Online Task Allocation Based on Lyapunov Optimization and Fuzzy Control System in Vehicular Mobile Crowdsensing

Wei You, Tao Peng, *Member, IEEE,* Zhidong Xie, Guojun Wang, *Member, IEEE,* Weizhi Meng, *Senior Member, IEEE,* Qin Liu, *Member, IEEE*, Entao Luo, *Member, IEEE*

***Abstract*—As a powerful and promising paradigm, vehicular mobile crowdsensing (VMCS) recruits sensing devices on vehicles to collect sensory data, offering advantages such as low cost, wide coverage, and high data quality. Task allocation plays a crucial role in determining the efficiency and performance of vehicular mobile crowdsensing systems. However, existing task allocation methods are mostly designed for offline, single-task crowdsensing scenarios and rarely consider privacy protection, which cannot meet the dynamic and complex requirements of real-world applications. Therefore, we propose an online task allocation and privacy preservation scheme called Lya-FCGA (Lyapunov optimization with Fuzzy Controller-based Genetic Algorithm). To maintain long-term system stability, we utilize Lyapunov optimization to transform long-term objectives into per-time-slot queue control problems. Additionally, we introduce a fuzzy control system to adaptively adjust the genetic algorithm parameters, enhancing the accuracy and efficiency of task allocation. Moreover, differential privacy technology is employed to protect workers' privacy and increase their participation willingness. Extensive experiments on real datasets demonstrate that our scheme reduces sensing costs while ensuring long-term system stability and effectively protecting privacy, showcasing great advantages and potential.**

***Index Terms*—Vehicular mobile crowdsensing, task allocation, privacy preservation, Lyapunov optimization, fuzzy control system.**

I. Introduction

WITH the proliferation of vehicles equipped with advanced sensors and communication technologies, vehicular mobile crowdsensing (VMCS) has evolved to utilize the extensive network of vehicles to collect and process data. As dynamic sensing units, vehicles are increasingly integrated into the Internet of Vehicles (IoV) [1], [2], providing enhanced capabilities. Vehicular mobile crowdsensing capitalizes on these connected vehicles to offer several benefits, including improved data accuracy through diverse sensing perspectives, enhanced coverage due to the mobility of vehicles, and reduced data collection costs compared to traditional methods. Consequently, it plays a crucial role in fields such as environmental monitoring [3], [4], intelligent cities [5]–[7], healthcare [8]–[10], and public safety [11], [12].

The VMCS typically consists of the platform, data requesters, and workers. Data requesters send task requirements to the platform, and workers interested in tasks provide their information. The platform executes the task allocation algorithm, selecting suitable workers for each task. Then, workers proceed to the corresponding location to execute the task and upload sensing data to the platform. Finally, the platform processes the collected data and returns results to data requesters. In VMCS, the workers are vehicles equipped with sensing devices. As dynamic and mobile sensing nodes, these vehicles offer several distinct advantages. They can collect data from a wide range of locations as they move, providing comprehensive and diverse data coverage. Additionally, their ability to continuously gather real-time information contributes to more accurate and timely data collection. Vehicles also enhance the system's overall efficiency by leveraging their inherent mobility to cover larger areas and integrate seamlessly into existing transportation networks. Consequently, VMCS is particularly effective for real-time traffic monitoring, road condition assessment, and environmental sensing applications.

In such a large-scale, portable, distributed data collection and processing approach, the efficiency and performance of the VMCS systems are significantly influenced by task allocation. A practical task allocation scheme becomes a powerful means to control resource allocation and save sensing costs. A well-designed task allocation scheme can reduce workers' burden, enhance data quality, and thereby improve the performance of the VMCS systems. For instance, allocating tasks to distant workers may require higher payments to compensate for their travel costs, affecting workers' participation and diminishing data quality. Conversely, allocating tasks to nearby workers without considering their processing capabilities may lead to

Corresponding authors: Tao Peng; Entao Luo.
Wei You is with the School of Computer Science and Cyber Engineering, Guangzhou University. E-mail: iuvyouwei@e.gzhu.edu.cn.
Tao Peng is with the School of Computer Science and Cyber Engineering, Guangzhou University. E-mail: pengtao@gzhu.edu.cn.
Zhidong Xie is with the School of Computer Science and Cyber Engineering, Guangzhou University. E-mail: xiezhidong0413@e.gzhu.edu.cn.
Guojun Wang is with the School of Computer Science and Cyber Engineering, Guangzhou University. E-mail: csgjwang@gzhu.edu.cn.
Weizhi Meng is with the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Denmark. E-mail: weme@dtu.dk.
Qin Liu is with the College of Computer Science and Electronic Engineering, Hunan University. E-mail: gracelq628@hnu.edu.cn.
Entao Luo is with the School of Information Engineering, Hunan University of Science and Technology. E-mail: luoentaohuse@163.com.

task backlog, system blockage, and resource wastage.

According to the dynamic and real-time requirements of the applications, VMCS can be categorized into offline and online scenarios. Online VMCS considers the spatio-temporal variability of the sensing environment, as well as the diversity and dynamics of sensing targets, making it more practical. Furthermore, the task allocation process requires collecting a vast amount of worker information. Sensing tasks in VMCS are usually location-dependent, and the sensing data uploaded to the platform often contain information such as time, location, and trajectory, which may involve personal privacy, such as behavioral habits, home addresses, and interests [14]. Therefore, addressing privacy leakage in task allocation is imperative [15], [16]. Currently, several methods are commonly employed to prevent privacy leakage, including obfuscation-based approaches [17]–[19], cryptographic techniques [20]–[23], differential privacy methods, as well as blockchain [26]–[28], federated learning [29], [30], and other decentralized architectures [24]. In recent years, significant progress has been made in research on task allocation, while most existing methods have encountered the following limitations. **1) Limited Adaptability.** Many methods focus on offline, single-task scenarios, and proposed models fail to accurately reflect the demands of more general and typical practical applications, thus incapable of real-time updates. **2) Poor Stability.** Although dynamic factors such as changes in task requirements and worker status are sometimes considered, the need for system stability is often ignored, resulting in unbalanced task allocation and inefficient resource utilization, ultimately affecting the long-term operation of the system. **3) Privacy Concerns.** Most task allocation solutions do not sufficiently protect the sensitive data uploaded by workers, posing risks of privacy leakage and data exploitation, which may affect workers' participation and damage the platform's interests.

To address the above issues, we simulate online scenarios by considering dynamic task releases, workers' real-time states, and random movements, and we proposed a scheme called Lya-FCGA (Lyapunov optimization with Fuzzy Controller-based Genetic Algorithm). Specifically, Lya-FCGA addresses the three main limitations as follows. First, to enhance adaptability, a fuzzy control system is introduced to dynamically adjust genetic algorithm parameters, allowing real-time adaptation to environmental changes and task demands. Second, to improve system stability, we model the task allocation process as a queue control problem based on Lyapunov optimization to decouple the long-term optimization problem into short-term decision subproblems in each time slot, ensuring long-term balanced task assignment and efficient resource usage. Third, to address privacy concerns, differential privacy techniques are incorporated to protect workers' sensitive location information while maintaining high utility. The proposed online task allocation method is feasible, and it can effectively and dynamically allocate tasks, guarantee queue stability, and ensure the long-term stability of the VMCS system. The main contributions are as follows.

- We capture the spatio-temporal characteristics of VMCS, establish an online model focusing on multi-task allocation scenarios, and design a scheme named Lya-FCGA to solve the online task allocation problem.
- We consider privacy leakage issues in task allocation, employing differential privacy to protect sensitive worker information. We comprehensively consider workers' limited processing capabilities and design allocation schemes, effectively ensuring the long-term stable operations of the VMCS system.
- We conduct extensive experiments based on real-world dataset CRAWDAD [65], and the results demonstrate that Lya-FCGA ensures the stability of the VMCS system and effectively reduces the sensing cost incurred by workers traveling to corresponding task points compared to other schemes.

The rest of the paper is organized as follows. Section II introduces some related works. Section III presents the problem formulation. In Section IV, we focus on privacy preservation in online MCS and derive Lyapunov optimization. Section V presents a detailed demonstration of the fuzzy genetic algorithm for task allocation. Section VI shows some experimental results. Finally, Section VII concludes the paper.

II. Related Work

Task allocation can be categorized into offline allocation and online allocation based on different allocation modes. At present, extensive research has been conducted on task allocation in both offline and online MCS.

Offline task allocation refers to pre-allocating tasks based on prior information and historical data before the tasks begin. For offline task allocation, Wang et al. [33] took into account the maximum load capacity of users and the availability of sensing devices, transformed the multi-task allocation problem into a bipartite graph, and optimized the task allocation scheme using an iterative greedy method. Wang et al. [34] used a semi-Markov chain method to represent the mobility model of users, predicted the trajectories of them and constructed a task allocation model with expenditure minimization as the optimization objective. Li et al. [35] proposed a heuristic multi-task scheduling algorithm based on an ant colony algorithm to allocate tasks, aiming to maximize workers' benefits. Cheung et al. [36] divided tasks into urgent and normal based on their time sensitivity, considering factors such as participants' reliability and experience and using distributed methods for task allocation. Han et al. [37] proposed a distributed multi-task allocation method. By clustering tasks and user regions based on a greedy algorithm, users are selected for task allocation based on historical reputation records. Wang et al. [38] treated task-user adaptability prediction as a binary classification problem and integrated different factors into one framework using a Logit model to predict the matching rate of each task-user pair, recruiting users with the highest matching rate to perform tasks. Zhao et al. [39] considered factors such as the spatiotemporal distribution of tasks and users, user location preferences, and capabilities. They applied a spatiotemporal attention mechanism to capture complex relationships among multiple factors, thereby enhancing the precision of task allocation. Liu et al. [40] investigated a UAV-assisted task allocation method that jointly assigned tasks to both human participants and drones. This method employed deep reinforcement learning to predict user trajectories and allocated tasks based on participants' limited budgets, aiming to balance tolerance, budget, and coverage. In privacy-preserving offline task allocation, various techniques have been proposed to ensure user privacy. Jiang et al. [41] used proxy re-encryption, designed symmetric key generators to protect location information, and used reinforcement learning for task allocation to achieve high accuracy and efficiency. Gao et al. [42] achieved personalized location privacy protection through order-preserving encryption and hash trees and solved the user selection problem under budget constraints using dynamic programming. Li et al. [43] designed a flexible privacy model for multi-task assignment by introducing information entropy to dynamically quantify users' privacy preferences and proposed a privacy budget allocation method that flexibly assigns personalized privacy budgets to improve data utility. Wu et al. [44] proposed a blockchain-supported verifiable privacy-preserving task allocation scheme for crowdsensing, extending symmetric hidden vector encryption to support client-side expression matching without leaking privacy. Li et al. [45] presented a decentralized task matching system based on consortium blockchain, utilizing a multi-authority keyword-searchable attribute encryption scheme to realize bilateral privacy protection. In our previous work [46], we proposed a privacy-preserving multi-objective task assignment scheme using differential privacy and the non-dominated sorting genetic algorithm II to protect participants' location and bid privacy while minimizing the travel distance and the expenditure cost.

