Summer 2013

Location based services in wireless ad hoc networks

Neelanjana Dutta

Department: Computer Science

LOCATION BASED SERVICES IN WIRELESS AD HOC NETWORKS

by

NEELANJANA DUTTA

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

2013

Approved
Dr. Sriram Chellappan, Advisor
Dr. Ali Hurson
Dr. Sanjay Madria
Dr. Bruce M. McMillin
Dr. Maciej Zawodniok

ABSTRACT

In this dissertation, we investigate location based services in wireless ad hoc networks from four different aspects - i) location privacy in wireless sensor networks (privacy), ii) end-to-end secure communication in randomly deployed wireless sensor networks (security), iii) quality versus latency trade-off in content retrieval under ad hoc node mobility (performance) and iv) location clustering based Sybil attack detection in vehicular ad hoc networks (trust).

The first contribution of this dissertation is in addressing location privacy in wireless sensor networks. We propose a non-cooperative sensor localization algorithm showing how an external entity can stealthily invade into the location privacy of sensors in a network. We then design a location privacy preserving tracking algorithm for defending against such adversarial localization attacks. Next we investigate secure end-to-end communication in randomly deployed wireless sensor networks. Here, due to lack of control on sensors' locations post deployment, pre-fixing pairwise keys between sensors is not feasible especially under larger scale random deployments. Towards this premise, we propose differentiated key pre-distribution for secure end-to-end secure communication, and show how it improves existing routing algorithms. Our next contribution is in addressing quality versus latency trade-off in content retrieval under ad hoc node mobility. We propose a two-tiered architecture for efficient content retrieval in such environment. Finally we investigate Sybil attack detection in vehicular ad hoc networks. A Sybil attacker can create and use multiple counterfeit identities risking trust of a vehicular ad hoc network, and then easily escape the location of the attack avoiding detection. We propose a location based clustering of nodes leveraging vehicle platoon dispersion for detection of Sybil attacks in vehicular ad hoc networks.

ACKNOWLEDGMENT

Firstly, I want to thank my advisor Dr. Sriram Chellappan for mentoring me during my PhD. He has been a constant source of support and encouragement, often going beyond an extra mile to guide me. The long hours of discussions with him and his thoughtful inputs have been invaluable in defining and improving my research.

I would like to thank my PhD committee members - Dr. Ali Hurson, Dr. Bruce McMillin, Dr. Sanjay Madria and Dr. Maciej Zawodniok, for their time and guidance. I extend my thanks to all the academic and non-academic staffs of Missouri S&T, specially to that of Computer Science Department, for their services to innumerable students like me. I want to thank all the great teachers who have shaped my personality and intellect, including the faculty at Institute of Engineering & Management and Missouri S&T. A special note of respect goes to my secondary school - Ramakrishna Sarada Mission Sister Nivedita Girls' School, and the wonderful community of teachers and mentors over there. My school has been my biggest driving force and inspiration in life. I would also like to take this opportunity to acknowledge the excellent minds that are spread across the world and are contributing towards my growth everyday through their inventions, knowledge, ideas and writings.

My stay in Rolla would have not been the same without all the amazing friends I have. Some remain friends for a lifetime, some don't. But thanks to all for being there at some point. Ravi, Anusha, Priya, Brijesh, Vimal, Ragha, Shreya, Deepak, Vijay, Suchi, Amlanima, Debisree and all other friends - I love you all!

I have been blessed with a beautiful family. I do not need words to express or acknowledge what they mean to me. I just want my parents, Didi, SaptarshiDa, Shlok, Riddhi, and Munima to know that I love them the most in this world. As a small token of my love, this dissertation is for you, Baba-Ma!

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

LIST OF ACRONYMS

AoA	Angle of Arrival
CBF	Chained Bloom Filter
DSRC	Dedicated Short Range Communication
FSTS	Fuzzy Short Time Series
GPSR	Greedy Perimeter Stateless Routing
ITS	Intelligent Transport System
LBS	Location Based Service
LPPT	Location Privacy Preserving Tracking
MANET	Mobile Ad hoc Network
NCLOCS	Non-cooperative Localization of Sensors
OBU	On Board Units
P2P	Peer-to-Peer
RSSI	Received Signal Strength Indicator
RSU	Road Side Unit
RMSE	Root Mean Squared Error
TDOA	Time Difference of Arrival
VANET	Vehicular Ad hoc Network
WSN	Wireless Sensor Network

1. INTRODUCTION

Wireless ad hoc network is a class of network systems primarily characterized by wireless communication and infrastructure-less operations in a distributed environment. Due to the minimal configuration and deployment requirements, wireless ad hoc networks can be formed and dispersed very quickly. Hence this class of networks are very useful for various applications in environments with minimum infrastructure. With the emerging advances in wireless communication technology and increasing emphasis on user centric services, wireless ad hoc networks are becoming more practical. Three canonical examples of wireless ad hoc networks that are increasingly emerging in military and civilian missions are *Wireless Sensor Networks*, *Mobile Ad hoc Networks* and *Vehicular Ad hoc Networks*. Brief descriptions of these networks and their applications are provided below.

- **Wireless Sensor Network:** Wireless Sensor Network (WSN) is a network consisting of wireless sensing devices (*sensors* or *sensor nodes*) with limited processing and computing capabilities. Sensor nodes can sense, measure and collect data about physical or environmental conditions, process the data and communicate using wireless messages [162]. Wireless sensors use communication standards like IEEE 802.15.4, ISA100, ZigBee, WirelessHART etc. for sending or receiving messages. Figure 1.1 shows a typical wireless sensor node (or mote).

  In spite of various resource constraints of sensor nodes, such as limited memory, limited computational power and energy, low bandwidth, short communication range, WSNs have gained significant importance due to their huge range of applications in real life. The main reasons behind the vast growth of WSNs are the infrastructure-less, autonomous mode of operation, scalability and ease of deployment of sensors. A few examples of applications of WSNs are: surveillance [70], target tracking [164], [115], environment monitoring [151], health monitoring [114], habitat monitoring [108], natural disaster (landslide, forest fire etc) detection and relief [9], [172], emergency response [106], seismic sensing [98], structural health assessment [107] etc.

In a centralized WSN, sensor nodes collect data from the surrounding environment and collaboratively forward the data to more powerful nodes, commonly known as *base stations*. Base stations process the data and send information to different applications and services which use this information for various purposes. A host of applications enable sensors to operate in completely distributed fashion wherein data processing, computation and other algorithms are executed collaboratively on the sensor nodes without interference from any central authority. In either case, nodes in a WSN usually collaborate among themselves allowing hassle-free operations such as data gathering, information fusion, encryption and authentication, single or multi-hop routing etc. Usually sensor nodes are small, inexpensive and resource-constrained, making them vulnerable to internal or environmental failures. A number of sensor network algorithms suggest reconfiguration of nodes for increasing robustness and sleep / wake up cycles of nodes for optimized resource utilization. This behavior in a WSN leads to change of network topology over time.

Wireless Sensor and Actor Networks (WSANs) are a type of wireless sensor networks in which sensors and actors collaborate to perform a task. Actors are usually more powerful entities than sensors, often with special characteristics like mobility, higher processing power etc. Usually sensors collect sensing data whereas actors take decisions and perform appropriate actions depending on the environment [77], [111]. Coordination between sensors and actors and timeliness of reporting sensed date impose some additional research challenges in WSANs, as discussed in [77].

Figure 1.1. Wireless Sensor Node

- **Mobile Ad hoc Network:** Mobile Ad hoc Network (MANET) is a collection of autonomous mobile devices that can communicate over relatively bandwidth-constrained wireless links and dynamically self-organize in an arbitrary and temporary fashion [3]. In other words, MANETs are a class of mobile P2P networks with ad hoc self-organization and dynamic network topology. MANETs facilitate users and devices to seamlessly interconnect in areas with no pre-existing communication infrastructure [33]. With increasing use of hand-held mobile devices, laptops, PDAs and embedded devices, MANETs are becoming increasingly popular and practical. Some of the canonical applications of MANETs include military communication and operations [72], [62], emergency services and information propagation [153], [52], commercial and infotainment applications [134] etc.

MANET research has received significant attention ever since it emerged in the 1990's as a part of new generation networking advances. The reasons behind it are two-fold : i) the increasing application of this type of network in real life and ii) the need of special standards and protocols in absence of pre-existing fixed infrastructure in this environment. Localization, single or multi-hop routing, data management, user location privacy, secure communication etc are instances of some of the major research issues in MANETs.

- **Vehicular Ad hoc Network:** Vehicular Ad hoc Network (VANET) is a type of mobile ad hoc network in which moving vehicles serve as nodes. Vehicles are equipped with On Board Units (OBU) such as sensors, transceivers and computing devices. These nodes are used for routing of data packets as they move from one place to another. Road Site Units (RSU) usually comprise of smart embedded devices including sensors, traffic controllers etc. RSUs and OBUs can communicate among themselves. Nodes in VANETs usually communicate using short range wireless communication technology, such as Dedicated Short Range Communication (DSRC), bluetooth, IEEE 802.11 etc. Figure 1.2 shows a DSRC device.

  As the potential of VANETs in improving urban transportation and lifestyle has unfolded, particularly since the last decade, a great amount of research effort has been dedicated to VANETs. Eventually VANETs are becoming the backbone of *Intelligent Transport System* (ITS) [160], [163]. The role of VANETs is integral specially in an urban environment, in applications such as emergency disaster response and recovery [138], electronic toll services [97], ubiquitous information and service sharing [100], [144], traffic congestion control [37], route planning [160] etc. VANETs are potent in making driving a safer, more comfortable and time-efficient experience to consumers (i.e. the drivers and passengers) through the pervasive flow of notifications, route information and services [18], [80]. As made possible by VANETs, today's drivers and passengers can avail almost any kind of web technology, services and benefits while driving.

Recently there have been several collaboration between government and industry for deploying VANETs in order to improve ITS infrastructure, services and safety [122], [148]. A number of VANET test-beds are also deployed in different academic research labs. A few such initiatives are CarTel at MIT [75], [116], DRIVE-IN at CMU [128], DOME at UMASS-Amherst [150], CVET at UCLA [90] etc. Emergency notification, vehicle tracking, secure communication, information retrieval, multi-hop routing, location privacy etc are some of the issues gaining focus in contemporary research in VANETs.

Figure 1.2. DSRC transceiver

Location Based Service: The definition of Location Based Service (LBS) has evolved over time with the progress in sensing and mobile technology. In context of computer programming, LBS is defined as a general class of computer program-level services that include specific controls for location and time data [154]. As cellular technology was prospering in 1990's, the notion of location based services or applications started being used in context of cellular devices and networks. As discussed in [60], LBS entails the concept of subscribing and switching of services and service providers based on location of the mobile user. The network here represented new generation hierarchical wireless networks, particularly cellular networks. However even in 90's, some futuristic services extended scope of application in other next generation networks as well. For instance, location information enhanced emergency service architecture is proposed in realm of vehicular networks and personal communication networks [59]. A novel location based alarm detection and security service are proposed in [131].

With the emergence of pervasive computing, smart mobile devices have become an integral part of everyday life. Advances in mobile and sensor based technologies coupled with popularity of pervasive and ubiquitous computing have extended the scope of location based services to a host of new networks and systems. Particularly in the realm of wireless ad hoc networks, where location of nodes is a vital factor for many applications, use of location based services is on a rise. A host of new generation LBSs are available in different networks such as WSN, MANET and VANET. Routing, content management, data replication, context-aware service sharing and recommendation, security and privacy services, [170], [50], [49], [24], [76], [67] etc. are some instances of contemporary LBSs in wireless ad hoc networks.

1.1. MOTIVATION

As the impact and importance of wireless ad hoc networks grow extensively, more research challenges are arising in this arena. With computing evolving to be ubiquitous, contextual information is a fundamental input to many applications in the field of wireless ad hoc networks [147], [2], [87]. As mentioned earlier in this section, location information, hence, is a very critical issue to be studied in wireless ad hoc networks. Many of the application and operation of wireless ad hoc networks exploit location information. We now discuss the specific importance of location information in the different types of networks mentioned earlier in this section.

- **Location based services in WSNs:**

  WSNs typically operate in large scale and are deployed randomly (often dropped from air). To perform collaborative operations, sensor nodes in the network need to estimate the location of themselves as well as other sensors. For example, if a fire is detected by sensors in a forest, the location of the fire can be estimated if the locations of reporting sensors are known. An intuitive approach for localization of sensor nodes can be to have a GPS device mounted on each node. Considering that sensors are supposed to be cheap, this solution is a very expensive one and is not scalable too. Localization techniques using other network resources and properties, such as beacon messages, probabilistic methods, etc. have been proposed in literature. We discuss localization in WSNs in further detail in Section 2.2.1.

  From the purview of security and privacy, location information of sensors is a critical factor to many security threats and privacy breaches. For example, localization of sensors by an unauthorized entity in a battlefield can aid enemies to destroy the network completely or partially. It might also help them in deriving knowledge about the network, such as network topology, coverage holes, location of base station etc. and penetrate in a nation's defense network. To be precise, the network functionality can be compromised when location information of sensors is exposed. Preserving location information of sensors from adversarial agents is hence a very critical requirement in WSNs.

- **Location based services in MANETs:**

  With rapid increases in mobile networking today, mobile ad hoc networks have gained increased prominence. With mobility, information propagation is much faster compared to static environments, there are increased avenues for access to information in mobile spaces, newer and more robust connections can be discovered on the fly for improved services. Interestingly though, mobility brings in challenges along with opportunities. For instance, when connections are ephemeral, long-term trust associations cannot be enforced, routing paths can be interrupted under mobility, and establishing secure connections are increasingly challenging. At the heart of these features of MANETs, is location information whose dynamics create these opportunities and challenges. There are currently a number of research directions being pursued in the realm of routing [50], [117], content management and information retrieval [170], [49], [40], data replication and caching [31], security [169], [6] etc. in MANETs, where location plays a vital role.

- **Location Based Services in VANETs:**

  Given the highly mobile nature of nodes in VANETs, location based services receive significant importance in VANETs. We illustrate this premise using a few cases below.

  a) Routing in VANETs is mostly location based. Majority of the routing protocols in VANET use location information as one of the key inputs, as discussed in the surveys in [96], [100].

b) Information in VANETs is highly context-sensitive [125], [10], [36]. For example, the popularity relevance of a piece of content or relevance of traffic information vary with the location of the node which contains it.

c) Location privacy is one of the most emergent research topics in VANETs as providing location privacy aware services is considered to be very crucial [139], [140]. In the highly dynamic environment of a VANET, trust of users and providers vary significantly over time. So it is evident that a user is more likely to be concerned about his / her privacy in such an environment. Although location information is a valuable input in many applications, many users would agree to trade quality of service against location privacy.

The trends in recent technological progress and user demand clearly indicate that location based services are growing in prolific fashion [12]. This dissertation investigates several location based services in wireless ad hoc networks, as discussed in Section 1.2.

1.2. CONTRIBUTIONS

The contributions of this dissertation is four-fold.

- **Location Privacy in WSNs:**

  The dissertation first investigates location privacy in WSNs. We define a representative adversarial localization attack on WSN wherein a mobile adversary (typically a robot) surreptitiously moves in the network while simultaneously capturing sensor communication signals. The adversary can detect wireless signals and measure their physical properties like Angle of Arrival (AoA) and Receive Signal Strength Indicator (RSSI) and localize sensor positions using existing techniques in [27], [161] and [152]. The dissertation formalizes this attack model using theoretical analysis and simulations. The goal of the sensor network is to localize the adversary, while simultaneously preserving its location privacy from the adversary. We propose a light-weight distributed protocol for preserving sensor location privacy against adversarial localization. The main challenge comes from the sensors performing two conflicting objectives simultaneously: localize the adversary, and hide from the adversary. This dissertation investigates the defense approach in elaborate details under the adversarial localization attack.

- **Secure End-to-End Communication in Randomly Deployed WSNs:**

  We address the issue of providing end-to-end secure communications in randomly deployed wireless sensor networks. In order to address the location disparity issue between sensors and sinks stemming from random deployment of nodes, we propose differentiated key pre-distribution. The idea behind this approach is to distribute different number of keys to different sensors to enhance the resilience of certain links in the network. This feature is leveraged during routing, where nodes route through links with higher resilience. We present our end-to-end secure communication protocol based on the above methodology by extending well known location centric (GPSR) and data centric (minimum hop) routing protocols.

- **Quality versus Latency Trade-off in Content Retrieval under Ad hoc Node Mobility:**

  The next contribution of the dissertation is in the realm of query-driven content retrieval in MANETs where the main challenge is to optimize search latency while maintaining quality of response under ad hoc mobility of nodes. When a user submit a query to a mobile node, the local node may not have the most relevant content to the query. Searching the peer nodes is likely to retrieve more relevant content, but at the cost of search overhead as delay. There is a clear trade-off between search latency and quality of responses that become significant in MANETs.

In many applications, for mobile ad hoc networks, the content that is most accurately matching to the query, is not the only one requested. In many cases, users are willing to compromise with accuracy of response in the interests of retrieving contents in a more timely manner, even if they are only less accurate ones (but preferably more in number). Through our research in this topic, we have come up with a novel two-tiered architecture to address this trade-off. The first tier of our architecture attempts to retrieve a better matching content by searching peer nodes, whereas the second tier returns the user with reasonably relevant but popular contents with very less delay.

- **Location Clustering based Sybil Attack Detection in VANETs:**

  The final contribution of this dissertation is in proposing a location clustering based scheme for Sybil attack detection, in the realm of VANETs. We propose a fuzzy time-series clustering based algorithm for location clustering of mobile nodes for Sybil attack detection in VANETs. The proposed method identifies the number of Sybil nodes and their identities with a very low false positive and false negative rate under different attack intensities. We take into consideration the aberration in localizing moving vehicles in a practical scenario and use extensive preprocessing and feature extraction methods for improved the accuracy of detection.

1.3. ORGANIZATION

The rest of this dissertation is organized in the following way. We present research on location privacy in WSNs, including detailed study on both attack and defense model, in Section 2. Section 3 includes a scheme for end-to-end secure communication in WSNs. In Section 4, we present our work on the multi-tiered architecture for content retrieval in MANETs. In Section 5, we design and analyze a time-series clustering based method for Sybil attack detection in VANETs. We summarize all the contributions of this research in Section 6 and conclude this dissertation in Section 7 with summary of findings from this research and final remarks.

2. LOCATION PRIVACY IN WIRELESS SENSOR NETWORKS

In this section of the dissertation, we investigate a privacy based service, namely, providing location privacy in wireless sensor networks. We study this problem from perspectives of - i) the attacker that invades into location privacy of sensors and ii) the sensor network that defends against such an attack. We first define a practical wireless sensor network problem wherein an adversary that is not cooperating with the wireless sensor network attempts to surreptitiously discover locations of sensors in the network. The adversary (or localizer) leverages from analyzing raw wireless signals emanated by the sensors. Our objective in this section is to formally define and analyze this attack model and subsequently preserve location privacy of the sensor nodes under such attack. Although localization in wireless sensor networks is a widely researched topic, not many work address localization in scenarios where the nodes do not cooperate with the entity attempting to localize sensors. In this dissertation, we first propose a new method for localization of sensors in a non-cooperative environment by a mobile localizer, wherein the localizer receives no cooperation from the sensor nodes that constitute the sensor network. The localizer localizes the sensors using physical properties of the sensor communication messages: Angle of Arrival (AoA) and Received Signal Strength Indicator (RSSI). Using the proposed method, the localizer can determine the presence of sensor node at a certain location with some error margin. This work shows how an external entity can invade in the location privacy of sensors in a network without being localized by the sensors. We call this kind of attack as adversarial localization. In other words, adversarial localization refers to passive attacks where an adversary attempts to disclose physical locations of sensors in the network by physically moving in the network while eavesdropping on communication messages exchanged by sensors. Our next contribution towards location privacy in wireless sensor networks is in designing a novel solution for defending against adversarial localization using a location privacy preserving tracking algorithm. The principle of the proposed approach is to allow sensors intelligently predict their own importance in light of two conflicting goals they have - preserving location privacy and tracking the adversary. The proposed algorithms ensures high degree of adversary localization, while also protecting location privacy of many sensors. Theoretical analysis and extensive simulations are conducted to demonstrate the performance of both the attack and defense models.

2.1. THE ATTACK MODEL

Localization of sensors in a Wireless Sensor Network (WSN) has been an important topic of research over the past few years. Most of these works consider cooperation from the sensor nodes to estimate their locations themselves or using assistance from external agents. However, in real life scenarios, cooperation from the sensor network might not be available to the external agent that is trying to localize sensors in the network. A canonical example of such a situation is sensor localization in unfriendly or battlefield environments where the objective of the external agent is to stealthily localize nodes in a network belonging to an enemy agent. However, non-cooperative localization of sensors has not been researched much, which is the focus of this research. We formally define the problem and propose a novel method for localization of sensors in a WSN by a mobile localizer without any cooperation from the sensor nodes. The proposed approach, NCLOCS (Non-cooperative Localization of Sensors), employs a mobile agent (called the *localizer*) that moves passively in the sensor network and captures sensor communication messages. The *localizer* has no knowledge about the content of these messages which are encrypted using secure keys. However, the *localizer* can measure some physical properties of the communication signal, such as Angle of Arrival (AOA) and Received Signal Strength

Indicator (RSSI). Using the proposed NCLOCS method, the *localizer* can determine the presence of sensor node at a location with some error margin. The salient features of NCLOCS are that it can efficiently associate communication signals with sensors without any prior knowledge, and filters many likely false sensor locations over time. Using theoretical analysis and extensive simulations, we demonstrate the performance of NCLOCS from the perspective of localization accuracy and detection time. NCLOCS (Non-cooperative Localization of Sensors): In this section of the dissertation, we first discuss the proposed attack model for invading location privacy of sensors in a network. The work has been submitted to a journal, as mentioned in [41].

2.1.1. Background And Related Work. Over past few years, Wireless Sensor Networks (WSNs) have become a critical component of a host of services and applications. Some such applications include area surveillance, wildlife monitoring, pervasive health monitoring, driving alert generation, seismic monitoring etc. In specific, wireless sensors have been of great importance in security applications leading to the extensive usage of WSNs in military environment. As a result, numerous testbeds have been designed and practically deployed in military settings. In this section, we address the issue of an external mobile agent attempting to localize wireless sensors without cooperation from the sensors themselves, an issue which has not been addressed much.

To understand the importance of non-cooperative localization in WSNs, it is necessary to first understand the threat imposed by maliciously operated WSNs. In specific, these threats are relevant and critical for military applications. In many missions of late, military personnel are being routinely employed in enemy battlefields with minimal prior knowledge of threats imposed in such fields. Traditional threats included landmines, IEDs, sniper fires etc. However, with the advances in sensor network technologies, coupled with their wide dissemination and acceptance, it is quite reasonable to envisage a WSN employed as a threat against military personnel in terms of monitoring their movements, triggering explosives, notifying enemy agents etc. Another representative threat occurs when enemy agents seize control over critical infrastructures in war zones like oil-fields, airports, power plants etc. and deploy a sensor network to guard such infrastructures. How to defeat such types of maliciously deployed WSNs is our motivation behind this work.

Towards this end, having the location information of the sensors in the unfriendly WSN can be useful in the aforementioned battlefield scenarios. A host of advantages are present when locations of sensors in an unfriendly network are known. For instance, number of nodes in the network can be estimated using location information of sensors helping in the measurement adversary's strength. The topology of the network can be estimated which can assist in coordinated and maximal impact counter-measures against the network. Location information of sensors will useful in derivation of further crucial information like coverage holes in the network, identification of the most important nodes with maximum connectivity, determination of optimal intrusion paths involving minimal detection through the network etc. Also, the sensors can be physically destroyed or deactivated to defeat the purpose of the enemy WSN. In fact, the localizer's side can selectively deactivate or launch cyber attacks at some sensors at crucial locations to cripple the functioning of the entire network. Therefore, clearly the location information of enemy sensors can provide one with significant amount of advantage in battlefields and security applications.

We point out that there is more than one approach to defeat a maliciously operated WSN, such as, destroying the nodes physically over a larger scale; launching packet drop or falsification attacks using active agents to subvert the functionality of the network, etc. Such approaches have several shortcomings. Large scale physical attacks can be cost-prohibitive and also may cause irreparable damages to the deployment field which we may need to protect (e.g., oil-fields, airports, power-plants etc.). Active agents interrupting network performance can be detected using advanced security algorithms [136]. On the other hand, a passive countermeasure is to listen to the message content of the sensors and leverage it to design subsequent defense strategies. However, it may be likely that the inimical sensors encrypt their messages. Discovering keys and encryption protocols must entail breaking into and capturing sensors which again violates the stealth requirement. The objective of this work is to design a mechanism that can cause a high degree of destructive potential to maliciously operated sensor networks, while still maintaining a sufficient degree of stealth during execution.

Location information of sensor nodes being a critical input to many of the existing WSN applications, localization of sensors has been widely researched [4]. However, most of these techniques involve co-operation from the sensor nodes in order to enable estimation of their locations by nodes in the network or outside agents, whereas localization of sensors in unfriendly and non-cooperative environment has not investigated often. Towards this premise, we design a technique to localize sensor nodes without any cooperation from the sensor network. We propose to employ a physically mobile agent called the *localizer* (typically a mobile robot), which will stealthily move in the network listening for sensor-to-sensor communication signals. It can be noted that different mobility platforms are available now enabling agents to move within a field [26]. Any such agent can be used as the mobile *localizer* which will attempt to measure the Angle of Arrival (AoA) and the Received Signal Strength Indicator (RSSI) of the wireless sensor messages it can receive. However, the *localizer* does not have any information on the message content or the id of the sensor sending the message (potentially due to message encryption). Using this information, we design a method for the *localizer* to estimate sensor locations in the network. At the initial stages of the protocol execution, location estimates are derived. Since the *localizer* does not know which sensor is sending which signal, there will be many false estimates during localization. We then incorporate a novel location scoring mechanism with a corresponding score translation mechanism, such that with the reception of more and more sensor signals, the protocol will filter out many false positives and gradually converge to real sensor locations. Integrating location information with the existing defense mechanisms can minimize the affect of enemy network and expedite the recovery from the attack, thereby enhancing security and survivability in battlefield applications to a great extant. We conduct a detailed theoretical analysis and extensive numerical simulations to demonstrate the performance of our protocol. Our analysis demonstrates that the localization protocol can effectively determine adversarial sensor locations with reasonable small false positives. Furthermore, we demonstrate that when the *localizer* has additional information on network behavior like transmission ranges and communication model, the localization accuracy improves.

It can be noted that, mobile agents in sensor networks can be mobility-enabled robots or other sensor based actuators. A survey of mobility in WSNs can be found in [26].

Related Work: Localizing sensor nodes in unfriendly environment falls under the purview of security in WSNs. There is a host of existing literature which deal with different security related problems in WSNs. Therefore, a pertinent point to study here is the difference between existing works in sensor networks security from our work. Existing work on WSN security by far and by large consider the sensor network to be benign whereas role of the adversary is to disrupt the network operations. Recently there is an increasing research interest which use mobile equipment to assist node getting their positions. Although these researches only consider the positioning in a cooperative situation, they shed some light on how to solve our problem. Typically, the standard attack model used in existing WSN security works is where the adversary captures a small percentage of network nodes that then behave maliciously. How to harness the potential of a relatively large number of benign sensor nodes to defeat the malicious operations of a few compromised sensors is the major theme in existing WSN security research. Instances of works in the framework include secure key management [54, 64, 102, 21, 74], location verification [48, 7, 141, 105, 47], secure localization [165, 93, 92, 91, 166], secure routing [83], maintaining integrity of sensor identities [124] etc.

Localization in WSNs has been an eminent topic of research. Existing literature usually propose collaborative localization of sensor nodes wherein the nodes cooperate with the *localizer* in the process of localization. Although existing researches mostly consider the positioning in a cooperative situation, they shed some light on how to solve our problem. In [20, 121, 69] sensors utilize their connectivity information to a small number of static beacons for localization. The localization algorithm includes determining centroids of triangles formed with beacon nodes, determining orientations of nodes with respect to beacons, triangle overlaps formed between beacons and regular nodes. In [142], distance measurement between sensors and static beacons are utilized for location estimation. In the Cricket indoor location support system [129], the range estimation between sensors and beacons is done using time difference of arrival between RF and ultrasound signals. Typically, the range is determined as a function of the time difference of arrival between an RF signal and an ultrasound signal, since RF signals travel much faster than ultrasound signals. A similar method is used for range determination in [142], although the authors focus on ad hoc deployments in an out door environment in [142].

