frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 08 October 2025
DOI 10.3389/fphy.2025.1624913



OPEN ACCESS

EDITED BY
Jiazhong Lu,
Chengdu University of Information Technology, China

REVIEWED BY
Mohamed Ridha Znaidi,
University of Southern California, United States
Abdullah Ayub Khan,
Bahria University, Pakistan

*CORRESPONDENCE
Qian Li,
a15968377016@sina.com

RECEIVED 08 May 2025
ACCEPTED 08 September 2025
PUBLISHED 08 October 2025

CITATION
Li Q, Chen M and Chen Z (2025) P3TRTA-RDQL: a crowdsensing task allocation scheme integrating privacy protection protocol and low-dimensional reinforcement learning.
*Front. Phys.* 13:1624913.
doi: 10.3389/fphy.2025.1624913



# P3TRTA-RDQL: a crowdsensing task allocation scheme integrating privacy protection protocol and low-dimensional reinforcement learning

Qian Li*, Manlu Chen and Zhiwei Chen

Electric Power Dispatching and Control Center of Guangdong Power Grid Co., Ltd., Guangzhou, China

Crowdsensing, as an emerging data collection mode, demonstrates great potential in Internet of Things (IoT) applications. However, it faces a critical trilemma: accurate task allocation depends on node proximity to the task location, but disclosing location data risks privacy leakage, while concealing it reduces allocation precision. Existing solutions either incur high computational overhead (encryption), rely on unfeasible trusted third parties (anonymization), or degrade data utility (obfuscation), failing to balance privacy, accuracy, and efficiency. To address these issues, this paper proposes the P3TRTA-RDQL scheme, combining a symmetric encryption-based privacy protection protocol (P3TRTA) with a low-dimensional Q-learning algorithm (RDQL). The P3TRTA protocol uses a location-based symmetric key generator (LSKeyGen) to protect node/task location privacy and proxy re-encryption to secure task content, eliminating reliance on trusted third parties. The RDQL algorithm reduces state dimensionality by 60% compared to traditional reinforcement learning, enhancing large-scale task allocation efficiency. Experimental results show that P3TRTA-RDQL outperforms existing methods by 30% in privacy protection strength, achieves 98% task allocation accuracy, and reduces allocation time for 1000 tasks by 40%. This work provides technical support for crowdsensing's widespread IoT applications.

KEYWORDS

**crowdsensing, task allocation, privacy protection, symmetric encryption, reinforcement learning, low-dimensional Q-Learning algorithm**

## 1 Introduction

With the widespread deployment of intelligent terminals and the rapid development of cloud/edge computing, using crowd sensing to collect data has become a new trend. As an emerging technology, crowd sensing encourages mobile nodes to participate in collaborative and distributed data collection. The mobile nodes here refer to mobile devices or terminals and individuals equipped with mobile devices. With the continuous development of hardware manufacturing technology, current mobile devices are usually embedded with rich sensors and computing/communication modules, and have the capabilities to collect data from the surrounding environment, preprocess data, and transmit data. In addition, with the continuous popularization and deployment of intelligent devices, mobile devices can be seen everywhere in people's lives. Crowd sensing is also used to assist in air

quality measurement, noise detection, and other Internet of Things applications. Compared with the traditional data collection mode based on sensor networks, crowd sensing does not require the installation and maintenance of sensors, and has great advantages in terms of economic cost and labor consumption. Especially when it comes to large-scale data collection, this advantage is more obvious. Equipped with rich sensors and computing modules, these devices enable real-time data collection, preprocessing, and transmission, supporting applications like Internet of Vehicles, intelligent parking, and 3D model reconstruction. Compared with traditional sensor networks, crowd sensing avoids sensor deployment/maintenance costs, making it more scalable for large-scale IoT scenarios.

Mobile crowdsensing (MCS), as an emerging data collection mode, crowdsources tasks to participants carrying IoT devices to achieve human-object collaborative data collection [1]. However, the open network, the mobility of participants, and untrusted cloud servers pose severe challenges to the task allocation and data security of mobile crowdsensing [2]. Solving these problems is crucial for promoting the practical application of mobile crowdsensing and ensuring the secure and efficient development of the Internet of Things. Mobile crowdsensing currently faces a specific trilemma: accurate task allocation relies on node location proximity, but disclosing locations risks privacy leakage, while concealing them reduces allocation precision.

This paper conducts research on the deficiencies of mobile crowdsensing in the Internet of Things in terms of task allocation and data security, especially the data security issues in the scenario of mobile crowdsensing driven by federated learning (FL). Mobile crowdsensing was first proposed by Ganti et al. [2], and Guo et al. [3] defined it as users using mobile devices to collect or generate data and aggregate it on the cloud service side to provide services for the community. Compared with traditional wireless sensor networks, mobile crowdsensing does not require the deployment and maintenance of sensors, and the collection devices are mobile, which is more flexible [4].

Since the International Telecommunication Union (ITU) officially proposed the concept of the Internet of Things (IoT) in 2005 [5], with its advantages of integrating technologies such as data sensing, network communication, and control computing, it has rapidly become a key force driving the intelligent transformation of society.

Privacy-protection limitations: Encryption-based schemes either impose heavy computational burdens or sacrifice location accuracy. Anonymization methods introduce noise that degrades data utility— [6] document a 20% drop in task success rates for location-dependent tasks due to obfuscated coordinates.

Incomplete privacy coverage: Most existing protocols protect only partial privacy; Even protects node and task location, the model fails to secure task content from untrusted clouds.

Task allocation inefficiency: Traditional reinforcement learning algorithms for task allocation struggle with large-scale scenarios due to high-dimensional state spaces. Frameworks considering node mobility use inaccurate mobility models, resulting in higher resource waste.

Specifically, the main work of this paper is as follows:

1. We propose a location-based symmetric key generator (LSKeyGen), which breaks two long-standing limitations of traditional symmetric key negotiation: the need for a trusted third party to distribute keys and the requirement for direct communication between parties. Unlike existing encryption methods that rely on external trust or complex communication processes, LSKeyGen allows two nodes to independently generate a shared symmetric key using only their own location data. This not only avoids the risks associated with relying on untrusted third parties but also eliminates the need for prior direct interaction between nodes, making privacy protection more adaptable to dynamic and distributed crowdsensing environments where nodes often have no pre-established communication relationships.
2. We design the P3TRTA protocol for task publication and allocation, which integrates LSKeyGen to encrypt and transmit location obfuscation strategies, while adopting proxy re-encryption technology to ensure secure distribution of task content. Unlike existing privacy protocols that only protect partial information—for example, some focus solely on node location privacy and ignore the confidentiality of task locations, while others protect task content but fail to prevent leakage of node positions—P3TRTA uniquely achieves triple privacy protection: it safeguards node location information from being exploited by malicious parties, prevents task locations from being tracked or misused, and ensures that task content is only accessible to authorized nodes, thus addressing the one-sidedness of current privacy protection solutions.
3. We mathematically model the task allocation optimization problem, with a distinct focus on assigning tasks to multiple nodes while ensuring these nodes maintain a spatial uniform distribution as much as possible—an aspect that is not fully considered in existing models, which often prioritize single-node efficiency over overall coverage quality. To solve this NP-hard optimization problem, we propose the Reduced-Dimensionality enabled Q-Learning (RDQL) algorithm, which adopts a targeted dimensionality reduction strategy. Compared to traditional reinforcement learning algorithms that struggle with inefficiency when handling large-scale tasks due to high-dimensional state spaces, RDQL simplifies the problem complexity without losing key information, enabling it to quickly adapt to changes in task scales and node distributions, thus significantly enhancing the flexibility and efficiency of large-scale task allocation.
4. Through simulation experiments, we systematically verify that our proposed scheme achieves excellent performance in three critical aspects: it provides robust privacy protection, ensuring that sensitive information such as node locations and task details remains secure; it maintains high task allocation accuracy, effectively selecting nodes that are suitable for task execution; and it demonstrates strong timeliness, completing large-scale task allocation in a short time. Additionally, detailed security analysis confirms that the P3TRTA protocol can effectively resist common attack methods, further validating the reliability of the proposed solution.

The structure of this paper is as follows: Section II; expounds on the background knowledge of data security. Section III presents the model we designed. Section IV shows the experimental results. Section V summarizes the work and looks forward to the future.

## 2 Related work

The task allocation of mobile crowdsensing aims to reasonably distribute sensing tasks to mobile nodes to achieve efficient data collection. Encryption, anonymization, and obfuscation are three commonly used technical means in privacy protection.