Online task allocation is more general and practical. In online MCS, workers and tasks are dynamic, and related spatiotemporal information is difficult to obtain in advance, making it difficult for traditional offline allocation schemes to achieve optimal real-time allocation. A few studies have proposed solutions to the online task allocation problem. Peng et al. [47] studied AP-assisted task assignment in MCS, proposing two algorithms to reduce the average or worst-case makespan of tasks, effectively reducing task completion time. Lin et al. [48] predicted the data quality of users completing new tasks based on historical data and designed offline and online user recruitment algorithms based on greedy strategies to improve data quality. Liu et al. [49] proposed a dynamic user recruitment strategy with truthful pricing to address the online recruitment problem in mobile crowdsensing under budget and time constraints. Aiming to improve task assignment accuracy and result quality, Shi et al. [50] proposed a Bayesian probabilistic model named Gaussian Latent Topic Model (GLTM) to estimate fine-grained worker reliability for latent topics in numerical tasks, along with a truth inference algorithm and an online task assignment mechanism. To et al. [51] introduced geo-indistinguishability to perturb task and worker locations, quantified reachability probabilities, and developed task assignment strategies balancing task completion, worker travel, and system overhead. To addresses dynamic crowdsensing systems with spatio-temporal worker mobility and stochastic task arrivals, Wang et al. [52] proposed online control policies using Lyapunov optimization for fairness and system stability. Tao et al. [53] investigated privacy protection in spatial crowdsourcing for online task assignments, focusing on minimizing total distance. It proposed a novel privacy mechanism based on Hierarchically Well-Separated Trees (HSTs), proven to be ϵ-Geo-Indistinguishable. Wang et al. [54] used probabilistic graphical models to unify task types and proposed unsupervised learning for dynamic truth inference and worker expertise. It introduced online task allocation schemes based on probability improvement and entropy reduction, demonstrating superior performance in experiments compared to existing methods. Tao et al. [55] considered the location dependency and time sensitivity of tasks, as well as users' resource constraints in terms of maximum movement distance. They modeled the task allocation problem as a path planning problem with time windows and used double deep Q-learning (DDQN) to solve it. Wang et al. [56] proposed a Planar Laplace distribution-based Privacy mechanism (PLP) to obfuscate worker and task locations. Additionally, it introduced a Threshold-based Online task Assignment mechanism (TOA) to handle efficient assignments. Zhang et al. [57] proposed the Bilateral Online Priority Reassignment (BOPR) algorithm. They designed a two-stage allocation strategy to complete all matches while maximizing the benefits for both users and service requesters. Zhang et al. [58] designed an opportunistic user recruitment strategy based on deep learning using an LSTM model to predict user locations, and combined with a best-worst solution distance algorithm for user evaluation and dynamic recruitment, thereby optimizing task allocation. Li et al. [59] also employed the LSTM neural network model in their framework, proposing a two-stage user recruitment framework based on trajectory prediction. They first integrated LSTM with Geohash to detect an opportunistic user set, followed by an online task assignment algorithm based on geographic locations, although they did not consider dynamic task arrivals. Chen et al. [60] introduced Paillier encryption technology, designing an online task allocation model that optimizes task quality while achieving bidirectional privacy protection of sensing data. To address privacy-preserving online multi-task assignments, Zhang et al. [61] introduced a scheme that employs secure path planning to optimize routing without additional noise, achieving significant reductions in moving distances compared to existing obfuscation-based approaches. Ning et al. [66] constructed a blockchain-enabled crowdsensing framework for distributed traffic management, selected active miners and transactions, and maximized social welfare using a Deep Reinforcement Learning (DRL)-based algorithm and a Distributed Alternating Direction Method of Multipliers (DIADEM) algorithm. Liu et al. [67] considered the use of unmanned vehicular workers in spatial crowdsourcing and proposed a deep reinforcement learning approach for worker scheduling to achieve an optimal trade-off between maximizing the collected amount of data and coverage fairness, and minimizing the overall energy consumption of workers.

Existing methods have made progress in both offline and online task allocation. However, there are still some limitations in current research. For instance, some schemes focus primarily on single-task allocation, while offline methods struggle to adapt to dynamic environments. Although some online methods offer real-time performance, they fail to consider the limited execution capabilities of workers and the stability of the system. Moreover, most schemes do not effectively protect the sensing data, increasing the risk of privacy leakage, which reduces user participation and affects the platform's overall performance. To address these issues, we propose a dynamic multi-task allocation scheme with privacy protection, combining Lyapunov optimization theory with the collaborative optimization capabilities of fuzzy control systems and genetic algorithms, aiming to enhance system stability, and data security, and improve the efficiency and fairness of task allocation, thus supporting the sustainable development of mobile crowdsensing systems.

III. Problem Formulation

This paper focuses on the multi-task allocation problem in vehicular mobile crowdsensing. Let $U = \{u_1, u_2, \ldots, u_i, \ldots, u_N\}$ represent the set of N workers, where each worker is a vehicle equipped with sensing devices and u_i represents the i-th worker. $W = \{w_1, w_2, \ldots, w_j, \ldots, w_M\}$ denotes the task set, where w_j represents the j-th task, and there are M tasks. These tasks have K types, denoted as $\theta = \{\tau_1, \tau_2, \ldots, \tau_k, \ldots, \tau_K\}$, and w_j^k is used to indicate that w_j belongs to type k. The VMCS system runs in a discrete equal-length time $t \in \{0, 1, 2, \ldots\}$, the types and quantities of tasks arriving in each time slot are assumed to be independent and identically distributed random variables. Let $u(t)$ represent the sensing cost in time slot t, and we use the Euclidean distance between workers and tasks as the sensing cost required for workers to go to the corresponding locations. For ease of expression, the main notations involved in this paper are listed in TABLE I.

TABLE I
Main Notations

Notation	Description
M, N, K	Number of tasks, workers and task types
w_j, W	The j-th task, task set
u_i, U	The i-th worker, worker set
τ_k, Θ	The k-th task type, task type set
t	Time slot index, $t \in \{0, 1, 2, \ldots\}$
$a^k(t)$	Admitted tasks of the k-th type in time slot t
$A^k(t)$	Arrival tasks of the k-th type in time slot t
$c_{ij}(t)$	Distance between w_j and u_i
$Q_k(t)$	Sensing task queue
$P_i(t)$	Worker u_i's capacity deficit queue
$h_i(t)$	Worker u_i's capacity in time slot t
$u(t)$	Sensing cost in time slot t
$\bar{u}$	Time average sensing cost
$L(Z(t)), \Delta(Z(t))$	Lyapunov function, Lyapunov drift
V	Control parameter

As illustrated in Fig. 1, in online VMCS, workers constantly move within the area, and they randomly come online and go offline, while tasks dynamically arrive over time. Specifically, as time elapses, worker 2 and worker 5 go offline, while worker 7 and worker 8 come online. Meanwhile, worker 1, worker 3, worker 4, and worker 6 move randomly over time.

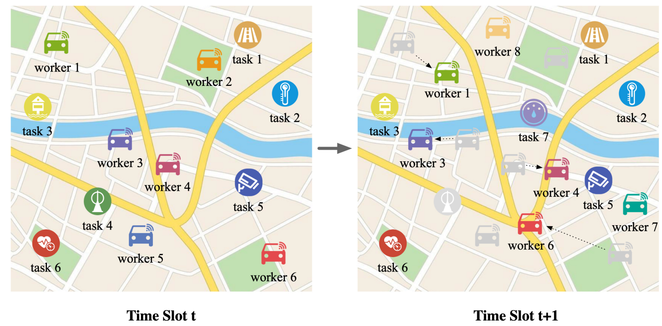


Fig. 1. Online vehicular mobile crowdsensing.

Task 7 is published on the platform, and task 4 is removed from the platform.

