AN ABSTRACT OF THE THESIS OF

Mohamed Grissa for the degree of Master of Science in Electrical and Computer Engineering presented on May 22, 2015.

Title: Location Privacy Preservation for Optimal Sensing in Cognitive Radio Networks

Abstract approved: ______________________________________

Bechir Hamdaoui

Cognitive Radio Networks (CRNs) enable opportunistic access to the licensed channel resources by allowing unlicensed users to exploit vacant channel opportunities. One effective technique through which unlicensed users, often referred to as Secondary Users (SUs), acquire whether a channel is vacant is cooperative spectrum sensing. Despite its effectiveness in enabling CRN access, cooperative sensing suffers from location privacy threats, merely because the sensing reports that need to be exchanged among the SUs to perform the sensing task are highly correlated to the SUs' locations.

In this thesis, we propose three private sensing protocols. The first scheme, *Location Privacy for Optimal Sensing (*LPOS*)* preserves the location privacy of SUs while achieving optimal sensing performance through voting-based sensing. In addition, *LPOS* is the only alternative among existing CRN location privacy preserving schemes (to the best of our knowledge) that ensures high privacy, achieves fault tolerance, and is robust against

the highly dynamic and wireless nature of CRNs. We provide also a second variant of *LPOS*, that we call *REP-LPOS* which incorporates a reputation mechanism and uses Elliptic Curve El Gamal with Pollard lambda method to boost the decryption. The third scheme is called *Public Register Private Sensing (*PRPS*)* which is the most efficient scheme but offers lower privacy than *LPOS* and *REP-LPOS*.

Location Privacy Preservation for Optimal Sensing in Cognitive Radio Networks

by

Mohamed Grissa

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented May 22, 2015
Commencement June 2015

Master of Science thesis of Mohamed Grissa presented on May 22, 2015.

APPROVED:

Major Professor, representing Electrical and Computer Engineering

Director of the School of Electrical Engineering and Computer Science

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Mohamed Grissa, Author

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude and appreciation to my academic advisor Professor Bechir Hamdaoui, Associate Professor in EECS at OSU, for offering me the chance to join his group and work under his supervision. His expertise, suggestions, and continuous support, added considerably to my graduate experience and my ability to carry out scientific research at OSU.

Also I would like to thank Professor Attila Yavuz, Assistant Professor in EECS at OSU, for his collaboration in this project, his time, help, valuable suggestions and constructive discussions that we had and for sharing with me his knowledge and expertise throughout this project and the classes I took with him.

I would like to express my sincere gratitude also to my committee members Professor Maggie Niess, Professor Lizhong Chen and Professor J. Eduardo Cotilla-Sanchez, for accepting to serve as members of the committee of my defense and for the time they devoted to read this thesis.

I deeply thank my dear parents, Zakia Sta and Abderraouf Grissa, and would like to dedicate this thesis to them for their unlimited support, love and encouragements throughout all the endeavors that I have been through and to my brothers, Wassim, Belhassen, Ahmed and Abdallah to whom I wish happiness and success in their lives.

Finally and most importantly, I would like to thank my dear wife and best friend Lamia Ben Lagha for her patience, love and constant support during this very important step of our live.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

Chapter 1: Introduction

1.1 Problem Statement

Cognitive Radio Networks (CRNs) have emerged as a key technology for improving spectrum utilization efficiency by enabling opportunistic access to the wireless channel resources. They do so by allowing unlicensed spectrum users, often referred to as Secondary Users (SUs), to identify and exploit unused opportunities of licensed channels, so long as they do not cause any interference to licensed users, often referred to as Primary Users (PUs).

Two main approaches can be used by SUs to acquire whether PUs are present in a licensed channel [7]. The first approach is based on geo-location databases and is very similar to what is used in LBSs (location-based services). The second approach, referred to as cooperative spectrum sensing, relies on the SUs themselves to visit and sense the licensed channels, on a regular basis, to collaboratively decide whether a channel is vacant or not. In this work, we focus on the cooperative spectrum sensing approach whose general architecture is shown in Fig. 1.1. In this architecture, the Fusion Center (FC) is the entity responsible for orchestrating the SUs to perform the sensing task so as to collectively decide whether PUs are present or not. Through a control channel, FC queries SUs, each having sensing capability, to tune to specific channels/frequencies, measure the energy level (known as Received Signal Strength (RSS)) observed in each of these

Table 1.1: Privacy, dynamism handling, fault tolerance and sensing performance proposed and previous schemes

Evaluation		**Location Privacy**	**Dynamism**	**Fault Tolerance**	**Sensing Performance**
LPOS		Very High	Multiple	yes	optimal [29]
REP-LPOS		Very High	Multiple	yes	optimal
PRPS		High	Multiple	yes	optimal
Generic	*ECEG*	Low	Multiple	yes	not optimal
	PDAFT [13]	Low	Multiple	yes	not optimal
PPSS [19]		Medium	Single	No	not optimal

Privacy: If *FC* can learn the aggregated result we evaluate the privacy to be low since an estimation of sensing reports of some users is possible when there are users leaving/joining the network. *Medium* privacy if there is a mechanism to cope with the mentioned problem but still using aggregation. We qualify our scheme to have *High* privacy since it does not have this vulnerability. **Dynamism:** *Multiple* when the scheme can handle multiple users leaving/joining the network simultaneously and *Single* when only one SU joining/leaving the network is supported. **Fault Tolerance:** whether or not the system still works normally when one of the SUs fails to send its report. **Sensing Performance:** a scheme is optimal if its sensing performance is proven to be optimal otherwise it is not optimal

channels, and report the observed RSS values back to FC[1]. FC then first combines the RSS values collected from the different SUs and then compares the combined value against a detection threshold, τ, to decide whether a channel is available. Channel availability decisions are sent back to the SUs to rely on during their opportunistic spectrum access.

Despite its effectiveness in improving sensing performance, cooperative spectrum sensing suffers from many security and privacy threats that make SUs shy away from joining and participating in the cooperative sensing task. One of these threats is the disclosure of SUs' location information. Cooperative spectrum sensing exploits spatial

[1] Energy detection is the most popular method for signal detection due to its simplicity and small sensing time [15].

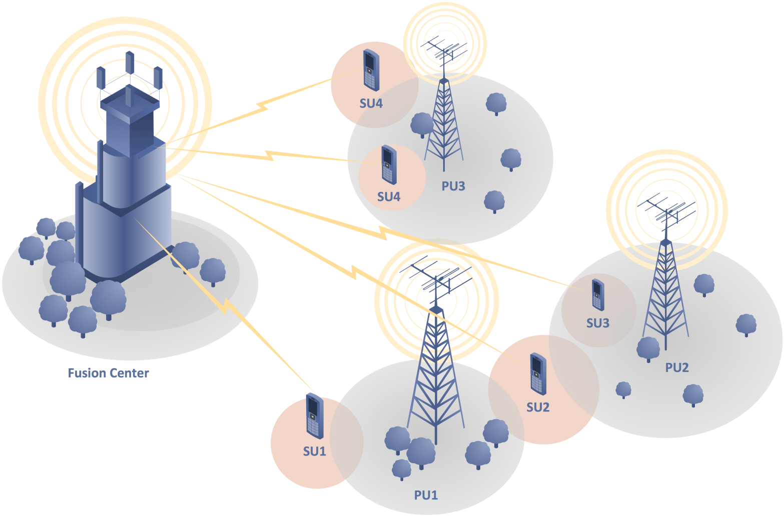


Figure 1.1: Cooperative spectrum sensing architecture

diversity for enhancing accuracy of sensing and this can jeopardize the location privacy of SUs. It has been shown in [19] that RSS values are heavily correlated to the SUs' physical locations, thus making it not too difficult to compromise the location privacy of SUs. Disclosing the location information is undesirable especially when FC is run by an untrusted service provider [9]. The fine-grained location data can be used to determine a lot of information about an individual's beliefs, preferences, and behavior [26]. In fact, by analyzing location traces of a user, an adversary can learn that he/she regularly goes to a hospital, and may then sell this information to pharmaceutical advertisers without the user's consent. In addition, malicious adversaries with criminal intent could use this information to pose a threat to an individual's security and privacy. Being aware of such potential privacy risks, SUs may not want to share their data with FC s or databases [26], making the need for preserving the location privacy of these users of a high importance.

In this work, we address the SUs' location disclosure threat, which is considered as one of the most important threats to cognitive radio users' privacy. We design three different protocols that guarantee a high privacy of the SUs' location by concealing the RSS values from FC while enabling optimal sensing using the half-voting rule proposed in [29].