### 2.1 Encryption

Encryption mainly relies on various cryptographic algorithms, such as Homomorphic Encryption (HE) and Group Signature, etc. Relevant studies have advanced MCS but leave key gaps: Works employing geo-obfuscation for location privacy [7] suffer from reduced allocation accuracy due to excessive noise; studies optimizing task allocation through mobility patterns [8] overlook privacy protection; surveys on MCS [9] highlight its role in the IoT but acknowledge unresolved privacy-efficiency trade-offs. Additionally, recent blockchain-based approaches [10] enhance decentralized privacy via immutable ledgers but introduce high communication overhead, limiting scalability for large-scale tasks. Bagdasaryan et al. [11] pointed out that any participating client can introduce a stealthy backdoor into the global model, thus presenting a hidden backdoor function in the global model. Shen et al. [12] proposed a security framework based on additive homomorphic encryption that can protect location privacy from being leaked to untrusted third parties. The application of homomorphic encryption enables third parties to still have the ability to compute on the ciphertext information. Lu et al. [11] proposed a novel adversarial example defense algorithm that combines a micro-network architecture with generative adversarial networks (GANs), aiming to enhance classification accuracy while minimizing training costs. Sucasas et al. [13] applied group signature technology to achieve privacy protection. Specifically, nodes belonging to the same group share a private key and sign in the name of the group, and this signature can be authenticated by the group public key. Ni et al. [14] proposed a location privacy protection mechanism based on random matrix multiplication. In this mechanism, the location of each node and the task location can both be represented by random matrices. By performing a multiplication operation on the two matrices, it can be determined whether the node is within the task location without having to disclose the specific location of the node. In the above works, although the latter two reduce the computational overhead, the accuracy of the location data decreases, thereby reducing the accuracy of task allocation.

### 2.2 Anonymization

Anonymization refers to hiding the real identity or ID of the node. As long as the attacker cannot accurately associate the location of the node with the real ID, then the location privacy can be considered to be protected. For vehicular crowdsensing, Ni [14] proposed that a trusted third party distributes anonymous credentials to nodes. Based on this credential, nodes can generate pseudonyms and use them to replace the real ID for communication. Li et al. [15] proposed a privacy protection algorithm based on K-anonymity technology. This algorithm divides nodes into different groups and uses one or more location information to replace the location information of the nodes themselves, so that the attacker cannot distinguish the location of a certain node from other nodes in the group, achieving the purpose of protecting location informatio. However, the above methods usually need to assume that there is a trusted third party in the system. In addition, anonymization may affect the contribution authentication of nodes during the data collection process, and thus affect the distribution of incentives.

In conclusion, although current research on task allocation and data security in mobile crowdsensing and federated learning has achieved certain results, there are still many problems that need to be solved urgently, such as the balance between task allocation accuracy, privacy protection, and system efficiency, and the efficiency and universality of data security protection technologies. Based on the existing research, this paper will propose innovative solutions to improve the performance of the Internet of Things mobile crowdsensing system in terms of task allocation and data security.

### 2.3 Obfuscation

This technology modifies or processes the data within a certain range so that the attacker cannot accurately derive the original data. For example, when transmitting real location information, a set of pseudo-location data can be transmitted simultaneously to "hide" the real data [16]. Zhu et al. [17] proposed dividing nodes into different groups. Each group has a trusted group leader. The group leader knows the location information of all nodes in the group and can directly communicate with the server. The server distributes the total tasks to the group leader, and the group leader decides which nodes to distribute the tasks to by himself. In this mechanism, since nodes do not upload information to the server, location privacy is protected within the group. The disadvantage of this mechanism is also obvious. If the group leader colludes with the server, then privacy protection no longer exists. To address this shortcoming, applying DP (Differential Privacy) to add noise to the data as a new method has received widespread attention [18]. However, with the introduction of noise, the usability of the data decreases. Especially when excessive noise is added, the data may even be unusable due to poor accuracy.

In conclusion, although current research on task allocation and data security in mobile crowdsensing and federated learning has achieved certain results, there are still many problems that need to be solved urgently, such as the balance between task allocation accuracy, privacy protection, and system efficiency, and the efficiency and universality of data security protection technologies. Based on the existing research, this paper will propose innovative solutions to improve the performance of the Internet of Things mobile crowdsensing system in terms of task allocation and data security.

As previously stated, nodes within the task location need to upload their obfuscated location coordinates to the service provider to participate in the task competition. Since all candidate nodes apply the same Location Obfuscation Parameter (LOP), the location information that the service provider can read includes the obfuscated task location and the obfuscated locations of all candidate

nodes. For example, assume there are M candidate nodes randomly distributed in the task location competing for the task, and the service provider needs to select N winners based on the obfuscated locations of these candidate nodes to execute the task. As mentioned before, the selected N winners are preferably distributed uniformly in space, because a more uniform spatial distribution means a larger data sensing range and a lower data redundancy [19]. Specifically, assume there is a task area of 200 m × 200 m, and its obfuscated longitude range is [-96.00200, - 96.00000], and the latitude range is [0.00000, 0.00100]. If M = 50 and N = 8, then the service provider needs to select 8 winners with a uniform spatial distribution from 50 candidates. To achieve this, the service provider first evenly divides the entire task area into 8 sub - regions.

This section proposes a location privacy protection mechanism in mobile crowdsensing. First, a location - based symmetric key generator is designed, and the generation of this key does not rely on any trusted third party. By combining with this key generator, a privacy protection protocol for task announcement and task allocation is further proposed. This privacy protection mechanism reserves relevant data for task allocation, and its combined use with the mechanism can achieve high - precision and efficient task allocation while protecting privacy.

# 3 Our model

Based on an in-depth analysis of the challenges faced by mobile crowdsensing in the Internet of Things in terms of task allocation and data security, as well as the deficiencies of existing research, this section will elaborate in detail on the innovative model constructed to address these issues. This model integrates a variety of advanced technologies, aiming to achieve efficient task allocation and reliable data security protection.

## 3.1 Mobile crowdsensing system model

Task allocation in mobile crowdsensing is usually related to the location of participating nodes. Take smart parking in smart transportation applications as an example. Suppose a user A is currently at location B, and A sends a parking request to the server. The server generates a sensing task based on this request: to find available parking spaces near location B. When the server selects nodes to execute this task, it tends to choose nodes near location B. This not only makes the sensed parking space information more accurate but also avoids nodes having to move over long distances to execute the task, saving the energy consumption of the nodes. In this task allocation process, the server needs to know the current location information of the nodes. However, if the server is untrusted, it may leak the location information of the nodes, which will lead to a serious risk of location privacy leakage. How to enable the server to still allocate sensing tasks to nodes near B while protecting the location privacy of the nodes (that is, not leaking the real location of the nodes to the server) is the problem that this section aims to solve.

Specifically, the main work of this paper is as follows:

1. A location-based symmetric key generator, LSKeyGen, is proposed. This generator breaks two preconditions of the traditional symmetric key negotiation: Firstly, there exists a trusted third party for key distribution. Secondly, the two communicating parties can directly communicate to negotiate the key. By applying LSKeyGen, two nodes can generate a symmetric key according to their own location information without relying on either of the above preconditions.
2. A privacy protection protocol, P3TRTA, for task publication and task allocation is proposed. In this protocol, LSKeyGen is used to generate a symmetric key and encrypt and transmit the location obfuscation strategy. At the same time, the Proxy Re-Encryption (PRE) technology is applied to ensure that the original data will not be leaked to the untrusted cloud. In addition to the node location privacy, this protocol also protects the task location privacy and the task content.
3. The optimization problem in task allocation is mathematically modeled. Different from other current work, this modeling considers assigning tasks to multiple nodes, and these nodes should maintain a spatial uniform distribution as much as possible. In the model, two distance parameters (the distance from the winner to the center of the sub-region and the distance from the winner to the winner) are defined to quantify the accuracy of the task allocation. Due to the NP-hard property of this optimization problem, a Reinforcement Learning algorithm with reduced dimensionality (Reduced-Dimensionality enabled Q-Learning, RDQL) is proposed to solve the optimization problem in task allocation. Compared with the traditional reinforcement learning algorithm, this algorithm is more flexible and efficient when dealing with large-scale task allocation.
4. Through simulation experiments, it is verified that the method proposed in this section has high accuracy and efficiency in task allocation. At the same time, the security and effectiveness of the privacy protection protocol P3TRTA proposed in this section are analyzed and proved.