In [121], the authors addressed the problem of how nodes in a connected ad hoc network collaboratively find their headings and positions under the assumption that all nodes have the AoA capability with some precision but only some nodes know their positions. An ad hoc positioning system (APS) with some error control mechanism was proposed. In this method, nodes adjacent to some landmarks get their bearings directly from the landmarks, and this information is propagated to their neighbors. Nodes not immediately adjacent to landmarks use that information to infer their own bearings with respect to the landmarks. This process is continued in a hop by hop fashion. When enough bearings with respect to the non-collinear landmarks are collected, a node can estimate its position. The position calculation is done at the node side even when moving landmarks are present. This approach needs higher degree requirements (≥ 9) than complete connectivity (6) and many landmarks (35%) to achieve high coverage. The precision which is on the order of one radio hop is very low for an attacking purpose.

While the scheme in [121] is based on AoA, in [32] some representative range based localization techniques in sensor network were reviewed and evaluated. The results show that these kinds of techniques also require high node density (degree of ten) with a moderate beacon fraction (20%) to achieve acceptable coverage and precision. Analysis and simulation showed that higher performance can be obtained by using both range and bearing information, even with imprecise bearing. It is known that received signal strength indicator (RSSI) is a nonlinear function of the range given the transmit power, so study the relationship between the distance probability distribution function with respect to RSSI can help measure the distance using RSSI. In [145], a mobile beacon aware of its position is broadcasting continuously its instant coordinates while moving around in the sensor network. A node measures RSSI when it receives a beacon signal. Each RSSI and the associated known coordinates form a constraint on node position. After multiple measurements are taken, the Bayesian inference is applied to compute the node's position either at a base station or on the sensor node itself. Higher precision than the RSSI based multi-lateration approach is reported. However, this approach needs careful system calibration to build the distance probability distribution function at different RSSI, which is not an easy task. It fails when transit power varies, which is not rare in sensor networks.

Mobility assisted WSN localization is a topic that has received attention recently. In the probabilistic approach presented in [145], a mobile beacon aware of its position continuously broadcasts its coordinates. A sensor measures RSSI of the signal when it receives one from the mobile and estimates its likely range using a Bayesian inference method [130] proposed a method where the mobile measures distances between sensor pairs until these distance constraints form a globally rigid structure that guarantees a unique localization. These pair-wise distances were then used to get the coordinates of the nodes.

Different from all the other approaches in which the computation of the nodes' positions is done either on the sensor nodes or in a base station, authors in [126] use one mobile robot aware of its position to perform location estimation based on the RSSI measurements of the radio message from the nodes. Their contribution is the use of a robust extended Kalman filter-based state estimator to solve the localization. A small scale experiment (4 indoor nodes) showed it can achieve one meter accuracy. Although the convergence speed is not quick enough, (since it took about 5 minutes (150 steps) to achieve a relatively precise estimation, while in this period of time some nodes may be inactive,) this is still a promising approach to our problem since a sensor node does nothing special except normal communication. It is well known that how to choose the weighting matrices in Kalman filter is not trivial because it needs some prior knowledge of the measurement noise. This approach assumes that the robot knows which RSSI measurement comes from which transmitter when multiple nodes can be sensed, which may not be true in a non-cooperative situation.

Differences between the above works and ours: In our work, the sensor network under consideration belongs to the unfriendly entities. In other words, the localized sensor nodes are inimical nodes where no co-operation can be received from the sensors during localization. Our problem is to defeat the operations of an *entire network* of sensors and not just a few sensors in the network. Furthermore, we will have virtually no information on any aspect of the network or its operation characteristics. Consequently, approaches that leverage knowledge of the network behavior, and/ or the presence of a large number of benign entities cannot be leveraged as a defense mechanism. An added challenge is the requirement of stealth in the defense mechanism which makes our problem quite harder from existing problems in WSN security.

However, Yang et. al. in [161] have studied a similar problem where the goal is to localize sensors in a network deployed by an adversary. In their solution, a set of monitors are deployed at the boundaries of the network to receive sensor signals and localize sensors. Deploying such monitors can be an expensive operation. Furthermore, it is assumed that *all* monitors can listen to *all* the communication signals of *all* sensors which is impractical for large area networks. Furthermore, in [161] it is assumed that the monitors are aware of the initial transmission power of all sensors in the network. This information is possible to obtain only if the monitors have insider information on the sensor network (obtained possibly by breaking into sensor nodes) which violates the stealth concept that we believe is critical. In this work, we design a new approach for localizing maliciously deployed sensors using a physical mobile agent moving in the network, and collecting sensor signals. Our approach does not need any expensive equipment, global network view, or *insider* information on the sensors present in the network.

2.1.2. Problem Definition. The system model in this problem comprise of two entities: the sensor network and the *localizer*. In this section, we present the models of both entities from the perspective of their features and capabilities.

Sensor Network Model: In this work, we consider a static sensor network that has been randomly (not necessarily uniformly) deployed with N sensors. For simplicity, we assume that the network is square in shape with dimensions $L \times L$. In real life, the area of interest is not likely to be of perfect square shape. But for the ease of computation, even area of irregular shape can be enclosed within an imaginary square. As we assume that the *localizer* is aware of the boundary of the area of interest, it can only visit the area of overlap between the imaginary square and the actual area of interest. In real life, it is also possible that the exact network area covered by the enemy sensors is unknown to the *localizer*. However we propose that the *localizer* can initially enclose the most critical area for localization and later increase the size of surveyed area by visiting squares adjacent to the initial area. The sensors in the network communicate with each other using encrypted communication messages. Each sensor is assumed to transmit with the same initial transmit power, P_{tx}. Note that P_{tx} is unknown to the mobile *localizer*.

The traffic model of the sensors depends on the network application and the behavior of sensed events. The data reporting process in WSNs is usually classified into three categories: event-driven, time-driven and query-driven [8]. In the time-driven case, sensors send their data periodically to the sink. Event-driven networks are used when it is desired to inform the data sink about the occurrence of an event. In query-driven networks, sink node sends a request for gathering data when needed. Our main focus will be on the event-driven networks with Poisson model for packet generation. We suppose that the events are independent (both temporally and spatially) and occur with equal probability over the area. In this case, Poisson distribution can be used effectively to model the generation of data packets [133].

When the average rate of packet generation, λ, is known, the distribution of the number of data packets, Z, generated by each node, from time 0 to T is,

$$P\left(Z=z\right)=\frac{e^{-\lambda T}\left(\lambda T\right)^{z}}{z!} \tag{1}$$

where z is a nonnegative integer. In the case of the packet generation distribution obeying the Poisson model, the time duration between two consecutive packet transmissions, t, has an exponential distribution with mean $\frac{1}{\lambda}$:

$$f_{t}\left(x\right)=\lambda e^{-x\lambda}u\left(x\right) \tag{2}$$

where $u(x)$ denotes the unit step function. We will consider a Poisson sensors traffic model in this study.

Localizer Model: The *localizer* in our problem is a mobile agent that can physically move from one location to another. For practical purposes a miniature robot serves this purpose. The *localizer* is equipped with the capability to measure angle of arrival (AoA) and received signal strength (RSSI) of a source signal. Note that AoA measurements typically require either an antenna array, or several ultrasound receivers. This is currently available in small formats in wireless nodes such as the one developed by the Cricket Compass project [129] from MIT. We assume that the *localizer* can detect any signal it receives provided the received power level is $\geq \bar{P}_{rx}$, which is the *localizer*'s receiver threshold. The *localizer* is aware of the network boundary within which it wishes to localize sensors. We assume that the sensors deployed are not equipped to track mobile intruders. In this context, it can be noted that certain existing works address the issue of sensors tracking the *localizer* to protect their own location privacy and report to the base station about the *localizer* [45].

However, these protocols are mostly valid for a particular type of sensors, for example vibration sensors. Besides, in Section 2.1.7 we show that by random movement policy, the *localizer* maximizes the probability of localization over time.

2.1.3. Proposed Solution.

In this Section, we present our localization protocol. The protocol is executed in three phases: The estimation phase, measurement phase and the localization phase. Each phase is discussed in detail below. For reader's convenience, important notations and their terminologies are presented in Table 2.1.

Table 2.1. Important Notations and Terminologies

Term	Description
θ	Sensor angle of arrival measured by localizer in *degrees*
ϵ_{AoA}	Error bound in angle of arrival measured by localizer in *degrees*
ϵ_{RSSI}	Error bound in RSSI measured by localizer in *metres*
$L \times L$	Total network area in m^2
$M \times M$	The area to be localized m^2
$g \times g$	Total number of grids in the network
$d = \frac{M}{g}$	Grid size in m
N	Number of Sensors
T_x	Actual transmission range of sensor
$\bar{T}_x$	Estimated transmission range of sensor
λ	Packet inter arrival rate
ρ	Density of sensors

Initiation phase: Without loss of generality, we assume that the *localizer* has to localize within a square area of size $M \times M$. Note that when $M = L$, the area to be localized is the entire sensor network deployment area. The *localizer* initially divides the area of interest into a 2$-$D rectangular grid ($g \times g$) where each grid is a square of dimension of size $d = \frac{M}{g}$. The objective of the *localizer* is to eventually determine those grids that contain atleast one sensor in them. The size of the grid is an application parameter and is variable. For high accuracy of localization, d can be set quite small, while for lower accuracies, d can be set correspondingly larger. We want to point out that in most of the applications, locating sensors to a grid level instead of exact point-level is sufficient. Usually, point level localization involves more number of measurements and incurs some error as well. We relax the point level localization requirement, thereby expediting the overall localization process. In security sensitive situations as the one considered in this work, longer time to localize a node increases the probability of detection and risk of attack for the *localizer*. Hence in this case, trading localization accuracy with delay in localization is sensible.

Observation time: The *localizer* traverses the entire network area, stopping at each intersection of vertical and horizontal grid lines. The time spent at each observation point (T_{obs}) is chosen such that the *localizer* can observe at least one transmission from each node in the neighborhood. Thus, T_{obs} depends on the underlying traffic pattern. For instance, for a Poisson model with rate λ, a $T_{obs} = \frac{10}{\lambda}$ would ensure that the *localizer* is able to observe messages from a particular sensor in the neighborhood with a probability of atleast 0.99995 (from equation 1). Waiting for longer durations improves this probability further. For scenarios where λ is not known apriori, we outline a simple method for obtaining a rough estimate. The *localizer* at various random locations observes the packet intervals from multiple sensors. This average packet interval multiplied by the average number of neighbors (which is again an estimate at different locations using the AoA) gives an approximate value for λ. We again note that waiting for longer durations only improves the accuracy of localization; thus, overestimating inter-packet arrival time is more helpful than harmful and an accurate estimate is not required.

Note that it is not compulsory that the sensor traffic model follows Poisson distribution. In scenarios where the traffic model is not Poisson, similar to the above approach, the *localizer* can estimate mean (μ_T) and standard deviation (σ_T) of inter-packet arrival times. We can then use Chebyshev's inequality, which states that "in any data sample or probability distribution, no more than $\frac{1}{k^2}$ of the values are more than k standard deviations away from the mean" [85]. Thus, the *localizer* waits for $T_{obs} = \mu_T + k \times \sigma_T$ (where $k = 6$), ensuring it observes a message from a sensor in the neighborhood atleast 97.2% of the time, where μ_T and σ_T are the means and standard deviations of the distribution respectively.

Transmission power and Range of the sensors: Recall that the *localizer* is unaware of the initial transmission power (P_{Tx}) of the sensors. To estimate P_{tx}, the following approach is used. The *localizer* as usual traverses the entire network, listens to various messages and collects information regarding angle of arrival and received power P_{Rx}. We note that *closeness* here is relative to the sensor's *transmission range* T_x, which is again an unknown. First, when a message is received, the probability that the source is within 5% of T_x (again, T_x is unknown) can be computed as,

$$P_{5\%} = \frac{\pi \left(0.05T_x\right)^2}{\pi {T_x}^2}. \tag{3}$$

Thus, the probability that the source is not within 5% of T_x is $\bar{P}_{5\%} = 1 - P_{5\%}$. Further, if the *localizer* receives m messages, then the probability that atleast one message was transmitted by a node within $0.05T_x$ can be computed as $1 - \bar{P}^m_{5\%}$. For example, if 1000 messages were observed during the entire process of localization, then the probability that atleast one message was transmitted by a node within $0.05T_x$ is 0.92. For a network of sufficient scale, these many number of messages is actually quite reasonable for the *localizer* to have listened to during the entire process of localization. Finally, based on the above reasoning, we use the maximum receiving power observed over all the messages as approximate receiving power at a distance of $5\%T_x$, based on which the transmission range T_x can be estimated.

Once, the transmission power is estimated, the upper bound of the transmission range ($\bar{T_x}$) of the sensors can be estimated. We note that, in practice, wireless transmissions are not circular and for several uncontrollable reasons, the attenuation cannot be accurately estimated [171]. However, from our protocol's perspective, we are only interested in the upper bound. Again, if the attenuation is varying drastically, the upper bound might not be tight; however, this will only cause a slight drop in protocol performance. Further, to improve the performance, we propose to consider only the messages received with a signal strength above a given threshold. We elaborate on this later in this section.

2.1.4. Localization Phase Using Only AoA. Our localization protocol is comprised of two phases: A grid score assignment phase and a score translation phase, as described in Algorithm 1.

Algorithm 1 Grid Score Assignment Algorithm Executed by the localizer

```
for each grid $G_{i,j}$ in the network do
    $Grid\_Score\ P_{i,j} = 0$
end for
for each grid intersection point in the network do
    Listen to messages for a duration of $T_{obs}$
    for each received message $k$ do
        Measure AoA/ RSSI of $k$
        Determine $Localized\_Zone$ of $k$
        Area of $Localized\_Zone = Z_k$
        for each grid $(i, j)$ overlapped with $Localized\_Zone$ do
            $A_{i,j,k}$ = Area overlapped between grid $G_{i,j}$ and $Z_k$
            $P_{i,j} = 1 - \prod_{\forall k}(1 - p_{i,j,k})$
        end for
    end for
end for
Return $Grid\_Score\ P_{i,j}$ for all grids
```

First, the *localizer* starts by traversing the entire grid, stopping at each intersection of vertical and horizontal lines. At each stop, the *localizer* listens for messages. We propose the *localizer* to listen for a duration of $10/\lambda$ to ensure that atleast one transmission from each sensor in the neighborhood is observed with high probability, where λ is the packet arrival rate assuming Poisson distribution.

For each message received, the *localizer* stores the following information: the angle of arrival, received signal strength (RSSI) and *localizer*'s location. Based on AoA information (i.e., θ) and the estimated transmission range ($\bar{T}_x$, as estimated earlier in the section), the *localizer* computes the sector that would enclose the transmitting node. We also refer to this sector as the *zone* that would enclose the transmitting node. The *localizer* assigns score to each grid within the zone based on area of overlap between the zone and grid.

2.1.5. Improving Accuracy by Using RSSI. Using only AoA information might generate several false positives. Moreover, if the *localizer* spends more time in vicinity of a certain point, the grids which are far away from that point will accumulate more score irrespective of position of the transmitting node. For instance, consider a scenario where the transmitting sensor is very close to the *localizer*. In this case, the *localizer* wrongly assigns higher probability to farther grid cells (since, farther the grid, the wider is the sector/zone, and hence larger the area of overlap). To reduce the number of false positives, we propose to use RSSI information. We note that RSSI might vary significantly even for messages from same node and hence, the distance estimates using RSSI might also vary significantly. So, instead of using RSSI directly, we propose to use RSSI only to filter some messages rather than for computing distances. In other words, the *localizer* would consider a message for score computation, only if the RSSI for the message is greater than a threshold - $\bar{P}_{Rx-Th}$. $\bar{T}_{Rx-Th}$ would then be the corresponding maximum distance a transmitting sensor could be from the *localizer*, beyond which the message would not be considered for localization. This limits the width of the sector (hence decreasing the numerator in Equation 4) and thus reduces the false positives as illustrated through simulations. We also note the trade-off in choosing $\bar{P}_{Rx-Th}$: a high value would mean that large fraction of messages are filtered out and *localizer* might have to stop at several more locations to ensure all sensors are localized; smaller values would increase false positives. We further study the choice of $\bar{T}_{Rx-Th}$ through simulations.

The *localizer* also uses the RSSI information to estimate the location of transmitting sensor with more accuracy and assign score on grids depending upon the distance between the grid and the estimated location of the sensor. Section 2.1.6 describes the score assignment method used by the *localizer*.

2.1.6. Score Assignment. Once a *zone* is computed, the *localizer* assigns grid scores as follows: For each grid $G_{i,j}$ that overlaps with the *zone* Z_k corresponding to a message transmission k, the grid score $p_{i,j,k}$ is the probability that the node corresponding to message k is located in the grid $G_{i,j}$ and is computed as,

$$p_{i,j,k} = f(x) * \frac{Area\ of\ overlap\ between\ G_{i,j}\ and\ Z_k}{Area\ of\ Z_k} \tag{4}$$

where $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x-\mu^2}{2\sigma^2}}$, μ is the distance corresponding to measured RSSI (T_{Rx}) and σ^2 is the error in RSSI measurement, ϵ_{RSSI}.

Finally, the cumulative score $P_{i,j}$ represents the probability that a grid $G_{i,j}$ consists of atleast one sensor. Initially, $P_{i,j}$ is set to zero. Subsequent values are computed using,

$$P_{i,j} = 1 - (1 - P_{i,j}) * (1 - p_{i,j,k})$$

$$P_{i,j} = 1 - \prod_{\forall k} (1 - p_{i,j,k}) \,. \tag{5}$$

Finally, the *localizer* uses the aggregate scores (i.e., $P_{i,j}$) for each grid cell to check if it consists a node or not. We propose a simple approach for score translation. A grid cell is assumed to consist a sensor if its score is greater than a certain threshold. If score is lesser, it is assumed not to contain a sensor. For illustration, assume a grid cell size of $d = \bar{T}_x/5 = 1$. Then, the maximum overlap area for a grid cell is approximately 0.39 (when it is farthest from the *localizer*). Thus, the maximum score is the overlap area divided by the sector area i.e. 0.179. We note that during every observation interval, at each corner of the grid cell (total of four corners), the *localizer* receives an average of 10 messages from each sensor, since it waits for $10/\lambda$ duration. The *localizer* receives more than 40 messages as it might receive messages from other locations as well. Now, assuming an average score of around 0.09 (half the maximum) and 40 messages, the aggregate score would be $1-(1-0.09)^{40} \approx 0.98$. Thus, a grid containing a sensor should get an aggregated score very close to 1. In this work, we assume a grid contains a sensor if $P_{i,j} > 0.95$.

2.1.7. Analysis. In this section, we derive some properties of the proposed protocol using theoretical analysis. First, we derive a lower bound on the probability of false positive when the proposed scheme is being used, where false positive is defined as follows. A grid cell is treated as a False Positive (FP) if the protocol incorrectly concludes that it contains a sensor while it does not.

Theorem 1 *Let there be a grid $G_{i,j}$ with no sensor in it. The probability of it being detected to be one with sensor is $T_{Th} \geq (1 - N(\frac{1}{2}(\frac{d}{2} + \epsilon_{RSSI}), \epsilon_{RSSI}) * \frac{d^2}{2Z_k})$, where,*

- *d is the diagonal length of a grid,*
- *ϵ_{RSSI} is the error in RSSI measurement, and*
- *Z_k is the area of the circle, a grid present within which will have a non-zero probability to be assigned with some score if the localizer is present in the circle.*

Proof 1 *Let r denote the radius of the circle within which $G_{i,j}$ will have a non-zero probability to be assigned with some score if the localizer is present in the circle. We approximate the radius $r = \bar{T}_{Rx-Th} + \frac{g}{2}$. The dotted area in Figure 2.1, shows the area, presence of a sensor in which will lead to score assignment to $G_{i,j}$ due to error in RSSI measurement, ϵ_{RSSI}. Radius of this circle, $\bar{r}$ is approximated as $(\epsilon_{RSSI} + \frac{d}{2})$.*

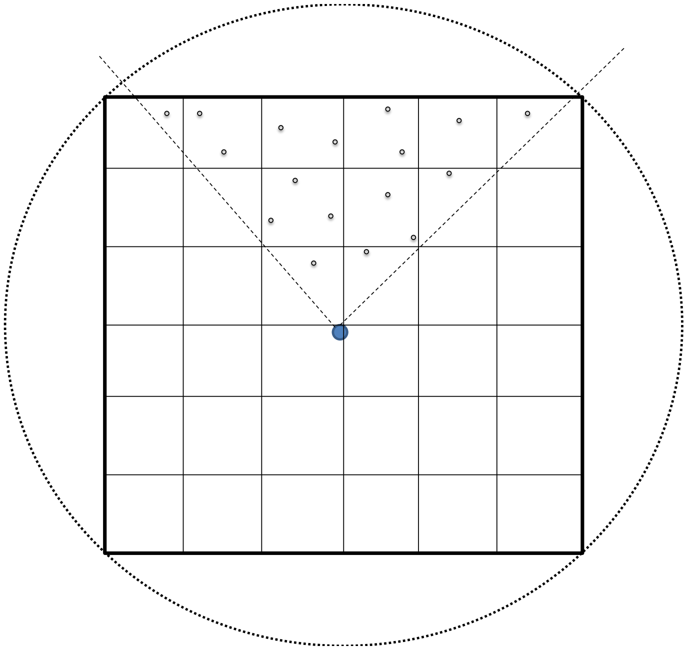


Figure 2.1. The *localizer* represented by the big dot and the area Z_k represented by dotted area

Number of grids enclosed in this circle is approximately $\frac{A_{circle}}{A_{grid}} - \frac{C_{circle}}{d}$ where A_{circle} is the area of the circle, A_{grid} is the area of the grid, C_{circle} is circumference of the circle and $d = \sqrt{2}g$ is the diagonal length of the grid. So the number of grids (n_{grids}) is,

$$n_{grids} = \frac{\pi r^2}{d^2} - \frac{2\pi r}{\sqrt{2}d} = \frac{\pi r}{d}(\frac{r}{d} - \sqrt{2}) = \frac{\pi r}{d^2}(r - \sqrt{2}d) \tag{6}$$

As in the proposed protocol the localizer *stops at every corner of a grid to capture sensor messages, the number of times the* localizer *will stop within the circle (n_{stops}) can be expressed as,*

$$n_{stops} \approx 4 * n_{grids} \tag{7}$$

The dotted area in Figure 2.1 is approximately,

$$A_{false-positive} = \pi\bar{r}^2 - d^2 = \pi(\epsilon_{RSSI} + \frac{d}{2})^2 - d^2 \tag{8}$$

Hence the number of messages originated from $A_{false-positive}$ at every time instant is denoted by $\rho A_{false-positive}\lambda$. The localizer *will be there for the time duration xn_{stops}. Hence total number of messages originated (n_{msg}) is as follows -*

$$\begin{aligned} n_{msg} &= x\ n_{stops}\ \rho\ A_{false-positive}\ \lambda \\ &= x\ \lambda\ \rho\ \frac{4\ \pi\ r}{d^2}(r - \sqrt{2}d)\ [\pi(\epsilon_{RSSI} + \frac{d}{2})^2 - d^2] \\ &= 4\ x\ \lambda\ \rho\ \pi\ r(r - \sqrt{2}d)[\frac{\pi}{d^2}(\epsilon_{RSSI} + \frac{d}{2})^2 - 1] \end{aligned} \tag{9}$$

Considering average case, $\frac{A_{overlap}}{A_{Z_k}} = \frac{\frac{d^2}{2}}{A_{Z_k}}$ Average distance between localizer and sensor within circle of radius $\frac{d}{2} + \epsilon_{RSSI}$ is D_{avg}, where

$$D_{avg} = (\bar{T}_{Rx-Th} + \frac{d}{2}) - [\frac{\frac{d}{2} + \epsilon_{RSSI}}{2} + \frac{\bar{T}_{Rx-Th} - \epsilon_{RSSI}}{2}] = \frac{\bar{T}_{Rx-Th} + \frac{d}{2}}{2}. \tag{10}$$

Now the score accumulated for $G_{i,j}$ due to error ϵ_{RSSI} can be expressed as, $(1 - \Pi(1 - f(x) * \frac{\frac{d^2}{2}}{Z_k}))$. So the probability of false positive is, $T_{Th} \geq (1 - N(\frac{1}{2}(\frac{d}{2} + \epsilon_{RSSI}), \epsilon_{RSSI}) * \frac{d^2}{2Z_k})$.

2.1.8. Performance Evaluations.

The performance of the proposed protocol is evaluated and analyzed in this section.

Experimental Setup: We developed a simulator using C++ for data collection. In order to emulate random sensor deployment real network scenario, varying number of nodes are randomly placed in an area of $1000m \times 1000m$ with the average density of nodes per grid as a constant for each run. For all the simulations, transmission range of the sensors is kept as 100m and message transmission rate λ is set to 0.05. A *localizer* moves in the network at a constant speed while pausing at corners of a grid for a wait time of T_{obs}. The maximum observation time of the mobile *localizer* at every point is set as $T_{obs} = \frac{10}{\lambda}$. For each set of parameters, we repeat the experiment for 20 different seeds for statistical reasons. We vary the number of nodes from 250 to 1000. A grid cell is categorized to contain a sensor if $P_{i,j} > 0.95$. Further, we introduce additional random attenuation/noise factor that could reduce the RSSI signal strength by up to 40% [171]. We also assumed an error bound in angle of arrival degrees of $\epsilon = 6\,\text{deg}$ [129].

Metrics: In specific, we define three metrics to study the performance of the proposed protocol, as presented below.

- Percentage of False Positives (P_{fp}): A grid cell is treated as a False Positive (FP) if the protocol incorrectly concludes that it contains a sensor while it does not. The percentage of FPs is computed as number of FPs divided by total number of grid cells i.e, $g \times g$.

- Percentage of False Negatives (P_{fn}): A grid cell is treated as a False Negative (FN), if the protocol concludes that the cell does not contain a sensor while the cell indeed contains atleast one sensor. The percentage of FNs is the number of FNs divided by total number of grid cells.

- Detection Time (T_d): The average time taken by the *localizer* to detect if there is a sensor in the grid or not. Whenever a *localizer* observes that a grid exceeded this threshold of 0.95, the *localizer* records the total time spent by far within the transmission range of $\bar{T}_{Rx-Th}$ distance [1].

We note that the aim to minimize both N_{fp} and N_{fn}. First, we study the trade-offs between the grid size and false positives/negatives when only AoA is considered. Later, we analyze the performance for the proposed improved technique where both AoA and RSSI are considered, along with error in estimation, $epsilon_{rssi}$. To elaborate, we study the improvement in the performance by selectively ignoring the messages with RSSI below $\bar{P}_{Rx-Th}$.

We perform three sets of experiments to study the effect of three different localization techniques on the false positives and false negatives. The three localization methods use score assignment using -

1. only AoA
2. AoA and RSSI
3. AoA and RSSI with error ($epsilon_{rssi}$)

Apart from that, in this work we also study the effect of varying RSSI threshold (illustrated later in this section) on FP and FNs, as well as the effect of varying λ on detection time.

