# MetaTrading: An Immersion-Aware Model Trading Framework for Vehicular Metaverse Services

Hongjia Wu, Hui Zeng, Zehui Xiong, Jiawen Kang, Zhiping Cai,
Tse-Tin Chan, *Member, IEEE*, Dusit Niyato, *Fellow, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*—Timely updating of Internet of Things (IoT) data is crucial for immersive vehicular metaverse services. However, challenges such as latency caused by massive data transmissions, privacy risks associated with user data, and computational burdens on metaverse service providers (MSPs) hinder continuous collection of high-quality data. To address these issues, we propose an immersion-aware model trading framework that facilitates data provision for services while ensuring privacy through federated learning (FL). Specifically, we first develop a novel multi-dimensional metric, the immersion of model (IoM), which assesses model value comprehensively by considering freshness and accuracy of learning models, as well as the amount and potential value of raw data used for training. Then, we design an incentive mechanism to incentivize metaverse users (MUs) to contribute high-value models under resource constraints. The trading interactions between MSPs and MUs are modeled as an equilibrium problem with equilibrium constraints (EPEC) to analyze and balance their costs and gains, where MSPs as leaders determine rewards, while MUs as followers optimize resource allocation. Furthermore, considering dynamic network conditions and privacy concerns, we formulate the reward decisions of MSPs as a multi-agent Markov decision process. To solve this, we develop a fully distributed dynamic reward algorithm based on deep reinforcement learning, without accessing any private information about MUs and other MSPs. Experimental results demonstrate that the proposed framework outperforms state-of-the-art benchmarks, achieving improvements in IoM of $38.3\%$ and $37.2\%$, and reductions in training time to reach the target accuracy of $43.5\%$ and $49.8\%$, on average, for the MNIST and GTSRB datasets, respectively.**

***Index Terms*—Equilibrium problem with equilibrium constraints, immersion-aware, incentive mechanism, resource allocation, vehicular metaverse.**

## I. Introduction

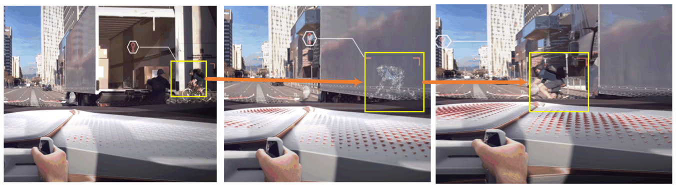


Fig. 1. An example of AR services in the vehicular metaverse[1].

THE metaverse, envisioned as a major evolutionary step for the Internet, aims to create a fully immersive, self-sustaining virtual environment for activities such as playing, working, and socializing [1]. This vision is propelled by advances in fifth- and sixth-generation (5G/6G) communication technologies, which offer low latency and high data throughput. These technologies play a critical role in seamlessly integrating the Internet of Things (IoT) data into metaverse services, thus bringing the once-fictional concept of immersive experiences closer to reality.

Metaverse services are beginning to reveal their vast potential across a broad spectrum of industries, from gaming and autonomous driving to education and marketing. Notably, the application of vehicles within the metaverse [2], [3] has attracted significant interest, particularly for the enhanced traffic safety enabled by state-of-the-art augmented reality (AR) technologies. Market report [4] forecasted that the global automotive metaverse market will grow from 1.9 billion in 2022 to 6.5 billion by 2030. Automakers such as BMW have increasingly invested in AR technology. As shown in Fig. 1, augmented information makes driving safer by showing potential hazards hidden behind the vehicle in front of you. Moreover, Nissan's [5] upcoming technology utilizes a 3D AR interface that merges the real world with the virtual world to provide the driver with augmented information about the surrounding area. In addition, Neuron mobility[2] has announced the launch of an innovative AR parking assistance system with the intention of improving passengers' parking routines and trip-end experiences.

The ability to capture information from the "real" world, particularly the ability to collect and process massive data from IoT devices, is the key to determining the success of immersive services (e.g., AR) in the vehicular metaverse. Meanwhile, the data must be processed and presented in a meaningful, responsive, and appropriately protected way. Technically, a

*(Corresponding author: Tse-Tin Chan.)*

H. Wu and T.-T. Chan are with the Department of Mathematics and Information Technology, The Education University of Hong Kong, Hong Kong SAR, China (e-mails: whongjia@eduhk.hk, tsetinchan@eduhk.hk).

Z. Xiong is with the Pillar of Information Systems Technology and Design, Singapore University of Technology and Design, Singapore (e-mail: zehui_xiong@sutd.edu.sg).

J. Kang is with the School of Automation, Guangdong University of Technology, China (e-mail: kavinkang@gdut.edu.cn).

H. Zeng and Z. Cai are with the College of Computer Science and Technology, National University of Defense Technology, China (e-mails: zenghui116@nudt.edu.cn, zpcai@nudt.edu.cn).

D. Niyato is with the College of Computing and Data Science, Nanyang Technological University, Singapore (e-mail: dniyato@ntu.edu.sg).

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

[1] Image source: https://www.jasoren.com/ar-in-automotive/

[2] https://immersive-technology.com/augmentedreality/neuron-introduces-new-ar-parking-assistant/

high-quality experience with AR services relies on accurate detection and classification of real-world objects (e.g., cars and pedestrians) under complex conditions [6]. To achieve this goal, sufficient valid data needs to be collected and processed in depth to detect and classify objects accurately. Therefore, it is essential to focus on effectively collecting, processing, and protecting the data that supports a safer and more enjoyable driving experience.

**Motivations.** The widely used data collection method, as adopted by Nissan [7] and studied in [8], involves gathering massive amounts of data through vehicle sensors, cameras, and roadside devices, and all data being processed centrally. While this approach has been effective for various applications, when it comes to the situation with multiple metaverse users (MUs) and metaverse service providers (MSPs) associated with different companies, the centralized data collection approach is not applicable and may lead to the following issues. First, AR services in the vehicular metaverse need to be highly immersive so that MUs feel fully immersed in the rendered environment, such as visualized driving. However, the data synchronization will be hindered by the latency from massive real-time data updates under unstable and resource-limited communication conditions. Note that the value of real-time data diminishes over time [9]. Also, delays can severely impact the MU's experience and cause dizziness [10]. Second, the data to be transmitted may be sensitive and private, such as location, movement, and biometrics, which can create a better immersive experience but may inevitably increase the privacy risk of MUs.

Federated learning (FL) has been adopted in prior work [6], [11]–[13] to enable collaborative model training without sharing raw data (e.g., sensor/imaging data from vehicles). In this approach, learning models are uploaded by individual MUs to MSPs for AR services, rather than transmitting large volumes of data centrally, thus significantly reducing the communication burden. However, MUs are typically self-interested and are reluctant to share learning models with MSPs due to the additional computation, communication, and energy overhead. To overcome this, incentive mechanisms using strategies such as contract theory [14], Stackelberg game [15], [16] and multi-winner sealed-bid auction [17], have been proposed to encourage MUs to contribute models. However, existing studies fail to explicitly assess the model value from multiple dimensions, making it difficult for both MSPs and MUs to quantify the benefits of the model for MSPs. Furthermore, most of these solutions are designed for a single MSP and do not consider the joint optimization of MUs' limited computational and communication resources. As a result, an efficient and privacy-preserving framework for data synchronization is needed to improve the immersive AR experience in the vehicular metaverse.

To address the above research gaps, we propose an immersion-aware model trading framework for a multi-MSP, multi-MU vehicular metaverse. This framework is designed to incentivize MUs to become active contributors by providing models tailored to the specific needs of MSPs. However, MUs have different sampling costs and limited computational and communication resources, while MSPs differ in their model preferences and compete with each other. These inherent asymmetries and competitive dynamics make it challenging to establish an efficient model trading ecosystem. This leads to two key research questions that underpin our work:

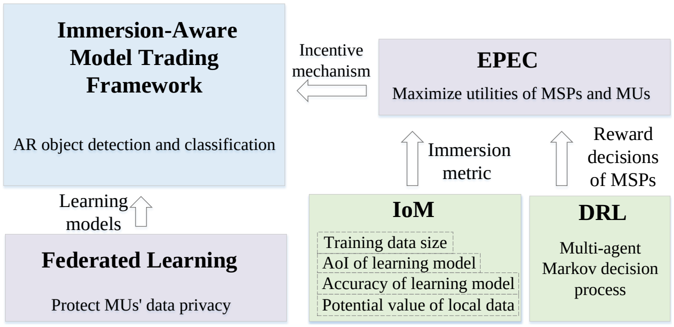

Immersion-Aware Model Trading Framework
AR object detection and classification
Incentive mechanism
EPEC
Maximize utilities of MSPs and MUs
Immersion metric
Reward decisions of MSPs
Learning models
Federated Learning
Protect MUs' data privacy
IoM
Training data size
AoI of learning model
Accuracy of learning model
Potential value of local data
DRL
Multi-agent Markov decision process

Fig. 2. The outline of an immersion-aware framework for FL-assisted vehicular metaverse.

- How can MUs be effectively incentivized to contribute high-value learning models that benefit vehicular metaverse services for diverse MSPs?
- How can the dynamic competitive interactions among MSPs be modeled to achieve a trade-off in gains between MUs and MSPs, ensuring sustainable and stable trading of learning models in the vehicular metaverse?

**Proposed Framework and Contributions.** As depicted in Fig. 2, our model trading framework consists of four components: FL mechanism, metric design, game modeling, and dynamic algorithms. Specifically, we first design a new metric called the immersion of the learning model (IoM) to evaluate the value of learning models contributed by MUs to MSPs. This metric jointly considers the freshness and accuracy of the learning model, as well as the amount and potential value of raw training data. Building on this metric, we propose an immersion-aware incentive mechanism that aligns the interests of both MUs and MSPs. Then, we model the dynamic competitive trading interactions as an equilibrium problem with equilibrium constraints (EPEC), which is a hierarchical optimization problem with equilibria at two levels [18]. Moreover, given the dynamic networks and the privacy concerns of MSPs, we formulate the reward decisions of MSPs as a multi-agent Markov decision process (MAMDP) and develop a multi-agent deep reinforcement learning (DRL)-based dynamic reward (MDDR) approach to obtain the reward decisions in a fully distributed manner. In summary, the key contributions of this work are as follows.

- *Incentive mechanism design for immersion-aware model trading.* From the perspectives of both MUs and MSPs, we propose an incentive mechanism that encourages MUs to contribute high-value learning models tailored to the specific demands of MSPs. To our knowledge, this is the first study focusing on incentive mechanism design for efficient and privacy-preserving data synchronization in multi-MU, multi-MSP vehicular metaverse environments.
- *Novel design of multi-dimensional metric.* To quantify immersion enhancement provided by MUs for AR services, we design an immersion metric of the learning model integrating four critical dimensions: the freshness and

accuracy of the learning model, as well as the amount and potential value of raw training data. Freshness is captured through age of information (AoI), while potential value is evaluated by the difference between model predictions and true labels. This metric enables fine-grained evaluation of model value under resource constraints and provides a basis for incentive and decision-making strategies in model trading.

- *Theoretical analysis and algorithm designs.* Given multiple resource-constrained MUs and competing MSPs, we model their interactions as an equilibrium problem with equilibrium constraints (EPEC). We theoretically prove the existence and uniqueness of the equilibrium, and further develop a fully distributed MDDR approach that adapts to dynamic environments and operates without accessing any private information of MUs or MSPs.
- *Performance evaluation.* We conduct extensive numerical simulations based on AR-related vehicle datasets to validate the efficacy and efficiency of MDDR and the proposed immersion-aware model trading framework. Numerical results show that our proposed mechanism improves the IoM by $38.3\%$ and $37.2\%$, and reduces the training time to reach the target accuracy by $43.5\%$ and $49.8\%$, on average, for the MNIST and GTSRB datasets, respectively, compared with state-of-the-art benchmarks.

The rest of this paper is organized as follows. Section II discusses the related work. In Section III, we present the system overview and design the immersion metric of the learning model. Section IV gives the game formulation, and Section V analyzes the existence of the equilibria at two levels. In Section VI, we give the detailed design of MDDR. Section VII shows numerical experiments to evaluate the framework performance, and finally Section VIII concludes the paper.

## II. Related Work

In this section, we survey work related to our study in terms of edge-enabled vehicular metaverse services, FL for AR, and incentive mechanisms for data synchronization in the metaverse.

### *A. Edge-Enabled Vehicular Metaverse Services*

Vehicular metaverse services require ultra-low latency and high reliability to enable immersive experiences such as AR navigation and real-time digital twins. To meet these demands, researchers have extensively explored edge-based strategies for task offloading and resource management. For instance, Feng *et al.* [19] proposed a resource allocation framework for AR-powered vehicular edge metaverses that aims to maximize overall system utility. Similarly, Khan *et al.* [20] introduced a cooperative framework integrating task offloading, sensing, learning, and communication to reduce transmission energy consumption and latency in resource-constrained environments. Beyond static offloading, Chen *et al.* [21] developed a multi-agent deep reinforcement learning approach for dynamic avatar task migration. In terms of security and privacy, Kang *et al.* [22] presented a cross-metaverse empowered dual pseudonym management framework to mitigate privacy leakage risks during the dynamic communications among vehicular edge metaverses. Moreover, an optimization framework for AR-assisted driving was proposed in [23] to enhance the robustness of metaverse maps against adversarial attacks. This framework maximizes the mean average precision (mAP) of adversarial patch detection while minimizing AR scene uplink latency and worst-case battery consumption. Recent studies have also explored applying FL to vehicular metaverse services as a means of preserving user privacy during distributed model training. These efforts, particularly in AR-enhanced metaverse environments, are discussed in the next section. To ensure service continuity and prevent user disengagement during unexpected failures, Qiu *et al.* [24] proposed deploying redundant backup virtual vehicle services and keeping the age of backup information to improve the reliability of services.

