Privacy-Preserving Mobile Crowd Sensing

by

Zhijie Wang

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved February 2016 by the
Graduate Supervisory Committee:

Dijiang Huang, Chair
Guoliang Xue
Arunabha Sen
Jing Li

ARIZONA STATE UNIVERSITY

May 2016

ABSTRACT

The presence of a rich set of embedded sensors on mobile devices has been fuelling various sensing applications regarding the activities of individuals and their surrounding environment, and these ubiquitous sensing-capable mobile devices are pushing the new paradigm of Mobile Crowd Sensing (MCS) from concept to reality. MCS aims to outsource sensing data collection to mobile users and it could revolutionize the traditional ways of sensing data collection and processing. In the meantime, cloud computing provides cloud-backed infrastructures for mobile devices to provision their capabilities with network access. With enormous computational and storage resources along with sufficient bandwidth, it functions as the hub to handle the sensing service requests from sensing service consumers and coordinate sensing task assignment among eligible mobile users to reach a desired quality of sensing service. This paper studies the problem of sensing task assignment to mobile device owners with specific spatio-temporal traits to minimize the cost and maximize the utility in MCS while adhering to QoS constraints. Greedy approaches and hybrid solutions combined with bee algorithms are explored to address the problem.

Moreover, the privacy concerns arise with the widespread deployment of MCS from both the data contributors and the sensing service consumers. The uploaded sensing data, especially those tagged with spatio-temporal information, will disclose the personal information of the data contributors. In addition, the sensing service requests can reveal the personal interests of service consumers. To address the privacy issues, this paper constructs a new framework named Privacy-Preserving Mobile Crowd Sensing (PP-MCS) to leverage the sensing capabilities of ubiquitous mobile devices and cloud infrastructures. PP-MCS has a distributed architecture without relying on trusted third parties for privacy-preservation. In PP-MCS, the sensing service consumers can retrieve data without revealing the real data contributors. Be-

sides, the individual sensing records can be compared against the aggregation result while keeping the values of sensing records unknown, and the k-nearest neighbors could be approximately identified without privacy leaks. As such, the privacy of the data contributors and the sensing service consumers can be protected to the greatest extent possible.

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude to my supervisor Prof. Dijiang Huang for the generous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. His guidance is indispensable in my research and thesis writing.

Aside from my supervisor, I would like to thank my other committee members: Prof. Guoliang Xue, Prof. Arunabha Sen and Prof. Jing Li, for their in-depth suggestions and continuous encouragement, but also for the insightful question which invoke inspirations and motivate me to dig deep into my research.

I also thank my labmates for their stimulating discussions, for the days and nights when we worked together on the projects to meet the deadlines, and for all the sorrows and the joys we shared in the past.

Finally, I would like to thank my family and my friends to support me spiritually in my life, no matter where they are.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Chapter 1

INTRODUCTION

Recent advances in micro-electro-mechanical systems (MEMS) technology have enabled the development of low-cost, low-power, multi-functional and small-size sensors to be embedded in mobile devices, such as smartphones, wearable devices and in-vehicle sensors. Many mobile devices come with Internet connectivity and embedded sensors (e.g., accelerometers, gyroscope, microphone, video camera, GPS, and speed sensors), thereby turning themselves into well-functioned sensor boxes to probe personal activities and environmental phenomena in the vicinity. Consequently, a new sensing paradigm named Mobile Crowd Sensing (MCS) comes into being to harness the potential of the widespread mobile sensors and describe the dynamic patterns of the physical world across a wide variety of application domains.

These sensing capable mobile devices, which consist of sensing, data processing, and communicating components, represent a significant improvement over traditional sensors networks. Most existing sensor networks still require overwhelming expenditure and professional personnel for both deployment and long-term maintenance, which are unaffordable for individuals and small companies. As a result, many areas of interest are out of sensing coverage and the collected data are insufficient for various application requirements. Worse still, the underutilization of current sensor networks and sensor-equipped mobile devices are wasteful of existing investment, as the sensor motes are static and they stay in hibernation mode most of time. Comparatively, MCS offers several advantages over the traditional sensor network infrastructures. First, MCS is built on the already-existing mobile devices with broad network access (e.g., cellular base stations and wifi access points), which are globally widespread and

ready to be used. The International Data Corporation reported that vendors shipped a total of 258.4 million smartphones in the third quarter of 2013 [1]. The persuasiveness of smartphones on an unprecedented scale reduces the deployment cost to almost zero, while the traditional static sensor networks involve overwhelming deployment costs that cannot be afforded by individuals and small enterprises. What's more, the movement of mobile device carriers implicates high spatio-temporal coverage and increases the possibility of capturing unexpected events as compared to static sensor networks. In addition, many existing mobile devices come with open-source platforms, cloud back-end support and app stores (e.g., Google Play, Windows Store, Apple App Store) allowing third party programmers to deploy their sensing applications across the globe and enable sophisticated analysis of big sensor data. Last but not the least, the human involvement could provide additional intelligence such as persuasive user feedback on the sensor data, thus promoting the quality of service and improving the user experience of sensing applications.

1.1 MCS Overview

The MCS system is built on a generic multi-tier service model as illustrated in Figure . A typical MCS system consists of the following parties:

- *Mobile Device Owners* ($MDOs$): The geographically distributed physical sensing networks comprise of conventional sensor nodes and programmable smartphones equipped with embedded sensors. The owners of these sensing devices are termed as Mobile Device Owners (MDOs). The MDOs register their heterogeneous sensor nodes, either static or mobile, at Cloud Provider (CP) to maximize the utilization of the physical sensing resources. As such, the MDOs collaboratively construct the physical substrate sensing networks.

- *Cloud Providers* (CPs): The CPs provide all the necessary IT hardware infrastructures and system software to interface and manage the substrate physical sensing network resources. Note that there may exist multiple CPs and they can collaborate with each other through pre-defined interfaces.

- *Sensing Service Consumers* ($SSCs$): The SSCs issue queries for sensing data (e.g., "the average temperature in Phoenix in the past month") and obtain reports.

- *Sensing Service Providers* ($SSPs$): The SSPs are the running applications hosted by the CPs. They act as intermediaries between the SSCs and the CPs. The SSPs can avoid the overwhelming expenditure and technical pitfalls resulted from the deployment and long-term maintenance of traditional sensor networks, and enjoy greatly simplified processing of the sensing service requests by acquiring available sensing resources through the CP and searching historical sensing records stored in the cloud databases. In this way, the CP not only acts as mediator between the SSPs and the MDOs, but also maintain a buyer-seller relationship between SSPs and MDOs.

1.1

In the MCS infrastructure, MDOs need to create sensor abstracts for sensing devices including the communication protocols, the operating systems, the deployment positions and the mobility state as well as hardware details, such as the microcontroller, on-board RAM, flash memory, battery power and storage. In addition, MDOs need to append their individual network topologies as well as adjacent reachability information to establish the links across different sensing domains. Subsequently, MDOs register their sensing devices at the CP with corresponding sensor abstracts, such that their individual infrastructure can be leased to different SSPs to

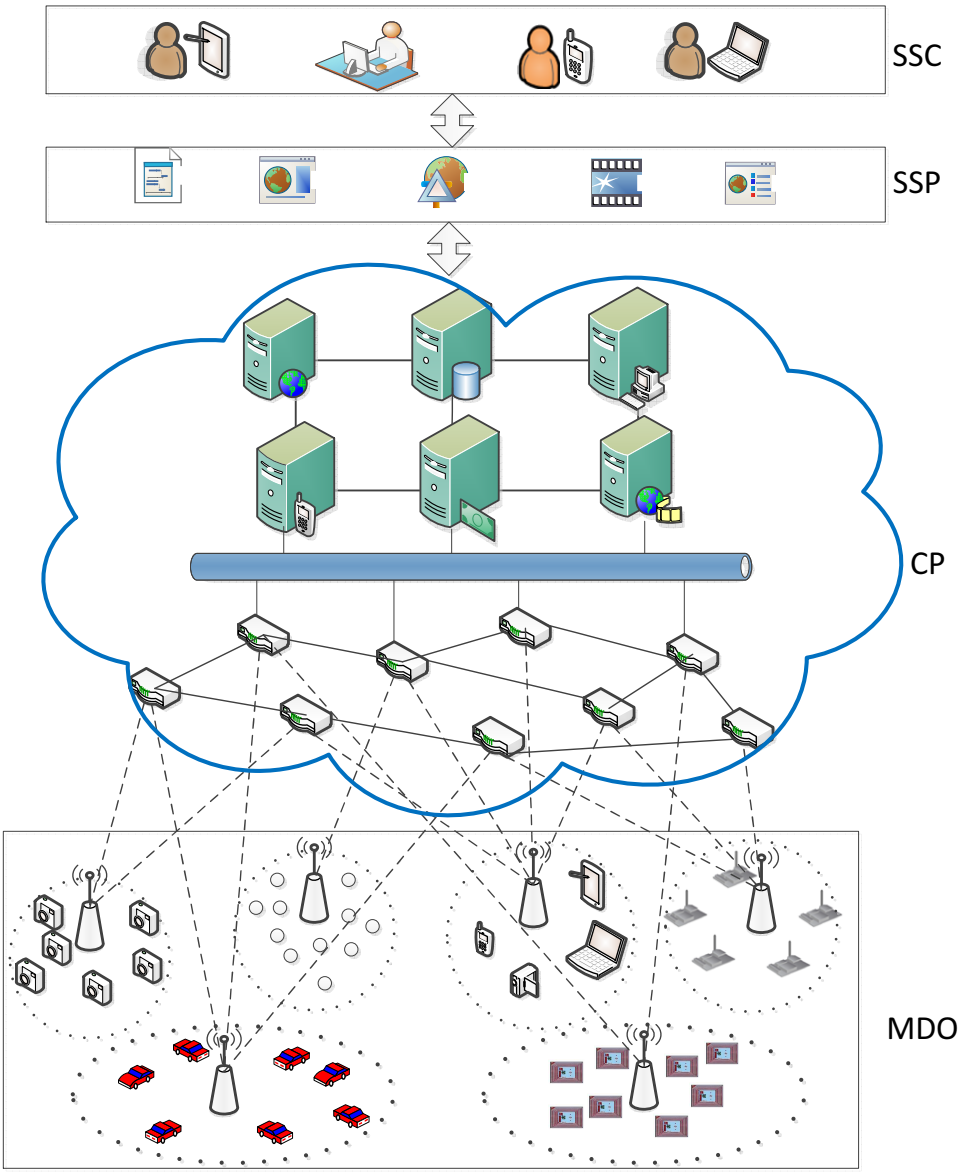

Figure 1.1: Architectural Overview of MCS

run sensing applications through the CP. Correspondingly, the CP can label sensing resources with their sensor abstracts, and interconnect them based on their physical links and topology information. As a result, MCS creates an ecosystem wherein different parties can mutually benefit from each other.

1.2 Sensing Service Query Processing and Access Control

Sensing service query preprocessing is critical to the performance of MCS in practice. First of all, the mobile cloud enforces authentication and authorization over the SSCs, and only the eligible SSCs are allowed to pose sensing service requests along with bid prices for specific sensing services. In addition, many SSCs subscribe to events that satisfy certain predicates, and the SSPs and the CP should guarantee timely delivery of published events to all the interested service consumers. Generally speaking, each query consists of a series of predicates, such as the temperature lying between $25°C$ and $30°C$, and events occurred within a certain area. Evaluation of these predicates result in significant system resource consumption, especially when there exists a large number of queries waiting in a queue. However, some queries may share the same predicates, and such overlap can be exploited to reduce the evaluation cost. In addition, a query usually consists of multiple predicates, and the '*false*' evaluation result of the predicate at the beginning removes the need to evaluate the following predicates. Moreover, if the such a predicate is shared by many other queries, all the identical predicates in these queries can be ignored, and this can significantly accelerate the query evaluation process. Therefore, Liu *et al.* [2] proposed a sub-order algorithm to first evaluate the predicates with low selectivity (i.e., the probability that an event satisfies a given predicate), such that they have larger chances to eliminate other predicates. In the meantime, it utilizes linked-chains to connect any two queries if they share the same predicate. As such, the evaluation result of any given predicate can be shared across different queries, and the system evaluation overhead can be remarkably reduced. Nevertheless, it does not support the range predicate case. In addition, it brings remarkable overhead in case of inserting and deleting queries, because it has to maintain a graph where there exists a link

between every two queries if they share the same predicate.

Some clustering schemes [3, 4] were also proposed to improve the sensing event matching by clustering the queries, such as K-means clustering, spanning tree based clustering and grid partitioning. However, the resulting clusters are partially overlapped and this incurs false negative events or false positive events. In addition, the queries can be aggregated, filtered and matched by Bloom filters [5, 6]. However, their requirements of limited set of possible publication and query attributes limit the expressiveness, and the usage of Bloom filter brings false positive or false negative results. Mehedi Hassan *et al.* [7] developed a dynamic and fast content-based publish/subscribe information dissemination system for automatic fusion and delivery of large amounts of sensor data to service consumers. It exhibits good support of range predicates while eliminating false positive and false negative results.

An important privacy issue related to the sensing service query is that it can expose the interest of sensing service consumers. One solution to this issue is to conceal the identity information by applying attribute-based access control and relying on independent identity service providers, which is detailed in chapter 4. Attribute-based access control offers a salient feature of anonymity, as it grants data access to service consumers with attributes satisfying specific attribute policies. The identity service providers takes IDs in the registration process and offers proofs of attribute ownership for authorization purpose. Another solution is to hide critical content of the sensing queries by mixing the targeting data sources (e.g., mobile device owners, locations, time periods) with untargeted data sources with cryptographic techniques, which is detailed in chapter 2.

1.3 Sensing Task Assignment

Upon receiving the processed sensing service requests, the CP should accordingly locate appropriate sensing resources based on the sensor abstracts taking account of sensing capability, resource limitation and infrastructure support. In addition, different sensing service requests have different requirements in spatio-temporal coverage. The event-driven sensing applications (e.g., continuous surveillance in public places in case of emergency) usually require high spatiotemporal coverage ratio or even a complete spatiotemporal coverage, while low spatiotemporal coverage ratio would suffice some data-driven sensing applications, such as the pattern extraction of long-term variations in air quality in a city, because the segmentation and sampling algorithms can be leveraged to infer the measurements in uncovered spatiotemporal space and reduce costs. Consequently, the CP only selects the sensing devices equipped with sensors corresponding to the sensing service requests, and the traces of the sensing devices should appear in concerned areas during time intervals of interest.

The behaviour patterns of the mobile device owners also have direct impact on the quality of the collected sensing data, and it is important to select appropriate sensing devices from a pool of candidates based on the evaluation results of the historical information regarding the corresponding MDOs. As with the recruitment framework [8], it checks the candidates' previous spatiotemporal coverage over the area of interest and data collection reputation based on the historical data sets, and selects well-suited participants out of the candidates to achieve maximum utility. Specially, participants who are active in sensing data collections are preferred than passive participants, but the resulting costs should be taken into consideration, as the active ones may demand high payback. The quality assessment can take place periodically, and additional participants might be recruited if the sensing campaign

experiences underperformance. In addition, strong incentive mechanisms should be implemented to motivate the participants to carry out the sensing tasks and prevent them from opt-out. For instance, the participants with similar interest can form a virtual group to take sensing tasks together due to their common concerns (e.g., the concern about the air pollution density of their ambient environment). Also, the mobile clouds can provide the participants with free services (e.g., health monitoring) in exchange for their participation in some sensing tasks. A straightforward method is to offer the participants monetary incentives, and the participants with outstanding performance can receive more monetary rewards than those who underperform, but the overall budget should not exceed what the SSCs can afford.

1.4 Privacy-Preserving Sensing Data Processing

In some cases, the sensing service query only need to retrieve data originated from one independent data source (e.g., the SSP or a single MDO). Choi *et al.* [9] applied homomorphic encryption to allow the client to securely determine if her location is in the proximity zone of a target, or if the target is in the proximity zone of the client. More often than not, the MCS usually relies on many mobile devices to co-operatively feed the sensing readings into the data stores of the CP in response to various sensing service requests. Specifically, the sensing service consumers need to study the distribution of the input sensing data, and their sensing service requests only require aggregation results (e.g., summation, mean, deviation, maximum, minimum, k-nearest) derived from the sensing readings of multiple mobile devices rather than individual sensing record values. As such, it is necessary to derive aggregate result from various data resources without breaching individual privacy by revealing specific sensing record values. Chapter 3 explores how to privately compare each individual sensing record value against specified ratio of the summation of all sensing

values derived from secure multi-party computation. It further studies how to approximately identify the k-nearest neighboring sensing readings without disclosing the exact values of the benchmark location and any sensing readings. Note that the same techniques can be easily extended to the benchmark data and the sensing readings of any dimensions other than the two-dimensional location data.

1.5 Parallel Data Processing

The MCS rests on potential millions of mobile devices to feed the massive volumes of data streams into the data stores of the CP in response to various sensing service requests. Imagine a scenario wherein a sensing application for real-time dust storm monitoring in Phoenix area requires each participant to transmit a 60-byte data packet with coordinates to the CP every three seconds. Suppose one million local residents subscribe for this application, and together they will contribute 1.2 Gbytes of data per minute. These collected data need to be analysed in a timely manner to extract knowledge and broadcast alarms, which cannot be done in traditional computing environment.

Hadoop is an implementation of the emerging parallel computing model MapReduce [10]. It is able to process the potentially massive amount of sensor readings concurrently. Two types of nodes, a *jobtracker* and multiple *tasktrackers*, are used to control the job execution process as shown in Figure 1.2(a). The *jobtracker* schedules sensing data processing tasks to run on the *tasktrackers*, and the *tasktrackers* keep records of the running jobs and submit reports to the *jobtracker*. Once a sensing data processing tasks fails, the *jobtracker* immediately reschedules it to a different *tasktracker* to run. The *tasktracker* works by breaking the MapReduce job into the *map* task and the *reduce* task. All the sensor readings are prepared in the form of key-value pair $< k1, v1 >$, and each *map* processor is assigned its $k1$ to work on. The

map processors take in $< k1, v1 >$ data chunks and emit the output key-value pairs $< k2, v2 >$. Subsequently, the output is shuffled by grouping the pairs with the same key $k2$. In consequence, the *reduce* processor takes $< k2, list(v2) >$ as input to apply *reduce* operation and produce new output $< k3, v3 >$.

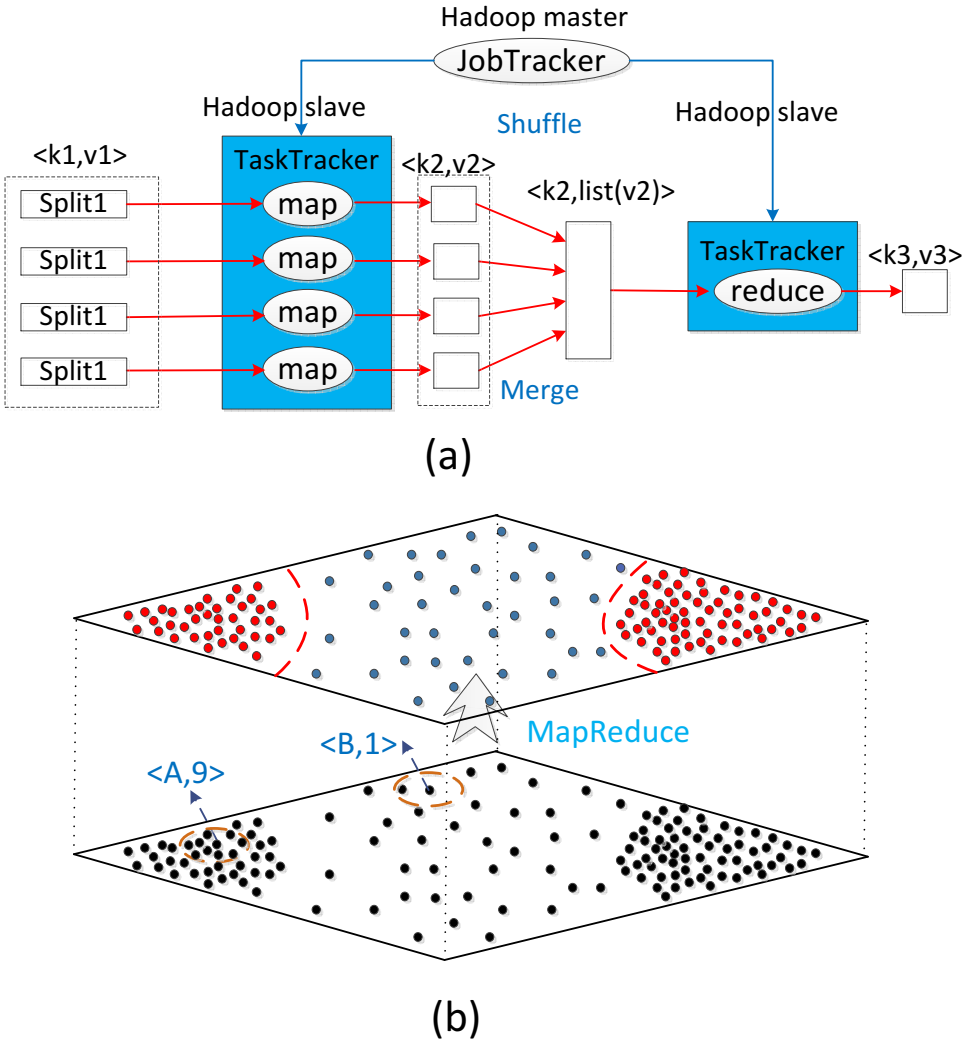

Figure 1.2: Hadoop: a) MapReduce Data Flow b) A Clustering Example of Parallel Processing

A simple example of the data analysis with parallel processing is to partition all the mobile devices within a monitored region into different clusters with *high* and

low population density. The Hadoop in MCS can take advantage of the locality of mobile devices and uses two map-reduce phases to perform partitioning as illustrated in Figure 1.2(b). The first map-reduce phase computes the neighbour density for all the mobile devices. Assume we have m mobile devices within the monitored region. For a specific mobile device $DevID_i (i \in [1, m])$, the other mobile devices that have a distance no more than a given radius r are deemed as its neighbours, and the map function produces a key-value pair $\langle DevID_i, yes \rangle$ for each neighbour and $\langle DevID_i, no \rangle$ for a non-neighbour. As a result, the map function produces $m(m-1)$ pairs of $\langle DevID_i, yes/no \rangle$ in total. The shuffle function derives the number of neighbours for each mobile device $DevID_i$ by counting the number of $\langle DevID_i, true \rangle$ locally, while the reduce function counts the number of neighbours n_i for each mobile device $DevID_i$ globally and emit $\langle DevID_i, n_i \rangle$. For example, the resulting key-value pair for the mobile device A is $\langle A, 9 \rangle$ and that for mobile device B is $\langle B, 1 \rangle$. In the second phase, two centroids n_j and n_k are selected for the high-density cluster and low-density cluster respectively where $j, k \in [1, m]$. The map function assigns each $\langle DevID_i, n_i \rangle$ to the closest centroid, and the reduce function computes the average number of neighbours in each cluster and updates the two centroids. The map-reduce operation proceeds recursively until the centroids do not change and the clustering is complete.

1.6 Data Presentation

The processed sensing data sets need to be presented to the SSCs with application-specific formats. It directly impacts the SSCs' evaluations of sensing services, therefore it must convey the sensing results effectively and elegantly. A variety of conventional methods are leveraged to present the data, such as tables, plots, histograms, pie charts and so on. Accordingly, the data must be formatted first to be able to feed

into the application interfaces. For the location-based sensing services, the sensing results are usually converted into geographic markup languages such as csv and KML, and demonstrated to the SSCs using Geographic Information System (GIS) mapping applications.

Chapter 2

RESEARCH BACKGROUND

MCS possesses the potential to merge into the fabric of everyday life by offering highly personalized services based on fine-grained spatiotemporal data. Nevertheless, the potential leakage of personal information regarding the involved parties could could adversely influence the growth of MCS market. Hence, the privacy concerns must be addressed for various sensing applications to benefit both the academia and the industry. The privacy attacks that are investigated in this paper are categorized in the following descriptions. The issues in existing privacy protection techniques are analyzed, and new solutions are presented.

2.1 Sensing Task Assignment with QoS Constraints

In the procedure of sensing task assignment, the clouds assign the tasks to the devices of the selected participating MDOs after evaluation. A typical MDO recruitment process [8] in MCS includes three steps. The first step is to find suitable MDOs who are equipped with required sensors and present in target areas within time periods of interest. The second step evaluates the participants based on various criteria including traces, spatiotemporal coverage, sensing costs and utilities. The third step pushes the sensing tasks to the selected participants. In this procedure, the requirement of Quality of Service (QoS) regarding the spatiotemporal coverage for the specific sensing tasks should be taken into consideration. Different sensing applications may have different spatiotemporal coverage requirements. For instance, a fire alarm sensing application system require nonstop monitoring of temperature, carbon monoxide and carbon dioxide density levels every day in every spot of a region. Com-

paratively, a environment monitory application system for nocturnal animal behavior research only start to work at night around some animal habitats. Accordingly, the sensing task assignment strategies should be adjusted based on various QoS requirements associated with spatiotemporal coverage. This paper first investigates greedy approaches for the cost minimization and utility maximization with QoS constraints, and then adopts a hybrid approach combining both greedy method and bee algorithm for better results.

2.2 Privacy Leakage

The first category of privacy leakage stems from the sensing task assignment. In the MDO recruitment process, it is inevitable for the clouds or other parties to obtain the profile information of all possible candidates, and the task selections can be exploited to learn the private information of participants. A semi-hones CP or SSP may create sensing tasks with strict limitations on the attributes of participants by requiring a special sensor type, a specific sensing area or certain behavior pattern to narrow down the possible identities of participants. Besides, a malicious SSP can contrive multiple sensing tasks in an attempt to link the participants for denonymization. For example, the CP or the SSP may attempt to identify the home location or working hours of a candidate in the name of participant selection for sensing task assignment.

The second category of privacy leakage comes from the queries of the SSCs. The SSCs can subscribe queries to the appropriate SSPs in request of one or more types of sensor readings in specific regions during designated time periods. For instance, Alice is interested in the traffic information in North Phoenix between 8am and 9am on weekdays, and Bob subscribes to "available pools in East Tempe at noon". Besides, the selected participant may demonstrate common features and behavioral patterns,

thereby incurring privacy leakage. The queries can be sensitive as they reflect the SSCs's interest and behavior pattern, they also need to be hidden and protected.

The third category of privacy leakage result from the data collections over MDOs. In MCS, an enormous amount of potentially sensitive information could be generated by tracking the users automatically on an ongoing basis, plentiful of sensitive information about MDOs can be collected, thereby resulting in the violation of the privacy of the participants' traces, interests, life styles and so on. For example, the CP learns that Alice is in a political parade on South Mill Avenue, Tempe, and Bob and Carol are together in Starbucks Coffee on Thursday afternoon. As a result, the CP is able to profile the MDOs thanks to the continuous personal data collection over long time periods. Even one single task action does not breach the privacy, the identities and attributes can be disclosed by linking multiple task actions together.

The fourth category of privacy leakage derives from the access control of shared sensing data. In some cases, the MDOs may voluntarily collect data for future public use of specific SSCs in certain attribute domains without sensing service queries in advance. For example, some volunteers donate their health information to some repositories for the possible future use of public healthcare research institutes. However, the conventional methods of showing certificates would expose the identities of the SSCs and raise privacy concerns. Therefore, specific access control strategies need to be designed such that the SSCs may want to access these data anonymously after demonstrating that they belong to the attribute domain of public healthcare without exposing their identities.

2.3 Issues in Existing Privacy Protection Techniques

Information access control techniques [11, 12] require a trusted middle-ware service lying between location-based applications and the mobile users to enforce access

control over geo-spatial data by rule-based access policies. Specifically, LocServ [11] answers queries of three types: i) requests for user location identified by the users' unique identifiers ii) enumeration requests to return lists of users at specific locations iii) asynchronous requests to notify events when users enter or leave areas of interest. The enforcement mechanism in [12] consists of a spatio-temporal module, an encoder and the ASM-trie, and it follows a hierarchical access control model to enable adaptive search and support positive and negative location-based data access. While the techniques stated above depend on the third party to enforce the access control over the location-based data, they are vulnerable to the malicious behaviors of the third party.

Mix-zones [13, 14] also utilize middle-ware between the mobile users and the location-based application such that the location-based applications receive and reply pseudonymous messages from the mobile users. Specifically, the middle-ware assures the unlinkability of their pseudonyms by assigning a new, unused pseudonym to the mobile users when they enter a mix zone, and thus the location-based application cannot link the users emerging from the mix zone to the ones going into the mix zone. MobiMix [14], an application of mix-zones over road networks, develop a suite of construction methods to protect the location privacy of the mobile users. By the same token, it is subject to malicious behaviors of the third party middle-ware, as the middle-ware functions as a proxy to anonymize the locations of mobile users.

K-anonymity [15] ensures the information for each person in a release cannot be distinguished from at least $k-1$ individuals whose information are also in the same release. As a result, an attacker can identify a user based on the location information with probability no more than $1/k$. PRIVACYGRID [16] provides effective cloaking algorithms for location k-anonymity and l-diversity in a mobile environment wherein the mobile users communicate with the location-based service servers via location

anonymization servers. Casper [17] uses the location anonymizer server to blur the users' exact information into cloaked spatial regions based on user-specified privacy requirements, and the privacy-aware query processor of the database only deal with the cloaked spatial areas instead of the exact location. In [18], a mobile user has to collaborate with $k-1$ peers to cloak her exact location into a spatial region before querying the location-based database server. All these k-anonymity schemes rely on the assumption that the third party is entrusted, which is usually infeasible in real-world settings.