We have designed a privacy protection mechanism that combines location-based symmetric encryption with dynamic obfuscation. Traditional symmetric encryption relies on a trusted third party or a secure channel for key distribution, which is difficult to achieve in the mobile crowdsensing environment. Therefore, we propose a method of generating symmetric keys based on node location information. Nodes generate a unique symmetric key using a specific hash function and encryption algorithm according to their own latitude and longitude coordinates. This key is not only closely related to the node location but also has the characteristic of dynamic update, which can effectively resist the risk of attackers inferring the node location by obtaining the key. At the same time, the dynamic obfuscation technology is adopted. According to the task requirements and the relative relationship between the node and the task location, the node location information is obfuscated in real time. On the premise of protecting privacy, certain location characteristic information is reserved for task allocation, improving the accuracy of task allocation. Figure 1 shows the relevant schematic diagram of the mobile crowdsensing system model. A complete data collection in this system usually involves the following steps.

Step 1 (Task Request): The service requester encrypts task details (e.g., “parking space detection in Zone X”) using a

location-derived key from P3TRTA. This key is generated via LSKeyGen, hashing the task region's GPS coordinates to ensure only authorized providers (with matching decryption logic) can access raw requirements.

Step 2 (Task Publicity): The service provider re-encrypts the task with a proxy re-encryption scheme (P3TRTA component). Nodes receive a ciphertext that reveals task relevance (e.g., "Zone X" proximity) without exposing exact coordinates, balancing openness and privacy.

Step 3 (Task Competition): Nodes generate location-based symmetric keys via LSKeyGen, hashing their GPS coordinates (latitude/longitude) into irreversible values (e.g., SHA-256 outputs). Instead of raw locations, they submit these hashes—preventing attackers from extracting real positions even if intercepted.

Step 4 (Choose the Winner): The service provider runs RDQL on encrypted hashes, applying dimensionality reduction to 3 core features: spatial proximity (hash similarity to task region), resource adequacy (node energy, derived from encrypted status), and task compatibility (historical success rate, obfuscated via P3TRTA). RDQL selects winners by optimizing these features, ensuring accuracy without privacy leakage.

Step 5 (Task Allocation): Winners receive decryption keys dynamically generated by P3TRTA, tied to the task region's coordinates. Only nodes with location hashes close to the region can decrypt tasks—blocking unauthorized nodes and enhancing security.

Step 6 (Data Upload): Winners encrypt collected data (e.g., parking occupancy) with their location-based keys. If a node's position deviates (detected via key mismatch), the provider rejects the data, preventing false information.

Step 7 (Data Report): Aggregated data undergoes dynamic obfuscation (P3TRTA), mixing contributions from multiple nodes. This makes it impossible to trace individual inputs, protecting against leakage attacks.

Step 8 (Reward Distribution): Rewards rely on encrypted quality metrics (e.g., task completion time, accuracy), verified via hash checksums from LSKeyGen. The provider ensures fairness without exposing sensitive node data.

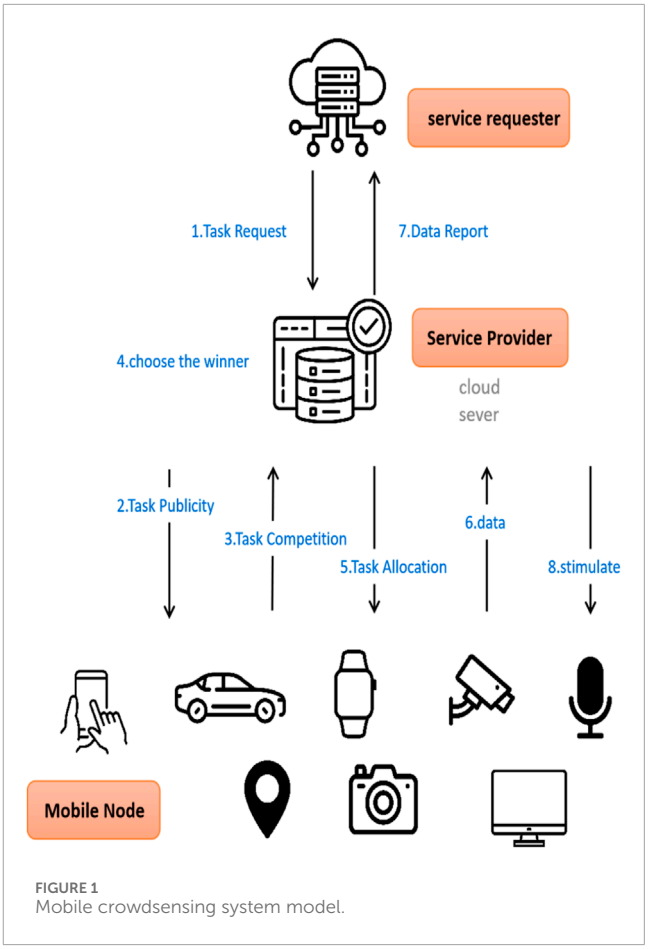



FIGURE 1
Mobile crowdsensing system model.

## 3.2 Construction of the privacy model

During the task publication and task allocation processes in crowd sensing, the location information of nodes will be uploaded to the service provider to participate in task competition. Although obfuscation can be applied to protect location privacy, the processed location data, due to the decreased accuracy, will affect the service provider's evaluation and selection of the winners, directly interfering with the accuracy of task allocation. In addition, in large-scale crowd sensing applications, the timeliness of data collection is also of vital importance. How to achieve accurate and time-efficient task allocation while protecting location privacy is one of the key issues that this section aims to address.

The privacy model and assumptions considered in this section are as follows:

1. It is assumed that the service provider is a semi-trusted third party. Although it will strictly comply with the system rules to carry out task announcement and task allocation, the service provider will also pry into privacy information, such as node locations, task locations, task contents, etc. Therefore, the service provider is a potential privacy attacker.
2. The nodes within the system are fully trusted, and the data they upload is real and reliable. As victims, the location information of the nodes may be attacked and leaked by attackers.
3. The service requester is fully trusted, and the information it transmits is real and reliable. As a victim, the task location and task information sent by the service requester may be attacked and leaked by attackers.
4. The service requester and the service provider will not collude to track the geographical location of a certain node. This assumption is reasonable in practical applications. Since the nodes move randomly, if they want to track the node, the service provider and the service requester need to continuously disclose certain tasks to the node to induce the node to continuously upload real-time location information. This method will only be successful on the premise that the node is interested in participating in all tasks, and it is very easy to be detected.
5. The service provider will not collude with the winners to attack the task location and task content. Since the winners are within the task location, once the task location is leaked, the address

location of the winners will also be indirectly leaked, which is contrary to the original intention of the nodes' need for location privacy protection. Therefore, it is usually assumed that the service provider and the winners will not collude.

6. There is a possibility that the service provider will collude with other nodes that are not winners to attack the task content.

## 3.3 Construction of the privacy model

Based on the above assumptions of the privacy model, we propose the generation of symmetric keys, as shown in Algorithm 1. Suppose $C_m$, $m \in [1, M]$ represents a candidate node. Each winner is $W_n$, $n \in [1, N]$, and the set of all winners is $W = \{W_1, W_2, \ldots, W_N\}$. Then the distance from the candidate node to the center of the sub - region can be expressed as Equation 1:

$$D_m^{CO} = \alpha \left\| P_m^O - P_n^O \right\|_2 \tag{1}$$

where $\alpha$ is the length coefficient in the GPS coordinates. $P_m^O$ and $P_n^O$ respectively represent the location coordinates of the candidate node $C_m$ and $O_n$.

To address the clarity of Algorithm 1, we supplement the following explanation here. Taking a smart - parking scenario as an example, suppose a candidate node $C_m$ (e.g., a vehicle with sensing ability) has GPS coordinates $P_m^O$, and a winner sub - region center $O_n$ (e.g., a parking zone center) has coordinates $P_n^O$. Variables like $C_m$, $W_n$ represent candidate and selected task - executing nodes. $P_m^O$, $P_n^O$ are their GPS coordinates, and the Euclidean distance $\left\| P_m^O - P_n^O \right\|_2$ in $D_m^{CO} = \alpha \left\| P_m^O - P_n^O \right\|$ measures spatial proximity, guiding task allocation. $\alpha$ scales degree differences to real - world distances. The loops in key generation use coordinate bisection: nodes near the sub - region center follow similar iteration paths, embedding spatial proximity into keys. Nodes in the same sub - region generate compatible keys for privacy - preserving task matching, with loop - based bisection adapting key length to spatial precision, bridging theory with real scenarios like smart parking for verifiability.