We consider that the VMCS system operates within discrete time slots $t \in \{0, 1, 2, \ldots\}$, which vary from a few minutes to several hours according to sensing requirements. Let $A^k(t)$ represent the number of tasks of type k arriving at the platform in time slot t, and $A^k(t)$ is an independently and identically distributed random variable. The platform needs to make allocation decisions in each time slot, where $x_{ij}^k[t]$ indicates whether w_j^k is assigned to worker u_i in time slot t. After allocation in time slot t, the number of tasks of type k assigned in t is denoted by $r^k(t)$, where $r^k(t) = \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij}^k[t]$. In time slot t, the sensing cost for workers to travel to execute tasks can be represented in Eq. (1).

$$u(t) = \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{k=1}^{K} x_{ij}^k[t] \cdot c_{ij}(t) \tag{1}$$

Thus, the time average sensing cost is defined in Eq. (2).

$$\bar{u} = \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} E\{u(t)\} \tag{2}$$

Eq. (3) is the constraint we define. Since the mobile devices equipped by workers have limited storage, computing, and processing capabilities, the ability of worker u_i to handle task $h_i(t)$ in each time slot is limited. There exists a threshold H_i, the number of tasks allocated to u_i in each time slot $o_i(t)$ cannot exceed H_i.

$$o_i(t) \le H_i, \quad \forall u_i \in U \tag{3}$$

The goal that we need to optimize is to minimize the sensing cost incurred by all workers while subject to the constraint of worker processing capacity. Formally, the online task allocation of the sensing cost minimization problem is formulated in Eq. (4).

$$\begin{aligned} \min: \quad & \bar{u} \\ \text{s.t.} \quad & x_{ij}^k[t] \in \{0, 1\}, \quad \forall u_i \in U, \forall w_j \in W, \forall t \\ & 0 \le o_i(t) \le H_i, \quad \forall u_i \in U \end{aligned} \tag{4}$$

Let $Q_k(t)$ represent the task processing queue in time slot t, where $A^k(t)$ is the number of tasks of type k arriving at the platform, $a^k(t)$ is the number of tasks allowed to enter the platform in time slot t, which also represents the enqueue rate

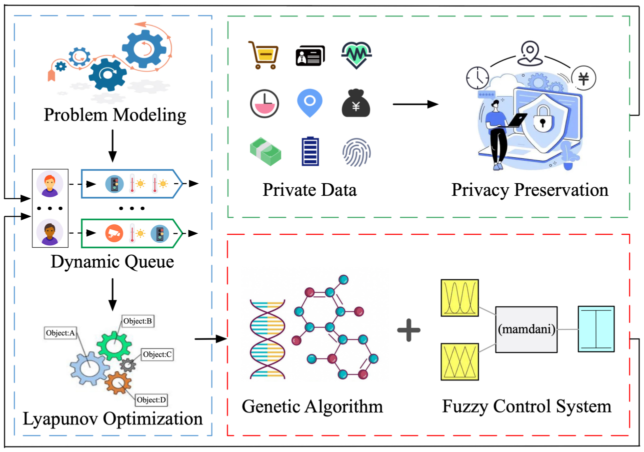


Fig. 2. The framework of our scheme.

of $Q_k(t)$. $r^k(t)$ is the number of tasks of type k assigned to workers in time slot t, which is also the dequeue rate of $Q_k(t)$.

IV. Lyapunov Optimization-Based Online Control with Privacy Preservation

As shown in Fig. 2, the framework of the proposed scheme consists of three modules: differential privacy preservation, Lyapunov optimization decoupling, and fuzzy genetic algorithm-based allocation.

The differential privacy preservation module employs the differential privacy technique to add Laplace-distributed noise to the location uploaded by workers. Task allocation is based on the obfuscated locations, thereby preventing the leakage of workers' locations. The Lyapunov optimization decoupling module constructs a dynamic queue system. It uses Lyapunov optimization to decouple long-term objectives from short-term decisions, transforming the issue of long-term system stability into a queue control problem in each time slot. The fuzzy genetic algorithm-based allocation module designs a fuzzy control system to optimize the genetic algorithm, enabling adaptive adjustments of the crossover and mutation rates during the algorithm's execution. In this section, we provide details about privacy preservation and Lyapunov optimization.

A. Local Differential Privacy Preservation

Uploading location information risks privacy leakage, which may undermine the willingness of workers to participate. Therefore, it is necessary to protect worker location information to prevent attackers from obtaining and exploiting workers' locations. We employ local differential privacy to protect the locations of workers, ensuring that the platform receives only perturbed location information. Specifically, the location of each worker can be represented by coordinates consisting of longitude and latitude. Since coordinates are numerical, the Laplace mechanism is taken into consideration. It adds noise to the longitude and latitude components, achieving the goal of location preservation. The Laplace mechanism adds random noise satisfying the Laplace distribution to the actual results, making the output results of the query indistinguishable between any two datasets D and D'. Eq. (5) is the probability density function of Laplace distribution. The added noise is a random variable following the $Laplace(b)$ distribution with mean 0 and variance $2b^2$.

$$f(x) = \frac{1}{2b} e^{-\left(\frac{|x|}{b}\right)} \tag{5}$$

where $b = \Delta f/\epsilon$ is the noise scale factor, Δf is sensitivity, and ϵ is the privacy budget. The privacy budget ϵ controls the strength of privacy protection. A smaller ϵ implies stronger privacy guarantees, as it results in larger noise. Conversely, a larger ϵ provides weaker protection but better utility. The sensitivity Δf refers to the maximum possible difference in the output when a single record in the dataset changes. These parameters define the scale of added noise and determine the trade-off between privacy and data accuracy.

Assuming the coordinates of u_i are represented by (x_i, y_i), where x_i and y_i represent the longitude and latitude, respectively. Firstly, determine the sensitivity and privacy budget, then generate noise satisfying the Laplace distribution, and add it to the actual coordinate to obtain the perturbed locations in Eq. (6).

$$L_i' = (x_i + \eta_1, y_i + \eta_2), \eta_1, \eta_2 \sim \text{Laplace}(0, \frac{\Delta f}{\epsilon}) \tag{6}$$

The mechanism satisfies ϵ-differential privacy if, for any two neighboring inputs $L = (x, y)$ and $L' = (x', y')$, and for any measurable set S, $\Pr[M(L) \in S] \leq e^{\epsilon} \cdot \Pr[M(L') \in S]$ holds.

Proof. The mechanism adds independent Laplace noise with scale $b = \frac{\Delta f}{\epsilon}$ to each coordinate. For any output point (x_o, y_o), the joint probability density is calculated from Eq. (7).

$$\begin{aligned}\Pr[M(L) = (x_o, y_o)] &= \frac{1}{4b^2} \exp\left(-\frac{|x_o - x| + |y_o - y|}{b}\right)\\ \Pr[M(L') = (x_o, y_o)] &= \frac{1}{4b^2} \exp\left(-\frac{|x_o - x'| + |y_o - y'|}{b}\right)\end{aligned} \tag{7}$$

The ratio of these probabilities can be computed from Eq. (8).

$$\begin{aligned}&\frac{\Pr[M(L) = (x_o, y_o)]}{\Pr[M(L') = (x_o, y_o)]}\\ &= \exp\left(\frac{1}{b}\left(|x_o - x'| + |y_o - y'| - |x_o - x| - |y_o - y|\right)\right)\end{aligned} \tag{8}$$

By applying the triangle inequality $|a| - |b| \leq |a - b|$, we obtain Eq. (9).

$$|x_o - x'| - |x_o - x| \leq |x - x'|, |y_o - y'| - |y_o - y| \leq |y - y'| \tag{9}$$

Therefore, the total ratio is bounded by Eq. (10).

$$\exp\left(\frac{|x - x'| + |y - y'|}{b}\right) \leq \exp\left(\frac{\Delta f}{\Delta f/\epsilon}\right) = \exp(\epsilon) \tag{10}$$

Hence, the mechanism satisfies ϵ-differential privacy.

□

After obtaining the perturbed locations of workers and the locations of tasks, the sensing platform needs to allocate tasks to appropriate workers to minimize the total travel distance of all workers so that the sensing cost incurred tends to be minimal. We use Lyapunov optimization and fuzzy genetic algorithm to solve the online task assignment problem, which will be introduced in detail in the following section.

B. Lyapunov Optimization Decoupling

1) Dynamic Queue: In time slot t, $a^k(t)$ new tasks are allowed to enter the sensing task queue $Q_k(t)$, and $r^k(t)$ tasks are completed. Eq. (11) is the dynamic evolvement of the sensing task queue $Q_k(t)$.

$$Q_k(t+1) = \max[Q_k(t) - r^k(t), 0] + a^k(t) \tag{11}$$

It is important to note that the number of tasks entering the queue in each time slot cannot exceed the number of tasks arriving at the platform during that slot, i.e., $a^k(t) \leq A^k(t)$. Assuming that each type's maximum number of tasks arriving at the platform across all time slots is A_{max}, we have $A^k(t) \leq A_{max}$. According to the stability principle of queues, if the sensing task queue $Q_k(t)$ satisfies Eq. (12), it remains stable.

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[Q_k(t)] < \infty \tag{12}$$

Additionally, each worker has a processing capacity threshold H_i, and the number of tasks allocated to each worker in each time slot cannot exceed this threshold, i.e., $o_i(t) \leq H_i$, where $H_i \leq H_{max}$. To meet this constraint, we introduce a worker capacity deficit queue $P_i(t)$ to describe the remaining processing capacity of workers. $P_i(t)$ dynamically changes according to the rule in Eq. (13).

$$P_i(t+1) = \max[P_i(t) - H_i(t), 0] + o_i(t) \tag{13}$$

where $o_i(t) = \sum_{j=1}^{M} \sum_{k=1}^{K} x_{ij}^k[t]$. Similarly, the worker capacity deficit queue $P_i(t)$ remains stable if it satisfies Eq. (14).