1.2 Related Work

Shuai Li et al. [19] showed that location information of SUs could be inferred from the sensing reports, and called this attack Single CR Report Location Privacy (SRLP) attack. Another attack in the same context occurs when a user joins or leaves the network. Any malicious entity can estimate the report of a user and hence its location from the variations in the final aggregated RSS measurements when the node joins and leaves the network. This is termed Differential Location Privacy attack. To cope with these attacks, the authors propose *PPSS*, a Privacy Preserving collaborative Spectrum Sensing protocol, that uses secret sharing and the Privacy Preserving Aggregation (PPA) process to hide the content of specific sensing reports. It also uses dummy report injections to cope with the Differential Location Privacy attack. However, *PPSS* has several limitations. First, it requires all the sensing reports in order to decode the aggregated result, which makes it quite impractical since the wireless channel may be unreliable, making some sensing reports not accessible by FC . Hence, FC will not be able to decrypt the aggregated sensing result. Moreover, it cannot cope with the dynamics resulting when multiple users join or leave the network simultaneously. In addition, the pairwise secret sharing process incurs extra communication overhead, which results in an additional delay especially when all the keys need to be updated when a user joins or leaves the network. Also, the encryption scheme used here is practical only when the plaintext space is small, since the decryption of the aggregated result requires solving the DLP problem, which is very costly as shown in Table 5.1.

Some other approaches focused on solving the location privacy problem of SUs in completely different settings and scenarios. Goa et al. [16] provided a solution that guarantees users' location privacy in database-driven cognitive radio networks using a blinding factor to hide the location information when querying the database. In [21], Liu et al. proposed a location privacy preserving dynamic spectrum auction approach to protect SUs' location using prefix membership verification based range queries that protect the bid items and prices of the SUs from which they show that an attacker can determine the location information.

Despite the importance of this issue and the potential that CRNs present, little attention has been paid to this problem. This drove us to look outside the context of CRNs and try to find an approach that might be applied to our setup. We were particularly interested in the work proposed by Chen et al. [13] where they present a privacy-preserving data aggregation scheme with fault tolerance for smart grid communications, termed *PDAFT*. They considered a setting very similar to the one we study in this work, and tried to preserve users' privacy when smart meters installed within each house sense the consumption information and send it to the control center.

PDAFT combines Paillier cryptosystem with Shamir's secret sharing, where a set of smart meters sense the consumption of different households, encrypt their reports using Paillier, then send them to a gateway. The gateway multiplies these reports and forwards the result to the control center, which selects a number of servers (among all servers) to cooperate in order to decrypt the aggregated result. However, *PDAFT* requires a dedicated gateway to collect the encrypted data and a minimum number of working servers in the control center to be able to decrypt the aggregated result. In addition, *PDAFT*, like most of the aggregation-based methods, is prone to differential attacks that we mentioned earlier, and does not provide a mechanism that prevents this attack. Another drawback, which is common to simple aggregation-based methods, is that they usually do not provide optimal sensing performance and might be affected by the distribution of the RSS values. Throughout this thesis, by optimal sensing we mean final decision accuracy regrading the channel availability.

1.3 Our Contribution

We propose new location privacy-preserving sensing schemes for cognitive radio networks (CRNs) *LPOS*, *REP-LPOS* and *PRPS*. To the best of our knowledge, these are the first schemes that can preserve SUs' location privacy for CRNs but at the same time enable an optimal sensing performance using the half-voting rule. The main idea behind our schemes is to enable privacy-preserving comparison of RSS values and FC 's threshold in an efficient manner via a novel integration of Order Preserving Encryption (OPE) [11] and Yao's Millionaires' protocol [28]. We summarize the desirable properties of our different schemes below, and we further compare them to existing approaches with respect to different metrics as outlined in Table 1.1. We give detailed performance analysis and comparison with further discussions in Chapter 5.

Desirable Properties: Compared to their counterparts, *LPOS*, *REP-LPOS* and *PRPS* achieve the following desirable properties:

1) Location Privacy with Optimal Sensing Performance: To the best of our knowledge, the proposed schemes are the first schemes that enable location privacy for CRNs with an optimal spectrum sensing performance. That is, they permit privacy-preserving realization of the *half-voting* rule proposed in [29], which has been shown to be the optimal decision rule for spectrum sensing using energy detection. Unlike aggregation methods that may be vastly impacted by the distribution of RSS values (and misleading FC to inaccurate decisions), this rule enjoys an optimal sensing performance.

2) High Location Privacy of Secondary Users: Unlike some aggregation type protocols [13, 19], our schemes do not leak RSS information during users joining/leaving operations, nor do they require dummy report injection to prevent differential attacks as in [19].

3) Fault Tolerance: In our schemes, if some users cannot sense the channels or fail to send their reports, FC only needs to update the voting threshold, λ, with the available users to make an accurate decision. However, some existing schemes cannot handle such failures. For example, *PPSS* [19] requires inputs from all (pre-determined) users to be able to decrypt the aggregated RSS. Hence, if one of the encrypted reports is missing, FC will not be able to make a decision. The proposed schemes do not have such a limitation, since they rely on a voting-based approach and FC evaluates each contribution of users towards the decision individually, which makes our schemes more fault-tolerant compared to *PPSS* [19].

4) Scalability and Computational Efficiency: Our schemes offer the smallest communication overhead among their counterparts for large network sizes. The computational complexity is logarithmic in the number of users for both *LPOS* and *REP-LPOS* and is constant for *PRPS*, which makes them more practical and scalable (a detailed performance analysis is given in Chapter 5).

5) Handling Dynamism Effectively in the Network: When a group of users join or leave the network (a common scenario in CRNs), the system security and performance should be maintained. Unlike some alternatives (e.g., *PPSS* [19]) which can deal with the joining/leaving of only a single user at a time, the proposed schemes can effectively handle multiple, simultaneous join/leave operations.

The remainder of this thesis is organized as follows. Chapter 2 presents our preliminary concepts and definitions. Chapter 3 provides an extensive explanation of the different schemes that we propose in this thesis. Chapter 4 gives the security analysis of the different proposed protocols. Chapter 5 presents performance analysis of the proposed schemes and a comparison with existent approaches. Finally, Chapter 6 concludes this work.

Chapter 2: Preliminaries

CRN System and Sensing Model. We consider a centralized CRN that consists of a FC and n SUs, as shown in Figure 1.1. We assume that each SU is capable of assessing RSS values of channels through energy detection methods [15], and communicating them to FC, which it then combines them to make decisions regarding whether channels are available. FC then broadcasts the final decisions back to SUs.

Half-voting rule. Two reasons motivated our choice of a voting-based rule over an aggregation-based fusion rule: (i) it has a better sensing performance than aggregation-based rules [24], and (ii) it does not expose users to the privacy issues, we mentioned earlier, that would otherwise be exposed to when aggregation-based rules are used.

The authors in [29] derived a voting threshold, λ, for optimal spectrum sensing in voting-based CRNs, which is termed half-voting rule. With this, when the number of users whose RSS values are greater than τ is higher than λ, then FC can conclude that the channel is busy.

Beta Reputation Mechanism. To make the voting rule more reliable we incorporate a reputation mechanism which allows FC to progressively eliminate malicious users that try to falsify their reports along with faulty SUs by decreasing their contribution to the sensing at each iteration. Thus, whenever FC detects that one of the users is malicious or has erroneous reports, it will penalize it by updating a reputation score that reflects the level of reliability that the user has. In this work we use the robust Beta reputation system proposed by Arshad et al. [8]. After running YM with the users in the network, FC obtains a decision vector $\boldsymbol{b} = [b_1, \ldots, b_n]^T$ that contains the binary decision corresponding to each user. FC then combines the different binary decisions to make a global decision:

$$B = f(\boldsymbol{w}, \boldsymbol{b}) \tag{2.1}$$

where $\boldsymbol{w} = [w_1, \ldots, w_n]^T$ is the the weights vector calculated by FC based on the credibility score of each user and f is the fusion rule defined as

$$f(\boldsymbol{w}, \boldsymbol{b}) = \begin{cases} 1, & \text{if } \sum_{i=1}^{n} w_i \times b_i \geq \lambda \\ 0, & \text{otherwise} \end{cases} \tag{2.2}$$

where λ is the voting threshold determined by the Half-voting rule. FC categorizes the contribution of each SU U_i as positive or negative based on its observation by computing a positive rating and negative rating coefficients ζ_i and η_i that are updated every sensing period t_w as follows:

$$\zeta_i(t_w) = \zeta_i(t_w - 1) + \nu_1, \ \eta_i(t_w) = \eta_i(t_w - 1) + \nu_2 \tag{2.3}$$

