMM, ACM SIGMETRICS, IEEE INFOCOM, and many others. He is the steering committee chair of IEEE/ACM IWQoS. He served as the TPC co-Chair of IEEE/ACM IWQoS 2020. His recent research focus on smart energy systems. He won the Best Paper Awards of ACM e-Energy 2018 and ACM Buildsys 2018. He has served as a TPC co-Chair of the ACM e-Energy 2020 and he will serve as General co-Chair of the ACM e-Energy 2022. He is a steering committee member of ACM e-Energy. He serves as a founding area editor of ACM SIGEnergy Energy Informatics Review. His research has been adopted by industry, e.g., Henderson, Huawei, and IBM. He won the Global Innovation Award, TechConnect, in 2017.



Yifei Zhu (Member, IEEE) received the BE degree from Xi'an Jiaotong University, Xian, China, in 2012, and the MPhil degree from the Hong Kong University of Science and Technology, Hong Kong, China, in 2015, and the PhD degree in computer science from the Simon Fraser University, BC, Canada, in 2020. He is currently an assistant professor with Shanghai Jiao Tong University. His research interests include edge computing, multimedia networking, distributed machine learning system, and network economics.

Zhu Han (Fellow, IEEE) received the BS degree in electronic engineering from Tsinghua University, in 1997 and the MS and PhD degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively. From 2000 to 2002, he was an R&D Engineer of JDSU, Germantown, Maryland. From 2003 to 2006, he was a research associate with the University of Maryland. From 2006 to 2008, he was an assistant professor with Boise State University, Idaho. Currently, he is a John and Rebecca Moores professor in the Electrical and Computer Engineering Department as well as in the Computer Science Department with the University of Houston, Texas. He is main research targets on the novel game-theory related concepts critical to enabling efficient and distributive use of wireless networks with limited resources. His other research interests include wireless resource allocation and management, wireless communications and networking, quantum computing, data science, smart grid, security and privacy. He received an NSF Career Award, in 2010, the Fred W. Ellersick Prize of the *IEEE Communication Society,* in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing,* in 2015, IEEE Leonard G. Abraham Prize in the field of Communications Systems (best paper award in IEEE JSAC), in 2016, and several best paper awards in IEEE conferences. He was an IEEE Communications Society Distinguished Lecturer from 2015-2018, AAAS fellow since 2019, and ACM distinguished Member since 2019. He is a 1% highly cited Researcher since 2017 according to Web of Science. He is also the winner of the 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: "for contributions to game theory and distributed management of autonomous communication networks."