Comparatively, the dummy location approaches [19, 20] eliminate the need of a third party server by generating redundant location-related data. In [19], a mobile user sends true location data with several dummy location data to a location-based service provider. The dummy locations are generated randomly and they are not real user locations as in the k-anonymity schemes. The privacy of the user location is protected as the location-based service provider cannot distinguish the true location from several dummy locations. SibilQuery [20] allows the user to generate $k-1$ Sybil queries to achieve k-anonymity, such that the location-based server is unable to distinguish between the user's real query and the Sybil queries. As the dummy location approaches rely on data redundancy rather than third-party anonymizer, they significantly increase the system overhead and complexity. At the same time, the location-based service provider can narrow down to the sub-space of the exact location, thereby resulting a weak privacy.

Data transformation involves data owners, data users and clouds. The data owner uses certain encoding methodology to transform his data sets before outsourcing them to the cloud, while the data user attempts to retrieve the encrypted data sets with queries and perform decryption with the transformation keys derived from the data owner. The cloud is honest-but-curious about the encrypted data sets, and the cloud

can perform search over the encrypted data sets based on the data user's encoded queries although they are unreadable. OPES [21] encrypts the data in an order-preserving manner to enable distance comparison operations. Wong *et al.* [22] allows kNN processing by using a secure point transformation to preserve the distances of points of interests relative to any query points. Khoshgozaran *et al.* [23] proposed to use Hilbert transformation to transform the points while the parameters (e.g., scale, order) keep secret. The data transformation techniques are vulnerable to access pattern attacks, and they are not scalable in real-world settings, because each data user has to get the transformation key from the data owner, thereby resulting in unbearable offline communication overhead.

Personal Information Retrieval (PIR) [24, 24–30] protocols allow a user to obtain the i-th record from the database server without disclosing which record he is obtaining. First, the user pins down his location index in the database via the location-based service provider without revealing his whereabouts. In this phase, Paulet *et al.* [29] utilize oblivious transfer while Ghinita *et al.* [25,26] employ homomorphic encryptions. After that, the user relies on PIR protocols to retrieve points of interests associated with the index from the location-based service provider. Trusted hardware [28] can be used to generate the secret key and permute the database. Generally speaking, PIR protocols are secure against access pattern attacks, whereas they are too costly to be applied in real-world settings.

At the same time, l-diversity and t-closeness has been taken into consideration in MCS to increase anonymity and enhance privacy of the MDOs who contribute data. In k-anonymity, any quasi-identifier present in the released data set must appear in at least k records. Nevertheless, k-anonymity does not protect privacy when sensitive values in an equivalence class lack diversity or the attacker has some background knowledge. Hence, it is necessary to diversify sensitive attributes within each quasi-

identifier equivalence class to achieve l-diversity, such that each equivalence class has at least l well-represented sensitive class. But the sensitive values are usually not evenly distributed, so l-diversity does not prevent probabilistic inference attacks. Therefore, the distribution of sensitive attributes in each quasi-identifier group should be "close" to the entire data set to achieve t-closeness. As a result, the complexity of data preprocessing is significantly raised while the data utility is seriously reduced due to masked data.

Pseudonym-based methods help offer anonymity to the MDOs, but they have difficulty in maintaining the reputations associated with their online identities. The retrieval of reputation points for a given public profile would endanger the privacy of the MDO, as it could link to his real identity. Consequently, the process of retrieving reputation points should be decoupled from updating the reputation value of some MDO's profile. A straightforward approach is to use a pseudonym-based credit system, where a MDO trades sensing data submission for credits from the CP. In this manner, the MDO's identity is untraceable as it is hidden by the pseudonym, but the pseudonym is bound to credits. Credits are anonymous but traceable, which implies the CPs can profile the MDOs behind pseudonyms by linking the credits. To overcome this issue, it is necessary to change pseudonyms frequently. As a result, the unlinkability and anonymity are preserved, while the credits associated with that pseudonym are lost, which could demotivate the MDOs to participate in performing the sensing tasks.

Data perturbation is useful in reducing the risk of compromising privacy, the MDOs and the SSCs tend to submit perturbed sensing data and queries with generalized context to the CPs and the SSPs respectively. Consequently, the system becomes less efficient and obtain reduced utility, which represents the usefulness of sensing tasks, because the CPs may have to task a larger pool of participants and

the SSPs need to conduct more computation to reach a certainty like a non-private process. It is self-evident that the two goals are hard to be optimized at the same time. Without a detailed knowledge of the context and raw data, it is hard to select the best participants and filter out noise and corrupt data to obtain the maximal utility. Our system should optimize the MCS processes to maximize the expected utility while subject to privacy concerns concurrently.

2.4 The Proposed Solutions for Privacy Preservation

Homomorphic encryption plays a significant role in the proposed solutions to preserve the privacy of SSC's sensing queries, MDOs' profiles and individual data records without any trusted third parties. Homomorphic encryption allows computations to be performed over ciphertext ending up with a ciphertext which equals the results of operations performed on the plaintext when it is decrypted. Specifically, it could be designed in sophisticated ways such that some MDOs' computations can be neglected in the final aggregation result while other MDOs' computations take effect without the awareness of the involved MDOs. As a result, no MDO knows if it is selected as the real data source and the content privacy of the SSC's queries are preserved. Additionally, the signs of polynomials can be derived without disclosing the numeric values of the data records of the involved MDOs. As such, given a baseline value d, the difference between the sensing reading of any MDO and d can be compared against a designated proportion of the summation of sensing reading differences of a group of MDOs without revealing the vaues of d and any sensing readings, and it is easy to learn if one MDO's sensing reading difference is above or below the average level. If there exist a large number of MDOs involved in this procedure, the parallel data processing framework based on mapreduce concept can be implemented to break the computation task of aggregation into parts and process them on different

computation nodes concurrently to speed up the process. Although there exists a large body of research work on parallel data processing frameworks, little attention has been paid to the computation of data in encrypted forms in the past, and this makes the contribution of this paper more significant.

In addition, given a benchmark location, a novel divide-and-conquer solution is provided to approximately identify k-nearest MDOs without revealing the benchmark location and any locations of MDOs. In this solution, all the possible distance values between the benchmark location and MDO location are included in a distance range, and all the MDOs agree on the privacy window of the smallest size such that the privacy level regarding the distances of some MDOs in the worst case is still acceptable. In each step, polynomial inequalities in encrypted form would be constructed for each MDO's distance and a designated pivot value, and the sign of each polynomial inequality indicates if a MDO's distance is below the designated pivot value. Accordingly, the search range narrows down step by step in a divide-and-conquer manner until k-nearest MDOs are identified or the size of the search range has shrunk to the smallest acceptable size of the privacy window. The details of this solution is elaborated in chapter 4.

Finally, a distributed access control system is designed for secure sensing data sharing in a decentralized MCS scenarios in chapter 5. It allows the SSCs to outsource the burdensome task of identity management to multiple trusted identity providers and avoid single point of failure by utilizing decentralized attribute-based encryption and identity-based encryption. At the same time, the MDOs can specify various access control policies without causing significant system overhead by using linear secret sharing scheme. The identity providers can prove the attribute domains that the MDOs belong to without directly contacting the SSPs and the SSPs can make access control decisions accordingly, thereby protecting the privacy of the historical

access of the SSCs to the sensing data hosted by the SSPs. Centralized access control mechanism with one trusted authority for key generation based on attribute-based encryption can be applied in centralized MCS scenarios for easy management.

Chapter 3

SENSING TASK ASSIGNMENT IN MCS

The timeliness and quality of the collected data become the major concerns in MCS due to its infrastructureless and distributed nature, and thus it is critical to select appropriate participants carried with mobile devices to provide sufficient data about the target spatial areas during the time periods of interest and meet the application needs at various levels. Generally, we can classify the participating candidates into two major categories, namely *regular participants* and *opportunistic participants*. The *regular participants* follow repetitive traces with a regular spatiotemporal moving pattern during a time period (e.g., a day), and their locations at a specific time slot can be determined a priori. Examples of *regular participants* include city buses, school buses, trams, street sweepers, and so on. In contrast, the *opportunistic participants* have opportunistic daily traits due to their uncontrolled mobility (e.g., pedestrians, taxis), and their locations at a specific time cannot predicted. To maintain a stable spatiotemporal coverage, we only consider *regular participants* and hereafter use participant to refer to *regular participants* in this paper.

There have been some work on the sensing coverage problems in mobile sensing. Reddy *et al.* [8] proposed a recruitment framework to maximize the utility associated with spatiotemporal coverage with constrained budget in persuasive sensing, and Riahi *et al.* [31] presented efficient algorithms to deal with queries of different types and maximize the total utility in participatory sensing. As the spatiotemporal coverage has direct impact on the sensing service quality, we consider the sensing task assignment problem from the perspective of spatiotemporal coverage ratio and define it as the Quality of Service (QoS) in our MCS scenario. Different sensing applications have

different QoS requirements. The event-driven sensing applications (e.g., continuous surveillance in public places in case of emergency) usually require high spatiotemporal coverage ratio or even a complete spatiotemporal coverage, while low spatiotemporal coverage ratio would suffice some data-driven sensing applications, such as the pattern extraction of long-term variations in air quality in a city, because the segmentation and sampling algorithms can be leveraged to infer the measurements in uncovered spatiotemporal space and reduce costs [32].

Consequently, we study the strategies of sensing task assignment in MCS with QoS constraints in this paper. As illustrated in Figure 3.1, the sensing campaign organizer investigates the empirical data sets with respect to the candidates' historical mobility traces and transportation modes, estimates their stability in the behavioral space, and accordingly select well-suited participants to meet various QoS requirements. As the execution of sensing tasks inevitably incur costs due to sensor installation, battery consumption, data storage and transmission, etc., we formulate the QoS-Constrained Sensing Cost Minimization Problem (QSCM) with the objective of minimizing the sensing cost while adhering to the QoS constraints. On the other hand, the spatotemporal coverage yields benefits for the sensing campaign organizers. We hereby define the sensing utility as the difference between the benefits and the costs. The benefits are proportional to the spatotemporal coverage and the costs increase with the number of selected participants. Hence, more selected participants does not necessarily result in higher utility, as the costs grow with number of participants while the spatotemporal coverage derived from different paticipants' mobility traces may overlap with each other. Consequently, we formulate the QoS-Constrained Sensing Utility Maximization Problem (QSUM) with the objective of maximizing the sensing utility while adhering to the QoS constraints. Our contributions in this paper are three-fold: i) we formulate the problems of cost minimization and utility maximiza-

tion with QoS constraints in terms of spatiotemporal coverage ratio and prove that they are NP-hard problems; ii) we present greedy approaches to address them, and propose new heuristic hybrid approaches combining the Bees algorithm and greedy algorithm to provide better performance; iii) we conduct extensive simulation and the numerical results prove that the hybrid approaches outperform the greedy approaches with lower cost and higher utility.

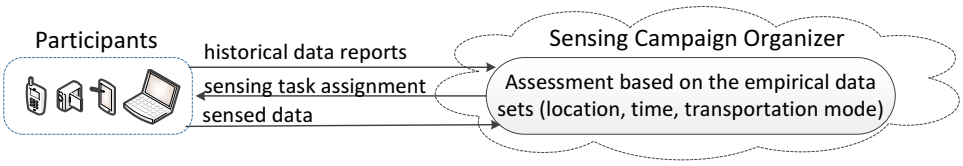

Figure 3.1: Sensing Task Assignment in MCS

The remainder of this paper is organized as follows. Section II gives problem formulations and offers preliminary analysis. Section III presents greedy approaches and Section IV describes the hybrid approaches in detail. Section V evaluates their performance and provides analysis. Section VI discusses the related work, and Section VII concludes thise paper.

3.1 Problem Formulation and Analysis

The spatiotemporal coverage is an important metric in MCS, since the location and time are crucial context in analysing the semantics of sensing data and exploring the phenomena of interest. As each sensing device can only cover a spacial range at a time, the regions of interest can be partitioned into many smaller subregions which fit the sensing range. Also, the time span of interest can be discretized into many fine-grained time units of equal length, e.g., 5 minutes. Consequently, the sensing space is composed of spatiotemporal blocks along the spacial dimension and the temporal dimension. As such, the mobility trace of each participant can be

modelled as a series of spatiotemporal blocks in the sensing space. Consider an urban area in Figure 3.2 where m hotspot regions $S_j(1 \leq j \leq m)$ needs to be monitored within different time intervals of interest during a time period T (e.g., a day), which is sliced into time units with equal duration. Accordingly, each hotspot region S_j corresponds to a spatiotemporal domain ST_j of which the projection on the temporal axis span across $|T_j|$ time units, which could be discontinuous. There exist a participant pool $\{p_1, p_2, \ldots, p_n\}$ consisting of n participants, and each participant p_i moves along statistically equivalent trace tr_i consisting of spatiotemporal units during time period T.

As illustrated in Figure 3.2, there exist three hotspot regions $S_j(1 \leq j \leq 3)$ in the urban area corresponding to the interested time intervals $T_j(1 \leq j \leq 3)$ respectively. Assume the sensing campaign organizer selects participants p_1 and p_2, and each participant moves at a speed of one spatial unit per time unit. The participant p_1's trace tr_1 has 3 overlapped spatiotemporal blocks with S_1 and 2 overlapped spatiotemporal blocks with S_2, while tr_2 has 3 overlapped spatiotemporal blocks with S_2 and 3 overlapped spatiotemporal blocks with S_3. As a result, the spatiotemporal coverage ratio of ST_1, ST_2 and ST_3 can be computed as $3/|ST_1| = 12.5\%$, $(3+2)/|ST_2| = 10.2\%$ and $3/|ST_3| = 20\%$.

3.1.1 QSCM: The QoS-Constrained Sensing Cost Minimization Problem

The engagement of a new participant p_n in the sensing campaign can help increase the spatiotemporal coverage ratios of the hotspot regions, whereas it also raises the cost c_n stemmed from mobile sensor installation, battery consumption, etc. As more participants with various traces join the sensing campaign, the actual spatiotemporal coverage ratios could exceed the QoS requirements of the sensing applications with unnecessary cost. The goal of QSCM is to find a subset of participants that minimize

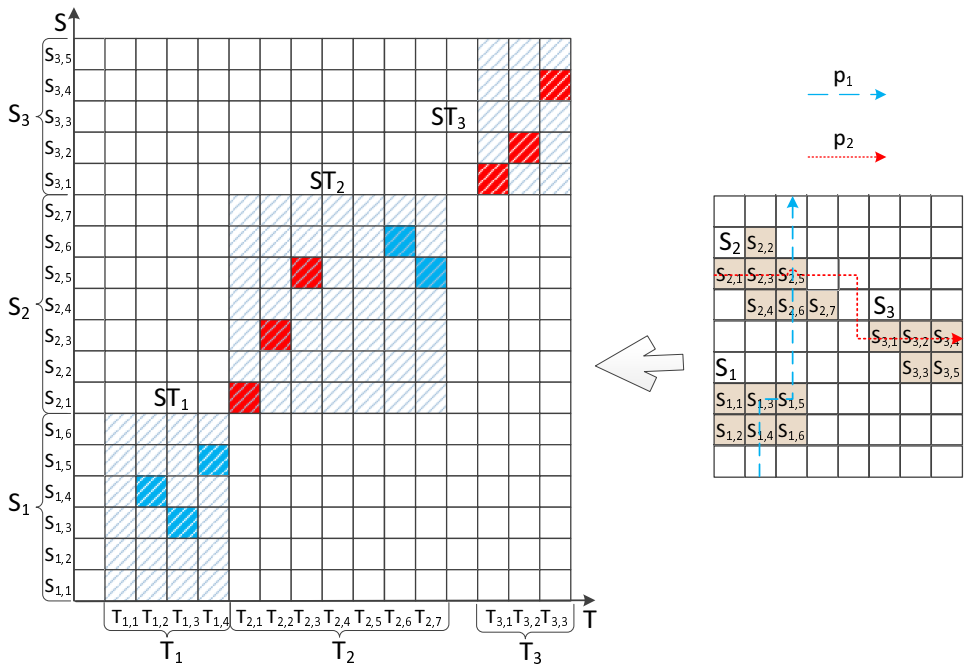

Figure 3.2: Illustration of Spatiotemperal Coverage in MCS

the overall cost while fulfilling the QoS requirement. We define the total sensing cost as follows:

$$C(\vec{\mathbf{x}}) = \sum_{i \in N}(d_i + \sum_{j \in M} |tr_{i,j}|b_i)x_i \ ,$$

where $\vec{\mathbf{x}} = (x_1, x_2, \cdots, x_n)$ is a n-dimensional $(0, 1)$ vector, $M := \{1, \ldots, m\}$ is the set of hotspot indexes and $N := \{1, \ldots, n\}$ is the collection of participant indexes. The participant selection vector $\vec{\mathbf{x}}$ represents the participant selection results (i.e., $x_i = 1$ iff participant p_i is selected otherwise 0). For each participant p_i, d_i denotes the corresponding sensor installation and maintenance cost, b_i denotes the sum of the battery consumption cost and data transmission cost per each spatiotemporal block, and tr_i represents p_i's mobility trace consisting of spatiotemporal blocks. We use $tr_{i,j} = tr_i \cap ST_j$ to represent the set of the overlapped spatiotemporal blocks between tr_i and ST_j. Consequently, we define the QoS criterion of ST_j as shown below:

$$QoS_j(\vec{\mathbf{x}}) = \frac{\left| \bigcup_{x_i=1, i \in N} tr_{i,j} \right|}{|ST_j|},\ j \in M$$

and the QSCM problem can be formulated as below:

$$\begin{aligned} & min\, C(\vec{\mathbf{x}}) \\ s.t.\quad & QoS_j(\vec{\mathbf{x}}) \geq r_j,\ \forall j \in M \end{aligned}$$

where the constraints indicate that the QoS of ST_j should be no less the designated threshold for all $j \in M$. Furthermore, the QSCM problem is an NP-hard problem, which can be proved with the Theorem 1 below:

THEOREM 1. *The QSCM is an NP-hard optimization problem.*

Proof of Theorem 1. *This can be proved by reduction from k-partial set cover [33]. The k-partial set cover is a generalization of the well-known set cover problem. It strives to select a minimum number of sets to cover at least k elements and it is NP-hard. Given an instance of k-partial set cover problem (U, S, k) where U is a set of all elements and S is the set of subsets with elements from U, we can construct a corresponding QSCM problem $(\{ST_j, r_j\}_{j \in M}, \{tr_{i,j}\}_{i \in N, j \in M}, \{b_i, d_i\}_{i \in N})$ with $M := \{1\}, U = ST_1, r_1 = k/|U|, S = \{tr_{i,1}\}_{i \in N}$. Furthermore, we have $b_i = 0$ and $d_i = 1$ for every $i \in N$. This construction can be done in linear time that is the same size of k-partial set cover instance. On the other hand, if we have a QSCM in the constructed problem with $M := \{1\}$, we can choose the subsets corresponding to the selected participants. Consequently, the k-partial set cover can be reduced to QSCM with $M := \{1\}$ in polynomial time, which is a subproblem of QSCM. Therefore, the QSCM problem is an NP-hard problem.*

3.1.2 QSUM: The QoS-Constrained Sensing Utility Maximization Problem

In QSUM, the utility is defined as the difference between the benefits attributed to the spatiotemporal coverage of hotspot regions and the total cost resulting from the sensing campaign. Different hotspots are at different levels of interest, and their spatiotemporal coverage ratios should be assigned with different weights. Hence, the sensing utility is defined as follows:

$$U(\vec{\mathbf{x}}) = \sum_{j \in M} w_j | \bigcup_{x_i=1, i \in N} tr_{i,j} | - \sum_{i \in N} (d_i + \sum_{j \in M} |tr_{i,j}| b_i) x_i,$$

where w_j denotes the utility weight associated with ST_j, the condition $(x_i = 1, i \in M)$ denotes the index of selected participant p_i, and $\bigcup$ is the disjoint set union of all the overlapped spatiotemporal blocks between tr_i and ST_j. The QoS criteria are defined in the same manner as with QSCM. The objective of QSUM is to find a subset of participants to maximize the utility while ensuring the spatiotemporal coverage ratio associated with each hotspot is above the corresponding QoS-designated threshold, and it can be formulated as below:

$$\begin{aligned} &\max\ U(\vec{\mathbf{x}}) \\ s.t.\quad &QoS_j(\vec{\mathbf{x}}) \geq r_j, \forall j \in M \end{aligned}$$

It can be seen that QSUM considers the utility of sensing coverage by computing the difference between the total benefits and total costs in the objective function with the same QoS constraints as in QSCM, and it can be proved QSUM is also an NP-hard problem in the same manner as in Theorem 1.

3.2 The QoS-Constrained Greedy Approaches

This section presents greedy approaches to achieve the goals of cost minimization or utility maximization while satisfying the QoS constraint. The algorithms are

analyzed accordingly.

3.2.1 The QoS-Constrained Greedy Algorithm for Cost Minimization

In order to achieve the minimal cost with QoS constraint, a greedy approach is proposed hereby to carry out the process of participant recruitment. It is meant to select the participant with the maximum ratio of marginal benefit to the cost from the pool of remaining unselected participants in each iteration of selection. For each participant p_i, its total number of spatiotemporal blocks overlapped with hotspots is $\sum_{j \in M} |ST_j \bigcap tr_i|$, and the associated cost is $c_i = d_i + \sum_{j \in M} |ST_j \bigcap tr_i| b_i$. As a result, its unit cost can be expressed as $uc_i = \frac{c_i}{\sum_{j \in M} |ST_j \bigcap tr_i|}$. In addition, we define a function $\psi : I \rightarrow \vec{\mathbf{x}}$ to map the collection of participant indexes to a n-dimensional $(0, 1)$ vector $\vec{\mathbf{x}}$, such that the q-th element x_q in $\vec{\mathbf{x}}$ is set as 1 if $q \in I$, and 0 otherwise. The algorithm is detailed in **Algorithm 1**.

3.2.2 The Greedy QoS-Constrained Utility Maximization Algorithm

A similar greedy approach is proposed to achieve the maximum utility with QoS constraint in the process of participant recruitment. It is meant to choose the participant with the maximum ratio of marginal benefit to the cost from the pool of remaining unselected participants in each iteration of selection. The algorithm is detailed in **Algorithm 2**. As both the two algorithms iterate through all the remaining participants in each round for no more than n iterations, their time complexity are both $O(n^2)$. The bound of the greedy algorithms can achieve $H(\Delta)$ approximation as shown in [33, 34] where Δ denotes the largest size of tr_i and $H(\Delta)$ is the Δ-th Harmonic number.

Algorithm 1: The QoS-Constrained Greedy Algorithm for Cost Minimization (QGA-CM)

$I^* \leftarrow \emptyset, tr^* \leftarrow \emptyset, I \leftarrow N = \{1, 2, \cdots, n\}$, $ST \leftarrow \bigcup_{j \in M} ST_j$ and $ST^* \leftarrow ST$, ;

if *there exists a hotspot* S_j *such that* $\frac{|\bigcup_{i \in N} tr_{i,j}|}{|ST_j|} < r_j$ **then** `// the pool of participants cannot satisfy the QoS constraint`

  $I^* \leftarrow \emptyset$;

else

  while $|I| > 0$ *and there exists a hotspot* S_j *such that* $\frac{|ST_j \cap tr^*|}{|ST_j|} < r_j$ **do**

    $i^* \leftarrow arg\,min_{i \in I} \frac{d_i + |\bigcup_{i \in N} tr_{i,j}| b_i}{|ST^* \bigcap tr_i|}$;

    $I \leftarrow I \setminus \{i^*\}, I^* \leftarrow I^* \bigcup \{i^*\}$;

    $tr^* \leftarrow tr^* \bigcup tr_{i^*}, ST^* \leftarrow ST^* \setminus \{tr^*\}$;

  end

end

$\vec{\mathbf{x}}_{best} \leftarrow \psi(I^*)$;

return $\vec{\mathbf{x}}_{best}$

3.3 The QoS-Constrained Hybrid Approaches

In this section, we present hybrid approaches to fulfil the QoS requirements for task assignment. They apply Bees algorithm [35–37] on top of the participant selection results derived from previous greedy approaches.

Algorithm 2: The QoS-Constrained Greedy Algorithm for Utility Maximization (QGA-UM)

$I^* \leftarrow \emptyset, tr^* \leftarrow \emptyset, I \leftarrow N = \{1, 2, \cdots, n\}$, $ST \leftarrow \bigcup_{j \in M} ST_j$ and $ST^* \leftarrow ST$;

if *there exists a hotspot* S_j *such that* $\frac{|\bigcup_{i \in N} tr_{i,j}|}{|ST_j|} < r_j$ **then** `// the pool of participants cannot satisfy the QoS constraint`

  $I^* \leftarrow \emptyset$;

else

  while $|I| > 0$ *and there exists a hotspot* S_j *such that* $\frac{|ST_j \cap tr^*|}{|ST_j|} < r_j$ **do**

    $ST'_j \leftarrow ST_j \setminus tr^*$;

    $i^* \leftarrow arg\, max_{i \in I} \frac{\sum_{j \in M} w_j |ST'_j \bigcap tr_i| - c_i}{|ST^* \bigcap tr_i|}$;

    $I \leftarrow I \setminus \{i^*\}, I^* \leftarrow I^* \bigcup \{i^*\}$;

    $tr^* \leftarrow tr^* \bigcup tr_{i^*}, ST^* \leftarrow ST^* \setminus tr^*$;

  end

end

$\vec{\mathbf{x}}_{best} \leftarrow \psi(I^*)$;

return $\vec{\mathbf{x}}_{best}$

3.3.1 The QoS-Constrained Greedy Bees Algorithm for Cost Minimization (QGBA-CM)

In this subsection, we propose a QoS-Contrained Greedy Bees Algorithm to minimize the cost for sensing task assignment. In this algorithm, the employed bees, onlooker bees and scout bees cooperatively forage the optimal solution in the solution space of $\vec{\mathbf{x}}$ within an acceptable time period. In the first step, the algorithm initiates a randomly distributed population of food source positions (i.e., possible

solutions) $\vec{\mathbf{x}}_\alpha (1 \leq \alpha \leq E\}$ in the solution space, where $\vec{\mathbf{x}}_\alpha$ is a n-dimensional vector and E is the maximum number of employed bees. Specifically, we initiate one of the positions with the resulting vector derived from QGACM, and then apply the random global search in the solution space to select no more than $E-1$ food positions for the employed bees with no more than n^ε attempts to avoid infinite searching loops, where ε is an adjustable parameter. Next, the algorithm starts a repeated cycle

Algorithm 3: The QoS-Constrained Greedy Bees Algorithm for Cost Minimization (QGBA-CM)

1 Apply QGA-CM and derive the participant selection result $\vec{\mathbf{x}}_{best}$;
2 $cycle \leftarrow 1, \vec{\mathbf{x}}_e \leftarrow \{\vec{\mathbf{x}}_{best}\}, \vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}_e, \alpha \leftarrow 2, k_1 \leftarrow 1$;
3 **while** $\alpha \leq E$ *and* $k_1 \leq n^\varepsilon$ **do** `// the employed bees' global search`
4 randomly generates a n-dimensional $(0,1)$ vector $\vec{\mathbf{x}}_\alpha$;
5 **if** $\vec{\mathbf{x}}_\alpha \notin \vec{\mathbf{x}}^*$ *and* $QoS_j(\vec{\mathbf{x}}_\alpha) \geq r_j,\ \forall j \in M$ **then**
6 $\alpha \leftarrow \alpha + 1$;
7 **end**
8 $\vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}^* \bigcup \{\vec{\mathbf{x}}_\alpha\}$,$\vec{\mathbf{x}}_e \leftarrow \vec{\mathbf{x}}_e \bigcup \{\vec{\mathbf{x}}_\alpha\}$, $k_1 \leftarrow k_1 + 1$;
9 **end**
10 Continued;

as follows: each employed bee first makes modifications on the its assigned position (i.e., solution), and their memory are shared with the onlooker bees. Accordingly, the $L(L > E)$ onlooker bees explore the neighbourhood of the food source positions where each food position is selected with probabilities proportional to the corresponding nectar amount (i.e., the reciprocal of cost), and they look for new positions that can meet QoS requirements using the local search algorithm. The local search algorithm uses $F_{flip}(\vec{\mathbf{x}}, \epsilon)$ as shown below to derive new vectors (i.e., food positions) in

Algorithm 4: The QoS-Constrained Greedy Bees Algorithm for Cost Minimization (QGBA-CM) Continued

repeat
  foreach *employed bee* $\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e$ **do**
    $L_\alpha \leftarrow \frac{C^{-1}(\vec{\mathbf{x}}_\alpha)}{\Sigma_{1\leq\alpha'\leq E} C^{-1}(\vec{\mathbf{x}}_{\alpha'})} \cdot L, \beta \leftarrow 1, k_2 \leftarrow 1$;
    while $\beta \leq L_\alpha$ *and* $k_2 \leq n^\varepsilon$ **do** `// the onlooker bees' local search`
      $\vec{\mathbf{x}}_{\alpha,\beta} \leftarrow F_{flip}(\vec{\mathbf{x}}_\alpha, \epsilon)$;
      if $\vec{\mathbf{x}}_{\alpha,\beta} \notin \vec{\mathbf{x}}^*$ *and* $QoS_j(\vec{\mathbf{x}}_{\alpha,\beta}) \geq r_j, \forall j \in M$ **then**
        $\beta \leftarrow \beta + 1, \vec{\mathbf{x}}_\alpha \leftarrow arg\, min_{\vec{\mathbf{x}}}\{C(\vec{\mathbf{x}}_{\alpha,\beta}), C(\vec{\mathbf{x}}_\alpha)\}$;
      end
      $\vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}^* \bigcup \{\vec{\mathbf{x}}_{\alpha,\beta}\}$, $k_2 \leftarrow k_2 + 1$;
    end
  end
  Generate a scout bee $\vec{\mathbf{x}}'$as in the global search; $\vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}^* \bigcup \{\vec{\mathbf{x}}'\}$;
  if $\max\{\{C(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\} > C(\vec{\mathbf{x}}')$ **then**
    $\vec{\mathbf{x}}_{\alpha-max} \leftarrow arg\, max\{\{C(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}, \vec{\mathbf{x}}_{\alpha-max} \leftarrow \vec{\mathbf{x}}'$;
  end
  if $C(\vec{\mathbf{x}}_{best}) > \min\{\{C(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}$ **then**
    $\vec{\mathbf{x}}_{best} \leftarrow arg\, min_{\vec{\mathbf{x}}}\{\{C(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}$;
  end
  $cycle \leftarrow cycle + 1$;
until *the stopping conditions are satisfied*;
return $\vec{\mathbf{x}}_{best}$

the neighbourhood of $\vec{\mathbf{x}}$:

$\vec{\mathbf{x}}_f \leftarrow F_{flip}(\vec{\mathbf{x}}, \epsilon)$**:** the onlooker bee randomly selects ϵ' elements in $(0, 1)$ vector $\vec{\mathbf{x}}$ and flips them where $0 \leq \epsilon' \leq \epsilon$.