These works, from diverse perspectives, have significantly advanced the security, reliability, and performance of vehicular metaverse services, providing valuable insights for future research. Notably, distinct from existing studies, our work focuses on designing an incentive mechanism that transforms MUs into active contributors. These contributors efficiently provide high-value models to MSPs, thereby enhancing the overall service experience.

### *B. Federated Learning for AR in the Metaverse*

Privacy preservation is a critical concern in vehicular metaverse services, particularly when training models using sensitive user data. Federated learning (FL) has emerged as a promising technique to enable collaborative model training without sharing raw data. For example, Chen *et al.* [11] proposed a framework that integrates mobile edge computing and FL to solve the computational efficiency and low-latency object classification problem for AR services. However, they did not address how communication and computational resources should be allocated within the FL framework. To overcome this limitation, Zhou *et al.* [12] incorporated FL into mobile AR systems in the metaverse to enable collaborative model training. Their approach considers the trade-offs among energy consumption, execution latency, and model accuracy to meet varying demands across different application scenarios. To further improve learning efficiency, an adaptive and resource-efficient FL algorithm tailored for AR applications was proposed in [6]. This approach mitigates the effects of non-IID data distributions, reduces resource usage, and enhances the quality of experience. Additionally, Hazarika *et al.* [13] explored the integration of quantum computing with FL to provide a cost-efficient and adaptive solution for the dynamic nature of vehicular environments.

Incorporating FL into AR services brings the benefits of reduced communication latency and privacy protection. However, the existing studies fail to consider the integration of FL into AR for the multi-MSP and multi-MU metaverse scenario, while overlooking the selfishness of MUs, i.e., whether they are willing to participate in FL learning.

### *C. Incentive Mechanism for Data Synchronization in the Metaverse.*

The quality of metaverse services depends heavily on massive sensing data from the real world. To facilitate this, incentive mechanisms are widely employed to motivate MUs to contribute data and participate in model construction, thereby enhancing their service experience. For instance, Zhang *et al.* [25] proposed a vehicle-assisted data sensing framework that incentivizes vehicles to contribute sensing data to metaverse service, improving the driving experience for vehicle users. Similarly, Lin *et al.* [26] introduced a contract-based incentive design aimed at optimizing the long-term profits of digital twin service providers, addressing service congestion caused by stochastic demand responses. Xu *et al.* [27] developed a metaverse-based unmanned aerial vehicle (UAV) swarm system that uses digital twins and semantic communication to enhance data synchronization and reduce communication latency. Moreover, Lotfi *et al.* [28] introduced VMGuard, a novel four-layer security framework for the vehicular metaverse that protects against data poisoning attacks by using a reputation-based incentive mechanism. The framework leveraged user feedback and subjective logic modeling to assess the trustworthiness of devices. However, while these works successfully incentivize MUs to contribute raw data, they also pose a risk to user privacy due to potential data leakage.

To address the privacy concern, Kang *et al.* [14] developed a cross-chain FL framework with an AoI-based contract theory model under prospect theory to incentivize data sharing. They utilized a hierarchical cross-chain architecture with a main chain and multiple subchains to perform decentralized, privacy-preserving, and secure data training in both virtual and physical spaces. Baccour *et al.* [15] proposed a dual game-theoretic framework for federated meta-learning (FML) in metaverse services. They introduced a reputation system and a Stackelberg game-based incentive mechanism to enhance data privacy, minimize energy costs, and improve FML efficacy, outperforming traditional clustering methods in training performance and metaverse utility. Li *et al.* [16] designed a satisfaction metric that considers data size, AoI, and training latency. They integrated this metric into utility functions to incentivize node participation in FL while optimizing resource allocation. However, these works are designed for a single MSP scenario, limiting scalability and applicability in multi-server scenarios commonly encountered in large-scale metaverse environments.

Building on the concept of multiple MSPs, Zhang *et al.* [17] proposed a privacy-preserving auction mechanism for learning model as an NFT in Blockchain-driven Metaverse. They employed an imperfect information Stackelberg game to optimize the strategies of MUs and MSPs, and introduced a multi-winner sealed-bid auction mechanism to enhance the quality of FL-NFT. However, this approach primarily focuses on modeling NFTs and overlooks the joint optimization of computational and communication resources. Moreover, it fails to evaluate the model's value from multiple dimensions.

Inspired by the above existing works, this study focuses on multi-MSP and multi-MU scenarios in vehicular metaverse services. By assigning MUs the dual roles of service consumers and model contributors, we build a low-cost, privacy-preserving data support framework. This allows MUs to enjoy immersive in-vehicle services (e.g., AR navigation, digital twin interactions) while participating as distributed model contributors in service optimization. This framework continuously incentivizes MUs to contribute high-value learning models for metaverse services, thereby enhancing the overall service experience.

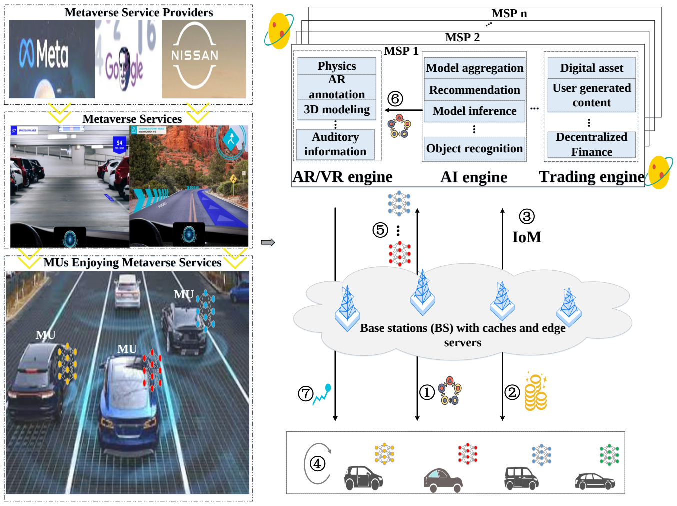


Fig. 3. Workflow of the immersion-aware model trading framework[3].

## III. System Model

We first present the system overview of the immersion-aware model trading framework in Section III-A. Next, we introduce the FL mechanism adopted by our framework in Section III-B and design the immersion metric of the learning model in Section III-C.

### *A. System Overview*

Fig. 3 illustrates the workflow of our immersion-aware model trading framework, designed to enhance the immersive and interactive AR experiences of MUs. MUs can contribute learning models to the AR services of MSPs for better immersive experiences. For example, services such as object detection can be improved by training on images of streets, pedestrians, and so on, captured by vehicles' cameras. In this way, AR services can be better applied to visible driving, with more accurate and timely hazard warnings from windscreens. The framework mainly consists of MSPs owned by different companies (e.g., Meta, Nissan, and Google), an infrastructure layer, and an interaction layer with the MUs that enjoy vehicular metaverse services. MSPs are powered by various technologies supporting vehicular metaverse services, such as AI engines, AR/VR engines, metaverse trading engines. The infrastructure layer with base stations (equipped with caches

[3] Image source: https://mixed-news.com/en/google-transforms-google-maps-into-the-backbone-of-our-ar-future/, https://about.facebook.com/meta, https://www.vanarama.com/blog/cars/4-ways-augmented-reality-will-revolutionise-the-automotive-industry.

TABLE I
DEFINITIONS OF NOTATION

| Notation | Definition |
|---|---|
| $m, M$ | Index of an MU, Number of MUs |
| $n, N$ | Index of an MSP, Number of MSPs |
| $\overline{\Delta}_{mn}$ | The average AoI of MU $m$'s model provided for MSP $n$ |
| $V_{mn}$ | The IoM contributed by MU $m$ to MSP $n$ |
| $f_{mn}$ | MU $m$'s computational resource used for MSP $n$ |
| $B_{mn}$ | MU $m$'s communication resource used for MSP $n$ |
| $X_{mn}$ | MU $m$'s training data set for MSP $n$ |
| $\omega_{mn}$ | The potential value of MU $m$'s local data for MSP $n$ |
| $\theta_m$ | MU $m$'s local accuracy threshold |

and edge servers) provides the basis of 5G/6G communication services for the interactions among MUs and MSPs. MUs play essential roles as both consumers and contributors to vehicular metaverse services.

The framework is a multi-engine configuration, and this paper focuses on two components: the trading engine and the AI engine of the MSP. The trading engine is responsible for incentivizing MUs to participate in mutually beneficial collaborations with MSPs, providing them with a trading platform to make their decisions (i.e., computational and communication resource allocation of MUs and reward decisions of MSPs). Then, each MU deciding to contribute utilizes the FL mechanisms integrated into AI engines to provide learning models for MSPs under the strategic guidance of the trading results. The specific workflow of the framework involves the following two phases.

**Phase I** (Incentive process): Initially, MSPs' AI engines broadcast their global models to all MUs in the vehicular metaverse[4] (① in Fig. 3). Then, the trading engines of MSPs determine their digital currency prices per IoM, i.e., rewards to MUs, and broadcast them to MUs (②). Next, MUs determine their allocation of computational and communication resources for contributing learning models to various MSPs based on the IoM, costs, and rewards given by MSPs (③). Based on the responses from MUs, the trading engines of MSPs adjust the digital currency prices to maximize their own utilities. Steps ② and ③ are repeated until an agreement is reached among MUs and MSPs.

**Phase II** (FL process): Each MU, guided by decisions obtained through tradings, allocates computational resources to local training based on different global models from MSPs and generates local learning models for different MSPs (④). Then, MUs upload their updated learning models to the corresponding AI engines of MSPs by consuming communication resources (⑤). Following this, AI engines aggregate models from all MUs to update their global models and transmit them to their AR/VR engines for utilization (⑥). Finally, the AR/VR engines provide enhanced AR services to MUs (⑦). Steps ④-⑦ are repeated until the guidance time $T$ ends. To ease the presentation, we summarize some important notations in Table I.

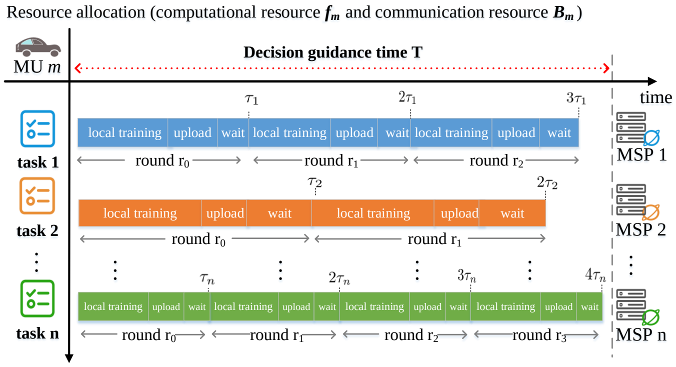


Fig. 4. Illustration of the FL mechanism with $n$ tasks for MU $m$.

### B. FL Framework Mechanism Adopted

Here, we consider a set of $\mathcal{M} = \{1, \dots, M\}$ MUs with the capacities of local computation and a set of $\mathcal{N} = \{1, \dots, N\}$ MSPs with FL synchronous tasks[5]. Each MSP initiates an FL synchronous task with a virtual deadline $\tau_n (n \in \mathcal{N})$. The MUs can join FL to generate local learning models and contribute to enhancing vehicular metaverse services. As shown in Fig. 4, MU $m$ provides learning models to different MSPs simultaneously for a given time period $T$, meaning multiple FL tasks can be processed in parallel. Moreover, the data collected at the $(r-1)$-th round is used as the local data for training at the $r$-th round. Without loss of generality, we consider that the trading decisions among MUs and MSPs are made at the beginning of the period $T$.

In detail, let $X_{mn}$ be the set of data samples for task $n$ of MU $m$ as a set of input-output pairs $\sum_{i=1}^{|X_{mn}|}(y_i, z_i)$, where $y_i$ is an input sample vector with $d$ features, $z_i$ is the labeled output (i.e., ground truth) value of sample $y_i$, and $|X_{mn}|$ is the size of set $X_{mn}$. The data, such as images of streets and pedestrians, can be generated by high-definition cameras inside and outside the vehicles. The steps involved in each iteration $r \in \{1, 2, \dots, R\}$ are as follows:

- ***Publishing Tasks*** : The AI engine of MSP $n$ broadcasts its global model $\jmath_n^{(r)}$ (e.g., object detection models used in AR services for visible driving and intuitive road safety) to all MUs in the $r$-th round.
- ***Local Training***: Each MU, for example, MU $m$, trains $\jmath_n^{(r)}$ with the most recently collected data $X_{mn}^{(r)}$ by stochastic gradient descent (SGD) within certain local rounds. The number of local training rounds depends on its local accuracy threshold.
- ***Uploading and Aggregation***: MUs transmit their updated local model $\jmath_{mn}^{(r)}$ to the AI engine of MSP $n$. After receiving all the updated models, the MSP $n$'s AI engine aggregates them through the following weighted average:

$$\jmath_n^{(r+1)} = \jmath_n^{(r)} + \sum_{m=1}^{M} \frac{|X_{mn}^{(r)}|}{|X_n^{(r)}|} (\jmath_{mn}^{(r)} - \jmath_n^{(r)}), \tag{1}$$

[4] The FL global models owned by different MSPs we consider could be object detection or classification models for intelligent terrain mapping, intuitive road safety, visible driving, etc. Note that it is feasible to train on multiple tasks with the same local data. For instance, for the same image, intuitive road safety focuses on pedestrians and vehicles, while intelligent terrain mapping focuses on landmarks and routes.

