Abstract—Spatial crowdsourcing engages individuals to collect and process social, environmental and other information with spatio-temporal features, making the data collection and analysis efficient, scalable and smart. The quality of task fulfillment strongly depends on the set of recruited workers. The more suitable workers are engaged, the better results may be obtained, meanwhile, the more privacy of workers will be disclosed. In this paper, we propose LATE, a novel location privacy-aware task recommendation framework in spatial crowdsourcing, which enables spatial crowdsourcing servers (SC-servers) to recommend spatial tasks released by customers to the workers in geocast regions. Based on Lagrange Interpolating Polynomials, we design a privacy-preserving location matching mechanism to allow the SC-server to determine whether a worker is in geocast region of a spatial task or not without any knowledge about the task's geocast region and the worker's location. In addition, the spatial tasks and crowdsourcing reports are protected against privacy leakage for both customers and workers. Finally, we discuss the security properties of LATE and demonstrate its efficiency on computation and communication.

I. INTRODUCTION

Spatial crowdsourcing [1] is a compelling paradigm that engages individuals in collecting, processing and analyzing data about environmental phenomena, social events, and other spatio-temporal information. With spatial crowdsourcing, customers outsource their spatial tasks to a group of workers, i.e., mobile users that collect data from specific regions using their devices [2], to improve the quality of task accomplishment and reduce the cost on data analysis. Typically, a spatial crowdsourcing server (SC-server) acts as a broker to recruit workers for task fulfillment, and the workers participate in spatial crowdsourcing activities voluntarily or motivated by benefits. As this human-centric problem-solving paradigm is highly flexible, it can significantly reduce the cost and shorten the time on task accomplishment. Furthermore, with human intelligence, spatial crowdsourcing can improve the quality of task completion, such as translation and labelling. Currently, spatial crowdsourcing supports numerous applications in domains, e.g., journalism, environmental sensing, crisis response and urban planning.

Unlike traditional companies, in which the tasks are accomplished by fixed employees, spatial crowdsourcing recruits a set of workers from Internet to perform spatial tasks. Thus, this paradigm is feasible only if workers and tasks are matched effectively on both time and locations [3]. For example, to measure the traffic congestion in downtown Toronto at the morning peak hours, the SC-server should assign tasks to the workers driving on the roads in downtown Toronto at those hours. Otherwise, the workers have to pay extra costs on travel and time to reach the required locations for task performing, which may discourage workers participating in spatial crowdsourcing activities. Therefore, it is necessary for the SC-server to take into account the workers’ locations in spatial task allocation. However, the SC-server may not be fully trusted, and the disclosure of individual locations has serious privacy implications from the perspective of workers. It is possible for attackers to predict the trajectory and living habit of a specific worker [4]. Protecting location privacy is essential in spatial crowdsourcing, as the workers may refuse to engage in spatial tasks if their privacy is invaded. To preserve worker’s location privacy on task recommendation, many solutions have been proposed based on mix network, anonymity techniques and location differential privacy in spatial crowdsourcing. Unfortunately, these techniques have their inherit drawbacks. Specifically, mix network is built on the assumption that at least one of the network nodes is not compromised; the anonymity techniques, such as pseudonyms, blind signatures and group signatures, require either pseudonym management or complex zero-knowledge proof to protect the workers’ identities; and the location differential privacy sacrifices the accuracy of location matching to ensure the location privacy [5]. Therefore, exploring new approaches to preserve the location privacy for workers still deserves to pay more efforts.