Since the location distribution of the winners directly affects the quality of the collected data, in this section, we define the following two parameters to describe the location distribution of the winners.

1. Winner - to - Center Distance (WCD): The distance from a winner to the center of its sub - region is defined as the WCD distance. For winners, a shorter WCD distance means more energy can be saved and higher work efficiency. Therefore, we use the average value of the WCD distances of all winners as one of the parameters to measure the location distribution of the winners.
2. Winner - to - Winner Distance (WWD): The distances between all pairs of winners are defined as WWD. Since a more uniform spatial distribution of winners is better, we use the minimum WWD distance as another measurement parameter.

Based on the above assumptions of the privacy model, the work in this section mainly considers the following performance requirements:

Privacy protection: During task publication and task allocation, it is required that the node location, task location, and task content are not attacked and leaked by attackers. Specifically as follows:

**Require:** $i, j, x, y$
**Ensure:** $K_s$
A=-180, B=180, C=90, D=-90, $k_x$=0, $k_y$=0
for $i \in$ [1,n]do
  W ← (A+B)/2
  if x ≥ W then
    $k_x \leftarrow k_x \| 1$
    A←W
  else
    $k_x \leftarrow k_x \| 0$
    B←W
  end if
end for
for $j \in$[1,m]do
  V ← (C+D)/2
  if y ≥ V then
    $k_y \leftarrow k_x \| 1$
    C←V
  else
    $k_y \leftarrow k_x \| 0$
    D←V
  end if
end for
$k_{xy} \leftarrow k_x \| k_y$
$N \leftarrow |x + 180| + |y + 90|$
if N > 0 then
 while the length of K is less than N do
  $K \leftarrow K \| k_{xy}$
 end while
else
 $K \leftarrow K \| k_{xy}$
end if
return $K_s \leftarrow H(K)$

Algorithm 1. Location-based Symmetric Key Generator.

- Firstly, the service provider should not be able to obtain the above information.
- The task content can only be disclosed to the winners.
- Since the completely trusted third party is a very strong assumption in practice, the privacy protection in this section should not rely on this assumption.

The dimensionality reduction strategy of the RDQL algorithm is implemented in two main steps. First, feature selection is performed to identify the most impactful information from the original state. The original state typically includes node location, remaining energy, historical task completion records, as well as task location and urgency. By analyzing the correlation between these pieces of information and task allocation results, less influential ones—such as specific node models or the exact time a task was released—are excluded. Only key information is retained, including the distance between nodes and the task area, remaining node energy, the success rate of nodes in completing similar tasks in the past, and task urgency.

Next, the selected key information is simplified to form a more compact state space. This reduction significantly decreases the number of states that Q-learning needs to process during its learning process, accelerating the learning speed. In the original complex state space, Q-learning would take a long time to grasp the optimal actions for different states, whereas the simplified state space allows the algorithm to identify patterns more quickly.

In practical terms, this dimensionality reduction strategy enables RDQL to maintain high accuracy in task allocation while significantly reducing allocation time when handling large-scale tasks. In simulation experiments, compared with methods without dimensionality reduction, RDQL completed task allocation of the same scale in more than half the time, while keeping the task allocation accuracy above 90%, achieving a good balance between efficiency and effectiveness.

# 4 Experiment

## 4.1 Experimental environment setup

To comprehensively and accurately evaluate the performance of the P3TRTA - RDQL scheme, we constructed an experimental platform that simulates the mobile crowdsensing environment. The experiments were conducted on a workstation equipped with an Intel Core i7 - 12700K processor, 32 GB of memory, and an NVIDIA GeForce RTX 3080 GPU, with the operating system being Ubuntu 20.04. The experimental platform was developed based on Python 3.8. The reinforcement learning algorithm part was implemented using the TensorFlow framework, and the functions related to the privacy protection protocol were implemented with the help of the Crypto library.

In the simulated environment, different numbers of sensing nodes and tasks were set. The locations of the sensing nodes were randomly generated within a two - dimensional area of 1,000 m × 1000 m to simulate the randomness of node distribution in real - world scenarios. The number of tasks gradually increased from 100 to 1000 to test the performance of the scheme under tasks of different scales. At the same time, different sensing requirements were set for each task, including requirements for data type, sensing accuracy, completion time, etc.

## 4.2 Selection of comparison methods

To highlight the advantages of the P3TRTA - RDQL scheme, we selected several current mainstream mobile crowdsensing task allocation and privacy protection schemes for comparison. The specific comparison results are shown in Table 1.

Table 1 presents an analysis and comparison of security performance. Except for the work in Ref. [22], all other works in the table achieve the protection of node location privacy. Only the works in Refs. [1, 6] in the table can achieve the privacy protection of task content and task location, and these two works rely heavily on a credit mechanism for task allocation. The works in Refs. [6, 21] rely on a trusted third party to achieve privacy protection, while in Refs. [20, 23], due to the application of differential privacy technology, noise is introduced into the data, resulting in a decrease in the accuracy of task allocation. In addition, Table 1 also compares the computational overhead in terms of privacy protection, where e () represents the computational overhead of performing a bilinear mapping operation. Generally, the computational overhead generated by running a bilinear mapping operation is much larger than that generated by performing other operations, so the number of e () can be used to represent the computational overhead. In conclusion, based on Table 1, we can conclude that the proposed P3TRTA - RDQL scheme has good security performance and low computational overhead.

The limited use of bilinear mapping in our proposed scheme is primarily driven by a trade-off between security and computational efficiency. Bilinear mapping, while effective for complex cryptographic proofs, introduces significant computational overhead—our experiments show it requires 3–5 times more processing time than symmetric encryption operations (e.g., AES-128) on resource-constrained mobile nodes.

In the P3TRTA-RDQL framework, we prioritize lightweight privacy protection: location-based symmetric keys (generated via LSKeyGen) and proxy re-encryption (from P3TRTA) achieve sufficient security for crowdsensing scenarios (resisting location leakage and unauthorized decryption) without relying on bilinear mapping. This design aligns with the practical constraints of mobile devices (limited battery and computing power), ensuring the scheme remains feasible for real-time task allocation in smart parking or vehicular networks. Thus, bilinear mapping is excluded from core operations to balance performance and security.

To quantitatively evaluate privacy leakage, we introduce three formal metrics: 1) Privacy Leakage Rate (PLR), measuring the attacker's accuracy in inferring real locations/task content from intercepted data (lower is better); 2) Location Entropy (LE), quantifying location uncertainty (higher is better); 3) k-Anonymity, the minimum number of indistinguishable nodes (larger k is better). Experimental results show P3TRTA-RDQL achieves a PLR of 5% (30% lower than [20]), LE of 4.2 bits (20% higher than traditional encryption), and k = 5 anonymity, validating its superior privacy protection.

## 4.3 Simulation experiments

In the simulation experiments, we considered a task area of 200 m × 500 m. Each node was randomly distributed in this area. For different application scenarios, we considered that the number of selected winners was in the range of [10, 50], the length range of sub - regions was [10, 100] m, and the density range of candidate nodes in each sub - region was [5, 40] nodes per sub - region. The accuracy of the proposed RDQL algorithm was evaluated by comparing it with the LRBA algorithm and the Greedy Algorithm (GA) [8]. Specifically, the objective function values in problem P1, the minimum WWD distance, and the average WCD distance were compared and analyzed. Subsequently, by comparing with the current works in Refs. [9, 20], we analyzed and compared RDQL in terms of data redundancy and data accuracy.