Results: The performance of the proposed protocol is studied with respect to two different varying network parameters, viz, number of nodes in the network and grid size, d (length of a side of grid). Figure 2.2 present the performance of the

[1] Referring back to Section 2.1.7, $\bar{T}_{Rx-Th}$ is the distance corresponding to the threshold RSSI. The threshold RSSI value, $\bar{P}_{Rx-Th}$, is defined as the minimum RSSI of a message such that if $\bar{P}_{Rx-Th} > P_{received}$, then the message is not considered towards score accumulation by the *localizer*. Please note that $P_{received}$ is the RSSI of a message received by the *localizer* at any time.

proposed localization protocol for varying number of nodes in the network with only AoA information. We simulated the performance for various accuracies i.e., grid sizes (i.e., d = (20, 40, 60, 80) meters). Firstly, we note that the number of false negatives is very less than the corresponding number of false positives. In other words, if a node is present in a grid, the *localizer* correctly identifies it to contain a sensor with high very probability. On the other hand, the *localizer* might incorrectly identify grids to be containing a sensor even though they do not. We attribute the increased number of false positives to scenarios where a sensor is located closely to the grid boundary lines, i.e., close to the border of a grid. In such cases, most of the times, a false positive is produced. As the proposed protocol cannot accurately identify which grid the sensor is located in, and assigns high scores for multiple grids adjoining the borders of the grid where the sensor is located, false positives are likely to occur. Figure 2.2 show that in specific, with higher number of nodes in the network, the possibility of FPs and FNs usually increase as more number of sensors enhance the number of sensors located close to the grid boundaries. Furthermore, it can be observed that FPs tend to reduce with increasing value of d, as higher values of d result in reduction of number of sensors that are close to grid borders.

In Figure 2.3 both AOA and RSSI are considered to study FPs and FNs respectively. Like Figure 2.2, Figure 2.3 also include varying number of nodes as input parameter, and the results are plotted for all the four different values of d. The trend of results obtained in Figure 2.3 reflects that of Figure 2.2, except for the decline number of FPs and FNs for all the cases. It can also be noted that FP and FN rates stabilize for larger values of number of nodes. This happens because beyond a certain point, even if number of nodes and thereby message communication increases, it does not affect the score accumulation.This characteristics show the scalability of our approach.

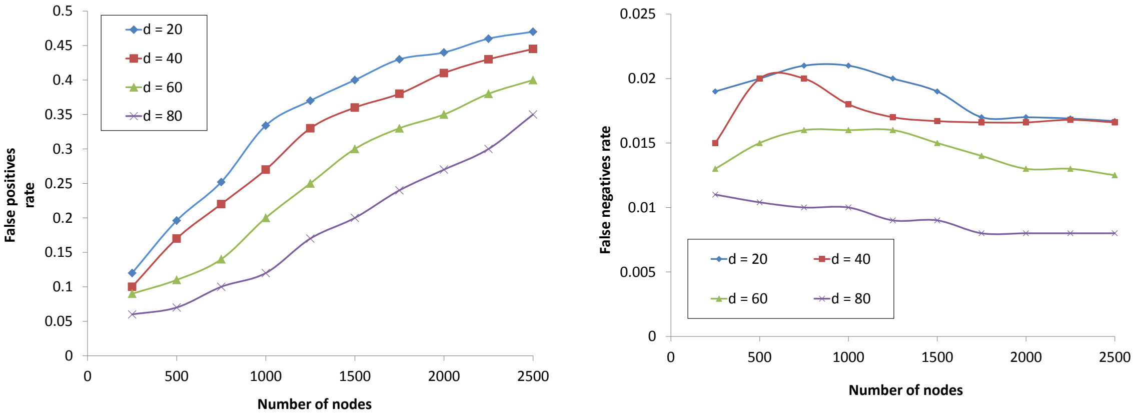


Figure 2.2. (a) False positives rate and (b) false negatives rate for different node density with AOA

In Figure 2.4, both AoA and RSSI with error, $epsilon_{rssi}$, are considered. The network parameters are same as the previous two figure, i.e., number of nodes and grid size. The improved performance of the proposed technique which includes AoA as well as RSSI information with it's error, $\epsilon_{RSSI} = 1.5$ for localization can be verified from the reduced number of FPs and FNs. The results of the same are plotted in Figure 2.4. They reflect similar trends as the previous figures, . These figures demonstrate the effectiveness of the proposed protocol over methods which do not consider both AOA and RSSI information, or fail to include the RSSI error margin. A better rate of convergence is achieved in results presented in Figure 2.4, as number of nodes increase.

It can be noted that in all these three figures, that is Figures 2.2, 2.3 and 2.4, better performance is obtained for larger values of d, compared to smaller values. This phenomenon can be explained from the fact that smaller d implies smaller error margin, which is more difficult to attain. But for larger values of the grid, score accumulation is fast, and probability of the grid having a sensor is higher too.

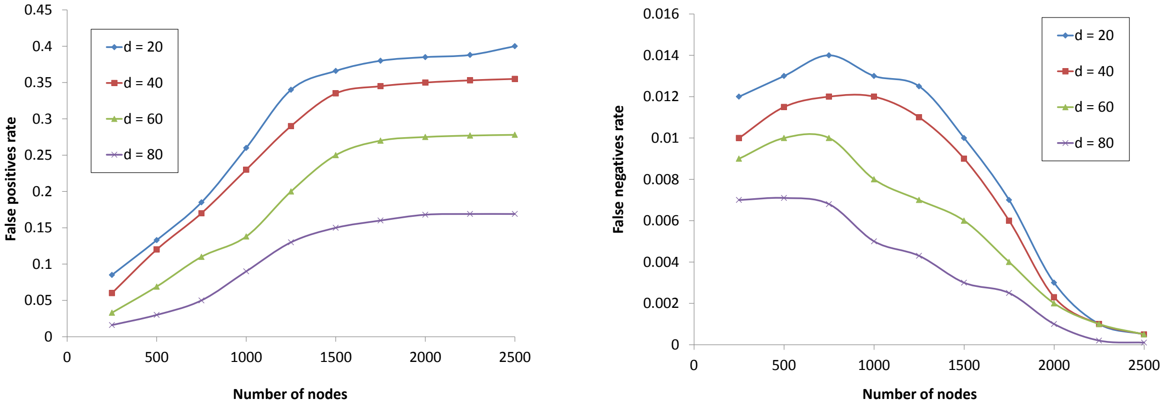


Figure 2.3. (a) False positives rate and (b) false negatives rate for different node density with AOA and RSSI

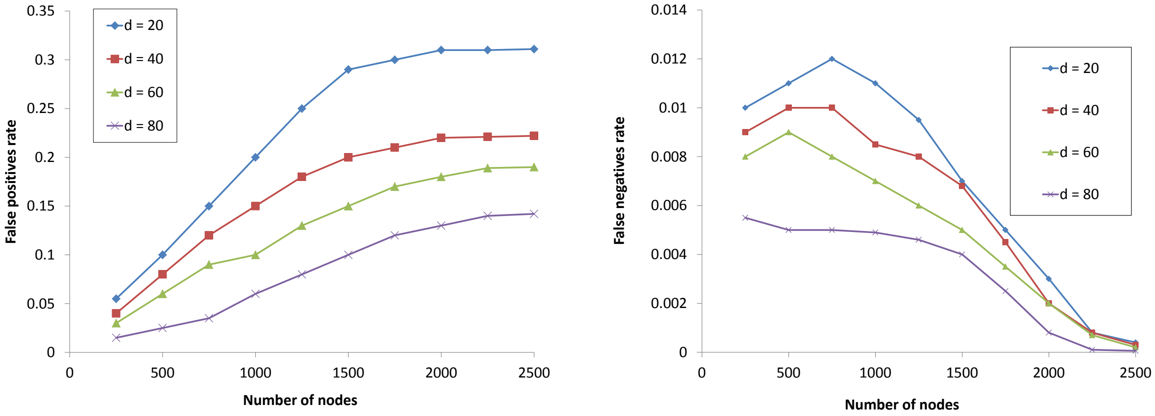


Figure 2.4. (a) False positives rate and (b) false negatives rate for different node density with AOA and RSSI with $epsilon_{rssi}$

Owing to this property of the system, both FP and FN rates are better for bigger size of the grid.

Figure 2.5 presents the results for different $\bar{P}_{Rx-Th}$, i.e., when we use RSSI information to filter messages received from farther sensors. Here $d = 20m$. For ease of presentation, we use the term *maximum sensor distance threshold* ($\bar{T}_{Rx-Th}$) to represent a corresponding maximum distance the filtering would permit. In other words, for a given $\bar{P}_{Rx-Th}$, $\bar{T}_{Rx-Th}$ is the distance that corresponds to a RSSI of $\bar{P}_{Rx-Th}$.

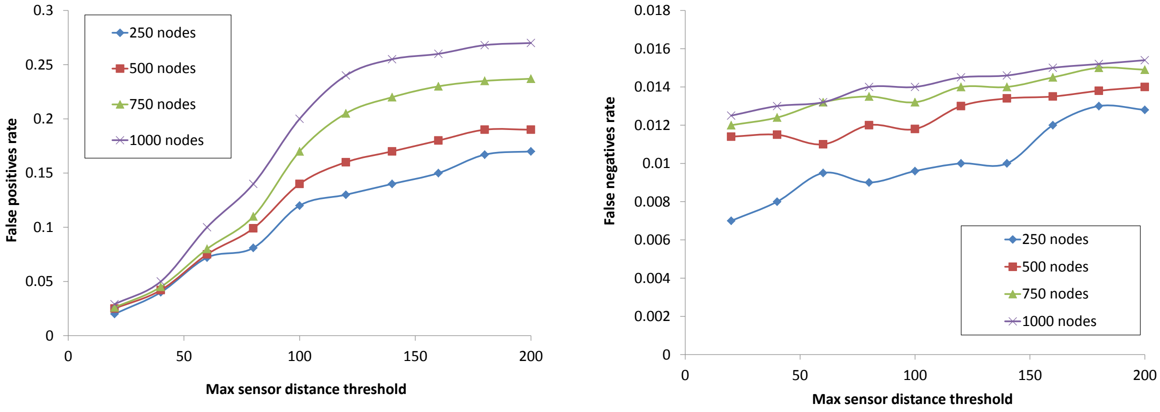


Figure 2.5. (a) False positives rate and (b) false negatives rate for different scenarios with AoA and RSSI with $epsilon_{rssi}$

We can see that choosing a high $\bar{P}_{Rx-Th}$ that corresponds to a low $\bar{T}_{Rx-Th}$ drastically reduces the number of FPs. The reason for this behavior (as explained in the previous section) is the shrinking of sector widths (to minimize far away grids from receiving higher scores) with RSSI information that was not the case with pure AoA. This enforces better fairness in eliminating far away unlikely locations, hence reducing the percentage of False Positives. The number of False Negatives does not change appreciably with RSSI , since RSSI filtering only helps eliminate potentially unlikely sensor locations; potentially correct locations are still retained.

Effect on Detection Time: The effect of varying packet arrival rate, λ, on detection time is studied in Figure 2.6. For cases when both AOA and RSSI are used, the average detection time is usually more than when only AoA is used. Considering RSSI error model with AOA improves the performance farther as the average detection time is the least for all values of λ when AOA and RSSI with error is used. These plots also demonstrate that for higher packer arrival rate, detection time is comparatively low, due to faster score accumulation under higher message traffic. However, for all the cases, detection time converges for a certain value of λ indicating the minimum detection time even with high message traffic. These plots clearly show that the proposed method is more efficient in reducing detection time compared to other methods.

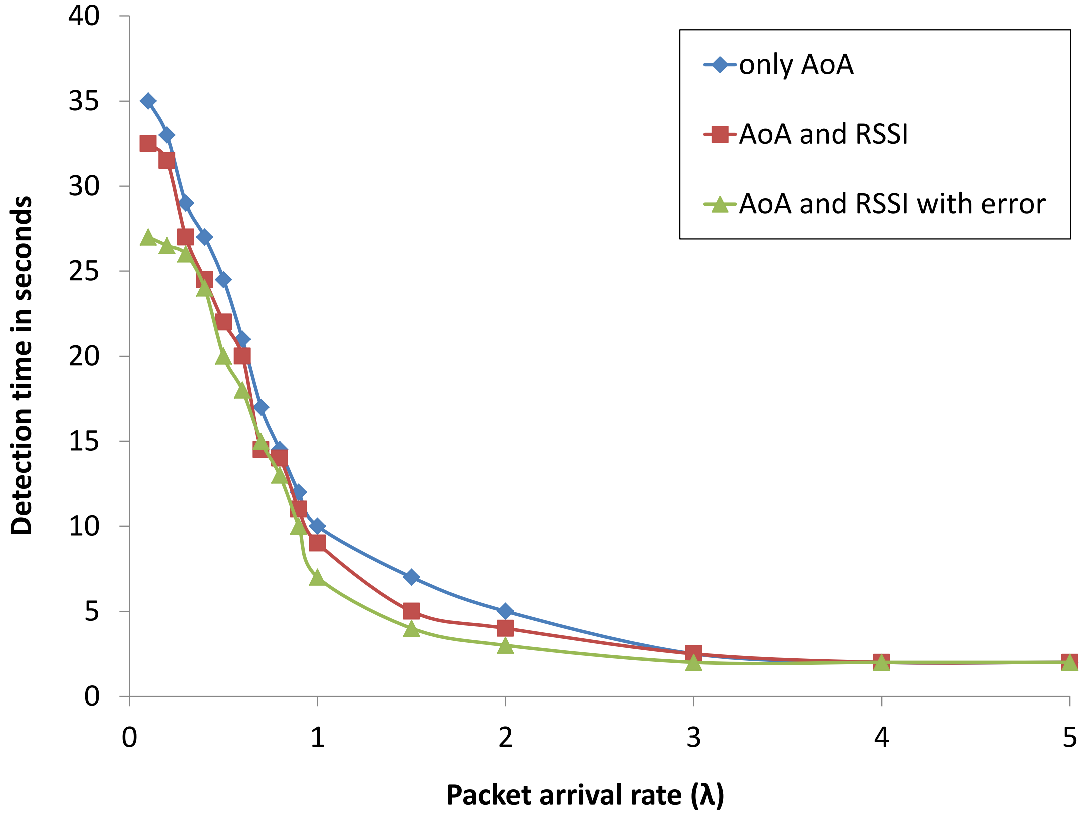


Figure 2.6. Detection time versus packet arrival rate

2.1.9. Final Remarks. This work studies the problem of localizing nodes in a wireless sensor networks without cooperation from sensors themselves. This is a practical problem in scenarios like battlefields where cooperation from sensors may not be available for localization. We propose a new method called NCLOCS wherein a localizer moves in the network and detects raw sensor communication signals, while measuring AoA and RSSI of the signals. We incorporate practical error models in these measurements and design a score assignment scheme for grid-based localization of the sensors. We theoretically derive a lower bound on the false positive for the proposed method. Depending on desired accuracy, our protocol can achieve very low false positives and false negatives. The detection time of sensors also lower significantly when the proposed method is employed. The work presented in this section shows that it is possible an outside entity to localize sensors in a wireless sensor network with some amount of error, even without any cooperation from the sensor network, and thereby compromising the location privacy of the entire network. This location information can be used to infer farther critical information that can be utilized in defending against or imposing security threats on inimical networks, making the proposed non-cooperative localization a novel addition to existing security issues in wireless sensor networks. Towards this premise, our next contribution in this section is in designing a light-weight distributed protocol for tracking a mobile intruder in a WSN while simultaneously preserving location privacy of the sensor nodes.

2.2. THE DEFENSE MODEL

Security in WSNs has been researched from various aspects such as confidentiality, availability and integrity. A critical input to many of the security threats discussed in the existing literature is sensors' location information. A host of benefits are patent to adversaries when sensor location privacy is compromised. For instance, the number of sensors nodes in the network can be estimated which can help gauge network strength; optimal intrusion paths involving minimal detection through the network can be determined, physical destruction of sensors can be accomplished to compromise network functionality etc. Hence *Adversarial localization* is an important privacy problem in Wireless Sensor Networks. Adversarial localization refers to attacks wherein an adversary aims to discover position of sensors in a network. Under such attacks *location privacy* of sensors is compromised. Defending against adversarial localization by protecting location privacy of sensors is hence a critical security requirement.

LPPT (Location Privacy Preserving Tracking): In this work, we address the problem of defending against adversarial localization by securing location privacy of sensors in wireless sensor networks. The contribution of our work is three-fold.

- We propose a technique for preserving sensor location privacy against adversarial localization. The proposed protocol, viz, *Location Privacy Preserving Tracking* or $LPPT$, reduces loss of location privacy upon detection of adversarial entity in the network.

- In addition to aiding location privacy of sensors, the proposed $LPPT$ protocol allows sensors to track the adversary with very few communication messages. The core challenge comes from the sensors performing two conflicting objectives: simultaneously localize the adversary, and hide from the adversary. The principle of the proposed approach is to allow sensors intelligently predict their own importance as a measure of these two conflicting requirements. Only a few *important* sensors will participate in any message transmissions during adversary localization. This ensures sufficient degree of adversary localization, while also protecting locations of many sensors.

- We study the adversary performance extensively through theoretical analysis as well as via simulations. We comparatively discuss the performance of a naive tracking protocol and $LPPT$ in securing sensors' location privacy while achieving better tracking accuracy. We also evaluate the energy efficiency of the proposed protocol through simulations.

The research presented in this section has been published in [46] and a journal version is ready to be submitted.

2.2.1. Background and Related Work.

Research on location privacy in WSN is presented in many of existing literature [110], [156], [81], [120]. There are two intuitive approaches to protect location privacy in WSNs. The first is to encrypt all sensor messages using techniques proposed by [55], [22], [104] etc. Adversaries will hence not be able to decrypt messages, and hence the identities of sensors appear to be preserved. Unfortunately, this technique fails since (even with encryption) the adversary can still measure raw physical (and location specific) properties of the wireless signals like Angle of Arrival (AoA) and Receive Signal Strength Indicator (RSSI) emitted by sensors, and then use triangulation/ trilateration techniques to localize sensors. Repeated messages from the same sensors naturally leak more location information until eventually sensors are accurately localized. The second approach is to let all sensors sleep, and so no information of sensors' positions is leaked. Unfortunately, the sensors do not accomplish the WSN mission in this case, and the network is rendered useless. Preserving location privacy of sensors while still maintaining sufficient network performance is very challenging, and is the focus of this work.

We point out that to the best of our knowledge, the work presented in this chapter is unique in terms of defending sensor networks against adversarial localization. This chapter is a revised and expanded version of [45] wherein we design a protocol for defending sensor networks against adversarial localization based on the attack model in [27]. However, the work in [45] had some limitations which are addressed in this paper, primarily from the perspective of evaluating the protocol from the perspective of location privacy, adversarial localization and energy efficiency.

Localization in WSNs: We now talk about some existing techniques for localization in WSNs. The problem of sensor localization has been very well studied, although in co-operative environments only. Usually, one or more beacon nodes with known positions assist in localizing other sensors. In works like [20, 121] sensors utilize their connectivity information (without making any distance estimates) to a small number of static beacons for localization. The localization algorithm includes determining centroids of triangles formed with beacon nodes, determining orientations of nodes with respect to beacons, triangle overlaps formed between beacons and regular nodes.

Localization in WSNs can be classified into range-free and range-based. In the former, sensors do not consider the physical distance between themselves and the sources of beacons. Rather they use just the connectivity information. In [20], sensors localize themselves as the centroid of reference beacons. The accuracy here is dependent on the separation between the beacons and their transmission range. In [121], the authors use angle of arrival (AoA) measure to beacons for localization. Sensors adjacent to beacons get their bearings directly by measuring AoA to beacons, and this information is propagated to their neighbors. This process is continued in a hop by hop fashion. When enough bearings with respect to the non-collinear beacons are collected, a node can estimate its position. In the APIT algorithm [69], a sensor can be located within a certain number of triangles estimated by the algorithms with respect to beacon positions. The final position is determined to be one within the center of gravity of the overlapping area of the triangles.

On the other hand, range based approaches use the distance between sensors and beacons for location estimation. Received signal strength indicator (RSSI) and time difference of arrival (TDOA) are two metrics that are typically used for range estimation. It is generally accepted that RSSI is not a good indicator of range, as power in radio signal can be significantly attenuated depending on the environment. Also the obstructions in the environment and shadow effects prevent RSSI from being a reliable metric. In works like [129, 142], distance measurement between sensors and static beacons are utilized for location estimation. In the Cricket indoor location support system [129], the range estimation between sensors and beacons is done typically using the Time Difference of Arrival (TDOA) method. In this method, the range is determined as a function of the time difference of arrival between an RF signal and an ultra sound signal, since RF signals travel much faster than ultrasound signals. A similar method is used for range determination in [142], although the authors focus on ad hoc deployments in an outdoor environment. A bio-inspired distance estimation based collaborative location technique is presented in [88]. Another distance reconstruction based localization method is proposed in [101]. Time of arrival is used for source node localization in the work in [51].

2.2.2. Problem Definition. The problem of defending WSNs against adversarial localization can be viewed as a game played between two opposing entities - the sensors in the network and the adversary. The goal of the sensors is to localize the adversary, while simultaneously minimizing information leakage in terms of communication messages. The adversary's goal is to physically move in the network, while simultaneously attempting to localize sensors.

Sensor Network Model: In our work, we consider a sensor network where the deployment field is clustered into multiple grids. Clustering a sensor network has been widely adopted in practice like [157], [71], [1]. Advantages of clustering include better network scalability, decreased routing complexity, improved power efficiency etc. We assume that sensors know their positions in the network, which can be accomplished using localization techniques in [19], [73]. We also assume that sensors encrypt their messages using light weight techniques like [55], [22]. The mission of the sensors is to localize adversaries physically moving in the network. To do so, sensors are equipped with ranging hardware that they use to determine distances from the adversary (can be accomplished by typical vibration or infrared sensors).

Adversary Model: We consider an adversary that is physically moving in the network like a programmable robot. The adversary's movement is either random or controlled. While the adversary can have any objective in its mobility, it also has the objective of localizing sensor positions passively. By passive, we mean that the adversary will not launch any active attack on the network like breaking into sensors to determine their positions, or disclose encryption keys. Rather the adversary will discover sensor positions based on information leakage of radio signals which sensors transmit in the network. The adversary will accomplish this by passively intercepting communication messages, and measuring raw physical properties like RSSI or AoA or both. Using these measurements, sensor localization can be done via triangulation/ trilateration techniques [28]. Clearly, more the number of messages from the sensors, more is the information leaked to the adversary, and better is the adversary's estimate of the sensor positions. The performance of the sensors is measured by means of a metric called *Adversary Location Certainty*, which denotes how accurately the sensors localize the adversary. The success of the adversary is measured by a metric called *Sensor Location Leakage*, which is quantified by the area within which the adversary can correctly localize the sensors. The main problem addressed in this work is to design a protocol to be executed by sensors at run-time that -

1. maximizes *Adversary Location Certainty*,
2. minimizes *Sensor Location Leakage* simultaneously.

Based on the defined problem, a location privacy preserving tracking algorithm is proposed.

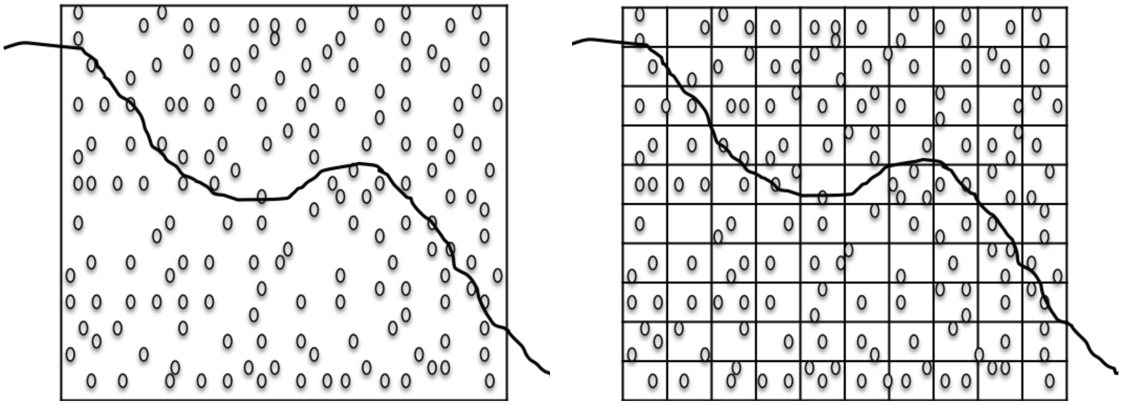
2.2.3. Proposed Solution. Exploiting the trade-off between localization performance and location privacy, we propose *LPPT*, a light-weight and distributed protocol for defending wireless sensor networks against adversarial localization.

Preliminaries: Before introducing the steps involved in $LPPT$, we first define and discuss important definitions used later in the section.

1) Grid Level Localization: The role of the sensors in the network is to primarily localize the adversary. Since sensor network hardware and the nature of wireless medium are error prone, accurate tracking of any intruder necessitates significantly large number of sensor readings and messages as demonstrated by work in [1], [99], [66], [17], [159], [89]. However, in many practical applications of sensor networks for target tracking, accurate (point level) localization may be an overkill. For example, in a typical indoor environment, it may be enough if sensors can localize a target to a room rather than a point. Even in outdoor battlefield environments, localizing an intruder within (say) a few meters is practically enough during surveillance. Motivated by these practical considerations, in this section, the localization accuracy is *Grid Level.* By Grid Level localization we mean that the adversary is considered to be *localized* at all times when sensors are correctly aware of grid where the adversary is physically present, and *lost* at other times. Note that the grid size is application specified depending on the nature of the sensor network mission.

In this work, we initially assume (for ease of elucidation) that the deployment field is fully covered, i.e., every point in the field is within the sensing range of one or more sensors. However, the proposed scheme will work without modification even if this assumption does not hold. Note that in order to localize any adversary in the network, sensors will have to communicate using wireless messages. Even in grid level localization, since multiple sensors will likely be sensing the adversary at any point in time, there may be multiple messages transmitted. Clearly there is a tradeoff between accuracy of localizing the adversary and number of messages transmitted by sensors even if the localization accuracy is relaxed.

Figure 2.7(a) and Figure 2.7(b) represent the difference between point level and grid level localization using a simple network structure. The solid line in both the figures correspond to the path taken by a localizer in the network. The shaded dots represent the sensors. In Figure 2.7(a) the sensors aim at point level localization, which is to localize the adversary with a negligible error. In Figure 2.7(b) the entire network is divided into square grids of equal size. The objective of a scheme using such an approach is to localize an adversary to be inside one of the grids. In this work, we follow the second approach for adversarial localization.


(a) point level localization

(b) grid level localization

Figure 2.7. Different levels of localization

Although we assume that the network area is virtually divided into a number of square grids, it can be noted that the grids can as well be of an irregular shape. In practical scenarios, dividing a network area into perfect square grids might not be always feasible because of the irregularity of the shape of the region. Still, a virtual grid structure can still be overlaid by approximating the ends of the network area to be part of a perfect square grid.

2) Fixed Parameters: Here we discuss definitions for Fixed Parameters, which are those parameters in $LPPT$ whose values remain unchanged during the entire network mission. There are five *fixed parameters* used in the $LPPT$ protocol -

Neighboring Grids, d^i_{max}, d^i_{min}, d_{max} *sensor* and d_{min} *sensor*. For ease of understanding, we consider a square grid. However, the definitions below (and the proposed protocol) can work for arbitrary shaped grids as well.

Neighboring Grids: For each grid in the network, we divide their neighboring grids into two classes Regular Neighbors and Corner Neighbors. Regular Neighbors of a Grid, g, are those neighboring grids which are in the immediate Up, Down, Left and Right position of g. Corner Neighbors of a Grid, g, are those neighboring grids which are in the immediate diagonal positions of g. We also assume that the transmission range of every sensor is long enough so that it can communicate with all sensors in its neighboring grids.