To avoid infinite search loops, all visited positions are recorded in the tabu list $\vec{\mathbf{x}}^*$,and the number of attempts is limited to n^{ε}.

After the local search, if the new positions bring more nectar amount (i.e.,less cost), then the employed bees' memory will be updated with the new positions of onlooker bees. Next, a scout bee randomly selects a new food source position to replace one of the previous positions which brings the highest cost. The search loops stop if two conditions are satisfied: i) the number of iterations has reached Maximum Number of Cycles (MNC); ii) the resulting cost remains unchanged for n_{stable} iterations. The algorithm is detailed in **Algorithm 3-4**.

3.3.2 The QoS-Constrained Greedy Bees Algorithm for Utility Maximization (QGBA-UM)

Similarly, we apply the QoS-constrained Bees Algorithm to achieve maximum utility, which is detailed in **Algorithm 5**. QGBA-UM differs from QGBA-CM in that the food positions of the employed bees are selected with probabilities proportional to the corresponding utilities rather than the reciprocal of cost, and the employed bees update their memory when the new positions bring higher utility rather than lower cost. In addition, the scout bee updates one of the employed bees' position with the lowest utility rather than the highest cost. Their remaining parts are the same. As there exit no more than MNC iterations, the time complexity of the hybrid approaches are $O(n^2 + MNC \cdot E \cdot n^{\varepsilon})$.

Algorithm 5: The QoS-Constrained Greedy Bees Algorithm for Utility Maximization (QGBA-UM)

Apply QGA-UM and get corresponding result $\vec{\mathbf{x}}_{best}$;

$cycle \leftarrow 1, \vec{\mathbf{x}}_e \leftarrow \{\vec{\mathbf{x}}_{best}\}, \vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}_e$;

the employed bees conduct the same global search and update $\vec{\mathbf{x}}^*, \vec{\mathbf{x}}_e$ as in QGBA-CM;

repeat

  foreach *employed bee* $\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e$ **do**

    $L_\alpha \leftarrow \frac{U(\vec{\mathbf{x}}_\alpha)}{\Sigma_{1 \le \alpha' \le E} U(\vec{\mathbf{x}}_{\alpha'})} \cdot L$;

    the onlooker bees conduct the same local search and update $\vec{\mathbf{x}}_\alpha$ using $\vec{\mathbf{x}}_{\alpha,\beta}$ with higher utility as in QGBA-CM;

  end

  Generate a scout bee $\vec{\mathbf{x}}'$as in the global search;

  $\vec{\mathbf{x}}^* \leftarrow \vec{\mathbf{x}}^* \bigcup \{\vec{\mathbf{x}}'\}$;

  if $\min\{\{U(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\} < U(\vec{\mathbf{x}}')$ **then**

    $\vec{\mathbf{x}}_{\alpha-min} \leftarrow arg\,min\{\{U(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}$;

    $\vec{\mathbf{x}}_{\alpha-min} \leftarrow \vec{\mathbf{x}}'$;

  end

  if $U(\vec{\mathbf{x}}_{best}) < \max\{\{U(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}$ **then**

    $\vec{\mathbf{x}}_{best} \leftarrow arg\,max_{\vec{\mathbf{x}}}\{\{U(\vec{\mathbf{x}}_\alpha)\}_{\vec{\mathbf{x}}_\alpha \in \vec{\mathbf{x}}_e}\}$;

  end

  $cycle \leftarrow cycle + 1$;

until *the stopping conditions are satisfied*;

return $\vec{\mathbf{x}}_{best}$

3.4 Performance Evaluation

In this section, we implemented both the greedy approaches and the hybrid approaches in QSCM and QSUM, and evaluated their performance using metrics including sensing cost, sensing utility, spatiotemporal coverage ratio and the number of participants. All the simulations ran on a Windows machine with Intel(R) Xeon(R) CPU and 4 GB memory.

We assume the whole region of size $1000m \times 1000m$ is griditized into spatial blocks of size $20m \times 20m$. Three hotspots are distributed in the whole region, and their projections in the spatiotemperal space consist of $60 \times 4, 60 \times 5, 60 \times 6$ spatiotemporal blocks similar to Figure 3.2. The spatiotemporal coverage ratio of each participant's trace tr_i over each hotspot ST_j is uniformly distributed over $[0, 16\%]$. The number of participants n varies from 10 to 30 with the increment of 10. We also assume the participant p_i's static cost d_i is uniformly distributed over $[1, 5]$ and its unit cost per block b_i is uniformly distributed over $[1, 3]$, while the utility weight w_j is uniformly distributed over $[4, 10]$. In the hybrid algorithm QGBA-CM and QGBA-UM, we set $\epsilon = 3$,$\varepsilon = 3$,$n_{stable} = 3$, $MCN = 6$,$E = 10$, and $L = 50$. For simplicity, it is assumed that the sensing applications have the same QoS requirement r for different hotspots where r ranges from 10% to 100% with the increment of 10%. We generate 50 instances for each set of r and n and derive the graphs with error bars.

From Figure 3.3 we can learn that the hybrid approaches can derive better results than greedy approaches with respect to both cost and utility. Specifically, the hybrid approaches achieve the same results as the optimal solution when $n = 10$ as shown in Figure 3.3(a) and Figure 3.3(d). It can be seen that the results of hybrid approaches and greedy approaches are getting closer to the optimal results as r approaches 100%, because the solution space shrinks with the increase of r. In addition, the gaps

between the greedy approaches' results and the optimal results become larger due to the growing solution space as n increases from 10 to 30, while the hybrid approaches' results keep close to the optimal results.

3.5 Discussions on Privacy-Preserving Task Assignment in MCS

In some cases, the CP can select the MDOs/participants with profiles that meet the requirements of sensing service queries without knowing the exact numeric values of their profile data, thereby offering a layer of privacy protection over the MDOs/-participants. Li *et al.* [38] proposed to match the profiles of two users based on a privacy-preserving computation of the intersection of their attribute sets, and the techniques can be used to find appropriate MDOs/participants with profiles to match the requirements of sensing service queries for sensing task assignment in MCS. In addition, Ghinita *et al.* [39] designed a new method to compare two integers without disclosing their values based on Paillier cryptosystem [40]. Accordingly, the CP can divide a region into multiple convex polygon cells of which each side of each polygon cell is represented by a line equation. As such, the CP can send the line equations of a target cell in encrypted form to the MDOs/participants, while the MDOs/participants perform computations on the encrypted line equations with their locations and send them back to the CP. Consequently, the CP can derive the signs of line inequalities with the input of the locations of MDOs/participants and deduce which of them lie inside the target cell without knowing their exact location coordinates. It can be learned that Paillier cryptosystem is a very useful technique for privacy-preserving task assignment in MCS. The details will be discussed in the next chapter.

3.6 Related Work

Substantial research has been done for resource allocation and task assignment in traditional sensor networks. A couple of efficient near-optimal algorithms are provided in [41–43] to achieve a complete spatial coverage of the sensing field in wireless sensor networks. Kallitsis *et al.* [44] constructed a resource allocation model based on pricing scheme to maximize the provider's utility with QoS requirements in network delay. Koulali [45] *et al.* presented an optimal distributed relay selection policy to optimize duty-cycling sensor's energy consumption with QoS constraints on transmission delay. Bagaria *et al.* [46] proposed a polynomial-time approximation algorithm to maximize the lifetime of coverage of targets in a wireless sensor network with battery-limited sensors. Two placement algorithms CSD-DC and CSD-NDC [47] are presented to find a deployment curve to minimize the number of sensors while ensuring the barrier coverage. Chen *et al.* [48] offered a novel algorithm $maxL\text{-}minE$ to find a landmark placement pattern to minimize the maximum localization error and demonstrated its improved performance using Wifi and Zigbee networks in real building environment. All of the work above are designed for static sensors while less work have been done for mobile sensing. Reddy *et al.* [8] proposed a recruitment framework to maximize the coverage-associated utility with budget constraint along with reputation-based assessment. Yang *et al.* [49] designed incentive mechanisms in platform-centric and user-centric models for mobile phone sensing. OptiMoS [32] devises a two-tier mobile sensing model to balance sensor coverage and energy cost. Unlike previous work, we identify the task assignment problems with QoS constraints in terms of spatiotemporal coverage and propose efficient hybrid methods on top of the greedy algorithm and bees algorithm to provide better performance.

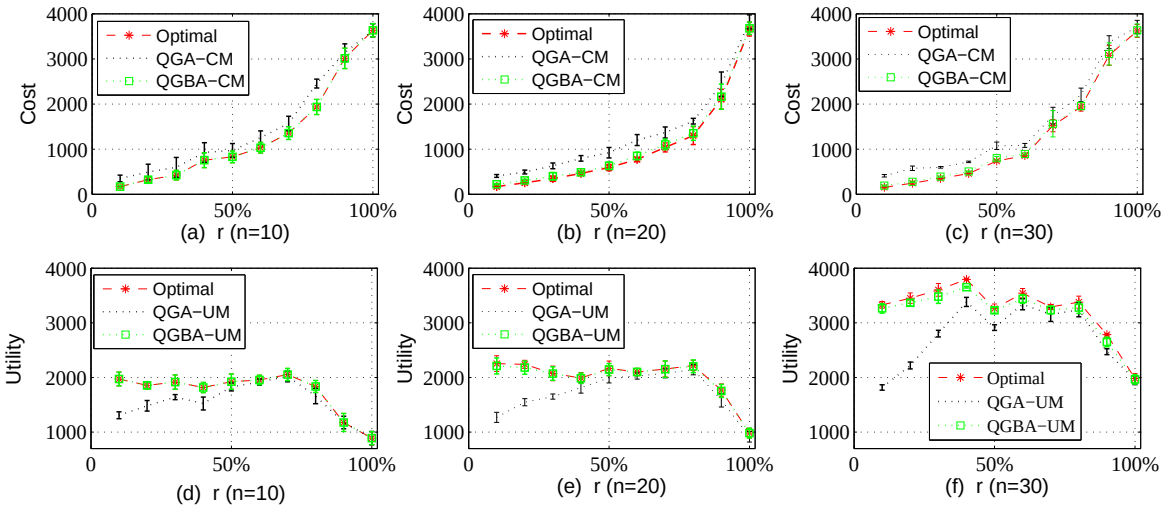

Figure 3.3: Evaluation Results of Different Approaches

3.7 Conclusion

The spatiotemporal coverage of the mobile sensing devices over the target areas during time periods of interest has direct impact on the data quality and quantity in MCS. We identify the problems of sensing cost minimization and utility maximization with QoS constraints to fulfil different requirements of sensing applications, and propose greedy approaches as well as heuristic hybrid approaches with greedy algorithm and Bees algorithms to address them. Our evaluation results show that our hybrid approaches approximate the optimal solutions when the solution space is small, and the results of hybrid approaches are more close to the optimal solutions than the greedy approaches when the size of solution space grows large.

Chapter 4

PRIVACY-PRESERVING SENSING QUERY AND MULTI-PARTY SENSING COMPUTATION

The existing dynamics in MCS, which have found a growing popularity in the real-world applications, is a promising direction to generate physical-world data and share knowledge for the benefits of the academia and the industry. With the advent of MCS, it is possible that people's history of past activities could be recorded each second with high accuracy and inspected by others, more and more attention have been given to privacy leakage. The profiles and the daily traces of MDOs, which collect and submit sensing data, could be harnessed by adversaries without permissions. At the same time, the sensing queries of the SSCs also expose their interest. All these privacy leakage, if not properly resolved, would impede the development of MCS.

To address the issues stated above, PP-MCS is proposed to protect the privacy of MDOs and SSCs without relying on any online third party. It will exploit homomorphic encryptions to aggregate the private data of MDOs without revealing MDOs' individual data records. An encryption scheme is defined as an additive homomorphic one if and only if

$$E(m_1) \oplus E(m_2) = E(m_1 + m_2)$$

where $\oplus$ is an operator, and m_1 and m_2 are the numeric values to be encrypted. Similarly, an encryption scheme is defined as a multiplicative homomorphic one if and only if

$$E(m_1) \otimes E(m_2) = E(m_1 * m_2)$$

where $\otimes$ is an operator, and m_1 and m_2 are the numeric values to be encrypted.

Specifically, we rely on Paillier cryptography [40] of which the most expensive operations are encryption and decryption, while the operations with ciphertexts are relatively inexpensive [39, 40]. Given the ciphertexts $E(m_1)$ and $E(m_2)$, the public key $pk = \{g, N\}$ and the secret key $sk = \{\lambda, \mu\}$, the sum of $m_1 + m_2$ can be derived by computing

$$D(E(m_1, pk) \oplus E(m_2, pk), sk) = (m_1 + m_2) \, mod \, N$$

Additionally, we can derive the product $r * m$ from the ciphertext operations based on the multiplicative homomorphic property as follows:

$$D(E(m, pk)^r, sk) = r * m \, mod \, N$$

where r is a random number.

4.1 Privacy-Preserving Sensing Query

4.1.1 System Model and Attack Model

Assume the SSCs request the summation/average values of the sensing readings of some of the MDOs distributed across a large geographical area through the cloud mediator. The system model is illustrated in Figure 4.1 and it consists of five parties:

- *Sensing Service Consumers* (*SSCs*): The SSCs are the sensing service consumers to issue sensing service requests to the Cloud Mediator for the summation/average values of the sensing readings of some of the MDOs in a target region. To preserve the privacy of the sensing requests, each SSC specifies a cloaked ID set where the IDs of the real target MDOs are included, and generates its own public/private key pair based on Paillier cryptosystem to encrypt its requests.

- *Cloud Mediator*: The Cloud Mediator consists of both the SSP and the CP where the MDOs register and get their IDs maintained. It forwards the encrypted sensing request to the MDOs specified in the cloaked ID set. Upon receiving the sensing results, the Cloud Mediator aggregates the encrypted sensing readings from the MDOs and forwards the aggregation result back to the SSC.

- *Mobile Device Owner* (*MDO*): The MDOs encrypt the sensing readings upon receiving the encrypted sensing requests and send them back to the Cloud Mediator for aggregation.

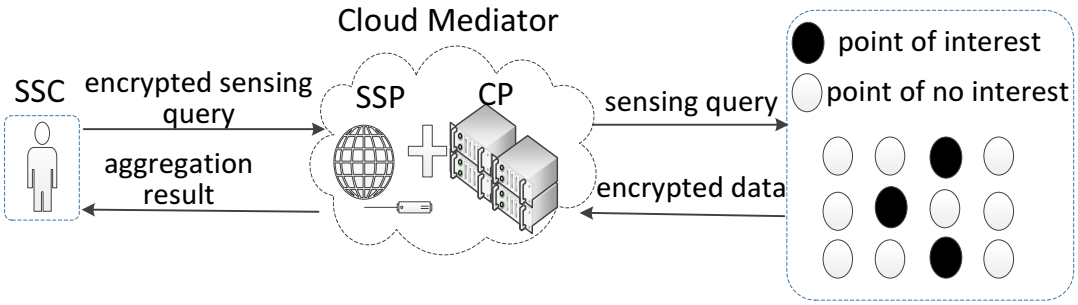

Figure 4.1: Privacy-Preserving Sensing Query in MCS

We make the following security assumptions for the attack model: 1)The Cloud Mediator and the MDOs are semi-honest attackers. In other words, they honestly follow the procedures while they are interested in which MDOs that the SSCs try to retrieve sensing readings from; 2) the data privacy of the MDOs are not concerned in this scenario; that is to say, the identity of the MDO which a specific data record belongs to is not important.

4.1.2 Construction

In this subsection, we describe how to construct our privacy-preserving sensing query scheme. We assume the Cloud Mediator hosts the registry including the sensing categories, geographic locations and identification set $\mathbb{I}$ of all the MDOs for public use where there exists $|\mathbb{I}| = n$. Accordingly, the SSC selects the target MDOs $\mathbb{I}^{(t)}$ with appropriate sensing categories and geo-locations as well as a cloaking identification set $\mathbb{I}^{(*)}$ of MDOs to hide the real target MDOs $\mathbb{I}^{(t)}$, where there exist $|\mathbb{I}^{(t)}| = n^{(t)} \leq |\mathbb{I}^*| = n^* \leq n$. Each MDO_i holds the sensing reading d_i where there exist $d_i < d_{max}$ $\forall i \in \mathbb{I}$. Our scheme consists of three algorithms including Privacy-Preserving Query Generation, Response Generation, Response Aggregation and Response Retrieval as shown in **Algorithm 6** — **Algorithm 9**:

Algorithm 6: Privacy-Preserving Query Generation (SSC)

---

1 The SSC randomly selects two large primes p, q such that $N = pq > nd_{max}$ and derives $\lambda = lcm(p-1, q-1)$;

2 The SSC also chooses a random $g \in \mathbb{Z}^*_{N^2}$, such that $gcd(L(g^\lambda \ mod \ N^2), N) = 1$, where $L(x) = (x-1)/N$;

3 For each $i \in \mathbb{I}^{(*)}$, the SSC picks a random integer $r_l \in \mathbb{Z}^*_{N^2}$ and computes

$$c_i = \begin{cases} E(1, pk) = g^1 r_i^N (mod \ N^2), \text{ if } i \in \mathbb{I}^{(t)} \\ E(0, pk) = g^0 r_i^N (mod \ N^2), \text{ if } i \notin \mathbb{I}^{(t)} \end{cases}$$

4 The SSC transmits $Q = \{\mathbb{I}^{(*)}, c_1, c_2, \cdots, c_{n^*}, pk\}$ to the Cloud Mediator;

Correctness: For the target $MDO_i (i \in \mathbb{I}^{(t)})$, its sensing reading would be embedded into $C_i = c_i^{d_i} = (g^1 r_i^N (mod \ N^2))^{d_i} = g^{d_i} (r_i^{d_i})^N (mod \ N^2)$. On the contrary, for the $MDO_i (i \notin \mathbb{I}^{(t)})$, its sensing reading would be canceled out by computing

Algorithm 7: Response Generation (Cloud Mediator + MDOs)

1 Upon receiving the encrypted sensing request Q from the SSC, the Cloud Mediator looks up in the registry based on $\mathbb{I}^{(*)}$, and forwards (c_i, pk) to MSC_i for all $i \in \mathbb{I}^{(*)}$;

2 On receiving (c_i, pk), each MDO_i encrypts its sensing reading d_i by computing $C_i = c_i^{d_i}$, and sends it back to the Cloud Mediator as the response.

Algorithm 8: Response Aggregation (Cloud Mediator)

1 After receiving the response $\{C_i\}_{i \in \mathbb{I}^{(*)}}$ from the all the MDOs, the Cloud Mediator computes the aggregation result $C = \prod_{i \in \mathbb{I}^{(*)}} C_i$;

2 The Cloud Mediator forwards C to the SSC.

Algorithm 9: Response Retrieval (SSC)

1 After receiving the aggregation result C from the Cloud Mediator, the SSC performs the decryption to get the summation of the sensing reading values of the target MDOs by computing
$sum = Decrypt(C, sk) = \dfrac{L(C^{\lambda} \ mod \ N^2)}{L(g^{\lambda} \ mod \ N^2)} \ mod \ N$;

2 The SSC further computes the average value of the sensing readings of the target MDOs by computing $ave = sum/n^{(t)}$.

$C_i = c_i^{d_i} = (g^0 r_i^N (mod\ N^2))^{d_i} = g^0 (r_i^{d_i})^N (mod\ N^2)$. Hence, the aggregation of the sensing reading values of the target MDOs can be computed by

$$C = \prod_{i \in \mathbb{I}^{(*)}} C_i = \prod_{i \in \mathbb{I}^{(t)}} C_i \cdot \prod_{i \in \mathbb{I}^{(*)}, i \notin \mathbb{I}^{(t)}} C_i = g^{\sum_{i \in \mathbb{I}^{(t)}} d_i} (\prod_{i \in \mathbb{I}^{(*)}} r_i^{d_i})^N (mod\ N^2)$$

As a result, the summation of the sensing reading values of the target MDOs can be derived by $sum = Decrypt(C, sk)$ without any sensing reading values of other MDOs in the identification set included.

Security Analysis:The Paillier cryptosystem offers semantic security [39, 40], which is secure against chosen plaintext attacks. Therefore, given the public key pk of the Paillier cryptosystem, the Cloud Mediator and the MDOs can hardly differentiate between the ciphertexts $c_1, c_2, \cdots, c_{n^*}$ with encrypted "0" or "1". Consequently, the Cloud Mediator and the MDOs cannot distinguish the target MDOs $\mathbb{I}^{(t)}$ from the cloaked identification set $\mathbb{I}^{(*)}$. As a result, the privacy of the sensing query of the SSC are protected.

4.2 Privacy-Preserving Multi-Party Sensing Computation

In this section, we first present two techniques as the building blocks to preserve the data privacy among two and multiple parties, respectively. Subsequently, we construct a privacy-preserving scheme to protect MDOs' data privacy in the procedure of multi-party sensing computation.

4.2.1 Building Block I: Privacy-Preserving Comparision of Two Integers

The millionaire's problem has bee proposed and addressed by Yao [50]. Its goal is to solve the inequality $\Delta_1 \geq \Delta_2$ without revealing the actual values of Δ_1 owned by Party A and Δ_2 owned by Party B, respectively. Ghinita *et al.* [39] proposed an easy solution to this problem based on Paillier cryptosystem with order of N. It assumes

that $\Delta_1, \Delta_2 \in \mathbb{Z}_{N'}$ where $N' \ll (N-1)/2$. Party A generates the public/private key pair pk/sk, and sends $E(N-\Delta_1, pk)$ to Party B. Accordingly, B generates a random integer $r \in \mathbb{Z}_M^*$ as a blinding factor where $M \leq \lfloor \frac{N-1}{2N'} \rfloor$ and computes

$$(E(N-\Delta_1, pk) \oplus E(\Delta_2, pk))^r = E(N+\Delta_2-\Delta_1, pk)^r = E(r(N+\Delta_2-\Delta_1), pk)$$

and sends it back to A. Subsequently, A decrypts this message and derives $r(N+\Delta_2-\Delta_1)$. If $\Delta_2-\Delta_1 \geq 0$, then $r(N+\Delta_2-\Delta_1) \in I_1 = \{0, 1, \cdots, M \cdot N'\}$, otherwise $r(N+\Delta_2-\Delta_1) \in I_2 = \{N-M \cdot N', \cdots, N-1\}$ where $I_1 \bigcap I_2 = \emptyset$

Ghinita *et al.* [39] pointed out this approach is feasible in real-world settings. It suggested the magnitude of modulus N should be at least 768 bits large to guarantee security strength, and values of Δ_1 and Δ_2 can be represented by 64 bits, which suffice in most real-world applications. At the same time, the random blinding factor domain will be bounded by $M = \frac{2^{768}}{2 \cdot 2^{64}}$ with the order of 2^{700}, which is sufficiently large to provide a strong degree of security.

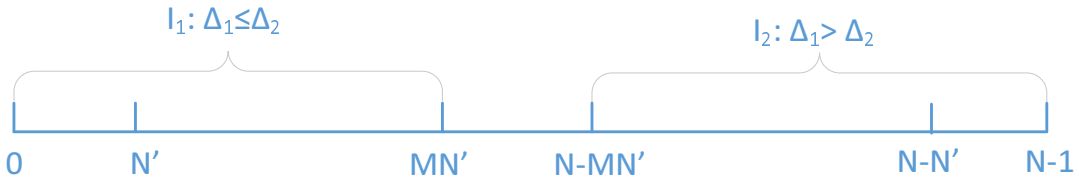

Figure 4.2: Privacy-Preserving Integer Comparison in MCS

4.2.2 Building Block II: Secret Sharing Among Distributed Multiple Parties

Chase *et al.* [51] proposed a secret-sharing technique among n distributed parties with $n-2$ collusion-resistance, which implies at least 2 out of n parties are outside the collusion group. It assumes each pair of parties (i, j) share a secret s_{ij} where $i, j \in \mathbb{Z}_N^*$. As such, each party i generates a function $P(i) = \sum_{j<i, j \in \mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j \in \mathbb{Z}_N^*} s_{ij}$ as the blinding value such that there exists $\sum_{i \in \mathbb{Z}_N^*} F(i) = 0$. In this manner, any adversary

who knows the secrets of no more than $n-2$ parties cannot derive the secrets of the remaining parties. This technique can be used to protect the data privacy in the process of multi-party computation as detailed in the next section.

4.2.3 System Model and Attack Model

In this subsection, we only concern about the data privacy of the SSC and the MDOs. Assume a SSC has a baseline value d, and some MDOs $\mathbb{I} = \{0, 1, \cdots, n\}$ have the sensing reading $\{d_i\}_{i\in\mathbb{I}}$ and their absolute differences from d are $\{\delta_i\}_{i\in\mathbb{I}} = \{|d_i - d|\}_{i\in\mathbb{I}}$ where $|\mathbb{I}| = n$. The SSC is concerned about which MDO_i have the absolute difference between the sensing reading d_i and d above a certain proportion p_1/p_2 of the summation of the absolute differences between all MDOs' sensing reading and d where $p_1, p_2 \in \mathbb{Z}$ and $0 < p_1 < p_2$, such that $\Delta_i = |d_i - d| \geq \frac{p_1 \sum_{i\in\mathbb{I}} \Delta_i}{p_2}$, and which MDO_i have the sensing reading d_i with the absolute difference $\Delta d_i = |d_i - d| < \frac{p_1 \sum_{i\in\mathbb{I}} \Delta_i}{p_2}$ in Figure 4.2 without revealing d to any parties including the Cloud Mediator and all of the MDOs. When $p_1 = 1$ and $p_2 = n$, it is reduced to the special case that the SSC is interested in learning which MDO_i have the sensing reading d_i with the absolute difference above the average difference such that $\Delta_i \geq \frac{\sum_{i\in\mathbb{I}} \Delta_i}{n}$, and which MDO_i have the sensing reading d_i with the absolute difference $\Delta_i < \frac{\sum_{i\in\mathbb{I}} \Delta_i}{n}$.

We make the following security assumptions for the attack model: 1)The Cloud Mediator and at most $n-2$ out of n MDOs are semi-honest attackers. In other words, they honestly follow the procedures, while the malicious MDOs are interested in other MDOs' sensing readings and the SSC's baseline value d, the SSC is interested in all MDOs' sensing readings, and the Cloud Mediator is interested in both MDOs' sensing readings and the SSC's baseline value d; ii) the privacy of the sensing query of the SSC is out of concern; iii)the MDOs are $n-2$ collusion-resistant; in other

words, at most $n-2$ out of n MDOs collude with the SSC and the Cloud Mediator to breach the data privacy of the honest MDOs. We believe this assumption is feasible, as most MDOs could be semi-honest while a few MDOs could be still honest in most application settings.