[5] That is, the update time of the global model is limited by the slowest MU, i.e., the AI engine of the MSP must wait to receive updated models from all MUs before performing model aggregation. The advantages of synchronous FL are high model accuracy and fast convergence [29].

and then obtains the new global model $\jmath_n^{(r+1)}$ for the next iteration.

- ***Model Utilization***: After global aggregation, MSP $n$'s AR engine obtains an updated global model that can be used for corresponding 3D modeling and other functions [11]. The global rounds are iterated until a specific requirement is met, such as reaching a certain level of accuracy or a deadline.

### *C. Immersion Metric of Learning Model*

MSPs determine the rewards for MUs based on the immersion of the learning model (IoM) metric. Correspondingly, MUs need to decide the IoMs of the models provided to MSPs by allocating their computational and communication resources to maximize the benefits. This process highlights the key role of "IoM" in the interactions between the two parties, serving as both an evaluation criterion and a basis for decision-making. Unlike traditional FL and incentive mechanisms, in vehicular metaverse scenarios with immersion requirements, both the contribution and freshness of the learning model need to be considered[6], which will collectively affect the immersive experience of AR services, i.e., whether virtual objects can be placed in the physical world accurately and promptly. With the above consideration, we propose the IoM, a metric that jointly considers the contribution prediction $I_{mn}$ and the age of information (AoI) $\Delta_{mn}$ [30] to measure the value of a learning model that MU $m$ brought to the vehicular metaverse service of MSP $n$, denoted as

$$V_{mn} = I_{mn}(\tau_n - \overline{\Delta}_{mn}), \tag{2}$$

where the components are detailed as follows. In the rest of this paper, we refer to the trading engine of MSP as MSP.

*1) Contribution Prediction $I$:* The contribution prediction $I_{mn}$ from MU $m$ to MSP $n$ is determined by the accuracy $\theta_m$ of the learning model, the total amount of training data $\lfloor\frac{T}{\tau_n}\rfloor|X_{mn}|$, and the potential value $\omega_{mn}$ of local data. Prior work [31] characterized the contribution of a learning model in terms of a logarithmic function of the amount of local training data. However, evaluating the contribution solely by the training data size is one-sided, as there may be a large amount of redundant data or the validity of the learning model cannot be guaranteed. It is more practical to comprehensively evaluate the amount of training data, model accuracy, and the potential value of local data; thus, we denote $I_{mn}$ as

$$I_{mn} = \frac{\omega_{mn}\epsilon\ln(1+\eta\lfloor\frac{T}{\tau_n}\rfloor|X_{mn}|)}{\theta_m}, \tag{3}$$

where $\lfloor\frac{T}{\tau_n}\rfloor$ is the number of iterations that the task of MSP $n$ can be performed within $T$. $\epsilon$ and $\eta$ are system parameters from experiments as obtained in [32]. $\theta_m \in (0,1)$ characterizes the accuracy of local training as decided by the MU. Here, $\theta \to 0$ means that a high-accuracy learning model is available, while $\theta \to 1$ implies that the accuracy of the learning model is low. The local data potential value $\omega_{mn}$ describes the difference between model predictions and true labels, defined as

$$\omega_{mn} = \frac{1}{|X_{mn}|}\sum_{i=1}^{|X_{mn}|}(\hat{z}_i - z_i)^2, \tag{4}$$

where $\hat{z}_i$ denotes the inferred label from the current global model and $z_i$ is its ground-truth label. If the value of $\omega_{mn}$ is large, it indicates that the current global model is not working well on local data or there are unseen samples. Therefore, updating the model by training on these data may lead to better performance of the global model. If the value of $\omega_{mn}$ is small, it means that the current data has little potential to further improve the global model since the knowledge of the data has already been learned or the data size is small. Note that to shorten the decision time, we use the potential value of the initial local data to be a proxy for the average potential value of all data within $T$. This is a reasonable approach when $T$ is not too long.

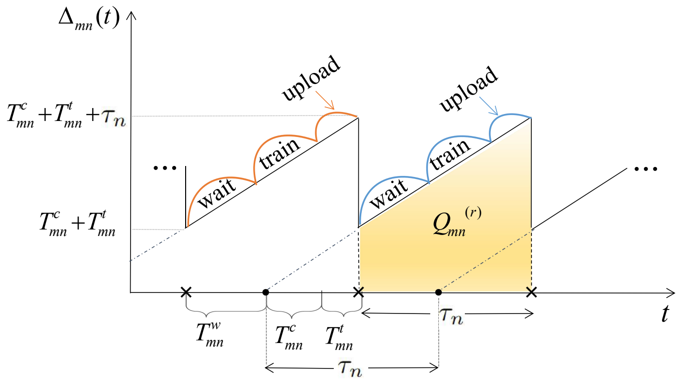


Fig. 5. The AoI evolution of the models sent from MU $m$ to MSP $n$.

*2) Age of Information $\Delta$:* The whole FL process involves model training, uploading, and waiting, which together constitute a complete cycle of model updating. From the perspective of MSP $n$, the age evolution of the model contributed by MU $m$ is shown in Fig. 5. Here, the filled circles denote the time instances when MU $m$ starts training the learning model. The intersections show the time instances when MSP $n$ receives the corresponding updated model, at which point the AoI drops to the lowest. We denote that at time $t$, the generation time of the last update received by MSP $n$ from MU $m$ is $s(t)$. Then, the instantaneous AoI of MU $m$'s learning model measured at MSP $n$, $\Delta_{mn}(t)$, is

$$\Delta_{mn}(t) = t - s(t), \tag{5}$$

which shows the elapsed time since the generation of the latest updated model. A smaller instantaneous AoI means that the latest model received is fresher at that time. To obtain a comprehensive understanding of the overall freshness of model updates over a certain period, we use *average* AoI to evaluate the freshness of the models. With the aid of Fig. 5, the time-averaged AoI of the learning model from MU $m$ to MSP $n$

[6] For example, an MSP may receive fresher learning models but with lower contributions, or less fresh learning models but with higher contributions.

can be expressed as

$$\begin{aligned}\overline{\Delta}_{mn} &= \lim_{Z\to\infty}\frac{1}{Z}\int_0^Z \Delta_{mn}(t)\,dt = \lim_{Z\to\infty}\sum_{r=1}^{Z}\frac{{Q_{mn}}^{(r)}}{Z}\\ &= \frac{1}{2}\tau_n + T^c_{mn} + T^t_{mn},\end{aligned} \tag{6}$$

where ${Q_{mn}}^{(r)}$ is the $r$-th trapezium under the curve. A lower average AoI indicates that the updated models are generally fresher over a long period.

Since MU $m$ constantly collects data throughout the FL process, the amount of training data collected for training in each round is $|X_{mn}| = x_m\tau_n$, where $x_m$ denotes the data (number of floats) collected per unit time. The cumulative time $T^c_{mn}$ for local training is determined by the amount of training data, computational resource $f_{mn}$ allocated to MSP $n$, and its local accuracy threshold $\theta_m$, which is equivalent to

$$T^c_{mn} = \log(1/\theta_m)\frac{x_m\tau_n}{f_{mn}}. \tag{7}$$

A smaller value of $\theta$ indicates higher accuracy, but leads to a higher MU cost, i.e., the number of local iterations, which is upper bounded by $\log(1/\theta_m)$ [33].

Furthermore, the upload time $T^t_{mn}$ is directly related to the number of model parameters $b_{mn}$ and the resource $B_{mn}$ (i.e., uplink bandwidth) used to communicate between MU $m$ and MSP $n$, defined as

$$T^t_{mn} = \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})}, \tag{8}$$

where $\varsigma_{mn} = \frac{p^t_m g_{mn}}{\sigma^2}$ denotes the signal-to-noise ratio (SNR) [34] for the communication between MU $m$ and MSP $n$. $g_{mn}$ and $p^t_m$ are the corresponding channel gain and the transmission power of MU $m$, respectively. $\sigma^2$ is additive white Gaussian noise (AWGN).

Finally, by substituting (7) and (8) into (6), $\overline{\Delta}_{mn}$ can be expressed as

$$\overline{\Delta}_{mn} = \frac{1}{2}\tau_n + \frac{x_m\tau_n\log(1/\theta_m)}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})}. \tag{9}$$

## IV. Game Formulation

We first define the utility functions of MU $m$ and MSP $n$ in Section IV-A and Section IV-B, respectively. Then, we formulate the interactions among MUs and MSPs as an equilibrium problem with equilibrium constraints (EPEC) in Section IV-C, where the equilibrium criterion exists at both the level of MUs and MSPs due to the conflicting interests between them.

### A. Utility of MU

We define the utility of MU $m$ composed of the rewards from MSPs and the cost incurred for contribution, which is also affected by resource constraints and immersion requirements for consuming vehicular metaverse services. The cost incurred due to local training and model uploading can be expressed as

$$C_{mn} = c^f_m\log(1/\theta_m)f_{mn} + c^B_m B_{mn}, \tag{10}$$

where $c^f_m$ and $c^B_m$ denote the cost factors of computational and communication resources, respectively. Let $\boldsymbol{f}_m \triangleq (f_{m1}, f_{m2}, \ldots, f_{mn})$ and $\boldsymbol{B}_m \triangleq (B_{m1}, B_{m2}, \ldots, B_{mn})$ be the computational and communication allocation profiles of MU $m$, respectively. Then, the utility maximization problem for MU $m$ within $T$ can be formulated as

**Problem 1.**

$$\begin{aligned}&\max\ \Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m) = \sum_{n=1}^{N}(p_{mn}V_{mn} - C_{mn}),\\ &s.t.\, C1: \sum_{n=1}^{N} f_{mn} < f^{max}_m,\ \sum_{n=1}^{N} B_{mn} < B^{max}_m,\\ &\quad C2: \log(1/\theta_m)\frac{x_m\tau_n}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})} \le \tau_n,\\ &\quad C3: \frac{S_m}{f^{max}_m - \sum_{n=1}^{N} f_{mn}} < T^{req},\end{aligned} \tag{11}$$

where $S_m$ indicates the minimum computational resource required to enjoy other basic services. Constraints C1 and C3 indicate that the MU's total computational and communication resources are limited and cannot be fully used to contribute learning models, given the MU's demand for basic services. Constraint C2 ensures that the total time for local training and uploading cannot exceed the time constraint $\tau_n$ for each round. The goal of each MU is to choose the optimal allocation of computational $\boldsymbol{f}_m$ and communication $\boldsymbol{B}_m$ resources to maximize its utility. Note that $V_{mn}$ must be greater than 0. This is because if $V_{mn} \le 0$ and MU $m$ is involved in contributing learning models, the utility is calculated as $p_{mn}V_{mn} - C_{mn} \le 0$, resulting in MU $m$ choosing not to participate in the trading.

### B. Utility of MSP

Each MSP has a gain function $\psi$ associated with the IoM to measure the benefits of all the learning models from MUs to the service. In this paper, we adopt $\psi_n = \mu_n \ln(1 + \sum_{m=1}^{M} V_{mn})$ as a monotonically increasing, differentiable, strictly concave IoM function, which is a simplification of the widely adopted function [35]. Let $\boldsymbol{p}_n \triangleq (p_{1n}, \ldots, p_{mn})$ and $\boldsymbol{p}_{\text{-}n}$ represent the reward profile of MSP $n$ and all other MSPs except MSP $n$, respectively. Then, the optimization problem for the MSP is defined as

**Problem 2.**

$$\begin{aligned}&\max\ \Psi_n(\boldsymbol{p}_n, \boldsymbol{p}_{\text{-}n}) = \psi_n - \sum_{m=1}^{M} p_{mn}V_{mn},\\ &\qquad s.t.\ p_{mn} > 0, n \in \mathcal{N}, m \in \mathcal{M},\end{aligned} \tag{12}$$

where $\mu_n$ refers to the profit conversion coefficient [36] from IoM, which is adjusted according to the AR services offered by the different MSPs. Note that the other MSPs' decisions $\boldsymbol{p}_{\text{-}n}$ are captured by the MUs' responses. Moreover, when MSP $n$ determines its reward $\boldsymbol{p}_n$ for different MUs, the MSP needs to consider the rewards offered by other MSPs (i.e., $\boldsymbol{p}_{\text{-}n}$ ) as well as the strategies of all MUs (i.e., $(\boldsymbol{f}_m, \boldsymbol{B}_m), m \in \mathcal{M}$). This

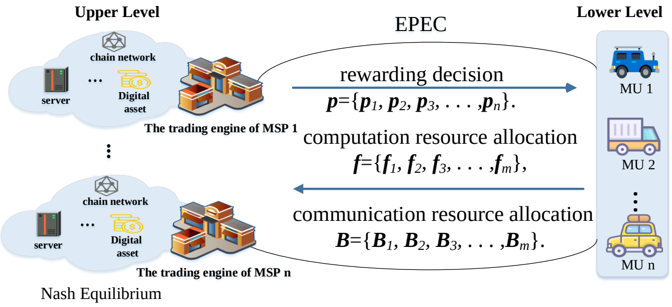

Upper Level
EPEC
Lower Level
chain network
server
Digital asset
The trading engine of MSP 1
rewarding decision
$\boldsymbol{p}$={$\boldsymbol{p}_1$, $\boldsymbol{p}_2$, $\boldsymbol{p}_3$, . . . ,$\boldsymbol{p}_n$}.
computation resource allocation
$\boldsymbol{f}$={$\boldsymbol{f}_1$, $\boldsymbol{f}_2$, $\boldsymbol{f}_3$, . . . ,$\boldsymbol{f}_m$},
communication resource allocation
$\boldsymbol{B}$={$\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{B}_3$, . . . ,$\boldsymbol{B}_m$}.
MU 1
MU 2
MU n
chain network
server
Digital asset
The trading engine of MSP n
Nash Equilibrium

Fig. 6. The hierarchical structure of equilibrium problem with equilibrium constraints (EPEC).

thereby leads to reward competition among MSPs. Therefore, the optimization among MSPs can be considered as a non-cooperative game, termed a multi-MSP rewarding game $\Omega$ as follows.