Even the location privacy leakage in task allocation is prevented, the crowdsourcing reports may still expose location information about workers [6]. For instances, from photos, videos and other spatial data, the attackers can know the places that these photos and videos are taken and spatial data are collected, and thereby learn the locations of data sources. As a result, the location privacy of workers is violated. Moreover, the exposure of crowdsourcing reports might leak other personal information about workers, such as identities, occupations, references, home addresses, social relations, health status, and political ideology, which may cause plenty of troubles in daily life, e.g., malicious advertisements and harassing phone calls, even result in economic loss. In addition, the spatial tasks would expose the points of interest of customers and their intentions to release these tasks. In short, protecting crowdsourcing reports and spatial tasks are quite vital for both customers and workers in spatial crowdsourcing.
In this paper, we propose a Location privacy-Aware Task rEcommendation framework, called LATE, to protect workers’ locations during task recommendation in spatial crowdsourcing. By leveraging Lagrange Interpolating Polynomials, we achieve the privacy-preserving matching between the locations of workers and the geocast areas of spatial tasks in LATE. Specifically, the main contributions of our paper are as follows:

- A privacy-preserving location matching mechanism is designed from Lagrange Interpolating Polynomials. The geocast region of the spatial task is encrypted using a temperate public key and a searchable tag is generated from the worker’s location and the corresponding temperate secret key. Having the ciphertext and tag, anyone can test whether the worker’s location is one of the places in the geocast region.

- By leveraging the designed privacy-preserving location matching mechanism, LATE achieves secure spatial task recommendation with location privacy preservation for workers. The SC-server cannot know the geocast regions of spatial tasks and the geographic locations of workers, but enable to determine whether the workers are located in the geocast regions of spatial tasks. Thereby, the SC-server can recommend the spatial tasks to the workers for fulfillment. In addition, we utilize proxy re-encryption to encrypt the spatial tasks and crowdsourcing reports to prevent privacy leakage for both customers and workers.

- We prove the security of LATE to show that attackers can learn nothing about the locations of workers and the geocast areas of spatial tasks, and demonstrate that the LATE is efficient and practical in terms of computational and communication overhead.

The remainder of the paper is organised as follows. We first define the system model, security threats and design goals in section II. Then, we describe the LATE framework in section III and discuss its security in section IV, followed by the performance evaluation in section V. Finally, we review related work in section VI and conclude our paper in section VII.

II. PROBLEM STATEMENT

In this section, we define system models and security threats, and identify the design goals.

A. System Model

The spatial crowdsourcing system provides a people-centric approach to customers for data collection and analysis. As shown in Fig. 1, the architecture consists of four entities: a SC-server, a trust management server, customers and workers.

**SC-Server:** The SC-server offers the spatial crowdsourcing service to customers. It has sufficient storage space, computational capability and communication bandwidth. It is responsible to receive spatial tasks from customers, recommend tasks to workers and collect the crowdsourcing reports to fulfill the tasks for customers.

**Trust Management Server:** The trust management server aims to manage the trust levels of all workers. It keeps the trust levels or reputations of workers and enables the SC-server to check whether the participating worker is honest enough to perform the recommended spatial tasks.

**Customers:** The customers can be individuals, corporations or organizations. They have some spatial tasks to accomplish, but they are unwilling to perform by themselves, and thereby they outsource their tasks to the SC-server and recruit workers to fulfill for them.

**Workers:** Each worker has the devices to perform the spatial tasks, e.g., smart phones, tablets, vehicles, computers and other items with sensors, computational units and storage spaces. These devices are carried by their owners wherever they go and whatever they do. The workers also make sure the sufficient power on devices to support their functions. With the devices, the workers can participate in spatial tasks by collecting data from environment, analyze data, process images and upload the crowdsourcing reports to the SC server.

B. Security Threats

The spatial crowdsourcing system is confronted with serious security threats from both external and internal attackers. Specifically, the external attackers, e.g., eavesdroppers and hackers, wiretap on wireless communication channels to capture the messages exchanged between workers and the SC-server, and hack the SC-server or devices to obtain the administration rights. The internal adversaries include the SC-server and workers. The SC-server is honest to offer spatial crowdsourcing service to customers, but it may be curious on the workers. It may strive to know the spatio-temporal probability distribution of a specific worker and other sensitive information about workers and customers, e.g., preference, social relation, political affiliation and purchase intention, from the maintained information, including spatial tasks and crowdsourcing reports. The workers also try to learn sensitive information about the customers and the other workers. In particular, they are willing to know the other workers participating in the same tasks, and learn more knowledge about customers to reach their expectancy. The geographic locations are extracted from Google Maps or GPS trusted chips on devices, such that modifying the locations of workers is
infeasible. The trust management server is protected by trusted components and it is fully trusted by customers, workers and the SC-server. The customers are honest as well, since, as the beneficiaries, they have no incentive to disrupt the spatial crowdsourcing service.