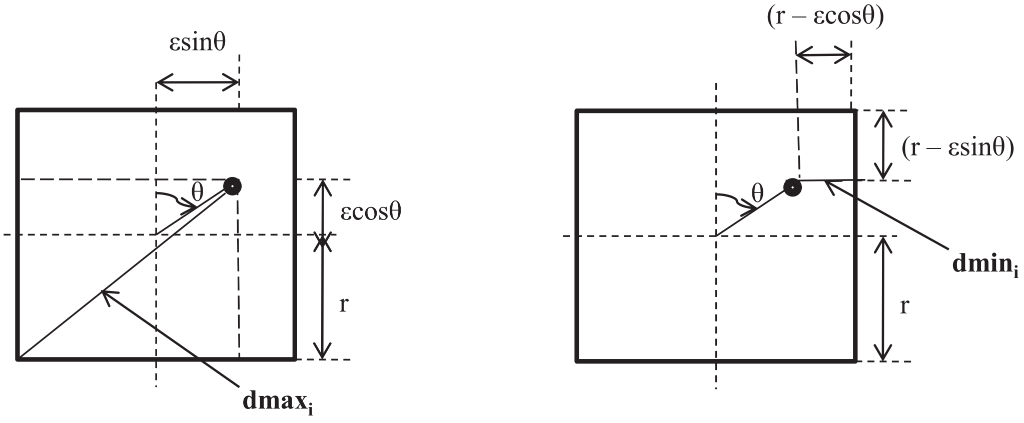
d^i_{max}: For each grid in the network, d^i_{max} is the minimum Euclidean distance between the sensor i in the grid and the adversary, beyond which the sensor i can deterministically assume that the adversary is not in the same grid as itself. In other words, it is the distance between a sensor and the farthest boundary point of the grid in which the sensor is present. We consider every grid to have a vertical axis passing through its center, and clockwise angle is measured to be positive. Consider a sensor i (represented as a dot) in Figure 2.8(a). Let the angle made by a straight line drawn between the center of the grid and the sensor i, and the vertical axis be θ and the distance between the center of the grid and the sensor be ϵ. Considering that the sensor might be included in any of the four quadrants:

$$d^i_{max} = \sqrt{2r^2 + (-1)^n\, 2r\epsilon \left(\sin\theta + (-1)^k \cos\theta\right) + \epsilon^2} \qquad (11)$$

$$\text{where, } n = \left\lfloor \frac{\theta}{180} \right\rfloor \text{ and } k = \left\lfloor \frac{\theta}{90} \right\rfloor .$$

d^i_{min}: For each grid in the network, d^i_{min} is the maximum Euclidian distance between the sensor i in the grid and the adversary, within which a sensor can deterministically assume that the adversary is in the same grid as itself. That is, it is the distance between a sensor and the closest boundary point of that grid in which the sensor is present, as shown in Figure 2.8(b), and given by:

$$d^i_{min} = \lceil \{r + (-1)^n \epsilon \sin\theta\}, \{r + (-1)^m \epsilon \cos\theta\} \rceil \tag{12}$$
$$\text{where, } n = \left\lceil \frac{\theta}{180} \right\rceil \text{ and } m = \left\lceil \frac{\theta + 90}{180} \right\rceil.$$



(a) d^i_{max} calculation

(b) d^i_{min} calculation

Figure 2.8. Illustration of d^i_{max} and d^i_{min}

d_{max} and d_{min} *sensor*: For each grid, the sensor with the minimum value of d^i_{max} among all sensors in that grid is the d_{max} *sensor* of the grid. Similarly, the sensor with the maximum value of d^i_{min} among all sensors in that grid is the d_{max} *sensor*. We also define the d_{max} and d_{min} circles as the circles whose centers are the positions of the d_{max} and d_{min} sensors, and whose radii are d_{max} and d_{min} respectively for each grid.

We wish to point out that each sensor can calculate the above parameters independently with knowledge of other sensor positions in the grid. The parameters once determined are fixed, and do not change subsequently.

3) Dynamic Parameters: We now discuss the parameters used in $LPPT$ whose values are dynamically altered as a function of adversary's last estimated location, namely, $d^{'i}_{max}$, $d^{'i}_{min}$, $d^{'}_{max}$ *sensor* and $d^{'}_{min}$ *sensor*. For all subsequent discussions in this section, we consider that the adversary is currently localized in Grid g.

$d^{'i}_{max}$: For each sensor i in a corner neighbor of grid g, we define its $d^{'i}_{max}$ as the distance between sensor i and the edge of the corner neighbor, closest to the adversary.

$d^{'i}_{min}$: For each sensor i in the neighboring grids (both regular and corner neighbors) of grid g, we define $d^{'i}_{min}$ as the distance between sensor i and the perpendicular distance between sensor i and the boundary of Grid g.

$d^{'}_{max}$ *sensor*: Recall that there can be different values for $d^{'i}_{max}$ for each sensor i in the neighboring grids of grid g. For each sensor in a neighboring grid, we define its $d^{'i}_{max}$ circle as a circle of radius $d^{'i}_{max}$ centered at the location of sensor i. Among all such sensors, the one whose $d^{'i}_{max}$ circle overlaps the most with its neighboring grids is considered to be $d^{'}_{max}$ *sensor* of Grid g.

$d^{'}_{min}$ *sensor*: Recall that there can be different values for $d^{'i}_{min}$ for each sensor i in the corner neighbors of grid g. For each sensor in a particular corner neighbor, we define its $d^{'i}_{min}$ circle as a circle of radius $d^{'i}_{min}$ centered at the location of sensor i. Among all such sensors, the one whose $d^{'i}_{min}$ circle covers the most area in its grid is considered to be d^{i}_{min} sensor of Grid g.

We wish to point out here that the values for the dynamic parameters can change based on which grid the adversary is currently localized. However, irrespective of where the adversary is localized, depending on the current position of the adversary, each sensor can calculate each of the above parameters independently with known positions of other sensors in the grid.

Location Privacy Preserving Tracking Protocol Description

We now discuss the details of the $LPPT$ protocol. The protocol is presented in Algorithm 2, and is divided into two broad phases: *Initialization phase* and *Post Initialization Phase*. During the initialization phase, the adversary is localized for the first time in the network. The post initialization phase deals with the tracking of the adversary as it moves in the network, while sensors are also attempting to preserve their location privacy from the adversary. The details of both phases are discussed below:

1. *Initialization Phase:* Initially when the adversary enters the network, the first three sensors sensing the adversary will exchange messages and triangulate the adversary to a grid. This is the initialization phase, following which all sensors start executing the protocol for adversary localization and sensor location privacy preservation as discussed later in this section.

2. *Post Initialization Phase:* In this phase, the sensors in neighboring grid of the adversary calculate $d^{'}_{max}$ and $d^{'}_{min}$ $sensors$ and update them with each new grid location of the adversary. We define adversary-sensing circle of a sensor S_X to be the circle with S_X at center and radius equal to the distance between adversary and sensor S_X.

   Since adversary localization is grid level, the protocol allows message transmission by sensors only when a grid switch by the adversary is detected. At different steps of the $LPPT$ protocol, the sensors use four different types of messages - $definitive\ grid\ switch\ indicator$ (m_1), $call\ for\ collaboration$ (m_2), $collaboration\ for\ localization$ (m_3) and $triangulation\ initiator$ (m_4). Each message contains a $message\ type$ field whose value indicates which of the four types the message is, and is sent out based on two possible cases of the adversary switching grids, as presented subsequently. We are going to discuss the purpose of these four messages subsequently in this section.

Case 1 – Definitive Grid Switch Detection: This is the case where sensors in the network can detect a grid switch by the adversary and also are aware of which grid the adversary switches to. In this case, only a single message m_1 is transmitted to update sensors in vicinity of the adversary about its grid switch. The two possibilities leading to *Definitive Grid Switch Detection* are following.

(a) When a d_{min} sensor of a neighboring grid, say g^*, starts sensing the adversary, it indicates the adversary's movement into the d_{min} circle, which is possible only in case of a grid switch to g^*. The d_{min} sensor broadcasts message m_1 updating current position of adversary to be g^* and adversary's distance from it. As a grid switch is detected, all the sensors receiving m_1 update the set of neighbors of current grid, $d^{'}_{max}$ and $d^{'}_{min}\ sensors$. In this case, only a single message m_1 is transmitted to update sensors in vicinity of the adversary about its grid switch.

Every time the location of the adversary is updated through different messages throughout execution of $LPPT$, sensors sending and receiving those messages store the updated locations. This information is later reported to the base station through collaborative routing. We do not discuss about the routing as it is beyond the scope of the proposed technique, however existing techniques can be used for the same.

(b) Another possibility of identifying and broadcasting adversary's grid switch using only one message is when d'_{min} *sensor* of a neighboring grid g^* starts sensing the adversary, indicating adversary's grid switch to g^*. Similar to *Case 1 (a)*, the d'_{min} *sensor* of g^* updates the location information and its distance from adversary by broadcasting message m_1.

Case 2 – Potential Grid Switch Detection: This is the case where sensors in the network can detect a grid switch by the adversary, but are uncertain of the grid the adversary has switched to. This case can also occur following any of the two possibilities, as discussed subsequently.

(a) Before *Case 1 (a)* or *Case 1 (b)* occurs for a particular grid switch of the adversary, the grid switch can be detected when the d_{max} *sensor* of current grid stops sensing the adversary. Although it signifies adversary's movement outside the current grid, this itself cannot be used to determine which neighboring grid the adversary has moved to. Therefore the following steps will be executed.

Step 1: The d_{max} *sensor* will broadcast message m_2 to initiate the localization procedure.

Step 2: Upon receiving message m_2, every sensor sensing the adversary computes the two points of intersection of the sensing circles of the sender of m_2 and itself. The adversary should be present in either of these two points. Based on the location of these points, the sensors intelligently try to localize the adversary. The next steps of the protocol can be divided into two more sub-cases based on the location of the intersection points.

(i) If the two points are included in a single grid, say g^{**}, the adversary has clearly moved to grid g^{**}. Otherwise, if only one of the intersection points falls in a neighboring grid (say g^{**}) of last known adversary grid, g^{**} is the new location of the adversary, as the adversary can move only to neighboring grids

in a single step. After concluding about adversary's new location, a sensor S_i waits for a time t, (where t is the distance of the adversary from S_i), to avoid any redundant message communication. In this time interval t, the adversary might have moved within the grid or to a new grid, leading a d_{min} *sensor* or d'_{min} *sensor* to start sensing the adversary and sending m_1. Hence, after waiting for time t, if no other sensor has sent m_1 in this interval of time t, S_i broadcasts m_3 updating the grid location of adversary and its distance from S_i. In this case, two messages, m_2 and m_3, are broadcast among the sensors to determine and update the new grid location of the adversary.

(ii) If both the points of intersection fall into different neighboring grids, it is possible that the adversary has moved to any of the two grids. So after waiting for time t, if another sensor has sent m_1 or m_2 in that time, a sensor S_i broadcasts message m_3. Every sensor S_j receiving m_3 and sensing the adversary computes point of intersection of sensing circles of sender of m_2 and m_3 and itself. The grid containing this point is recognized by sensor S_j as the new location of the adversary. To avoid redundant message, sensor S_j waits for time t'. We define t' as, $t' = [(\text{distance of the adversary from } S_j) + c]$, where c is a constant. The constant c is chosen large enough to ensure that the value of t' is most likely greater than the value of t (for other sensors whose positions are known to Sensor S_j), so that if any sensor can satisfy the criteria mentioned in *(i)*, it can send message before S_j and thus only two messages will be required to perform the localization. If no other sensor has sent message m_4 in that time t', S_j broadcasts m_4 updating new position of the adversary. In this case, three messages, m_2, m_3 and m_4 are required to localize the adversary.

b) Before a grid switch is detected by any of the previous cases, d'_{max} *sensor* of a corner neighbor of the current grid might start sensing the adversary. It signifies that the adversary has moved to a new grid, but this information is

Algorithm 2 Pseudocode of the *LPPT* defense protocol

```
Initialization Phase
while When adversary is first sensed by three sensors do
  Triangulate adversary to a grid
end while
End Initialization Phase

Post Initialization Phase
for Each step of adversary do
  while Adversary position is in grid $g$ do
    if Adversary enters $d_{min}$ circle of Grid $g^*$ then
      Refer to Section 2.2.3 Case 1 (a)
    else if Adversary enters $d'_{min}$ circle of Grid $g^*$ then
      Refer to Section 2.2.3 Case 1 (b)
    else if Adversary enters $d_{max}$ circle of Grid $g$ then
      Refer to Section 2.2.3 Case 2 (a)
    else if Adversary enters $d'_{max}$ circle of corner neighbor $g^{**}$ of Grid $g$ then
      Refer to Section 2.2.3 Case 2 (b)
    end if
  end while
end for
```

insufficient to determine exactly which grid the adversary has moved to. In this case, two steps similar to *Case 2 (a)* are followed.

Step 1: The d'_{max} *sensor* broadcasts message m_2 and the localization procedure begins.

Step 2: The successive steps in this case are exactly similar to the *Step 2* in *Case 2 (a)*. Even in this case, either two or three messages are required to locate the adversary.

Summary: The proposed *LPPT* protocol enables sensors localize the adversary with minimal messages via a combination of intelligent use of the previous adversary location, and sensors' current locations. However, it is likely that under unfavorable sensor deployments, and due to constraints on preserving location privacy, the adversary can be *lost* in some cases. We study this trade-off (i.e., accuracy of localizing adversary vs. preserving sensor location privacy) extensively in Section 2.2.5. We wish to point out that $LPPT$ protocol is executed by sensors independent of the objective of the adversary.

2.2.4. Analysis of LPPT. In this section we perform theoretical analysis of $LPPT$ protocol from the perspective of sensor location privacy compared to a brute force tracking algorithm where all sensors send messages on sensing an adversary.

Let us define the circle C_X as one with radius p centered at the sensor S_X such that any message S_X transmitted can be sensed by the adversary if it is within C_X. The adversary is moving within the network with speed v. The sensors send periodic messages at a rate λ when they sense the adversary. On an average, the adversary will spend $\pi p^2/2v$ time units in the circle C_X and receive $N = \lambda\pi p^2/2v$ messages from S_X.

Now we analyze the scenario considering the proposed defense protocol being employed by the sensors in the network. We assume that the sensing range of a sensor is similar to that of the adversary. The probability that a sensor is a d^i_{min} sensor or a d^i_{max} sensor is $1/\rho$, where ρ is the average density of sensors per grid. The probability that a sensor is d'^i_{min} or a d'^i_{max} sensor is given by $\sum_{k=0}^{8} \rho^{-k}$. Hence the average number of messages sent per time unit is $0.67 \left(1/\rho + \sum_{k=0}^{8} \rho^{-k}\right)$. So messages sent during the average time adversary is in C_X is $N_{new} = 0.67 \left(1/\rho + \sum_{k=0}^{8} \rho^{-k}\right) \frac{\pi p^2}{2v}$.

Figure 2.9 demonstrates the sensor location leakage for $LPPT$ vs. the brute force tracking. Since more number of messages leads to more location leakage of the sensors, the simulation data presented shows that location privacy of sensors is significantly high in our protocol. With lower values of speed of adversary, the number of message savings (and consequently location privacy) is more compared to brute force method. This phenomenon can be explained from the fact that the adversary has more chances to receive sensor messages in the brute force protocol (compared to $LPPT$ protocol) when it moves slow. The $LPPT$ protocol is consistent in terms of message conservation for all varying adversary speeds, since it requires message transmissions only upon grid switch by adversary. We present more results and insights pertaining to actual location disclosure of sensors by the adversary with *LPPT* protocol in Section 2.2.5. Note that the different speeds of adversary chosen in Figure 2.9 are consistent with speeds of typical miniature robots as evidenced in [34], [78], [57].

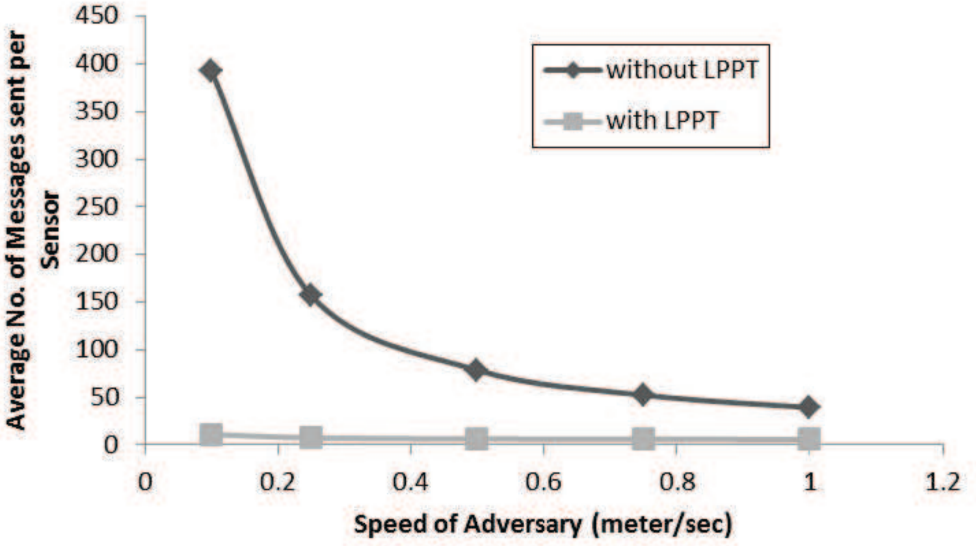


Figure 2.9. Comparison of the average number of messages sent with/without the LPPT protocol

2.2.5. Performance Evaluation. In this section, we evaluate the performance of our defense protocol via simulations. We consider a sensor network clustered into 15×15 grids, where the sensors are uniformly and randomly deployed. By default the sensor density is 10 sensors per grid. The adversary randomly moves in the network for 12,000 time units, and is equipped with unlimited memory to store and process communication messages sent by sensors. To clearly demonstrate the strength of our protocol, we assume that the radio range of the adversary is unlimited (i.e., the adversary can eavesdrop on any message sent by any sensor in the network). We have three key performance metrics, described below.

Metrics: The performance metrics that we use to evaluate the proposed $LPPT$ protocol are as following:

1. *Adversary Location Certainty:* It is the percentage of time that the adversary's position is correctly localized to a grid. We also study the percentage of currently detected grid switches by the adversary to show the true positive rate of our protocol in terms of tracking adversary's movement.

2. *Sensor Location Leakage:* Sensor location leakage is quantified by the area within which the adversary can correctly localize the sensors. Ideally, when the area of localization is large, the leakage is less and more is the location privacy.

3. *Energy Efficiency:* We also study the energy efficiency of our protocol in terms of how the motivation for protecting sensor location privacy (via message conservation) also impacts energy efficiency of the overall network.

Simulation Results: The results obtained from the self-made C++ simulator is plotted and analyzed to evaluate the performance of the proposed protocol. *Evaluating Adversary Location Certainty :* In Figure 2.10, we plot *Adversary Location Certainty* as a function of number of time units spent by the adversary in the network for a network density of 10 sensors per grid. As we can see close to 90% of the time the sensors are able to correctly localize the adversary in our protocol. The loss of about 10% in localization certainty comes from the trade-off is minimizing message communication which is evaluated subsequently.

In Figure 2.11, we plot *Adversary Location Certainty* as a function of number of sensors per grid when the time spent by the adversary is $12,000$ time units. As

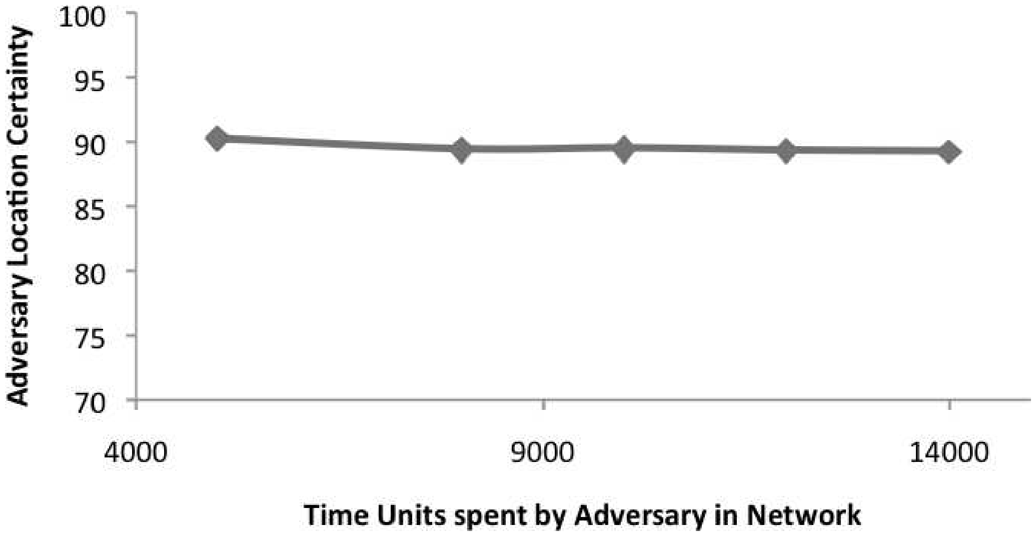


Figure 2.10. Adversary's location certainty Vs. Time spent by adversary in the network

we can see, even for very low sensor densities of 3 sensors per grid, the *Adversary Location Certainty* is 75%. When the density increases, the success of localizing the adversary dramatically increases. The success rate begins to flatten at around 95% beyond when the number of sensors per grid is 15. This is due to certain ephemeral stay at certain grids, which are unavoidable when the adversary moves in the network. Such ephemeral transitions are very difficult to detect in practice even when sensor density is extremely large.

It can be noted that the number of grid switches detected by the sensors can be different from the adversary's location certainty. This difference occurs due to the fact that, a grid switch can be detected in some cases, without correctly detecting the exact grid the adversary has moved to. In Figure 2.12 and Figure 2.13, we study the accuracy of grid switch detection over different network parameters.

As we propose grid level localization to $LPPT$, detecting each grid switch of the adversary can be considered to be as good as tracking it with 100% precision. However, due to range-based measurement errors and presence of uncertainty region

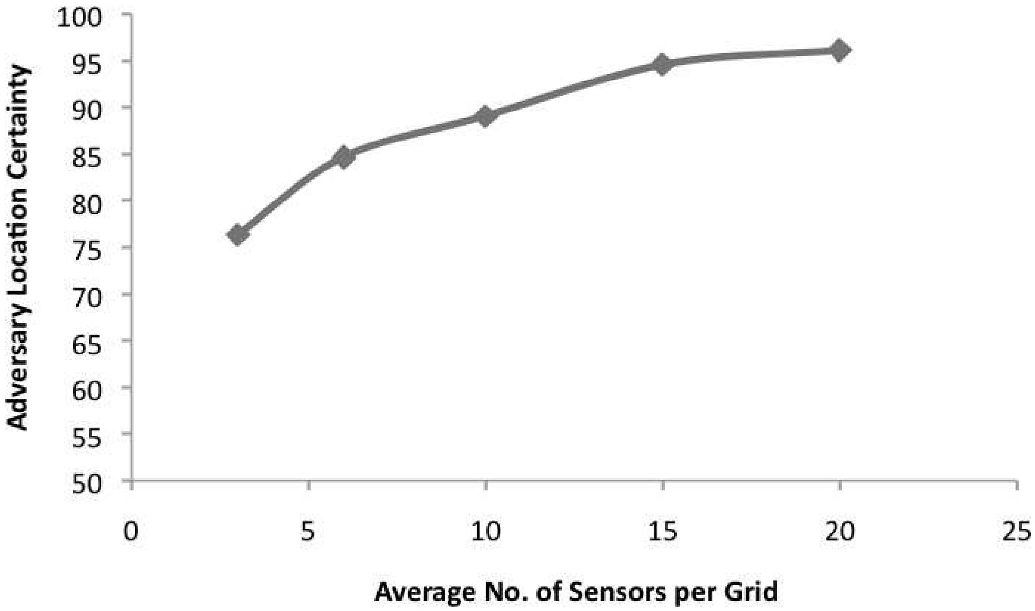


Figure 2.11. Adversary's location certainty Vs. Density of sensors in the network

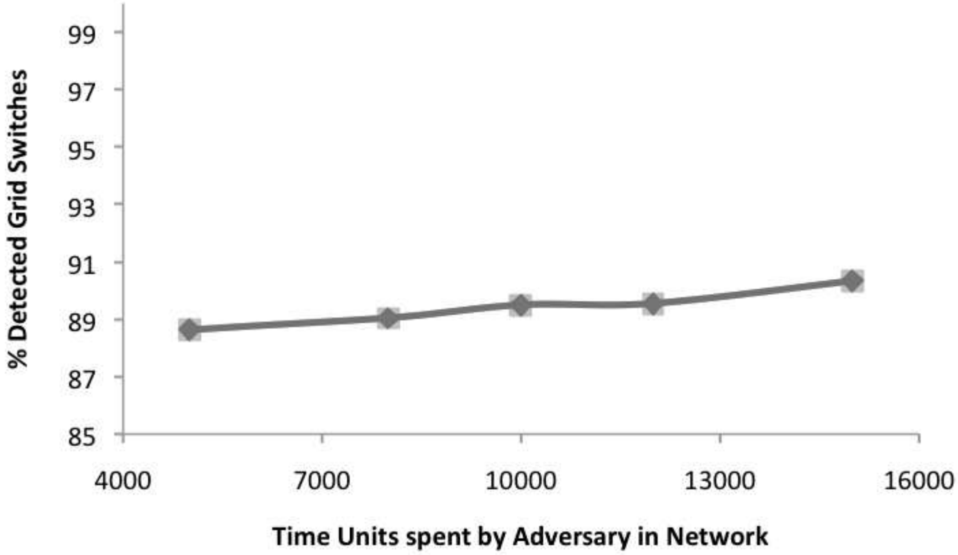


Figure 2.12. Percentage of detected grid switch by adversary Vs. Time spent by adversary in the network

in the network, the grid switches might not be detected accurately all the time. But our simulation results show that in about 90% of the cases, grid switch is detected correctly. Figure 2.13 denotes that the percentage of grid switch detected by the adversary is not affected by time spent by the adversary in the network as it remains almost parallel to X-axis.

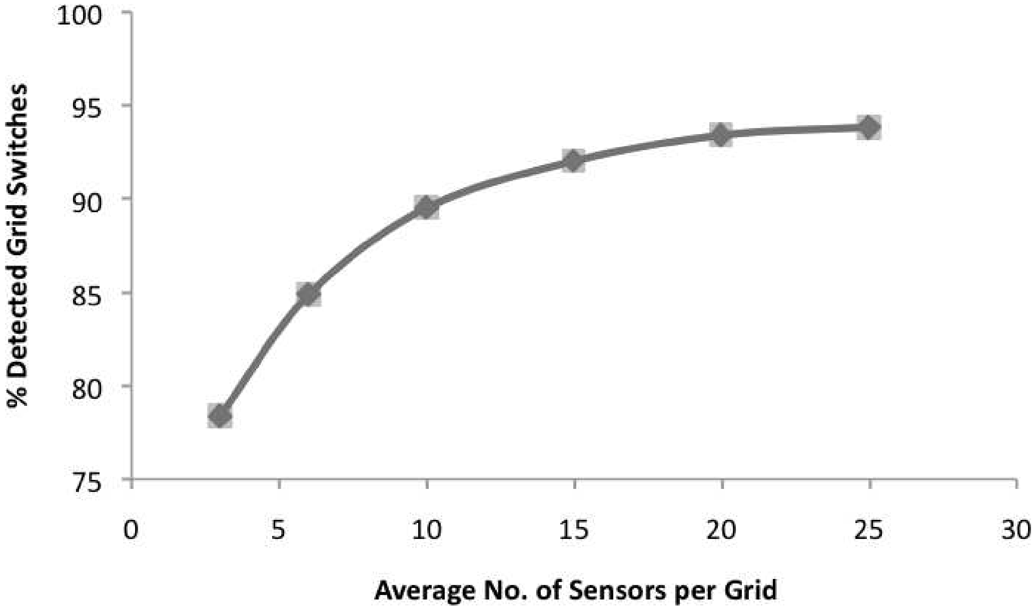


Figure 2.13. Percentage of detected grid switch by adversary Vs. Density of sensors in the network

Figure 2.13 depicts that with increasing sensor density in the network, percentage of detected grid switch improves, and gradually converges close to 95%. It does not reach 100% for the same reason as above in that ephemeral grid switches are very difficult to detect even with high sensor density.

Evaluating Sensor Location Leakage : We now study the location privacy gained by sensors while tracking the adversary with $LPPT$ protocol. Our metric of *sensor location leakage* is the area within which adversary can localize a sensor. When the area of localization by the adversary is larger, less is the location leakage, and more is the location privacy for the sensor network. Figure 2.14 shows the trend of Location Leakage with respect to time units spent by the adversary in the network. First, we observe that when the adversary spends more time in the network, area within which sensor is localized decreases. This phenomenon occurs as with more time spent in the network, the adversary can capture more messages transmitted by sensors. However, with $LPPT$ protocol, the area within which a sensor is localized is much larger (i.e, sensor location privacy increases) compared to the brute force protocol, clearly depicting the effectiveness of $LPPT$ in preserving location privacy of sensors.