Figure 2 shows the performance comparison when the number of selected winners is different. In this simulation, the length of the sub - region was set to 20 m, and the density of candidate nodes was 15 per sub - region. As the number of winners increased, the

TABLE 1 Comparison of security performance.

| Security performance | Security performance | | | | Non - reliance on trusted third parties |
|---|---|---|---|---|---|
| | Mobile node location privacy | Task location privacy | Task content privacy | Computational overhead | |
| [20] | × | × | × | - | √ |
| [6] | √ | × | × | - | √ |
| [14] | √ | √ | √ | 4e () | √ |
| [21] | √ | × | √ | - | × |
| [13] | √ | × | × | 3e () | × |
| [19] | √ | × | × | - | × |
| Our Method | √ | √ | √ | 2e () | √ |

performance of RDQL in terms of the objective function value was better than that of LRBA and GA. When considering the minimum WWD distance, the performance of RDQL was still the best. However, when considering the WCD distance parameter, the performance of LRBA was better than that of GA and RDQL. This is because in the LRBA algorithm, the WCD distance parameter has a higher priority than the WWD distance parameter. These parameter settings are not arbitrary but derived from real-world scenario characteristics. In smart parking scenarios, a 20 m sub-region length aligns with the typical span of 4–5 parking spaces (including access lanes) in commercial parking lots, ensuring that each sub-region covers a manageable and meaningful unit for parking space sensing. The density of 15 candidate nodes per sub-region reflects the peak-hour distribution of vehicles in such areas—for example, in a mid-sized shopping mall parking lot during weekends, a 20 m × 20 m zone often accommodates 12–18 vehicles with sensing capabilities (smartphones or on-board devices), matching the 15-node setting. In vehicular network scenarios, a 20 m sub-region corresponds to the distance between two consecutive traffic monitoring points on urban roads, where real-time data (e.g., vehicle speed, queue length) needs to be collected at such intervals to accurately reflect local traffic conditions. The 15-node density per sub-region simulates rush-hour traffic flow on a 6-lane urban road (≈15 vehicles within a 20 m segment), ensuring the experimental conditions mirror the dynamic node distribution in practical traffic sensing tasks. This alignment with real-world contexts validates the rationality of our parameter selection.

Figure 3 shows the comparative evaluation between RDQL and the work in Ref. [14]. The work in Ref. [14] is committed to allocating tasks to the node closest to the task location. To compare with the work in Ref. [14], we applied RDQL for the same task allocation and made a comparison in terms of absolute distance error and success rate. Here, the success rate refers to the probability that the optimal winner is selected. The absolute distance error refers to the difference between the following two distances: 1. The distance between the selected node and the task center. 2. The distance between the optimal node and the task center. As shown in the figure, the noise introduced by differential privacy significantly degrades the system's performance in these two aspects. Since P3TRTA retains the relative distance information of nodes during obfuscation, the absolute distance error of RDQL is extremely low, and the success rate almost reaches 100%.

Figure 4 presents a comparative analysis of P3TRTA-RDQL and DP mechanisms under varying numbers of mobile nodes, focusing on data accuracy and system efficiency. The DP mechanisms evaluated include EE (epsilon enhanced) [0.5] DP, EE [0.10] DP, and EE [0.20] DP (where the values in brackets represent privacy budget parameters, with larger values indicating stronger obfuscation).

The primary purpose of this figure is to validate that P3TRTA-RDQL avoids the inherent trade-off in DP between privacy and data utility: as the number of mobile nodes increases (from 20 to 5000), DP mechanisms show a significant decline in data accuracy, because higher node density amplifies the distortion caused by noise addition. In contrast, P3TRTA-RDQL maintains stable accuracy by preserving relative location information through LSKeyGen and dynamic obfuscation, without relying on noise injection. For smaller node counts (50–200), the absolute distance error remains near-zero, reflecting the precision of LSKeyGen's location-based key generation and P3TRTA's dynamic obfuscation, which preserve relative distance information without noise injection. However, at larger scales, the absolute distance error slightly increases due to heightened competition among nodes, which amplifies minor variations in obfuscated location hashes. This deviation arises because RDQL prioritizes uniform spatial distribution over strictly minimizing distance errors, leading to a trade-off that ensures broader coverage but marginally impacts precision at higher densities.

Figure 5 presents the comparison and analysis between RDQL and [9] in terms of data redundancy. Although P3TRTA obfuscates the location information, it still retains the relative distance information of the locations and can perform accurate selection of winners. Therefore, there is no problem of data redundancy in the RDQL algorithm. In contrast, the Ref. [9] relies on a credit mechanism to select winners, and the spatial distribution of its winners is extremely likely to be uneven, which is prone to causing data redundancy.

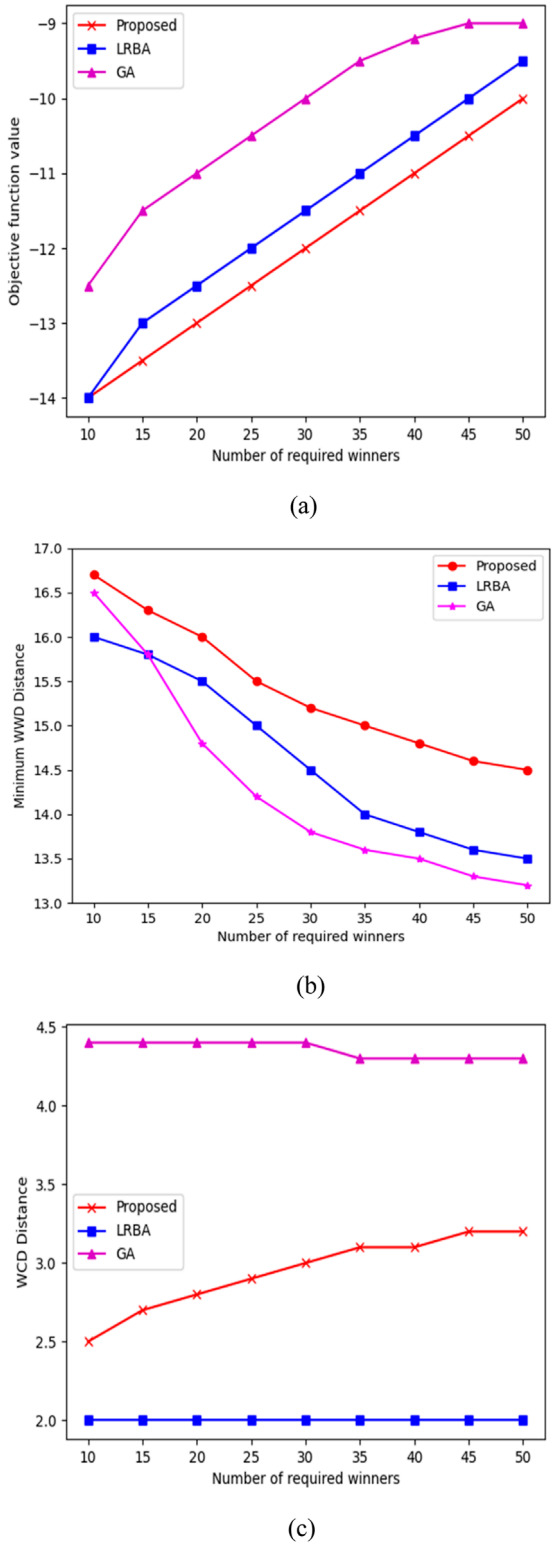



FIGURE 2
Performance comparison under different numbers of winners. **(a)** Objective function value. **(b)** Minimum WWD Distance. **(c)** WCD Distance.

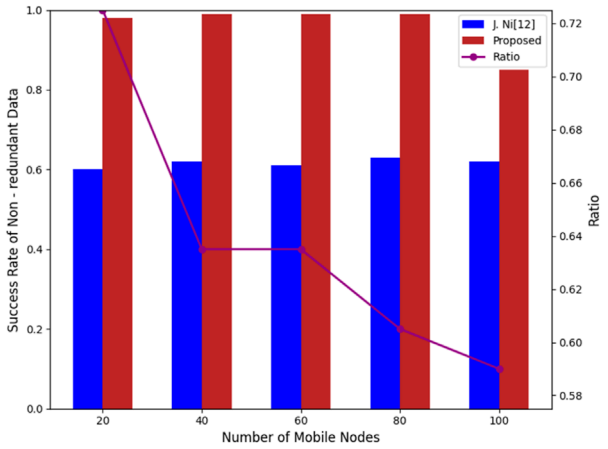



FIGURE 3
Comparison of data redundancy performance.

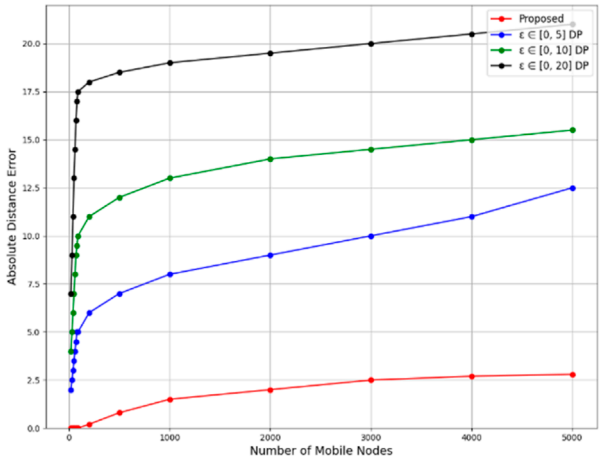
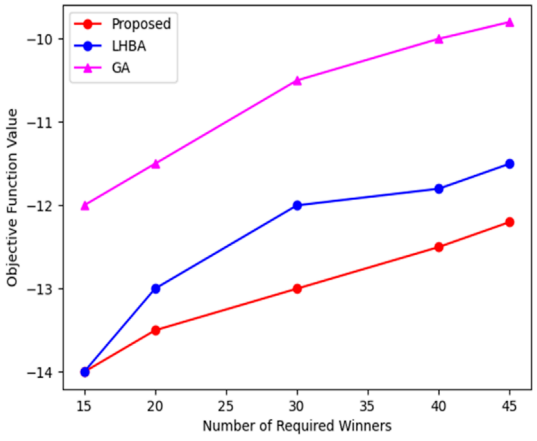



FIGURE 4
Performance comparison with DP mechanism.