where ζ_i reflects the number of times user U_i's observation b_i follows the global decision B made by the FC and η_i reflects the number of times user U_i's observation disagrees with the global decision. ν_1 and ν_2 are obtained as follows:

$$\nu_1 = \begin{cases} 1, & \text{if } b_i(t_w) = B(t_w) \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

$$\nu_2 = \begin{cases} 1, & \text{if } b_i(t_w) \neq B(t_w) \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

then the credibility score φ_i of SU U_i is given by

$$\varphi_i = \frac{\zeta_i + 1}{\zeta_i + \eta_i + 2} \tag{2.6}$$

Finally, based on the credibility score of user U_i, FC computes the weight w_i that will be given to the contribution of this user using Equation 2.7:

$$w_i = \frac{\varphi_i}{{}^{n}_{j=1}\,\varphi_j} \tag{2.7}$$

Notation. Operators $||$ and $|x|$ denote the concatenation and the bit length of variable x, respectively. $x \overset{\$}{\leftarrow} \mathcal{S}$ denotes that x is randomly and uniformly selected from the set $\mathcal{S}$. Large primes q and $p > q$ such that $q|(p-1)$, and a generator α of the subgroup G of order q in $\mathbb{Z}_p^*$ are selected such that Discrete Logarithm Problem (DLP) [22] is intractable. (sk, PK) denotes a private/public key pair of ElGamal Encryption [14], generated under (G, p, q, α). $c \leftarrow OPE.E_K(M)$ denotes order preserving encryption (as defined in Definition 1) of a message $M \in \{0,1\}^d$ under private key K, where integer d is the block size of OPE.

Cryptographic Building Blocks. Our scheme utilizes various cryptographic building blocks, which are described below:

• *Order Preserving Encryption (OPE):* Recall that the definition of OPE, introduced by Boldyreva et al. in [11], is:

Definition 1 *An OPE is a deterministic symmetric encryption scheme whose encryption operation preserves the numerical ordering of the plaintexts, i.e. for any two messages m_1 and m_2 $s.t.$ $m_1 \leq m_2$, we have $c_1 \leftarrow OPE.E_K(m_1) \leq c_2 \leftarrow OPE.E_K(m_2)$.*

The OPE concept was first introduced by Agrawal et. al [6] and then formalized by Boldyreva et. al [11]. Note that our scheme can use *any secure OPE* scheme (e.g., [18, 23]) as a building block, and receive the benefits of the security enhancement (e.g., [18]). However, we chose the publicly available implementation of Boldyreva's scheme [11] so as to evaluate our scheme in terms of execution time. In [11], an ideal security notion, called *indistinguishability under ordered chosen-plaintext attack (IND-OCPA)*, was introduced, which implies that OPE has no leakage, except the order of ciphertexts. However, Boldyreva et. al [12] showed that the ideal OPE security is unachievable, since it requires a ciphertext size that is at least exponential in the size of the plaintext, leading to the introduction and adoption of a weaker security notion of *Random Order-Preserving Functions (ROPF)*, as defined below.

Definition 2 *An OPE based on* ROPF *leaks the order of plaintexts and also at least half of the high-order bits of the plaintext [12, 23].*

• *Secure Comparison Protocol:* The Yao's Millionaires' (YM) protocol [28] enables two parties to execute "the greater-than" function, $GT(x, y) = [x > y]$, without disclosing any other information apart from the outcome of the comparison. In ***LPOS***, we used an efficient YM scheme [20], referred to as $YM.ElGamal$, which ensures that only the initiator learns the outcome. In this work we modify this protocol by using elliptic curve El Gamal instead of using the El Gamal scheme and use this modified version in ***REP-LPOS***. This has several benefits as will discussed later in this thesis. We refer to the original YM scheme as $YM.ElGamal$ and the modified scheme as $YM.ECElGamal$.

Definition 3 *Let* $(\mathcal{X}, \mathcal{Y})$ *and* $(x, y) \in \{0, 1\}^\gamma$ *be two parties and* γ*-bit integers to be compared, respectively. Let* $\pi = (\gamma, q, p, \alpha, \{sk, PK\})$ *be* $YM.ElGamal$ *parameters generated by the protocol initiator* $\mathcal{X}$*.* $YM.ElGamal$ *returns a bit* $b \leftarrow YM.ElGamal$ (x, y, π)*, where* $b = 0$ *if* $x < y$ *and* $b = 1$ *otherwise. Only* $\mathcal{X}$ *learns* b *but* $(\mathcal{X}, \mathcal{Y})$ *learn nothing else.* $YM.ECElGamal$ *is secure in the semi-honest setting if El Gamal encryption scheme [14] is secure.*

We give the description of the Yao's Millionaires' protocol as in [20]. This protocol was reduced by [20] to the set intersection problem and is based on the fact that x *is greater than* y *iff* S_x^1 *and* S_y^0 *have a common element* where S_x^1 and S_y^0 are the 1-encoding of x and the 0-encoding of y respectively. The 0-encoding of a binary string $s = s_\gamma s_{\gamma-1} \ldots s_1 \in \{0, 1\}^l$ is given by $S_s^0 = \{s_\gamma s_{\gamma-1} \ldots s_{i+1} 1 | s_i = 0, 1 \leq i \leq \gamma\}$ and the 1-encoding of s is given by $S_s^1 = \{s_\gamma s_{\gamma-1} \ldots s_i | s_i = 1, 1 \leq i \leq \gamma\}$. $\mathcal{X}$ with a private input $x = x_\gamma x_{\gamma-1} \ldots x_1$ generates π as in Definition 3 for encryption and decryption (E, D) then prepares a $2 \times \gamma$-table $T[i, j]$, $i \in 0, 1, 1 \leq j \leq \gamma$ such that $T[x_i, i] = E(1)$

and $T[\bar{x}_i, i] = E(r_i)$ for a random r_i in the subgroup G_q and finally sends T to $\mathcal{Y}$. $\mathcal{Y}$ with private input $y = y_\gamma y_{\gamma-1} \dots y_1$ computes c_t for each $t = t_l t_{\gamma-1} \dots t_i \in S_y^0$ as follows

$$c_t = T[t_\gamma, l] \times T[t_{\gamma-1}] \dots \times T[t_i, i] \tag{2.8}$$

then it prepares $l = \gamma - |S_y^0|$ random encryptions $z_j = (a_j, b_j) \in G_q^2, 1 \leq j \leq l$ and permutes c_t's and z_j's as $c_1, c_2, \dots, c_\gamma$ which are sent back to $\mathcal{X}$. $\mathcal{X}$ now decrypts $D(c_i) = m_i$, $1 \leq j \leq \gamma$ and decides $x > y$ *iff* some $m_i = 1$.

- *Group Key Establishment and Management:* We use a dynamic and contributory group key establishment and management protocol for secure group communication purposes.

Definition 4 Tree-based Group Elliptic Curve Diffie-Hellman (TG ECDH) *[27] permits n distinct users to collaboratively establish and update a common group key K by extending 2-party ECDH key exchange protocol to n-party.* TGECDH *is secure if Elliptic Curve Discrete Logarithm Problem (ECDLP) is intractable [27].*

Chapter 3: The Proposed Schemes

3.1 Location Privacy for Optimal Sensing *LPOS*

Voting-based spectrum sensing offers several advantages over its aggregation-based counterparts as discussed in Chapter 2. However, this approach requires comparing FC's threshold τ and the RSS value r_i of each user U_i, thereby forcing at least one of the parties to expose its information to the other. One solution is to use a secure comparison protocol, such as $YM.ElGamal$, between FC and each user $U_1, \ldots, U_n$ in the network, which permits FC to learn the total number of users above/below threshold τ (as discussed in Chapter 1.2) but nothing else. However, secure comparison protocols involve several costly public key crypto operations (e.g., modular exponentiation), and therefore $\mathcal{O}(n)$ invocations of such a protocol per sensing period incur prohibitive computational and communication overhead.

The key observation that led us to overcome this challenge is the following: If we enable FC to learn the relative order of RSS values but nothing else, then the number of $YM.ElGamal$ invocations can be reduced drastically. That is, *the knowledge of relative order permits FC to execute $YM.ElGamal$ protocol at worst-case $\mathcal{O}(log(n))$ by utilizing a binary-search type approach*, as opposed to running $YM.ElGamal$ with each user in total $\mathcal{O}(n)$ overhead.