C. Design Goals

To enable location privacy-aware task recommendation under the aforementioned system model and resist security threats, the LATE should achieve the following design goals:

- **Task Recommendation**: The spatial tasks should be recommended to the workers located in the geocast regions of spatial tasks to reduce the travel cost and time on task fulfillment, and the other workers outside the geocast regions should not learn any knowledge about the spatial tasks, even they can obtain the encrypted spatial tasks.
- **Location Privacy Preservation**: The geographic locations of workers and the geocast regions of tasks are protected against malicious attackers and curious entities. The SC-server or a worker only aware whether the geographical position is in the geocast area or not.
- **Data Confidentiality**: The crowdsourcing reports can only be accessed by the delegated customers, such that the privacy of workers would not be exposed to others.

### III. THE LATE PROTOCOL

In this section, we propose the LATE consisting of four phases, Service Setup, Task Releasing, Task Recommendation and Task Fulfillment. We first review the preliminaries, which are the basis of the LATE.

#### A. Preliminaries

**Lagrange Interpolating Polynomial Theorem** [8]: Let $F(x) = \sum_{i=0}^{n} y_i f_i(x) = \sum_{i=0}^{n-1} a_i x^i$ be a polynomial of degree $n-1 \geq 0$ that passes through $n$ points $(x_1, y_1), \ldots, (x_n, y_n)$ where for each $i$,

$$f_i(x) = \prod_{1 \leq j < i \leq n} \frac{x - x_j}{x_i - x_j}$$

$$= \begin{cases} 1, & x = x_i, \\ 0, & x \in \{x_1, \ldots, x_n\} \setminus \{x_i\}. \end{cases}$$

**Bilinear Pairing**: Suppose $G$ be a cyclic additive group with a prime order $q$, and $G_T$ be a cyclic multiplicative group of the same order $q$. $P$ is a generator of $G$. The map $\hat{e}: G \times G \rightarrow G_T$ is an admissible bilinear pairing [9] if the following conditions hold:

1. **Bilinearity**: for all $a, b \in \mathbb{Z}_q^*$, $\hat{e}(P, P)^{ab} = \hat{e}(aP, bP)$.
2. **Non-degeneracy**: $\hat{e}(P, P) \neq 1_{G_T}$.
3. **Computability**: there exists an efficient algorithm to compute $\hat{e}$.

**Complexity assumption**: The intractable mathematical problem and complexity assumption used are as follows.

**Co-Decisional Bilinear Diffie-Hellman (Co-DBDH) problem** [10]: Given $\langle P, aP, bP, Q, Z \rangle$ for some $a, b \in \mathbb{Z}_q^*$, $P, Q \in G$ and $Z \in G_T$, output “1” if $Z = \hat{e}(P, Q)^{ab}$ and “0”, otherwise.

**Definition 1.** An algorithm $B$ with an output $\beta \in \{0,1\}$ has an advantage $\varepsilon$ in solving the Co-DBDH problem that $\Pr[B(P, aP, bP, Q, \hat{e}(P, Q)^{ab}) = 1] - \Pr[B(P, aP, bP, Q, Z) = 1] \geq \varepsilon$ where $a, b$ are randomly chosen from $\mathbb{Z}_q^*$ and $Z$ is a random element in $G_T$.

The $(\tau, \varepsilon)$-Co-DBDH assumption holds if no polynomial-time algorithm has an advantage $\varepsilon$ within running time $\tau$ in solving the Co-DBDH problem.