4.2.4 Protocol

Without loss of generality, we assume there exists n MDOs in the system and each MDO_i has a sensing reading $d_i \in [0, N']$. For each $d_i \in \mathbb{Z}_{N'}$, it can be converted to a binary vector $\vec{d_i} = \{\hat{d}_{i,0}, \hat{d}_{i,1}, \cdots, \hat{d}_{i,N'-1}\}$ where $\hat{d}_{i,k} = 1(k \in [0, d_i])$ and $\hat{d}_{i,k} = 0(k \in (d_i, N'-1])$. The SSC has a baseline value d, and it generates Paillier public key $pk = (N, g)$ and private key sk such that $nN' \ll N$. By the same token, d can also be converted to a binary vector $\vec{d} = \{\hat{d}_0, \hat{d}_1, \cdots, \hat{d}_{N'-1}\}$ where $\hat{d}_k = 1(k \in [0, d])$ and $\hat{d}_k = 0(k \in (d, N'-1])$. Similar to [52], we can derive the absolute difference between d_i and d by computing

$$\begin{aligned}\Delta_i &= |d_i - d| = \textstyle\sum_{k=0}^{N'-1} |\hat{d}_{i,k} - \hat{d}_i| = \sum_{k=0}^{N'-1} |\hat{d}_{i,k} - \hat{d}_i|^2 \\ &= \textstyle\sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2\sum_{k=0}^{N'-1} \hat{d}_{i,k}\hat{d}_i + \sum_{k=0}^{N'-1} \hat{d}_i^2 \\ &= \textstyle\sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2\vec{d_i}\vec{d} + \sum_{k=0}^{N'-1} \hat{d}_i^2\end{aligned}$$

Accordingly, the detailed operations of our protocol proceeds as follows:

I Query Generation The SSC generates sensing query as shown in **Algorithm 10** :

II Response Generation: The cloud Mediator multicasts $(pk, \{E(\hat{d}_k, pk)\}_{k\in[0,N'-1]})$ to all the MDOs, and the MDOs generate response as shown in **Algorithm 11** :

III Response Retrieval: For each $Resp_i \in Response(i \in \{1, 2, \ldots, n\})$, the SSC derives $result_i = r_i(\sum_{j\neq i, 1\leq k\leq n} \Delta_j - (n-1)\Delta_t)$ by decrypting $Resp_t$. If $result_t \in [0, nN']$, then it indicates $\Delta d_i = |d_i - d| < \dfrac{p_1 \sum_{i\in\mathbb{I}} \Delta_i}{p_2}$; otherwise if $result_t \in (N -$

Algorithm 10: Query Generation (SSC)

1 The SSC constructs a vector $\vec{d} = \{\hat{d}_0, \hat{d}_1, \cdots, \hat{d}_{N'-1}\}$ where $\hat{d}_k = 1(k \in [0, d))$ and $\hat{d}_k = 0(k \in [d, N'-1])$ based on its baseline value d;

2 The SSC picks a distinct random integer $r_k, r'_k \in \mathbb{Z}_N$ and computes $\prod_{k=0}^{N'-1} E(\hat{d}_k^2, pk) = E(\sum_{k=0}^{N'} \hat{d}_k^2, pk) = E(d, pk) = g^d r_k^N (mod\ N^2)$ and $E(\hat{d}_k, pk) = g^{\hat{d}_k} r_k'^N (mod\ N^2)$ for each $k \in [0, N'-1]$;

3 SSC transmits $(pk, E(d, pk), \{E(\hat{d}_k, pk)\}_{k\in[0,N'-1]})$ to the Cloud Mediator.

$nN'-1, N-1]$, then it indicates $\Delta d_i = |d_i - d| \geq \dfrac{p_1 \sum_{i\in\mathbb{I}} \Delta_i}{p_2}$.

We herein explain the correctness of **Algorithm 11** described above. Each MDO_i first computes the encrypted sensing reading difference $E(\Delta_i)$ without knowing the real values of d and Δ_i. Subsequently, the MDOs compute

$$\begin{aligned} E_{sum} &= E(\textstyle\sum_{i=1}^{n}(\Delta_i + P_i), pk) \\ &= E(\textstyle\sum_{i=1}^{n}(\Delta_i + \sum_{j<i, j\in\mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j\in\mathbb{Z}_N^*} s_{ij}, pk) \\ &= E(\textstyle\sum_{i=1}^{n} \Delta_i, pk) \end{aligned}$$

Afterwards, **Algorithm 11** takes n iterations to yield $Response = \{Resp_1, Resp_2, \ldots, Resp_n\}$. Specifically, for MDO_t, we have

$$\begin{aligned} Resp_t &= ((E_{sum})^{p_1} \cdot E(\Delta_t, pk)^{N-p_2})^{r_{t2}} \cdot E(r_{t1}, pk) \\ &= E(r_{t2}(p_1 \textstyle\sum_{i=1}^{n} \Delta_i + (N - p_2)\Delta_t) + r_{t1}), pk) \\ &= E(r_{t2}(p_1 \textstyle\sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}), pk) \end{aligned}$$

Accordingly, the SSC can derive $r_{t2}(p_1 \sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}$ by decrypting $Resp_t$ with the secret key sk and infers the sign of $r_{t2}(p_1 \sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}$. For $y = r_{t2}x + r_{t1}$ where $x \in \mathbb{Z}$, it is self-evident that if $x \geq 0$ then $y > 0$ and $x < 0$ then $y < 0$ because of $r_{t2} > r_{t1}$. Therefore, the SSC get the knowledge that if $\Delta_i = |d_i - d|$ is smaller than $\dfrac{p_1 \sum_{i=1}^{n} \Delta_i}{p_2}$ or not.

Algorithm 11: Sensing Reading Difference Generation

$Response = \{Resp_1, Resp_2, \ldots, Resp_n\} \leftarrow \{1, 1, \ldots, 1\}, E_{sum} \leftarrow 1$;

foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ **do**

The $MDO_i (1 \leq i \leq n)$ constructs $\vec{d_i} = (\hat{d}_{i,0}, \hat{d}_{i,1}, \cdots, \hat{d}_{i,N'-1})$ where $\hat{d}_{i,k} = 1(k \in [0, d_i))$ and $\hat{d}_{i,k} = 0(k \in [d_i, N'-1])$, and computes $E(d_i, pk) = E(\sum_{k=0}^{N'-1} \hat{d}_{ik}^2, pk) = \prod_{k=0}^{N'-1} E(\hat{d}_{ik}^2, pk) = g^{d_i} r_k^N (mod\, N^2)$;

The $MDO_i (1 \leq i \leq n)$ selects distinct random integers $\{r_{ik}\}_{k \in [0,d_i]}$ where $r_{ik} \in \mathbb{Z}_{N'}^*$ and computes $E(\vec{d} \cdot \vec{d_i}, pk) = E(\sum_{k=0}^{d_i} \hat{d}_k \hat{d}_{ik}, pk)$ $= \prod_{k=0}^{d_i} E(\hat{d}_k \hat{d}_{ik}, pk)\ mod\ N^2$;

The $MDO_i (1 \leq i \leq n)$ further computes $E(-2\vec{d} \cdot \vec{d_i}, pk) =$ $E((N-2)\vec{d} \cdot \vec{d_i}, pk) = E^{(N-2)}(\vec{d} \cdot \vec{d_i}, pk)$;

Consequently, each $MDO_i (1 \leq i \leq n)$ computes $E(\Delta_i) = |d_i - d| =$ $E(\sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2\vec{d_i}\vec{d} + \sum_{k=0}^{N'-1} \hat{d}_i^2) = E(d_i, pk) \cdot E^{(N-2)}(\vec{d} \cdot \vec{d_i}, pk) \cdot E(d, pk)$;

end

foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ **do**

The MDO_i shares a secret s_{ij} with MDO_j where $i, j \in [1, n], j \neq i$, $s_{ij} = s_{ji} \in \mathbb{Z}_N^*$, and derives $P_i = \sum_{j<i, j \in \mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j \in \mathbb{Z}_N^*} s_{ij}$;

The MDO_i computes $E_{sum} = E_{sum} \cdot E(\Delta_i, pk) \cdot E(P_i, pk)$;

end

The MDO_n broadcasts E_{sum} to all the MDOs;

foreach $i \in \{1, 2, \ldots, n\}$ **do**

The MDO_i picks a random integer $r_{i1}, r_{i2} \in \mathbb{Z}_{N'}^*$ where $r_{i2} > r_{i1} > 0$, and computes $Resp_t = ((E_{sum})^{p_1} \cdot E(\Delta_i, pk)^{N-p_2})^{r_{i2}} \cdot E(r_{i1}, pk)$;

end

The MDOs transmit *Response* to the SSC through the Cloud Mediator.

4.3 Approximate K-Nearest Neighbor with Privacy Preservation

The query for the k-nearest neighbors has significant implications in location-based sensing scenarios, and the K-Nearest Neighbor algorithm can be used in numerous fields of applications including classification and regression. In this section, we discuss how to identify the k-nearest neighboring MDOs around a specific benchmark location given by the SSC with the guarantee of the privacy window with the smallest size δ for any MDOs in the worst case. Specifically, the locations of any MDOs remain hidden, and the neighbors have no knowledge of their distance to the benchmark location, while the distances between the benchmark location and any of its neighbors are masked by privacy windows with the smallest size δ from the SSC.

4.3.1 System Model and Attack Model

The benchmark location provided by the SSC is $L_0 = (x_0, y_0)$, and the location of each neighbor $MDO_i (i \in [1, n])$ is denoted by $L_i = (x_i, y_i)$. Note the locations are derived from the latitude x_i and the longitude y_i which are both integers (e.g.,$(33.423856, 111.939575) \rightarrow (33423856, 111939575)$). Assume d^* is the distance threshold to separate MDO_0's actual k-nearest neighbors from other neighbors, and we define a distance window δ for privacy preservation, such that the neighbors falling within the distance range $[d^*, d^* + \delta]$ can be taken as alternative ones equivalent to some of the actual k-nearest neighbors. We believe this assumption holds as many location-based sensing service applications are tolerant to location deviations to some extent.

We make the following security assumptions for the attack model: 1)All the involving parties are semi-honest attackers. In other words, they honestly follow the procedures, but the SSC and the Cloud Mediator are interested in the neighbors'

locations, and the neighbors and the Cloud Mediator attempt to pinpoint the benchmark location; ii) the SSC should not know the exact distances of the neighbors to the benchmark location masked by the privacy window with the smallest size δ, and each neighbor cannot learn its distance to the benchmark location.

Without loss of generality, we assume there exist $x_i, y_i \in [1, N^{(L)}]$. For the sake of security, the Cloud Mediator generates $N^{(L)}, N^{(\gamma)}$ where $2(N^{(L)})^2 N^{(\gamma)} \leq \lfloor \frac{N-1}{2} \rfloor$. Accordingly, the Cloud Mediator calculates the distance array $D = \{D[k]\}_{k\in[1,K]}$ as shown in Figure 4.3 (i.e., $D = \{0, 1, \sqrt{2}, 2, \sqrt{5}, \sqrt{8}, 3, \sqrt{10}, \sqrt{13}, \cdots\}$) based on all the possible distances within the range $[0, \sqrt{2}N^{(L)}]$. At the same time, all the MDOs agree on the minimum size w_{min} (e.g., $w_{min} = 2$) of the privacy window and make it public. In each iteration of the process, the SSC privately updates the identity set $\mathbb{I}_l$, which includes the indices of $|\mathbb{I}_l|$ nearest neighbors where $|\mathbb{I}_l| \leq k$, and it also privately updates $|\mathbb{I}_r|$, which includes the identity set the indices of $|\mathbb{I}_r|$ nearest neighbors where $|\mathbb{I}_r| > k$. The implication of $N^{(\gamma)}$ is described in the following part. **Algorithm 12** elaborates how the SSC identifies k approximately nearest neighbors by narrowing down the privacy window and the pivot *pivot* via a divide-and-conquer approach. This algorithm adopts a binary search approach with complexity of $O(\log N)$. As D is an already-sorted array and the maximal distance would not exceed N based on the assumption, the complexity of this algorithm is $O(n \log N)$ where n is the number of MDOs involved.

4.3.2 Construction

Correctness of Algorithm 12: We hereby provides the proof of the correctness of **Algorithm 12**. Given $(E(x_0^2 + y_0^2, pk), E(x_0, pk), E(y_0, pk))$, MDO_i computes $E(-2x_0x_i, pk) = E((N-2)x_0x_i, pk) = E(x_0, pk)^{(N-2)x_i}$, $E(-2y_0y_i, pk) = E((N-2)y_0y_i, pk) = E(y_0, pk)^{(N-2)y_i}$, and derives $E(\Delta_i) = E((x_i - x_0)^2 + (y_i - y_0)^2, pk) =$

Algorithm 12: Approxiamte K-Nearest Neighbors with Privacy Window

The SSC generates Paillier public/private key $pk = \{g, n\}$, $sk = \{\lambda, \mu\}$ and $E = (E(x_0^2 + y_0^2, pk), E(x_0, pk), E(y_0, pk))$, and sends pk, E to Cloud Mediator ;

The Cloud Mediator initializes $\overrightarrow{sign} = \{sign_i\}_{i\in[1,n]} \leftarrow \{1, 1, \ldots, 1\}$, the loop index $t \leftarrow 0$, $\mathbb{I}_l \leftarrow \emptyset$ and $\mathbb{I}_r \leftarrow [1, n]$. It also calculates all the possible distances within the range $[1, 16N^{(\gamma)}(N^{(L)})^2]$ and derives the array $D = \{D[k]\}_{k\in[1,K]}$. Accordingly, it initializes the privacy window $\vec{w} = \{w_{left}, w_{right}\} \leftarrow [0, K-1]$ and publicizes $(pk, E, \overrightarrow{sign}, t, D, \vec{w})$;

while *the number of elements in* $\overrightarrow{sign}$ *equivalent to* 1 *is not* k **do**

  The Cloud Mediator sets $t \leftarrow t + 1$, $pivot^{(t)} \leftarrow \lfloor \frac{w_{left} + w_{right}}{2} \rfloor$;

  if $|\mathbb{P}| \geq 1$ and $Diff_{min}(pivot^{(t)}, \mathbb{P}) \leq w_{min}$ **then** break **end**;

  $\mathbb{P} \leftarrow \mathbb{P} \cup \{pivot^{(t)}\}$;

  foreach $MDO_i \in \{MDO_{i'}\}_{i'\in[1,n]}$ **do**

    MDO_i computes $E(-2x_0x_i, pk)$, $E(x_i^2 + y_i^2, pk)$, $E(-2y_0y_i, pk)$ and derives $E(\Delta_i, pk) = E((x_i - x_0)^2 + (y_i - y_0)^2, pk)$ just once; it also picks random $\gamma_{i1}^{(t)} \in (1, N^{(\gamma)}]$, $\gamma_{i2}^{(t)} \in [1, \gamma_{i1})$, computes $E(N - D^2[pivot^{(t)}], pk)$, derives $E(\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)}, pk)$ and sends it to the SSC ;

    The SSC derives $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)}$ by decryption, and updates $sign_i$ as 1 if it is positive or -1 if it is negative ;

  end

  if more than k $sign_i \in \overrightarrow{sign}$ is -1 **then** $\mathbb{I}_r \leftarrow \{i | sign_i = -1, i \in [1, n]\}$, $w_{right} \leftarrow pivot^{(t)}$ **else** $\mathbb{I}_l \leftarrow \{i | sign_i = -1, i \in [1, n]\}$, $w_{left} \leftarrow pivot^{(t)}$ **end**;

end

The SSC randomly picks $k - |\mathbb{I}_l|$ MDO_i where $i \in (\mathbb{I}_r - \mathbb{I}_l)$ and adds them to $\mathbb{I}_l$;

return $\mathbb{I}_l$;

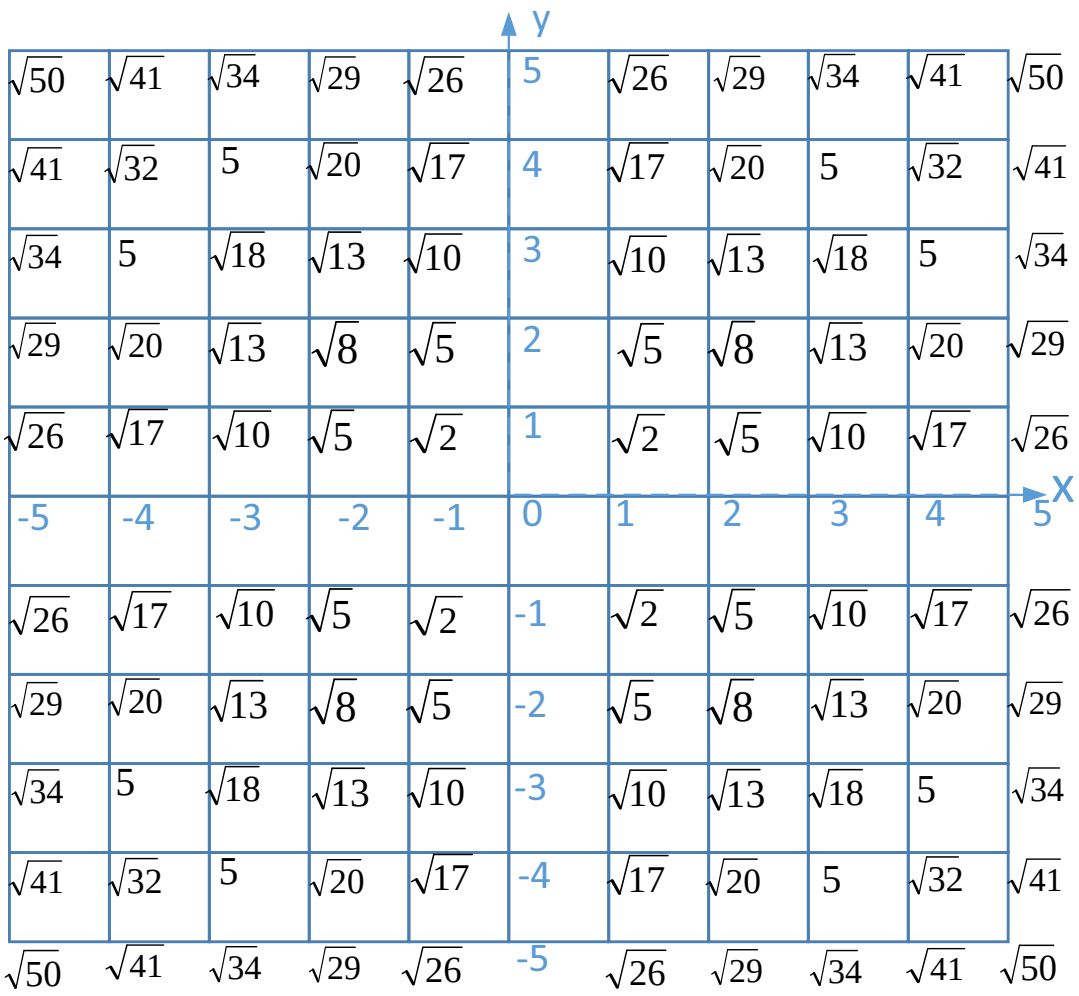

Figure 4.3: Illustration of All Possible Distances

$E(x_i^2+y_i^2+x_0^2+y_0^2-2x_ix_0-2y_iy_0, pk) = E(x_i^2+y_i^2, pk)\cdot E(x_0^2+y_0^2, pk)\cdot E(-2x_ix_0, pk)\cdot E(-2y_iy_0, pk)$.

In t-th iteration, MDO_i perturbs $\Delta_i - D^2[pivot^{(t)}]$ by $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$ where $\gamma_{i1}^{(t)} > \gamma_{i2}^{(t)}$. As a result, if $\Delta_i - D^2[pivot^{(t)}] > 0$, there exists $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)} > 0$; if $\Delta_i - D^2[pivot^{(t)}] \leq 0$, there exists $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)} < 0$.

Accordingly, the SSC can derive n linear inequalities in t-th iteration as follows:

$$r_{11}^{(t)}(\Delta_1 - D^2[pivot^{(t)}]) - r_{12}^{(t)} \lessgtr 0$$

$$r_{21}^{(t)}(\Delta_2 - D^2[pivot^{(t)}]) - r_{22}^{(t)} \lessgtr 0$$

...

$$r_{n1}^{(t)}(\Delta_n - D^2[pivot^{(t)}]) - r_{n2}^{(t)} \lesseqgtr 0$$

Given the array $D = \{D[k]\}_{k\in[1,K]}$ including all the possible distances, the goal of the SSC is to find the privacy window $[w_{left}, w_{right}]$ with the minimum size $|w_{right} - w_{left}| \geq w_{min}$ where $\mathbb{I}_l$ has the maximum number of MDOs when $|\mathbb{I}_l| \leq k$ and $\mathbb{I}_r$ has the minimum number of MDOs when $|\mathbb{I}_r| > k$. The first iteration starts with $[w_{left}, w_{right}] = [0, K]$ and $pivot^{(t)} = \lfloor \frac{w_{left} + w_{right}}{2} \rfloor$. In the t-th iteration, if there exists more than k linear inequality with '$<$' sign, w_{right} should be decreased by setting $w_{right} = pivot^{(t)}$ in the $(t+1)$-th iteration; otherwise, w^{left} should be increased by setting $w_{left} = pivot^{(t)}$ in the $(t+1)$-th iteration. The iterations stop when there exists k linear inequality with the '$<$' sign or the privacy window size has shrunk to w_{min} indicated by $Diff_{min}(pivot^{(t)}, \mathbb{P}) \leq w_{min}$, which denotes the minimum difference between $pivot^{(t)}$ and any element of $\mathbb{P}$. Aside from the $\mathbb{I}_l$ selected MDOs, the SSC randomly selects another $k - \mathbb{I}_l$ MDOs from the MDOs whose distances fall within $[D[w_{left}], D[w_{right}]$, because we assume selecting any $k-\mathbb{I}_l$ MDOs with distance within $[D[w_{left}], D[w_{right}]$ approximate the real $k-\mathbb{I}_l$ MDOs from the actual k-nearest MDOs.

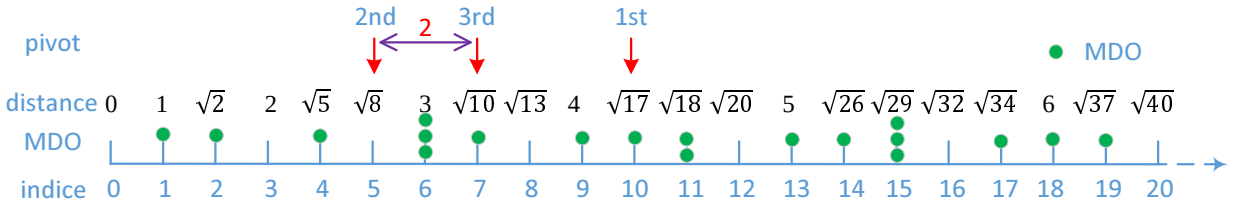

Figure 4.4: Search Process

Security of Algorithm 12: As the SSC keeps the private key sk, therefore it is impossible for any MDOs to derive (x_0, y_0) or $x_0^2 + y_0^2$ due to the semantic security of Paillier cryptosystem. Because the SSC keeps the signs of all the linear inequalities hidden from the MDOs, the MDOs cannot get the knowledge of the approximate

distances from (x_0, y_0), and thus this prevents them from narrowing down the possible range of (x_0, y_0) by collusion.

Meanwhile, $MDO_i (i \in [1, n])$ generates random $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$ in the t-th iteration to mask the value of $\Delta_i - D^2[pivot^{(t)}]$ where $\gamma_{i1}^{(t)} > \gamma_{i2}^{(t)}$. As a result, it is impossible for the SSC to derive Δ_i with the known $D^2[pivot^{(t)}]$ due to the lack of knowledge of $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$.

Meanwhile, the malicious SSC or the Cloud Mediator might manipulate $[w_{left}, w_{right}]$ in the t-th iteration to get the knowledge of the exact distance from (x_0, y_0) to each MDO. However, as $[w_{left}, w_{right}]$ and $pivot^{(t)}$ are made public in each iteration, any MDOs can easily check if the SSC or the Cloud Mediator honestly meet the following security requirements: i) $[w_{left}, w_{right}]$ and $pivot^{(t)}$ are updated in a divide-and-conquer manner correctly ii) the size of the privacy window $Diff_{min}(pivot^{(t)}, \mathbb{P})$ decreases by iterations iii) $Diff_{min}(pivot^{(t)}, \mathbb{P}) \geq w_{min}$. If not, this malicious behavior can be detected immediately and any MDOs can decline to continue the process.

4.4 Experiment

The proposed privacy-preserving schemes are emulated on the DEll OPTIPLEX 390 desktop with Intel(R) Core(TM) i3-2100 CPU at 3.10GHz and 4GB memory running 64-bit Windows 7 Enterprise. The whole processes are programmed on top of the java version of Paillier cryptosystem provided by Kun Liu in UMBC [53]. The emulations assess the computational cost of different scheme in terms of timing or the number of iterations without considering the communication cost.

Figure 4.5 illustrates the results of privacy-preserving sensing query in section 2.1. In the emulation, a random number $|\mathbb{I}^t|$ of the real target MDOs $\mathbb{I}^t$ are selected from the cloaking identification set $\mathbb{I}^*$ with the size $|\mathbb{I}^*|$ where $|\mathbb{I}^t| \in \{1, 2, 3, \cdots 18\}$ and $|\mathbb{I}^*| \in \{20, 30, 40\}$. The average computational cost of each customized selection is

derived by running random selections with the same $|\mathbb{I}^t|$ and $|\mathbb{I}^*|$ for 10 times. It can be learned that the average computation cost almost remains constant regardless of the number of the actually selected MDOs when the size of the cloaking identification set $\mathbb{I}^*$ is fixed. This also proves the privacy preservation as the SSC cannot estimate the number of the actually selected MDOs based on the computational cost. In addition, the average computation cost increases with the size of the cloaking identification set $\mathbb{I}^*$ as more MDOs get involved into the computation process.

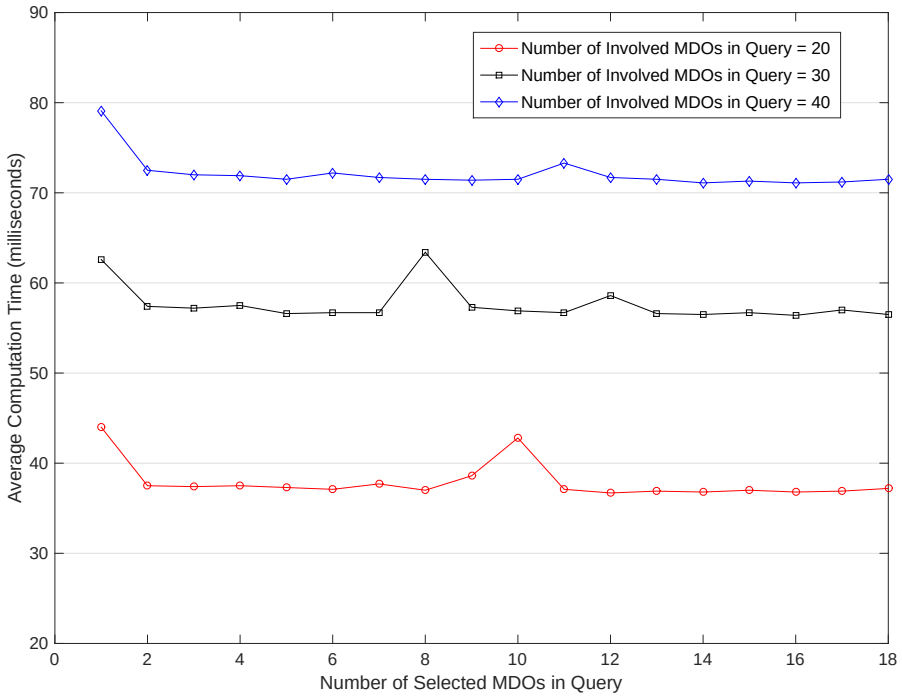

Figure 4.5: The Average Computational Cost Regarding Different Number of Target MDOs

Figure 4.6 illustrates the results of privacy-preserving multi-party sensing computation in section 2.2. The emulation runs 10 times for each designated number of involved MDOs. It can be learned that the average computation cost increases with

the number of the involved MDOs, as the addition and the comparison of the sensing reading of more MDOs would take more time.

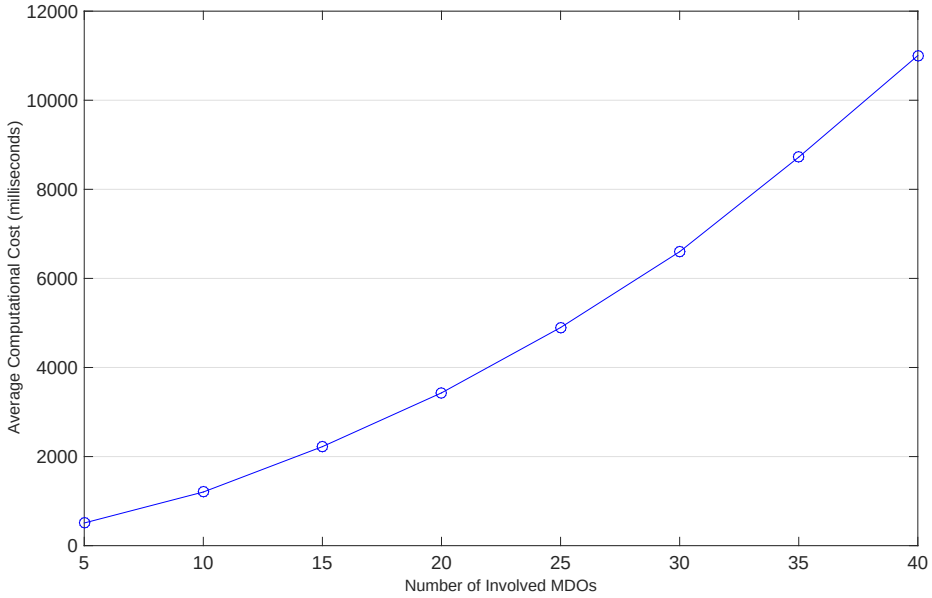

Figure 4.6: The Average Computational Cost Regarding Different Number of Involved MDOs

Figure 4.7 illustrates the results of approximate k-Nearest Neighbor with privacy-preservation in section 2.3. The emulation utilizes the distance array $D = \{0, 1, \sqrt{2}, 2, \sqrt{5}, \sqrt{8}, 3, \sqrt{10}, \sqrt{13}, \cdots, 60\sqrt{2}\}$ with 1446 elements under the assumption of 50 MDOs in total. The ranges for x and y coordinates grow from $[0, 10]$ to $[0, 60]$, and the number of the nearest neighbors grows from 1 to 45. The emulation derives the average number of iterations by running 10 times for each specified k and (x, y), which are randomly selected from each specified coordinate range. It can be learned that the average number of iterations needed gradually decreases as the coordinate range grows from $[0, 10]$ to $[0, 60]$, because the density of MDOs would be reduced with the fixed number of $MDOs$ and it is easy for the divide-and-conquer approach to identify

the k nearest neighbors. For the designated coordinate range and the fixed number of $MDOs$, the average number of iterations fluctuates around a certain value due to the $O(log(n))$ complexity of the divide-and-conquer approach.