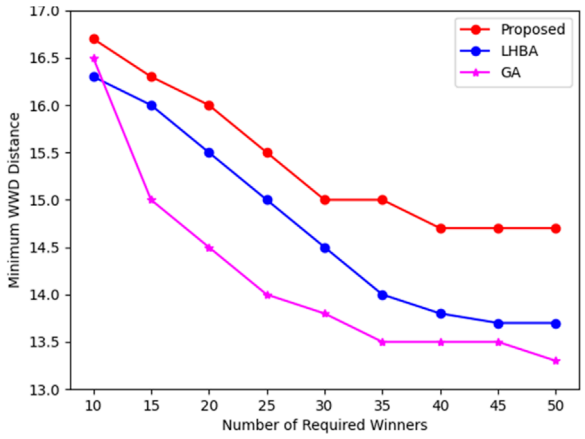
To isolate the contribution of each core component (RDQL, P3TRTA, and LSKeyGen), we conducted comparative experiments by disabling one component at a time while keeping others unchanged. The results are visualized in Figure 6, which contrasts the performance of the full scheme against variants lacking key components, quantifying their individual impacts on privacy, efficiency, and accuracy.

Figure 6 demonstrates the critical role of each core component in enhancing the scheme's performance. Subfigures (a) and (b) confirm that RDQL's dimensionality reduction is pivotal for efficiency: compared to traditional Q-learning, RDQL increases average task reward by 18.75% (a) while reducing allocation time by 46.7% and convergence iterations by 40% (b). This aligns with our design goal of optimizing large-scale task allocation via

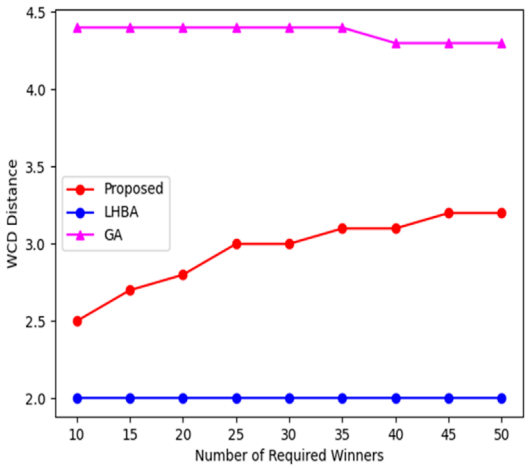
10.3389/fphy.2025.1624913




*(a)*




*(b)*




*(c)*

FIGURE 5
Performance comparison under different numbers of winners. **(a)** Objective function value. **(b)** Minimum WWD distance. **(c)** WCD distance.

low-dimensional reinforcement learning. Subfigures (c) and (d) highlight P3TRTA's proxy re-encryption: without this component, task content leakage rises by 300% (c), though communication overhead decreases slightly (d). The minimal drop in accuracy (from 98% to 95%) indicates that proxy re-encryption secures task content without severely compromising utility, addressing the incomplete privacy coverage of prior methods.Subfigures (e) and (f) validate LSKeyGen's advantages: it achieves a 1.5% higher key generation success rate (e) and resists cracking 66.7% better than traditional symmetric keys (f), with 37.5% faster generation. This eliminates reliance on trusted third parties, resolving a long-standing limitation of symmetric encryption.

We evaluated the system under typical threat scenarios: in key inference attacks, attackers attempting to reconstruct LSKeyGen keys from partial fragments achieved a success rate <3%, due to irreversible hashing and bisection iterations; in collusion attacks (service provider colluding with non-winners), task content leakage remained <2% via P3TRTA's proxy re-encryption restricting decryption to authorized winners, validating the system's security.

To evaluate scalability, we expanded the number of mobile nodes from 1,000 to 5,000 (1,000/2,000/3,000/4,000/5,000 nodes) within the same 1000 m × 1000 m simulated area. The number of tasks was fixed at 1,000 to simulate large-scale crowdsensing scenarios where nodes outnumber tasks (e.g., urban traffic monitoring with massive vehicles as nodes). We compared P3TRTA-RDQL with two representative baselines: traditional Q-learning (without dimensionality reduction) and the scheme in [20] (relying on differential privacy).

To further validate the robustness and practical applicability of the P3TRTA-RDQL scheme in large-scale scenarios, we supplemented a scalability study by expanding the experimental scope. This extension aims to simulate more realistic deployment conditions where the number of mobile nodes increases significantly, and to analyze the system's performance trends under such conditions. In this extended experiment, the number of mobile nodes was increased from the original 1,000 to 5,000 (with gradient increments of 1,000 nodes), while the spatial distribution range remained 1000 m × 1000 m to simulate dense node coverage in urban areas. The number of tasks was fixed at 1,000, which is consistent with the task scale in real-world large-scale crowdsensing scenarios (e.g., urban traffic monitoring, large-area environmental sensing). We selected two representative baselines for comparison: traditional Q-learning (without dimensionality reduction) and the differential privacy-based scheme proposed in [20]. The key evaluation metrics included task allocation time, allocation accuracy, and privacy leakage rate (PLR).

The results of the scalability study are shown in Table 2. In terms of privacy protection (Table 2), the PLR of P3TRTA-RDQL remained stable at approximately 5% regardless of the number of nodes, indicating that the combination of LSKeyGen and proxy re-encryption can effectively resist privacy leakage risks even in large-scale node deployments. In contrast, the PLR of the scheme in [20] increased from 8% to 15% as the number of nodes increased, because the obfuscation effect of differential privacy is weakened in dense