This simple yet powerful observation enables us to develop *LPOS*, which achieves

Algorithm 1 *LPOS* Algorithm

```
**Initialization**: Executed once at the beginning of the protocol.
$FC$ sets its energy sensing and optimal voting thresholds $\tau$ and $\lambda$, respectively as in [29]. Bit-length $\gamma = |\tau| = |r_i|$ for $i = 1, \ldots, n$, where $r_i$ denotes RSS value of user $U_i$.
$FC$ generates $YM.ElGamal$ parameters $\pi$ (as defined in Definition 3) and pre-computes *El Gamal* encryption values in $\pi$ based on $\tau$ to accelerate $YM.ElGamal$ protocol. $FC$ also generates a random padding $D \overset{\$}{\leftarrow} \{0,1\}^{d-\gamma-1}$, where $d$ is the block size of $OPE$. $D$ is known to all users.
There are $n$ users $\{U_i\}_{i=1}^{n}$ in the system, whose RSS values are denoted as $r_i$ for $i = 1, \ldots, n$, respectively.
$\{U_i\}_{i=1}^{n}$ collaboratively establish a group key $K$ via *TGECDH* protocol (Definition 4). We denote this group of users as $\mathcal{G}$.
$FC$ establishes an authenticated secure channel $chn_i$ with each user $U_i$ for $i = 1, \ldots, n$.    ▷ (e.g., via SSL/TLS) .

**Private Sensing**: Executed every sensing period $t_w$
$U_i$ computes $c_i \leftarrow OPE.E_K(D||r_i)$ for $i = 1, \ldots, n$.
$U_i$ sends $c_i$ to $FC$ over $chn_i$ for $i = 1, \ldots, n$.
$FC$ sorts encrypted RSS values as $c_{min} \leq \ldots \leq c_{max}$ (by Definition 1).
$FC$ initiates $YM.ElGamal$ as $b \leftarrow YM.ElGamal$ $(r_{id_{max}}, \tau, \pi)$ with user $id_{max}$ having the maximum $c_{max}$.
**if** $b = 1$ **then**
    $decision \leftarrow$ Channel is free.
**else**
    $FC$ initiates $YM.ElGamal$ as $b \leftarrow YM.ElGamal(r_{id_{min}}, \tau, \pi)$ with user $id_{min}$ having the minimum $c_{min}$.
    **if** $b = 0$ **then**
        $decision \leftarrow$ Channel is busy.
    **else**
        $FC$ initiates $YM.ElGamal$ with a subset of users based on a binary search of $\tau$ on the remaining encrypted RSS values as described below. Let index $I$ be the index of user $c_I$, where $YM.ElGamal$ with binary search process is finalized (i.e., $r_{I-1} \leq \tau \leq r_I$).
        $FC$ counts the number of $U_i$s s.t. $\tau \leq r_i : z \leftarrow n - I$
        **if** $z \geq \lambda$ **then**
            $decision \leftarrow$ Channel is busy
        **else**
            $decision \leftarrow$ Channel is free
**return** $decision$
```

Algorithm 2 *LPOS* Algorithm - continued

```
    Update Private Sensing after Group Membership Changes:
If new user(s) join/leave $\mathcal{G}$ in $t_w$, the new set of users $\mathcal{G}'$ forms a new group key $K'$ by following the key update of TGECDH protocol. $FC$ may update threshold and $YM.ElGamal$ parameters as $\lambda$' and $\pi$', respectively, if required.
Follow the private sensing steps with new $(K', \lambda', \pi')$.
```

the above objective via an innovative integration of the OPE scheme, ***TGECDH*** and $YM.ElGamal$ protocols. *The crux of the idea is to make users OPE encrypt their RSS values under a group key K, which is derived via* TGECDH *at the beginning of the sensing period.* In this way, FC can learn the relative order of encrypted RSS values but nothing else (and users do not learn each others' RSS values, as they are sent to FC over a pairwise secure channel). FC then uses this knowledge to run $YM.ElGamal$ *protocol by utilizing a binary-search strategy*, which enables it to identify the total number of users above/below threshold τ (as defined by voting-based optimal sensing in [29]) with only $\mathcal{O}(log(n))$ complexity. This strategy makes ***LPOS*** the only alternative among its counterparts that can achieve CRN location privacy with an optimal spectrum sensing, fault-tolerance and network dynamism simultaneously (as discussed in Chapter 1.3 and Chapter 5).

We give the detailed description of ***LPOS*** in Algorithm 1, and further outline the high-level description of ***LPOS*** as below:

• *Initialization:* FC sets up spectrum sensing and crypto parameters for cryptographic building blocks. Users establish a group key K via ***TGECDH***, with which they will OPE encrypt their RSS values during the private sensing. FC also establishes a secure channel chn_i with each user U_i.

- *Private Sensing:* Each user U_i OPE encrypts its RSS value r_i with group key K and sends ciphertext c_i to FC over chn_i. This permits FC to sort ciphertexts as $c_{min} \leq \ldots \leq c_{max}$ without learning corresponding RSS values, and the secure channel chn_i protects the communication of U_i from other users (as each r_i is encrypted under the same K) as well as from outside attackers. FC then initiates $YM.ElGamal$ first with the user that has the highest RSS value r_{max}. If it is smaller than energy sensing threshold τ then the channel is free. Otherwise, FC initiates $YM.ElGamal$ with the user that has r_{min}. If it is bigger than τ then the channel is busy. Otherwise, to make the final decision based on the optimal sensing threshold λ, FC runs $YM.ElGamal$ according to the binary-search strategy as described in Steps 17-22, which guarantees the decision at the worst $\mathcal{O}(log(n))$ invocations.

- *Update Private Sensing after Group Membership Changes:* At the beginning of each sensing period t_w, according to the membership changes in the user group, a new group key may be formed via the update procedure of *TGECDH* efficiently. FC may also optionally update sensing parameters. The private sensing for the new sensing period then begins with new sensing parameters and group key K' and is executed as described above.

3.2 Reputation and Elliptic Curve based Location Privacy for Optimal Sensing *REP-LPOS*

This protocol is another variant of *LPOS* that incorporates a robust reputation mechanism and uses a modified version of the YM protocol based on an optimized implementation of the *Elliptic Curve El Gamal* scheme that uses *Pollard-Lambda* algorithm to solve the *ECDLP* problem for the reverse map during the decryption phase. We refer to the modified version of the YM protocol as $YM.ECElGamal$. In $YM.ECElGamal$, we modify the YM protocol described in Chapter 2 by replacing the multiplication operations $\times$ in Equation 2.8 by Elliptic Curve addition operations $\oplus$ to make the protocol work with additive homomorphic encryption and more specifically with *Elliptic Curve El Gamal*. Now the decision of $x > y$ is made *iff* some $m_i = 0$.

The *Pollard-Lambda* method is designed to solve the *ECDLP* problem for points that are known to lie in a small interval which is the case for the RSS values. This method, also known as kangaroo method, uses two random walks one performed by a tame kangaroo who jumps off into the wild, digs a hole and waits for the wild kangaroo to fall into it [10].

The reputation mechanism allows to minimize the contribution of potential malicious or faulty SUs to the global decision and thus makes our scheme more reliable in real situations. We use the reputation mechanism proposed in [8] that was proven to be robust in detecting malicious SUs that intentionally or unintentionally modify their measurements.

We provide C++ Implementations of the optimized *Elliptic Curve El Gamal* and the $YM.ECElGamal$ schemes which are available for public use and can be found in [1].

As shown in Chapter 5, this protocol enjoys a smaller communication overhead compared to the original *LPOS* and this is one benefit of using elliptic curve cryptography over usual public encryption schemes. In addition, *REP-LPOS* considerably reduces the computational overhead of SUs and puts most of the computation to FC. These benefits are very desirable especially when the SUs are battery constrained. In fact, reducing the computational and communication overhead of these users will drastically increase the batteries lifetime. These modifications have no impact on the security level of the original scheme. The different steps of *REP-LPOS* are depicted in Algorithm 3. For brevity, we omit the initialization and the update phases from the algorithm since they are identical to the ones in *LPOS* but with the use of $YM.ECElGamal$ instead of $YM.ElGamal$. The main difference with the *LPOS* algorithm resides in the use of $YM.ECElGamal$ and the reputation mechanism.

In Steps 7, 12 and 15 of Algorithm 3, FC constructs the vector of local decisions of SUs after running the private comparisons between τ and the RSS values. Based on the decision vector $\boldsymbol{b}$ and the weights vector $\boldsymbol{w}$ that was computed previously, FC computes the global decision B in Step 16 using Equations 2.1 and 2.2 and voting threshold λ. Then FC computes the credibility score and the weights that will be given to every user in the next sensing period. Initially, all the users are considered credible and the weight vector $\boldsymbol{w}$ will be constituted of ones and whenever a SU U_i has a decision $b_i = B$, it will see its assigned weight decreasing. In the other hand, the users that make the same decision as FC will be assigned the highest weight.