**Definition 1.** *A multi-MSP rewarding game $\Omega$ is a tuple $\Omega = \{\mathcal{N}, \boldsymbol{p}, \boldsymbol{\Psi}\}$ defined by*

- *Players: The set of MSPs;*
- *Strategies: The reward decisions $\boldsymbol{p}_n$ of any MSP $n$;*
- *Utilities: The vector $\boldsymbol{\Psi} = \{\Psi_1, \Psi_2, \ldots, \Psi_n\}$ contains the utility functions of all the MSPs defined in (12).*

### C. Multi-MSP Multi-MU Game as EPEC

As shown in Fig. 6, MUs and MSPs negotiate allocation strategies of computational and communication resources and reward decisions to maximize their benefits. In the upper level, MSPs determine the rewards they are willing to offer by considering their cost, the responses of MUs, and the decisions of other MSPs. In the lower level, each MU $m$ gives the optimal allocation response for computational and communication resources by considering its resource constraints, cost, and rewards from different MSPs. The objective of EPEC is to find the equilibria at two levels, i.e., the point at which the MSPs' (leaders') utilities are maximized given that the MUs (followers) will choose their best responses. For the proposed EPEC, the equilibria at two levels are defined as follows.

**Definition 2.** *Let $(f^*_{mn}, B^*_{mn})$ and $p^*_{mn}$ denote the optimal computational and communication resource allocation of MU $m \in \mathcal{M}$ and the optimal reward decision of MSP $n \in \mathcal{N}$, respectively. Then, the points $(f^*_{mn}, B^*_{mn})$ and $p^*_{mn}$ are the equilibria at two levels if the following conditions hold:*

$$\begin{aligned}
&\Phi_m((\boldsymbol{f}^*_m, \boldsymbol{B}^*_m), \boldsymbol{p}^*_n) \geq \Phi_m((\boldsymbol{f}_m, \boldsymbol{B}_m), \boldsymbol{p}^*_n), \forall m \in \mathcal{M},\\
&\Psi_n\left(\boldsymbol{f}^*_m(\boldsymbol{p}^*_n, \boldsymbol{p}^*_{-n}), \boldsymbol{B}^*_m(\boldsymbol{p}^*_n, \boldsymbol{p}^*_{-n}), \boldsymbol{p}^*_n, \boldsymbol{p}^*_{-n}\right) \geq\\
&\Psi_n\left(\boldsymbol{f}^*_m(\boldsymbol{p}^*_n, \boldsymbol{p}^*_{-n}), \boldsymbol{B}^*_m(\boldsymbol{p}^*_n, \boldsymbol{p}^*_{-n}), \boldsymbol{p}_n, \boldsymbol{p}^*_{-n}\right), \forall n \in \mathcal{N},
\end{aligned} \tag{13}$$

*where $\boldsymbol{p}^*_{-n}$ denotes the optimal rewards offered by other MSPs.*

In summary, the MSPs' optimization problems are formulated as the following EPEC problems:

$$\begin{aligned}
&\max_{p_n}\ \Psi_n = \psi_n - \sum_{m=1}^{M}\left[p_{mn}I_{mn}\left(\frac{1}{2}\tau_n\right.\right.\\
&\qquad\left.\left. - \frac{x_m\tau_n\log(1/\theta_m)}{f^*_{mn}} - \frac{b_{mn}}{B^*_{mn}\log_2(1+\varsigma_{mn})}\right)\right],\\
&\text{s.t.}\quad \begin{cases} p_{mn} > 0, n \in \mathcal{N}, m \in \mathcal{M},\\ (\boldsymbol{f}^*_m, \boldsymbol{B}^*_m) = \arg\max \Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m),\\ \qquad\qquad \text{s.t.}\, C1, C2, C3. \end{cases}
\end{aligned} \tag{14}$$

To investigate the above EPEC, we address the lower level (**Problem 1**) and the upper level (**Problem 2**) by using the backward induction methods in the following section.

## V. EPEC Analysis and Solutions

In this section, we utilize backward induction to analyze the EPEC formulated in (14). Specifically, we prove the existence and uniqueness of the equilibrium solutions at two hierarchical levels: (1) the optimal allocation of computational and communication resources for MUs (lower level), and (2) the optimal reward decisions for MSPs (upper level).

### A. Lower Level: Optimal Resource Allocation for MUs

In the lower level of EPEC, for any reward decisions $\boldsymbol{p}$ given by MSPs, MU $m$ aims to solve **Problem 1** in (11) to determine its optimal computational and communication resource allocations, i.e., $\boldsymbol{f}^*_m$ and $\boldsymbol{B}^*_m$, to maximize its utility. Below, we analyze and derive the unique optimal allocation.

**Definition 3.** *A maximization problem has a unique global optimum if the following conditions are satisfied [37]: (1) The objective function is strictly concave; (2) All constraints (equality and inequality) are affine; and (3) The strategy space is compact and convex.*

**Theorem 1.** ***Problem 1** is strictly concave and has a unique globally optimal solution in the lower level. That is, for each MU $m$, there exists a unique resource allocation tuple $(\boldsymbol{f}^*_m, \boldsymbol{B}^*_m)$ that maximizes its utility.*

*Proof.* We first examine the Hessian matrix of MU $m$'s utility $\Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m)$ with respect to $(f_{mn}, B_{mn})$. Let this Hessian matrix be denoted by $\boldsymbol{H}_m$, which can be block-diagonalized as

$$\boldsymbol{H}_m = \begin{bmatrix} \boldsymbol{H}^f_m & 0 \\ 0 & \boldsymbol{H}^B_m \end{bmatrix}. \tag{15}$$

The block matrix $\boldsymbol{H}^f_m$ can be computed as the second-order partial derivative of $\Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m)$ with respect to $f_{mn}$, i.e.,

$$\begin{aligned}
\boldsymbol{H}^f_m &= \left[\frac{\partial^2\Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m)}{\partial f_{mn}\partial f_{mn'}}\right]_{n,n'\in\mathcal{N}}\\
&= -diag\left(h^f_{m1}, h^f_{m2} \ldots, h^f_{mn}\right) < \boldsymbol{0},
\end{aligned} \tag{16}$$

where $h^f_{mn} = p_{mn}I_{mn}\frac{2x_m\tau_n}{f^3_{mn}}\log(1/\theta_m)$. Similarly, the block matrix $\boldsymbol{H}^B_m$ can be calculated by

$$\begin{aligned}
\boldsymbol{H}^B_m &= \left[\frac{\partial^2\Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m)}{\partial B_{mn}\partial B_{mn'}}\right]_{n,n'\in\mathcal{N}}\\
&= -diag\left(h^B_{m1}, h^B_{m2}, \ldots, h^B_{mn}\right) < \boldsymbol{0},
\end{aligned} \tag{17}$$

where $h_{mn}^{B} = \frac{2b_{mn}p_{mn}I_{mn}}{B_{mn}^{3}\log_2(1+\varsigma_{mn})}$.

We find that $\boldsymbol{H}_m^f$ and $\boldsymbol{H}_m^B$ are both negative semi-definite. Hence, $\boldsymbol{H}_m$ is negative semi-definite, implying $\Phi_m(\boldsymbol{f}_m, \boldsymbol{B}_m)$ is strictly concave. Since the constraints of **Problem 1** are affine and the strategy space is closed and bounded, **Definition 3** ensures the existence of a unique global optimum $(\boldsymbol{f}_m^*, \boldsymbol{B}_m^*)$. □

Consequently, MU $m$ has a unique best-response resource allocation for each MSP by solving **Problem 1**. From the Karush–Kuhn–Tucker (KKT) conditions, we establish the following proposition.

**Proposition 1**. Given the digital currency price offered by the MSPs and the Lagrange multipliers induced by the constraints, the optimal computational resource $f_{mn}$ allocated for MSP $n$ by MU $m$ satisfies:

$$f_{mn}^* = \begin{cases} \sqrt{\frac{p_{mn}I_{mn}x_m\tau_n}{c_m^f}}, & p_{mn} \geq \frac{F^2}{I_{mn}}, \\ F\sqrt{\frac{x_m\tau_n}{c_m^f}}, & 0 < p_{mn} < \frac{F^2}{I_{mn}}, \\ 0, & \text{otherwise}, \end{cases} \tag{18}$$

and the optimal communication bandwidth $B_{mn}$ is:

$$B_{mn}^* = \begin{cases} \sqrt{\frac{p_{mn}I_{mn}b_{mn}}{c_m^B\log_2(1+\varsigma_{mn})}}, & p_{mn} \geq \frac{F^2}{I_{mn}}, \\ F\sqrt{\frac{b_{mn}}{c_m^B\log_2(1+\varsigma_{mn})}}, & 0 < p_{mn} < \frac{F^2}{I_{mn}}, \\ 0, & \text{otherwise}, \end{cases} \tag{19}$$

where $F = \log(1/\theta_m)\sqrt{\frac{x_m c_m^f}{\tau_n}} + \frac{\sqrt{b_{mn}c_m^B}}{\tau_n\sqrt{\log_2(1+\varsigma_{mn})}}$. For the detailed proof of **Proposition 1**, please refer to **Appendix A**.

According to **Proposition 1**, the optimal strategy for each MU $m$ is influenced by five main factors: $p_{mn}$, $I_{mn}$, $c_m^f$, $c_m^B$ and $\tau_n$. Specifically, MU $m$ focuses on the rewards paid by MSPs, and the MU tends to put more computational and communication resources into the contribution when $p$ increases. Moreover, MUs are more willing to invest more resources when the potential contribution prediction $I_{mn}$ is higher, resulting in greater benefits. In addition, the virtual deadline $\tau_n$ set by the MSP $n$ determines the minimum criteria for resource allocation.

### *B. Upper Level: Optimal Reward Equilibrium among MSPs*

In the upper level of the EPEC, each MSP $n$ competes with other MSPs and determines its reward vector $\boldsymbol{p}_n$. Specifically, given the responses $(\boldsymbol{f}, \boldsymbol{B})$ from all MUs, and other MSPs' decisions $\boldsymbol{p}_{\text{-}n}$, the optimal reward decision $\boldsymbol{p}_n$ of MSP $n$ can be obtained by solving **Problem 2**, defined as follows.

**Proposition 2**. Given other MSPs' reward vectors $\boldsymbol{p}_{\text{-}n}$, the optimal strategy of MSP $n$ is

$$\boldsymbol{p}_n^* = arg \max_{p_{mn}>0} \Psi_n\left(\boldsymbol{f}_n^*(\boldsymbol{p}_n, \boldsymbol{p}_{-n}), \boldsymbol{B}_n^*(\boldsymbol{p}_n, \boldsymbol{p}_{-n}), \boldsymbol{p}_{-n}\right). \tag{20}$$

Then, we analyze the existence and uniqueness of the optimal reward decision equilibrium in **Theorem 2**.

**Theorem 2.** *There exists a unique Nash equilibrium in the multi-MSP rewarding game $\Omega$, ensuring a unique optimal set of reward decisions $\{\boldsymbol{p}_n^*\}_{n\in\mathcal{N}}$.*

*Proof.* We define the Hessian matrix of $\Psi_n$ with respect to its reward vector $\boldsymbol{p}_n$ as $(\boldsymbol{\Lambda}_n + \boldsymbol{H}_n)$. The matrix $\boldsymbol{\Lambda}_n = diag\left(\frac{\partial^2\Psi_n}{\partial p_{1n}^2}, \ldots, \frac{\partial^2\Psi_n}{\partial p_{mn}^2}\right)$ and the second-order partial derivative matrix $\boldsymbol{H}_n$ is expressed by

$$\boldsymbol{H}_n = \begin{bmatrix} 0 & \frac{\partial^2\Psi_n}{\partial p_{1n}\partial p_{2n}} & \cdots & \frac{\partial^2\Psi_n}{\partial p_{1n}\partial p_{mn}} \\ \frac{\partial^2\Psi_n}{\partial p_{2n}\partial p_{1n}} & 0 & \cdots & \frac{\partial^2\Psi_n}{\partial p_{2n}\partial p_{mn}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2\Psi_n}{\partial p_{mn}\partial p_{1n}} & \frac{\partial^2\Psi_n}{\partial p_{mn}\partial p_{2n}} & \cdots & 0 \end{bmatrix}, \tag{21}$$

where

$$\begin{aligned} \frac{\partial^2\Psi_n}{\partial p_{mn}^2} = \mu_n &\frac{V_{mn}''(1+\sum_{m=1}^M V_{mn}) - (V_{mn}')^2}{(1+\sum_{m=1}^M V_{mn})^2} \\ &- 2V_{mn}' - p_{mn}V_{mn}'' < 0, \forall m \in \mathcal{M}, \end{aligned} \tag{22}$$

$$\begin{aligned} \frac{\partial^2\Psi_n}{\partial p_{mn}\partial p_{m'n}} = -\,\mu_n &\frac{V_{mn}'V_{m'n}'}{(1+\sum_{m=1}^M V_{mn})^2} < 0, \\ &\forall m \in \mathcal{M}, m \neq m'. \end{aligned} \tag{23}$$

The proof of $\frac{\partial^2\Psi_n}{\partial p_{mn}^2} < 0$ and $\frac{\partial^2\Psi_n}{\partial p_{mn}\partial p_{m'n}} < 0$ can be found in **Appendix B**. We randomly choose a vector $\boldsymbol{h} \in \mathbb{R}^{M\times 1}$ with elements not all 0. For $\boldsymbol{\Lambda}_n$, it is easy to see that $\boldsymbol{h}^T\boldsymbol{\Lambda}_n\boldsymbol{h} = \sum_m \frac{\partial^2\Psi_n}{\partial p_{mn}^2}(h^m)^2 < 0$. Furthermore, $\boldsymbol{h}^T\boldsymbol{H}_n\boldsymbol{h} = \frac{-\mu_n}{(1+\sum_m V_{mn})^2}\left(\sum_m h^m V_{mn}'\right)^2 < 0$. Thus, we have $\boldsymbol{h}^T(\boldsymbol{\Lambda}_n + \boldsymbol{H}_{\boldsymbol{n}})\boldsymbol{h} = \boldsymbol{h}^T\boldsymbol{\Lambda}_n\boldsymbol{h} + \boldsymbol{h}^T\boldsymbol{H}_n\boldsymbol{h} < 0$, indicating that the utility function $\Psi_n$ is strictly concave. So far, the uniqueness of the Nash equilibrium [37] solution to the multi-MSP rewarding game $\Omega$ has been proved. □

Utilizing **Theorem 1** and **Theorem 2**, we prove the existence and uniqueness of the hierarchical equilibrium in the proposed EPEC model. Concretely, each MU has a unique best-response resource allocation strategy in reaction to any given MSP rewards, while each MSP determines a unique optimal pricing to maximize its own utility given the rewards of other MSPs.