#### B. The Detailed LATE

To achieve privacy-aware task recommendation, we uniquely utilize the Lagrange Interpolating Polynomials to design the location matching in LATE. We allow the SC-server to check whether the locations of workers are in the geocast areas of spatial tasks, without exposing the locations of workers and the geocast areas of spatial tasks. Intuitively, the public key encryption with keyword search scheme [7] is utilized to achieve the privacy-preserving location matching for workers. Specifically, the geocast areas of spatial tasks are represented as a vector $L = \{l_1, l_2, \ldots, l_n\}$, and the location of a worker $U_l$ is supported to be $l$. The SC-server can learn whether $l \in L$ with no knowledge about $l$ and $L$. Thus, the SC-server can recommend the task with $L$ to $U_l$, if $l \in L$. The detailed construction of LATE is described below.

1) **Service Setup**: The SC-server bootstraps the whole service and setups the public parameters. It chooses a security parameter $k$, which ensures the security level of the system and determines the prime order $q$ of bilinear groups. In general, $k = 160$ or $256$. Let $G$ be an additive cyclic group with a generator $P$ and $G_T$ be a multiplicative cyclic group equipped with $q$. $\hat{e}$ is a bilinear map as $\hat{e}: G \times G \rightarrow G_T$. The SC-server picks a random $Q \in G$ and two collision resistant hash functions $H_1: \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ and $H_2: \{0,1\}^* \rightarrow G$. $C = E_{AES}(K, M)$ and $M = D_{AES}(K, C)$ are the encryption and decryption algorithms of AES. The public parameter is $g p = \{q, G, G_T, \hat{e}, P, Q, H_1, H_2\}$. Besides, the SC-server initializes the service geographic region for customers by defining the points of interest in the regions, such as shopping malls, plazas, museums and buildings.

A worker $U_l$ is required to generate a public-private key pair by picking a random $s \in \mathbb{Z}_q^*$ to compute $P_{pub} = sP$. The public key is $P_{pub}$, while the private key is $s$. The certificate authority (CA) in public key infrastructure (PKI) issues and signs the certificate $cert_a$ to $U_l$, which includes series number, $P_{pub}$, signature algorithm, and signature, etc. The certificate is allowed the public to check the validity of $U_l$’s public key, and $U_l$’s secret key is kept on a Memory Protection Unit (MPU) to restrict publicly access.

A customer $C$ randomly chooses $v \in \mathbb{Z}_q^*$ as the secret key and computes the public key as $V_{pub} = vP$. The CA also generates and issues a certificate $cert_c$ to $C$. $C$ keeps its secret key on an MPU to restrict publicly access.

The trust management server (TMS) initializes its service of trust management for workers. It maintains a list $L_M$ to record the trust level of each worker. TMS randomly picks $t \in \mathbb{Z}_q^*$ as the secret key and calculates the public key as $T_{pub} = tP$. The intractable mathematical problem and complexity assumption used are as follows.
The CA also generates and issues a certificate \( cert_t \) to TMS. TMS keeps its secret key on the MPU.

2) Task Releasing: When the customer \( C \) is willing to use spatial crowdsourcing, it generates a spatial task \( ST = (Cont, Expt, L) \), which indicate the content (what to sense), the expiration time (when to sense) and the geocast area (where to sense). Other attributes (e.g., reporting intervals, benefits, reporting periods) can be illustrated in \( Cont \). \( C \) picks a random number \( num \) as the identifier. The geocast area \( L \), denoted as \( L = \{l_1, l_2, \cdots, l_n\} \), is a set of points of interest from which \( C \) needs to collect and analyze data. To prevent the exposure of geocast area \( L = \{l_1, l_2, \cdots, l_n\} \), \( C \) generates a series of encrypted points of interest in the following way:

1) Pick a random value \( k \in Z_p^* \) as the temperate secret key and compute the corresponding temperate public key \( K_{pub} = kP \).
2) Pick a random value \( \gamma \in Z_p^* \) to compute \( C_1 = \gamma P \) and \( h = H_1(num, cert_t, e(K_{pub}, Q)\gamma) \).
3) For \( i = 1, \ldots, n \), compute \( x_i = H_1(l_i) \), and
   \[
   f_i(x) = \prod_{1 \leq j \neq i \leq n} \frac{x-x_j}{x_i-x_j} = a_{i,1} + a_{i,2}x + \cdots + a_{i,n}x^{n-1}.
   \]
   where \( a_{i,1}, \ldots, a_{i,n} \in Z_p^* \).
4) For \( i = 1, \ldots, n \), randomly pick \( a_i \in Z_p^* \), and calculate \( y_i = \alpha_i^{-1}a_i \) and \( U_i = \sum_{j=1}^n a_{j,i}\alpha_j K_{pub} \).
5) For \( i = 1, \ldots, n \), compute \( X_i = H_2(l_i \parallel num) \) and \( R_i = \sum_{j=1}^n a_{i,j}y_j \).
6) Set the ciphertexts of geocast area \( L = \{l_1, l_2, \cdots, l_n\} \) of \( C \) as \( C = (R_1, \ldots, R_n, U_1, \ldots, U_n, C1, h) \).

Further, to prevent the task exposure, \( C \) utilizes CA’s public key to encrypt \( Cont \), that is, picks a random value \( w \in Z_p^* \), \( Z \in GT \) and computes \( D_0 = E_{AES}(H_1(num, Z), Cont) \), \( D_1 = T_{pub}^{w} \), and \( D_2 = Z \oplus e(P, P)^w \). After that, \( C \) utilizes the TMS’s public key \( T_{pub} \) to encrypt the temperate secret key \( k \) by randomly choosing \( r \in Z_p^* \), \( Y \in GT \) to compute \( E_0 = E_{AES}(H_1(num, Y), k) \), \( E_1 = T_{pub}^{r} \), and \( E_2 = Y \oplus e(P, P)^r \). Finally, \( C \) sends the encrypted spatial task \( T = (num, Expt, C, D_0, D_1, D_2, E_0, E_1, E_2) \) to the SC-server, and the SC-server releases \( T \) on its website.

3) Task Recommendation: When the worker \( U \) wants to participate in spatial crowdsourcing activities, \( U \) performs the following interactions with TMS and SC-server to retrieve the recommended spatial task:

1) \( U \) forwards \( cert_u \) to the TMS.
2) The TMS checks \( U \)’s trust level in \( L_M \). If \( U \) is trusted, the TMS computes \( E = e(E_1, T_{pub})^{t-1} \) and returns \( (cert_u, E) \) to \( U \).
3) \( U \) utilizes its secret key \( s \) to decrypt \( (E, E_0, E_2) \) as \( Y = E_2 \oplus e^{s^{-1}} \) and \( k = D_{AES}(H_1(num, Y), E_0) \), and uses the location \( l \) to compute a location trapdoor \( T_1 = (T_1, T_2) \) as \( T_1 = H_1(l) \) and \( T_2 = k(Q + H_2(l \parallel num)) \). \( U \) sends \( (cert_u, T_1) \) to the SC-server.
4) The SC-server uses \( T_1 \) to compute
   \[
   \nu = U_1 + U_2T_1 + \cdots + U_nT_1^{n-1} \quad \text{(mod } q)\]
   and then checks whether
   \[
   h \overset?{=} H_1(num, cert_c, e(C_1, T_2) / e(\nu, \lambda)). \tag{1}
   \]
   If the equation (1) holds, which means that \( U \)’s location \( l \) is in the geocast area \( L \), the SC-server returns \( (num, cert_u, D_0, D_1, D_2, Expt) \) to the TMS; otherwise, it returns failure to \( U \) and aborts.
5) The TMS re-encrypts the ciphertext of \( Cont \) to be decryptable for \( U \) as \( D = e(D_1, P_{pub})^{t-1} \) and sends \( (num, cert_u, D_0, D_2, Expt) \) to \( U \).
6) \( U \) utilizes its secret key \( s \) to decrypt \( (D, D_0, D_2) \) as \( Z = D_2 \oplus D^{s^{-1}} \) and \( Cont = D_{AES}(H_1(num, Z), D_0) \). Finally, \( U \) obtains the task \( ST \) and performs the task if \( ST \) is not expired.