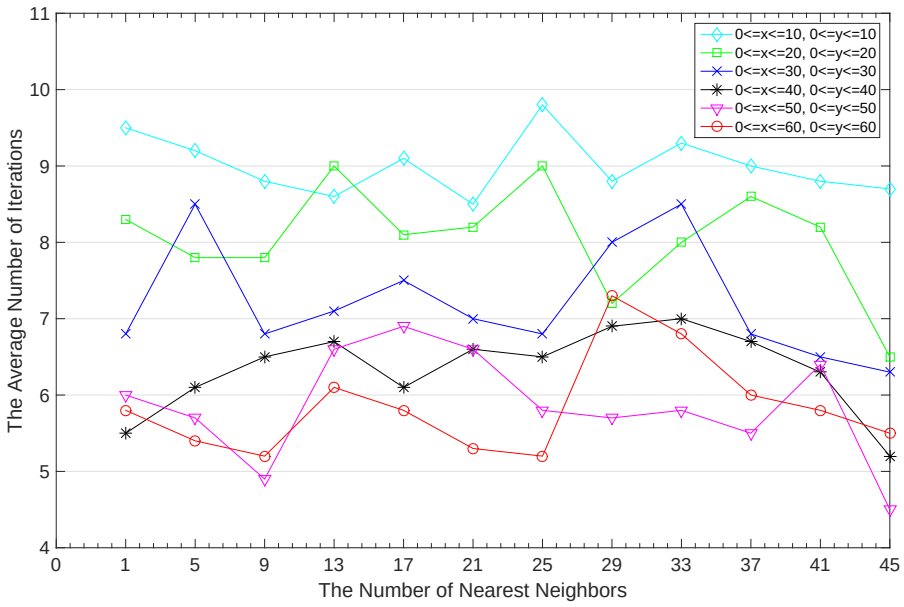

Figure 4.7: The Average Number of Iterations Regarding Different Number of Nearest Neighbors

4.5 Extensions

4.5.1 Private Participant Selection for Sensing Task Assignment in MCS

In MCS, appropriate MDOs with specific traits need to be identified for sensing task delegation to achieve desired quality of sensing data. More often than not, the SSCs are reluctant to disclose the specific traits of interest (e.g., the target region or time periods). At the same time, the MDOs refuse to disclose their traits due to privacy concerns, as those specific traits can be utilized to track and identify themselves. This conflict becomes a thorny problem and degrade the accuracy of

assigning sensing tasks to appropriate participants in MCS.

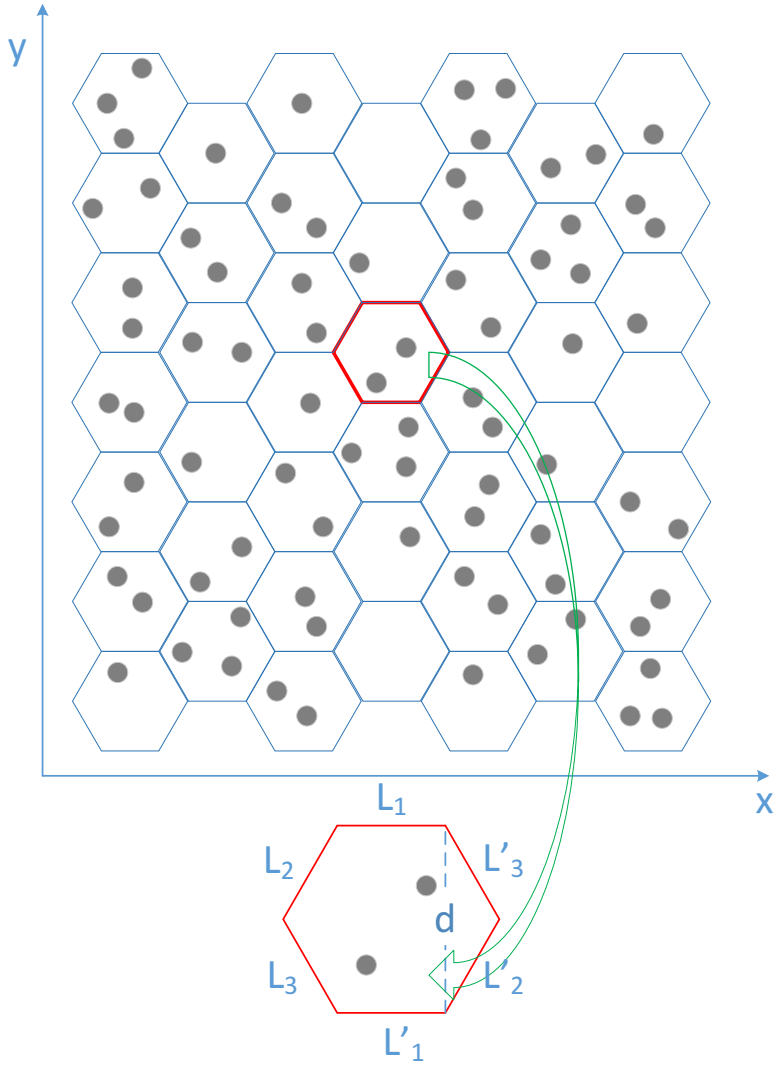

Figure 4.8: An Example of Privacy-Preserving Participant Selection in MCS

Homomorphic Encryptions can be utilized to address this thorny problem by secretly applying profile constraints, such that appropriate MDOs can be identified without disclosing the exact traits of MDOs or profile constraints specified by the SSCs. For the sake of simplicity, an example of location-related sensing task assignment is given in the following. Assume a SSC is interested in some sensing readings associate with a hexagonal cell with red edges as shown in Figure 4.8. The hexagonal

cell can be defined by the following six lines in the following mathematical forms:

$$L_1(x,y):\ k_1a_1x + a_1y + c_1$$

$$L_2(x,y):\ k_2a_2x + a_2y + c_2$$

$$L_3(x,y):\ k_3a_3x + a_3y + c_3$$

$$L_1^{'}(x,y):\ k_1a_1x + a_1y + c_1 - d$$

$$L_2^{'}(x,y):\ k_2a_2x + a_2y + c_2 - d$$

$$L_3^{'}(x,y):\ k_3a_3x + a_3y + c_3 + d$$

where d is the distance between any two parallel edges of the hexagonal cell, and k_1, k_2, k_3 are derived by rounding the slopes of L_1, L_2, L_3 to the nearest integers, respectively.

Accordingly, the SSC encrypts L_1, L_2, L_3 and sends $\{E(a_1, pk), E(c_1, pk), E(a_2, pk), E(c_2, pk), E(a_3, pk), E(c_3, pk), k_1, k_2, k_3, d\}$ to all MDOs. Subsequently, each $MDO_i (i \in [1, n])$ embeds its coordinates (x_0, y_0) into the line equation. Afterwards, each $MDO_i (i \in [1, n])$ selects random $r_{i1}, r_{i1}^{'}, r_{i2}, r_{i2}^{'}, r_{i3}, r_{i3}^{'}, r_{i4}, r_{i4}^{'}, r_{i5}, r_{i5}^{'}, r_{i6}, r_{i6}^{'}$ where $r_{i1} > r_{i1}^{'} > 0, r_{i2} > r_{i2}^{'} > 0, r_{i3} > r_{i3}^{'} > 0, r_{i4} > r_{i4}^{'} > 0, r_{i5} > r_{i5}^{'} > 0, r_{i6} > r_{i6}^{'} > 0$, computes $E(r_{i1}(k_1a_1x_i + a_1y_i + c_1) + r_{i1}^{'}, pk)$, $E(r_{i2}(k_2a_2x_i + a_2y_i + c_2) + r_{i2}^{'}, pk)$,$E(r_{i3}(k_3a_3x_i + a_3y_i + c_3) + r_{i3}^{'}, pk)$, $E(r_{i4}(k_1a_1x_i + a_1y_i + c_1 - d) + r_{i4}^{'}, pk)$, $E(r_{i5}(k_2a_2x_i + a_2y_i + c_2 - d) + r_{i5}^{'}, pk)$,$E(r_{i6}(k_3a_3x_i + a_3y_i + c_3 + d) + r_{i6}^{'}, pk)$ to SSC for further checking. This process is illustrated in **Algorithm 13** . It is obvious to see that as long as $a_1, a_2, a_3, c_1, c_2, c_3$ remain hidden, the MDOs cannot identify the hexagonal cell with the already known k_1, k_2, k_3, d.

4.5.2 Discussion on Parallel Computing in MCS

As MCS relies on potential millions of MDOs to feed the sensing reading into the system, the computational overhead of the cryptographic operations of sensing read-

Algorithm 13: Private Participant Selection

$\mathbb{I} \leftarrow \emptyset$;

The SSC generates $\{E(a_1, pk), E(c_1, pk), E(a_2, pk), E(c_2, pk), E(a_3, pk), E(c_3, pk)\}$, and sends them with $\{k_1, k_2, k_3, d\}$ to MDOs ;

foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ **do**

The $MDO_i (1 \leq i \leq n)$ constructs $\vec{d_i} = (\hat{d}_{i,0}, \hat{d}_{i,1}, \cdots, \hat{d}_{i,N'-1})$ where $\hat{d}_{i,k} = 1(k \in [0, d_i))$ and $\hat{d}_{i,k} = 0(k \in [d_i, N'-1])$, and computes $E(d_i, pk) = E(\sum_{k=0}^{N'-1} \hat{d}_{ik}^2, pk) = \prod_{k=0}^{N'-1} E(\hat{d}_{ik}^2, pk) = g^{d_i} r_k^N (mod\, N^2)$;

The $MDO_i (1 \leq i \leq n)$ selects random $r_{i1}, r_{i1}^{'}, r_{i2}, r_{i2}^{'}, r_{i3}, r_{i3}^{'}, r_{i4}, r_{i4}^{'}, r_{i5}, r_{i5}^{'}, r_{i6}, r_{i6}^{'}$ where $r_{i1} > r_{i1}^{'} > 0, r_{i2} > r_{i2}^{'} > 0, r_{i3} > r_{i3}^{'} > 0, r_{i4} > r_{i4}^{'} > 0, r_{i5} > r_{i5}^{'} > 0, r_{i6} > r_{i6}^{'} > 0$;

The $MDO_i (1 \leq i \leq n)$ further computes $E(r_{i1}(k_1 a_1 x_i + a_1 y_i + c_1) + r_{i1}^{'}, pk)$, $E(r_{i2}(k_2 a_2 x_i + a_2 y_i + c_2) + r_{i2}^{'}, pk)$,$E(r_{i3}(k_3 a_3 x_i + a_3 y_i + c_3) + r_{i3}^{'}, pk)$, $E(r_{i4}(k_1 a_1 x_i + a_1 y_i + c_1 - d) + r_{i4}^{'}, pk)$, $E(r_{i5}(k_2 a_2 x_i + a_2 y_i + c_2 - d) + r_{i5}^{'}, pk)$,$E(r_{i6}(k_3 a_3 x_i + a_3 y_i + c_3 + d) + r_{i6}^{'}, pk)$, and sends them to SSC ;

if $L_1(x_i, y_i) \leq 0$ and $L_1^{'}(x_i, y_i) \geq 0$ and $L_2(x_i, y_i) \geq 0$ and $L_2^{'}(x_i, y_i) \leq 0$ and $L_3(x_i, y_i) \geq 0$ and $L_3^{'}(x_i, y_i) \leq 0$ **then** $\mathbb{I} \leftarrow \mathbb{I} \cup \{i\}$;

end

return $\mathbb{I}$.

ing values in previous privacy-preserving sensing query scheme and privacy-preserving multi-party sensing computation scheme are intensive. To provide sensing service in a timely manner, it is critical to exploit existing parallel computing techniques to reduce the processing time. Hadoop is a widely-used implementation of the parallel computing model MapReduce, and it can be applied int both the two privacy-preserving schemes to speed up the process without breaking the privacy.

In the privacy-preserving sensing query scheme, each MDO responds key-value pairs on receiving the sensing requests. Assume there exist three MDOs MDO_1, MDO_2, MDO_3 and two sensing queries from two SSCs u_1, u_2 respectively. MDO_i sends a key-value pair $< MDO_{i-j}, C_i^{u_j} >$ in response to the query from u_j where $C_i^{u_j} = c_j^{d_i}$ and $c_j = g^0 r_j^N$ or $g^1 r_j^N$. The mappers hosted by the CP take $< MDO_{i-j}, C_i^{u_j} >$ and output new key-value pair $< u_j, C_i^{u_j} >$. Each reducer takes the list of new key-value pairs with the same key u_j and produces the aggregation result $< u_j, C^{u_j} >$ where $C^{u_j} = \prod_{i \in [1,3]} c_j^{d_i}$. This process is illustrated in Figure 4.9

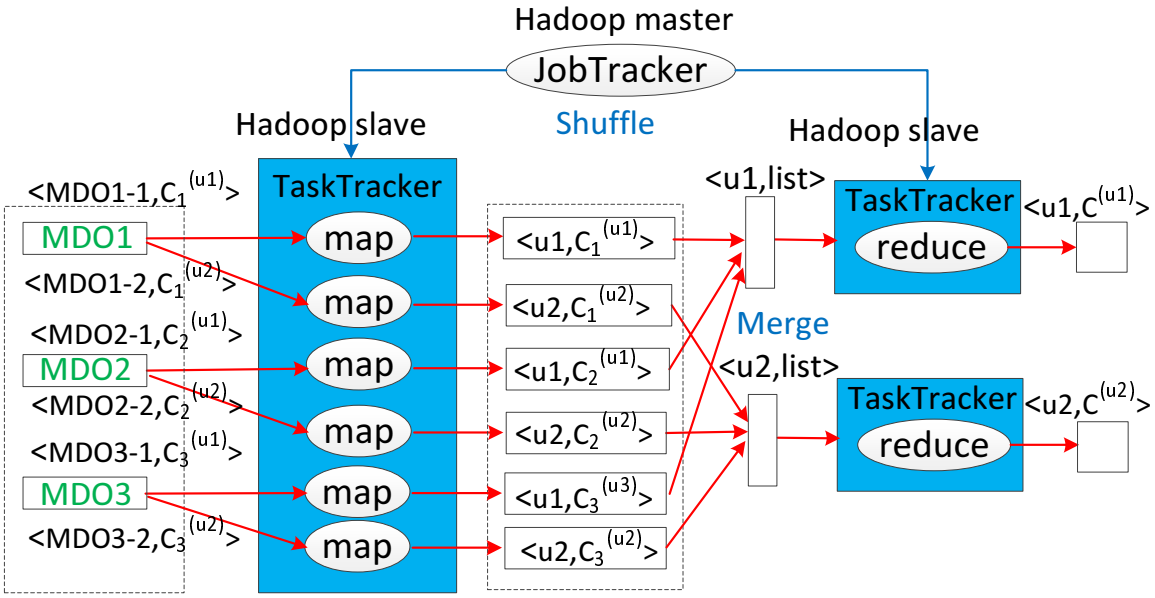

Figure 4.9: An Example of Privacy-Preserving Sensing Query Using Hadoop

By the same token, in the privacy-preserving multi-party sensing computation scheme, each MDO responds key-value pairs on receiving the sensing requests. Assume there exist three MDOs MDO_1, MDO_2, MDO_3 and two sensing queries from two SSCs u_1, u_2 respectively. MDO_i sends a key-value pair $< MDO_{i-j}, E_i^{u_j} >$ in response to the query from u_j where $E_i^{u_j} = E(\Delta_i^{(u_j)} + P_i, pk)$. The mappers hosted by the CP take $< MDO_{i-j}, E_i^{u_j} >$ and output new key-value pair $< u_j, E_i^{u_j} >$. Each reducer takes the list of new key-value pairs with the same key u_j and produces the aggregation result $< u_j, E^{u_j} >$ where $E^{u_j} = \prod_{i\in[1,3]} E_j^{d_i}$. This process is illustrated in Figure 4.10.

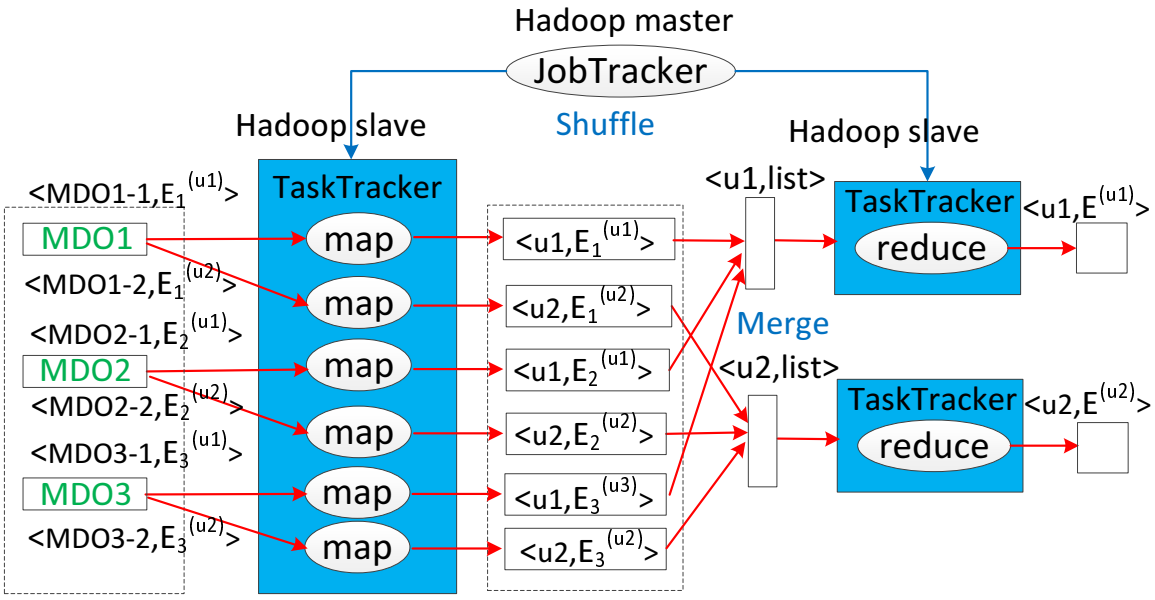

Figure 4.10: An Example of Privacy-Preserving Multi-Party Sensing Computation Using Hadoop

4.6 Related Work

Numerous techniques have been studied to secure the privacy of multiple-party data computation and data sharing in sensing scenarios. Substantial research work

[19,20,54] have been done for privacy protection by data perturbation or dummy data generation and these incur significant overhead in mobile devices and decrease data utility. Some other approaches adopt k-anonymity [16–18], which heavily depends on the distribution and density of the mobile users, thereby rendering it impossible in many real settings. Many other approaches [11–13,30] rely on a trusted third party to host the individual data of mobile users for sensing query requests, and compromising the third party can result in the breach of the private data.

Homomorphic Encryptions provide an important solution to privacy preservation for multiple-party data computation and data sharing in sensing scenarios. Lin *et al.* [55] addressed the issue of securing two-parties' data comparison in a privacy preserving manner by exploiting ElGamal encryption, Paillier encryption, 0-Encoding and 1-Encoding. Erkin *et al.* [55] explored relatively efficient cryptographic privacy techniques based on Paillier cryptosystem to allow spatial and temporal aggregation of smart meter measurements. Privacy-preserving face recognition is also studied in [56] and extensive experiments are done by running the standard Eigenfaces recognition algorithm. Bilogrevic *et al.* [57] proposed two privacy-preserving algorithms for the fair render-vous point problem with transformation functions based on homomorphic encryption for location-based services. However, the distances to certain users are exposed to multiple mobile users, and these mobile users can collude together to pinpoint the exact coordinates of the target victim. Ghinita *et al.* [39] first addressed the issue of private comparison of two integers, and accordingly proposed approximate and exact hybrid algorithms for private nearest-neighbor queries based on multi-level index infrastructure and Voronoi cells. Nonetheless, the two algorithms rely on the trusted location server to partition the two-dimensional plane into small convex polygon areas first. Yi *et al.* [58] studied how to preserve the privacy of k-nearest neighbor queries from the semi-honest location-based service provider based on Paillier cryp-

tosystem. All the location data info have to be revealed to the location-based service provider. The whole area is divided into small regions, and the location-based service provider returns a cluster of cells with points of interest that could be more than k. Choi *et al.* [9] addressed the issue of secure mutual proximity zone enclosure evaluation with homomorphic encryption and order-preserving encryption. Specifically, it proposed two protocols, such that a client can securely determine her location is enclosed in the proximity zone of a target, and the client learns if the target's location is within the client's proximity zone, respectively.

4.7 Conclusion

This chapter investigates how to preserve the privacy of the sensing query of the SSC and the sensing readings of MDOs in MCS based on the homomorphic properties of Paillier cryptosystem. It first present a privacy-preserving scheme such that the real target MDOs in the sensing query are kept secret. Then it describes how to privately compare the absolute difference between each MDO's sensing reading and a baseline value with a specified proportion of aggregated absolute difference of all MDOs, followed by private identification of approximate k-nearest neighbors. Private participant selection and the utilization of parallel computing are also discussed int the extension.

Chapter 5

PRIVACY-PRESERVING SENSING DATA ACCESS CONTROL IN MCS

In MCS scenarios, the MDOs yield data from ambient environment and store them on the SSPs for the SSCs across different security domains to use. In some cases, the SSPs are lack of means to validate the authenticity of the SSCs' identity information, and they may not have enough computation and storage resources to manage the online setup of large volume of digital identities. Furthermore, the storage and management of sensitive personal identity information incur overwhelming finance burden over the startup SSPs with limited investment, as these SSPs have to deal with the personal information with extreme attention. Otherwise, the leakage of personal identities and associated personal proprietary resources could destroy the effort of the startup SSPs to establish better brand recognition and incur unaffordable cost. Accordingly, the startup SSPs are motivated to cut through the complexity and potential risk of maintaining a large pool of identities by shifting the tedious task of identity management to a few well-known third-party identity providers at the beginning.

In this chapter, we present our previous work, which could be applied to the access control over SSCs to handle the issues mentioned above in distributed MCS environment. Access to sensing services for SSCs should be allowed only based on the pre-established trust between the SSCs and the SSPs, and the SSPs can only place trust according to the identities and attributes of the SSCs. In fact, the fast-growing online applications and services for smartphones boost the online setup of mobile identities, and large quantities of identity setups have proven to be a financial and management burden for many start-up entrepreneurs. On one hand, since smartphones store tremendous personal information about the individual users, the

management of mobile identities must be handled with extreme caution as it is often related to personal proprietary resources. On the other hand, there is a rising demand on an open and federated mobile identity management system to facilitate the access control of resource sharing between collaborating parties across heterogeneous security domains. Therefore, these factors necessitate an open, comprehensive and secure mobile access control system whereby different mobile service providers can outsource the access control tasks and reduce the management cost.

A natural approach to address this problem is to leverage the existing federated identity management framework based on the concept of Identity-as-a-Service (IDaaS). The federated identity management frameworks built on IDaaS, such as OpenID [59], Centralized Authentication Service (CAS) [60] and Shibboleth [61], emerges as widely-accepted solutions to integrate multiple identity service providers and consumers together. The OpenID is an open standard for authentication delegation as illustrated in Figure 5.1. It architects a framework in which the Identity Provider (IdP) acts as a centralized authority by holding the identities and credentials of the End Users (EU). When the EU interacts with the Relying Partiy (RP)(e.g., a website or application), the RP delegates the authentication process to the IdP. Specifically, the EU provides the RP with his/her public OpenID identifier in the form of URL or XRI (i.e., eXtensible Resource Identifier), and accordingly the RP redirects the EU's user agent to the IdP for authentication, and then the Identity Provider sends back an assertion notifying the authentication result. In this manner, the RPs can reduce the administrative burden of maintaining the vast amount of digital identities and credentials, while the EUs can eliminate the frustration of memorizing multiple pairs of username and password, thereby gaining more control of their online identities. Please note that the concept of SSP is equivalent to RP, and the concepts of SSC, EU and MU are also equivalent.

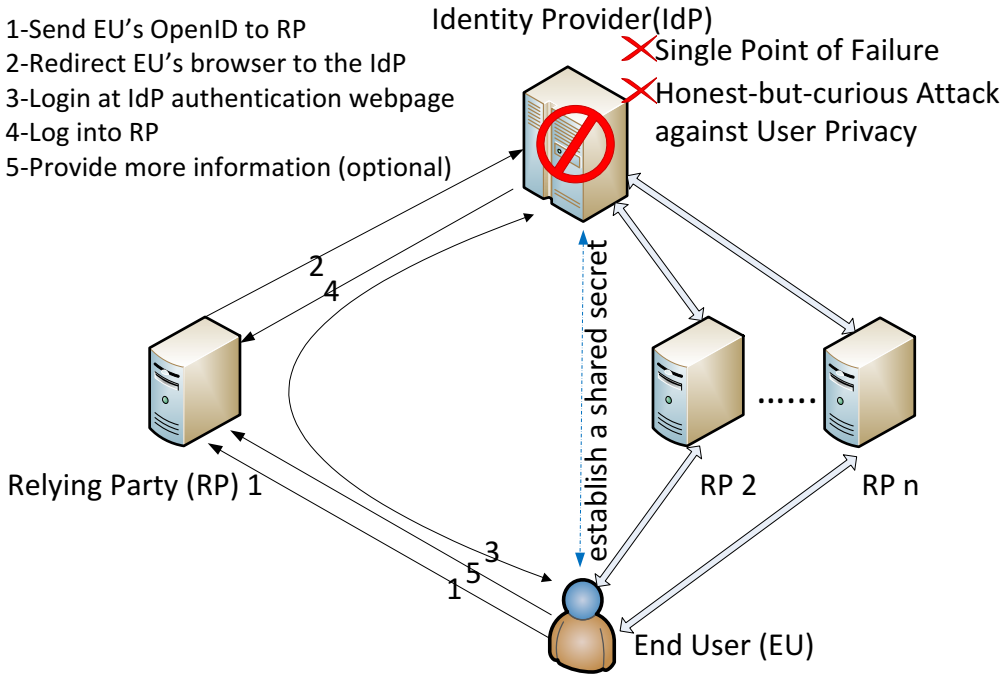

Figure 5.1: The Security Issues in OpenID Framework

At with OpenID, CAS adopts a similar centralized approach as illustrated in 5.2. When the EU visits an application service requiring authentication, the service redirects the EU to the CAS server with its service ID. Subsequently, the CAS server validates the EU's authenticity by checking his/her username and password against a database, such as Active Directory, Kerberos or LDAP (i.e., Lightweight Directory Access Control). If the authentication succeeds, the CAS server redirects the client to the application service along with a security ticket while the EU can keep the cookie. Afterwards, the application validates the ticket with the CAS server by providing the ticket and its service ID. After the ticket is verified, the authentication is considered successful and the EU is allowed to access this application service.

Shibboleth is another logging-in system for federated identity-based authentication and access control, and it adopts similar centralized approach as illustrated in Figure 5.3. When an EU tries to access the resource hosted by a Sensing Service

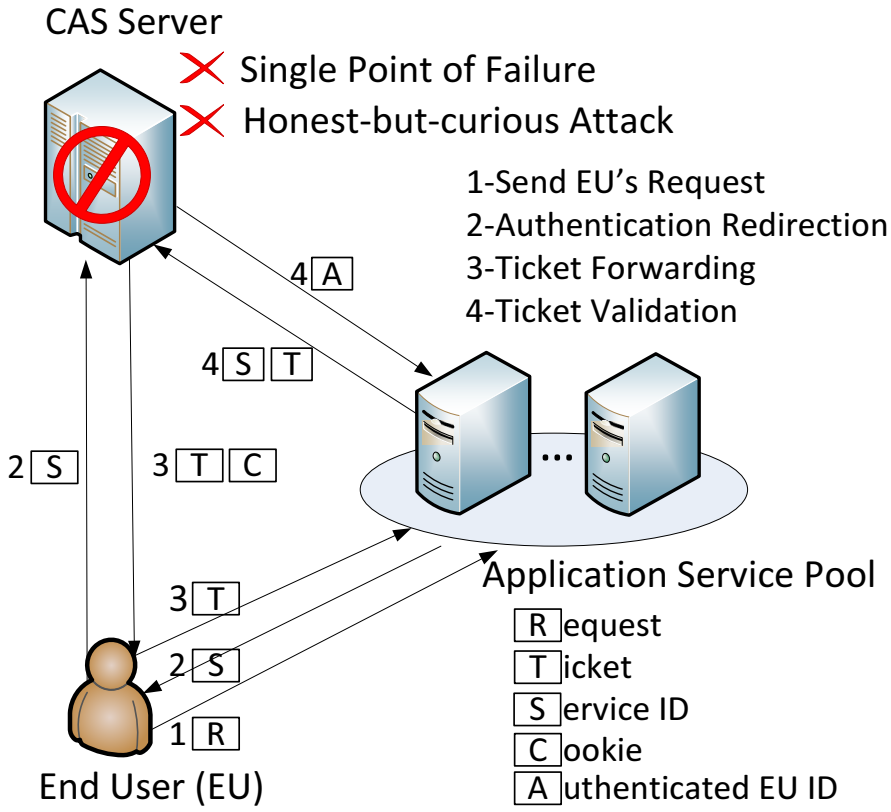

Figure 5.2: The Security Issues in CAS Framework

Provider (SSP), she would be redirected to Where-Are-You-From (WAYF) service to select her own home organization. Afterwards, she is redirected to the home organization's IdP and enter her credentials. After authentication, the IdP redirects the EU back to the SSP along with assertion messages. The SSP validates the assertion messages and it may also require additional attribute information from the IdP. On receiving these attributes, the SSP makes the decision if the EU is allowed to access the resources. Note that the authentication mechanism used by the IdP can be OpenID, Kerberos, etc.