Although the backward induction approach theoretically ensures an equilibrium solution to (14), solving it in real-world vehicular metaverse environments can be challenging due to dynamic network conditions and limited system knowledge (e.g., private cost parameters). In the following section, we address these practical issues by introducing a fully distributed reinforcement learning approach, enabling MSPs to adaptively and privately optimize their reward decisions in real time.

## VI. Dynamic Reward Strategy for MSPs

According to the EPEC proven above, it is possible to obtain an optimal solution. However, the optimal approach faces the following practical challenges: (i) In a time-varying network, the optimal approach is time-consuming due to the complexity of (12), which is a non-linear problem with a complicated structure. Additionally, the reward decisions of MSPs are tightly coupled. (ii) MSPs must have prior knowledge of private information about MUs (e.g., cost factors) to determine their rewards, which raises privacy concerns for MUs. To address these challenges, we further formulate the multi-MSP

**Algorithm 1:** Entire Process of the Framework.

```
1  Input: Action threshold $[p_{min}, p_{max}]$ and number of episodes $\mathcal{T}$;
2  Initialization: local observation $s_n$, actor network $\boldsymbol{\alpha}_n$, critic network $\boldsymbol{\beta}_n$, and episode buffer $\mathcal{D}_n$ for each MSP agent $n \in \mathcal{N}$;
3  # Trading process using MDDR:
4  for episode $= 1, 2, \ldots, \mathcal{T}$ do
5      Concurrently for each MSP agent $n \in \mathcal{N}$:
6      for epoch $k = 1, \ldots, |\mathcal{D}_n|$ do
7          Observe space $s_n(k)$;
8          Choose price $\boldsymbol{p}_n(k) \in [p_{min}, p_{max}]$ by sampling from its current policy $\pi_{\alpha_n}(s_n(k))$;
9          Broadcast reward decisions to MUs;
10         Interact with the environment and receive responses from the MUs;
11         Calculate utility according to (12) ;
12         Store transition $e_n(k) = [s_n(k), \boldsymbol{p}_n(k), \mathcal{U}_n(k), s_n(k+1)]$;
13     end
14     Update actor network $\boldsymbol{\alpha}_n$ and critic network $\boldsymbol{\beta}_n$ for each MSP agent $n$;
15     Clear the episode buffer $\mathcal{D}_n$ for each MSP agent $n$;
16 end
17 Return: Reward decisions $\boldsymbol{p}$ of MSPs and resource allocation $[\boldsymbol{f}, \boldsymbol{B}]$ of MUs;
18 # FL training guided by trading results within time $T$:
19 foreach MSP agent $n \in \mathcal{N}$ do
20     foreach MU $m \in \mathcal{M}$ do
21         Based on the trading results, utilize $f_{mn}$ and $B_{mn}$ to perform the local FL task for MSP $n$, and then upload the model to MSP $n$.
22     end
23     Aggregate received local models and issue an updated global model;
24     Supply the updated model to the AR engine;
25 end
```

rewarding game as a multi-agent Markov decision process (MAMDP) [35], which adapts to dynamic channel conditions. In this framework, each MSP is modeled as an individual agent that makes intelligent reward decisions.

### A. MAMDP for Multi-MSP Rewarding Game $\Omega$

We train the reward model based on the state-of-the-art policy gradient method proximal policy optimization (PPO) for the reasons described in [38]. The $MAMDP = \langle \mathcal{S}_n, \mathcal{A}_n, \mathcal{P}_n, \mathcal{U}_n \rangle$ for multi-MSP rewarding game is composed of state space $\mathcal{S}_n \triangleq \{s_n\}$, action space $\mathcal{A}_n \triangleq \{\boldsymbol{p}_n\}$, state transition probability $\mathcal{P}_n \triangleq \{P_n\}$, and utility $\mathcal{U}_n \triangleq \{\Psi_n\}$. For the MSP agent, the local observation contains the channel information and the responses of MUs, all of which are captured in the IoM of (2). Thus, we set the local observation of MSP $n$ at the $k$-th stage game defined as

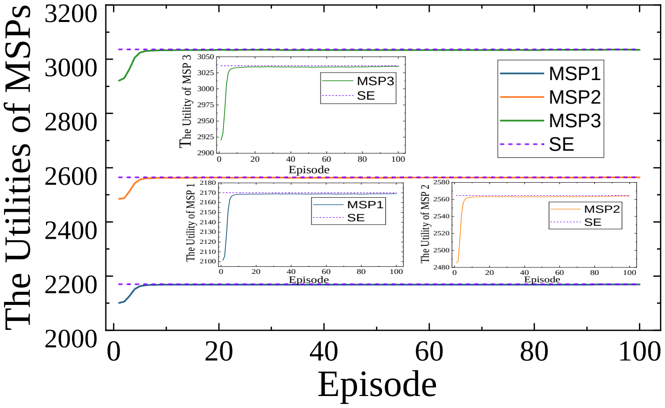


Fig. 7. The utilities of MSPs.

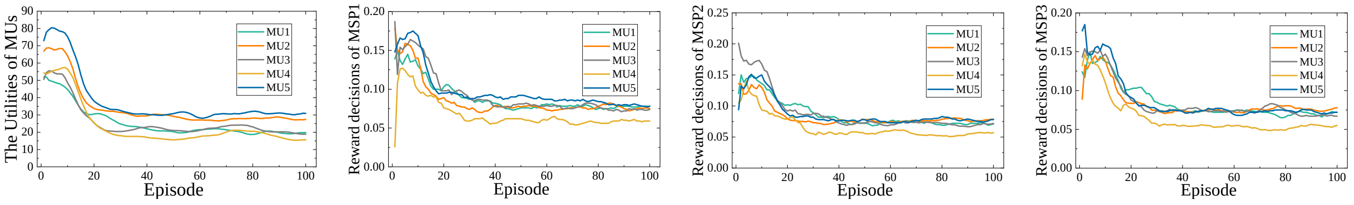
$s_n(k) = [V_{1n}(k), V_{2n}(k), \ldots, V_{mn}(k)]$, and the state of the environment is $\mathcal{S}(k) = [s_1(k), \ldots, s_n(k)]$. At the $k$-th stage game, MSP $n$ observes a state $s_n(k)$ and determines an action $\boldsymbol{p}_n(k)$ within $[p_{min}, p_{max}]$. When an action $\boldsymbol{p}_n(k)$ is applied to state $s_n(k)$, the agent $n$ receives a utility $\mathcal{U}_n(k)$ from the environment. Taking into account the competition among agents, we define the reward $\mathcal{U}_n$ for each MSP by (12). The state transition probability $P(s_n(k+1) \mid s_n(k), \boldsymbol{p}_n(k))$ leads to the new state $s_n(k+1)$ after executing an action $\boldsymbol{p}_n(k)$ at the state $s_n(k)$.

### B. The Multi-agent DRL-based Dynamic Reward (MDDR)

In our work, each MSP agent operates the DRL-based dynamic reward algorithm in a fully distributed manner. **Algorithm 1** shows the pseudocode for model trading using MDDR under a dynamic environment (Lines 4-17) and the guidance for the FL process (Lines 19-25).

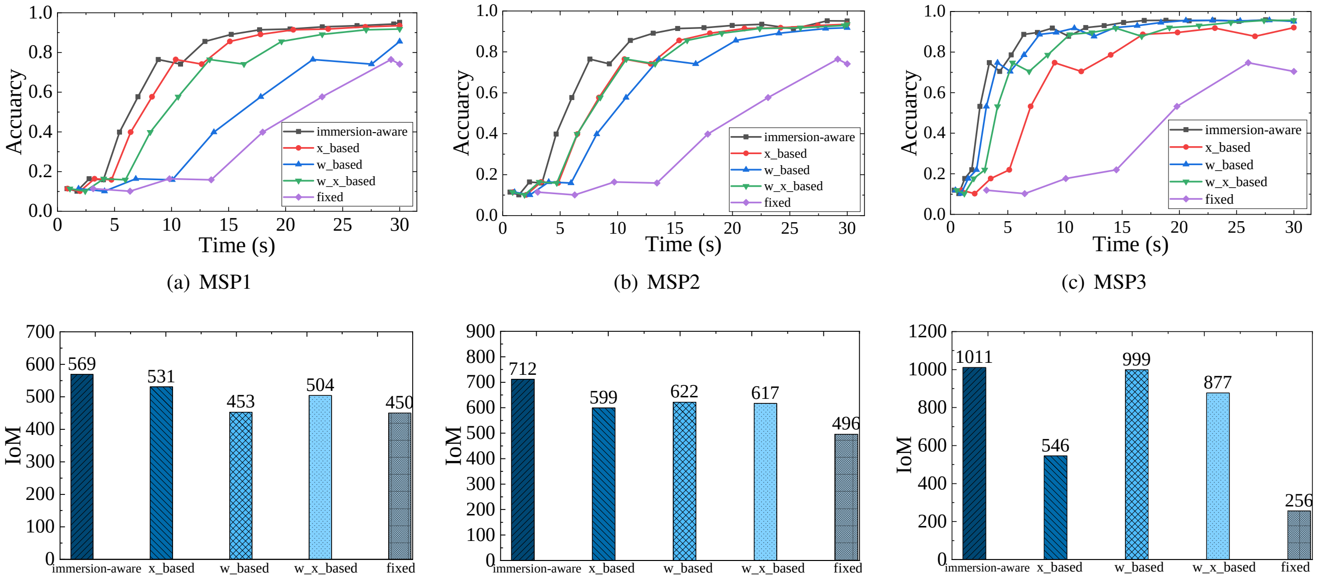
**Trading process:** At the start of the game, each MSP $n$ initializes its actor $\boldsymbol{\alpha}_n$ and critic network $\boldsymbol{\beta}_n$, and episode buffer $\mathcal{D}_n$ for each MSP agent $n$ (Line 2). Each agent $n$ feeds the observed information into the policy network $\boldsymbol{\alpha}_n$ to derive its reward policy (Lines 7-8). Subsequently, each MSP agent $n$ broadcasts its reward decisions, interacts with the environment, receives feedback from the MUs, calculates its utility, and stores the experience in the episode buffer $\mathcal{D}_n$ (Lines 9-12). The experience is collected through this interactive process (Lines 7-12) until the buffer $\mathcal{D}_n$ is full. Once the buffer is full, the critic network $\boldsymbol{\beta}_n$ evaluates the actor network $\boldsymbol{\alpha}_n$ based on the collected experience and updates it accordingly (Line 14). Finally, the episode buffer is cleared in preparation for the next episode (Line 15). The game will terminate when the maximum number of training episodes is reached and then the trading results will be obtained.

**FL training guided by trading results:** After obtaining the trading results (MSPs' reward decisions $\boldsymbol{p}$ and resource allocation $(\boldsymbol{f}, \boldsymbol{B})$), MUs and MSPs enter the FL phase. MUs follow the resource allocation scheme for local training and model uploading (Line 21). Then, the MSP is responsible for aggregating and updating the global model (Line 23). Finally, the updated model is provided for use by the AR engine (Line 24). This process iterates multiple times until the total FL time $T$ is reached. During real-time interactions, the MSP's AR



(a) The utilities of MUs (b) The reward decisions of MSP 1 (c) The reward decisions of MSP 2 (d) The reward decisions of MSP 3

Fig. 8. The convergence processes of MDDR under dynamic networks.



(a) MSP1 (b) MSP2 (c) MSP3

(d) The IoM for MSP 1 (e) The IoM for MSP 2 (f) The IoM for MSP 3

Fig. 9. Evaluation results for MNIST.

TABLE II
THE UTILITIES OF THREE REWARD APPROACHES

| | MSP 1 | MSP 2 | MSP 3 | Total |
|---|---|---|---|---|
| Optimal | 2169.99 | 2564.56 | 3035.96 | 7770.51 |
| MDDR | 2169.11 | 2563.95 | 3034.94 | 7768.00 |
| MAPPO | 2169.23 | 2563.96 | 3035.26 | 7768.45 |

service immersion and quality can be enhanced by utilizing the updated model.