3.3 Public Register based Private Sensing *PRPS*

We propose another algorithm for private sensing in CRNs called *Public Register based Private Sensing* PRPS. This protocol presents a trade-off between privacy and computational and communication efficiency. In fact this algorithm requires much less computa-

Algorithm 3 *REP-LPOS* Algorithm

```
Private Sensing: Executed every sensing period $t_w$
$U_i$ computes $c_i \leftarrow OPE.E_K(D||r_i)$ for $i = 1, \dots, n$.
$U_i$ sends $c_i$ to $FC$ over $chn_i$ for $i = 1, \dots, n$.
$FC$ sorts encrypted RSS values as $c_{min} \leq \dots \leq c_{max}$ (by Definition 1).
$FC$ initiates $YM.ECElGamal$ as $b_{id_{max}} \leftarrow YM.ECElGamal\ (r_{id_{max}}, \tau, \pi)$ with user $id_{max}$ having the maximum $c_{max}$.
if $b_{id_{max}} = 0$ then
    $decision \leftarrow$ Channel is free.
    $b_i \leftarrow 0$ for $i = 1, \dots, n$.
else
    $FC$ initiates $YM.ECElGamal$ as $b_{id_{min}} \leftarrow YM.ECElGamal(r_{id_{min}}, \tau, \pi)$ with user $id_{min}$ having the minimum $c_{min}$.
    if $b_{id_{min}} = 1$ then
        $decision \leftarrow$ Channel is busy.
        $b_i \leftarrow 1$ for $i = 1, \dots, n$.
    else
        $FC$ initiates $YM.ECElGamal$ with a subset of users based on a binary search of $\tau$ on the remaining encrypted RSS values as described below. Let index $I$ be the index of user $c_I$, where $YM.ECElGamal$ with binary search process is finalized (i.e., $r_{I-1} \leq \tau \leq r_I$).
        $FC$ assigns $b_i \leftarrow 0$ for $i = 1, \dots, I-1$ and $b_j \leftarrow 1$ for $j = I, \dots, n$
        $FC$ computes the global decision $B \leftarrow f(\boldsymbol{w}, \boldsymbol{b})$ as in equations 2.1 and 2.2
        if $B = 1$ then
            $decision \leftarrow$ Channel is busy
        else
            $decision \leftarrow$ Channel is free
$FC$ updates the credibility score $\varphi_i$ and weight $w_i$ of user $U_i$ as in equations 2.6 and 2.7 for $i = 1, \dots, n$
return $decision$
```

tional and communication overhead than *LPOS* and *REP-LPOS* but this comes with the cost of loosing in terms of privacy compared to the previous schemes as will be shown later in this thesis. The different steps of *PRPS* are depicted in Algorithm 4. Just like *LPOS*, *PRPS* has three phases as described below:

- *Initialization:* Very similar to the initialization phase of *LPOS* but it further requires FC to construct a public register to obfuscate the value of the energy threshold τ in such way that one of the values in this register is very close to τ as shown in Step 3 of Algorithm 4. This register, which consists basically of a set of energy values, is sent to one or multiple users to be encrypted with OPE under the group key K.

- *Private Sensing:* Here again Each user U_i OPE encrypts its RSS value r_i with group key K and sends ciphertext c_i to FC over the secure channel chn_i. Now from the encrypted register ς that was obtained from Step 8, FC will only need ς_ℓ which is the OPE encryption of ρ_ℓ that satisfies $\mid \rho_\ell - \tau \mid < \varepsilon$. So when ε is small enough, comparing an RSS value to τ becomes equivalent to comparing it to ρ_ℓ with high probability. The FC counts the number of c_i's s.t. $c_i > \varsigma_\ell$ and then compares it to the optimal voting threshold λ to decide about the availability of the channel.

- *Update Private Sensing after Group Membership Changes:* At the beginning of each sensing period t_w and whenever there is a membership change in the network, a new group key may be formed and the voting threshold may be updated if required. Then the private sensing will be executed using the new parameters.

The parameter s controls the security level of *PRPS*. A small value of s can easily expose the threshold τ. In the other hand, a large value of s makes it very difficult to an attacker to guess the value of τ. The parameter ϵ determines the accuracy level of the scheme in making the decision.

Algorithm 4 *PRPS* Algorithm

```
**Initialization**: Executed once at the beginning of the protocol.
$FC$ sets its energy sensing and optimal voting thresholds $\tau$ and $\lambda$, respectively as in [29]. Bit-length $\gamma = |\tau| = |r_i|$ for $i = 1, \ldots, n$, where $r_i$ denotes RSS value of user $U_i$.
$FC$ generates a random padding $D \xleftarrow{\$} \{0,1\}^{d-\gamma-1}$, where $d$ is the block size of $OPE$. $D$ is known to all users.
$FC$ constructs a public register $\rho = [\rho_1 \ldots \rho_s]\ s.t.\ \exists\ \ell < s\ and \mid \rho_\ell - \tau \mid < \varepsilon$
There are $n$ users $\{U_i\}_{i=1}^{n}$ in the system, whose RSS values are denoted as $r_i$ for $i = 1, \ldots, n$, respectively.
$\{U_i\}_{i=1}^{n}$ collaboratively establish a group key $K$ via *TGECDH* protocol (Definition 4). We denote this group of users as $\mathcal{G}$.
$FC$ establishes an authenticated secure channel $chn_i$ with each user $U_i$ for $i = 1, \ldots, n$.    ▷ (e.g., via SSL/TLS).
$FC$ shares $\rho$ with a user $U_j$.
$U_j$ computes $\varsigma = [\varsigma_1 \ldots \varsigma_s] \leftarrow OPE.E_K(D||\rho) = [OPE.E_K(D||\rho_1) \ldots OPE.E_K(D||\rho_s)]$ and sends it to the $FC$.

**Private Sensing**: Executed every sensing period $t_w$
$U_i$ computes $c_i \leftarrow OPE.E_K(D||r_i)$ for $i = 1, \ldots, n$.
$U_i$ sends $c_i$ to $FC$ over $chn_i$ for $i = 1, \ldots, n$.
$FC$ sets $j \leftarrow 0$.
for $i \leftarrow 1$ to $n$ do
    if $c_i > \varsigma_\ell$ then
        $j++$                                  ▷ # of $RSSs > \tau$ with high probability
if $j \geq \lambda$ then
    $decision \leftarrow$ Channel is busy
else
    $decision \leftarrow$ Channel is free
      return $decision$

**Update Private Sensing after Group Membership Changes**:
If new user(s) join/leave $\mathcal{G}$ in $t_w$, the new set of users $\mathcal{G}'$ form a new group key $K'$ by following the key update of *TGECDH* protocol. $FC$ may update half-voting threshold as $\lambda$' if required.
Follow the private sensing steps with new $(K', \lambda')$.
```

Chapter 4: Security Analysis

Threat Model: Our threat model focuses on the *location privacy (i.e., RSS values) of SUs*. We consider *honest but curious (semi-honest)* setting for FC and SUs forming group $\mathcal{G}$(no party, including FC, maliciously modies the integrity of its input). That is, they execute the protocol honestly but will show interest in learning information about the other parties. That is, FC and other SUs in the group $\mathcal{G}$ may target the location information of a SU U_i. RSS value r_i of U_i reveals this location information and therefore should be protected. SUs also may target the threshold value τ of FC. However, we assume that FC does not collude with some SUs to localize the other SUs, nor do SUs collude with each others or expose the group key K to FC or external parties maliciously. Similarly, we assume that FC and SUs do not inject false τ or RSS values into spectrum sensing. Finally, an external attacker $\mathcal{A}$ may launch passive attacks against the output of cryptographic operations and active attacks including packet interception/modification to FC and SUs. We rely on traditional authenticated secure channel to prevent such an external attacker $\mathcal{A}$.

Security Objectives and Analysis: We give our security objectives and their security analysis as below.

Definition 5 *Under our threat model described above, our security objectives are: (i) RSS values r_i of each U_i remain confidential during all sensing periods. (ii) The sensing threshold τ of FC remains confidential for all sensing periods. (iii) A secure channel is* *maintained between each SU and FC. (iv) Objectives (i)-(iii) are maintained for every membership changes in $\mathcal{G}$.*

It is easy to show that *LPOS* is secure according to Definition 5, as long as its underlying cryptographic building blocks are secure.