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[P_i(t)] < \infty \tag{14}$$

2) Lyapunov Optimization: The Lyapunov function is defined in Eq. (15). To ensure the long-term stability of VMCS, all queues need to remain stable. We denote $Z(t) = (Q(t); P(t))$ as the total queue backlog, including all the sensing task queues and worker capacity deficit queues.

$$L(t) = L(Z(t)) = \frac{1}{2} \sum_{k=1}^{K} Q_k(t)^2 + \frac{1}{2} \sum_{i=1}^{N} P_i(t)^2 \tag{15}$$

The Lyapunov function $L(Z(t))$ represents the total queue backlog in time slot t. The backlog of the sensing task queue indicates the number of unprocessed tasks over time. A large backlog suggests that many tasks remain unassigned, which may lower the overall task completion rate and affect system efficiency. Similarly, a high backlog in the worker capability deficit queue reflects a mismatch between task demands and worker capabilities, leading to inefficient resource utilization and potential wastage. Therefore, maintaining queue backlogs at a manageable level is essential for improving task completion rate, enhancing resource efficiency, and ensuring timely task execution. These factors can significantly impact the overall efficiency of the system. To maintain system stability, we need to control the stability of the queue. In mathematics, derivatives are often used to analyze the stability of functions. Thus, we take the derivative of the Lyapunov function to obtain the Lyapunov drift in Eq. (16).

$$\Delta(Z(t)) = E\{L(Z(t+1)) - L(Z(t)) \mid Z(t)\} \tag{16}$$

The Lyapunov drift represents the change in the Lyapunov function in slotted time. Minimizing this drift in every time slot will reduce the queue backlog, ensuring queue stability. By adding the objective function as a penalty term to the Lyapunov drift, we obtain the Lyapunov drift-plus-penalty function. Our goal is to minimize the sensing cost, thus the Lyapunov drift-plus-penalty function can be represented by Eq. (17).

$$\Delta(Z(t)) + V \cdot E\{u(t) \mid Z(t)\} \tag{17}$$

where V is a non-negative control parameter used to balance system stability and sensing cost minimization. A larger V indicates a lower sensing cost but might impact system stability. The smaller the drift-plus-penalty function, the more stable the system will be, and the lower the sensing cost is.

Theorem 1. *In each time slot t, the Lyapunov drift-plus-penalty function satisfies the inequality given in Eq. (18).*

$$\begin{aligned}
&\Delta\left(Z\left(t\right)\right) + V \cdot E\left\{u\left(t\right)\middle|\, Z\left(t\right)\right\} \leq D + \\
&E\Big\{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}\Big[\frac{1}{2}(x_{ij}^k[t])^2 + P_i(t)\cdot x_{ij}^k[t] - Q_k(t)\cdot(x_{ij}^k[t])^2 + \\
&V \cdot x_{ij}^k[t] \cdot c_{ij}^k(t)\Big] | Z(t)\Big\} + E\Big\{\sum_{k=1}^{k}[Q_k(t)\cdot a^k(t)] | Z(t)\Big\} \\
&- E\Big\{\sum_{i=1}^{N}[P_i(t)\cdot H_i(t)] | Z(t)\Big\}
\end{aligned} \tag{18}$$

where $D = \frac{1}{2}A_{\text{max}}^2 + H_{\text{max}}^2$. By minimizing the upper bound of the drift-plus-penalty function in Theorem 1, we can achieve a balance between queue stability and minimizing the sensing cost.

Proof. As is well known, for any $a \geq 0, b \geq 0, c \geq 0$, the inequality $(\max[a-b, 0] + c)^2 \leq a^2 + b^2 + c^2 - 2a(b-c)$ holds. Squaring both sides of Eq. (11), subtracting $Q_k^2(t)$, and considering $a^k(t) \leq A^k(t) \leq A_{max}$, Eq. (19) is obtained.

$$\begin{aligned}
&Q_k^2(t+1) - Q_k^2(t) \\
&= \left\{\max[Q_k(t) - r^k(t), 0] + a^k(t)\right\}^2 - Q_k^2(t) \\
&\leq [r^k(t)]^2 + [a^k(t)]^2 - 2Q_k(t)\cdot[r^k(t) - a^k(t)] \\
&\leq [r^k(t)]^2 + A_{\text{max}}^2 - 2Q_k(t)\cdot[r^k(t) - a^k(t)]
\end{aligned} \tag{19}$$

Similarly, since $o_i(t) \leq H_i \leq H_{max}$, squaring both sides of Eq. (13), subtracting $P_i^2(t)$, the inequality is obtained from the Eq. (20).

$$P_i^2(t+1) - P_i^2(t) \leq 2H_i^2 - 2P_i(t)\cdot[H_i(t) - o_i(t)] \tag{20}$$

From this, the Lyapunov drift is derived from the Eq. (21).

$$\begin{aligned}\Delta(Z(t)) \leq& \frac{1}{2}\sum_{k=1}^{K}\left\{[r^k(t)]^2 + A_{\max}^2 - 2Q_k(t)\cdot[r^k(t)-a^k(t)]\right\} \\ &+\frac{1}{2}\sum_{i=1}^{N}\left\{2H_i^2 - 2P_i(t)\cdot[H_i - o_i(t)]\right\} \\ =& \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}\left\{(x_{ij}^k[t])^2 + \frac{A_{\max}^2}{2}\right\} \\ &-\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}Q_k(t)\cdot(x_{ij}^k[t])^2 + H_{\max}^2 \\ &-\sum_{i=1}^{N}P_i(t)\cdot H_i(t) + \sum_{k=1}^{K}Q_k(t)\cdot a^k(t) \\ &+\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}P_i(t)\cdot x_{ij}^k[t]\end{aligned} \tag{21}$$

Building on the above result and incorporating Eq. (1) as the penalty function, simplifying further, the Lyapunov drift-plus-penalty function satisfies the inequality in Eq. (22).

$$\begin{aligned}&\Delta\left(Z\left(t\right)\right) + V\cdot E\left\{u\left(t\right)|\,Z\left(t\right)\right\} \leq D+ \\ &E\Big\{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}\Big[\frac{1}{2}(x_{ij}^k[t])^2 + P_i(t)\cdot x_{ij}^k[t] - Q_k(t)\cdot(x_{ij}^k[t])^2 + \\ &V\cdot x_{ij}^k[t]\cdot c_{ij}^k(t)\Big]|Z(t)\Big\} + E\Big\{\sum_{k=1}^{k}[Q_k(t)\cdot a^k(t)]|Z(t)\Big\} \\ &- E\Big\{\sum_{i=1}^{N}[P_i(t)\cdot H_i(t)]|Z(t)\Big\}\end{aligned} \tag{22}$$

where $D = \frac{1}{2}A_{\max}^2 + H_{\max}^2$. □

3) Online Control: The objective becomes minimizing the upper bound of the drift-plus-penalty function as defined in Theorem 1. Here, D is a constant, $Q_k(t)$ and $P_i(t)$ can be observed. Given that H_i and $c_{ij}(t)$ are known, we need to minimize the upper limit of the Eq. (23).

$$\begin{aligned}&E\Big\{\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}\Big[\frac{1}{2}(x_{ij}^k[t])^2 + P_i(t)\cdot x_{ij}^k[t] - Q_k(t)\cdot(x_{ij}^k[t])^2 \\ &+ V\cdot x_{ij}^k[t]\cdot c_{ij}^k(t)|Z(t)\Big]\Big\} + E\Big\{\sum_{k=1}^{k}[Q_k(t)\cdot a^k(t)]|Z(t)\Big\}\end{aligned} \tag{23}$$

For $E\left\{\sum_{k=1}^{K}\left[Q_k(t)\cdot a^k(t)\right] \mid Z(t)\right\}$, it is necessary to control the enqueue rate $a^k(t)$. For each $Q_k(t)$, there are $A^k(t)$ tasks of type k arriving at the platform, but only $a^k(t)$ are allowed to enter the queue. The value of $a^k(t)$ obey the rule in Eq. (24).

$$a^k(t) = \begin{cases} A_{\max} & \text{if } Q_k(t) = 0 \\ 0 & \text{if } Q_k(t) > 0 \end{cases} \tag{24}$$

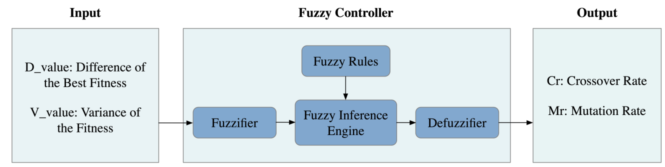


Fig. 3. The framework of fuzzy control system.

V. Fuzzy Genetic Algorithm-Based Allocation

After completing the task admission control, we employ an adaptive genetic algorithm based on a fuzzy controller to allocate tasks. In traditional genetic algorithms, the crossover and mutation rates are often fixed values, which causes the algorithm to get stuck in local optima quickly. Some schemes improve genetic algorithms by adopting dynamic adjustment strategies, adjusting parameters according to the algorithm's execution state and search progress. For instance, researchers use a higher mutation rate in the early stages of the search to increase population diversity, and the mutation rate is gradually decreased in the later stages to accelerate convergence. We enhance the traditional genetic algorithm by using a fuzzy control system to achieve adaptive adjustment of the crossover and mutation rates. Compared to dynamic adjustment strategies, this approach is more flexible and general.

A. Design of Fuzzy Controller

As shown in Fig. 3, the fuzzy control system designed for adaptively adjusting the parameters of the genetic algorithm takes the difference of the best fitness value (D_value) and the variance of the fitness (V_value) as inputs. After processing by the fuzzy controller, it outputs the crossover and mutation rates. The fuzzy controller consists of four components: the fuzzifier, fuzzy inference engine, defuzzifier, and fuzzy rules.

1) Fuzzifier: As inputs to the fuzzy controller, D_value and V_value are crisp values. The fuzzifier converts these two crisp values into linguistic values using membership functions. We predefine several linguistic values for the input variables. Specifically, for D_value and V_value, we use L (Low), M (Medium), and H (High) as linguistic values. Similarly, for the output variables crossover rate (Cr) and mutation rate (Mr), we define the same linguistic values: L (Low), M (Medium), and H (High). The definitions of the linguistic values used in the fuzzy controller are listed in TABLE II.