4) Task Fulfillment: \( U \) performs the spatial task \( ST \) and generates a crowdsourcing report \( R_c \). To protect \( R_c \), \( U \) uses \( C \)’s public key \( V_{pub} \) to encrypt \( R_c \) using AES to generate a ciphertext \( F_c \) and sends \( (num, cert_u, F_c) \) to \( C \). Finally, \( C \) decrypts \( F_c \) to recover \( U \)’s report \( R_c \) and fulfills the spatial task \( ST \) according to the crowdsourcing reports from workers.

C. Correctness of LATE

Suppose \( l \in L = \{l_1, l_2, \cdots, l_n\} \) without loss of generality, the correctness of task recommendation can be justified as follows:

\[
\begin{align*}
\lambda &= R_1 + \cdots + R_iT_1^{i-1} + \cdots + R_nT_1^{n-1} \\
&= a_{1,1}y_1X_1 + \cdots + a_{1,n}y_nX_n + \cdots \\
&+ a_{i,1}y_1X_1^{i-1} + \cdots + a_{i,n}y_nX_n^{i-1} + \cdots \\
&+ a_{n,1}y_1X_1^{n-1} + \cdots + a_{n,n}y_nX_n^{n-1} \\
&= (a_{1,1} + \cdots + a_{1,n}T_1^{i-1})y_1X_1 + \cdots \\
&+ (a_{i,1} + \cdots + a_{i,n}T_1^{i-1})y_iX_i + \cdots \\
&+ (a_{n,1} + \cdots + a_{n,n}T_1^{n-1})y_nX_n \\
&= y_1X_1 \\
\nu &= U_1 + \cdots + U_nT_1^{n-1} \\
&= (a_{1,1} + \cdots + a_{1,n}T_1^{n-1})\alpha_1K_{pub} + \cdots \\
&+ (a_{i,1} + \cdots + a_{i,n}T_1^{n-1})\alpha_iK_{pub} + \cdots \\
&+ (a_{n,1} + \cdots + a_{n,n}T_1^{n-1})\alpha_nK_{pub} \\
&= \alpha_1K_{pub}.
\end{align*}
\]

Thus,

\[
\frac{e(C_1, T_2)}{e(\nu, \lambda)} = \frac{e(\gamma P, k(Q + H_2(l \parallel num)))}{e(\alpha_iK_{pub}, \gamma X_i)} \frac{1}{e(\alpha_iK_{pub}, \gamma X_i)} = e(K_{pub}, \gamma X_i).
\]

\[
\frac{e(\gamma P, kQe(\gamma P, kH_2(l \parallel num)))}{e(\alpha_iK_{pub}, \gamma X_i)} \frac{1}{e(\alpha_iK_{pub}, \gamma X_i)} = e(K_{pub}, Q)^\gamma.
\]
IV. SECURITY DISCUSSION

In this section, we discuss the confidentiality of the workers’ locations and crowdsourcing reports in LATE.