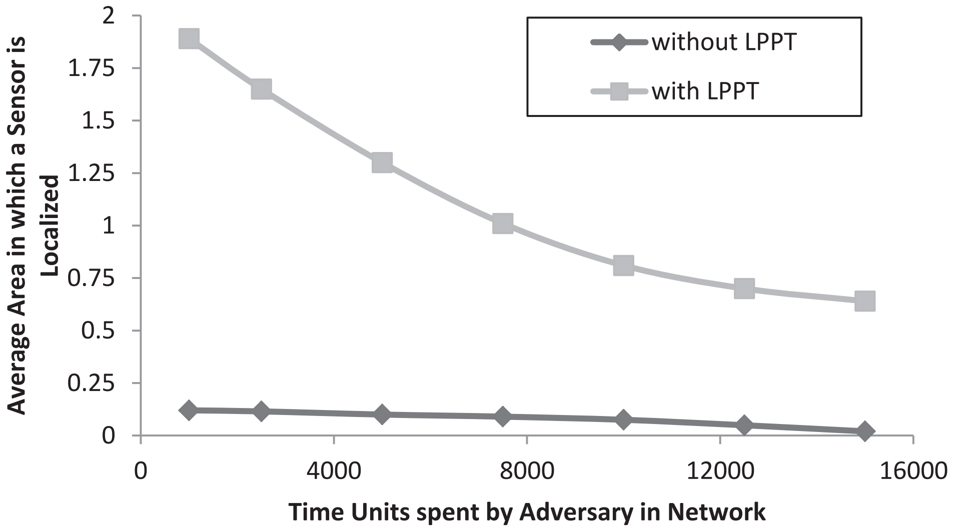


Figure 2.14. Sensor location leakage Vs. varying times spent by adversary in the network

Evaluating Energy Efficiency : We study energy efficiency as an orthogonal feature of the proposed $LPPT$ protocol in Figure 2.15 and Figure 2.16. Figure 2.15 shows that the number of messages spent per detected grid switch by adversary remains almost constant over varying adversary movement duration. The strength of the proposed method is that it prevents the adversary from obtaining more location information of sensors by simply spending more time in the network. The steadiness in the curve over time clearly depicts this idea.

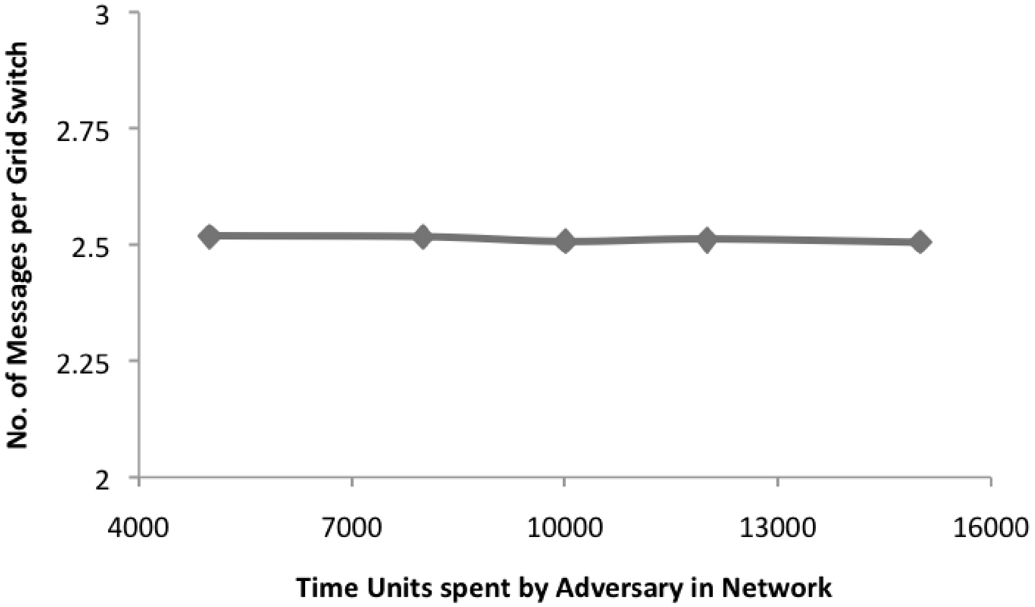


Figure 2.15. Messages per grid switch Vs. time units spent by adversary in the network

Figure 2.16 on the other hand, demonstrates that with increasing density of the network the average messages to detect each grid switch decreases, thereby enhancing energy efficiency of the proposed protocol.

In Figure 2.17 and Figure 2.18, we study energy efficiency with respect to varying network parameters, for adversary localization with and without $LPPT$. Figure 2.17 shows that for varying time units spent by the adversary, the total number of messages sent by all sensors remains very low with $LPPT$, compared to the naive protocol where the number of messages increases linearly. Throughout this set of simulations, the sensor network density has been kept constant at 10 sensors per grid.

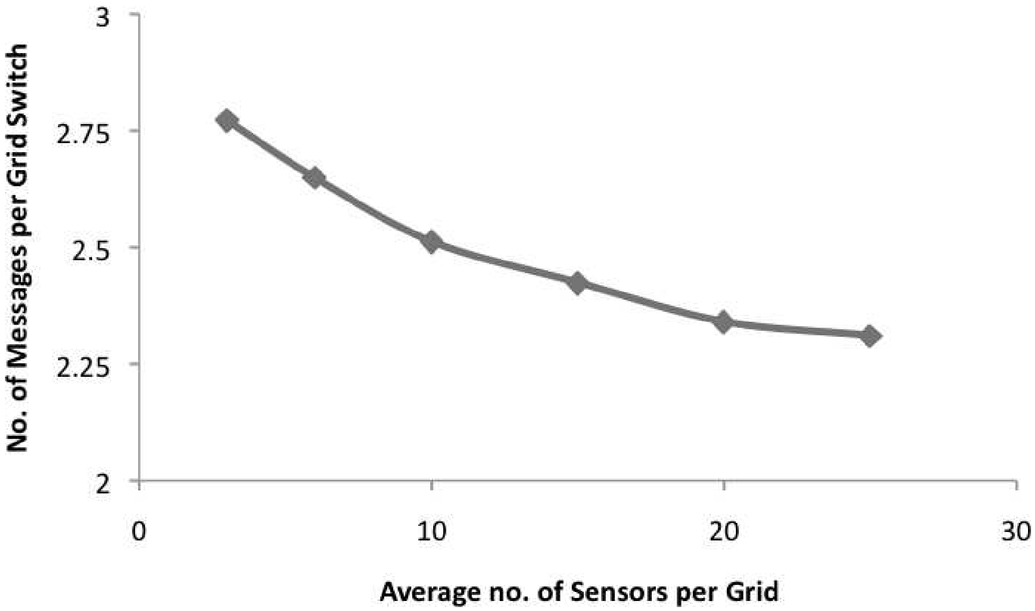


Figure 2.16. Messages per grid switch Vs. Density of sensors in the network

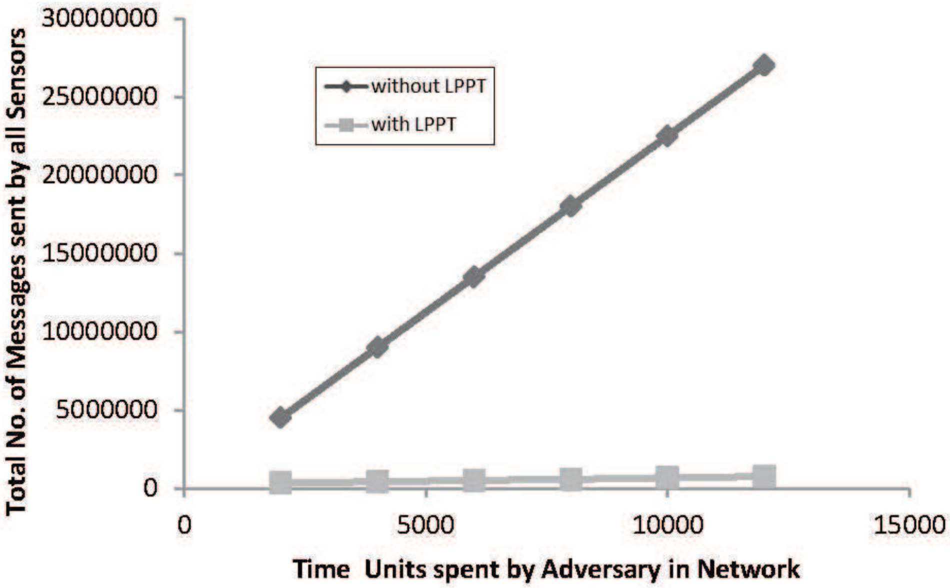


Figure 2.17. Total number of messages sent Vs. Time spent by adversary in the network

Figure 2.18 also shows that for varying density of sensors in the network, the total number of messages sent by all sensors remains very low with $LPPT$, compared to the naive protocol where the number of messages increases linearly. Throughout this set of simulations, the time spent by adversary in the network has been kept constant at $12,000$ time units.

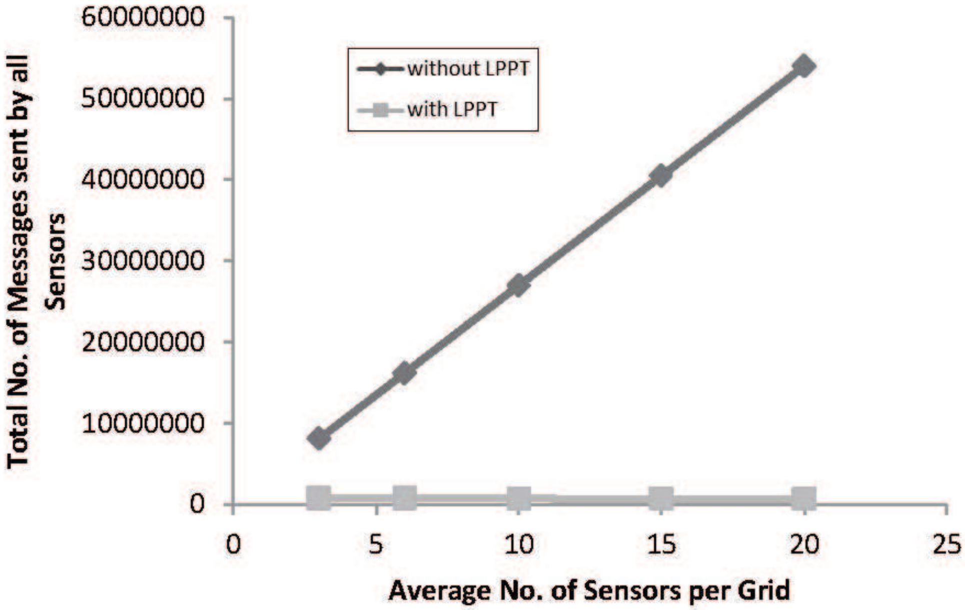


Figure 2.18. Total number of messages sent Vs. Average number of sensors per grid

2.2.6. Final Remarks. We addressed an important problem, namely the defense of sensor networks against adversarial localization. The problem spanned the three critical dimensions of target tracking, sensor localization and privacy in sensor networks, which to the best of our knowledge is unique. The principle of our defense protocol, $LPPT$ is to allow sensors to intelligently predict their own importance as a measure of the two conflicting requirements: adversary localization and sensor location privacy. Only a few such important sensors will participate in any message transmission. This ensures high degree of adversary localization, while also protecting location privacy of many sensors. However, there still remains scope of improvement for this work. The results presented in this dissertation are obtained from a C++ simulation developed to evaluate the proposed solution. We show that if $LPPT$ is used by the nodes in the network, high degree of adversary localization accuracy is achieved while preserving location privacy of a large portion of the nodes.

3. END-TO-END SECURE COMMUNICATION IN RANDOMLY DEPLOYED WIRELESS SENSOR NETWORKS

In randomly deployed wireless sensor networks, one of the most fundamental challenges comes from lack of control where sensors are located in the network after deployment. Particularly, under larger scale deployments, pre-establishing neighbor proximity information is not feasible leading to the impossibility of pre-fixing pairwise keys between sensors. Beyond this challenge to securing communications in wireless sensor networks, energy limitations of sensor nodes clearly imply that complex cryptographic operations like public key based schemes are harder to implement in wireless sensor networks. Finally, while existing work focuses primarily on securing node-to-node communications, the issues of end-to-end secure communications (i.e., between a node to base station) is mostly ignored, especially considering the significant location disparities between nodes and the base station in large scale sensor networks.

In this section, we design an end to end secure communication protocol WSNs taking into consideration the location disparity issues arising from random deployment of sensor nodes. Specifically, our protocol is based on a methodology called differentiated key pre-distribution. The core idea is to distribute different number of keys to different sensors to enhance the resilience of certain links. This feature is leveraged during routing, where nodes route through those links with higher resilience. The work presented in this section has been published in [65].

3.1. BACKGROUND AND RELATED WORK

In order to provide secure communications between neighboring nodes in randomly deployed WSNs, Random key pre-distribution (*RKP*) was proposed [56]. In its basic version, each sensor is pre-distributed with k distinct keys randomly chosen from a large pool of K keys. After deployment, neighboring nodes use these pre-distributed keys to establish a pairwise key between them. Communications between neighboring sensors in each hop are encrypted/decrypted using these pairwise keys. Many key management protocols have been proposed based on key pre-distribution [23], [39], [94], [95], [103], [174], [112] etc., mostly improving upon one or more features of [56]. Cryptographic solutions for secure communication is proposed in some existing work [25], [123].

Attack Models: In the standard attack model used in secure communications in WSNs [23], [56], [174], etc., the attacker launches two types of attacks. In *node capturing* attack, the attacker physically captures a certain percent of sensor nodes, and is able to disclose the pre-distributed and pairwise keys stored in those captured nodes. The sink node is assumed to be well protected and cannot be captured. In *link monitoring* attack, the attacker monitors all wireless links after deployment. Clearly, all communications of captured nodes are deciphered by the attacker. Furthermore, by combining the disclosed pre-distributed keys and messages recorded, the attacker can infer some pairwise keys between other nodes that are not captured. The attack model used in our work is one where the attacker launches both *node capturing* and *link monitoring* attacks.

The resilience of each hop (link) can be reflected by the number of shared pre-distributed keys in the link. It is known that under uniform key distribution, i.e. each sensor pre-distributed with equal number of keys, will achieve maximum average number of shared pre-distributed keys in each link. However, there is an inherent limitation in uniform key distribution as demonstrated in Figure 3.1. In Figure 3.1, we have 1000 nodes randomly deployed in a circular network with radius 500 $meters$, where $k = 40$, $K = 10000$ and communication range of each node is 100 $meters$. We can see that a majority of links have low resilience (i.e., small number of shared keys), while the percentage of links that are highly resilient is quite

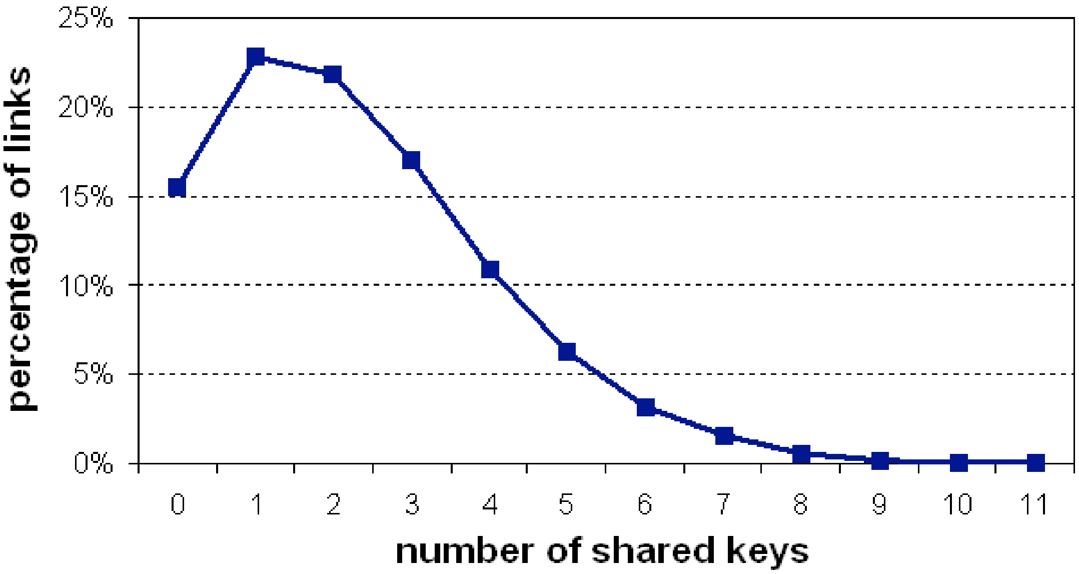


Figure 3.1. Percentage of links with varying number of shared keys.

low. This clearly restricts the room for routing protocols to choose more resilient links during end to end communications. Installing more keys into each node is not always preferable since it enables the attacker to disclose more keys upon node captures, which could again compromise the link resilience. Table 3.1 shows that using the proposed differentiated key pre-distribution, number of links with high resilience has been increased than random key pre-distribution, demonstrating the potential effectiveness of the proposed approach in secure communication.

In this section, we design a scheme based on differentiated key pre-distribution among sensor nodes that significantly improves resilience among sensor nodes under key capture, and where routing is adapted towards those links that are more secure.

3.2. PROBLEM DEFINITION

We will now introduce our differentiated key pre-distribution methodology. In order to provide a high quality of end to end secure communications, it is clear that

Table 3.1. Increase of the number of Links with Different number of Shared Keys under Differentiated Key Pre-distribution

# of shared keys	0	1	2	3	4
# of links increase	54%	−8%	−20%	−29%	−19%
# of shared keys	5	6	7	8	>8
# of links increase	−2%	25%	56%	183%	475%

we should enhance the resilience of individual links in the network. An intuitive way to do so is to increase the number of keys pre-distributed into each node (k). When the number of shared keys in each link increases, resilience seems to increase since all shared keys have to be disclosed to compromise the link.

However, such a solution is counter-productive. When k increases, more keys are disclosed per node capture. The compromise of only a small percent of nodes can disclose many more keys to the attacker, which compromises the resilience of links. We need an approach by which link resilience can be enhanced without the downside of disclosing more keys to the attacker. On the other side, the number of pre-distributed keys (k) is also subject to the memory constraint of sensor nodes.

In this section, we propose a methodology called *differentiated key pre-distribution* to enhance the quality of end to end secure communications in randomly deployed WSNs. Our methodology is based on the observation that links in the network are not equally important with respect to secure communications. Only the links used for data transmission have impacts on security. The core idea of our methodology is to pre-distribute different number of keys to different nodes. We keep the average number of keys per node the same as that in uniform key pre-distribution, so that the attacker impact (e.g., average number of keys disclosed per node capture) remains the same. By distributing more keys to some nodes, the links between those nodes tend to have much higher resilience than the link resilience under uniform key pre-distribution. These high resilient links are preferred during routing to enhance the end to end secure communications.

Use of this methodology to provide end to end secure communications between sensor nodes and the sink in randomly deployed WSNs, raises the following important questions:

- *How to determine the parameters in key pre-distribution?* We need to determine the number of node classes, the number of nodes in each class, and the number of keys distributed into nodes in each class. Determining the optimal values of these parameters needs a rigorous derivation of end to end secure communication performance.

- *How to pre-distribute different number of keys into different classes of nodes?* An intuitive way is always choosing keys randomly from the key pool regardless of node class. Is there any better way to achieve higher resilience?

- *How to perform routing given links have different resilience?* In this situation, the length of routing path and energy balancing are not the only factors to consider during routing path selection. Link resilience also plays a role. Care should be taken to make a good balance among these factors.

Based on these objectives, we design the proposed solution to the problem discussed.

3.3. PROPOSED SOLUTION

We now present our end to end secure communication protocol based on the methodology above. Our protocol consists of two components: differentiated key management and resilience aware routing. Table 3.2 lists the parameters in protocol description and their notations.

Table 3.2. Protocol Parameters

Notation	Protocol parameter
S	network area ($= \pi R^2$)
r	communication range
N	number of nodes in the network
c	number of node classes
n_i	number of class i nodes ($1 \leq i \leq c$)
k_i	number of keys pre-distributed in class i node ($1 \leq i \leq c$)
K	number of keys in key pool
N_c	number of captured nodes

Differentiated Key Management: The proposed differentiated key management consists of two stages: *key pre-distribution* and *pairwise key establishment*. The main difference between the proposed key management protocol and traditional RKP based key management protocols lies in the stage of key pre-distribution.

3.3.1. Key Pre-distribution. We study a network with N sensor nodes and one sink node. The N sensor nodes are divided into c classes, each of which has n_i ($1 \leq i \leq c$) nodes. We call the sensors in the i^{th} class as class i nodes. We then pre-distribute k_i ($1 \leq i \leq c$ and $k_1 \geq k_2 \geq \cdots \geq k_c$) unique keys chosen from a large key pool with size K into each class i node, detail of which will be discussed subsequently in this section. Note that, the sink node is pre-distributed with all K keys in the key pool. After this, the sink is deployed strategically at certain position, while the N sensor nodes are deployed randomly in the network. The N sensor nodes will execute the following protocols for pairwise key establishment and routing.

We detail our key pre-distribution in the following. For each class 1 node, its k_1 unique keys are chosen randomly from the key pool. However we use a semi-random way to distribute keys into all other nodes to increase the chance of key sharing between these nodes and class 1 nodes. We define $\lfloor x \rfloor$ as the largest integer no more than x, and define $\lceil x \rceil$ as the smallest integer no less than x. For a node in class i

$(i > 1)$, $\lfloor k_i/n_1 \rfloor$ keys are first chosen randomly from the distributed keys in each of $n_1 - (k_i - \lfloor k_i/n_1 \rfloor \cdot n_1)$ class 1 nodes, which are chosen randomly from all n_1 class 1 nodes. For the remaining $k_i - \lfloor k_i/n_1 \rfloor \cdot n_1$ class 1 nodes, $\lceil k_i/n_1 \rceil$ keys are chosen randomly from the distributed keys of each node. If some of the chosen keys are the same, the redundant keys will be re-chosen until all k_i keys are distinct.

We illustrate with a simple example. Let $N = 100$, $c = 2$, $n_1 = 20$, $n_2 = 80$, $k_1 = 80$, $k_2 = 30$. The following is the key pre-distribution procedure. For each of the 20 nodes in class 1, we choose 80 distinct keys randomly from the key pool. For each of the 80 nodes in class 2, we choose 30 distinct keys as follows. We first randomly classify class 1 nodes into two types. Type A has 10 (i.e., $n_1 - (k_2 - \lfloor k_2/n_1 \rfloor \cdot n_1)$) nodes, and Type B has 10 (i.e., $k_2 - \lfloor k_2/n_1 \rfloor \cdot n_1$) nodes. Now, $\lfloor k_2/n_1 \rfloor = 1$ key is chosen randomly from the pre-distributed keys in each of the 10 Type A class 1 nodes above, and is distributed into the class 2 nodes under discussion. Then, $\lceil k_2/n_1 \rceil = 2$ keys are chosen randomly from the pre-distributed keys in each of the 10 Type B class 1 nodes above, and are distributed into the class 2 nodes under discussion. At this point, the class 2 node has 30 keys distributed. If these 30 keys are unique, key distribution is over. Otherwise, we redo the preceding 2 steps for the duplicate keys until all 30 keys are unique. Note that since $k_1 > k_2$, uniqueness can always be guaranteed.

By distributing keys in this way, we guarantee k_i unique keys are distributed, and the number of keys chosen from each class 1 node are balanced and differs by at most 1. The reason we pre-distribute keys for class i nodes in the above semi-random way instead of purely randomly is two folded. First, we can enhance the probability that a class i node shares key with a class 1 node. Second, we do not decrease the probability that a class i node shares key with a non-class 1 node. Both facts are confirmed by our simulation, and can help increase link resilience. Besides, pre-distributing keys for non-class 1 nodes in the above way will not decrease the *effective key space* much. *Effective key space* is defined as the number of keys in the key pool that are distributed in at least one sensor node. This is because when the values of n_1, k_1 and K are carefully chosen, the number of unique pre-distributed keys among the n_1 class 1 nodes is already close to K.

3.3.2. Pairwise Key Establishment. Once nodes are pre-distributed with keys and deployed, they start to discover their neighbors within their communication range r via local communication, and obtain the key IDs of their neighbors' pre-distributed keys. With the above information, each node constructs all the one-hop and two-hop key paths to all its neighbors. If node i shares pre-distributed keys with a neighbor j, there is one direct key path with one hop between them. However, node i will also construct all the two-hop key paths with each of its neighbors, regardless of whether a one-hop key path has been constructed or not, to enhance the link resilience (the attacker has to compromise all key paths for a link between two nodes in order to compromise this link). Suppose node i wants to construct all two-hop key paths with node j now. To do so, node i sends a request to its neighbors, containing the node IDs of i and j. After a neighboring node m receives the request, it checks if it shares pre-distributed keys with node i and shares pre-distributed keys with node j. If both conditions are satisfied, node m sends a reply back to node i. In this way, a two-hop key path $i - m - j$ is constructed. If possible, other two-hop key paths are also constructed as above. After node i constructs all two-hop key paths to node j, node i will generate multiple random key shares, and transmit each key share on each key path. Key shares are encrypted/decrypted hop by hop by a combination (e.g., XOR) of all shared keys on that hop. Ultimately, the pairwise key between nodes i and j is a combination of all the key shares (e.g, XOR) transmitted. Nodes also estimate and store the number of protection keys for each link as follows. Assume there are p two-hop key paths between i and j, each with the help of proxy

s_l $(1 \leq l \leq p)$, and denote $k(i,j)$ as the number of shared keys between i and j. The number of protection keys between i and j $(key(i,j))$ is,

$$key(i,j) = k(i,j) + \sum_{l=1}^{p} min(k(i,s_l), k(s_l,j)). \tag{13}$$

We calculate $key(i,j)$ like this because the resilience of a two-hop key path is mainly decided by the weaker link (the one with fewer shared keys). The larger the number of protection keys for a link, the more resilient is the link in general.

3.3.3. Resilience Aware Routing. In this section, we will describe how to incorporate our differentiated key pre-distribution with popular WSN routing protocols for end to end secure communications. We particularly focus on one popular location centric routing protocol and one popular data centric routing protocol. Incorporation with other routing paradigms is similar. The basic idea is to *tune* the routing protocols such that they consider link resilience as a metric during routing. In order to prevent overuse of a few nodes, we will let nodes choose several next hop nodes, and use one at each time to prolong network lifetime.

- Extensions to location centric routing protocol: The location centric routing protocol we extend is GPSR [84]. In GPSR, each node chooses a neighbor as the next hop that is closest to the sink. In order to achieve high end to end secure communications without compromising network lifetime, we extend GPSR protocol as follows. Each node i assigns a weight to all its secure neighbors (neighbors with which a pairwise key is established) that are closer to the sink than itself. We denote $U(i)$ as the set of node i's secure neighbors that are closer to the sink than itself, and recall $key(i,j)$ is the number of protection keys for the link between nodes i and j. We assign weight to each node j in set $U(i)$ as,

$$w_j = \frac{key(i,j)^{\alpha}}{\sum_{m \in U(i)} key(i,m)^{\alpha}}. \tag{14}$$

Here w_j is the probability that i chooses j as the forwarder. When $\alpha = 0$, all nodes in $U(i)$ are given equal priority regardless of link resilience. When α is positive, more resilient links are given higher priority. When α approaches infinity, only the most resilient links are chosen for routing. An intermediate value of α can be used to achieve a good balance between security and lifetime, which can be decided by security policy and other factors. For example, a large value of α can be chosen when high resilience is preferred and energy consumption imbalance is not a serious issue, while a small value of α can be chosen when energy is limited and energy consumption balance is critical. We will study the sensitivity of security and lifetime to α in Section 3.4.