TABLE II
DEFINITION OF LINGUISTIC VALUES

Variable	Description	Linguistic Value
D_value, V_value, Cr, Mr	Difference of the Best Fitness (Absolute Value), Variance of the Fitness, Crossover Rate, Mutation Rate	L (Low), M (Medium), H (High)

The membership functions corresponding to these four variables are illustrated in Fig. 4. The horizontal axis represents the numerical values of the input crisp values, and the vertical

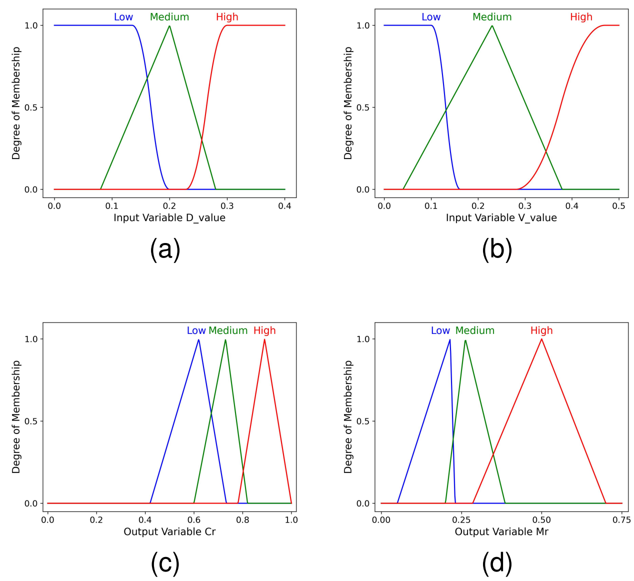


Fig. 4. Membership functions. (a) Membership function of input D_value. (b) Membership function of input V_value. (c) Membership function of output Cr. (d) Membership function of output Mr.

axis represents the degree of membership. The membership functions map the inputs to values between 0 and 1, indicating the degree of membership to different fuzzy sets [62].

2) Fuzzy Inference Engine: Fuzzy inference matches the input values with fuzzy rules and determines the membership values. During the fuzzy inference process, a set of fuzzy rules is defined. These rules are formatted as "IF-THEN" rules [63] with conditions and conclusions. Each rule consists of a condition part (IF) and a conclusion part (THEN), where the condition part describes the relationship between the input variables, and the conclusion part describes the value of the output variables. We use the Mandani fuzzy inference method [64] to calculate the output fuzzy set. The complete set of defined fuzzy rules is listed in TABLE III.

TABLE III
FUZZY RULES

D_value	V_value	Cr	Mr
L	L	M	H
L	M	M	M
L	H	H	M
M	L	L	H
M	M	L	M
M	H	M	H
H	L	L	H
H	M	L	H
H	H	M	H

3) Defuzzifier: The fuzzy inference yields fuzzy values, converted back into crisp values through the defuzzifier. We use the commonly employed center of gravity (COG) strategy for defuzzification. The COG determines the crisp values by calculating the centroid of the membership function of the fuzzy output set. The two outputs obtained are the updated crossover rate and mutation rate, which will be used in the next generation of the genetic algorithm.

B. Lya-FCGA

We utilize the fuzzy control system to enhance the genetic algorithm, achieving adaptive adjustments of the crossover and mutation rates. The improved fuzzy genetic algorithm derives task allocation schemes.

In the Fuzzy Controller-based Genetic Algorithm (FCGA), we represent each individual as a permutation of length N, where N is the number of workers, and the permutation includes matched task indices. Before the algorithm starts, we perform a random initialization where each task index appears at most once. We use the upper bound of the drift-plus-penalty function as the fitness function. Each generation selects the solution with the minimum fitness as the optimal solution. Subsequently, crossover operations are performed by randomly selecting two crossover points and swapping the gene segments between these points from different individuals to generate two new individuals. Next, we choose a random start and end point and reverse the sequence of genes within the selected segment, completing the mutation operation. Finally, we merge the parent and offspring populations to form the next generation population. This process repeats until the maximum number of generations is reached.

The Lyapunov Optimization with Fuzzy Controller-based Genetic Algorithm (Lya-FCGA) combines Lyapunov optimization and fuzzy controller-enhanced genetic algorithms to allocate tasks in VMCS. The algorithm takes the task set (W), worker set (U), the sensing task queue ($Q_i(t)$), the worker capability deficit queue ($P_j(t)$), and the maximum number of generations ($maxgen$) as inputs. The output is the task allocation result (X). Algorithm 1 presents the pseudo-code of our proposed method. The algorithm starts by initializing the task and worker capability deficit queues. Then, for each time slot, the genetic algorithm is executed, where the fuzzy controller adapts the crossover and mutation rates to balance exploration and exploitation. The process iterates for a specified number of generations, and after completing task allocation, the task and capability deficit queues are updated in real-time.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed Lya-FCGA with privacy preservation.

A. Dataset

We use the widely utilized CRAWDAD dataset [65] for our experiments to effectively simulate the dynamic changes of tasks and workers in an online scenario. This dataset includes 600 sets of routes in London. Each route contains information such as travel duration, route length, traffic conditions, and accident situations. We preprocessed the dataset, selecting 200 trajectories as the worker set and another 400 as the task set.

Algorithm 1 Lya-FCGA Algorithm

Require: Task set W, Worker set U, Sensing task queue $Q_i(t)$, Worker capability deficit queue $P_j(t)$, Maximum number of generations $maxgen$
Ensure: Task allocation result X

```
Initialize Q_i(t) and P_j(t);
for each time slot do
    Encoding and initialize population;
    gen ← 1;
    while gen ≤ maxgen do
        Calculate individual fitness;
        Adaptively adjust the crossover rate and mutation rate using fuzzy controller;
        Perform selection operation;
        Perform crossover operation based on the crossover rate;
        Perform mutation operation based on the mutation rate;
        Generate the next generation population;
        gen ← gen + 1;
    end while
    Update Q_i(t) and P_j(t);
end for
```

B. Parameter Setting

We consider that the VMCS system runs in 500 time slots. The number of workers in each time slot is set to a minimum of 20 and a maximum of 60, while the number of tasks varies between 60 and 100. Each worker's processing capability is 1 per time slot. For tasks, the number of task types varies between 1 and 10. The privacy budget is set to 1 by default, as commonly used in the mobile crowdsensing domain [66], [67]. For the Lya-FCGA algorithm, the initial crossover rate is 0.8, the mutation rate is 0.1, the initial population size is 10, and the number of population iterations is 100. During the algorithm's execution, the designed fuzzy control system adaptively adjusts the crossover and mutation rates, with thresholds set to ensure they remain within a specific range.

We use queue backlog and sensing cost to measure the performance of task allocation. The system's queues include the sensing task queue $Q(t)$ and the worker capability deficit queue $P(t)$. Queue backlog can reflect the system's stability. Sensing cost, also known as travel cost, is incurred when workers travel to task points. Suppose $x_{ij}[t]$ denotes the allocation result of u_i to w_j in time slot t, and $c_{ij}(t)$ represents the Euclidean distance between this worker and the task. The sensing cost in time slot t can be computed from the Eq. (25).

$$u(t) = \sum_{i=1}^{N} \sum_{j=1}^{M} x_{ij}[t] \cdot c_{ij}(t) \tag{25}$$

The total sensing cost accumulates the travel costs generated in each time slot.

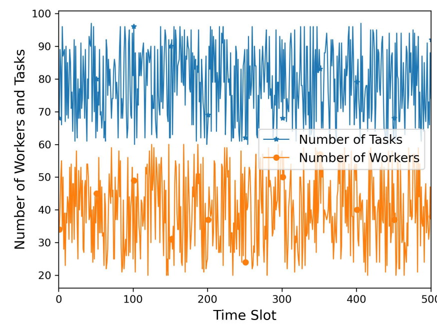


Fig. 5. Variation in the number of tasks and workers over time.

C. Baseline

To validate the performance of the proposed Lya-FCGA with privacy preservation, we compare it with the following schemes.

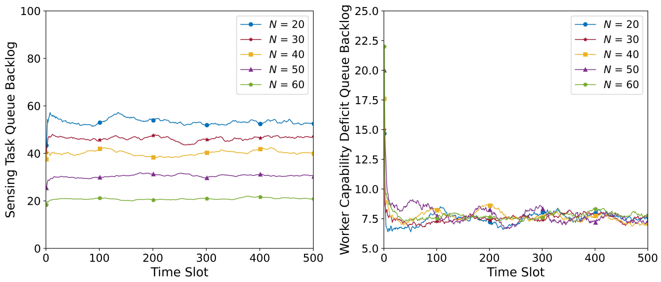
- Random Allocation **(RA)**. The VMCS platform randomly assigns tasks to workers in each time slot.
- Lowest Queue **(LQ)**. The VMCS platform assigns tasks to the worker with the lowest queue backlog in each time slot, ensuring minimal average queue backlog.
- Lyapunov optimization with Genetic Algorithm **(Lya-GA)**.This method uses Lyapunov optimization to transform the original problem into a queue control problem and employs a traditional genetic algorithm to solve the task allocation problem.
- Lyapunov optimization with Modified Fuzzy Adaptive Genetic Algorithm [68] **(Lya-MFGA)**. We use the MFGA algorithm for comparison. This algorithm devises a fuzzy logic controller to control the crossover and mutation rates that prevent the genetic algorithm from getting stuck in a local optimum.
- Lyapunov optimization with Fuzzy Controller-based Genetic Algorithm with No Privacy **(Lya-FCGA with NP)**. This method uses Lyapunov optimization to transform the original problem into a queue control problem and utilizes a fuzzy control system to optimize the genetic algorithm for task allocation without considering privacy preservation.

We run each algorithm ten times in each experiment and take the average values.

D. Experimental Results and Analysis

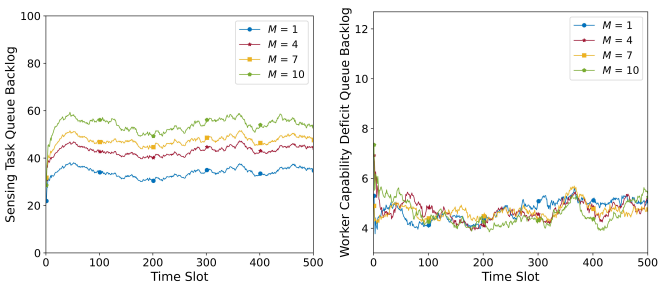
In online VMCS, the states of tasks and workers change over time, and their numbers fluctuate accordingly. Fig. 5 illustrates the variation in the number of tasks and workers over time.