The goal of location privacy is to protect the workers’ locations from being known by others. To protect the workers’ locations, it is necessary to preserve the geocast areas of spatial tasks. Thereby, the location privacy can be divided into two parts: geocast area privacy and worker’s location privacy. The geocast area L is encrypted by a temperate public key $K_{pub}$ to generate $C = (R_1, \ldots, R_n, U_1, \ldots, U_n, C_1, h)$. If there exists an adversary $A$ who can identify the points of interest in $L$, there exists another algorithm $C$ who can use $A$ to solve an instance of the Co-DBDH problem. That is, given $(P \in G, u_1 = aP, u_2 = bP, Q \in G, Z \in G_T)$, its goal is to tell whether $Z = \hat{e}(P, Q)^{ab}$. We employ a simulator $C$ to interact with $A$ to prove the indistinguishability of the points of interest in $L$. To prove that, $C$ sets a system by defining the public parameters $(q, G, G_T, \hat{e}, P, Q, K_{pub}, H_1, H_2)$, in which $P_{pub} = u_2$ and $H_1, H_2$ are random oracles, and sends the public parameters to $A$. $C$ can answer the $H_1$-queries, $H_2$-queries and trapdoor queries from $A$. $A$ produces two points of interest $l^*_0$ and $l^*_1$, and generates the ciphertext on one of them $e_{\beta}^*$, in which $\beta \in \{0, 1\}$. Finally, if $A$ outputs a correct guess $\beta' \in \{0, 1\}$ that $\beta' = \beta$, $C$ can utilize the guess to solve an instance of the Co-DBDH problem. Since the Co-DBDH problem is intractable, it is impossible for $A$ to distinguish the encrypted points of interest. Therefore, the LATE can achieve the confidentiality of geocast areas of spatial tasks. In terms of the secrecy of a worker’s location $l$, it is hashed to generate the location trapdoor $T_l$. As the hash function is one-way, it is impossible for the adversary $A$ to learn any knowledge about the worker’s location, unless it tests all the locations to find the proper one.

The spatial tasks and crowdsourcing reports are encrypted using the proxy re-encryption scheme [11] to prevent malicious attackers, e.g., eavesdroppers and hackers, from learning the private information about the customers and workers. Specifically, the customer encrypts the spatial task using the public key of TMS $T_{pub}$, and the TMS is able to transform the ciphertext of spatial task to be decryptable for the recommended workers. Similarly, the workers utilize the public key of the customer $V_{pub}$ to protect the crowdsourcing report $R_c$. The confidentiality of spatial tasks directly depends on the semantic security of the proxy re-encryption scheme, which can be reduced to the simplified $q$-DBDH assumption [11] and the secrecy of crowdsourcing reports can be reduced to the security of AES.

V. PERFORMANCE EVALUATION

To demonstrate the computational overhead of LATE, we count the number of complicated cryptographic operations, including scalar multiplication in $G$, multiplication in $G_T$, exponentiation in $G_T$ and bilinear pairing. We use $SM_G$, $Mul_{G_T}$, $Exp_{G_T}$ and $BP$ to denote the scalar multiplication in $G$, the multiplication in $G_T$, the exponentiation in $G_T$ and the bilinear pairing. We also execute our proposed LATE on a notebook with Intel Core i5-4200U CPU @2.29GHz and 4.00GB memory. We use MIRACL library 5.6.1 to implement number-theoretic based methods of cryptography. The Weil pairing is utilized to realize the bilinear pairing. To ensure the security of the LATE, the parameter $p$ is approximately 160 bits and the elliptic curve is defined as $y = x^3 + 1$ over $F_q$, where $q$ is 512 bits. The number of time-consuming cryptographic operations and the run time of each phase in LATE are shown in Table I.

TABLE I

<table>
<thead>
<tr>
<th>Phases</th>
<th>Operations</th>
<th>Run Time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup</td>
<td>$3SM_G$</td>
<td>17.4</td>
</tr>
<tr>
<td>Releasing</td>
<td>$(2n + 4)SM_G + 3Exp_{G_T} + 3BP$</td>
<td>185.4</td>
</tr>
<tr>
<td>Recommendation</td>
<td>$(2n - 1)SM_G + Mul_{G_T} + 4Exp_{G_T} + 4P$</td>
<td>206.9</td>
</tr>
<tr>
<td>Fulfillment</td>
<td>0</td>
<td>10.5</td>
</tr>
</tbody>
</table>