- Extensions to data centric routing protocol: In traditional minimum hop routing protocol [143], a variant of Directed Diffusion routing protocol, a node will choose a neighbor on the minimum hop path to the sink. We can extend this protocol in a similar way as above. During the next hop determination process, packets are forwarded only on the minimum hop secure paths. A secure path consists of links that have pairwise keys established. We denote the set of neighbors on the minimum hop secure path of node i by $U(i)$. Note that in a relatively dense network, there could be several minimum hop secure paths between node i and the sink. Node i then assigns a weight w_j to each of its secure neighbors j in the set $U(i)$. The expression of w_j is given in (2).

Remarks: In the following, we will discuss the issues of empty set $U(i)$ (discussed in Section 3.3.3), possibility of longer hops, extending our solutions to hierarchical networks, and the possibility of applying public key cryptography.

In extending GPSR, a node i may find that its set $U(i)$ is empty. In such case, node i can follow the right hand rule in [84] to choose a secure neighbor that is further away from the sink than i itself. If node i does not to have any secure neighbor, it may increase its communication range to find some secure neighbors. Applying such rules will eliminate loops and guarantee finding a secure path if it exists. Increasing communication range for more secure neighbors works for the extended minimum hop protocol as well.

We point out that the number of path hops in our schemes could be larger than that in traditional GPSR or minimum hop routing schemes. This is because in our schemes, nodes choose neighbors considering both path length and link resilience, and thus could choose neighbors on a path with more hops. Besides, as mentioned above, a node may choose a secure neighbor that is further away from the sink than itself. Intuitively, a path with more hops tends to decrease path resilience as the chance of attacker compromising at least one hop is increased. However, in our schemes, the path resilience improvement via choosing highly resilient links overwhelms the negative effect of a little longer paths of a small percentage of nodes. Overall, the path resilience will be improved.

In this work, we have focused on flat topologies. In some situations nodes could be deployed in hierarchies. The end to end routing here occurs in more than one plane, i.e., sensor to cluster head via multiple sensors, and cluster head to sink via multiple cluster heads. Our methodology and protocols are applicable in hierarchical networks. Cluster heads are chosen as class 1 nodes (provisioned with more keys), while other sensors are chosen as class 2 nodes, class 3 nodes and so on depending on number of hierarchy levels.

3.4. PERFORMANCE EVALUATION

In this section, we present performance evaluation using the data obtained from the self-made simulator. We first describe our simulation setup, and then report performance data and our observations.

3.4.1. Simulation Setup.

We conduct our simulation using a self-made simulator in C. The network is circular with radius 500 $meters$, where 1000 nodes are uniformly deployed at random. The sink is at the center of the network. Unless otherwise specified, the default parameters are: $c = 2$, $n_1 = 200$, $n_2 = 800$, $k_1 = 80$, $k_2 = 30$, $k = 40$, $K = 10000$, $r = 100$ $meters$, $\alpha = 1$ and $N_c = 50$. The default values of k_1, k_2 and k are chosen such that $k_1 n_1/(n_1 + n_2) + k_2 n_2/(n_1 + n_2) = k$, which means the average number of keys disclosed to the attacker is the same in our differentiated key pre-distribution and the original RKP scheme for the same number of captured nodes. Our communication model is one where sensors periodically transmit data to the sink. In the legend in all figures, *our GPSR* and *our minhop* refer to our protocols extending GPSR [84] and minimum hop [143] routing presented in Section 3.3.3 respectively. The legends *GPSR* and *minhop* refer to the traditional GPSR and minimum hop routing protocols following the uniform key pre-distribution respectively. Each point in the simulation data is the average of 100 runs based on independent random seeds, ensuring that the data presented and analyzed is free from any bias.

3.4.2. Simulation Results.

The simulation results are plotted and analyzed to evaluate the performance of the proposed scheme under different network parameters.

Sensitivity of P_{e2e} to Attack Intensity: In Figure 3.2, we first compare our differentiated key pre-distribution with the traditional uniform key pre-distribution (for both GPSR and minimum hop routing protocols) under different number of captured nodes N_c. We find that while the performance of all schemes degrades with increasing N_c, our schemes are consistently better than those of traditional schemes. We also find that the improvement increases with larger values of N_c. This is because when the attacker captures more nodes, the resilience of highly resilient links in our schemes degrades at a much slower pace than those of the less resilient links in traditional schemes. Besides, we can also observe that the end to end security under minimum hop based protocols is better than their GPSR counterparts. This is because minimum hop based protocols always choose the path with minimum hops, while the GPSR based protocols may choose longer paths, which compromises end to end resilience. The cost though is the increased initial energy consumption in query flooding.

Sensitivity of P_{e2e} to Network Density: In Figure 3.3, we compare our schemes and traditional schemes under different communication range r, which in turn corresponds to different network density (i.e., number of neighbors per node). When r is small, P_{e2e} is low due to both low connectivity (many nodes cannot find secure neighbors) and low resilience (fewer proxies resulting in fewer key paths for each link). When r increases, P_{e2e} increases correspondingly. For all values of r, our scheme performs consistently better.

Sensitivity of Network Lifetime to Parameter α: Recall from Section 3.3.3 that α is the *knob* that trades-off security with lifetime. In Figure 3.4, we compare our schemes and traditional schemes for varying α. We define network lifetime as the time until when the first node has used all its energy. Since traditional schemes do not have weight assignment, they are insensitive to α. The lifetime in our schemes decreases with larger values of α. This is because a larger value of α means more

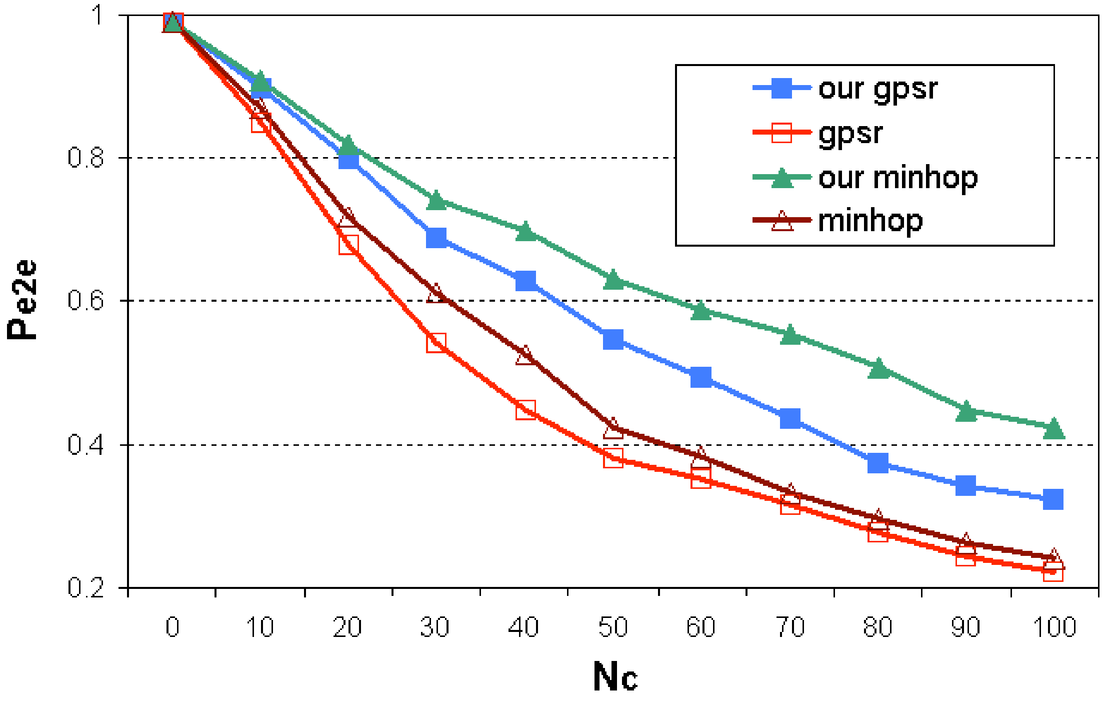


Figure 3.2. Sensitivity of P_{e2e} to number of captured nodes N_c.

priority is given to links with high resilience, thereby draining the corresponding neighbors more rapidly.

We also observe that the extended GPSR has higher lifetime compared with extended minimum hop for smaller values of α, and the difference diminishes as α increases. This is because for smaller values of α, lifetime is mainly decided by total number of candidate forwarders of each node. In extended GPSR, each node usually can find more forwarders (secure neighbors closer to sink) than it can find in extended minimum hop protocol (secure neighbors on minimum hop secure path). When α increases, lifetime is mainly decided by the number of most secure neighbors of each node. This number is similar for both protocols, and hence they have similar lifetimes when α increases. We also observe that lifetime of traditional GPSR scheme

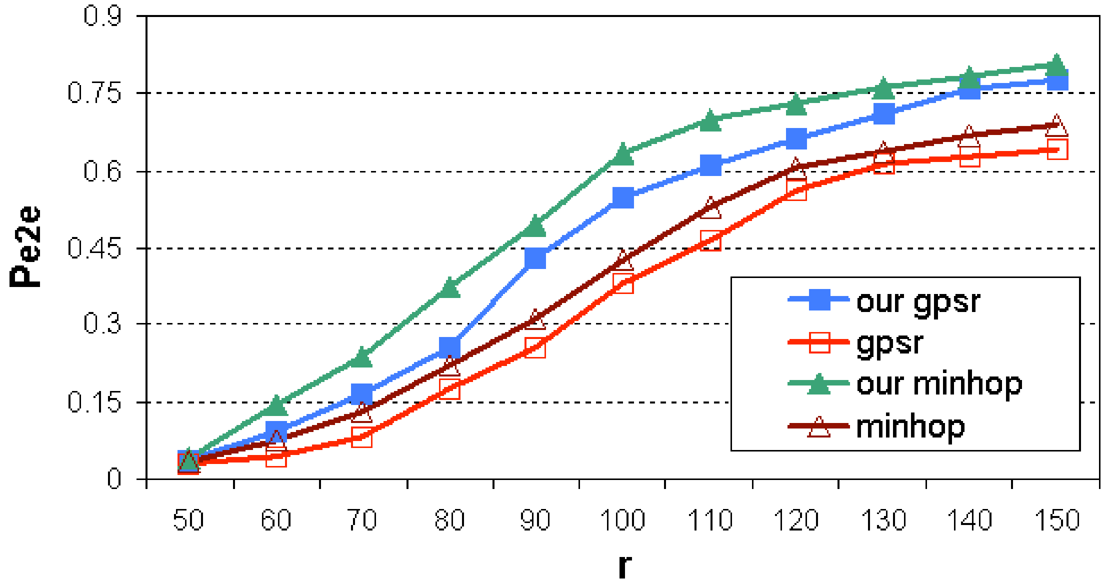


Figure 3.3. Sensitivity of P_{e2e} to communication range r.

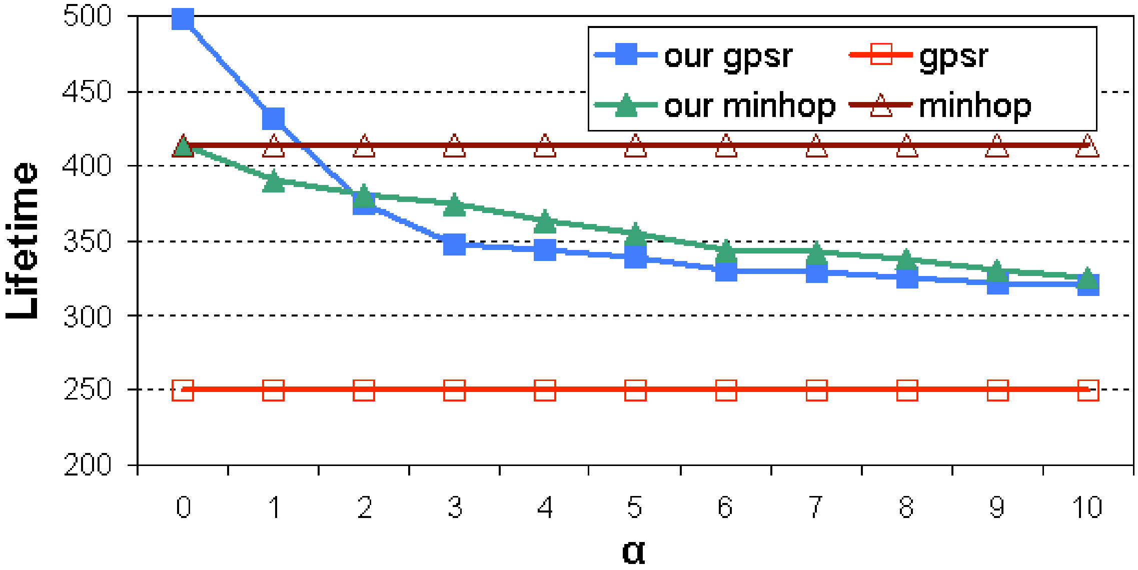


Figure 3.4. Sensitivity of lifetime to parameter α.

is lower than that of traditional minimum hop scheme. This is because in traditional GPSR scheme, some nodes are so positioned that most of their nearby nodes will choose them as forwarders, which results in their energy being drained quickly. While in traditional minimum hop scheme, nodes are less likely to be the only one on the minimum hop path of most of their neighbors, and thus traffic is more balanced.

Sensitivity of P_{e2e} and Network Lifetime to Number of Class 1 Nodes: In Figure 3.5 and Figure 3.6, we compare the traditional schemes, our schemes with default parameters, and our schemes with optimal parameters. The average number of keys pre-distributed per node is the same across all schemes for fairness of comparison. In Figure 3.5, we find that traditional schemes are insensitive to n_1 since all nodes are given same number of keys. Our schemes achieve much better performance under intermediate values of n_1, while the performance of our schemes is close to that of traditional schemes for very small and very large values of n_1. This is because when n_1 approaches 0 or 1000, all nodes will be given same number of keys, and thus our schemes degrade to traditional schemes.

In Figure 3.6, we also observe that lifetime of the traditional schemes is insensitive to n_1 due to the same reason as above. The lifetime of our schemes increases with the value of n_1. The case when $n_1 = 0$ can be treated as the same as $n_1 = 1000$. This is because for small values of n_1, the class 1 nodes are given many keys initially, and so they tend to be used as forwarders much more frequently and the lifetime tends to be small. When n_1 increases, the number of keys given to class 1 nodes decreases, thus helping to distribute the load more evenly and improve network lifetime.

3.5. FINAL REMARKS

In this section, we address the issue of providing end to end secure communications in randomly deployed wireless sensor networks addressing the challenges emanating from random locations of sensor nodes and sinks. We propose differentiated key pre-distribution, where the idea is to distribute different number of keys to different sensors to enhance the resilience of certain links in the network. This

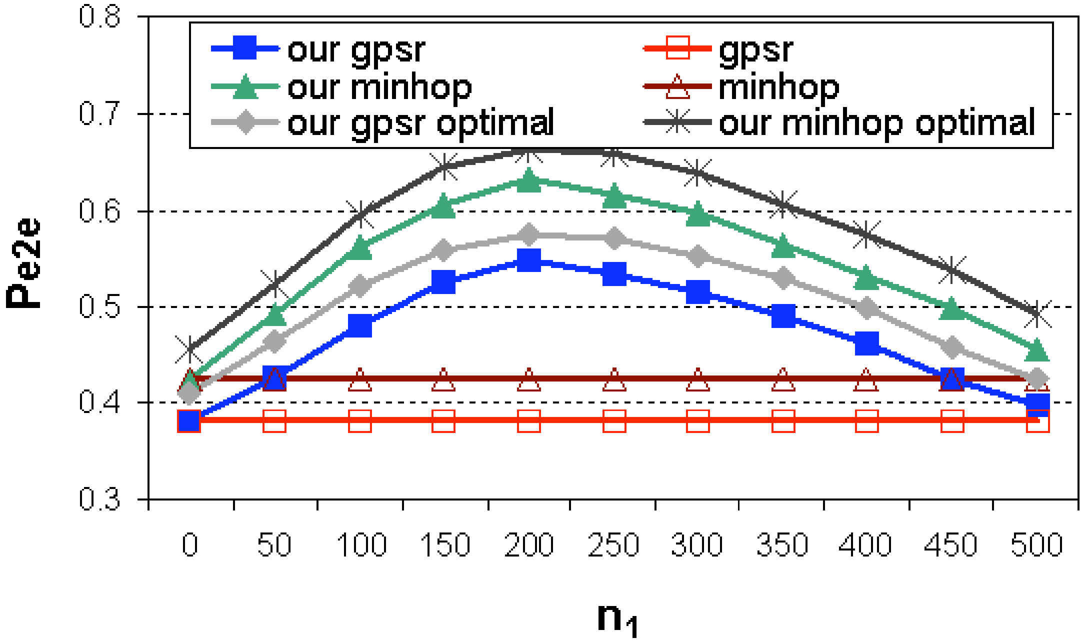


Figure 3.5. Sensitivity of P_{e2e} to number of class 1 nodes n_1.

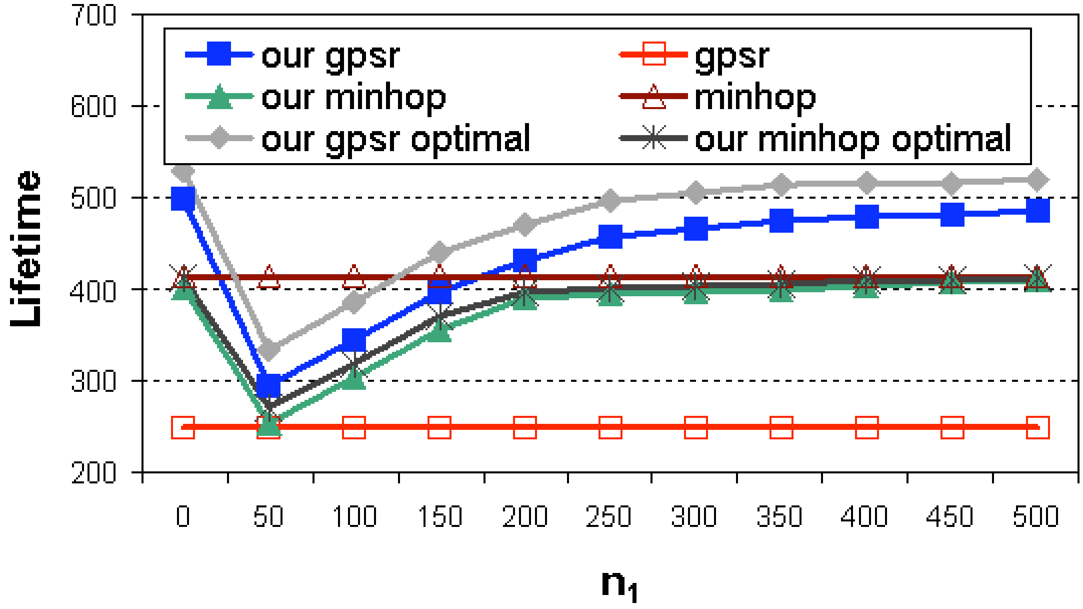


Figure 3.6. Sensitivity of lifetime to number of class 1 nodes n_1.

feature is leveraged during routing, where nodes route through links with higher resilience. We present our end to end secure communication protocol based on the above methodology by extending well known location centric (GPSR) and data centric (minimum hop) routing protocols. It can be noted that secure end-to-end communication among multiple collaborative sensor networks is a contemporary research problem. As the application and importance of WSNs extend over cutting edge technological advances, collaborative sensor networks are gaining potential in cloud computing, target tracking, secure communication and various other research areas. Network resource sharing and load-balancing are among the main advantages of collaborative sensor networks. Collaborative sensor networks are an emergent application in sensor clouds. Our approach can be extended to address secure communication in collaborative networks too. Each of the sensor networks can have their individual collection of secure communication keys distributed randomly among them. Intuitively, there is a positive probability of any two of the sensor networks sharing a few common keys, which they can use for secure intra-network communication. But as this number of shared keys increase, the communication within networks is facilitated although resilience of the link reduces and vice versa. Clearly, there is a trade-off between resilience and intra-network communication. The proposed method can be redesigned to address this trade-off and use as a solution for secure intra-network communication for collaborative WSNs.

4. QUALITY VS. LATENCY TRADE-OFF IN CONTENT RETRIEVAL UNDER AD HOC NODE MOBILITY

In this section, we address the issue of content retrieval in Mobile Ad hoc Networks (MANETs). In MANETs, the rapid mobility of nodes warrants the need of reducing search latency during peer-to-peer searches for query-driven content retrieval, so that response can be routed back to the source before it changes location. This implies that the fundamental trade-off between accurate searches for queries and associated latencies should be addressed in content management application under ad hoc node mobility. In this section, we investigate this quality versus latency trade-off in peer-to-peer searches for content retrieval in MANETs or general mobile P2P networks. We use the terms mobile P2P networks and MANETs interchangeable in this section as our proposed solution can be used for either of these environments.

Content retrieval is a canonical problem in mobile P2P networks. When a query is issued by a user in a mobile P2P network, it is unlikely that content most accurately matching the query is found in the database of the local node, and the query needs to be forwarded to peer nodes. Clearly, there is a trade-off between user satisfaction (i.e., accuracy or quality [1] of content retrieved to the query issued) and overhead. Unfortunately, in existing techniques, searches for queries at each node is a best effort process, and in the worst case, the entire database has to be searched. For any query issued by a user, we have two objectives: reduce system overhead in searching for accurate content in the network, and enabling the identification and retrieval of popular content related to the query issued. We aim to accomplish both

[1] We use the term quality of response and accuracy of response interchangeably in this dissertation. Quality or accuracy of response is defined by the similarity between the query and the content. The formal definition of metrics to measure it is included in section 4.2

objectives by incorporating adaptiveness to the retrieval process from both the user side and system side.

From scalability perspective, the overhead can be tremendous when the number of queries increases. Our first motivation is to reduce the search overhead based on two observations: a) Based on past knowledge of query searches, the system itself can derive some intelligence on the expected accuracy of content available for queries issued, and use this knowledge to limit wasteful searches and hence limit the search overhead for similar queries in the future; b) In many scenarios, it is likely that users may not always desire content perfectly matching queries issued, and if users can specify this in their queries, the searching overhead can be significantly reduced.

In mobile P2P networks with multiple users, different users will share similar interests and hence issue similar queries at different points in space and time. Since a search process for any query typically involves multiple nodes, each node in the system gradually can recognize queries that are popular among users. With this knowledge, each node in the system can naturally also learn to identify popular content in its local database. For any query issued subsequently by a user, if this query is similar to popular queries serviced by the local node earlier, then the local node can quickly retrieve corresponding popular content to the user, while continuing the regular process of searching for accurate content.

Our Contributions: The contributions of this research in content retrieval in mobile P2P networks is three-fold.

- We design a Multi-Tiered architecture and a suite of protocols for content retrieval in mobile P2P networks. Tier 1 in our architecture is designed for reducing the search overhead at each node when searching for content corresponding to queries issued. The premise of our approach stems from the observation that when queries are short, the system has a much higher chance of retrieving more accurate content. When queries get longer, then the chances that highly accurate content can be found is lower. In this section, we first demonstrate how the trend of accuracy vs. query length follows the trend of a logistic function in practice. With knowledge of this trend, each node can then make intelligent choices on when to stop searching the database further when the node determines that more accurate content is unlikely to be found further. Secondly, logistic functions are governed by a parameter α that governs the rate of growth. By making this parameter user adaptive, users can also decide the desired accuracy of content requested, using which overhead in the system can be reduced during searches. In this tier, flooding based routing is used to route queries and responses with reduced search latency so that responses can be returned to the source location on a timely manner under ad hoc node mobility.

- We design a novel technique for retrieving popular content for queries issued and present it as Tier 2 in our architecture. We exploit a basic feature in mobile P2P networks for this purpose, namely the fact that multiple nodes are searched for every query issued. As such, when similar queries are issued by multiple nodes, it naturally allows popular queries to be disseminated to many more nodes in the system. It can be noted that as any node retrieves files from the network via intermediate nodes, different nodes gain knowledge about the content of other nodes. This knowledge can be used to improve the efficiency of content retrieval from peer nodes. In our Tier 2 design, we define a new metric called *Rank* for each content in its local database, where the rank for each content is computed as a function of the popularity of its keywords. We then introduce a new concept called *Chained Bloom Filter*, where popular keywords already processed by the node are linked to popular content in a space efficient manner in the content. When new queries come in, we design a protocol that efficiently allows to determine if the keywords requested in the query are popular (i.e., they are in the Filter), and if they are, it quickly returns the correspondingly linked popular files.

- We conduct detailed theoretical analysis and simulations to analyze the performance of our proposed techniques. Our analysis demonstrate that the accuracy of content retrieved does follow a logistic trend that can be captured with a parameter α. We also show that by allowing this parameter to be user adaptive, the search overhead during the retrieval process dramatically reduces. Our analysis also demonstrate that the proposed rank and Chained Bloom Filter techniques are effective in both determination of popular content, and also for quick retrieval during subsequent searches.

The work presented in this section is published in [44].

4.1. BACKGROUND AND RELATED WORK

Mobile peer-to-peer networks has been an important area of research in the past several years. Within the realm of Mobile P2P networks, there are several interesting areas of research like content retrieval [29], data dissemination [127], aggregation [11], routing [86, 82], security and privacy [109], [119] etc. In this section, we provide a brief overview of important work related to the contributions of this work.

Query driven content retrieval is a problem that has received significant attention recently in the mobile P2P community. In [29], the authors present a content retrieval scheme for mobile P2P networks. Their scheme reduces communication cost and energy consumption using intelligent query routing. In the proposed method, a node gathers information about the possible location of a required data from its neighborhood. After evaluating the information obtained from neighbors, the node finds out another node which has more likelihood of retrieving the content from its neighborhood. This technique unlike ours does not address the issue of popularity of keywords and content In [58], a content retrieval scheme called Eureka is proposed. The mobile nodes estimate the information density of data in neighboring nodes and forwards the query towards the nodes with high density of requested data. A similar approach is proposed in [137], where adaptive content synopses dissemination strategy is content retrieval for content retrieval in peer to peer environment. They propose a bloom-filter based solution to dynamically update neighboring nodes about the synopsis of the content possessed by its neighbors as nodes keep joining and leaving the network. Both these techniques have high communication overhead due to frequent updates needed for data density estimation and synopses sharing, and also do not address popularity of content.

There have been recent efforts on data dissemination in vehicular networks, which are in a broad sense mobile P2P networks. In [168], a vehicle assisted data delivery approach is proposed to reduce delay in delivering the data. This technique is purely for routing purposes and not for query processing. In [167], a popularity aware content retrieval scheme in VANETs is proposed. The technique involves identifying popular content and replicating them in a distributed fashion at nodes in the network to increase availability. Our work addresses the problem of keyword based popularity management at local nodes to minimize overhead and bandwidth.

To summarize, content retrieval in mobile P2P networks has been well studied. The focus of this section is on provisioning adaptiveness to the retrieval process from the perspective of minimizing search overhead without significant losses in accuracy, and a keyword based popularity scheme. To the best of our knowledge, this work is the first to study the applicability of modeling the content retrieval process as a Logistic Function and exploit it to reduce search overhead. Our technique to address popularity based on knowledge of prior searches at local node significantly minimizes bandwidth and communication energy wastage, which are critical challenges in mobile P2P systems. There are some open issues still left in our scheme. One such issue is how to derive the Logistic Function for all nodes in the entire network. The training time to find optimal functions is a challenging problem that also takes a significant amount of time, and we did not address it in this section. Also, a challenge in Bloom Filters is that elements already hashed cannot be deleted easily. In our scheme, this is important, because, we would ideally want to remove stale keywords from the Chained Bloom Filter at each node. One approach we could use to address this problem is to design *Counting Bloom Filters* in Tier 2, and appropriately chain them to Popular files. Counting filters [14] basically provide a way to implement a delete operation on a Bloom filter without recreating the filter afresh. In a counting filter the array positions (buckets) are extended from being a single bit, to an n-bit counter. In fact, regular Bloom filters can be considered as counting filters with a bucket size of one bit. More details on Counting Filters can be found in [14]. We do not specifically address this issue in this section, since we believe that it is orthogonal (but still complementary) to the proposed research.