Fig. 6 depicts the changes in Lya-FCGA's queue backlogs over time as the number of workers N is altered. As shown in Fig. 6a, the sensing task queue backlog decreases as the number of workers increases. The reduction in backlog occurs because of the increased number of tasks being executed as the number of workers grows, thereby increasing the dequeue speed. Fig. 6b indicates that the worker capability deficit queue



(a) (b)

Fig. 6. Variation in queue backlogs over time with varying number of workers N. (a) comparison of sensing task queue backlog over time with varying number of workers N. (b) Comparison of worker capability deficit queue backlog over time with varying number of workers N.



(a) (b)

Fig. 7. Variation in queue backlogs over time with varying number of task types M. (a) Comparison of sensing task queue backlog over time with varying number of task types M. (b) Comparison of worker capability deficit queue backlog over time with varying number of task types M.

remains stable as the number of workers grows. This stability arises because the total processing capability of all workers and the total number of processed tasks increase at a similar rate in each time slot.

Fig. 7 illustrates the changes in sensing task queue backlog and worker capability deficit queue backlog over time as the number of task types M varies for Lya-FCGA. Fig. 7a shows that as the number of task types M increases, the sensing task queue backlog gradually increases due to the higher number of tasks generated. However, due to the use of Lyapunov optimization controlling the enqueue speed, the queue remains stable. Fig. 7b indicates that the worker capability deficit queue remains stable as the number of task types increases. The stability is maintained because changes in the number of task types do not affect workers' processing capabilities.

The control parameter V is used to balance queue stability and minimize sensing costs. Fig. 8 illustrates the trend of sensing cost as the control parameter V changes in Lya-FCGA. As the control parameter V increases, the sensing cost decreases. A larger control parameter indicates that the system emphasizes minimizing sensing costs, thus reducing the travel cost.

Fig. 9 shows the variation in total cost as the privacy budget changes in Lya-FCGA. The results indicate that the sensing cost decreases as the privacy budget increases because a larger privacy budget implies a lower level of privacy preservation,

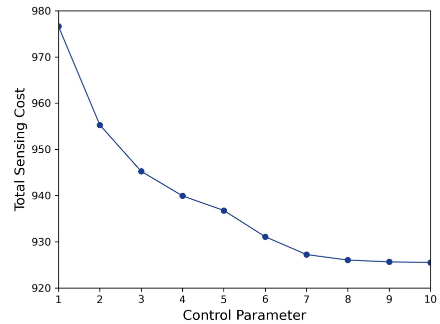


Fig. 8. Variation in total sensing cost with varying control parameter.

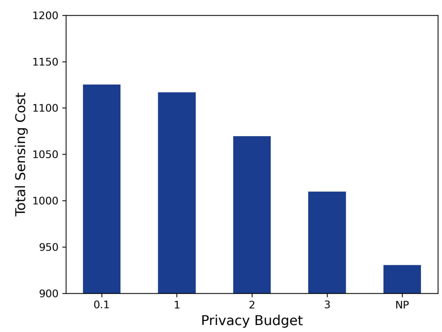


Fig. 9. Variation in total sensing cost with varying privacy budget.

causing the perturbed positions of workers to be closer to their actual locations and resulting in more accurate allocation results.

Fig. 10 depicts the variations in sensing cost over time when the number of task types is fixed at 4, comparing Lya-FCGA with RA, LQ, Lya-GA and Lya-MFGA. As seen in Fig. 10, Lya-FCGA consistently exhibits lower sensing costs over time than the other four schemes, demonstrating better performance. Fig. 11a and Fig. 11b depict the variations in sensing task queue backlog and worker capability deficit queue backlog over time. From Fig. 11, we can see that Lya-FCGA's queue backlog maintains long-term stability, although it does not always remain the lowest. Regarding sensing cost and queue stability, Lya-FCGA outperforms RA, Lya-GA, and Lya-MFGA. Although LQ achieves a lower queue backlog because its primary objective is to minimize queue backlog, Lya-FCGA's sensing cost is significantly lower.

Fig. 12 demonstrates the variation in sensing cost as the number of workers changes across the five schemes. It is evident that as the number of workers increases, the total cost rises for all schemes, with Lya-FCGA maintaining the lowest cost. This trend occurs because each worker's processing capability remains constant. When there are more available workers, more tasks will be allocated and executed per time slot, thereby increasing the total travel distance.

Fig. 13 demonstrates the variation in sensing cost as the number of task types M changes. We observe that as the number of task types increases, the total cost slightly decreases for all schemes, with Lya-FCGA achieving the lowest cost.

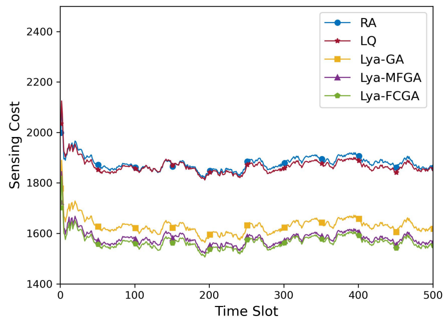


Fig. 10. Variation in sensing cost over time.

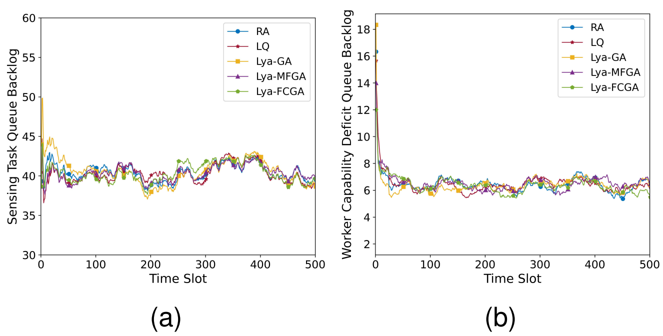


Fig. 11. Variation in queue backlogs over time. (a) Variation in sensing task queue backlog over time. (b) Variation in worker capability deficit queue backlog over time.

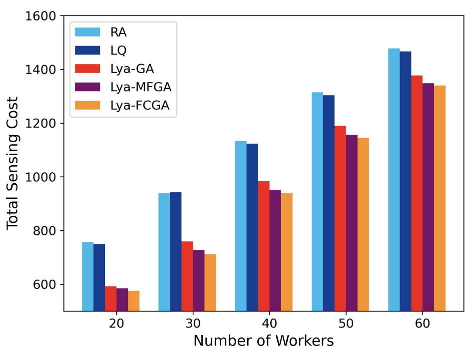


Fig. 12. Total sensing cost variation with varying number of workers.

This trend occurs because a greater variety of tasks results in a greater number of tasks, potentially offering workers better options.

Fig. 14 compares the fitness of Lya-FCGA, Lya-MFGA, and Lya-GA as the number of iterations increases. All these algorithms converge as the number of iterations increases, but Lya-FCGA converges faster and achieves a lower fitness value than Lya-MFGA and Lya-GA, indicating better performance. This improvement is attributed to the fuzzy control system, which adaptively adjust genetic algorithm parameters during the optimization process. Moreover, the integration of Lyapunov optimization ensures queue stability over time, further supporting overall convergence. These enhancements enable the proposed Lya-FCGA scheme to achieve efficient and stable convergence in practice.

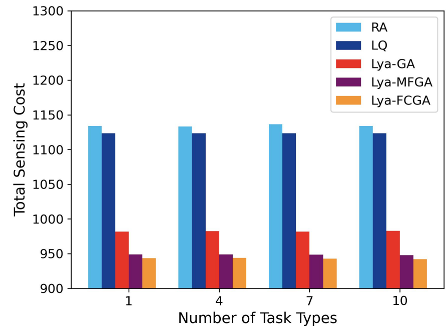


Fig. 13. Variation in total sensing cost with varying number of task types.

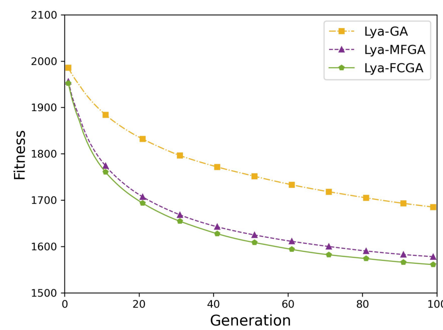


Fig. 14. Comparison of Lya-FCGA, Lya-MFGA, and Lya-GA.

VII. Conclusion

In this paper, an online task allocation method with privacy preservation was proposed for VMCS. By integrating Lyapunov optimization with a fuzzy genetic algorithm, the approach effectively balances long-term system stability and sensing costs while preserving location privacy. These capabilities not only enhance the efficiency and reliability of VMCS but also support the practical deployment of large-scale, sustainable systems in intelligent transportation scenarios. The proposed method's ability to ensure privacy protection and system stability is crucial for widespread adoption in real-world VMCS applications. Future work can be further extended by incorporating deep reinforcement learning to improve the adaptability and decision-making capability of task allocation in dynamic and uncertain environments.

VIII. Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 62472113, 62372121, 62272150 and 62172159, the Science and Technology Projects in Guangzhou under Grant 2025A03J3127, the Natural Science Foundation of Guangdong Province under Grant 2023A1515012358, the China Telecom Cloud Computing Research Institute under Grant HQCCR2400006EGN00, and the Soft Science Project of the National Intellectual Property Administration under Grant SS24-B-23.

REFERENCES

[1] L. Hu, A. Liu, M. Xie *et al.*, "UAVs joint vehicles as data mules for fast codes dissemination for edge networking in Smart City," *Peer-to-Peer Netw. Appl.*, vol. 12, pp. 1550-1574, Jun. 2019.

[2] C. Liu, G. Kou, X. Zhou *et al.*, "Time-dependent vehicle routing problem with time windows of city logistics with a congestion avoidance approach," *Knowl.-Based Syst.*, vol. 188, Jan. 2020, Art. no. 104813.