However, the federated identity management frameworks mentioned above are prone to several security attacks. First of all, the IdP could be Honest-But-Curious (HBC) and it can easily invade EU's privacy, because the IdP will know all the RP-

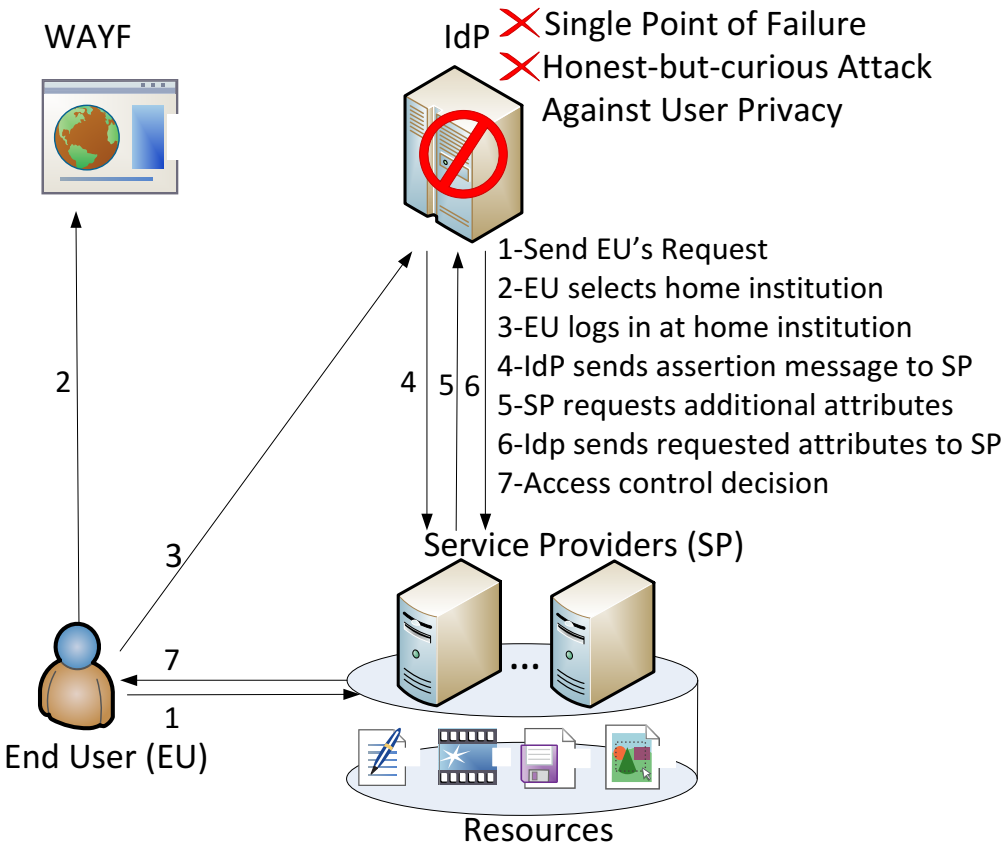

Figure 5.3: The Security Issues in Shibboleth Framework

s/SPs that the EU has been trying to log into, since the RPs/SPs have to delegate the authentication requests to the IdP, and the RPs/SPs need to directly communicate and verify with the IdP. Second, OpenID is vulnerable to SPOF. A single IdP (e.g., Google) can become the central axis by undertaking the authentication tasks for numerous RPs and EUs, and the crash of this IdP will imperil all the related authentication and access control processes. For example, Google crashed for five minutes with all its services on Aug. 16th, 2013, and this led to a 40% drop in the global Internet traffic [62]. Even worse, once the IdP is compromised, all the EUs' credentials stored on this IdP are exposed to the attacker, which implies the attacker can access the personal resources of each EU on all RPs/SPs. Last but not

the least, the malicious RP/SSP can launch a phishing attack by directing the EU to a bogus authentication webpage of a fake IdP. If the EU gives out her credentials, then the malicious RP/SSP can access the personal information of the EU on all the RPs/SPs. Likewise, the CAS framework suffers from similar drawbacks due to its similar architecture and authentication procedures.

Moreover, those federated identity management frameworks do not fit well with the collaborative environment wherein multiple authorities may dynamically cooperate with each other to undertake the authentication and access control tasks. For example, a joint lab co-founded by the computer science department and the biology department only permits the access of the faculty and staff who are affiliated with both of the two departments. A parking building allows the employees from any of its sponsored corporations nearby to park. A local veteran association only allows the veterans from several organizations to enter. In addition, the authorities might join or leave at any time, and the RPs/SPs should be able to update the access policies easily and minimize the impact of any changes.

To address the issues stated above, we propose a novel Distributed Privacy-preserving Mobile Access Control (DP-MAC) framework based on the integration of decentralized Attribute-Based Encryption (DABE) [63] and Identity-Based Encryption (IBE) [64]. In DP-MAC, the Sensing Service Provider (SSP) can create different access policies and dynamically select multiple IdPs to undertake the tasks for mobile access control whereby the SPOF issue can be solved. In addition, the IdPs are prevented from tracking the mobile users' historic access to SPs, because the IdPs are not allowed to directly communicate with the SPs and they cannot infer which SSP the user is trying to log in. A side benefit is that the phishing attack is eliminated as there is no webpage redirection. Moreover, the system utilizes the Dual-Root Trust (DRT) model wherein the Mobile User (MU) splits her secret credential into two parts, and

stores them on the mobile device and the user-centric Mobile Cloud (MC) respectively. **Note that different users can freely select different cloud providers (e.g., Amazon EC2, Microsoft Azure) to be the MC and store their credential part, such that compromising any cloud platform has no impact on the users on other cloud platforms and SPOF is mitigated.** Hence, the loss of mobile device does not cause identity theft, since both parts are indispensable to validate the user in the process of mobile access control. Additionally, the major computation cost is shifted from the resource-constrained mobile device to MC as a result. To sum up, our contributions are four-fold as follows: 1) a new dependable distributed framework with dual-root trust for mobile access control whereby the SPOF problem is avoided and the threat of mobile identity theft is alleviated; 2) a new mobile access control model to counter against HBC attack from IdP, thereby protecting the user's privacy with respect to the historic access information; 3) we implement the mobile authentication system in mobile cloud platform and android smartphones to prove its applicability in real-world settings.

- We architect a new distributed framework with dual-root trust for mobile access control whereby the SPOF problem is mitigated due to multiple IdPs and distributed MCs, and the threat of mobile identity theft is alleviated.

- We devise a new access control model to counter against HBC attack from IdP, thereby protecting mobile users' privacy with respect to the historic access information.

- In DP-MAC, the SPs do not need to establish shared secrets with any IdPs, and they do not maintain the credentials of any mobile users, thereby achieving the maximum scalability in the distributed environment where the number of IdPs and mobile users can grow exponentially.

- We implement DP-MAC in mobile cloud platform and android smartphones with jPBC [65] to prove its applicability in real-world settings.

On the other hand, in some MCS scenarios, a centralized sensing data access control mechanism is more favorable as it is easy for management. Ciphertext-Policy Attribute-Based Encryption (CP-ABE) [66] is a milestone in cryptographic research field in the past ten years. In CP-ABE, the centralized system can group the users on the coarse-grained attribute level rather than the fine-grained identity level, thereby enabling attribute associated group communications without ID involved and significantly reducing the overhead. It provides an effective approach in sensing data outsourcing and access control by encrypting data with attribute-based policies and offering efficient access to multiple eligible SSCs with designated attribute constraints. Without loss of generality, an attribute is a descriptive term and each data user owns one or multiple attributes. The MDO creates access policies in a flexible way by designating attribute constraints and embedding the data access policies into the ciphertext. Any SSC has to prove the validity of his attributes to access the data, and thus SSCs data access privileges are determined by both their own attributes and MDOs access policies. It is self-evident to see that CP-ABE also offers privacy as the SSCs do not need to reveal their identities to access the encrypted sensing data.

However, the benefits also come with cost. As CP-ABE systems do not differentiate individual SSCs on the ID level, it is very difficult to revoke any single SSC based on his/her ID, and this has a negative impact on the deployment of CP-ABE system in real-world settings. For example, if a SSC lost his/her electronic device with private keys, then his/her identity must be revoked from the CP-ABE system. In previous research work, all the SSC with the same attributes have to update their private keys as a result, thereby causing tremendous system overhead. Therefore, in this paper, a new ID-revocable technique in ABE systems is presented to revoke SSCs

such that the existing keys do not have to be updated and the users with associate attributes are not impacted. It is inspired by the revocation technology Lewko *et.al* introduced in the broadcast encryption [67]. The structure style of the scheme still follows Water's CP-ABE scheme [68]. Basically, ID can be taken as a special attribute owned by only one user in our scheme. In other words, each ID is a special attribute that is mutually exclusive to any other IDs. The structure style of the scheme still follows Water's CP-ABE scheme.

5.1 Distributed Sensing Data Access Control

5.1.1 The DP-MAC Framework Design and Preliminaries

In this section, we presents the system model of DP-MAC and adversary model as well as cryptographic preliminaries.

System Model

In DP-MAC, the mobile users can use credentials from multiple identity providers without disclosing their historical access information of application services to the identity providers. It utilizes a decentralized architecture as illustrated in Figure 5.4, and consists of five components as follows:

- *Distributed Public Key Generators* (*D-PKGs*): The D-PKGs are trusted parties to generate private keys for the Identity Providers based on their identity strings in the system setup. After that, the D-PKGs go offline until a new Identity Provider joins or leaves the system.

- *Identity Provider* (*IdP*): An IdP is an entity that issues unique identifiers to the registered mobile device users. It is responsible for authenticating the mobile users for the Sensing Service Providers.

- *Sensing Service Provider* (SSP): The SSP is usually a website or an application that provides services and resources for the mobile users. It delegates the authentication tasks to the trusted IdPs to reduce its own overhead. Note that the concept of SSP is equivalent to RP that appeared in the front part of this chapter.

- *Mobile Cloud* (MC): The MC provide a dedicated virtual machine to the mobile user for computation services and credential storage. It functions as a middle-man between multiple IdPs and the mobile user to help reduce the communication and computation cost of the mobile device.

- *Mobile User* (MU): The MU is the owner of a mobile device and she needs to log into the SSP for sensing services and resources. The term *Mobile User* (MU) is equivalent to *Sensing Service Consumer* (SSC) int this paper.

Adversary Model

We have the following assumptions: 1) Both the IdPs and MCs comply with the protocol and output the correct results; 2) The mobile devices perform secure execution of trusted codes in trusted execution environment 3)Different classes of participating parties do not collude together; 4) the communication between different classes of participating parties are protected by secure channels. Accordingly, we consider the following attacks in DP-MAC:

- *Single Point of Failure* ($SPOF$): The DP-MAC system cannot proceed due to the crash of any single party, which could be an IdP, a MC or a PKG.

- *Honest But Curious* (HBC): Both the IdPs and MCs honestly follow the

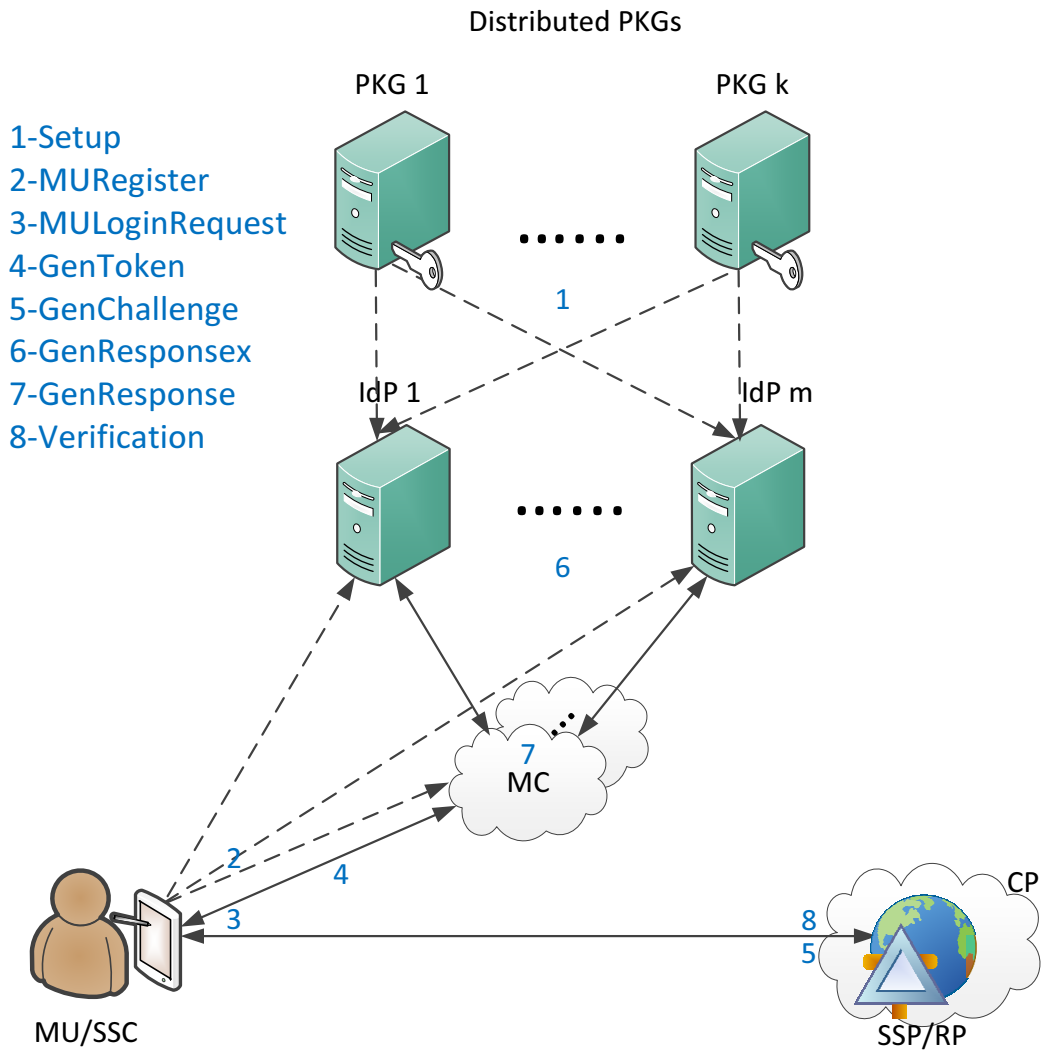

Figure 5.4: The Architecture of DP-MAC

procedures, but they are interested in tracking the MU's historical access information to different SPs.

- *Impersonation*: The malicious SSP might leverage the received token to impersonate the MU in the future mobile authentication and access control procedure. Meanwhile, the MU could be subject to identity theft due to the loss of her mobile device.

In section V, we will prove that our system is resilient against the attacks mentioned above.

Cryptographic Preliminaries

Bilinear pairings are the basic operations in our framework. Assume $\mathbb{G}$ and $\mathbb{G}_T$ are two cyclic groups with order q generated by a Bilinear Diffie-Hellman (BDH) parameter generator $\mathcal{G}$. Correspondingly we set up a bilinear map system $\mathbb{S} = (q, \mathbb{G}, \mathbb{G}_T, e)$ where e denotes a computable bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ with the following properties:

- Bilinearity: $\forall G, W \in \mathbb{G}, \forall a, b \in \mathbb{Z}, e(G^a, W^b) = e(G, W)^{ab}$;
- Non-degeneracy: G and W are the generators of $\mathbb{G}$, $e(G^a, W^b) \neq 1$;
- Computability: $e(G, W)$ is efficiently computable.

In our system,we set $G = W$ and make $G, \mathbb{G}, \mathbb{G}_T$ public.

The cryptographic techniques of our system are constructed on top of IBE with D-PKGs and DABE. IBE enables the encryptor to use the receipient's identity string (e.g., "bob@gmail.com||current-year") as the public key to encrypt messages, thus it eliminates the need for a public key distribution infrastructure. DABE enables the encryptor to encrypt data using a policy written over credentials across different authorities between which there is no coordination. Therefore, the SSP can use the IdP's identity as the public key to encrypt some secrets, and send them to the IdP through the mobile device and MU for authentication without worrying about being tampered. Also, the SSP can dynamically select multiple trusted IdPs for mobile authentication and access control by creating an access policy with LSSS access structure [69, 70]. Consequently, even if some IdPs are compromised, the SSP can easily adjust the authentication criterion by removing the impacted IdPs and adding other IdPs.

5.1.2 System Construction and Workflow

In this section, we describe how to construct our system and exhibit the workflow in Figure 5.5. The most commonly notations are listed in Table 5.1.

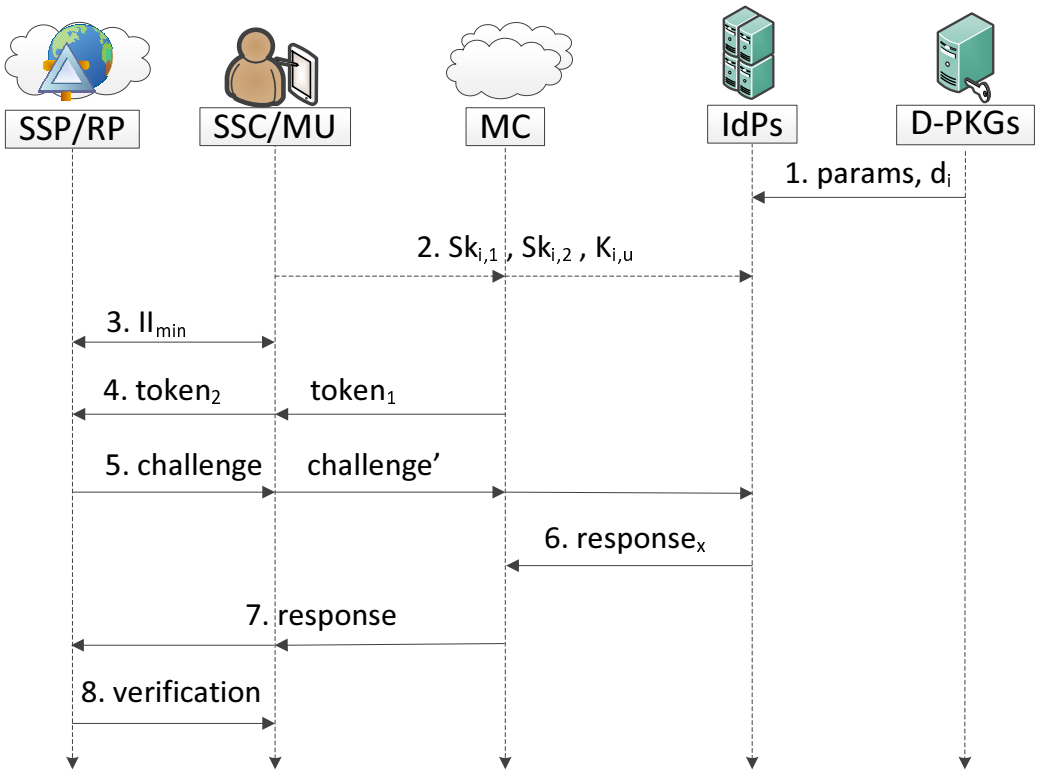

Figure 5.5: The DP-MAC Workflow

System Setup

In the system setup, the D-PKGs generates public parameters and computes private keys for IdPs. The MU also registers at selected IdPs and MC with secret keys and credentials. **1. Setup**$(\kappa, \mathcal{G}, \mathbb{I}) \rightarrow (params, \{d_i\}_{\mathtt{I}_i \in \mathbb{I}})$: Given a security parameter κ, a BDH parameter generator $\mathcal{G}$ and a collection $\mathbb{I}$ of IdPs, the k distributed PKGs do the following steps:

1. Given a security parameter κ, it runs $\mathcal{G}$ to generate a prime q, $\mathbb{G}, \mathbb{G}_T$ of order

Table 5.1: Notations

Notation	Description
GID_u	The unique ID of the mobile device (e.g., IMEI).
(k', k)	the threshold of D-PKGs; k is the total number of PKGs and at least k' of these PKGs can derive the secret key.
$\mathbb{I}, \mathtt{I}_i$	the IdP set and the i-th IdP in $\mathbb{I}$.
$\mathbb{I}_{MU}$	the set of the IdPs registered by the MU.
$\mathbb{I}_{SSP}$	the set of the IdPs trusted by the RP in a mobile authentication and access control process.
$\mathbb{I}_{min}$	the minimum subset contained in both $\mathbb{I}_{EU}$ and $\mathbb{I}_{SSP}$ that can satisfy the access policy.
ε_j	the master key share generated by the j-th PKG from the k' PKGs selected by the IdP.
$d_{i,j}$	the private key share generated by the j-th PKG from the k' PKGs selected by the IdP $\mathtt{I}_i$.
d_i	the private key of the IdP $\mathtt{I}_i$.
A	a LSSS-based access matrix used by the SSP to select the trusted IdPs.
A_x	the x-th row of A.
$\rho(x)$	the mapping from A_x to IdP x.
s	a random secret generated by the SSP in a mobile authentication and access control process; $s \in \mathbb{Z}_q$.
$SK_i = \{\alpha_i, y_i\}$	the secret of the MU generated in the registration process on IdP i where $\alpha_i, y_i \in \mathbb{Z}_q$.
$SK_{i,1} = g^{y_i}$	the secret key share stored on the MC with respect to IdP $\mathtt{I}_i$ where $g \in \mathbb{G}$.
$SK_{i,2} = g^{\alpha_i}$	the secret key share stored on the MU's mobile device with respect to IdP $\mathtt{I}_i$.
$K_{i,u}$	the credential of MU u stored on IdP $\mathtt{I}_i$.

q so that there exists a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$.

2. Pick random generators $P, g \in \mathbb{G}$ and $\varepsilon \in \mathbb{Z}_q^*$. As with the distributed PKGs

in IBE, each of the k PKGs keeps one share ε_j of a Shamir secret sharing of ε mod q in a k'-out-of-k fashion.

3. Choose three cryptographic hash functions H_1, H_2, H_3 such that there exist $H_1 : \{0,1\}^* \to \mathbb{G}^*$, $H_2 : \mathbb{G}_T \to \{0,1\}^n$ for some n, and $H_3 : \{0,1\}^* \to \mathbb{Z}_q^*$. The public system parameters are $params = \{q, \mathbb{G}, \mathbb{G}_T, e, n, P, g, P_{pub} = P^{\varepsilon}, H_1, H_2, H_3\}$

4. Given the i-th IdP's identity string $\mathtt{I}_i \in \{0,1\}^n$, each IdP $\mathtt{I}_i$ selects k' out of the k PKGs to compute $Q_i = H_1(\mathtt{I}_i) \in \mathbb{G}^*$ and $\{d_{i,j} \mid d_{i,j} = Q_i^{\varepsilon_j}\}_{1 \le j \le k'}$. Subsequently, the IdP $\mathtt{I}_i$ derives its private key by computing $d_i = \prod_j d_{i,j}^{a_j} = Q_i^{\sum_j a_j \varepsilon_j}$ where the a_j's are the appropriate Lagrange coefficients.

5. The D-PKGs goes offline until any of the IdPs need to be revoked or a new IdP joins the system.

2. MURegister$(params, \mathbb{I}) \to \{SK_{i,1}, SK_{i,2}, K_{i,u}\}_{\mathtt{I}_i \in \mathbb{I}_{MU}}$: In this process, the MU registers at a subset of IdPs $\mathbb{I}_{MU} \subseteq \mathbb{I}$. She also selects a MC at will and registers a dedicated virtual machine. All the keys and credentials are computed in an oblivious approach. The steps proceed as follows:

1. The MU selects multiple IdPs $\mathbb{I}_{MU} \subseteq \mathbb{I}$, and registers a unique account on each IdP with GID_u. For IdP $\mathtt{I}_i$, the MU randomly selects $\alpha_i, y_i \in \mathbb{Z}_q$ as the secrete key $SK_i = \{\alpha_i, y_i\}$, and stores the credential $K_{i,u} = g^{\alpha_i} H_1^{y_i}(\mathrm{GID}_u)$ on IdP $\mathtt{I}_i$.

2. The MU computes $SK_{i,1} = g^{y_i}$ for every selected IdP $\mathtt{I}_i$ and stores $\{SK_{i,1}\}_{\mathtt{I}_i \in \mathbb{I}_{MU}}$ on a dedicated virtual machine of the selected MC.

3. The MU computes $SK_{i,2} = g^{\alpha_i}$ for every selected IdP $\mathtt{I}_i$, stores $\{SK_{i,2}\}_{\mathtt{I}_i \in \mathbb{I}_{MU}}$ on the mobile device and discards the secret $\{SK_i\}_{\mathtt{I}_i \in \mathbb{I}_{MU}}$.

Mobile Authentication and Access Control

In this procedure, the MU and the SSP interactively negotiate to derive a minimum subset of IdPs $\mathbb{I}_{min}$ to satisfy the SSP's access policy. Afterwards, the MU provides proofs to the SSP with the help of the MC, and the SSP sends the encrypted proofs to the IdPs through the mobile device and the MC for validation. This process proceeds as follows:

3. MULoginRequest$(\mathbb{I}_{MU}, \mathbb{I}_{RP}) \rightarrow \mathbb{I}_{min}$: In this phase, the MU and the SSP cooperates to derive a minimum subset of IdPs from the intersection of the SSP-trusted IdPs and the MU-registered IdPs to satisfy the access policy. The steps go as follows:

1. The MU sends a login request to the RP, and the SSP shows the EU its access policy in the form of a boolean formula in terms of the trusted IdPs $\mathbb{I}_{SSP}$. For example, the policy $IdP1 \wedge IdP2 \wedge (IdP3 \vee IdP4 \vee IdP5))$ implies the MU must have registered at IdP1, IdP2, and one out of IdP3, IdP4 and IdP5.

2. The MU selects the IdPs she has registered, and they can derive the minimum subset $\mathbb{I}_{min}$ such that $\mathbb{I}_{min} \subseteq (\mathbb{I}_{MU} \bigcap \mathbb{I}_{SSP})$ to satisfy the authentication criterion. We do not discuss the details about how to derive $\mathbb{I}_{min}$ in this paper due to page limits. If $|\mathbb{I}_{min}| = 0$, then the login request is declined. Otherwise, the authentication and access control process proceeds.

4. GenToken$(params, \{SK_{x,1}, SK_{x,2}\}_{\mathtt{I}_x \in \mathbb{I}_{min}}) \rightarrow token_2$: In this phase, the SSP requests the validation that the MU has registered at the IdPs in $\mathbb{I}_{min}$, and the MU produces the proof with the help of the MC as follows:

1. The MU invokes the RESTful service running on the MC through SSL/TLS. The MC computes and transfers $token_1 = \{g^{r_{x,1}}, SK_{x,1}^{r_{x,1}}\}_{\mathtt{I}_x \in \mathbb{I}_{min}}$ to the MU by randomly selecting $r_{x,1} \in \mathbb{Z}_q$.

2. Upon receiving $token_1$, the MU randomly selects $r_{x,2} \in \mathbb{Z}_q$, and then computes $token_2 = \{g^{r_{x,1}}, SK_{x,1}^{r_{x,1}r_{x,2}}, SK_{x,2}^{r_{x,2}}, \}_{\mathtt{I}_x \in \mathbb{I}_{min}}$ and $\{C'_{2,x} = g^{r_{x,1}r_{x,2}}\}_{\mathtt{I}_x \in \mathbb{I}_{min}}$. Subsequently, it transfers $token_2$ to the SSP but keeps $\{C'_{2,x}\}_{\mathtt{I}_x \in \mathbb{I}_{min}}$.

5. GenChallenge$(params, token_2) \rightarrow challenge'$. In this phase, the SSP generates and transfers a challenge to the IdPs through the MU and the MC as follows:

1. On receiving $token_2$, the SSP picks a random number $s \in \mathbb{Z}_q$ to compute $C_0 = e(g,g)^s$ for validation in the end. It also selects a random vector $v \in \mathbb{Z}_q^l$ with s as its first entry, and a random vector $w \in \mathbb{Z}_q^l$ with 0 as its first entry. We use A_x to denote the row x of A. For each row A_x, it picks a random secret $R_x \in \{0,1\}^m$, and calculates $\lambda_x = A_x \cdot v$, $w_x = A_x \cdot w$ and $r_{x,3} = H_3(R_x)$, thereby deriving the challenge message as below:
   $challenge = (A, \{C_{1,x}, C_{3,x}, C_{4,x}\}_{\mathtt{I}_x \in \mathbb{I}_{min}})$,

   where

$$\begin{aligned} C_{1,x} &= e(g,g)^{\lambda_x} e(g^{r_{x,1}}, SK_{x,2}^{r_{x,2}})^{r_{x,3}} \\ &= e(g,g)^{\lambda_x} e(g,g)^{\alpha_{\rho(x)} r_x}, \\ C_{3,x} &= SK_{x,1}^{r_{x,1}r_{x,2}r_{x,3}} g^{w_x} = g^{y_{\rho(x)} r_x} g^{w_x}, \\ C_{4,x} &= \{P^{r_x^*}, R_x \oplus H_2(g_x^{r_x^*})\}, \end{aligned}$$

   and $r_x = r_{x,1}r_{x,2}r_{x,3}$, $g_x = e(H_1(I_x), P_{pub})$.