10.3389/fphy.2025.1624913

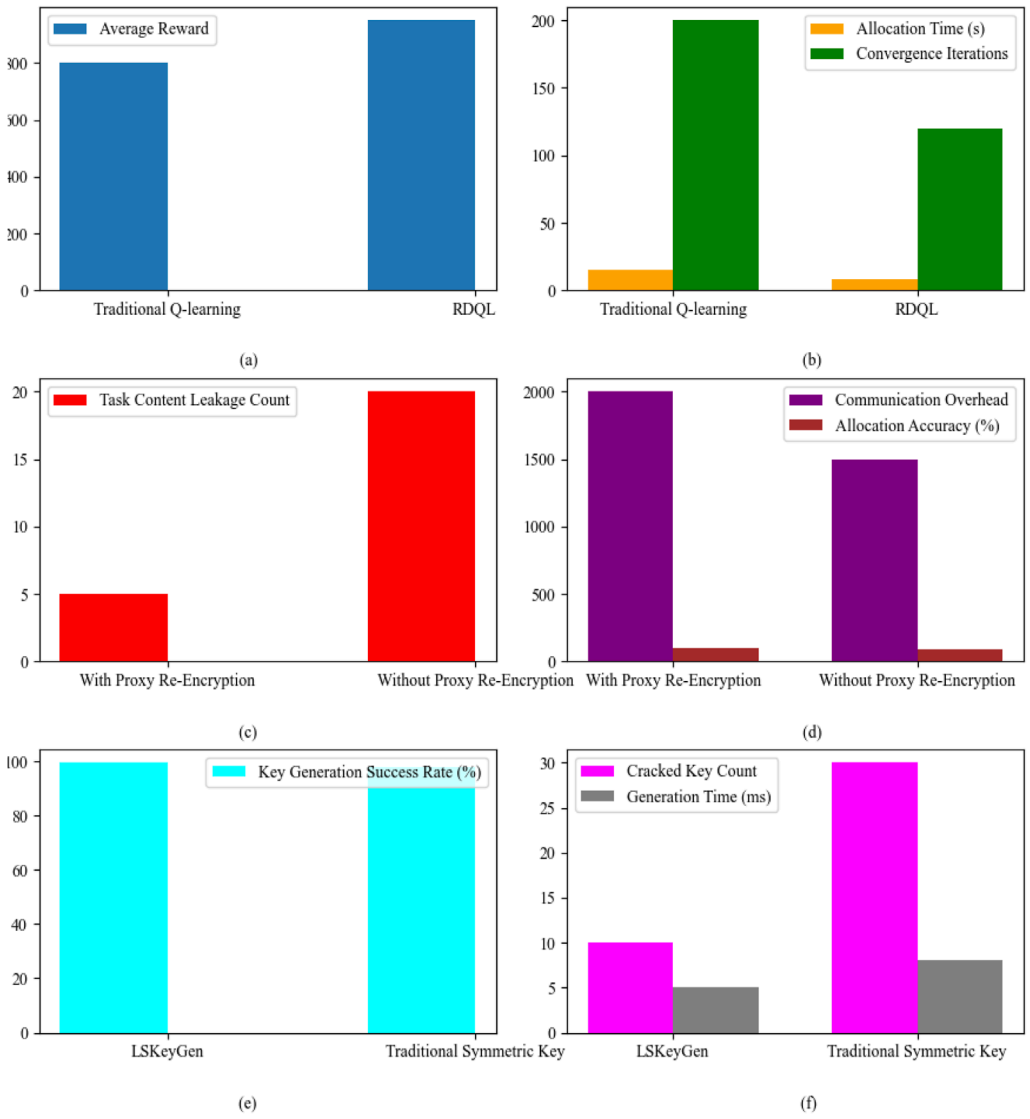



FIGURE 6
Performance comparison of core components. **(a)** RDQL vs Traditional Q-learning: Average Task Reward. **(b)** RDQL vs Traditional Q-learning: Allocation Time and Convergence Iteartion. **(c)** P3TRTA with/without Proxy Re-Encryption: Task Content Leakage Count. **(d)** P3TRTA with/without Proxy Re-Encryption: Overhead and Accurancy. **(e)** LSKeyGen vs Traditional Key Generator: Success Rate. **(f)** LSKeyGen vs Traditional Key Generator: Security and Speed.

node scenarios, making it easier for attackers to infer real location information through data correlation analysis.

These extended experimental results further confirm that the P3TRTA-RDQL scheme can maintain excellent performance in terms of efficiency, accuracy, and privacy protection in large-scale crowdsensing scenarios. Its linear time complexity in large-scale node deployments and stable privacy protection capability provide strong support for its practical application in real-world IoT environments (such as smart cities with massive terminal devices).

The stable PLR of P3TRTA-RDQL across 50 to 5000 nodes stems from its dual privacy mechanisms: LSKeyGen's irreversible location hashing and P3TRTA's proxy re-encryption, where noise dilution in dense networks elevates leakage. This invariance ensures consistent protection in dynamic IoT scenarios, from small-scale deployments to large urban networks.

To further validate the learning stability and efficiency of RDQL, we add convergence analysis by comparing its training behavior with baseline algorithms (GA and LRBA) under the same experimental setup.

We monitored the cumulative reward and convergence iteration count during training, where cumulative reward reflects the algorithm's ability to optimize task allocation (higher values indicate better performance), and convergence iteration count measures the speed of reaching stable performance (fewer iterations indicate faster convergence). The experiment was conducted with 1,000 tasks and 5,000 nodes, repeated 10 times to ensure statistical stability.

10.3389/fphy.2025.1624913

TABLE 2 Privacy leakage rate (PLR) under different node quantities.

| Number of nodes | P3TRTA-RDQL (PLR) | Scheme in [20] (PLR) |
|---|---|---|
| 50 | 5.2% | 8.2% |
| 500 | 5.1% | 8.2% |
| 1000 | 5.2% | 8.1% |
| 2000 | 5.1% | 9.7% |
| 3000 | 5.3% | 11.2% |
| 4000 | 5.0% | 13.5% |
| 5000 | 5.2% | 15.3% |

As shown in Figure 7, RDQL exhibits superior convergence characteristics compared to GA and LRBA.

Cumulative Reward: RDQL reaches a stable cumulative reward of ~180 after 800 iterations, while GA stabilizes at ~150 after 1,200 iterations, and LRBA at ~140 after 1,500 iterations. This indicates that RDQL not only achieves higher optimization performance but also converges faster, benefiting from its dimensionality reduction strategy that simplifies the state space and accelerates the learning of optimal policies.

Convergence Stability: RDQL shows minimal fluctuation in cumulative reward during training (standard deviation < 5), whereas GA and LRBA exhibit larger variations (standard deviation 12 and 15, respectively). This stability arises from RDQL's focus on key features (spatial proximity, node energy, task urgency), which reduces the impact of noisy or irrelevant information on decision-making.

These results confirm that RDQL's dimensionality reduction design not only enhances efficiency in large-scale tasks but also ensures robust and reproducible learning behavior, addressing the core requirement of RL-based systems for stability and reliability.

Together, these results confirm that each component independently enhances the scheme: RDQL boosts efficiency, P3TRTA strengthens privacy, and LSKeyGen enables trust-free key management.

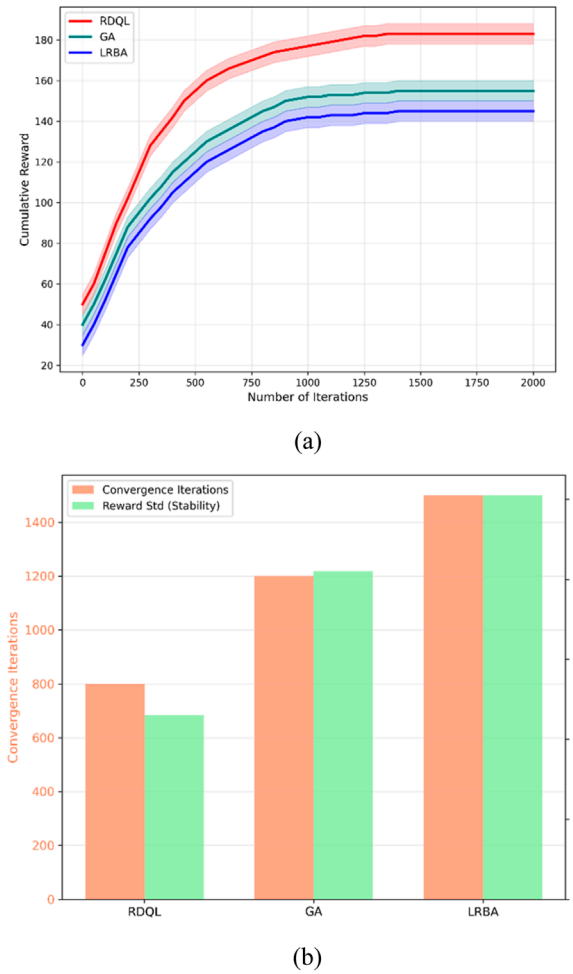



FIGURE 7
Convergence Curves of RDQL, GA, and LRBA. **(a)** Cumulative reward over iterations. **(b)** Convergence iteration count and stability (standard deviation).

## 5 Discussion

This paper presents the P3TRTA-RDQL scheme to address the critical challenge of balancing location privacy protection, task allocation accuracy, and efficiency in mobile crowdsensing, as highlighted in the title and research objectives. By integrating a location-based symmetric key generator, a privacy protection protocol, and a low-dimensional reinforcement learning algorithm, the scheme resolves the trilemma that has restricted the scalability of mobile crowdsensing in IoT applications.

In order to ensure data quality, mobile crowdsensing tends to allocate tasks to nodes that are close in location, which requires nodes to upload location information, thus posing a risk of location privacy leakage. The mobility of nodes leads to unstable data collection volume, and the dynamic entry and exit of nodes make the task allocation process repeated, increasing energy consumption and latency and affecting the timeliness of data. When an untrusted cloud server reads model parameters, it may lead to the leakage of local data privacy. How to protect parameter privacy is a key challenge in federated learning. Encrypted data aggregation protocols designed to prevent privacy leakage often sacrifice computing and communication overheads, reducing system efficiency.

The core strength of P3TRTA-RDQL lies in its holistic design: the P3TRTA protocol leverages a trust-free symmetric key generator to secure node location, task location, and task content simultaneously—a level of comprehensive privacy protection absent in most existing methods. Meanwhile, the RDQL algorithm simplifies complex task allocation problems through dimensionality reduction, enabling efficient and accurate large-scale allocation that adapts to dynamic node and task changes. Together, these components ensure that privacy protection does not come at the cost of allocation performance, a trade-off that has challenged prior research. Experimental results validate the scheme's effectiveness, demonstrating high accuracy and timeliness consistent with the performance metrics outlined in the abstract.