Theorem 1 LPOS *achieves security objectives in Definition 5, as long as* TGECDH, *OPE and $YM.ElGamal$ are secure according to Definition 2, Definition 3 and Definition 4, respectively.*

In sensing period t_w, objectives (i)-(iv) in Definition 5 are achieved as follows:

Initialization: In Step 1-2, FC sets up system and security parameters such that $YM.ElGamal$ and OPE are secure. Padding D and proper block size of OPE ensures that the leftmost bit leakage from OPE as defined in Definition 2 does not leak RSS value during the private sensing. In Step 4, SUs establish a group K, which protects r_i values against FC via OPE encryption (as required by (i) in Definition 5). In Step 5, FC and each U_i establish a secure channel, which protects OPE encrypted r_i values c_i (under the same group key K) from other SUs and external attacker $\mathcal{A}$ (as required by (i) and (iii) in Definition 5).

Private Sensing: OPE encryptions in Step 6 ensure the confidentiality of r_i values against FC during the ciphertext sorting (c_1,...,c_n) in Step 8, as long as OPE is secure according to Definition 2 (with proper padding and OPE block size as set in the initialization phase). Step 7 ensures the confidentiality of r_i of U_i against other SUs as well as the protection of the communication against an external attacker $\mathcal{A}$ via the secure channel. Hence, objective (i) in Definition 5 is achieved during OPE phase of *LPOS*.

Step 9 - Step 22 execute $YM.ElGamal$, which leaks no information on τ to SUs and r_i's to FC as required. Hence, objectives (i)-(iii) in Definition 5 are achieved during the whole private sensing steps.

Update Private Sensing after Group Membership Changes: Step 23 ensures that a new group key K' (based on Definition 4) and parameters (λ', π') are generated according to the membership status of the new group $\mathcal{G}'$. Step 24 ensures the private sensing steps are executed using new (K', λ', π') for each new sensing period. Consequently, security objectives (i)-(iv) in Definition 5 are achieved for all sensing periods as required.

Corollary 1 REP-LPOS *achieves security objectives in Definition 5 and reputation mechanism in Chapter 2*

The security of *REP-LPOS* is identical to *LPOS* with the exception of using of a reputation mechanism and a modified version of YM protocol, that we call $YM.ECElGamal$, based on Elliptic Curve El Gamal encryption that uses Pollard-Lambda algorithm for fast decryption.

Chapter 5: Analysis and Comparison

To show that our schemes can be applied to practical situations, we consider the *IEEE 802.22 standard* for TV white space management for our performance analysis as defined in the IEEE standard document [17]. Our analysis and comparison focuses on two aspects: (i) The level of location privacy, the accuracy of decision for spectrum availability and reliability. (ii) Communication, computational overhead introduced by cryptographic methods.

The execution times of the different primitives and protocols were measured on a laptop running Ubuntu 14.10 with 8GB of RAM and a core M 1.3 GHz Intel processor, with cryptographic libraries MIRACL [4], Crypto++ [2] and *Louismullie*'s Ruby implementation of OPE [5].

Location Privacy, Sensing Accuracy and Reliability: As shown in Table 1.1, *LPOS* and *REP-LPOS* achieve the highest level of privacy and decision accuracy among their counterparts. That is, they are the only schemes that achieve very high location privacy while enabling an optimal spectrum sensing. Moreover, they provide fault tolerance and support for dynamism of multiple SUs in the network, which makes them reliable. In addition, they both achieve low communication, computation overhead as discussed below. *PRPS* shares the same properties with *LPOS* and *REP-LPOS* except for the privacy level which is lower for *PRPS*. Still, *PRPS* offers higher privacy than state of art protocols as explained in Chapter 4. In addition, as we show in the follow-

Table 5.1: Communication and computation overhead of proposed and existent schemes

Evaluation		**Communication**	**Computation**	
			FC	***SU***
LPOS		$2\gamma \cdot \lvert p\rvert \cdot (2 + log\ n) + n \cdot \epsilon_{OPE} + \lvert Q\rvert \cdot log\ n$	$\gamma/2 \cdot (2 + log\ n) \cdot \lvert p\rvert \cdot Mulp$	$(2\gamma \cdot \lvert p\rvert + 2\gamma) \cdot Mulp + OPE + 2\ log\ n \cdot PMulQ$
REP-LPOS		$2\gamma \cdot \lvert Q\rvert \cdot (2 + log\ n) + n \cdot \epsilon_{OPE} + \lvert Q\rvert \cdot log\ n$	$\gamma/2 \cdot (2 + log\ n) \cdot (PMulQ + PAddQ + \sqrt{\delta} \cdot Pol)$	$(4\gamma - 6) \cdot PAddQ + OPE + (2\ log\ n + 2) \cdot PMulQ$
PRPS		$(\gamma + \epsilon_{OPE}) \cdot s + \lvert Q\rvert \cdot log\ n$	$s \cdot cmp$	OPE
Generic	*ECEG*	$4\lvert Q\rvert \cdot n$	$PMulQ + PAddQ + \sqrt{n \cdot \delta} \cdot Pol$	$2PMulQ + PAddQ(*) \quad \Vert \quad (n-2) \cdot PAddQ(\dagger)$
	PDAFT	$2\lvert N\rvert \cdot (n + 1)$	$2ExpN^2 + InvN^2 + y \cdot MulN^2$	$2ExpN^2 + MulN^2$
PPSS		$\lvert p\rvert \cdot n$	$H + (n + 2) \cdot Mulp + (2^{\gamma-1} \cdot n + 2) \cdot Expp$	$H + 2Expp + Mulp$

(i) Variables: γ: size of the sensing reports, n: number of SUs, N: modulus in Paillier, p: modulus of El Gamal, H: cryptographic hash operation, K: secret group key of OPE. $Expu$ and $Mulu$ denote a modular exponentiation and a modular multiplication over modulus u respectively, where $u \in \{N, N^2, p\}$. $InvN^2$: modular inversion over N^2, $PMulQ$: point multiplication of order Q, $PAddQ$: point addition of order Q. y: number of servers needed for decryption in *PDAFT*. cmp is the cost of one comparison and s is the size of the register in *PRPS*. **(ii) Parameter size:** For a security parameter $\kappa = 80$, suggested parameter sizes by *NIST 2012* are given by : $|N| = 1024$, $|p| = 1024$, $|Q| = 192$ as indicated in [3]. **(iii) OPE:** the computational complexity of the OPE is given by $OPE = (log\ |\mathscr{C}| + 1) \cdot T_{HGD} + (log\ |\mathscr{P}| + 3) \cdot (5log\ |\mathscr{C}| + \theta' + 1)/128 \cdot T_{AES}$, where $\mathscr{P}$,$\mathscr{C}$ are plaintext and ciphertext spaces respectively and θ' is a constant. ϵ_{OPE} is the maximum ciphertext size that could be obtained under the OPE encryption. This value was determined experimentally based on the OPE implementation in [5] and we noticed that it doesn't exceed the 128bits block size of the underlying AES block cipher $\Rightarrow \epsilon_{OPE} = 128\ bits$. **(iv) ECEG:** The SUs use the FC's *ECEG* public key to encrypt their RSSs and then one node is picked to collect the ciphertexts and multiply them together including its own encrypted RSS and then send the result to the FC. The decryption of the aggregated message in *ECEG* is done by solving the constrained ECDLP problem on small plaintext space similarly to [19] via Pollard's Lambda algorithm, which requires $O(\sqrt{n \cdot \delta}) \cdot Pol$ computation and $O(log(n\delta))$ storage [22], where $\delta = a - b$ if $RSS \in [a, b]$ and Pol is the number of point operations in Pollard Lambda algorithm which varies depending on algorithm implementation used. **(v) YM.ElGamal:** The communication cost for one comparison is $4\gamma \cdot |p|$. The total computational cost of the scheme for one comparison is $5\gamma\ log\ p + 2n$. Since in our scenario the value of the energy threshold τ remains unchanged, we can encrypt it only once and offline so the encryption cost can be omitted and the new total computational cost would be $(3\gamma \cdot |p| + 2\gamma) \cdot Mulp$ for each comparison operation. **(vi) YM.ECElGamal:** The communication cost for one comparison is $4\gamma \cdot |Q|$. The total computational cost of the scheme for one comparison is .**(vii) TGECDH**: It permits the alteration of group membership (i.e., join/leave), on average $\mathcal{O}(log(n))$ communication and computation (i.e., ECC scalar multiplication) [25]. **(*)** is the cost for a normal SU in *ECEG* and **(†)** is the cost of the SU that plays the role of a gateway in *ECEG*.