[3] R. K. Ganti, F. Ye and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32-39, Nov. 2011.

[4] F. Abbondati, S. A. Biancardo, R. Veropalumbo, and G. Dell'Acqua, "Surface monitoring of road pavements using mobile crowdsensing technology," *Measurement*, vol. 171, Feb. 2021, Art. no. 108763.

[5] A. Bhattacharya and D. De, "SigSense: Mobile crowdsensing based incentive aware geospatial signal monitoring for base station installation recommendation using mixed reality game," *Wireless Pers Commun*, vol. 123, pp. 2863–2894, Apr. 2022.

[6] L. Shang, Y. Zhang, Q. Ye, N. Wei, and D. Wang, "SmartWaterSens: A crowdsensing-based approach to groundwater contamination estimation," in *Proc. 2022 IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, Helsinki, Finland, Jul. 2022, pp. 48-55.

[7] V. Radu, L. Kriara, and M. K. Marina, "Pazl: A mobile crowdsensing based indoor WiFi monitoring system," in *Proc. 9th Int. Conf. Network and Service Manag. (CNSM)*, Zurich, Switzerland, Oct. 2013, pp. 75-83.

[8] M. Kim and Y. Kim, "Multi-Blockchain structure for a crowdsensing-based smart parking system," *Future Internet*, vol. 12, no. 5, pp. 90, May 2020.

[9] F. Montori, L. Bedogni, A. Di Chiappari, and L. Bononi, "SenSquare: A mobile crowdsensing architecture for smart cities," in *Proc. 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, Reston, VA, USA, Dec. 2016, pp. 536-541.

[10] M. M. Cruz *et al.*, "Assessing the level of acceptance of a crowdsourcing solution to monitor infectious diseases propagation," in *Proc. 2020 IEEE Int. Smart Cities Conf. (ISC2)*, Piscataway, NJ, USA, Dec. 2020, pp. 1-8.

[11] R. Pryss, J. Schobel, and M. Reichert, "Requirements for a flexible and generic API enabling mobile crowdsensing mHealth applications," in *Proc. 2018 4th Int. Workshop on Requirements Engineering for Self-Adaptive, Collaborative, and Cyber Physical Systems (RESACS)*, Banff, AB, Canada, Aug. 2018, pp. 24-31.

[12] C. Vogel, R. Pryss, J. Schobel, W. Schlee, and F. Beierle, "Developing apps for researching the COVID-19 pandemic with the TrackYourHealth platform," in *Proc. 2021 IEEE/ACM 8th Int. Conf. on Mobile Software Engineering and Systems (MobileSoft)*, Madrid, Spain, May 2021, pp. 65-68.

[13] Z. Xu *et al.*, "Social sensors based online attention computing of public safety events," *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 3, pp. 403-411, Jul.-Sept. 2017.

[14] T. Peng, K. Guan, J. Liu, J. Chen, G. Wang, and J. Zhu, "A blockchain-based mobile crowdsensing scheme with enhanced privacy," *Concurrency and Comput.: Practice and Experience*, vol. 35, no. 19, Dec. 2021, Art. no. e6664.

[15] T. Peng, W. Zhong, G. Wang, S. Zhang, E. Luo, and T. Wang, "Spatiotemporal-aware privacy-preserving task matching in mobile crowdsensing," *IEEE Internet of Things J.*, vol. 11, no. 2, pp. 2394-2406, Jan. 2024.

[16] T. Peng, W. Zhong, G. Wang, E. Luo, J. Xiong, Y. Liu, and W. Hao, "Privacy-preserving precise profile matching in mobile social network," *J. Commun.*, vol. 43, no. 11, pp. 90-103, Nov. 2022.

[17] S. Zhang, T. Guo, Q. Liu, E. Luo, K.-K. R. Choo, and G. Wang, "ALPS: Achieving accuracy-aware location privacy service via assisted regions," *Future Gen. Comput. Syst.*, vol. 145, pp. 189-199, Aug. 2023.

[18] T. Peng, Q. Liu, D. Meng, and G. Wang, "Collaborative trajectory privacy preserving scheme in location-based services," *Inf. Sci.*, vol. 387, pp. 165-179, May 2017.

[19] F. Fei, S. Li, H. Dai, C. Hu, W. Dou and Q. Ni, "A k-anonymity based schema for location privacy preservation," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 2, pp. 156-167, Apr.-Jun. 2019.

[20] Q. Liu *et al.*, "SlimBox: Lightweight packet inspection over encrypted traffic," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 5, pp. 4359-4371, Sept.-Oct. 2023.

[21] Q. Liu, Y. Tian, J. Wu, T. Peng, and G. Wang, "Enabling verifiable and dynamic ranked search over outsourced data," *IEEE Trans. Serv. Comput.*, vol. 15, no. 1, pp. 69-82, Jan.-Feb. 2022.

[22] Z. Liu *et al.*, "PPRU: A privacy-preserving reputation updating scheme for cloud-assisted vehicular networks," *IEEE Trans. Veh. Technol.* DOI: 10.1109/TVT.2023.3340723.

[23] K. Gai, D. Wang, J. Yu, L. Zhu and W. Meng, "A scheme of robust privacy-preserving multi-party computation via public verification," *IEEE Trans. Dependable Secure Comput.* DOI: 10.1109/TDSC.2025.3555284.

[24] S. Zhang, Y. Wang, E. Luo, Q. Liu, K. Gu, and G. Wang, "A traceable and revocable decentralized multi-authority privacy protection scheme for social metaverse," *J. Syst. Archit.*, vol. 140, Jul. 2023, Art. no. 102899.

[25] X. Zhao, N. Wang, R. Han, B. Xie, Y. Yu, M. Li, and J. Ou, "Urban infrastructure safety system based on mobile crowdsensing," *Int. J. Disaster Risk Reduction*, vol. 27, pp. 427-438, May 2018.

[26] Y. Zou, T. Peng, G. Wang, E. Luo, and J. Xiong, "Blockchain-assisted multi-keyword fuzzy search encryption for secure data sharing," *J. Syst. Archit.*, vol. 144, Jul. 2023, Art. no. 102984.

[27] M. Li, M. Zhang, L. Zhu, Z. Zhang, M. Conti, and M. Alazab, "Decentralized and privacy-preserving smart parking with secure repetition and full verifiability," *IEEE Trans. Mobile Comput.* DOI: 10.1109/TMC.2024.3397687.

[28] T. Peng, K. Guan, and J. Liu, "A privacy-preserving mobile crowdsensing scheme based on blockchain and trusted execution environment," *IEICE Trans. Inf. Syst.*, vol. E105-D, no. 2, pp. 215-226, Feb. 2022.

[29] J. Guo *et al.*, "TFL-DT: A trust evaluation scheme for federated learning in digital twin for mobile networks," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 11, pp. 3548-3560, Nov. 2023.

[30] X. Chen, G. Xu, X. Xu, H. Jiang, Z. Tian and T. Ma, "Multicenter hierarchical federated learning with fault-tolerance mechanisms for resilient edge computing networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 1, pp. 47-61, Jan. 2025.

[31] Z. Ning *et al.*, "Blockchain-enabled intelligent transportation systems: a distributed crowdsensing framework," *IEEE Trans. Mob. Comput.*, vol. 21, no. 12, pp. 4201-4217, Dec. 2022.

[32] C. H. Liu *et al.*, "Curiosity-driven energy-efficient worker scheduling in vehicular crowdsourcing: a deep reinforcement learning approach," in *Proc. 2020 IEEE 36th Int. Conf. on Data Engineering (ICDE)*, Dallas, TX, USA, Apr. 2020, pp. 25-36.

[33] J. Wang, Y. Wang, D. Zhang, F. Wang, Y. He, and L. Ma, "PSAllocator: Multi-task allocation for participatory sensing with sensing capability constraints," in *Proc. ACM Conf. on Embedded Networked Sensor Systems (SenSys)*, New York, NY, USA, Feb. 2017, pp. 1139-1151.

[34] E. Wang, Y. Yang, J. Wu, W. Liu, and X. Wang, "An efficient prediction-based user recruitment for mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 17, no. 1, pp. 16-28, Jan. 2018.

[35] W. Li, B. Jia, H. Xu, Z. Zong, and T. Watanabe, "A multi-task scheduling mechanism based on ACO for maximizing workers' benefits in mobile crowdsensing service markets with the Internet of Things," *IEEE Access*, vol. 7, pp. 41463-41469, Jan. 2019.

[36] M. H. Cheung, F. Hou, J. Huang, and R. Southwell, "Distributed time-sensitive task selection in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 20, no. 6, pp. 2172-2185, Jun. 2021.

[37] J. Han, Z. Zhang, and D. Kong, "Distributed multi-task allocation method for user area in mobile crowd sensing," *J. Comput. Appl.*, vol. 40, no. 2, pp. 358-362, 2020.

[38] Z. Wang *et al.*, "Towards personalized task-oriented worker recruitment in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 20, no. 5, pp. 2080-2093, May 2021.

[39] B. Zhao, H. Dong, Y. Wang, X. Gao and T. Pan, "An enhanced task allocation algorithm for mobile crowdsourcing based on spatiotemporal attention network," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 3, pp. 3803-3815, Jun. 2024.

[40] X. Liu *et al.*, "A coverage-aware task allocation method for UAV-assisted mobile crowd sensing," *IEEE Trans. Veh. Technol.*, vol. 73, no. 7, pp. 10642-10654, Jul. 2024.

[41] Y. Jiang, K. Zhang, Y. Qian, and L. Zhou, "P2AE: Preserving privacy, accuracy, and efficiency in location-dependent mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 22, no. 4, pp. 2323-2339, Apr. 2023.