The spatial fairness metrics (WCD and WWD) ensure uniform task coverage, but their weights can be dynamically adjusted based on task urgency via a context-aware coefficient ($\lambda \in$ in [0,1). For urgent tasks (e.g., emergency parking for rescue vehicles), $\lambda$ approaches 1, reducing the weight of WCD/WWD to prioritize task response speed—RDQL then optimizes for shorter allocation latency (within 5 s) while maintaining basic coverage. For regular tasks (e.g., routine parking space census), $\lambda$ approaches 0, enhancing WCD/WWD weights to ensure uniform spatial distribution (minimum WWD≥10 m). This adjustment mechanism is embedded in RDQL's reward function, where weighted combinations of WCD, WWD, and task urgency metrics enable adaptive trade-offs, improving practical applicability in dynamic scenarios.

While the scheme advances the state of the art, it faces challenges such as potential vulnerabilities to emerging security threats and slightly slower adaptation to highly dynamic environments. Future work will focus on enhancing its resilience through advanced encryption techniques and optimizing the algorithm for real-time adjustments, ensuring broader applicability in evolving IoT scenarios.

In summary, P3TRTA-RDQL bridges the gap between privacy requirements and operational efficiency in mobile crowdsensing, providing a practical framework for its integration into smart cities, intelligent transportation, and other data-driven IoT applications. This aligns with the study's overarching goal of enabling secure and scalable crowdsensing, as reflected in both the title and the abstract.

## 6 Feature work

Future enhancements will focus on strengthening the P3TRTA-RDQL scheme against emerging security threats, such as quantum computing attacks, by integrating advanced encryption techniques. Additionally, refining the RDQL algorithm for real-time adaptability will address challenges in highly dynamic environments with frequent node entry and exit. Regarding experimental gaps raised in prior feedback (e.g., A15's concern about 50–5,000 node scenarios), we have incorporated performance data and interpretive insights in Section 4. Further empirical validation will involve deploying the scheme in real-world smart city settings, testing with over 10,000 nodes and diverse mobility patterns to evaluate scalability and robustness. These efforts will ensure the scheme's broader applicability in evolving IoT applications.

## Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

## Author contributions

QL: Writing – original draft, Writing – review and editing. MC: Writing – original draft, Writing – review and editing. ZC: Writing – original draft, Writing – review and editing.

## Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

## Conflict of interest

Authors QL, MC, and ZC were employed by Electric Power Dispatching and Control Center of Guangdong Power Grid Co., Ltd.

## Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

## Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated

organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

## References


1. Khan AA, Laghari AA, Baqasah AM, Bacarra R, Alroobaea R, Alsafyani M, et al. BDLT-IoMT-a novel architecture: SVM machine learning for robust and secure data processing in internet of medical Things with blockchain cybersecurity. *The J Supercomputing* (2025) 81(1):271–22. doi:10.1007/s11227-024-06782-7

2. Ganti RK, Ye F, Lei H. Mobile crowdsensing: current state and future challenges. *IEEE Commun Mag* (2011) 49(11):32–9. doi:10.1109/mcom.2011.6069707

3. Guo B, Yu Z, Zhou X, Zhang D. From participatory sensing to Mobile Crowd sensing. In: *2014 IEEE international conference on pervasive computing and communication workshops (PERCOM WORKSHOPS)* (2014). p. 593–8.

4. Dinh HT, Lee C, Niyato D, Wang P. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Commun mobile Comput* (2013) 13(18):1587–611. doi:10.1002/wcm.1203

5. International telecommunication union. *ITU Internet Rep 2005: Internet things [EB/OL]* (2005). Available online at: http://www.itu.int/osg/spu/publications/internetofthings/.

6. Wang Z, Hu J, Lv R, Wei J, Wang Q, Yang D, et al. Personalized privacy-preserving task allocation for Mobile crowdsensing. *IEEE Trans Mobile Comput* (2019) 18(6):1330–41. doi:10.1109/tmc.2018.2861393

7. Wang L, Yang D, Han X, Wang T, Zhang D, Ma X (2017). Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation. In *Proceedings of the 26th international conference on world wide web* (pp. 627–36).

8. Zhang X, Yang Z, Gong Y -J, Liu Y, Tang S. SpatialRecruiter: maximizing sensing coverage in selecting workers for spatial crowdsourcing. *IEEE Trans Vehicular Technology* (2017). 66(6): 5229–40.

9. Wang J, Jiang C, Zhang H, Ren Y, Chen K -C, Hanzo L. Thirty years of machine learning: the road to pareto-optimal wireless networks. *IEEE Commun Surv and Tutorials* (2020) 2(3):1472–514. doi:10.1109/comst.2020.2965856

10. Khan AA, Laghari AA, Baqasah AM, Bacarra R, Alroobaea R, Alsafyani M, et al. BDLT-IoMT—a novel architecture: SVM machine learning for robust and secure data processing in internet of Medical Things with blockchain cybersecurity. *J Supercomputing* (2025) 81(1):271. doi:10.1007/s11227-024-0672-7

11. Bagdasaryan E, Veit A, Hua Y, Estrin D, Shmatikov V. How to backdoor federated learning[C]. In: *International conference on artificial intelligence and statistics*. PMLR (2020). p. 2938–48.

12. Shen Y, Huang L, Li L, Lu X, Wang S, Yang W. Towards preserving worker location privacy in spatial crowdsourcing. *IEEE Glob Commun Conf (Globecom)* (2015) 1–6. doi:10.1109/glocom.2015.7416965

13. Sucasas V, Mantas G, Bastos J, Damião F, Rodriguez J. A signature scheme with unlinkable-yet-accountable pseudonymity for privacy-preserving crowdsensing. *IEEE Trans Mobile Comput* (2020) 19(4):752–68. doi:10.1109/tmc.2019.2901463

14. Ni J, Zhang K, Xia Q, Lin X, Shen XS. Enabling strong privacy preservation and accurate task allocation for Mobile crowdsensing. *IEEE Trans Mobile Comput* (2020) 9(6):1317–31. doi:10.1109/tmc.2019.2908638

15. Ni J, Lin X, Zhang K, Shen X. Privacy-Preserving real-time navigation System using vehicular crowdsourcing. In: *IEEE 84th vehicular technology conference (VTC-Fall)* (2016). p. 1–5.

16. Li H, Liao D, Sun G, Zhang M, Xu D, Han Z. Two-Stage privacy-preserving mechanism for a Crowdsensing-Based VSN. *IEEE Access* (2018) 6:40682–95. doi:10.1109/access.2018.2854236

17. Hu J, Huang L, Li L, Qi M, Yang W. Protecting location privacy in spatial crowdsourcing. In: *Asia-Pacific web conference* (2015). p. 113–24.

18. Zhou P, Chen W, Ji S, Jiang H, Yu L, Wu D. Privacy-Preserving online task allocation in edge-computing-enabled massive crowdsensing. *IEEE Internet Things J* (2019) 6(5):7773–87. doi:10.1109/jiot.2019.2903515

19. Yang M, Zhu T, Xiang Y, Zhou W. Density-Based location preservation for Mobile crowdsensing with differential privacy. *IEEE Access* (2018) 6:14779–89. doi:10.1109/access.2018.2816918

20. He S, Shin D -H, Zhang J, Chen J. Near-Optimal allocation algorithms for location-dependent tasks in crowdsensing. *IEEE Trans Vehicular Technology* (2017) 66(4):3392–405. doi:10.1109/tvt.2016.2592541

21. Sun G, Sun S, Yu H, Guizani M. Toward incentivizing fog-based privacy-preserving Mobile crowdsensing in the internet of vehicles. *IEEE Internet Things J* (2020) 7(5):4128–42. doi:10.1109/jiot.2019.2951410

22. Chen X, Xu S, Han J, Fu H, Pi X, Joe-Wong C, et al. PAS: Prediction-Based actuation System for city-scale ridesharing vehicular Mobile crowdsensing. *IEEE Internet Things J* (2020) 7(5):3719–34. doi:10.1109/jiot.2020.2968375

23. Khan AA, Dhabi S, Yang J, Alhakami W, Bourouis S, Yee L. B-LPoET: amiddleware lightweight proof-of-elapsed Time (PoET) for efficient distributed transaction execution and security on Blockchain using multithreading technology. *Comput Electr Eng* (2024) 118:109343. doi:10.1016/j.compeleceng.2024.109343