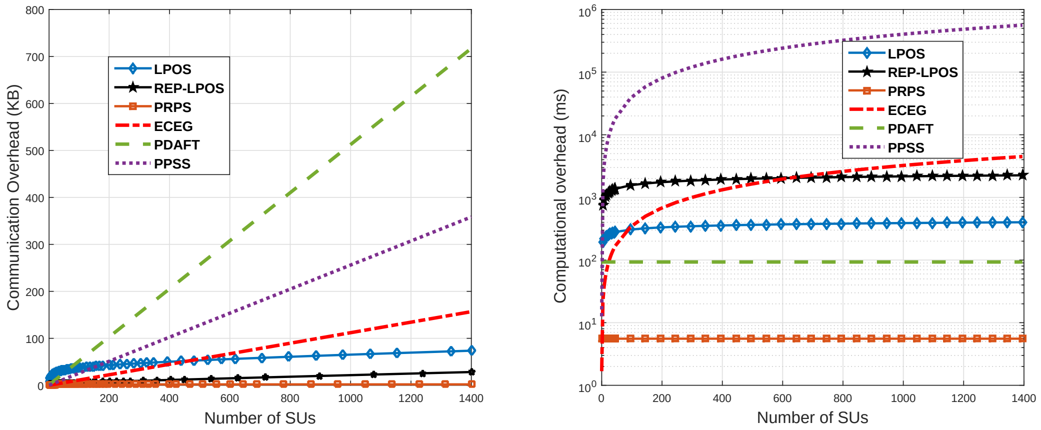
ing figures, it is the most efficient scheme in all aspects and its privacy level can be controlled by modifying the size s of the register ρ.

Communication and Computational Overhead: Our analytical comparison is summarized in Table 5.1, which also gives detailed explanations about variables, parameters sizes as well as overhead of building blocks and other schemes that are included in this comparison. The cost of ***LPOS*** is determined by $YM.ElGamal$, OPE, and TGECDH, whose costs are outlined in Table 5.1. Similarly the cost of ***REP-LPOS*** is determined by $YM.ECElGamal$, OPE, and TGECDH. Notice that the overall cost of ***LPOS*** and ***REP-LPOS*** is dominated by YM protocol, and yet $YM.ElGamal$ and $YM.ECElGamal$ are invoked only $\mathcal{O}(log(n))$ at the worst case (as explained earlier in Chapter 3 in detail). This permits high computational and communication efficiency for both protocols. Table 5.1 shows also that ***PRPS*** has the smallest overhead. In fact it has a constant end-to-end computational cost that does not depend on the number of users and only depends on the size s of the public register ρ. The communication overhead also depends only on s unless there is a need to update the group key which doesn't necessarily happen every sensing period. The cost of this update operation, performed using TGECDH, is given by the common logarithmic component in the three protocols as indicated in Table 5.1.

Figure 5.1(a)[1] compares the communication overhead of the different schemes for a security level $\kappa = 80$. As expected analytically, the figure shows how ***PRPS*** is the most efficient in terms of communication, followed by ***REP-LPOS*** and ***LPOS***. The gap between ***LPOS*** and ***REP-LPOS*** shows the impact of the modifications performed and the benefit of using ellitpic curve cryptography. This gap increases as we increase κ. They are followed by ***ECEG***, who has small key sizes for small number of users due to compact ECC parameters. ***PPSS*** has a high communication overhead, while ***PDAFT*** incurs extremely large communication overhead due to heavy Pailler encryption[2].

[1] Communication overhead of each scheme is calculated by evaluating its corresponding analytical results in Table 5.1 with the parameter sizes given in item (ii)-Table 5.1.

[2] *PDAFT* is adapted to CRN settings by dedicating one of the SUs to serve as a gateway, which collects encrypted RSSs and forwards their multiplication to the FC.



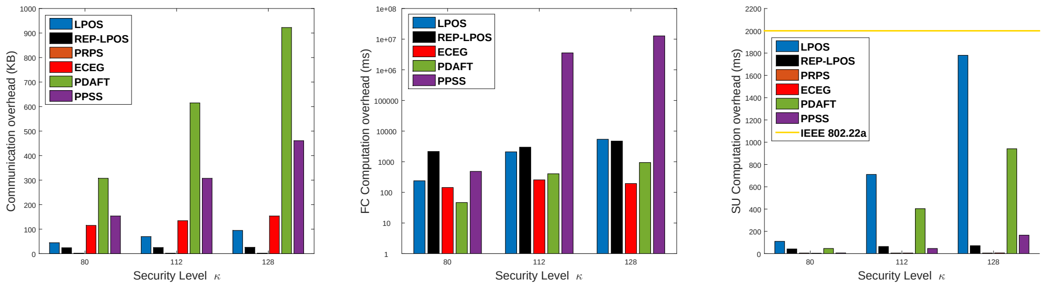
(a) Communication overhead

(b) Computation overhead

Figure 5.1: Performance Comparison

Then we compare the end-to-end computational overhead of the different schemes as shown in Figure 5.1(b). Here again *PRPS* has the smallest overhead since it requires only a small number of comparison operations and OPE encryptions and the cost doesn't depend on the number of SUs. Then comes *PDAFT* whose most of the computation is done by the dedicated gateway which explains its efficiency in terms of computation. *ECEG* is efficient only for very small network and its cost increases rapidly as the number of users increases. *LPOS* and *REP-LPOS* have similar logarithmic behaviors as the number of SUs increases, but *REP-LPOS* is clearly more expensive and this is due to the high cost needed to solve the *ECDLP*problem in the $YM.ECElGamal$. However most of this cost resides in FC side and the computational cost of SUs is much better in *REP-LPOS* than in *LPOS* as can be seen in Figure 5.2.

We also tried to modify the security level to see how the different schemes respond to the always increasing required size for the cryptographic keys. We show the vari-



(a) Communication overhead (b) FC Computation (c) SU Computation

Figure 5.2: Communication and Computational Overhead variation with respect to κ

ations incurred for three values of κ in Figure 5.2. Again our schemes perform much better than the other schemes in terms of communication overhead and the gap keeps increasing considerably with the increase in the security level as in Figure 5.2(a). *PRPS*, as mentioned previously, depends only on the parameter s and is not impacted by the change of κ. In terms of computation overhead required in FC side, *PRPS* turns out to be the best again and since it only requires very fast comparison operations, we were not able to measure the execution time. Thus, we omit it from Figure 5.2(b). This Figure shows that increasing security parameter κ doesn't incur much change on Elliptic Curve based protocols *REP-LPOS* and *ECEG* as opposed to the other schemes based on usual public encryption that has larger keys and that are more sensitive to the change of κ. The same observation is also valid and more obvious for Figure 5.2(c). Figure 5.2(c) shows how SUs computation was reduced considerably in *REP-LPOS* after the modifications that were performed over the original *LPOS*. However, *LPOS* requires less computation in FC side which could be seen as an advantage in systems where the QoS matters the most, meaning how fast FC can make the decision. *REP-LPOS*, in the other hand, could be seen as an excellent option for systems that have a powerful FC and battery constrained SUs. The different schemes, in particular ours, are below the requirement of $2s$ imposed by the IEEE 802.22a standard for the SUs time required to send their decisions.

Figure 5.3 shows the impact of the dynamism in the network on the different schemes. We start with 500 SUs in the network, the number of users that join and leave the network follow a uniform random distribution with values between 300 and 700 and with mean 500. we show in the Figure the standard deviation and the mean of the computational overhead over 1000 sensing rounds. It is clear from their curves heavy fluctuations that *PPSS* and *ECEG* are the most sensitive to the variation of the number of users in the network. This variation have a very little impact on our schemes.

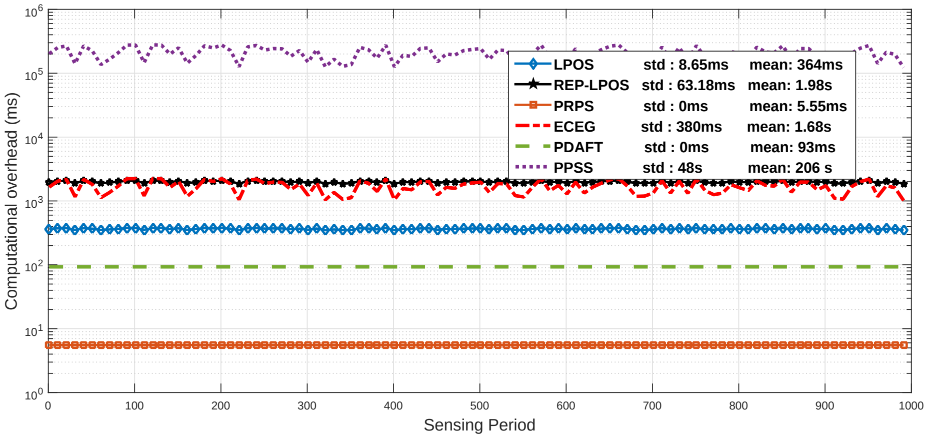


Figure 5.3: Dynamism in the network

We also try to measure the cumulative computational overhead that the system will experience after multiple sensing rounds as shown in Figure 5.4. After 1000 rounds, *PRPS* will require less than $600ms$ of system computation, *LPOS* will need less than

$40s$ and *REP-LPOS* less than $200s$. They are much more efficient to *PPSS* who will cost the system around 5 hours and 30 min of computation.