### C. The Complexity and Convergence of MDDR

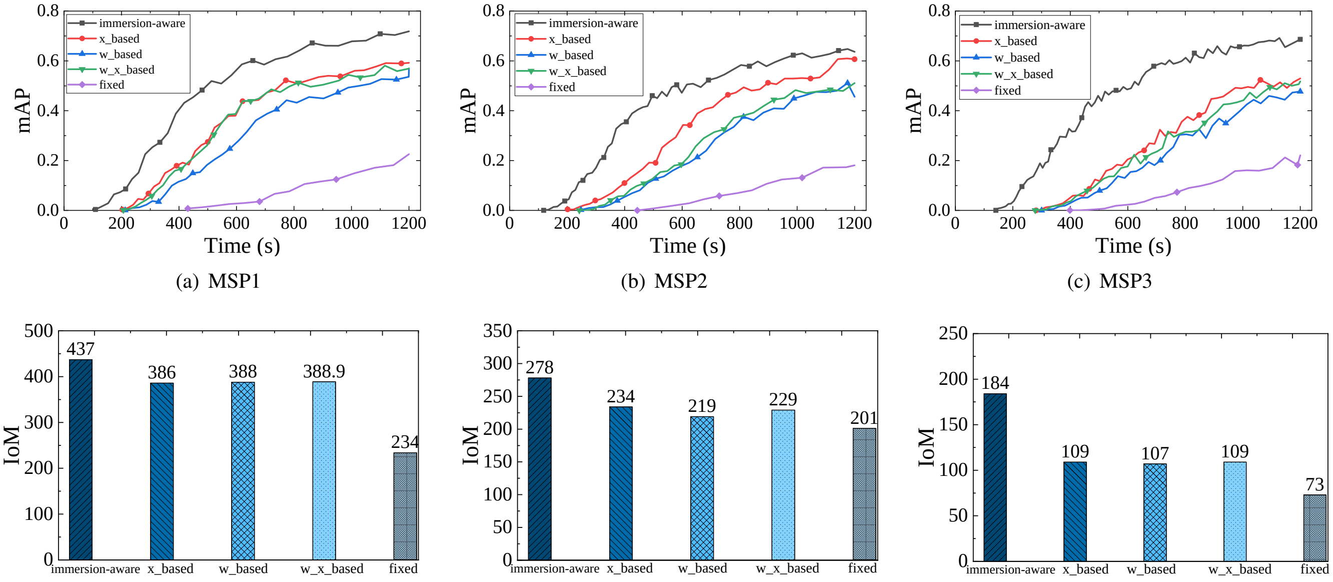
The complexity of MDDR is mainly influenced by the structure of the actor-critic network. According to [39], the total computational complexity of the fully connected layer can be expressed as the number of multiplications $\mathcal{O}(\sum_{u=1}^{U} \varrho_u \cdot \varrho_{u-1})$, where $\varrho_u$ is the number of neurons in the $u$-th hidden layer. In this work, each agent has its actor-critic network, structured as $[5, 256, 10]$ and $[5, 256, 1]$, respectively. Therefore, its complexity can be expressed as $\mathcal{O}(5 \times 256 + 256 \times 10 + 5 \times 256 + 256 \times 1)$ without burden on the MSP. After several attempts, the MDDR can achieve convergence [40] (as per Fig. 7 and Table II) with a suitable set of hyper-parameters.

## VII. PERFORMANCE EVALUATION

First, we verify the near-optimal performance of MDDR. Then, to evaluate the effectiveness of the proposed framework, we compare its performance with benchmark schemes in object detection and classification for AR services through FL.

### A. Simulation Settings

We conduct experiments on the MNIST dataset with ResNet-18 and the GTSRB dataset with Faster R-CNN. The GTSRB was recorded during daytime driving on different types of roads in Germany for vehicle AR scenario testing [41]. We set the number of MUs as $M = 5$ and the number of MSPs as $N = 3$. The configurations of the maximum computational and communication resource for each MU are randomly generated from the range $[3, 5]$GHz and $[1, 4]$MB [42]. For local training of MUs, we adopt the SGD optimizer with momentum $= 0.9$ and learning rate $= 0.001$. For model aggregation of MSPs, we employ the Federated Averaging (FedAvg) algorithm, applying equal weights to the updates from all participants. The trading guidance time $T$ for MNIST and GTSRB are 30s and $1,200$s, respectively. All baseline methods use the same settings as ours.



(a) MSP1 (b) MSP2 (c) MSP3

(d) The IoM for MSP 1 (e) The IoM for MSP 2 (f) The IoM for MSP 3

Fig. 10. Evaluation results for GTSRB.

TABLE III
TIME COMPARISON OF FIVE SCHEMES FOR ACHIEVING THE SAME PERFORMANCE.

| Schemes | MNIST | | | GTSRB | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Training Time to reach 90% Accuracy (s) | | | Training Time to reach 50% mAP (s) | | | Training Time to reach 70% mAP@50 (s) | | |
| | MSP 1 | MSP 2 | MSP 3 | MSP 1 | MSP 2 | MSP 3 | MSP 1 | MSP 2 | MSP 3 |
| immersion-aware | 15.23 | 13.09 | 7.59 | 513.16 | 581.09 | 623.89 | 418.28 | 447.69 | 451.27 |
| x_based | 17.79 | 18.03 | 19.80 | 812.53 | 898.59 | 1061.50 | 596.57 | 679.91 | 890.47 |
| w_based | 30+ | 24.04 | 10.28 | 1064.83 | 1175.58 | 1200+ | 774.09 | 953.50 | 1013.89 |
| w_x_based | 23.20 | 19.08 | 12.34 | 901.61 | 1190 | 1194 | 596.35 | 848.68 | 932.73 |
| fixed | 30+ | 30+ | 30+ | 1200+ | 1200+ | 1200+ | 1200+ | 1200+ | 1200+ |

Note: 30+ or 1200+ indicates that the scheme required more than 30 seconds or 1200 seconds, respectively, to achieve the target accuracy, exceeding the guideline time for the trading results.

### B. Benchmark Schemes

To our knowledge, there are no comparable solutions to the model trading problem in the vehicular metaverse. Therefore, under the uniform rewards of MSPs, we develop and extend four benchmark schemes for multi-MSP and multi-MU scenarios while meeting FL deadlines and resource constraints of devices.

- ***x_based***: Inspired by the incentive mechanism of the Stackelberg game for federated learning [31], the MU allocates computational and communication resources based on the amount of data involved in local training.
- ***w_based***: The MU adjusts the resource allocation according to the inference loss [43], i.e., the potential value of local data in our work.
- ***w_x_based***: Combining the ***x_based*** and ***w_based*** designs, the MU allocates resources considering both the data size and the potential value of data.
- ***fixed***: Taking into account the differences in performance of the various devices that exist in reality, we randomly allocate a fixed set of computational and communication resources to reflect the heterogeneity of devices.

### C. Performance Evaluation of MDDR

First, we consider a relatively static communication environment (fixed channel conditions) to show the effectiveness of MDDR compared to the optimal solution and the general multi-agent depth approximation policy optimization (MAPPO) approach. We trained our proposed MDDR and MAPPO until convergence, and the results are shown in Fig. 7 and Table II. We find that both MDDR and MAPPO achieve good results, but MAPPO slightly outperforms MDDR. This is because MAPPO is a centrally controlled distributed execution [44] that can utilize more information for decision making. However, this approach is not suitable for non-cooperation due to the need to share parameters among MSPs. For the fully distributed MDDR, it is more suitable for non-cooperative scenarios. The reason is that in MDDR, each agent has a self-contained actor-critic network, and information is not required to be shared. This suggests that the approach can achieve considerable utility while preserving privacy and saving communication costs significantly. Then, we deploy the MDDR in dynamic communication environments where channel conditions change over time. The behaviors of each MU and MSP in the trading are shown in Figs. 8(a), 8(b), 8(c) and 8(d), reflecting that the MDDR can converge rapidly and show good stability in dynamic network environments.

### D. Benchmark Comparison

Figs. 9(a), 9(b), and 9(c) show the performance improvement in classification accuracy over time, where the ***immersion-aware*** outperforms the other schemes. This is because ***immersion-aware*** utilizes IoM to incentivize MUs to allocate computational and communication resources for providing learning models to MSPs. The IoM metric not only evaluates the freshness and accuracy of the model but also considers the amount and potential value of raw data used for training. As a result, resource-limited MUs are motivated to contribute more resources, providing valuable learning models to MSPs at an accelerated pace. In contrast, baseline schemes like ***x_based***, ***w_based*** and ***w_x_based*** struggle to offer adequate incentives for MUs to contribute their available resources due to uniform rewards. For example, if a given MU has already reached the maximum resource it can contribute when the reward is $p$, then even if $p$ is increased further, the MSP's gains will be impaired rather than increased. Consequently, the uniform reward $p$ will not be increased, leading to a situation where other MUs with available resources will not contribute more resources for providing higher-value learning models.

In addition, Figs. 9(a), 9(b), and 9(c) reveal that MSP 3 performs better than MSP 1 and MSP 2. For example, at the 10th second, the accuracy of MSP 3 is about 88%, which is 14% and 3% higher than those of MSP 1 and MSP 2, respectively. This is because the potential value $\omega$ is higher for MSP 3, leading to more rewards being given by MSP 3. As a result, the MUs will allocate more computational and communication resources to provide learning models for MSP 3 than for MSP 1 and MSP 2. For the three different MSPs, some fluctuations are normal for model training (e.g., at time 10s). It can be seen that the effect of ***fixed*** is the worst due to synchronous FL, which is limited by the MU with the least resource. Meanwhile, it is clear from Figs. 9(d), 9(e) and 9(f) that the effectiveness of the five schemes is influenced by IoM. With the higher IoM, the performance is improved faster, resulting in a better immersive experience. Yet, the IoM of ***w_based*** in Fig. 9(e) is higher than ***x_based*** and ***w_x_based*** but the performance improvement rate is not as fast as them. The reason is that the IoM here represents the sum of all IoM from MUs, whereas the rate of performance improvement is hampered by the worst MU. Additionally,from Figs. 9(b) and 9(c), it can be observed that both ***w_x_based*** and ***w_based*** perform better when compared to Fig. 9(a). The main reason is the higher potential value $\omega$ associated with MSP 2 and MSP 3.

For object detection on GTSRB, Figs. 10(a), 10(b) and 10(c) show the mean average precision (mAP) and Table III gives the mAP@50 (The value of mAP when the intersection over union (IOU) threshold is larger than 0.5) results. We observe that compared with MNIST, the performance enhancement for object detection is slower due to the complexity of the detection model. Specifically, it takes at least 513.16 seconds to achieve a 50% mAP using the ***immersion-aware*** method. Our research introduces the flexibility to adjust decision guidance time, making it adaptable to various task types. Meanwhile, Figs. 10(d), 10(e), 10(f) and Table III demonstrate that MSP 1 with greater IoM takes less time to achieve the same mAP, which indicates that IoM works well to capture immersion. For instance, MSP 1 has the highest IoM, and the time that it takes to reach 50% mAP is 513.16s, which is 67.93s and 110.73s less than those of MSP 2 and MSP 3, respectively.

The comparison results validate the advantages of the proposed trading framework, which outperforms other schemes by improving IoM by 38.3% and 37.2% and reducing the training time to reach the target accuracy by 43.5% and 49.8%, on average, for the MNIST and GTSRB datasets, respectively.

## VIII. Conclusion

To facilitate the development of vehicular metaverse services, we propose an immersion-aware model trading framework that incorporates FL to incentivize MUs to contribute locally trained data for AR services such as object detection and classification. Specifically, we construct an EPEC problem with MSPs as leaders and MUs as followers to achieve an equilibrium among their interests. A new metric called "IoM" is designed to comprehensively evaluate the enhancement brought by the learning models of MUs for AR services. In addition, considering the competitive relationship among MSPs and the dynamic network environment, we develop a fully distributed MDDR approach to obtain the reward decisions of MSPs. Extensive simulations on AR-related vehicle and MNIST datasets demonstrate that the proposed framework enables more efficient and immersive AR services in the vehicular metaverse.