2. The SSP sends $challenge$ to the MC through the MU, and the MU derives $challenge' = (A, \{C_{1,x}, C'_{2,x}, C_{3,x}, C_{4,x}\}_{\mathtt{I}_x \in \mathbb{I}_{min}})$ by appending $\{C'_{2,x}\}_{\mathtt{I}_x \in \mathbb{I}_{min}}$ and transfers it to the MC. The MC splits and forwards the corresponding part of $challenge'$ to each selected IdP respectively.

6. GenResponsex$(params, challenge') \rightarrow response_x$: In this phase, each IdP $\mathtt{I}_x$ performs an initial decryption over the challenge and outputs $response_x$ as follows:

1. The IdP I_x derives $r_{x,3}$ by applying IBE decryption to $C_{4,x}$ with its private key d_x:

$$\begin{aligned} & R_x \oplus H_2(g_x^{r_x^*}) \oplus H_2(e(d_x, P^{r_x^*})) \\ = \; & R_x \oplus H_2(e(H_1(I_x), P_{pub})^{r_x^*}) \oplus H_2(e(d_x, r_x^* P)) \\ = \; & R_x \oplus H_2(e(H_1(I_x), P^{\varepsilon})^{r_x^*}) \oplus H_2(e(H_1^{\varepsilon}(I_x), P^{r_x^*})) \\ = \; & R_x \end{aligned}$$

and retrieves $r_{x,3} = H_3(R_x)$ thereafter. Then it derives $C_{2,x} = (C'_{2,x})^{r_{x,3}} = g^{r_x}$.

2. Subsequently, the IdP I_x looks up GID_u in the registered user table, and computes $response_x$:
$\frac{C_{1,x} \cdot e(H_1(\mathrm{GID}_u), C_{3,x})}{e(K_{\rho(x),u}, C_{2,x})} = e(g,g)^{\lambda_x} e(H_1(\mathrm{GID}_u), g)^{w_x}$ which is then sent back to the MC.

7. GenResponse$(params, \{response_x\}_{I_x \in \mathbb{I}_{min}}) \rightarrow response$: In this phase, the MC completes the final decryption as below:

1. Upon receiving $response_x$ from each IdP x, the MC selects constants $c_x \in \mathbb{Z}_q$ such that $\sum_x c_x A_x = (1, 0, \cdots, 0)$. Thereafter it computes $response$ as below:

$$\prod_x (e(g,g)^{\lambda_x} e(H_1(\mathrm{GID}_x), g)^{w_x})^{c_x} \stackrel{?}{=} e(g,g)^{s'}.$$

2. the MC sends $response$ to the SSP through the MU.

8. Verification the SSP verifies $response$ against C_0. If they are equal, then the MU is granted the access to log in; otherwise the MU's access request is rejected.

5.1.3 Security Analysis and Performance Evaluation

In this section, we provide the security assessment and the evaluate results regarding the participating parties.

Security Analysis

As the security of DP-MAC mainly relies on IBE and DABE, its security strength can be also proved using the same proofs in [64] and [63]. In addition, we use the two assumptions below to ensure the security in the following part.

DEFINITION 1 (Discrete Logarithm Problem(DLP) Assumption)**.** Given $g, g^x \in \mathbb{G}$ where $x \in \mathbb{Z}_n^*$, then for any probabilistic polynomial-time (PPT) algorithm $\mathcal{A}$ and negligible ε, we have $Pr[\mathcal{A}(g, g^x) = x] \leq \varepsilon$.

DEFINITION 2 (Decisional Diffie-Hellman Problem(DDH))**.** Given a randomly chosen 4-tuple $(g, g^a, g^b, g^c) \in \mathbb{G}^4$ with $a, b \in \mathbb{Z}_n^*$ to determine whether $c = ab$ or not. The DDH assumption holds in $\mathbb{G}$ if no PPT algorithm $\mathcal{A}$ has non-negligible advantage ε in solving the DDH problem in $\mathbb{G}$.

Theorem 1: DP-MAC is secure against the authentication spoofing attack committed by the MU.

Proof: DP-MAC ensures that the MU cannot prove her authenticity alone without delegating the authentication tasks to all the IdPs required by the access policy. In practice, the MU is motivated to spoof the SSP that she has been authenticated by all the target IdPs, which is not the truth. Nonetheless, the SSP generates a blinding factor R_x for each IdP I_x, and forwards it to the IdP I_x through the MU and the MC after encrypting it with I_x's public key. The cryptographic strength of IBE prevents the MU and her dedicated virtual machine from deriving R_x without the help of I_x. As such, the MU cannot derive $C_{2,x}$, and the cryptographic strength of DABE ensures that the MU cannot compute the correct *response* for validation without the knowledge of $C_{2,x}$. In addition, given $C_{1,x}$, $C_{3,x}$ and $token_2$, the blinding factor R_x prevents the MU from deducing $e(g, g)^{\lambda_x}$ and g^{w_x} due to the hardness of

DLP. As a result, the MU cannot derive the correct response and authenticate herself to the SSP.

Theorem 2: DP-MAC is secure against the impersonation attack launched by the SSP.

Proof: DP-MAC ensures that the malicious SSP cannot impersonate an authenticated MU with all the information in prior authentication and access control process. In actuality, the malicious SSP might impersonate the MU in previous authentication process to get validation from the RP in a new authentication process by reusing the received $token_2$. Nevertheless, the MU didn't disclose $C'_{2,x} = g^{r_{x,1}r_{x,2}}$ to the malicious SSP, and the SSP has to derive $C'_{2,x}$ in order to be successfully authenticated due to the cryptographic strength of DABE. Note that $token_2$ takes the form of $\{g^{r_{x,1}}, g^{y_x r_{x,1} r_{x,2}}, g^{\alpha_i r_{x,2}}\}$, and the unknown y_x renders it impossible for the SSP to derive $g^{r_{x,1}r_{x,2}}$ from $g^{y_x r_{x,1} r_{x,2}}$. Assume the malicious SSP has made a successful guess of $g^{r_{x,2}}$ from $g^{\alpha_i r_{x,2}}$, the hardness of DDH ensures that the SSP can derive $g^{r_{x,1}r_{x,2}}$ from $g^{r_{x,1}}$ and $g^{r_{x,2}}$ with a negligible probability in polynomial time. Hence DP-MAC prevents the SSP from launching any impersonation attacks against the MU.

Furthermore, the decentralized architecture enables DP-MAC to be robust towards the fault of any system component, and thus it mitigates the SPOF problem. Also, k' out of k PKGs are employed to generate secret key for IdP, and the adversary cannot get any advantage if less than k' PKGs are compromised. Note that the MU can register a dedicated virtual machine in any cloud, and the corruption of one cloud platform wouldn't impact other MUs in other cloud platforms.

In addition, DP-MAC protects the privacy of the MU against HBC attacks from the IdPs, because the SSP delegates the authentication and access control to the IdPs through MU without direct interactions with IdPs, and the challenge messages do not contain any information about the SSP. Both the IdPs and the MC cannot gain any

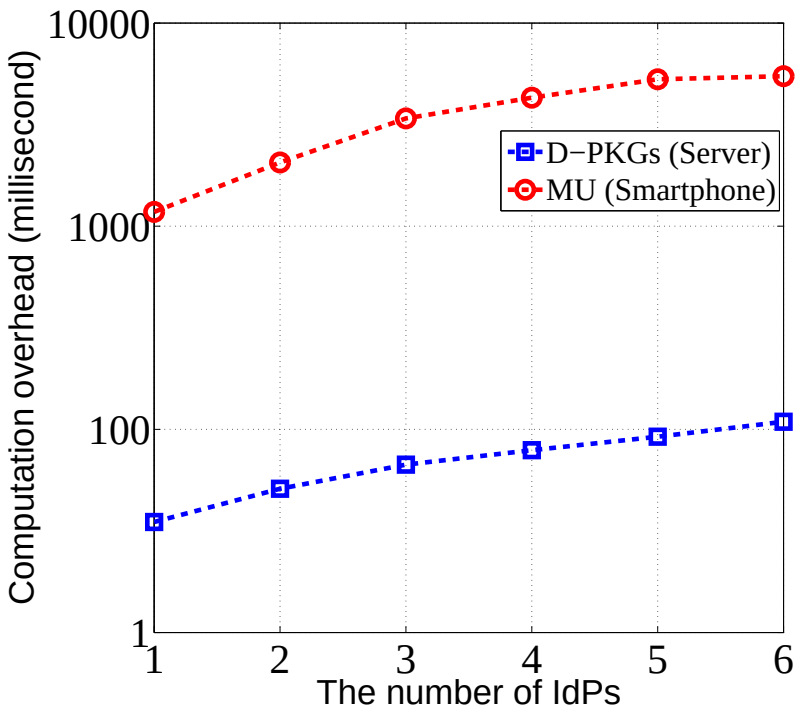

Figure 5.6: The Computational Cost in System Setup Phase

knowledge about the SSP, as the MU is placed between the MC and the SSP, and it cannot infer the SSP from the tokens and the challenge messages.

Last but not the least, DP-MAC can prevent identity theft in case of mobile device loss, because the adversary cannot establish SSL/TLS channel with the dedicated virtual machine in the MC, such that it is unable to complete the authentication and access control process using $SK_{x,2}$ only.

Performance Evaluation

In this subsection, we implement DP-MAC in Mobicloud [71] and android smartphone. The MU uses Samsung Galaxy Nexus with Dual core ARM Cortex-A9 at 1.2GHz and 1GB memory running Android 4.2 (Jelly Bean), and the other participating parties are simulated by virtual machines with Intel Core i3-2100 CPU at

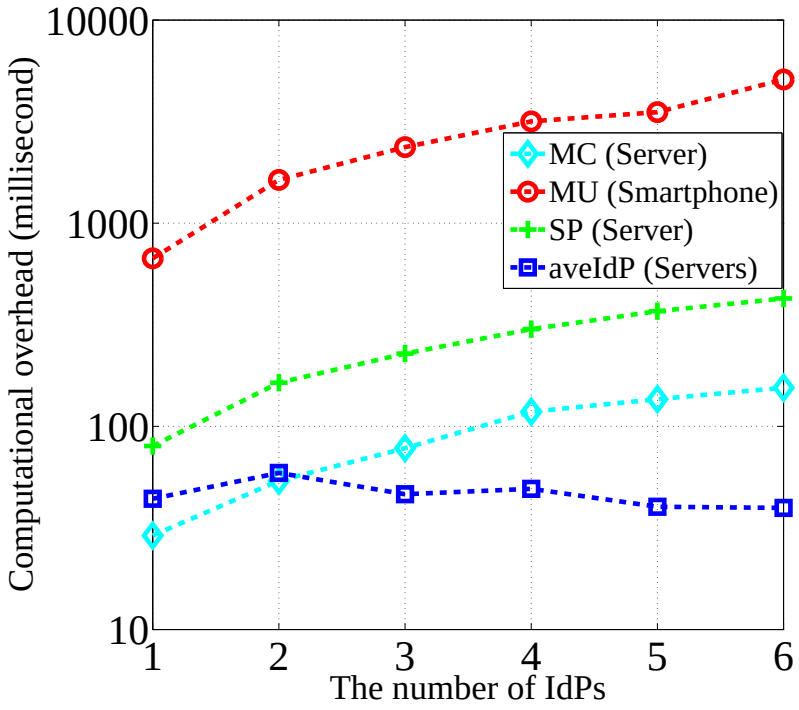

Figure 5.7: The Computational Cost in Authentication and Access Control Phase

3.10GHz and 2GB memory running 64-bit Ubuntu Lucid Lynx (Ubuntu 10.04) hosted by Mobicloud. The experiment utilizes Java Pairing-Based Cryptography (jPBC) library [65] based on Type-A ECC curve which features the fastest group operation. It is on the supersingular curve $y^2 = x^3 + x$ over the 512-bit finite field.

From Figure 5.6 it can be seen that as the number of associated IdPs grow from 1 to 6, the computation time of D-PKGs increases from 35 milliseconds to 109 milliseconds, while the computation time of the MU for generating credential and secret keys in *MURegister* increases from 1176 milliseconds to 5473 milliseconds. As the algorithms in the system setup phase run only one time, the computational overhead of the MU are acceptable in real life, as the users are unlikely to register at too many IdPs. Figure 5.7 illustrates the computational overhead in the authentication and access control phase. We do not consider the time cost resulting from

$MULoginRequest$ and $Verification$, as they do not involve any cryptographic operations, which are computation-intensive. It can be seen that as the number of selected IdPs grow from 1 to 6, the computation time of MC increases from 29 milliseconds to 155 milliseconds, and the computation time of the SSP increase from 80 milliseconds to 426 milliseconds, while that of the MU grows from 671 milliseconds to 5098 seconds. The computation time of IdPs on average keeps around 45 milliseconds. We believe the computational cost is acceptable for the MU, as this is only one-time authentication and access control phase, and the MU can be assigned a session key by the SSP to perform symmetric encryption after validation.

5.2 Centralized Sensing Data Access Control

The ID-revocable ABE scheme has the same cryptographic preliminaries as those in DP-MAC. Bilinear pairings are the basic operations and linear secret sharing are utilized in ID-revocable ABE. Assume $\mathbb{G}$ and $\mathbb{G}_T$ are two cyclic groups with order q generated by a Bilinear Diffie-Hellman (BDH) parameter generator $\mathcal{G}$. Correspondingly a bilinear map system $\mathbb{S} = (q, \mathbb{G}, \mathbb{G}_T, e)$ could be set up where e denotes a computable bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$.

In the following, it is first presented as single-ID revocation version, followed by multiple-ID revocation version. It is comprised of four algorithms, and the implications of the algorithms in ID-revocable ABE are elaborated as below:

- $Setup(\mathcal{U}, \mathcal{I})$: Given an identity set $\mathcal{I}$ and an attribute set $\mathbb{I}$, the trust authority publishes its public key PK but keep its master key MSK ;
- $KeyGen(MSK, S, ID)$: Given MSK, SSC's ID and attribute set S, the trust authority issues private keys SK;
- $Encrypt(PK, (M, \rho), \mathcal{M}, \{ID_j\})$: Given the public key PK, the LSSS matrix

M and its corresponding mapping ρ to each attribute, the message $\mathcal{M}$ and the revoked ID set $\{ID_j\}$, the MDO generates the ciphertext and send it to the Cloud Provider;

- $Decrypt(CT, SK)$: Given the ciphertext CT, the SSC derives the message $\mathcal{M}$ by decrypting with its private key SK.

5.2.1 Security Model

Setup: The challenger runs the Setup algorithm and gives the public parameters, PK to the adversary.

- $Phase1$. The adversary makes repeated private keys query corresponding to sets of attributes $P_1, ..., P_{q1}$ where each attribute set is owned by an entity with identity ID.

Challenge: The adversary submits two equal length messages M_0 and M_1. In addition the adversary gives a challenge access structure $\mathbb{A}^*$ and a set S of revoked identities such that S must include all identities that were queried and none of the sets $\mathcal{P}_1, ..., \mathcal{P}_{q1}$ from Phase 1 satisfy the access structure if its owner identity is not involved in S. The challenger picks up a random coin b, and encrypts M_b under the access structure $\mathbb{A}^*$ and . The ciphertext CT is given to the adversary.

- $Phase2$. Phase 1 is repeated with the restriction that none of sets of attributes $\mathcal{P}_{q1+1}, ..., \mathcal{P}_q$ where the corresponding owner's identity ID is not in S satisfy the access structure corresponding to the challenge or the corresponding owner's identiy is in the revocation set S.

Guess The adversary outputs a guess b' of b.

The advantage of an adversary A in this game is defined as $Pr[b' = b] - 1$. We note that the model can easily be extended to handle chosen-ciphertext attacks by allowing for decryption queries in Phase 1 and Phase 2.

A ciphertext-policy attribute-based encryption scheme is secure if all polynomial time adversaries have at most a negligible advantage in the above game. We say that a system is selectively secure if we add an Init stage before setup where the adversary commits to the challenge access structure $\mathbb{A}^*$ and the revocation ID set S. All of our constructions will be proved secure in the selective security model.

5.2.2 Assumptions

Decisional Bilinear Diffie-Hellman Assumption The decisional Bilinear Diffie-Hellman problem is defined as follows. We choose a group $\mathbb{G}$ of prime order p and a random generator g of $\mathbb{G}$ and random exponents $c_1, c_2, c_3 \in \mathbb{Z}_p$. Given $\overrightarrow{y} = \{g, g^{c_1}, g^{c_2}, g^{c_3}\}$, it is hard for the attacker to distinguish $e(g,g)^{c_1 c_2 c_3} \in \mathbb{G}_T$.

An algorithm $\mathbb{B}$ that outputs $z \in \{0,1\}$ has advantage ϵ in solving decisional BDH in $\mathbb{G}$ if

$$|Pr[\mathbb{B}(\overrightarrow{y}, T = e(g,g)^{c_1 c_2 c_3}) = 0] - Pr[\mathbb{B}(\overrightarrow{y}, T = R) = 0]| \geq \epsilon$$

The decisional BDH assumption holds if only negligible advantage exists for any algorithm to solve the decisional BDH problem in polynomial time.

q-Decisional Multi-Exponent Bilinear Diffie-Hellman Assumption We herein make an assumption named the q decisional Multi-Exponent Bilinear Diffie-Hellman assumption to prove the security and it is defined as follows. We select a group $\mathbb{G}$ of prime order p. A challenger picks a generator $g \in \mathbb{G}$ and random exponents $s, \alpha, \alpha_1, \alpha_2, \cdots, \alpha_q$. Given

$$g, g^s, e(g,g)^{\alpha}, \{g^{\alpha_i}, g^{\alpha_i s}, g^{\alpha_i \alpha_j}, g^{\alpha/\alpha_i^2}, \}_{1 \leq i,j \leq q}, \{g^{\alpha_i \alpha_j s}, g^{\alpha \alpha_j/\alpha_i^2}, g^{\alpha \alpha_i \alpha_j/\alpha_k^2}, g^{\alpha \alpha_i^2/\alpha_j^2}\}_{1 \leq i,j,k \leq q, i \neq j}$$

it remains hard for the attacker to distinguish $e(g,g)^{\alpha s} \in \mathbb{G}_T$ from a random element in $\mathbb{G}_T$.

An algorithm $\mathbb{B}$ that outputs $z \in \{0,1\}$ has advantage ϵ in solving decisional q-parallel BDHE in $\mathbb{G}$ if

$$|Pr[\mathbb{B}(\overrightarrow{y}, T = e(g,g)^{\alpha s}) = 0] - Pr[\mathbb{B}(\overrightarrow{y}, T = R) = 0]| \geq \epsilon$$

These four algorithms are detailed in the following subsections.

5.2.3 CP-ABE Scheme with single ID Revocation

The following presents the construction of centralized CP-ABE with one ID revocation.

a. Setup$(\mathcal{U}, \mathcal{I})$

The algorithm takes an attribute set $\mathcal{U}$ and an identity set $\mathcal{I}$ as input where $|\mathcal{U}| = m$ and $|\mathcal{I}| = n$. It chooses a group $\mathbb{G}$ of prime order p, a generator p and m random group elements $h_1, h_2, \cdots, h_m \in \mathbb{G}$ that are associated with the m attributes in the system. It also chooses random exponents $\alpha, b \in \mathbb{Z}_p$.

Therefore, the public keys are output as:

$$PK = \{g, g^b, g^{b^2}, e(g,g)^{\alpha}, h_1^b, \cdots, h_m^b\}$$

The Master secret key

$$MSK = \{\alpha, b\}$$

b. KeyGen$(\mathbf{MSK}, \mathbf{S}, \mathbf{ID})$

S is the attribute set of user $ID \in \mathcal{I}$. The algorithm chooses a random $t \in \mathbb{Z}_p$ and derive the secret keys as follows:

$$SK = (K = g^{\alpha} g^{b^2 t}, \{K_x = (g^{b \cdot ID} h_x)^t\}_{\forall x \in S}, L = g^{-t})$$

c. Encrypt$(\mathbf{PK}, (\mathbf{M}, \rho), \mathcal{M}, \mathbf{ID_j})$ It takes the input as an LSSS access infrastructure (M, ρ) and the function ρ associates rows of M to attributes. ID_j is assumed to be the identity which will be revoked. Let M be an $l \times n'$ matrix. The algorithm first chooses a random vector $v = (s, y_2, \cdots, y_{n'}) \in \mathbb{Z}_p^{n'}$. These values will be used to share the encryption exponent s. For $x \in [1, l]$, it calculates $\lambda_x = v \cdot M_x$, where M_x is the vector corresponding to the x-th row of M. The algorithm chooses random $r_1, \cdots, r_l \in \mathbb{Z}_p$. It generates the first part of ciphertext:

$$C = \mathcal{M}e(g,g)^{\alpha s}, C_0 = g^s, \hat{C} = \{C_1^* = g^{b \cdot \lambda_1}, C_1' = (g^{b^2 \cdot ID_j} h_{\rho(1)}^b)^{\lambda_1}, \cdots,$$

$$C_l^* = g^{b \cdot \lambda_l}, C_l' = (g^{b^2 \cdot ID_j} h_{\rho(l)}^b)^{\lambda_l}\}$$

d. Decrypt$(\mathbf{CT}, \mathbf{SK})$ $CT = (C, C_0, \hat{C}, (M, \rho))$ is the input ciphertext and SK is a private key for a set S. Suppose that S satisfies the access structure and let $I \subset \{1, 2, ..., l\}$ be defined as $I = \{i : \rho(i) \in S\}$. Let $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ be a set of constants such that if $\{\lambda_i\}$ are valid shares of any secret s according to M, then $\Sigma_{i \in I} \omega_i \lambda_i = s$. If the identity ID combined in the SK is not equal to the revocation identity ID_j in the ciphertext, this step proceeds as follows:

$$
\begin{aligned}
& \frac{e(C_0, K)}{(\prod_{i \in I}[e(K_{\rho(i)}, C_i^*) \cdot e(L, C_i')]^{\omega_i})^{1/(ID-ID_j)}} \\
= & \frac{e(g^s, g^{\alpha} g^{b^2 t})}{(\prod_{i \in I}[e((g^{b \cdot ID} h_{\rho(i)})^t, g^{b \cdot \lambda_i}) \cdot e(g^{-t}, (g^{b^2 \cdot ID_j} h_{\rho(i)}^b)^{\lambda_i})]^{\omega_i})^{1/(ID-ID_j)}} \\
= & \frac{e(g^s, g^{\alpha}) \cdot e(g^s, g^{b^2 t})}{(\prod_{i \in I}[e(g^{b \cdot ID \cdot t}, g^{b \cdot \lambda_i}) \cdot e(h_{\rho(i)}^t, g^{b \cdot \lambda_i}) \cdot e(g^{-t}, g^{b^2 \cdot ID_j \cdot \lambda_i}) \cdot e(g^{-t}, h_{\rho(i)}^{b \cdot \lambda_i})]^{\omega_i})^{1/(ID-ID_j)}} \\
= & \frac{e(g, g)^{\alpha s} \cdot e(g, g)^{b^2 s t}}{(\prod_{i \in I}[e(g^{b \cdot ID \cdot t}, g^{b \cdot \lambda_i}) \cdot e(g^{-t}, g^{b^2 \cdot ID_j \cdot \lambda_i})]^{\omega_i})^{1/(ID-ID_j)}} \\
= & \frac{e(g, g)^{\alpha s} \cdot e(g, g)^{b^2 s t}}{(\prod_{i \in I}[e(g, g)^{b^2 t \lambda_i (ID-ID_j)}]^{\omega_i})^{1/(ID-ID_j)}} \\
= & \frac{e(g, g)^{\alpha s} \cdot e(g, g)^{b^2 s t}}{(\prod_{i \in I} e(g, g)^{b^2 t \lambda_i \omega_i})} \\
= & \frac{e(g, g)^{\alpha s} \cdot e(g, g)^{b^2 s t}}{e(g, g)^{b^2 t \sum_{i \in I} \lambda_i \omega_i}} \\
= & e(g, g)^{\alpha s}
\end{aligned}
$$

From the algorithms described above, it can be learned that the size of the ciphertext grows linearly with the number of involved attributes. In addition, both the encryption and decryption cost also increase in proportion to the number of involved attributes.

5.2.4 CP-ABE Scheme with Multiple IDs Revocation

The following describes the construction of centralized CP-ABE with multiple ID revocation.

a. Setup$(\mathcal{U}, \mathcal{I})$

The algorithm takes an attribute set $\mathcal{U}$ and an identity set $\mathcal{I}$ as input where $|\mathcal{U}| = m$ and $|\mathcal{I}| = n$. It chooses a group $\mathbb{G}$ of prime order p, a generator p and m

random group elements $h_1, h_2, \cdots, h_m \in \mathbb{G}$ that are associated with the m attributes in the system. It also chooses random exponents $\alpha, b \in \mathbb{Z}_p$.

Therefore, the public keys are output as:

$$PK = \{g, g^b, g^{b^2}, e(g,g)^\alpha, h_1^b, \cdots, h_m^b\}$$

The Master secret key

$$MSK = \{\alpha, b\}$$

b. KeyGen(MSK, S, ID)

S is the attribute set of user $ID \in \mathcal{I}$. The algorithm chooses a random $t \in \mathbb{Z}_p$ and derive the secret keys as follows:

$$SK = (K = g^\alpha g^{b^2 t}, \{K_x = (g^{b \cdot ID} h_x)^t\}_{\forall x \in S}, L = g^{-t})$$

c. Encrypt(PK, (M, ρ), $\mathcal{M}$, S) It takes the input as an LSSS access infrastructure (M, ρ) and the function ρ associates rows of M to attributes. ID_j is assumed to be the identity which will be revoked. Let M be an $l \times n'$ matrix. The algorithm first chooses a random vector $v = (s, y_2, \cdots, y_{n'}) \in \mathbb{Z}_p^{n'}$. These values will be used to share the encryption exponent s. For $x \in [1, l]$, it calculates $\lambda_x = v \cdot M_x$, where M_x is the vector corresponding to the x-th row of M. Let $r = |S|$ and ID_j denote the j-th identity in S. The algorithm chooses random $\mu_1, ..., \mu_r \in \mathbb{Z}_p$ such that $\mu = \mu_1 + ... + \mu_r$. It generates the first part of ciphertext:

$$C = \mathcal{M} e(g,g)^{\alpha s \mu}, C_0 = g^{s\mu}$$

$$\begin{array}{ccccc}
C_{1,1}^* = g^{b \cdot \lambda_1 \mu_1}, C'_{1,1} = (g^{b^2 \cdot ID_1} h_{\rho(1)}^b)^{\lambda_1 \mu_1} & \cdots & C_{l,1}^* = g^{b \cdot \lambda_l \mu_1}, C'_{l,1} = (g^{b^2 \cdot ID_1} h_{\rho(l)}^b)^{\lambda_l \mu_1} \\
C_{1,2}^* = g^{b \cdot \lambda_1 \mu_2}, C'_{1,2} = (g^{b^2 \cdot ID_2} h_{\rho(1)}^b)^{\lambda_1 \mu_2} & \cdots & C_{l,2}^* = g^{b \cdot \lambda_l \mu_2}, C'_l = (g^{b^2 \cdot ID_2} h_{\rho(l)}^b)^{\lambda_l \mu_2} \\
\cdot & \cdot & \cdot \\
C_{1,r}^* = g^{b \cdot \lambda_1 \mu_r}, C'_{1,r} = (g^{b^2 \cdot ID_r} h_{\rho(1)}^b)^{\lambda_1 \mu_r} & \cdots & C_l^* = g^{b \cdot \lambda_l \mu_r}, C'_{l,r} = (g^{b^2 \cdot ID_r} h_{\rho(l)}^b)^{\lambda_l \mu_r}
\end{array}$$

d. Decrypt($\mathbf{CT}, \mathbf{SK}$) CT is the input ciphertext with access structure (M, ρ) and SK is a private key for a set S. Suppose that S satisfies the access structure and let $I \subset \{1, 2, ..., l\}$ be defined as $I = \{i : \rho(i) \in S\}$. Let $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ be a set of constants such that if $\{\lambda_i\}$ are valid shares of any secret s according to M, then $\Sigma_{i \in I} \omega_i \lambda_i = s$. If the identity ID combined in the SK is not equal to the revocation identity ID_j in the ciphertext, we can perform

$$
\begin{aligned}
& \frac{e(C_0, K)}{\prod_{i \in I}(\prod_{j=1}^{r}[e(K_{\rho(i)}, C^*_{i,j}) \cdot e(L, C'_{i,j})]^{1/(ID - ID_j)})^{\omega_i}} \\
= & \frac{e(g^{s\mu}, g^{\alpha} g^{b^2 t})}{\prod_{i \in I}(\prod_{j=1}^{r}[e((g^{b \cdot ID} h_{\rho(i)})^t, g^{b \cdot \lambda_i \mu_j}) \cdot e(g^{-t}, (g^{b^2 \cdot ID_j} h^b_{\rho(i)})^{\lambda_i \mu_j})]^{1/(ID - ID_j)})^{\omega_i}} \\
= & \frac{e(g^{s\mu}, g^{\alpha}) \cdot e(g^{s\mu}, g^{b^2 t})}{\prod_{i \in I}(\prod_{j=1}^{r}[e(g^{b \cdot ID \cdot t}, g^{b \cdot \lambda_i \mu_j}) \cdot e(h^t_{\rho(i)}, g^{b \cdot \lambda_i \mu_j}) \cdot e(g^{-t}, g^{b^2 \cdot ID_j \cdot \lambda_i \mu_j}) \cdot e(g^{-t}, h^{b \cdot \lambda_i \mu_j}_{\rho(i)})]^{1/(ID - ID_j)})^{\omega_i}} \\
= & \frac{e(g, g)^{\alpha s \mu} \cdot e(g, g)^{b^2 s \mu t}}{\prod_{i \in I}(\prod_{j=1}^{r}[e(g^{b \cdot ID \cdot t}, g^{b \cdot \lambda_i \mu_j}) \cdot e(g^{-t}, g^{b^2 \cdot ID_j \cdot \lambda_i \mu_j})]^{1/(ID - ID_j)})^{\omega_i}} \\
= & \frac{e(g, g)^{\alpha s \mu} \cdot e(g, g)^{b^2 s \mu t}}{\prod_{i \in I}(\prod_{j=1}^{r}[e(g, g)^{b^2 t \lambda_i \mu_j (ID - ID_j)}]^{1/(ID - ID_j)})^{\omega_i}} \\
= & \frac{e(g, g)^{\alpha s \mu} \cdot e(g, g)^{b^2 s \mu t}}{\prod_{i \in I}(\prod_{j=1}^{r} e(g, g)^{b^2 t \lambda_i \mu_j})^{\omega_i}} \\
= & \frac{e(g, g)^{\alpha s \mu} \cdot e(g, g)^{b^2 s \mu t}}{\prod_{i \in I}(e(g, g)^{(\sum_{j=1}^{r} \mu_j) b^2 t \lambda_i})^{\omega_i}} \\
= & \frac{e(g, g)^{\alpha s \mu} \cdot e(g, g)^{b^2 s \mu t}}{e(g, g)^{b^2 t \mu \sum_{i \in I} \lambda_i \omega_i}} \\
= & \ e(g, g)^{\alpha s \mu}
\end{aligned}
$$

From the algorithms described above, it can be learned that the size of the ciphertext grows linearly with the product between the number of involved attributes

and the number of revoked IDs. In addition, both the encryption and decryption cost also increase in proportion to the product between the number of involved attributes and the number of revoked IDs.