We also analyze the communication overhead of LATE. When a customer outsources a spatial task, it needs to send the encrypted spatial task $T$ to the SC-server, which is 4864+10243 bits, if we assume the binary length of $Cond$, $Expt$, $Cont$ and $num$ is 256 bits, respectively. A worker also sends its $(cert_c, T_l)$ to the SC-server and receives the recommended spatial task. In Fig. 2, we show the binary length of the encrypted spatial task with respect to the number of places in the geocast region of a spatial task. The binary length of the encrypted spatial task increases linearly with the growth of $n$.

VI. RELATED WORK

Spatial crowdsourcing is a popular model that enables individuals or organizations to obtain their spatio-temporal services or data from distributed contributions of workers on the Internet. Task recommendation is a key process for SC-server, in which many knowledge are leveraged to find proper
workers for performing spatial tasks. Yu et al. [12] proposed a reputation-aware task sub-delegation approach to identify reliable workers to delegate tasks in crowdsourcing. Boutsis and Kalogeraki [13] computes the reliability of workers and the probability that workers would execute tasks in time based on the characteristics of tasks and the profiles of workers, and thereby find a group of proper workers to perform tasks for customers. An et al. [14] studied the credible crowdsourcing assignment model based on social relationship cognition, and proposed service quality factor, link reliability factor and region heat factor to evaluate the crowdsourcing preferences of workers for improving the accuracy of task recommendation. Xiao et al. [15] introduced an offline task assignment scheme and an online task recommendation algorithm following the mobility patterns of workers. Guo et al. [16] discussed the multi-task-oriented worker selection problem and proposed two task allocation in mobile crowdsensing to improve the efficiency of large-scale spatial crowdsourcing platforms. One is a worker selection framework based on workers’ intentional movement for time-sensitive tasks, and the other is a task recommendation framework according to unintentional movement for delay-tolerant tasks. Unfortunately, these schemes disclose the sensitive information to the SC-server to support the task recommendation. Therefore, to resolve the privacy leakage, To et al. [17] introduced a framework to protect the locations of workers based on differential privacy and geocasting. This framework provides heuristics and optimizations to determine effective geocast regions for reaching high task assignment ratio with low overhead. Shen et al. [18] proposed a secure task assignment protocol by utilizing additive homomorphic encryption, which preserves workers’ location privacy in a semi-honest adversary model. Consequently, Ni et al. [19] designed a privacy-preserving location matching scheme for spatial tasks in mobile crowdsensing from matrix multiplication, in which the sensing area of tasks and geographic location of workers are randomized by random matrices to prevent the SC-server learn workers’ locations. Besides, some privacy-preserving schemes [20], [21] have been designed from anonymity techniques to achieve the unlinkability between the identities of workers and the sensitive information disclosed during spatial crowdsourcing services. In this paper, we propose a privacy-preserving task recommendation framework from Lagrange Interpolating Polynomial, which achieves higher accuracy on task recommendation than differential privacy, more efficient than anonymity schemes and matrix-based location matching scheme on computational and communication overhead.

VII. CONCLUSIONS

In this paper, we have proposed a location privacy-aware task recommendation framework to protect the location privacy for workers during task recommendation in spatial crowdsourcing. Specifically, we have designed a privacy-preserving location matching mechanism to enable the SC-server to determine whether the workers are located in the geographic areas of spatial tasks without exposing any information about workers’ locations. Thus, the SC-server is able to recommend the spatial tasks to the workers in the geocast regions with privacy preservation for workers, and it is unnecessary for the workers to travel to specific regions to perform the spatial tasks for customers. As a result, the travel and time costs on workers are reduced. In addition, the spatial tasks and crowdsourcing reports are encrypted during transmission to prevent workers’ privacy leakage. For the future work, we will study the location privacy-preserving multi-task recommendation for a group of workers in spatial crowdsourcing.

REFERENCES