4.2. PROBLEM DEFINITION

In this section, we address the problem of content retrieval in distributed mobile P2P networks, where the nodes are willing to share data among themselves in a query driven manner. The queries we consider are keyword-based and user-generated. Each user has a limited memory to store information, known as the local database of the node of the user. The files in the database are stored against keyword-based metadata which describes the content of the file briefly. We illustrate this further using a simple example. Let us consider a mobile node whose database contains the following files described using following keywords: *i) Beatles Because Rock MP3*, *ii) Chicago Downtown Parking Coupon July 5 2010* and *iii) Bloomington Traffic Congestion Prediction July 7 2010*. A user-originated query submitted to the node is: *Chicago Downtown Average Weekday Traffic Congestion.* Although the last two files have relevance to the query, the best matching file is not available in the local node, but another node in the system has a file with exactly matching keywords to the query. So when a user at a node submits a query, other non-local nodes might have information more relevant to the query than what the local node contains. The local node can access the best matching file by searching in the non-local nodes. We soon present a more lucid example to describe the same.

Network Model: The mobile P2P network we consider in this work is one where there are a number of nodes that are moving, and able to communicate using wireless medium with each other in a local scope. The nodes can communicate with each other using short range wireless communication standards like IEEE 802.11, bluetooth etc. Peer nodes present within the range of communication of a node is known as one-hop neighbors of the later. In mobile environment, set of neighbors of a node changes over time. So connectivity between two nodes is also subject to change over time. Other nodes in the network also have files which can match the query asked. It can be noted that incoming queries from different users might have different keywords with same semantic meaning. However, this work does not address the details pertaining to the methods and challenges involving the same and assumes that different keywords has different semantic meanings. Existing techniques [158], [13] for semantic-aware content retrieval can be used to extend our work in this context. Each node has a database of content (i.e., files) [2] that are meant to be shared. Each content contains a list of metadata that describes the corresponding content. An example of a database at a node is shown in Table 4.1. Users in the system issue queries which comprise of a set of keywords. For each query issued, the P2P network searches in the local vicinity of the requesting node (or user [3]) to retrieve content that accurately matches the query. In this work, we emphasize on making the retrieval process adaptive via two critical ancillary goals:

[2] We use the term Content and File interchangeably in this section.

[3] We use the term *node* and *user* interchangeably in this section.

Table 4.1. An Example of a Database at a Local Node

File 1	Beatles	mp3	Rock	English
File 2	Elvis Presley	mp3	Summer Kisses	English
-	-	-	-	-
-	-	-	-	-
-	-	-	-	-
File F	Target	Coupon	Labor Day	

- minimize *search overhead* during the search process as a function of past knowledge of searches,
- retrieve *relevant and popular content* where the popularity is governed by prior searches for queries for users with similar interests.

We propose a novel two tier architecture for efficient content retrieval in mobile P2P networks. In this subsection, we first define the metrics used for quantifying the performance of the proposed method.

Content Similarity Metrics: One of the critical issues in content retrieval is how to determine the degree of similarity of between a query and a piece of content (or file) in a node. Towards this premise, there are quite a few number of metrics that have been addressed in literature. In this section, we provide a brief overview of some well known metrics, their properties, and our metric of choice in this work.

Consider a query identified by q, and a file identified by f, both of which have a set of keywords denoted as $\bar{q} = (q_1, q_2, q_3, \ldots, q_n)$ and $\bar{f} = (f_1, f_2, f_3, \ldots, f_n)$. Denoting $S_{q,f}$ as the degree of similarity between the Query q and File f, in the *Sorensen Similarity* metric, we have

$$S_{q,f} = \frac{|\bar{q} \cap \bar{f}|}{|\bar{q}| + |\bar{f}|}. \tag{15}$$

The *Jaccard Coefficient* extends the Sorensen similarity index by considering the union of the terms between f and q to avoid counting the common terms between them twice. It is defined as,

$$J_{q,f} = \frac{|\bar{q} \cap \bar{f}|}{|\bar{q} \cup \bar{f}|}. \tag{16}$$

Another metric that is quite popular in data similarity comparison is the *Cosine Similarity* Metric, which borrows from the Vector Space Model (VSM). Assuming that there are D files in the database of a node and each file is tagged with upto n keywords per file, we represent each file as a row in a $D \times n$ matrix. In this manner, each file is projected as a binary vector in a n-dimensional vector space. Any incoming query can also be treated as a vector in the space, and so the similarity computation between the query and a file is determined by means of computing the angle (θ) between the query vector and the file vector. More formally, for a query q comprising of keywords $\overrightarrow{q} = (q_1, q_2, q_3, \ldots, q_n)$, and a file f comprising of keywords $\overrightarrow{f} = (f_{j1}, f_{j2}, f_{j3}, \ldots, f_{jn})$, the similarity between q and f denoted as $\theta_{q,f}$ is given by

$$\begin{aligned} \theta_{q,f} &= Cos^{-1}(\frac{\overrightarrow{q} \odot \overrightarrow{f}}{|\overrightarrow{q}| \cdot |\overrightarrow{f}|}) \\ &= Cos^{-1}(\frac{\sum_{i=1}^{n} q_i f_{ji}}{\sqrt{(}\sum_{i=1}^{n} q_i^2) \times \sqrt{(}\sum_{i=1}^{n} f_{ji}^2)}). \end{aligned} \tag{17}$$

Naturally, smaller the value of $\theta_{q,f}$, more accurate is File f for Query q, and vice versa.

While all the above metrics in a broad sense do capture the similarity between a query and a file in terms of number of matches and lengths of the query and file, the

Cosine Similarity is one that is quite popular for content similarity [53, 113, 68, 167]. The main reason for this is that since the Cosine Similarity index (interpreted as an angle θ) is inspired from the well known Vector Space Model, the interpretation of the similarity is very intuitive. A Query q and a File f that contains exactly the same set of keywords yields a $\theta_{q,f} = 0°$, while $\theta_{q,f} = 90°$ when there are no matches between q and f (i.e., orthogonal vectors). When there are partial matches between q and f, the metric yields a intuitive value between 0° and 90°. In this section, we use the *Cosine Similarity* Metric as our baseline. Note however that the techniques developed in this work are applicable to other similarity metrics as well.

We would like to mention about two well-known metrics for content similarity in the field of information retrieval and pattern recognition. Those are *Precision* and *recall*. Precision is the fraction of retrieved instances that are relevant and is represented by the number of relevant documents a search retrieves divided by the total number of documents retrieved. Recall is the fraction of relevant instances that are retrieved and represented by the number of relevant documents retrieved divided by the total number of existing relevant documents that should have been retrieved. From probabilistic point of view, precision is the probability that a randomly selected retrieved document is relevant. Recall is the probability that a randomly selected relevant document is retrieved in a search. Given the nature of services in mobile P2P networks, we focus more on retrieving some relevant content with optimized delay and search overhead than retrieving more number of relevant contents.

4.3. PROPOSED SOLUTION

In this section, we detail our multi-tiered architecture for adaptive content retrieval in mobile P2P networks. Section 4.3.1 presents an overview of the proposed architecture. In Section 4.3.2, we first detail the design of Tier 1 and then Tier 2, followed by a detailed theoretical analysis of the proposed protocols across several metrics.

4.3.1. Overview. In the following, we give a overview of our proposed multi-tier architecture for content retrieval in mobile P2P systems. The proposed architecture is comprised of 2 tiers: Tier 1 for reducing search overhead during searching for accurate content retrieval, and Tier 2 for efficiently retrieving popular content in the system. Note that there are two situations under which a node (say Node A for example) receives a query to process in mobile P2P networks. Either the local user of Node A issues a query to that node, or the Node A receives a query from a neighboring node. In either case, a node receiving a query first processes the query in Tier 1, where the local database of the node is searched. The novelty of Tier 1 is in the deign of a technique and a search protocol that minimizes search overhead as a function of knowledge of prior searches such that at a slight cost on accuracy, a significant amount of search overhead can be saved during searching. The query is then processed in Tier 2, where we design a Chained Bloom Filter technique to efficiently store popular content based on processing queries for other nodes in the system. We also design protocols in Tier 2 that efficiently store and retrieve popular content stored in the Chained Bloom Filter for the queries issued. Results from both tiers are subsequently returned to the node, which then forwards the results to the upstream node or the local user depending on where the query came from.

4.3.2. Tier 1 - Reducing Search Overhead. Content in a mobile P2P network is identified by a set of metadata provided by users that create and share files. While some users can provide a large number of descriptors for content shared, others may only provide a small number of descriptors. Since the amount and the nature of metadata provided for each content varies from user to user, this negates attempts to index the database. Consequently, for any Query q arriving at a node, the worst case searching time is $O(D) \times \bar{t}$, where D denotes the number of entries in the database, and $\bar{t}$ is the processing time to find the similarity between Query q and a File f in the database. This clearly imposes a tremendous overhead for a single query, and when one considers multiple queries, then the search overhead can really impose a bottleneck in the system. Our motivation for Tier 1 is to reduce the search space overhead with minimal impacts to accuracy of retrieved content as a function of the system's prior knowledge of query searches and also the user's own preferences on accuracy of retrieved content.

For any mobile P2P system, it is natural that as the system evolves, the amount of content available at nodes increase. Due to the nature of P2P systems being ad hoc, it is also natural that the descriptors in the metadata for each content in the database also varies significantly. Orthogonally, the same can be said for queries as well. While some users might be very specific about content desired, others can be more general. The former case happens when the number of keywords requested in the query is more, and the latter happens when number of keywords is relatively smaller. Naturally, when query lengths are smaller, it possible to return more accurate content, and when queries get longer, the possibility of finding highly accurate content decreases. Formally, θ (our similarity metric) initially increases (i.e., accuracy decreases) for increasing query lengths due to a combination of decreased keyword matches between queries and files, and increase in query lengths, as can be seen in Equation 3. However, the growth in the increase of the θ metric becomes progressively slower due to the Cos^{-1} function. Based on this intuition, we conjecture that content retrieval in ad hoc environments like mobile P2P networks follows the trend of a Logistic Curve in terms of Accuracy vs. Query Length.

It is straightforward to see that, when queries are more general, it is easier to find matches compared to more specific queries. More formally, our design for Tier 1 is based on our conjecture that *Content Retrieval in ad hoc environments like mobile*

P2P networks follows the trend of a Logistic Curve in terms of Accuracy vs. Query Length. Our intuition for this trend is a follows.

When query lengths are smaller, the queries tend to be more general, and it is hence possible to return more accurate content. When queries get longer, then they tend to be more specific, which means that the possibility of finding highly accurate content decreases. More formally θ (our similarity metric) initially increases (i.e., accuracy decreases) for increasing query lengths due to a combination of decreased keyword matches between queries and files, and increase in query lengths, as can be seen in Equation 18. However, the growth in the increase of the θ metric becomes progressively slower due to the Cos^{-1} function.

We have conducted an extensive simulation study to further validate this conjecture, and results are shown in Table 4.2. In each of the cases shown, we conducted simulations to obtain the best θ values for varying query lengths via an exhaustive search of the database, and tried to fit a curve to the plot of θ vs. Query Length. Note that in Table 4.2, D is the no. of files in the database; f_l is the maximum range of the no. of keywords in each file in the database; q_l is the length of the query; and RMSE is the root mean squared error between the θ (derived via simulations), and the θ value obtained from the Logistic Function correspondingly shown in Table 4.2, which was derived via symbolic regression techniques (a special variant of Genetic Algorithms) to fit the curve. Each simulation was conducted 100 times and averaged out for curve fitting. Note that the function $L(x)$ in Table 4.2 is the standard Logistic Function $L(x) = \frac{1}{1+e^{-x}}$. As, we can see the RMSE is quite low, demonstrating that the Logistic Function complies to the trend of Accuracy (θ) vs. Query Length. For ease of comprehension, we illustrate the trend of θ obtained via simulations for the case when $D = 10000$ and f_l is from 1 to 4 (first entry in Table 4.2), and the Logistic function fitted for this case in Figure 4.1. As we can see the trend of the Logistic Function holds for our results.

To the best of our knowledge, this is the first work that identifies the logistic trend in content retrieval in P2P based networks. In Tier 1, we exploit this trend for saving overhead during content retrieval with minimal compromise to accuracy. Each node first derives the Logistic Trend within its own database. The node can do this via prior knowledge of searches, or derive an estimate via periodic random sampling of the local database. We generalize Logistic Functions derived in Table 4.2 as $LF(q_l) = \beta \times L(\alpha \times q_l)$, where q_l is the length of the incoming query. Once each node derives, parameters α and β, for any incoming query, the node will first determine the expected accuracy of search results for that length based on the length of the query q_l, and deriving $LF(q_l)$. We denote the Expected Value as $\theta^E_{q_l}$. The node will then search the database to find a matching file, and it will stop searching

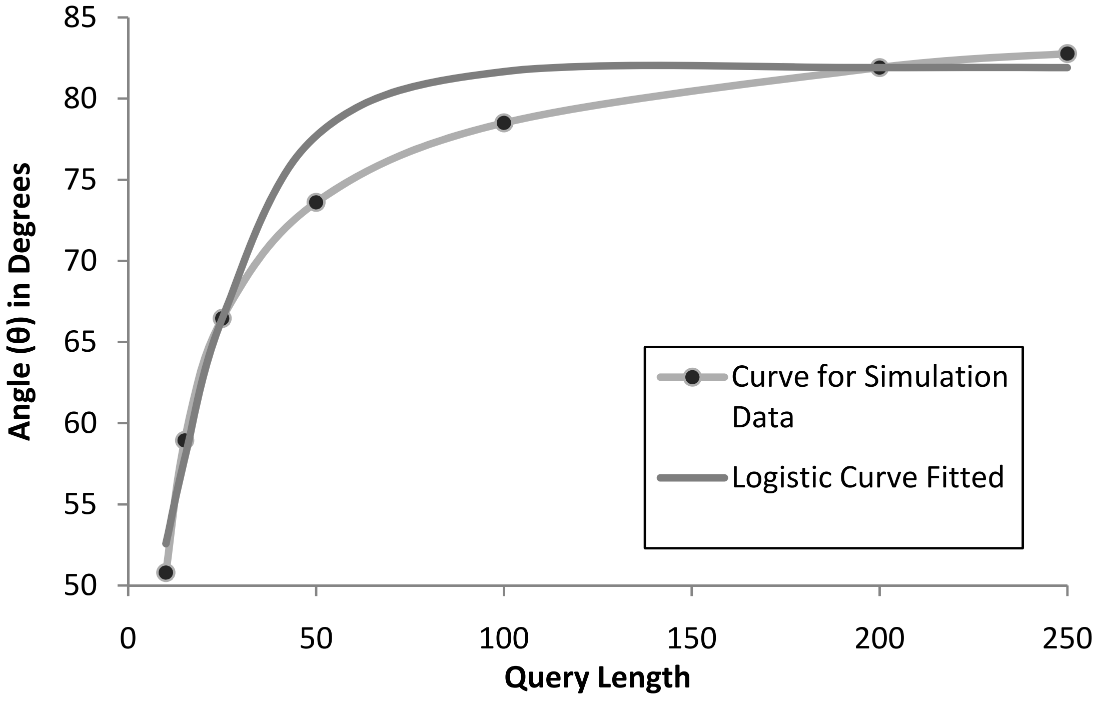


Figure 4.1. The Trend of Logistic Function

Table 4.2. The Logistic Function for Various System Parameters

Database Parameters	**Logistic Function**	**RMSE**
$D = 10000$ f_l from 1 to 4	$86.99 \times L(0.033q_l)$	4.33
$D = 12000$ f_l from 1 to 4	$86.37 \times L(0.042q_l)$	4.16
$D = 10000$ f_l from 1 to 6	$86.37 \times L(0.029q_l)$	4.57
$D = 12000$ f_l from 1 to 6	$86.38 \times L(0.03q_l)$	4.48

the database once a file is found that is less than or equal to the expected accuracy derived from the Logistic Function for that node. Algorithm 3 illustrates the workflow of the search process in Tier 1.

Algorithm 3 Executed by any Node N in its Tier 1

```
Initialization Phase
Derive Logistic Function, $\beta$ and $\alpha$ parameters
End Initialization Phase

At Run Time for Every Query $q$
Determine Query Length $q_l$; Set $\theta_{min} \leftarrow 0$
Determine $\theta^E_{q_l}$; Set $f_{best} \leftarrow$ Null
for Every File $f$ in Database do
  Compute $\theta_{q,f}$
  if $\theta_{q,f} \leq \theta_{min}$ then
    $\theta_{min} \leftarrow \theta_{q,f}$
    $f_{best} \leftarrow f$
  end if
  if $\theta_{q,f} \leq \theta^E_{q_l}$ then
    Break
  end if
end for
Return $f_{best}$
```

Discussions: A critical issue to observe in our Tier 1 design is the parameter α in the Logistic Function which decides the growth of the function. When each node sets its derived α parameter for a query, the node aims to return the expected level of service to the user. However, users can also manipulate this parameter.

When users pro-actively set higher values of α, the growth of the Logistic Function increases. This means that users for the same Query Length prefer a lower accurate file (higher θ value), which may lead to more overhead savings. Contrary to the above observation, it can be noted that users who set α to a lower value will expect more accurate results (lower θ values) resulting in more overhead in the search process. As we can see, the parameter α provides an added leverage for users to adaptively choose the desired level of content accuracy at a cost of search latency trade-off. We study this issue further using simulations in Section 4.5.

4.3.3. Tier 2 - Retrieving Popular Content. The motivation for our Tier 2 design is popularity aware content retrieval. While Tier 1 focused on retrieval of accurate content with simultaneous overhead savings, these are not the only considerations from the user side. In many P2P environments, a critical issue is popularity. Some types of queries (i.e., keywords) may be quite popular among users. Our goal in designing a content retrieval approach in mobile P2P networks is two-folded: *i) Improve accuracy of content retrieved* for the popular queries, and *ii) Retrieve popular content* in the database relevant to that query

The key principle behind our popularity scheme is to assign a metric, which we refer to as *Rank* of each file in the database. *Rank* of a file is a function of how popular the keywords in the file are in the network. Each node maintains the *Rank* for each file in its database independently. As and when keywords are serviced by a node, they are efficiently hashed into a novel *Chained Bloom Filter* (CBF), along with popular files for these keywords. When new queries comes in with keywords, these keywords are compared with those in the Chained Bloom Filter, and the corresponding popular files are returned. We discuss the details of Tier 2 subsequently in this section.

Past access frequency of a keyword: The past access frequency captures the popularity of a keyword in terms of the number of times the keyword has been queried in the recent past. However, we claim that the popularity of a keyword varies with time in a mobile environment. In other words, some keywords may have been historically popular, if they are not so popular in the recent past, it is unlikely that current users are interested in those keywords. In this work, we not only give importance to the number of times a keyword has been accessed, but also consider how recently the particular keyword was requested. To incorporate the change in popularity of a keyword, we use a *damping factor* for tuning keyword popularity, as also used in [167]. The importance of a keyword in our scheme decreases over time by a constant damping factor λ ($0 < \lambda < 1$). The past access frequency of every keyword is computed at a regular time interval of $\triangle t$. So at a time t_n the past access frequency $\mu_n(k)$ of a keyword k is defined as:

$$\begin{aligned} \mu_n(k) &= \lambda^{n-1} N_1(k) + \lambda^{n-2} N_2(k) + \dots \\ &\quad + \lambda^{n-i+1} N_{i-1}(k) + N_n(k), \end{aligned} \tag{18}$$

where $N_i(k)$ is the no. of times keyword k has been queried at the i^{th} instance within the past $\triangle t$ time interval ($1 \leq i \leq \triangle t$).

Rank of a file: We now address the issue of popularity of a file over time in a mobile environment. The objective of our *Rank* metric is to capture how popular a file is on the basis of the number of times its keywords have been accessed in past. In simple terms, a file, f, whose keywords have been queried more often recently tends to be more popular to users who issue queries with keywords common to those of f. In this context, a naive technique for rank computation of a file could be to simply sum up the past access frequencies of all keywords in that file. However, the downside is that this approach favors files with more keywords. As such, we compute the rank of a file as the average value of the access frequencies of keywords in the file. Note that the Rank computation is dynamic and is re-computed for each file in intervals of $\triangle t$. Formally, the Rank of a file f with z keywords $\{f_1, f_2, f_3, \ldots, f_z\}$ at time instant n is given by,

$$Rank_n(f) = \frac{\sum_{i=1}^{z} \mu_n(f_i)}{z}, \tag{19}$$

where $\mu_n(f_i)$ is the past access frequency of keyword f_i derived from Equation 19. Note that a high value of rank for a File f implies that the File f has atleast one keyword that has been requested very often. This means that for a Query q that comes in with keywords matching the keywords in File f, it is ideal to return File f as a popular file for Query q.

Our Chained Bloom Filter Technique: We now present our Chained Bloom Filter approach for storing and retrieving popular (highly ranked) content. Before that, we give a brief overview of Bloom Filters. The Bloom Filter is a probabilistic data structure to determine membership of an element in a set [16]. Very briefly, an empty Bloom filter is a bit array of m bits, all set to 0. We also define k hash functions, each of which maps any element in the set to one of the m array positions with a uniform random distribution. To add an element in the filter, we feed it to each of the k hash functions to get k array positions. We then set the bits at all these positions to 1. To query for an element in the set, we feed it to each of the k hash functions to then get k array positions. If any of the bits at these positions are 0, the element is not in the set, since otherwise, all the bits would have been set to 1 when it was inserted. If however all the bits are 1, then either the element is in the set, or the bits have been set to 1 during the insertion of other elements. As such while False Negatives are not possible during verification, False Positives are possible, the rate of which can be controlled by the parameters m and k. In this work, we extend the basic idea of Bloom Filters for our popular content retrieval technique in Tier 2, and term our approach as *Chained Bloom Filters.* There are two kinds of operations involved in Tier 2:

- Updating the Chained Bloom Filter,
- Retrieving content from the Chained Bloom Filter

We first discuss method to update the filter. Whenever a query comes in with keywords, then a node will hash the keywords into a regular Bloom Filter with k hash functions and m bits. The node then computes the Rank for files that have atleast one of the keywords in the query. The node determines the top x files in terms of Rank, and inserts the ids of these x files in another array of m bits at the same positions in the Bloom Filter that were set to 1. Note that each of these x files are the ones containing atleast one keyword from the incoming query and having highest ranks based on prior searches. The respective positions in both arrays are linked to each other, leading to the term *Chained Bloom Filter.* Figure 4.2 illustrates an example of a *Chained Bloom Filter* with 12 array bits, and the keyword hashes and correspondingly top ranked files linked to each other. We consider the list of file IDs chained against each bloom filter bit as a bucket for that bit position.

To summarize, our Chained Bloom Filter technique efficiently links prior keywords searched at a local node with relevant popular content in that node for subsequent retrieval. We next present the details of our scheme, followed by an analysis of the proposed scheme from the perspective of retaining popular files in the filter and the probability of returning relevant files.

We now discuss how to search the Chained Bloom Filter. For a Query q arriving in Tier 2 of a node, the node first checks if atleast one keyword in the Query q is present in the Bloom Filter. If not, then the keywords were never serviced by the node, and there is nothing to retrieve in Tier 2. Otherwise, for each position in the bloom filter where the bit is set as 1 corresponding to every keyword's hash, the

Hashed Key Words	1	0	0	1	1	0	1	0	0	0	0	1
Files Ids	f_1 f_2 f_4 f_9			f_2 f_7 f_3	f_{10} f_8 f_5		f_{11} f_4 f_8					f_{15} f_{16} f_3 f_4

Figure 4.2. A Snapshot of the Chained Bloom Filter

node retrieves the linked files linked in the corresponding bucket. The intersection, I of all the buckets is then the popular file(s) corresponding to atleast one keyword in the query (with high probability). I is returned to the node as result of the query.

Algorithm 4 Executed by any Node N in its Tier 2 for Retrieving Popular Content

```
**Input:** Query $q$ with keywords $\{q_1, q_2, q_3, \ldots, q_n\}$
**Output:** Set of Popular Files $\bar{P}$ relevant to Query $q$

Set $\bar{P} \leftarrow$ Null
Determine if all keywords in $q$ are in the Bloom Filter
if None of keywords are present then
    Break
end if
for Each Keyword found in Bloom Filter do
    $\bar{P} \leftarrow$ Intersection of all files chained to all corresponding 1 bit positions
end for
Return $\bar{P}$
```

One issue of importance in the proposed scheme is memory limitation at a node. With time, more and more queries arrive in the system, and more and more keywords need to be hashed. More importantly, since the bucket chained to each bit will also have memory limitations, and there will be a limit of the number of file ids stored there. In our scheme, we ensure that when buckets are full, and files have to be replaced to accommodate new queries, the newer files must have equal or higher rank than current files. Otherwise, they are deferred from being added to the Chained Bloom Filter until their Rank becomes more than those currently in the filter.

4.4. ANALYSIS

In this section, we conduct an analysis of our architecture and our popularity aware content retrieval protocol from two perspectives: retaining popular files in the filter, and the probability of returning relevant files. In the following, we denote $P(X)$ as the probability that Statement X is true. We also denote r_f as the Rank of file f. There are k hash functions used during hashing of a keyword in the Bloom Filter. We denote N as the total number of files in the database, and c is the capacity of each bucket, which is the number of file ids that can be stored in it.

$P_f^{f'}$: $P_f^{f'}$ is the probability of a file f' is not present in the filter at a node when File f is present, where $Rank(f') < Rank(f)$. It is easy to see that $P(f'$ is not in Filter when f is in the filter) $=$ P(keywords of f' is not hashed in same buckets as keywords in f) $\times$ P(keywords of f' is hashed in buckets where all files are ranked higher that ${r_f}'$).

In order to derive the worst case probability, we assume that all files have a minimum of l keywords. We also assume that in total there are S files in the node's database whose ranks are higher than rank(f'). We also assume that all l keywords of a file can be hashed to k' distinct buckets at the minimum.

As such, the probability that keywords of f' is not hashed in same buckets as keywords in f is,

$$P_1 = \frac{\binom{m-k'}{k'}}{\binom{m}{k'}}. \tag{20}$$

Similarly, the probability that keywords of f' is hashed in buckets where all files are ranked higher that r' is,

$$P_2 = \frac{\binom{S}{c}^{m-k'} - \sum_{i=1}^{m-k'} \sum_{j=1}^{\frac{m-k'}{k'}} \left(\binom{S}{j} \times \binom{m-k'}{k+i}\right)^j}{\binom{N-1}{c}^{m-k'} - \sum_{i=1}^{m-k'} \sum_{j=1}^{\frac{m-k'}{k'}} \left(\binom{N-1}{j} \times \binom{m-k'}{k+i}\right)^j}. \tag{21}$$

Consequently, we have

$$P_f^{f'} = P_1 \times P_2. \tag{22}$$

Probability of returning files irrelevant files to a query using Algorithm 4: If a query and a file have no keywords in common with each other, the file is called irrelevant to the query. This could happen in Bloom Filter based designs due to the inevitability of False Positives. We study this probability here.

Let i be the number of keywords in a current Query q_{cur} and ϵ $(i \geq \epsilon)$ be the maximum number of keywords matching in a cached query with the incoming query. Let us consider Query q_{arb} as the arbitrary cached query for which results were returned in response to q_{cur}, which is a False Positive. So the probability that a keyword of q_{arb} is also hashed to the same bits as query q_{cur} is,

$$\begin{aligned} &= \frac{k}{m}(m-i-1)\left(\frac{k-1}{m-1}\right)\left(\frac{k-2}{m-2}\right)...\left(\frac{1}{m-k+1}\right) \\ &= (n-i-1)\frac{k!}{(m-k+1)!} \end{aligned} \tag{23}$$

For files associated to q_{arb} to be returned, this process has to repeat atleast ϵ times. So the probability becomes,

$$\binom{ki}{\epsilon k}\left[(n-i-1)\frac{k!}{(m-k+1)!}\right]^{\epsilon} \tag{24}$$

So the probability of returning only irrelevant files is,

$$\sum_{j=i}^{i-\epsilon}\binom{ki}{k(\epsilon+j)}\left[(n-i-1)\frac{k!}{(m-k+1)!}\right]^{\epsilon+j} \tag{25}$$

Note that in the above expression if $\epsilon = 0$, it represents the case of false positive, which means no relevant query has been cached yet, still the system returns some irrelevant files. For $\epsilon > 0$ it means that there are relevant files in the system, but non-relevant files are returned instead.