## Appendix A
## Proof of Proposition 1

*Proof.* First, we transform **Problem 1** into an unconstrained optimization problem using the Lagrangian dual method [45], [46]. The Lagrangian function associated with MU $m$ is formulated as

$$\begin{aligned}
&L_m\left(\boldsymbol{f}_m, \boldsymbol{B}_m, \boldsymbol{p}_m, \lambda_m, \beta_m, \delta_{mn}, \gamma_m\right) = \sum_n p_{mn} V_{mn} \\
&- \sum_n \left[c_m^f f_{mn} \log(1/\theta_m) + c_m^B B_{mn}\right] \\
&- \lambda_m\Big(\sum_n f_{mn} - f_m^{max}\Big) - \beta_m\Big(\sum_n B_{mn} - B_m^{max}\Big) \\
&- \sum_n \delta_{mn}\Big[\log(1/\theta_m)\frac{x_m \tau_n}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})} - \tau_n\Big] \\
&- \gamma_m\Big[S_m - T^{req}\Big(f_m^{max} - \sum_n f_{mn}\Big)\Big],
\end{aligned} \tag{24}$$

where $\lambda_m \geq 0, \beta_m \geq 0$, $\delta_{mn} \geq 0$ and $\gamma_m \geq 0$ are the Lagrangian dual variables corresponding to constraints C1, C2 and C3, respectively. Then, the first-order derivative of $L_m$ with respect to $f_{mn}$ and $B_{mn}$ can be derived as

$$\begin{aligned}
\frac{\partial L_m}{\partial f_{mn}} &= p_{mn} I_{mn} \frac{x_m \tau_n \log(1/\theta_m)}{f_{mn}^2} - c_m^f \log(1/\theta_m) \\
&- \lambda_m + \delta_{mn} \log(1/\theta_m) \frac{x_m \tau_n}{f_{mn}^2} - \gamma_m T^{req},
\end{aligned} \tag{25}$$

and

$$\begin{aligned}\frac{\partial L_m}{\partial B_{mn}} &= p_{mn} I_{mn} \frac{b_{mn}}{B_{mn}^2 \log_2(1+\varsigma_{mn})} - c_m^B - \beta_m \\ &\quad + \delta_{mn} \frac{b_{mn}}{B_{mn}^2 \log_2(1+\varsigma_{mn})}.\end{aligned} \tag{26}$$

From (25) and the constraints in (11), the Karush-Kuhn-Tucker (KKT) [47] conditions are given by

$$\begin{aligned}&\frac{\partial L_m}{\partial f_{mn}} = 0,\ \lambda_m(\sum_n f_{mn} - f_m^{max}) = 0, \\ &\delta_{mn}[\log(1/\theta_m)\frac{x_m \tau_n}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})} - \tau_n] = 0, \\ &\gamma_m[S_m - T^{req}(f_m^{max} - \sum_n f_{mn})] = 0, \\ &\lambda_m, \delta_{mn}, \gamma_m \geq 0, C1, C2, C3.\end{aligned} \tag{27}$$

Since $S_m < T^{req}(f_m^{max} - \sum_n f_{mn})$ and $\sum_n f_{mn} < f_m^{max}$, we can obtain $\lambda_m = 0$ and $\gamma_m = 0$. Furthermore, $\frac{\partial L_m}{\partial f_{mn}} = 0$ can be converted to

$$(p_{mn} I_{mn} + \delta_{mn}) \frac{x_m \tau_n}{f_{mn}^2} - c_m^f = 0. \tag{28}$$

Similarly, the KKT conditions for $B_{mn}$ are obtained as

$$\begin{aligned}&\frac{\partial L_m}{\partial B_{mn}} = 0,\ \beta_m(\sum_n B_{mn} - B_m^{max}) = 0, \\ &\delta_{mn}[\log(1/\theta_m)\frac{x_m \tau_n}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})} - \tau_n] = 0, \\ &\beta_m, \delta_{mn} \geq 0, C1, C2.\end{aligned} \tag{29}$$

Since $\sum_n B_{mn} < B_m^{max}$, we can obtain $\beta_m = 0$. Furthermore, $\frac{\partial L_m}{\partial B_{mn}} = 0$ can be converted to

$$(p_{mn} I_{mn} + \delta_{mn}) \frac{b_{mn}}{B_{mn}^2 \log_2(1+\varsigma_{mn})} - c_m^B = 0. \tag{30}$$

Then, based on the value of $\delta_{mn}$, the optimal allocation of computation and communication resources exists in the following two cases:

- ($\delta_{mn} = 0$): According to (28) and (30), we have
$$f_{mn} = \sqrt{\frac{p_{mn} I_{mn} x_m \tau_n}{c_m^f}}, B_{mn} = \sqrt{\frac{p_{mn} I_{mn} b_{mn}}{c_m^B \log_2(1+\varsigma_{mn})}}. \tag{31}$$
- ($\delta_{mn} > 0$): Substituting $f_{mn}$ and $B_{mn}$ into $\delta_{mn}\Big[\log(1/\theta_m)\frac{x_m\tau_n}{f_{mn}} + \frac{b_{mn}}{B_{mn}\log_2(1+\varsigma_{mn})} - \tau_n\Big] = 0$, we can get
$$\begin{aligned}&\delta_{mn} = F^2 - p_{mn} I_{mn}, \\ &F = \log(1/\theta_m)\sqrt{\frac{x_m c_m^f}{\tau_n}} + \frac{\sqrt{b_{mn} c_m^B}}{\tau_n\sqrt{\log_2(1+\varsigma_{mn})}}.\end{aligned} \tag{32}$$
Next, bring $\delta_{mn}$ into $f_{mn}$ and $B_{mn}$, we have
$$f_{mn} = F\sqrt{\frac{x_m \tau_n}{c_m^f}}, B_{mn} = F\sqrt{\frac{b_{mn}}{c_m^B \log_2(1+\varsigma_{mn})}}. \tag{33}$$

□

## Appendix B
## Proof of $\frac{\partial^2 \Psi_n}{\partial p_{mn}^2} < 0$ and $\frac{\partial^2 \Psi_n}{\partial p_{mn} \partial p_{m'n}} < 0$

*Proof.* First, we need to compute $\frac{\partial f_{mn}}{\partial p_{mn}}$, $\frac{\partial^2 f_{mn}}{\partial p_{mn}^2}$, $\frac{\partial B_{mn}}{\partial p_{mn}}$, and $\frac{\partial^2 B_{mn}}{\partial p_{mn}^2}$ to obtain the values of $V'_{mn}$ and $V''_{mn}$. From (28), we can obtain $\frac{\partial f_{mn}}{\partial p_{mn}}$ and $\frac{\partial^2 f_{mn}}{\partial p_{mn}^2}$, the steps of which are shown as follows. The first-order derivative of $\frac{\partial L_m}{\partial f_{mn}} = 0$ with respect to $p_{mn}$ is expressed as $I_{mn}\frac{x_m\tau_n}{f_{mn}^2} - 2(p_{mn}I_{mn} + \delta_{mn})\frac{x_m\tau_n}{f_{mn}^3} \cdot \frac{\partial f_{mn}}{\partial p_{mn}} = 0$. Consequently, we can obtain

$$\frac{\partial f_{mn}}{\partial p_{mn}} = \frac{I_{mn} f_{mn}}{2(p_{mn} I_{mn} + \delta_{mn})} > 0. \tag{34}$$

The second-order derivative of $\frac{\partial L_m}{\partial f_{mn}} = 0$ with respect to $p_{mn}$ is expressed as

$$\frac{\partial^2 f_{mn}}{\partial p_{mn}^2} = \frac{I_{mn}\frac{\partial f_{mn}}{\partial p_{mn}}(p_{mn} I_{mn} + \delta_{mn}) - I_{mn}^2 f_{mn}}{2(p_{mn} I_{mn} + \delta_{mn})^2}, \tag{35}$$

by substituting (34) into (35), we have

$$\frac{\partial^2 f_{mn}}{\partial p_{mn}^2} = \frac{-I_{mn}^2 f_{mn}}{4(p_{mn} I_{mn} + \delta_{mn})^2} < 0. \tag{36}$$

Likewise, we derive $\frac{\partial B_{mn}}{\partial p_{mn}}$ and $\frac{\partial^2 B_{mn}}{\partial p_{mn}^2}$ with the following steps based on (30). The first-order derivative of $\frac{\partial L_m}{\partial B_{mn}} = 0$ with respect to $p_{mn}$ is expressed as $I_{mn}\frac{b_{mn}}{B_{mn}^2\log_2(1+\varsigma_{mn})} - 2(p_{mn}I_{mn} + \delta_{mn})\frac{b_{mn}}{B_{mn}^3\log_2(1+\varsigma_{mn})} \cdot \frac{\partial B_{mn}}{\partial p_{mn}} = 0$. Accordingly, we have

$$\frac{\partial B_{mn}}{\partial p_{mn}} = \frac{I_{mn} B_{mn}}{2(p_{mn} I_{mn} + \delta_{mn})} > 0. \tag{37}$$

Similarly, we obtain the second-order derivative of $\frac{\partial L_m}{\partial B_{mn}} = 0$ with respect to $p_{mn}$ as

$$\frac{\partial^2 B_{mn}}{\partial p_{mn}^2} = \frac{I_{mn}\frac{\partial B_{mn}}{\partial p_{mn}}(p_{mn} I_{mn} + \delta_{mn}) - I_{mn}^2 B_{mn}}{2(p_{mn} I_{mn} + \delta_{mn})^2}, \tag{38}$$

by substituting (37) into (38), we have

$$\frac{\partial^2 B_{mn}}{\partial p_{mn}^2} = \frac{-I_{mn}^2 B_{mn}}{4(p_{mn} I_{mn} + \delta_{mn})^2} < 0. \tag{39}$$

Based on (34), (36), (37), and (39), we can easily infer that $V'_{mn} > 0$ and $V''_{mn} < 0$, as shown in (40). Then, we can obtain $\frac{\partial^2 \Psi_n}{\partial p_{mn} \partial p_{m'n}} < 0$. For $\frac{\partial^2 \Psi_n}{\partial p_{mn}^2}$, We substitute the specific expressions for $V'_{mn}$ and $V''_{mn}$ into (22) and obtain the following result as shown in (41). Next, we randomly choose a vector $\boldsymbol{h} \in \mathbb{R}^{M\times 1}$ with elements not all 0. For $\boldsymbol{\Lambda}_n$, it is easy to see that $\boldsymbol{h}^T\boldsymbol{\Lambda}_n\boldsymbol{h} = \sum_m \frac{\partial^2 \Psi_n}{\partial p_{mn}^2}(h^m)^2 < 0$. Furthermore, $\boldsymbol{h}^T\boldsymbol{H}_n\boldsymbol{h} = \frac{-\mu_n}{(1+\sum_m V_{mn})^2}\left(\sum_m h^m V'_{mn}\right)^2 < 0$. Thus, we have $\boldsymbol{h}^T(\boldsymbol{\Lambda}_n + \boldsymbol{H}_n)\boldsymbol{h} = \boldsymbol{h}^T\boldsymbol{\Lambda}_n\boldsymbol{h} + \boldsymbol{h}^T\boldsymbol{H}_n\boldsymbol{h} < 0$, indicating that the utility function $\Psi_n$ is strictly concave. So far, the uniqueness of the Nash equilibrium [37] solution to the multi-MSP rewarding game $\Omega$ has been proved. □

$$
\begin{aligned}
V'_{mn} = \frac{\partial V_{mn}}{\partial p_{mn}} &= I_{mn}\left(\frac{x_m \tau_n \log(1/\theta_m)}{f_{mn}^2}\cdot\frac{\partial f_{mn}}{\partial p_{mn}} + \frac{b_{mn}}{B_{mn}^2 \log_2(1+\varsigma_{mn})}\cdot\frac{\partial B_{mn}}{\partial p_{mn}}\right) > 0,\\
V''_{mn} = \frac{\partial^2 V_{mn}}{\partial p_{mn}^2} &= I_{mn}\left[\frac{-2x_m \tau_n \log(1/\theta_m)}{f_{mn}^3}\cdot(\frac{\partial f_{mn}}{\partial p_{mn}})^2 + \frac{x_m \tau_n \log(1/\theta_m)}{f_{mn}^2}\cdot\frac{\partial^2 f_{mn}}{\partial p_{mn}^2}\right.\\
&\left.-\frac{2b_{mn}}{B_{mn}^3 \log_2(1+\varsigma_{mn})}\cdot(\frac{\partial B_{mn}}{\partial p_{mn}})^2 + \frac{b_{mn}}{B_{mn}^2 \log_2(1+\varsigma_{mn})}\cdot\frac{\partial^2 B_{mn}}{\partial p_{mn}^2}\right] < 0.
\end{aligned}
\tag{40}
$$

$$
\begin{aligned}
\frac{\partial^2 \Psi_n}{\partial p_{mn}^2} = -&\left[\frac{4a_1 I_{mn}^2 \delta_{mn} + a_1 I_{mn}^3 p_{mn}}{4f_{mn}(\delta_{mn} + I_{mn}p_{mn})^2} + \frac{4b_{mn} I_{mn}^2 \delta_{mn} + b_{mn} I_{mn}^3 p_{mn}}{4a_2 B_{mn}(\delta_{mn} + I_{mn}p_{mn})^2} + I_{mn}^2 \frac{\left(\frac{a_1 a_2 I_{mn} B_{mn} + b_{mn} I_{mn} f_{mn}}{2a_2 f_{mn} B_{mn}(\delta_{mn} + I_{mn} p_{mn})}\right)^2}{(1+\sum_m V_{mn})^2}\right.\\
&\left.+\frac{3I_{mn}^3(a_1 a_2 B_{mn} + b_{mn} f_{mn})}{4a_2 f_{mn} B_{mn}(\delta_{mn} + I_{mn}p_{mn})^2(1+\sum_m V_{mn})}\right] < 0, \text{where } a_1 = x_m \tau_n \log(1/\theta_m), a_2 = \log_2(1+\varsigma_{mn}).
\end{aligned}
\tag{41}
$$

## References

[1] Y. Wang *et al.*, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 1, pp. 319–352, 1st Quart., 2023.

[2] M. Deveci, D. Pamucar, I. Gokasar, M. Köppen, and B. B. Gupta, "Personal mobility in metaverse with autonomous vehicles using Q-rung orthopair fuzzy sets based OPA-RAFSI model," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15 642–15 651, Dec. 2023.

[3] M. Deveci, D. Pamucar, I. Gokasar, M. Köppen, B. B. Gupta, and T. Daim, "Evaluation of metaverse traffic safety implementations using fuzzy Einstein based logarithmic methodology of additive weights and TOPSIS method," *Technol. Forecasting Social Change*, vol. 194, Sep. 2023, Art. no. 122681.

[4] MarketsandMarkets, "Metaverse Market for Automotive by Products (Software, Hardware), Technology (Virtual Reality (VR), Augmented Reality (AR), Mixed Reality (MR), Non-fungible Token (NFT), Blockchain), Function, Application & Region - Global Forecast to 2030," Aug. 2022. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/automotive-metaverse-market-10437470.html

[5] D. S. Sarwatt, Y. Lin, J. Ding, Y. Sun, and H. Ning, "Metaverse for intelligent transportation systems (ITS): A comprehensive review of technologies, applications, implications, challenges and future directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 6290–6308, Jul. 2024.