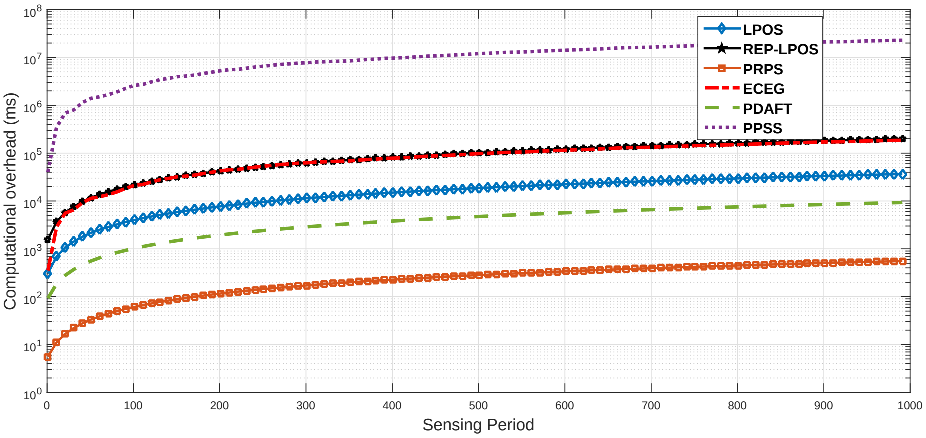


Figure 5.4: Cumulative overhead

Observe that while offering the smallest communication overhead (vital for scalability) and reasonable computational efficiency, our schemes are the only schemes that enable optimal spectrum sensing based on the half-voting voting approach and also provide the highest level of location privacy, fault-tolerance and network dynamism. All of these results show also how practical the different schemes that we propose are.

Chapter 6: Conclusion

In this work, we design three location privacy preserving protocols, called *LPOS*, *REP-LPOS* and *PRPS*. These protocols enable optimal sensing accuracy through the *half-voting* rule. We show that our schemes enjoy several desirable properties, making them more practical, secure, and reliable for small and large-scale CRNs. When compared to existing approaches, our schemes achieve optimal sensing performance with high user location privacy levels while being robust against user mobility and failures. *LPOS* through a small computational overhead required for FC offers better QoS to systems that require quicker decision about spectrum availability. *REP-LPOS*, via an efficient SU's computational overhead and low communication requirement, offers an excellent alternative to systems with battery constrained SUs. Both *LPOS* and *REP-LPOS* offer very high privacy. *PRPS*, which is extremely efficient in all aspects, is more suited to systems with very limited resources and that opt for lower location privacy level. The different schemes were shown to be practical under the IEEE 802.22a requirement [17]

Bibliography

[1] C++ implementations of elliptic curve el gamal and $YM.ECElGamal$. https://github.com/mohamedGr.

[2] Crypto++ library. http://www.cryptopp.com/.

[3] Cryptographic key length recommendation. http://www.keylength.com/en/compare/#Biblio6.

[4] Miracl library. http://www.certivox.com/miracl.

[5] Ruby ope implementation. https://github.com/louismullie/ope-rb.

[6] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Order preserving encryption for numeric data. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 563–574. ACM, 2004.

[7] Ian F. Akyildiz, Brandon F. Lo, and Ravikumar Balakrishnan. Cooperative spectrum sensing in cognitive radio networks: A survey. *Physical Communication*, 4:40–62, 2011.

[8] Kamran Arshad and Klaus Moessner. Robust collaborative spectrum sensing based on beta reputation system. In *Future Network & Mobile Summit (FutureNetw), 2011*, pages 1–8. IEEE, 2011.

[9] Shameek Bhattacharjee, Shamik Sengupta, and Mainak Chatterjee. Vulnerabilities in cognitive radio networks: A survey. *Computer Communications*, 36(13):1387–1398, 2013.

[10] Ian F Blake, Gadiel Seroussi, and Nigel Smart. *Elliptic curves in cryptography*, volume 265. Cambridge university press, 1999.

[11] Alexandra Boldyreva, Nathan Chenette, Younho Lee, and Adam O´ neill. Order-preserving symmetric encryption. In *Advances in Cryptology-EUROCRYPT 2009*, pages 224–241. Springer, 2009.

[12] Alexandra Boldyreva, Nathan Chenette, and Adam ONeill. Order-preserving encryption revisited: Improved security analysis and alternative solutions. In *Advances in Cryptology–CRYPTO 2011*, pages 578–595. Springer, 2011.

[13] Le Chen, Rongxing Lu, and Zhenfu Cao. PDAFT: A privacy-preserving data aggregation scheme with fault tolerance for smart grid communications. *Peer-to-Peer Networking and Applications*, pages 1–11, 2014.

[14] T. El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Proceedings of CRYPTO 84 on Advances in Cryptology*, pages 10–18. Springer-Verlag, 1985.

[15] Omid Fatemieh, Ali Farhadi, Ranveer Chandra, and Carl A Gunter. Using classification to protect the integrity of spectrum measurements in white space networks. In *NDSS*, 2011.

[16] Zhaoyu Gao, Haojin Zhu, Yao Liu, Muyuan Li, and Zhenfu Cao. Location privacy in database-driven cognitive radio networks: Attacks and countermeasures. In *INFOCOM, 2013 Proceedings IEEE*, pages 2751–2759. IEEE, 2013.

[17] IEEE Std 802.22a. IEEE Standard for Information Technology Telecommunications and information exchange between systems Wireless Regional Area Networks (WRAN) Specific requirements. Mar. 2014.

[18] Florian Kerschbaum and Axel Schroepfer. Optimal average-complexity ideal-security order-preserving encryption. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 275–286. ACM, 2014.

[19] Shuai Li, Haojin Zhu, Zhaoyu Gao, Xinping Guan, Kai Xing, and Xuemin Shen. Location privacy preservation in collaborative spectrum sensing. In *INFOCOM, 2012 Proceedings IEEE*, pages 729–737. IEEE, 2012.

[20] Hsiao-Ying Lin and Wen-Guey Tzeng. An efficient solution to the millionaires' problem based on homomorphic encryption. In *Applied Cryptography and Network Security*, pages 456–466. Springer, 2005.

[21] Sheng Liu, Haojin Zhu, Rong Du, Cailian Chen, and Xinping Guan. Location privacy preserving dynamic spectrum auction in cognitive radio network. In *Distributed Computing Systems (ICDCS), 2013 IEEE 33rd International Conference on*, pages 256–265. IEEE, 2013.

[22] A.J. Menezes, P. C. van Oorschot, and S.A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996. ISBN: 0-8493-8523-7.

[23] Raluca A Popa, Frank H Li, and Nickolai Zeldovich. An ideal-security protocol for order-preserving encoding. In *Security and Privacy (SP), IEEE Symposium on*, pages 463–477. IEEE, 2013.

[24] Junyang Shen, Tao Jiang, Siyang Liu, and Zhongshan Zhang. Maximum channel throughput via cooperative spectrum sensing in cognitive radio networks. *Wireless Communications, IEEE Transactions on*, 8(10):5166–5175, 2009.

[25] Michael Steiner, Gene Tsudik, and Michael Waidner. Diffie-hellman key distribution extended to group communication. In *Proceedings of the 3rd ACM conference on Computer and communications security*, pages 31–37. ACM, 1996.

[26] Wei Wang and Qian Zhang. *Location Privacy Preservation in Cognitive Radio Networks*. Springer, 2014.

[27] Yong Wang, Byrav Ramamurthy, and Xukai Zou. The performance of elliptic curve based group diffie-hellman protocols for secure group communication over ad hoc networks. In *Communications, 2006. ICC'06. IEEE International Conference on*, volume 5, pages 2243–2248. IEEE, 2006.

[28] Andrew C Yao. Protocols for secure computations. In*2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 160–164. IEEE, 1982.

[29] Wei Zhang, Ranjan K Mallik, and Khaled Letaief. Cooperative spectrum sensing optimization in cognitive radio networks. In *Communications, 2008. ICC'08. IEEE International Conference on*, pages 3411–3415. IEEE, 2008.