[42] S. Gao, X. Chen, J. Zhu, and L. Yuan, "A location privacy-preserving worker selection scheme under limited budget for blockchain-based crowdsensing," *Chinese J. Comput.*, vol. 45, no. 5, pp. 1052-1067, May 2022.

[43] Z. Li, J. Wu, S. Long, Z. Zheng, C. Li and M. Dong, "User-driven privacy-preserving data streams release for multi-task assignment in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 24, no. 5, pp. 3719-3734, May 2025.

[44] H. Wu, B. Düdder, S. Jiang and L. Wang, "VP22-match: Verifiable privacy-aware and personalized crowdsourcing task matching via blockchain," *IEEE Trans. Mobile Comput.*, vol. 23, no. 10, pp. 9913-9930, Oct. 2024.

[45] L. Li, H. Wu, L. He, J. Yang, Z. Cao and B. Düdder, "Matching as you want: A decentralized, flexible, and efficient realization for crowdsourcing

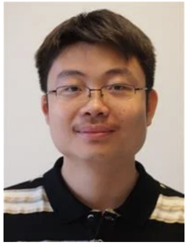
with dual-side privacy," *IEEE Trans. Netw. Sci. Eng.*, vol. 12, no. 2, pp. 1026-1040, Mar.-Apr. 2025.
[46] T. Peng, W. You, K. Guan, E. Luo, S. Zhang, G. Wang, T. Wang, and Y. Wu, "Privacy-preserving multiobjective task assignment scheme with differential obfuscation in mobile crowdsensing," *J. Netw. Comput. Appl.*, vol. 224, Apr. 2024, Art. no. 103836.
[47] S. Peng, W. Gong, B. Zhang, and C. Li, "AP-assisted online task assignment for mobile crowdsensing," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Waikoloa, HI, USA, 2019, pp. 1-6, Dec. 2019.
[48] Y. Lin, F. Wu, L. Kong, and G. Chen, "Quality-based user recruitment in mobile crowdsensing," in *Proc. 14th Int. Conf. Mobile Ad-Hoc and Sensor Networks (MSN)*, Shenyang, China, 2018, pp. 74-80, Dec. 2018.
[49] W. Liu, Y. Yang, E. Wang, and J. Wu, "Dynamic user recruitment with truthful pricing for mobile crowdsensing," in *Proc. IEEE INFOCOM*, Toronto, ON, Canada, 2020, pp. 1113-1122, Jul. 2020.
[50] Z. Shi, S. Jiang, L. Zhang, Y. Du, and X.-Y. Li, "Crowdsourcing system for numerical tasks based on latent topic aware worker reliability," in *Proc. IEEE INFOCOM*, Vancouver, BC, Canada, 2021, pp. 1-10, May 2021.
[51] H. To, C. Shahabi, and L. Xiong, "Privacy-preserving online task assignment in spatial crowdsourcing with untrusted server," in *Proc. IEEE 34th Int. Conf. Data Eng. (ICDE)*, Paris, France, 2018, pp. 833-844, Apr. 2018.
[52] X. Wang, R. Jia, X. Tian, X. Gan, L. Fu, and X. Wang, "Location-aware crowdsensing: Dynamic task assignment and truth inference," *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 362-375, Feb. 2020.
[53] Q. Tao, Y. Tong, Z. Zhou, Y. Shi, L. Chen, and K. Xu, "Differentially private online task assignment in spatial crowdsourcing: A tree-based approach," in *Proc. IEEE 36th Int. Conf. Data Eng. (ICDE)*, Dallas, TX, USA, 2020, pp. 517-528, Apr. 2020.
[54] X. Wang *et al.*, "Online spatial crowdsensing with expertise-aware truth inference and task allocation," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 412-427, Jan. 2022.
[55] X. Tao and W. Song, "Task allocation for mobile crowdsensing with deep reinforcement learning," in *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, Seoul, Korea (South), 2020, pp. 1-7, May 2020.
[56] H. Wang, E. Wang, Y. Yang, J. Wu, and F. Dressler, "Privacy-preserving online task assignment in spatial crowdsourcing: A graph-based approach," in *Proc. 41st IEEE Conf. Comput. Commun. (INFOCOM 2022)*, London, United Kingdom, May 2022, pp. 570-579.
[57] Q. Zhang, Y. Wang, G. Yin, X. Tong, A. M. V. V. Sai, and Z. Cai, "Two-stage bilateral online priority assignment in spatio-temporal crowdsourcing," *IEEE Trans. Serv. Comput.*, vol. 16, no. 3, pp. 2267-2282, May-Jun. 2023.
[58] J. Zhang, D. He, X. Chen, X. Zhong, and P. Tsai, "LSTM-Oppurs: Opportunistic user recruitment strategy based on deep learning in mobile crowdsensing system," *Future Gener. Comput. Syst.*, vol. 162, Jan. 2025, Art. no. 107490.
[59] Y. Li, Y. Wang, G. Li, X. Tong, and Z. Cai, "Determining task assignments for candidate workers based on trajectory prediction," *IEEE Trans. Mobile Comput.*, vol. 24, no. 5, pp. 3890-3902, May. 2025.
[60] Z. Chen, M. Xu, and C. Su, "Online quality-based privacy-preserving task allocation in mobile crowdsensing," *Comput. Netw.*, vol. 251, Sep. 2024, Art. no. 110613.
[61] C. Zhang, X. Luo, J. Liang, X. Liu, L. Zhu, and S. Guo, "POTA: Privacy-preserving online multi-task assignment with path planning," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5999-6011, May 2024.
[62] X. Li, X. Chen, "D-intuitionistic hesitant fuzzy sets and their application in multiple attribute decision making," *Cogn. Comput.*, vol. 10, pp. 496-505, Feb. 2018.
[63] G. Yang, Y. Zhang, B. Wang, et al., "Task allocation through fuzzy logic based participant density analysis in mobile crowd sensing," *Peer-to-Peer Netw. Appl.*, vol. 14, pp. 763–780, Jan. 2021.
[64] R. R. Panda and M. Reza, "Optimization of buffer overflow probability in Jackson queueing networks using Mamdani fuzzy inference system," in *Proc. 2nd Int. Conf. Bus. and Inf. Manag. (ICBIM)*, Durgapur, India, 2014, pp. 92-98, Jan. 2014.
[65] *CRAWDAD dataset urfj/london-trajectories*. CRAWDAD Team. [Online]. Available: https://ieee-dataport.org/open-access/crawdad-ufrjlondon-trajectories
[66] Q. Zhang, T. Wang , Y. Tao , F. Chen , D. Xie , and C. Zhao, "Trajectory privacy protection method based on differential privacy in crowdsensing," *IEEE Trans. Serv. Comput.*, vol. 17, no. 6, pp. 4423-4435, Nov.-Dec. 2024.
[67] Z. Li, X. Zeng, Y. Xiao, C. Li, W. Wu and H. Liu, "Pattern-sensitive local differential privacy for finite-range time-series data in mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 24, no. 1, pp. 1-14, Jan. 2025.
[68] N. Rizvi, D. Ramesh, L. Wang, and A. Basava, "A workflow scheduling approach with modified fuzzy adaptive genetic algorithm in IaaS clouds," *IEEE Trans. Serv. Comput.*, vol. 16, no. 2, pp. 872-885, Mar.-Apr. 2023.

Wei You received the bachelor's degree from the School of Information, Jiangxi University of Science and Technology in 2022 and is currently studying for a master's degree with the School of Computer Science and Cyber Engineering, Guangzhou University.

Tao Peng received the B.Sc. degree in computer science from Xiangtan University, China, in 2004, the M.Sc. degree in circuits and systems from Hunan Normal University, China, in 2007, and the Ph.D. degree in computer science from Central South University, China, in 2017. She is currently an associate professor in the School of Computer Science and Educational Software, Guangzhou University, China. Her research interests include network and information security issues.

Zhidong Xie received the bachelor's degree from the School of Computer Science and Cyber Engineering, Guangzhou University in 2022 and is currently studying for a master's degree with the School of Computer Science and Cyber Engineering, Guangzhou University.

Guojun Wang received his B.S. in Geophysics, in 1992, M.S. in Computer Science, in 1996, and Ph.D. in Computer Science, in 2002, from Central South University, China. He is the Pearl River Scholar Professor of Guangdong Province at School of Computer Science and Cyber Engineering, Guangzhou University, China. He has been an Adjunct Professor at Temple University, USA; a Visiting Scholar at Florida Atlantic University, USA; a Visiting Researcher at the University of Aizu, Japan; and a Research Fellow at the Hong Kong Polytechnic University. His research interests include network and information security, Internet of things, and cloud computing. He is a distinguished member of CCF, and a member of IEEE, ACM, and IEICE.



Weizhi Meng is currently an Associate Professor in the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Denmark. He received the IEEE ComSoc Best Young Researcher Award for Europe, Middle East, & Africa Region (EMEA) in 2020. His primary research interests are intersections among cyber security, artificial intelligence and blockchain technology, such as intrusion detection, IoT security, biometric authentication, and blockchain. He serves as associate editors / editorial board members for many reputed journals such as IEEE TDSC and IEEE TIFS, as well as chair for various international conferences such as ACM CCS'23 and ESORICS'22. He is a senior member of IEEE.

Qin Liu received her B.Sc. in Computer Science in 2004 from Hunan Normal University, China, received her M.Sc.in Computer Science in 2007, and received her Ph.D.in Computer Science in 2012 from Central South University, China. She has been a Visiting Student at Temple University, USA. Her research interests include security and privacy issues in cloud computing. Now, she is an Associate Professor in the College of Computer Science and Electronic Engineering at Hunan University, China.

Entao Luo is a professor at the School of Information Engineering, Hunan University of Science and Technology, He received his Ph.D. degree in Software Engineering from Central South University, China, His research fields include mobile social networking, machine learning, and edge computing security and privacy preserving. He has published more than 30 technical papers in the above areas, including Journal of Software, Journal of Computer Research and Development, Information Science, FGCS, IEEE Communications Letters etc.