[6] D. Qiao, L. Qian, S. Guo, J. Zhao, and P. Zhou, "AMFL: Resource-efficient adaptive metaverse-based federated learning for the human-centric augmented reality applications," *IEEE Trans. Neural Netw. Learn. Syst.*, early access. doi: 10.1109/TNNLS.2024.3409446.

[7] Nissan Motor Corporation. "Invisible-to-Visible (I2V)". https://www.nissan-global.com/EN/INNOVATION/TECHNOLOGY/ARCHIVE/I2V/ (accessed Apr. 20, 2025).

[8] I. Gokasar, D. Pamucar, M. Deveci, B. B. Gupta, L. Martinez, and O. Castillo, "Metaverse integration alternatives of connected autonomous vehicles with self-powered sensors using fuzzy decision making model," *Inf. Sci.*, vol. 642, Sep. 2023, Art. no. 119192.

[9] F. Chiariotti, "Age of information analysis for a shared edge computing server," *IEEE Trans. Commun.*, vol. 72, no. 12, pp. 7826–7841, Dec. 2024.

[10] L. Feng *et al.*, "Resource allocation for metaverse experience optimization: A multi-objective multi-agent evolutionary reinforcement learning approach," *IEEE Trans. Mobile Comput.*, vol. 24, no. 4, pp. 3473–3488, Apr. 2025.

[11] D. Chen *et al.*, "Federated learning based mobile edge computing for augmented reality applications," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, 2020, pp. 767–773.

[12] C. Zhou, C. Liu, and J. Zhao, "Resource allocation of federated learning for the metaverse with mobile augmented reality," *IEEE Trans. Wireless Commun.*, early access. doi: 10.1109/TWC.2023.3326884.

[13] B. Hazarika, K. Singh, O. A. Dobre, C.-P. Li, and T. Q. Duong, "Quantum-enhanced federated learning for metaverse-empowered vehicular networks," *IEEE Trans. Commun.*, early access. doi: 10.1109/TCOMM.2024.3502667.

[14] J. Kang *et al.*, "Blockchain-empowered federated learning for healthcare metaverses: User-centric incentive mechanism with optimal data freshness," *IEEE Trans. Cogn. Commun. Netw.*, vol. 10, no. 1, pp. 348–362, Feb. 2024.

[15] E. Baccour, A. Erbad, A. Mohamed, M. Hamdi, and M. Guizani, "A blockchain-based reliable federated meta-learning for metaverse: A dual game framework," *IEEE Internet Things J.*, vol. 11, no. 12, pp. 22 697–22 715, Apr. 2024.

[16] X. Li *et al.*, "Satisfaction-aware incentive scheme for federated learning in industrial metaverse: DRL-based Stackbelberg game approach," Feb. 2025, *arXiv: 2502.06909*. [Online]. Available: https://arxiv.org/abs/2502.06909

[17] Q. Zhang, Z. Xiong, J. Zhu, S. Gao, and W. Yang, "A privacy-preserving auction mechanism for learning model as an NFT in blockchain-driven metaverse," *ACM Trans. Multimedia Comput., Commun. Appl.*, vol. 20, no. 7, Mar. 2024.

[18] N. Raveendran, H. Zhang, L. Song, L.-C. Wang, C. S. Hong, and Z. Han, "Pricing and resource allocation optimization for IoT fog computing and NFV: An EPEC and matching based perspective," *IEEE Trans. Mobile Comput.*, vol. 21, no. 4, pp. 1349–1361, Sep. 2020.

[19] J. Feng and J. Zhao, "Resource allocation for augmented reality empowered vehicular edge metaverse," *IEEE Trans. Commun.*, vol. 73, no. 3, pp. 1987–2001, Mar. 2025.

[20] L. U. Khan, M. Alghfeli, M. Guizani, N. Saeed, and S. Muhaidat, "TCS: A joint task offloading, communication, and sensing framework for vehicular metaverse," Sep. 2024. [Online]. Available: https://www.techrxiv.org/users/692774/articles/1224428

[21] J. Chen *et al.*, "Multi-agent deep reinforcement learning for dynamic avatar migration in AIoT-enabled vehicular metaverses with trajectory prediction," *IEEE Internet Things J.*, vol. 11, no. 1, pp. 70–83, Jan. 2024.

[22] J. Kang *et al.*, "Blockchain-based pseudonym management for vehicle twin migrations in vehicular edge metaverse," *IEEE Internet Things J.*, vol. 11, no. 21, pp. 34 254–34 269, Nov. 2024.

[23] T. J. Chua, W. Yu, and J. Zhao, "Mobile edge adversarial detection for digital twinning to the metaverse: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, early access. doi: 10.1109/TWC.2023.3298265.

[24] Y. Qiu *et al.*, "Spotlighter: Backup age-guaranteed immersive virtual vehicle service provisioning in edge-enabled vehicular metaverse," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 13 375–13 391, Dec. 2024.

[25] Z. Zhang, F. Zeng, and F. Tang, "Vehicle-assisted data sensing in vehicle edge metaverse: A game theory approach," *IEEE Trans. Veh. Technol.*, vol. 74, no. 4, pp. 5664–5675, Apr. 2025.

[26] X. Lin, J. Wu, J. Li, W. Yang, and M. Guizani, "Stochastic digital-twin service demand with edge response: An incentive-based congestion control approach," *IEEE Trans. Mobile Comput.*, vol. 22, no. 4, pp. 2402–2416, Apr. 2023.

[27] J. Xu *et al.*, "Semantic-aware UAV swarm coordination in the metaverse: A reputation-based incentive mechanism," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 13 821–13 833, Dec. 2024.

[28] I. Lotfi, M. Qaraqe, A. Ghrayeb, and D. Niyato, "VMGuard: Reputation-based incentive mechanism for poisoning attack detection in ve-

hicular metaverse," *IEEE Trans. Veh. Technol.*, early access. doi: 10.1109/TVT.2025.3543650.

[29] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, and C.-Z. Xu, "FedDC: Federated learning with non-IID data via local drift decoupling and correction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 10 112–10 121.

[30] K. Wang, Z. Ding, D. K. C. So, and Z. Ding, "Age-of-information minimization in federated learning based networks with non-IID dataset," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8939–8953, Aug. 2024.

[31] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A learning-based incentive mechanism for federated learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6360–6368, Jan. 2020.

[32] S. Jiang and J. Wu, "A reward response game in the federated learning system," in *Proc. IEEE 18th Int. Conf. Mobile Ad Hoc Smart Syst. (MASS)*, 2021, pp. 127–135.

[33] X. Lin, J. Wu, J. Li, X. Zheng, and G. Li, "Friend-as-learner: Socially-driven trustworthy and efficient wireless federated edge learning," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 269–283, Jan. 2021.

[34] F. Li, P. Rao, W. Sun, Y. Su, and X. Chen, "A low signal-to-noise ratio infrared small-target detection network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 18, pp. 8643–8658, Mar. 2025.

[35] Y. Zhan, C. H. Liu, Y. Zhao, J. Zhang, and J. Tang, "Free market of multi-leader multi-follower mobile crowdsensing: An incentive mechanism design by deep reinforcement learning," *IEEE Trans. Mobile Comput.*, vol. 19, no. 10, pp. 2316–2329, Oct. 2020.

[36] W. Y. B. Lim *et al.*, "When information freshness meets service latency in federated learning: A task-aware incentive scheme for smart industries," *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 457–466, Jan. 2022.

[37] T. Kim, C. K. Kim, S.-s. Lee, and S. Lee, "Incentive-aware partitioning and offloading scheme for inference services in edge computing," *IEEE Trans. Serv. Comput.*, vol. 17, no. 4, pp. 1580–1592, Aug. 2024.

[38] J. Ji, K. Zhu, and L. Cai, "Trajectory and communication design for cache- enabled UAVs in cellular networks: A deep reinforcement learning approach," *IEEE Trans. Mobile Comput.*, vol. 22, no. 10, pp. 6190–6204, Oct. 2023.

[39] S. E. Li, "Deep reinforcement learning," in *Reinforcement Learning for Sequential Decision and Optimal Control*. Singapore: Springer, 2023, pp. 365–402.

[40] M. Samir, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Online altitude control and scheduling policy for minimizing AoI in UAV-assisted IoT wireless networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 7, pp. 2493–2505, Jul. 2022.

[41] D. Toshniwal, S. Loya, A. Khot, and Y. Marda, "Optimized detection and classification on GTRSB: Advancing traffic sign recognition with convolutional neural networks," Mar. 2024, *arXiv: 2403.08283*. [Online]. Available: https://arxiv.org/abs/2403.08283

[42] S. Jošilo and G. Dán, "Wireless and computing resource allocation for selfish computation offloading in edge computing," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2019, pp. 2467–2475.

[43] H. Zeng, T. Zhou, Y. Guo, Z. Cai, and F. Liu, "FedCav: Contribution-aware model aggregation on distributed heterogeneous data in federated learning," in *Proc. 50th Int. Conf. Parallel Process.*, 2021, pp. 1–10.

[44] Q. Han, X. Wang, and W. Shen, "Communication-dependent computing resource management for concurrent task orchestration in IoT systems," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 14 297–14 312, Dec. 2024.

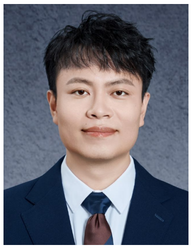
[45] R. T. Rockafellar, "Lagrange multipliers and optimality," *SIAM Rev.*, vol. 35, no. 2, pp. 183–238, 1993.

[46] Z. He, Y. Guo, X. Zhai, M. Zhao, W. Zhou, and K. Li, "Joint computation offloading and resource allocation in mobile-edge cloud computing: A two-layer game approach," *IEEE Trans. Cloud Comput.*, vol. 13, no. 1, pp. 411–428, Mar. 2025.

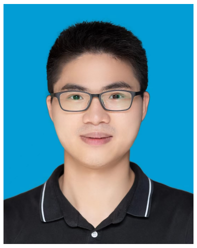
[47] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

**Hongjia Wu** received the Ph.D. degree in computer science from the National University of Defense Technology, China, in 2023. She was a visiting student at the Singapore University of Technology and Design. She is currently a Post-doctoral Fellow at the Education University of Hong Kong in China. Her research interests mainly focus on computational offloading, game theory, dispersed computing and internet of things.

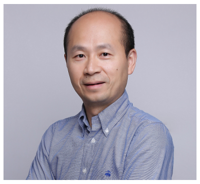
**Hui Zeng** received his B.S. degree from Sun Yat-sen University in 2020 and M.S. degree from National University of Defense Technology in 2022. He is currently a Ph.D. student at the National University of Defense Technology. His research interests include data privacy, federated learning, and distributed systems.



**Zehui Xiong** is currently an Assistant Professor at Singapore University of Technology and Design, and also an Honorary Adjunct Senior Research Scientist with Alibaba-NTU Singapore Joint Research Institute, Singapore. He received the PhD degree in Nanyang Technological University (NTU), Singapore. He was the visiting scholar at Princeton University and University of Waterloo. His research interests include wireless communications, Internet of Things, blockchain, edge intelligence, and Metaverse.



**Jiawen Kang** received the M.S. degree and the Ph.D. degree from the Guangdong University of Technology, China, in 2015 and 2018. He is currently a full professor at the Guangdong University of Technology. He has been a postdoc at Nanyang Technological University from 2018 to 2021, Singapore. His research interests mainly focus on blockchain, Metaverse, and AIGC in wireless communications and networking.



**Zhiping Cai** received the B.Eng., M.A.Sc., and Ph.D. degrees in computer science and technology from National University of Defense Technology (NUDT), China, in 1996, 2002, and 2005, respectively. He is a full professor in College of Computer Science and Technology, NUDT. His current research interests include artificial intelligence, network security and big data. He is a distinguished member of the CCF.

**Tse-Tin Chan** (Member, IEEE) received his B.Eng. (First Class Hons.) and Ph.D. degrees in Information Engineering from The Chinese University of Hong Kong (CUHK), Hong Kong SAR, China, in 2014 and 2020, respectively.

He is currently an Assistant Professor with the Department of Mathematics and Information Technology, The Education University of Hong Kong (EdUHK), Hong Kong SAR, China. Prior to this, he was an Assistant Professor with the Department of Computer Science, The Hang Seng University of Hong Kong (HSUHK), Hong Kong SAR, China, from 2020 to 2022. His research interests include wireless communications and networking, Internet of Things (IoT), age of information (AoI), and AI in wireless communications.

**Dusit Niyato** (Fellow, IEEE) is a professor in the School of Computer Science and Engineering, at Nanyang Technological University, Singapore. He received B.Eng. from King Mongkuts Institute of Technology Ladkrabang (KMITL), Thailand in 1999 and Ph.D. in Electrical and Computer Engineering from the University of Manitoba, Canada in 2008. His research interests are in the areas of sustainability, edge intelligence, decentralized machine learning, and incentive mechanism design.

**Zhu Han (S'01-M'04-SM'09-F'14)** received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. 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 at the University of Maryland. From 2006 to 2008, he was an assistant professor at 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 at the University of Houston, Texas. Dr. Han's 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. Dr. Han 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. Dr. Han was an IEEE Communications Society Distinguished Lecturer from 2015-2018, AAAS fellow since 2019, and ACM distinguished Member since 2019. Dr. Han is a 1% highly cited researcher since 2017 according to Web of Science. Dr. Han 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."