5.2.5 Security Proof

THEOREM 2. *Suppose the decisional q-MEBDH assumption holds. Then no poly-time adversary can selectively break our system with a ciphertext encrypted to $r^* \leq q$ revoked users.*

Proof. Suppose there is an adversary $\mathcal{A}$ with non-neglibile advantage $\epsilon = \text{Adv}_{\mathcal{A}}$ in the selective security game against our scheme. Moreover, suppose $\mathcal{A}$ attacks our construction with a ciphertext of at most q revoked users. We show how to build simulator,$\mathcal{B}$,that plays the decisional q-MEBDH problem.

Init The simulator takes in q-MEDDH challenge $\overrightarrow{X}, T$. Then the adversary gives the simulator algorithm a revocation set $S^* = ID_1, ..., ID_{r^*}$ of size $r^* \leq q$.

Setup The simulator chooses random $a_1, a_2, ..., a_{r^*}$ and implicitly sets $b = a_1 + a_2 + \cdots a_{r^*}$ by computing the public parameters as

$$g, g^b = \prod_{1 \leq i,j \leq r^*} g^{a_i}, g^{b^2} = \prod_{1 \leq i,j \leq r^*} (g^{a_i \cdot a_j})$$

To reflect the revocation set S^* in the public parameters $h_1, ..., h_U$, the algorithm chooses a random value z_x for each x for $1 \leq x \leq U$. Let X denote the set of indices i, such that $\rho(i) = x$. We program h_x as

$$h_x = \prod_{1 \leq i \leq r^*} g^{-a_i ID} \cdot g^{z_x}$$

Then the simulator publishes the above parameters $(g, g^b, g^{b^2}, h_1^b, h_2^b, ..., h_U^b, e(g,g)^\alpha)$ as public key. We observe that the public parameters are distributed randomly as the real system.

Phase I The algorithm simulates to answer private key queries. To construct all private keys in the revocation set S^*, the simulator chooses a random value θ_i for each identity ID_i and implicitly set the randomness t_i of the i-th identity as $t_i = -\alpha/a_i^2 + \theta_i$. By doing this, we can first cancel out the g^{α_i} that the simulator does not know in building the K component. Second, we can prevent the appearance of the term of the form g^{α/a_i} which would have been produced in programming several terms of the form $g^{\alpha a_j/a_i^2}$ of D_2 components in the case of $i = j$.

$$K = (\prod_{1 \le j,k \le n s.t. if j=k then j,k \ne i} g^{(-\alpha a_j a_k/a_i^2)}) \prod_{1 \le j,k \le n} (g^{a_i a_j})^{\theta_i}$$

$$K_x = (\prod (g^{-\alpha \cdot a_j/a_i^2})^{(ID_i - ID_j)} (g^{(ID_i - ID_j)\cdot a_j})^{y_i}) (g^{-\alpha/a_i^2})^{z_x} g^{z_x \theta_i}$$

$$L = g^{\alpha/a_i^2} g^{-\theta_i}$$

Challenge The adversary provides to the simulator two challenge messages M_0,M_1 with the challenge matrix M^* of dimension at most q columns. The simulator will choose random value $y_2', y_3', ..., y_{n^*}$ and share the secret s using the vector

$$\overrightarrow{v} = (s, y_2', y_3', ..., y_{n^*}).$$

Then the simulator can calculate $\lambda_i = v \cdot M^*$. The simulator continues to choose another r^* random values $\mu', \mu_1', \mu_2', ..., \mu_{r^*}$ such that $mu' = \sum_i mu_i'$. In computing the ciphertext, we will use randomness $\widetilde{\mu} = \mu + \mu'$ which can be conceptually broken into shares $\widetilde{\mu}_i = a_i \mu/b + \mu'$. As $b = \sum_j a_j$, $\sum_i \widetilde{\mu}_i = \widetilde{\mu}$. Finally, the simulator creates the challenge ciphtext CT as

$$C = (T \cdot e(g,g)^{\alpha\mu'})^s \cdot \mathcal{M}_\beta, C_0 = g^\mu \cdot g^{\mu'}$$

$$C^*_{i,j} = (g^{\mu a_i}(\prod_k g^{a_k})^{\mu'})^{sM^*_{j,1}+y'_2 M^*_{j,2},\ldots,+M^*_{j,n^*}}$$

$$C'_{i,j} = ((\prod_{1\le k\le r^* i\neq k} (g^{\mu a_i a_i a_k})^{ID_i - ID_k}) g^{a_i \mu z_\rho(i)} \cdot \phi_i^{\mu'_i})^{sM^*_{j,1}+y'_2 M^*_{j,2},\ldots,+M^*_{j,n^*}}$$

Guess The adversary will eventually output a guess β' of β. The simulator then outputs 0 to guess that $T = e(g,g)^\alpha \mu$ if $\beta' = \beta$; otherwise, it outputs 1 to indicate that it believes T is a random group element in $\mathbb{G}_T$. When T is a tuple the simulator $\mathcal{B}$ gives a perfect simulation so we have that

$$Pr[\mathcal{B}(\overrightarrow{X}, T = e(g,g)^{\alpha\mu}) = 0] = \frac{1}{2} + Adv_{\mathcal{A}}$$

When T is a random group element the message M_β is completely hidden from the adversary and we have $Pr[\mathcal{B}(\overrightarrow{X}, T = R) = 0] = \frac{1}{2}$. Therefore, B can play the decisional q-MEBDH game with non-negligible advantage.

□

5.3 Related Work

OpenID 2.0 [59] was developed as an open community-driven platform as a scalable user-centric identity infrastructure on the Internet. Seong *et al.* [72] architected a decentralized social networking infrastructure using the OpenID management system so users can retrieve data with their established personas. A framework named mSSL [73] is proposed to ensure user authenticity, data integrity and confidentiality in a peer-to-peer fasion. Bertino *et al.* [74] designed a privacy-preserving digital identity management system to authenticate users based on user identity properties, and the registrar signs on the commitment of users' attributes to generate certificate

for the users. Lin *et al.* [75] proposed a privacy-preserving protocol for vehicular communications based on group signature and identity-based signature. Chow *et al.* [76] presented SPICE whereby the users can authenticate themselves to the service providers with certificates when the registrar is offline, but the revoked users can use certificates to access services, which is still unknown to the service providers.

Shamir first proposed the concept of Identity-Based Cryptosystems [77] whereby the need for public-key certificates are eliminated. A practical and efficient Identity-Based Encryption (BF-IBE) [64] scheme was presented based on bilinear pairing in 1999, followed by SK-IBE [78], and BB_1-IBE [79]. Sahai and Waters proposed the first ABE scheme [80] in 2005 where an identity is viewed as a set of descriptive attributes. A derived version [81] was proposed to dynamically set up multicast groups with group membership anonymity. Melissa Chase demonstrated multi-authority ABE in [82] and it requires a central trusted party to issue the key to every user. An improved version [51] removes the central authority, but it requires all the attribute authorities to participate in the access control, which does not fit our scenario. The multi-authority ABE scheme proposed in [83] also requires a centralized authority to create the master key and accordingly generate keys to each user. In 2011, Allison Lewko and Brent Waters presented a multi-authority attribute-based encryption system DABE [63] in which no preset access structure exists and the key generation authorities work independently from each other. Our DP-MAC is built on top of the integration of IBE and DABE and they bring robustness and scalability. At the same time, the usage of Linear Secret-Sharing Scheme allows the SSP to easily adapt the access control policy. The MC assists in the computation and communication with different IdPs for the MU, and it helps alleviate the threat of identity theft.

Boldyreva *et al.* [84] proposed an identity-based scheme with efficient user revocation capability. It applies key updates with significantly reduced computational cost

based on a binary tree data structure and it is resistance against chosen-ciphertext attack. Nevertheless, Libert *et al.* [85] notified the security problem left by Boldyreva that its security can only be proved in the selective-ID setting where adversaries need to reveal the victims' identities at the beginning of the game. Consequently, they proposed an identity-based encryption scheme with stronger adaptive-ID sense to address the remaining issue. Li *et al.* [86] first introduced outsourcing computation in identity-based encryption and presented a revocable in the server-aided settings. As a result, it achieves constant computation cost at public key generator and private key size at user, and the user does not have to contact public key generator for key update. Chen *et al.* [87] presented an identity-based encryption scheme based on lattices to realize efficient key revocation. Binary tree data structure is utilized to achieve logarithmic complexity in key updates. EASiER [88] architecture is described to support fine-grained access control policies and dynamic group membership based on attribute-based encryption. It relies on a proxy to participate in the decryption and enforce revocation, such that the user can be revoked without re-encrypting ciphertexts or issuing new keys to other users. Lewko *et al.* [67] two novel broadcast encryption schemes with effective user revocation capability. The first scheme is selectively secure in the standard model, and the second scheme achieves adaptive security by exploiting dual encryption technique. The ciphertext size only relates to the number of revoked users and the size of public/private keys are constant. Yu *et al.* [87] proposed an attribute-based data sharing scheme where each attribute gets three distinct values for its positive form, negative form as well as "don't care" form. It addressed the issue of attribute revocation by relying on a semi-trustable online proxy to perform re-encryption and take most laborious tasks.

5.4 Conclusion

In this chapter, a distributed identity management system DP-MAC is presented to address a few security concerns in a decentralized MCS environment. The distributed architecture of DP-MAC mitigates SPOF attack and it yields high robustness against attacks over different participating parties. The MU's privacy of historical access information to different services is also ensured by separating the IdPs from the SSP, such that the IdPs have no knowledge about the SSP that the MU tries to log into. The SSP can dynamically adjust its access policy without incurring additional cost, and it does not store any shared secrets ot certificates of the IdPs, thereby bringing a high scalability to the system.

An effective ID-revocable scheme is presented to provide an efficient user revocation mechanism in CP-ABE systems. Most existing research work rely on either key regeneration or complicated tree structure based on presumed relationships between identities and attributes, thereby resulting in overwhelming overhead in a dynamic system where users frequently join or leave. In this ID-revocable scheme, key regeneration for non-revoked users is not needed, and identities and attributes are uncorrelated, thereby ensuring the greatest flexibility. Its ciphertext size, encrytion and decryption cost are proportional to the product of the number of attributes and the number of identities.

Chapter 6

CONCLUSION AND FUTURE RESEARCH

In the past few years, Mobile Crows Sensing (MCS) arises as an irreversible trend to harness the immense capability of the crowd jointly with the persuasive sensor-equipped mobile devices. The increasing computation power, storage size and popularity of sensing-capable mobile devices, when paired with omnipresent Internet access, allow the unprecedented acquisition of knowledge about the individual owner itself and its ambient environment. At the same time, the ascension of the cloud computing platforms with ubiquitous connectivity, vast computational power and unbounded storage offer perfect backend infrastructure to integrate MCS for data transmission, processing and storage and propell the MCS paradigm from concept into reality.

Meanwhile, MCS present distinct research challenges including sensing task assignment and privacy preservation in sensing query processing, multi-party data computation and access control in data sharing. To address this issues, this dissertation presents detailed description of solutions along with extensive simulations and experiments to prove their feasibility. Greedy approaches and bee algorithms are used to minimize the cost or maximize the utility while adhering to the Quality of Service for sensing task assignment among regular participants. Homomorphic encryptions are exploited to protect the privacy of user profile, sensing query content and the numeric values of individual sensing readings. Mapreduce framework is discussed to accelerate the data processing time. Identity-based encryption and distributed attribute-based encryption are utilized for access control in federated identity management to avoid single-point-of-failure of identity providers and privacy protection of SSCs' data access

history.

Further improvements can be made in the future. The participant selection framework to cover both regular participants and opportunistic participants as our next step. Accordingly, our future participant selection framework is anticipated to achieve better performance and better fit into real life, as the mobility traces of most participants in real world are inconsistent and their submissions can also contribute to the sensing campaign. Better homomorphic encryption techniques need to be investigated for more accurate k-nearest neighbor search and they can be extended for privacy-preserving data clustering. The complexity of distributed data access control for federated identity management need to be reduced for better user experience.

REFERENCES

[1] IDC, "Record smartphone shipments grow the market 38.8of 2013, making way for a strong holiday quarter, according to idc," October 2013. [Online]. Available: http://www.idc.com/getdoc.jsp?containerId=prUS24418013

[2] Z. Liu, S. Parthasarthy, A. Ranganathan, and H. Yang, "Scalable event matching for overlapping subscriptions in pub/sub systems," in *Proceedings of the 2007 inaugural international conference on Distributed event-based systems*. ACM, 2007, pp. 250–261.

[3] E. Casalicchio, F. Morabito, G. Cortese, and F. Davide, "A novel approach to adaptive content-based subscription management in dht-based overlay networks," *Journal of Grid Computing*, vol. 4, no. 3, pp. 343–353, 2006.

[4] E. Casalicchio and F. Morabito, "Distributed subscriptions clustering with limited knowledge sharing for content-based publish/subscribe systems," in *Network Computing and Applications, 2007. NCA 2007. Sixth IEEE International Symposium on*. IEEE, 2007, pp. 105–112.

[5] Z. Jerzak and C. Fetzer, "Bloom filter based routing for content-based publish/-subscribe," in *Proceedings of the second international conference on Distributed event-based systems*. ACM, 2008, pp. 71–81.

[6] I. Aekaterinidis and P. Triantafillou, "Publish-subscribe information delivery with substring predicates," *Internet Computing, IEEE*, vol. 11, no. 4, pp. 16–23, 2007.

[7] M. M. Hassan, B. Song, and E.-N. Huh, "A dynamic and fast event matching algorithm for a content-based publish/subscribe information dissemination system in sensor-grid," *The Journal of Supercomputing*, vol. 54, no. 3, pp. 330–365, 2010.

[8] S. Reddy, D. Estrin, and M. Srivastava, "Recruitment framework for participatory sensing data collections," in *Pervasive Computing*. Springer, 2010, pp. 138–155.

[9] S. Choi, G. Ghinita, and E. Bertino, "Secure mutual proximity zone enclosure evaluation," in *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2014, pp. 133–142.

[10] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[11] G. Myles, A. Friday, and N. Davies, "Preserving privacy in environments with location-based applications," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 56–64, 2003.

[12] M. Youssef, V. Atluri, and N. R. Adam, "Preserving mobile customer privacy: an access control system for moving objects and customer profiles," in *Proceedings of the 6th international conference on Mobile data management*. ACM, 2005, pp. 67–76.

[13] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive computing*, vol. 2, no. 1, pp. 46–55, 2003.

[14] B. Palanisamy and L. Liu, "Mobimix: Protecting location privacy with mix-zones over road networks," in *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*. IEEE, 2011, pp. 494–505.

[15] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[16] B. Bamba, L. Liu, P. Pesti, and T. Wang, "Supporting anonymous location queries in mobile environments with privacygrid," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 237–246.

[17] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *Proceedings of the 32nd international conference on Very large data bases*. VLDB Endowment, 2006, pp. 763–774.

[18] C.-Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*. ACM, 2006, pp. 171–178.

[19] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Pervasive Services, 2005. ICPS'05. Proceedings. International Conference on*. IEEE, 2005, pp. 88–97.

[20] P. Shankar, V. Ganapathy, and L. Iftode, "Privately querying location-based services with sybilquery," in *Proceedings of the 11th international conference on Ubiquitous computing*. ACM, 2009, pp. 31–40.

[21] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order preserving encryption for numeric data," in *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*. ACM, 2004, pp. 563–574.

[22] W. K. Wong, D. W.-l. Cheung, B. Kao, and N. Mamoulis, "Secure knn computation on encrypted databases," in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*. ACM, 2009, pp. 139–152.

[23] A. Khoshgozaran and C. Shahabi, "Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy," in *Advances in Spatial and Temporal Databases*. Springer, 2007, pp. 239–257.

[24] R. Ostrovsky and W. E. Skeith III, "A survey of single-database private information retrieval: Techniques and applications," in *Public Key Cryptography–PKC 2007*. Springer, 2007, pp. 393–411.

[25] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: anonymizers are not necessary," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 121–132.

[26] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Prive: anonymous location-based queries in distributed mobile systems," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 371–380.

[27] M. Naor and B. Pinkas, "Oblivious transfer with adaptive queries," in *Advances in CryptologyCRYPTO99*. Springer, 1999, pp. 573–590.

[28] S. Papadopoulos, S. Bakiras, and D. Papadias, "Nearest neighbor search with strong location privacy," *Proceedings of the VLDB Endowment*, vol. 3, no. 1-2, pp. 619–629, 2010.

[29] R. Paulet, M. G. Kaosar, X. Yi, and E. Bertino, "Privacy-preserving and content-protecting location based queries," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26, no. 5, pp. 1200–1210, 2014.

[30] S. Wang, X. Ding, R. H. Deng, and F. Bao, "Private information retrieval using trusted hardware," in *Computer Security–ESORICS 2006*. Springer, 2006, pp. 49–64.

[31] M. Riahi, T. G. Papaioannou, I. Trummer, and K. Aberer, "Utility-driven data acquisition in participatory sensing," in *Proceedings of the 16th International Conference on Extending Database Technology*. ACM, 2013, pp. 251–262.

[32] Z. Yan, J. Eberle, and K. Aberer, "Optimos: Optimal sensing for mobile sensors," in *Mobile Data Management (MDM), 2012 IEEE 13th International Conference on*. IEEE, 2012, pp. 105–114.

[33] R. Gandhi, S. Khuller, and A. Srinivasan, "Approximation algorithms for partial covering problems," *Journal of Algorithms*, vol. 53, no. 1, pp. 55–84, 2004.

[34] P. Slavík, "Improved performance of the greedy algorithm for partial cover," *Information Processing Letters*, vol. 64, no. 5, pp. 251–254, 1997.

[35] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation*, vol. 214, no. 1, pp. 108–132, 2009.

[36] D. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm-a novel tool for complex optimisation problems," in *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS 2006)*, 2006, pp. 454–459.

[37] B. Basturk and D. Karaboga, "An artificial bee colony (abc) algorithm for numeric function optimization," in *IEEE swarm intelligence symposium*, 2006, pp. 12–14.

[38] M. Li, N. Cao, S. Yu, and W. Lou, "Findu: Privacy-preserving personal profile matching in mobile social networks," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 2435–2443.

[39] G. Ghinita, P. Kalnis, M. Kantarcioglu, and E. Bertino, "Approximate and exact hybrid algorithms for private nearest-neighbor queries with database protection," *GeoInformatica*, vol. 15, no. 4, pp. 699–726, 2011.

[40] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in cryptologyEUROCRYPT99*. Springer, 1999, pp. 223–238.

[41] S. S. Dhillon and K. Chakrabarty, *Sensor placement for effective coverage and surveillance in distributed sensor networks*. IEEE, 2003, vol. 3.

[42] F. Y. Lin and P.-L. Chiu, "A near-optimal sensor placement algorithm to achieve complete coverage-discrimination in sensor networks," *Communications Letters, IEEE*, vol. 9, no. 1, pp. 43–45, 2005.

[43] K.-P. Shih, Y.-D. Chen, C.-W. Chiang, and B.-J. Liu, "A distributed active sensor selection scheme for wireless sensor networks," in *Proceedings. 11th IEEE Symposium on Computers and Communications. ISCC'06.* IEEE, 2006, pp. 923–928.

[44] M. G. Kallitsis, G. Michailidis, and M. Devetsikiotis, "Measurement-based optimal resource allocation for network services with pricing differentiation," *Performance Evaluation*, vol. 66, no. 9, pp. 505–523, 2009.

[45] M.-A. Koulali, A. Kobbane, M. El Koutbi, and J. Ben-Othman, "Optimal distributed relay selection for duty-cycling wireless sensor networks," in *Global Communications Conference (GLOBECOM), 2012 IEEE*. IEEE, 2012, pp. 145–150.

[46] V. K. Bagaria, A. Pananjady, and R. Vaze, "Optimally approximating the lifetime of wireless sensor networks," *arXiv preprint arXiv:1307.5230*, 2013.

[47] S. He, X. Gong, J. Zhang, J. Chen, and Y. Sun, "Barrier coverage in wireless sensor networks: From lined-based to curve-based deployment," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 470–474.

[48] Y. Chen, J.-A. Francisco, W. Trappe, and R. P. Martin, "A practical approach to landmark deployment for indoor localization," in *Sensor and Ad Hoc Communications and Networks. SECON'06.*, vol. 1, 2006, pp. 365–373.

[49] D. Yang, G. Xue, X. Fang, and J. Tang, "Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 173–184.

[50] A. C.-C. Yao, "Protocols for secure computations," in *FOCS*, vol. 82, 1982, pp. 160–164.

[51] M. Chase and S. S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in *Proceedings of the 16th ACM conference on Computer and communications security*. ACM, 2009, pp. 121–130.

[52] R. Zhang, R. Zhang, J. Sun, and U. Yan, "Fine-grained private matching for proximity-based mobile social networking," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 1969–1977.

[53] K. Liu, "Paillier cryptosystem." [Online]. Available: http://www.csee.umbc.edu/~kunliu1/research/Paillier.html

[54] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu, "Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services," in *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. IEEE, 2008, pp. 366–375.

[55] H.-Y. Lin and W.-G. Tzeng, "An efficient solution to the millionaires problem based on homomorphic encryption," in *Applied Cryptography and Network Security*. Springer, 2005, pp. 456–466.

[56] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *Privacy Enhancing Technologies*. Springer, 2009, pp. 235–253.

[57] I. Bilogrevic, M. Jadliwala, K. Kalkan, J.-P. Hubaux, and I. Aad, "Privacy in mobile computing for location-sharing-based services," in *Privacy Enhancing Technologies*. Springer, 2011, pp. 77–96.

[58] X. Yi, R. Paulet, E. Bertino, and V. Varadharajan, "Practical k nearest neighbor queries with location privacy," in *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*. IEEE, 2014, pp. 640–651.

[59] D. Recordon and D. Reed, "Openid 2.0: a platform for user-centric identity management," in *Proceedings of the second ACM workshop on Digital identity management*. ACM, 2006, pp. 11–16.

[60] "Central authentication service." [Online]. Available: http://en.wikipedia.org/wiki/Central_Authentication_Service

[61] W. Jie, J. Arshad, R. Sinnott, P. Townend, and Z. Lei, "A review of grid authentication and authorization technologies and support for federated access control," *ACM Computing Surveys (CSUR)*, vol. 43, no. 2, p. 12, 2011.

[62] "Google's crash took 40% of internet traffic down with it." [Online]. Available: http://www.businessinsider.com/google-goes-down-2013-8

[63] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," *Advances in Cryptology–EUROCRYPT 2011*, pp. 568–588, 2011.

[64] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in CryptologyCRYPTO 2001*. Springer, 2001, pp. 213–229.

[65] A. De Caro and V. Iovino, "jpbc: Java pairing based cryptography," in *the 2011 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2011, pp. 850–855.

[66] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Security and Privacy, 2007. SP'07. IEEE Symposium on*. IEEE, 2007, pp. 321–334.

[67] A. Lewko, A. Sahai, and B. Waters, "Revocation systems with very small private keys," in *Security and Privacy (SP), 2010 IEEE Symposium on*. IEEE, 2010, pp. 273–285.

[68] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography–PKC 2011*. Springer, 2011, pp. 53–70.

[69] A. Beimel, "Secure schemes for secret sharing and key distribution," Ph.D. dissertation, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[70] S. Müller, S. Katzenbeisser, and C. Eckert, "On multi-authority ciphertext-policy attribute-based encryption," *Bulletin of the Korean Mathematical Society*, vol. 46, no. 4, pp. 803–819, 2009.

[71] D. Huang *et al.*, "Mobile cloud computing," *IEEE COMSOC Multimedia Communications Technical Committee (MMTC) E-Letter*, vol. 6, no. 10, pp. 27–31, 2011.

[72] S.-W. Seong, J. Seo, M. Nasielski, D. Sengupta, S. Hangal, S. K. Teh, R. Chu, B. Dodson, and M. S. Lam, "Prpl: a decentralized social networking infrastructure," in *Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond*. ACM, 2010, p. 8.

[73] J. Li, "mssl: A framework for trusted and incentivized peer-to-peer data sharing between distrusted and selfish clients," *Peer-to-Peer Networking and Applications*, vol. 4, no. 4, pp. 325–345, 2011.

[74] E. Bertino, F. Paci, R. Ferrini, and N. Shang, "Privacy-preserving digital identity management for cloud computing," *Data Engineering*, vol. 32, no. 1, 2009.

[75] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "Gsis: a secure and privacy-preserving protocol for vehicular communications," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3442–3456, 2007.

[76] S. Chow, Y.-J. He, L. Hui, and S. Yiu, "Spice–simple privacy-preserving identity-management for cloud environment," in *Applied Cryptography and Network Security*. Springer, 2012, pp. 526–543.

[77] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in cryptology*. Springer, 1985, pp. 47–53.

[78] R. Sakai, M. Kasahara *et al.*, "Id based cryptosystems with pairing on elliptic curve," in *2003 Symposium on Cryptography and Information Security–SCIS*, vol. 2003, 2003.

[79] D. Boneh and X. Boyen, "Efficient selective-id secure identity-based encryption without random oracles," in *Advances in Cryptology-EUROCRYPT 2004*. Springer, 2004, pp. 223–238.

[80] A. Sahai and B. Waters, "Fuzzy identity-based encryption," *Advances in Cryptology–EUROCRYPT 2005*, pp. 557–557, 2005.

[81] S. Yu, K. Ren, and W. Lou, "Attribute-based on-demand multicast group setup with membership anonymity," *Computer Networks*, vol. 54, no. 3, pp. 377–386, 2010.

[82] M. Chase, "Multi-authority attribute based encryption," in *Theory of Cryptography*. Springer, 2007, pp. 515–534.

[83] S. Müller, S. Katzenbeisser, and C. Eckert, "Distributed attribute-based encryption," in *Information Security and Cryptology–ICISC 2008*. Springer, 2009, pp. 20–36.

[84] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 15th ACM conference on Computer and communications security*. ACM, 2008, pp. 417–426.

[85] B. Libert and D. Vergnaud, "Adaptive-id secure revocable identity-based encryption," in *Topics in Cryptology–CT-RSA 2009*. Springer, 2009, pp. 1–15.

[86] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, "Identity-based encryption with outsourced revocation in cloud computing," *Computers, IEEE Transactions on*, vol. 64, no. 2, pp. 425–437, 2015.

[87] J. Chen, H. W. Lim, S. Ling, H. Wang, and K. Nguyen, "Revocable identity-based encryption from lattices," in *Information Security and Privacy*. Springer, 2012, pp. 390–403.

[88] S. Jahid, P. Mittal, and N. Borisov, "Easier: Encryption-based access control in social networks with efficient revocation," in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*. ACM, 2011, pp. 411–415.