4.5. PERFORMANCE EVALUATION

In this section, we present the simulation results for evaluating the performance of our multi-tier architecture and protocols for content retrieval.

4.5.1. Simulation Setup.

The simulations are run in a self-made C++ simulator. In the simulation setup, we consider 100 nodes following Random Way Point Model in a 15 × 15 square unit area. Each node can store a number of files with keyword descriptors. The size of each file is considered as a single memory unit. Every node also maintains a CBF. Nodes generate queries containing keywords. After a query is generated at a node, it is processed in both the tiers and forwarded to intermediate nodes within 5 hops. We use flooding technique for forwarding of queries to peer nodes and responses are returned using the already established path while forwarding. Results returned are consolidated from multiple node searches.

Default parameters are listed in Table 4.3, and every data point is collected after taking an average from 50 runs of simulation to avoid any bias.

4.5.2. Performance Evaluation of Tier 1. In Figure 4.3 and Figure 4.4, we study the performance of the search protocol in Tier 1 from the perspective of search time overhead at a node, the similarity obtained in our proposed technique and the error between our technique and exhaustive search. In all the figures, the term *PS* stands for the *Proposed Search Technique*, while *ES* stands for the baseline *Exhaustive Search Technique*.

Figure 4.3 first shows that with more files in the database, the search overhead increases since more the no. of files more are the options for searching. Figure 4.3 also demonstrates that with increasing number of files in the database, our proposed protocol for Tier 1 greatly reduces search time overhead at every node, compared to exhaustive searches. As expected, with more no. of files in the database, the proposed technique converges. This phenomenon can be explained from the fact that with more files, the time taken to find expected similarity tends to grow very slowly. Queries with longer keywords converge slightly faster, again due to the reason that for longer queries highly accurate results are more difficult to find, and so the expected value of similarity and search time converges. In Figure 4.4, we study how much accuracy we are sacrificing in our motivation to save overhead. As we can see, the worst case error between our technique and an exhaustive search is roughly around 9 °, which is quite a small error specially considering the significant savings in overhead.

4.5.3. Performance Evaluation of Tier 2. We discuss the performance of Tier 2 from the perspective of the Chained Bloom Filter, in terms of retaining popular content and number of False Positives.

In Figure 4.5, we study the miss rate as a function of query rank (where the rank is based on popularity of the keywords in the query from Equation 5). The miss rate

Table 4.3. Simulation Parameters and Values

Parameter	**Default Values**
simulation time	120 units
simulation area	15 × 15 sq. units
no. of nodes	100
communication range	1 unit
no. of files per node	$1000 - 2000$
total no. of keywords	50
no. of keywords in file	$2 - 8$
no. of keywords in query	$2 - 8$
Wait time at a point	$5 - 15$ units
no. of nodes querying at a time instant	$1 - 5$
no. of bits in bloom filter	20
capacity of each bucket	10
no. of hash functions	3
damping factor (λ)	0.8
no. of top ranked files (x)	2
no. of hops searched	5 hops

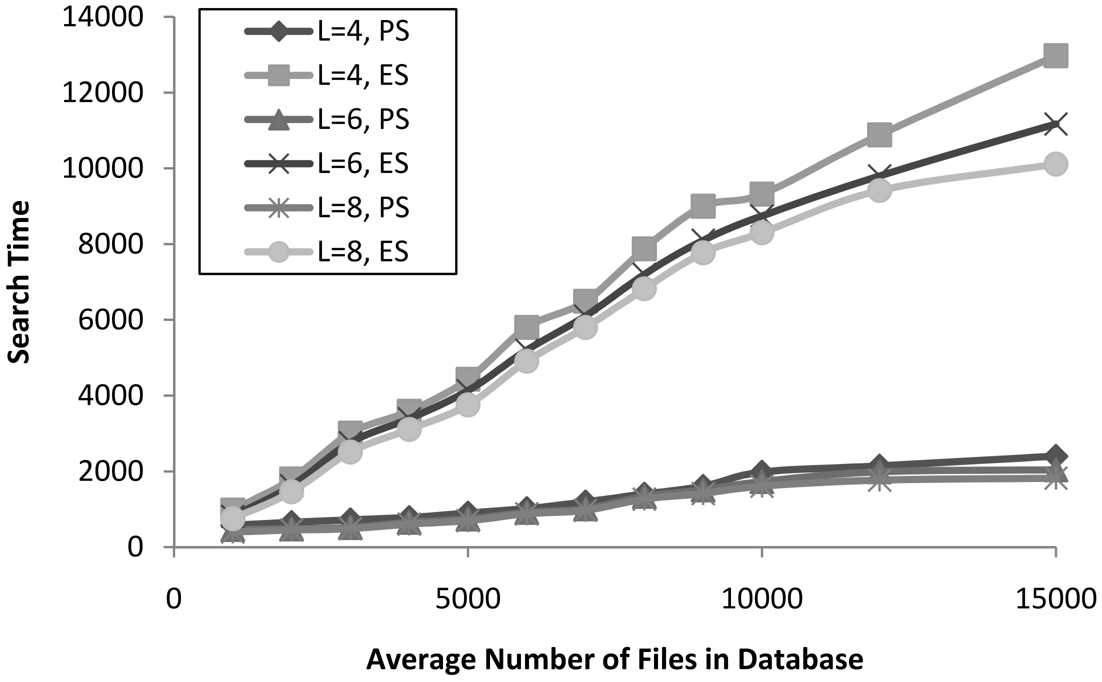


Figure 4.3. Search Time vs. Avg. No. of Files in Database for Different Query Lengths

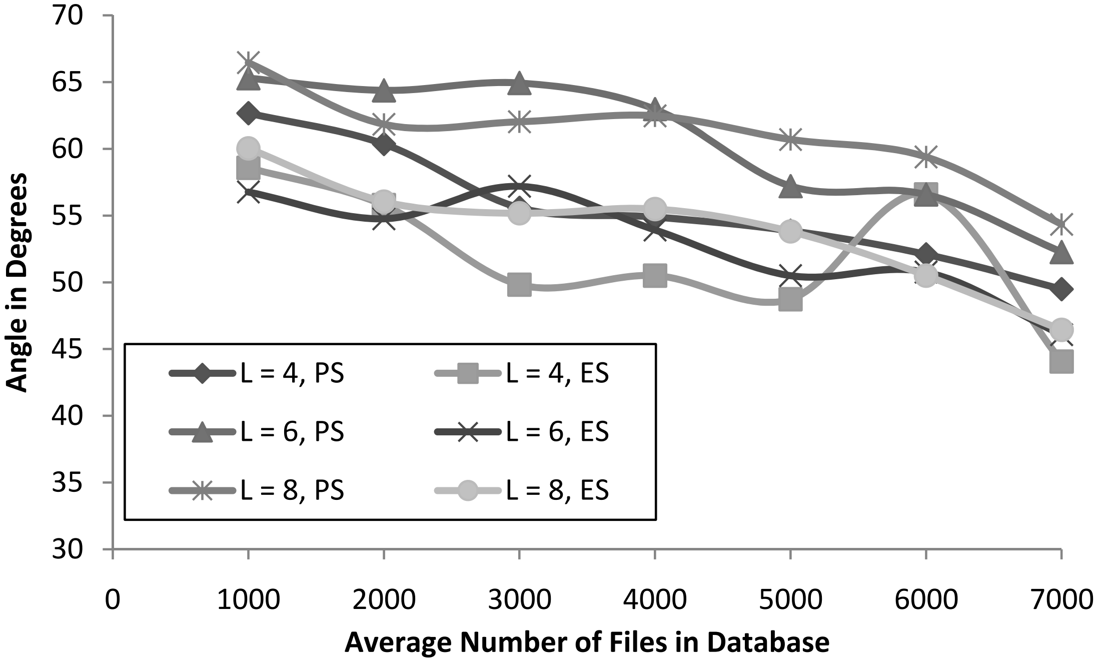


Figure 4.4. θ vs. Avg. No. of Files in Database for Different Query Lengths

is the percentage of time a relevant file was not found in the Chained Bloom Filter. As we can see, when queries contains more popular keywords (above 80 percentile), the miss rate is quite low, and it increases when queries contain unpopular keywords. This demonstrates the effectiveness of the proposed technique in retaining popular files in the Chained Bloom Filter. In Figure 4.6, we study the orthogonal issue of False Positives. Note that since False Positives are a problem with Bloom Filters, we would like to see how the proposed scheme is affected by it. We study this in Figure 4.6, where we plot the number of times an irrelevant file was returned from proposed Chained Bloom Filter as a function of number of Bloom Filter bits. Note that by irrelevant files, we mean files that did not contain any of the keywords in the query. We see that as the number of bloom filter increases, the number of irrelevant files returned goes down dramatically. This trend is significant considering that the number of files in the database was 2000, and even a small addition of bits can significantly lower the False Positives as the database size increases.

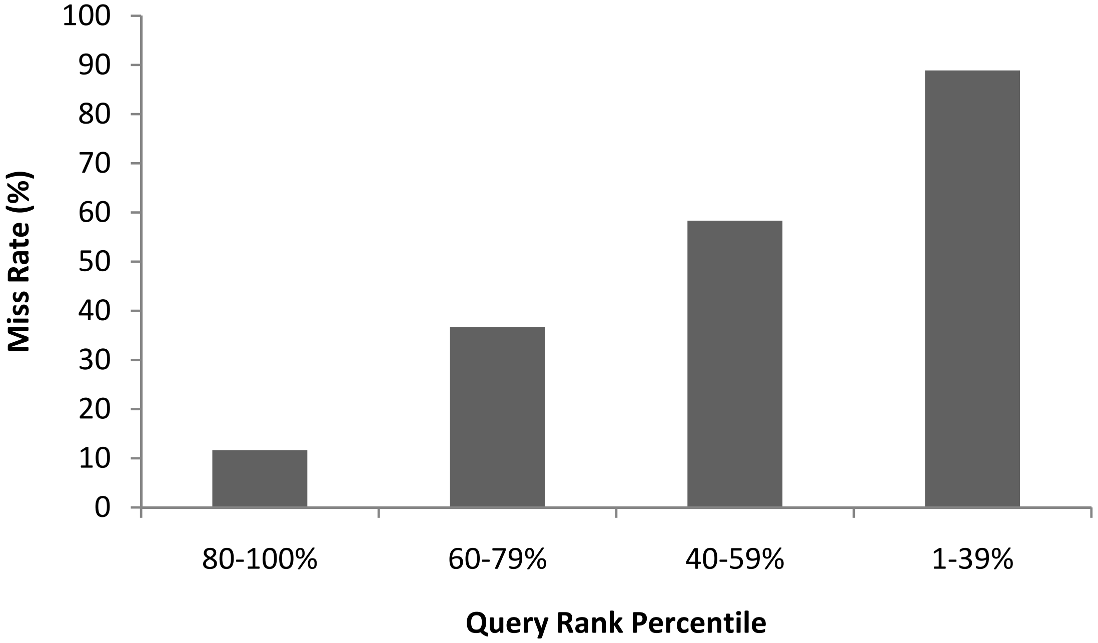


Figure 4.5. Miss Rate vs. Query Rank Percentile

4.5.4. Performance Evaluation of Tier 1 with Respect to Parameter α. Finally, we study the impact of letting the user modify α to see how this changes the search overhead. Recall from Section 4.3.2 that when the user sets α in the Logistic Function to be a higher value than the default, the user prefers lower accuracy files. The reverse is true when α is set to a lower value. As we can see, this trend holds in Figures 4.7 and 4.8. When α is set low, the search overhead increases, along with the accuracy, while when α increases the reverse happens. This trend has important impacts particularly from the perspective of pricing and incentive

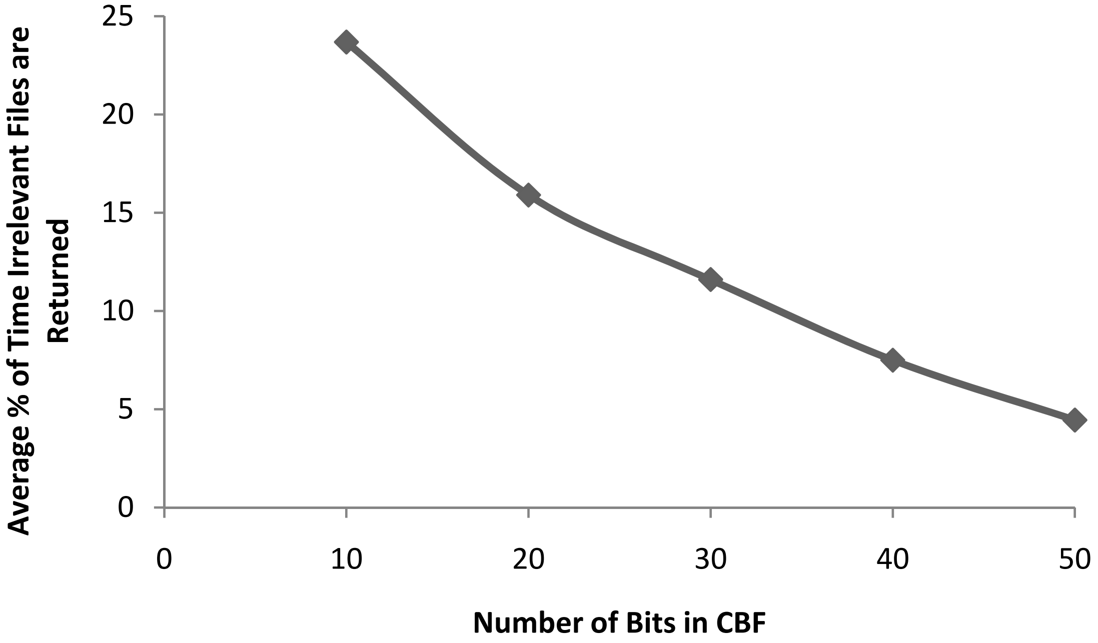


Figure 4.6. Percentage of Irrelevant Files vs. No. of Bloom Filter Bits

management in P2P systems. As we can see, while increased search overhead does bring in improved accuracy of search, the relationship is not linear. It can be noted here that, the question now is whether the system should decide pricing mechanisms based on accuracy or based on the overhead. How can the system resolve the trade-off in designing optimal pricing and incentive management schemes by exploiting this trend is part of future work. This problem is more challenging when users themselves can change the parameter α.

4.6. FINAL REMARKS

In this section, we address the issue of content retrieval in mobile P2P networks via a multi-tiered architecture and a suit of protocols. The quality of response versus search latency trade-off under ad hoc node mobility is investigated in this section. We first study the content retrieval process as a Logistic Function, and

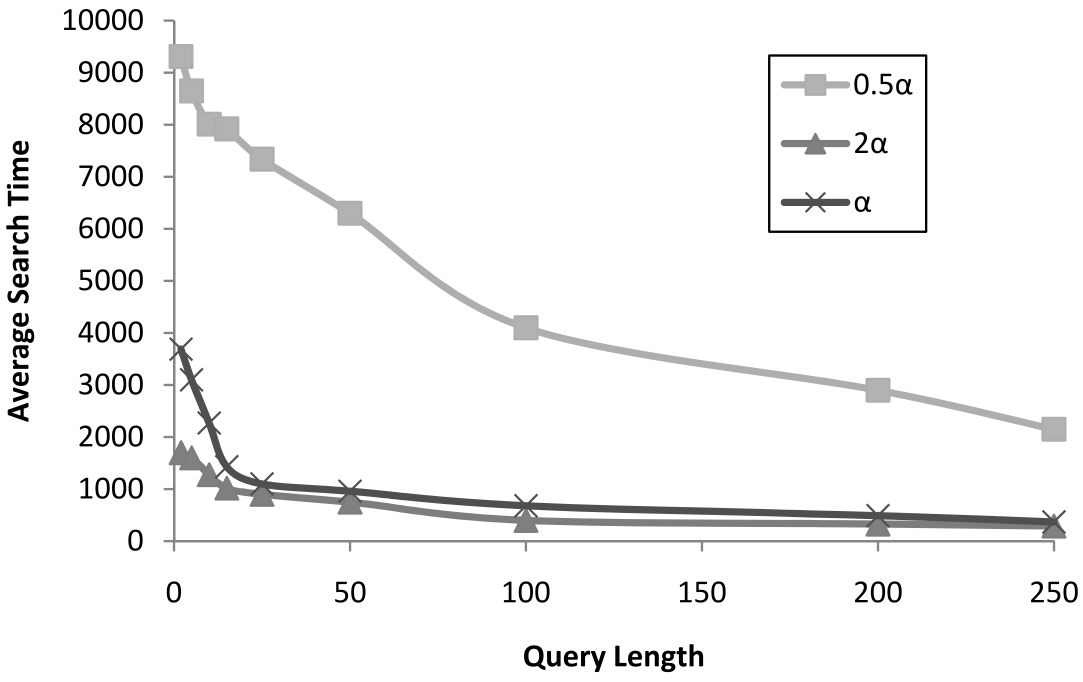


Figure 4.7. Search Time vs. Query Length for Different α

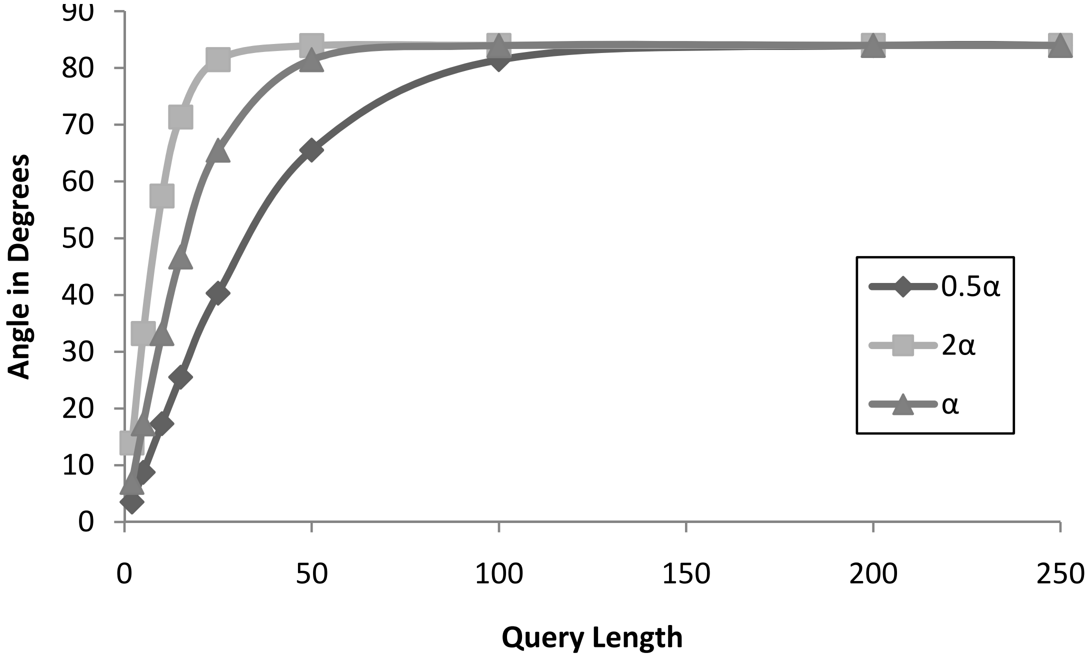


Figure 4.8. θ vs. Query Length for Different α

exploit this property for reducing search overhead with low impacts to accuracy. We also show how the growth parameter α in the Logistic Function provides the user with the ability to control the accuracy and overhead of search. We then design a novel Chained Bloom Filter technique that enables each node to store and retrieve popular keywords and content in a space and time efficient manner. We demonstrate the performance of the proposed techniques via extensive analysis and simulations.

5. LOCATION CLUSTERING BASED SYBIL ATTACK DETECTION IN VEHICULAR AD HOC NETWORKS

Sybil attack is a threat wherein an attacker creates and uses multiple counterfeit identities risking trust and functionality of a peer-to-peer system. Sybil attack in Vehicular Ad hoc Networks (VANET) is an emergent threat to the services and trust of the system. In the highly dynamic environment of a VANET, it is challenging to detect the nodes that are launching Sybil attack. This is because due to mobility, an attacker can easily create and use multiple fake identities, and exploit node mobility to exit the location of the attack. Consequently, detecting the presence of Sybil attack and identifying the Sybil nodes become a challenge considering the dynamic nature of vehicular networks, ephemeral neighborhood proximities and ad hoc mobility. Existing techniques mostly use additional hardware or complex cryptographic solutions for Sybil attack detection in VANETs. In this section, we propose a fuzzy time-series technique to cluster mobile nodes' locations based on neighborhood proximity. Our method does not require any additional hardware or infrastructural support for Sybil attack detection in VANETs. The underlying principle behind our approach is as follows. As a Sybil node counterfeits multiple identities and presents them to the system, those *fake* vehicles (represented by the counterfeited identities) will generally be reported around the Sybil vehicle that uses the identities leading these vehicles to violate normal dispersion dynamics. The proposed technique leverages the dispersion of vehicle platoons over time in a network and detects Sybil nodes as those which are located closely in a cluster as they move for an unusually long time. Simulation results and analysis show that the approach is able to identify Sybil nodes with very low false positive and false negative rates even under varying intensity of attack. The work presented in this section has been accepted for publication in proceedings of Vehicular 2013 as mentioned in [42].

5.1. BACKGROUND AND RELATED WORK

Vehicular Ad hoc Network (VANET) is a type of ad hoc network that is comprised of vehicles and road transportation infrastructure. The application of VANETs in different emergency notification system, safety-related and infotainment purposes have increased over past few years, leading it to become the backbone of *Intelligent Transport System* (ITS) [160]. Alongside, new security threats in VANETs have been investigated as well [79], [30], [43]. In this section a critical trust-based service, namely detection of *Sybil attack*, has been addressed and a time-series clustering based approach is proposed for detection of nodes that are launching this attack in VANETs.

Sybil attack [38] is defined as a security threat in large scale peer-to-peer system wherein a single malicious entity creates and uses multiple counterfeit identities over time. In Sybil attack, a peer-to-peer reputation system is subverted via counterfeit identities, hence compromising trust and functionality of the system. Sybil attack can enable the attacker to control a large portion of the system. In VANETs, Sybil attacks can affect services like emergency notification, route planning, congestion avoidance, etc. and deteriorate the overall performance of the system. Existing work present methods to detect the presence of Sybil attack in a network and localize the nodes that are malicious with the help of additional hardware, infrastructural support or complex cryptographic solutions. The techniques presented in this section can detect the nodes which launched the attack using the basic mobility property of nodes in a VANETs, that is, dispersion of vehicles over time.

While Sybil attacks have been addressed in social networks, Wireless Sensor Networks and Mobile Ad Hoc Networks, solutions in these domains require long term observation, collaboration and verification which are not possible in ephemeral networks like VANETs, where associations are short and unlikely to repeat. However, there have been research for detection of Sybil attack and identification of Sybil nodes in VANETs as well. In [15], a physical signal characteristics based technique was discussed for Sybil node detection in VANETS. A pair of nodes could be distinguished from each other using estimate of relative node localization that gives an indication of the coherence of the received signal. A signal strength distribution based method for detection and localization of Sybil nodes is proposed in [155] too. In [173], authors propose to employ road-side boxes (RSBs) that issue temporally varying pseudonyms to vehicles near their vicinity. A cryptographic solution to the problem of Sybil attack detection is proposed in [132]. In [146], spatial and temporal correlation between vehicles and RSBs is used to detect Sybil nodes, exploiting the fact that two vehicles passing by multiple RSBs at exactly the same time is rare. In [61], the authors presented a general approach to validate the VANET data, even in the presence of a few Sybil nodes. Anomalies are detected by checking the validity of the VANET data with respect to the VANET model and adversarial model. In [63], a neighborhood grouping based distributed Sybil detection method is proposed. A location-privacy aware trajectory tracking and authentication approach is used for Sybil attack detection in [24]. RSUs participate in message based authentication in this system. In [149], a dispersion based approach for Sybil detection in MANETs is proposed.

Our Contributions - Existing techniques for Sybil detection in VANETs mostly require additional hardware and overhead, but they do not use the availble network physics, physical infrastructure information and statistics. The Sybil node detection technique proposed in this section does not need any external support or complex algorithms, but rather relies on leveraging a basic mobility feature of nodes in VANET - the dispersion of vehicle platoons over time, or platoon dispersion [35]. Platoon dispersion indicates that in normal conditions, vehicles in proximity of each other at a certain time are unlikely to sustain their proximity clustering over time, i.e., proximity clusters are ephemeral. Our proposed solution is based on this premise and uses fuzzy time series clustering for detection of Sybil nodes. Fuzzy time series clustering involves fuzzy clustering of time series data collected over time with even or uneven sampling rate. In this work, *Fuzzy Short Time Series* (FSTS) clustering of location traces of mobile nodes over a time period is used for Sybil attack detection. The proposed technique is based on the FSTS algorithm presented in [118]. We incorporate data preprocessing and feature extraction phases to make the algorithm more efficient. We also perform theoretical analysis and simulations to derive threshold parameters and demonstrate performance of the technique. We also take into consideration various intensities of attack which the attacker can adopt by utilizing only a part of its available counterfeit identities at a time. Such a variation in attack model makes it all the more difficult to estimate consistent association of nodes with one another. Clearly, this variation adds to the challenge of clustering nodes based on location traces over time. Simulation results show that the proposed technique succeeds in identifying most of the Sybil nodes over a period of time under such conditions as well.

5.2. PROBLEM DEFINITION

The network model, attack model and the problem addressed are defined in this section.

Network Model - The main components of the VANET are - vehicles, Road Side Units (RSUs) and Certification Authority (CA). Vehicles are alternatively referred to as “nodes” in this section. Nodes in VANETs are equipped with on board units (OBUs) to communicate and compute messages. Nodes may also have sensors, navigation device or GPS, computing devices, display units, etc. Each node is aware of its own location and the map of the network area. Nodes usually communicate using short range wireless communication technology, such as Dedicated Short Range Communication (DSRC), bluetooth, IEEE 802.11, etc. RSUs usually comprise of cheap embedded devices including sensors, smart traffic controllers, etc. RSUs store secure information such as its secure communication keys, traffic information, safety-related information etc. An RSU can communicate with the nodes in the network and other RSU's. CA is a central authority which authenticates vehicles and RSUs using the secure authentication infrastructure like public key infrastructure. Each node is given a unique identity or ID by the CA. However, CA and cryptographic algorithms that are generally used for secure communication in VANETs [135] do not effect the proposed technique directly or indirectly.

Attack Model: A Sybil node is defined to be one which uses multiple counterfeit identities to pretend to be some other node(s). As discussed in Section 5.1, the benefits of the attacker in launching such attacks are multi-folded. A group of malicious nodes can subvert the trust and reputation system of the network if they conduct Sybil attack on the network for some time. Eventually this can deteriorate the overall performance of the system. In our model, we consider that a malicious vehicle, with original id V, has n different identities, $V_n = V_0$, V_1, ..., V_{n-1}. V can determine the intensity of attack by choosing to use only a certain percentage of the counterfeit identities at a time. Intuitively, using lesser number of ID's at a time will lower its chance of getting detected, but at the same time it will mitigate the intensity of attack as well. In our model, V uses $x\%$ of these ids over a time duration Δt where $x \in [0, 100]$. It is assumed that V randomly selects i different ids from the set V_n such that $x = \frac{100i}{n}$ and uses them to communicate for the next Δt, and then again repeats the same process. We assume that the vehicles follow predefined speed limits on the roads.

In the very dynamic environment of a VANET, it is challenging to identify a Sybil node due to the high mobility and density of nodes. In other words, a node can escape one part of the network and reach another part very fast. The large number of nodes in a network makes it all the more difficult to identify malicious node(s). These challenges warrant the need of a lightweight and efficient approach to detect Sybil nodes in VANET. The objective of the work is to propose an efficient method to detect Sybil nodes without using additional hardware or infrastructural support.

5.3. PROPOSED SOLUTION

In this section, a time-series clustering method called FSTS [118] is used to detect Sybil nodes in VANETs under varying attack intensity. Time series clustering helps to identify the nodes which are moving in proximity of each other over a time period based on the location traces of the nodes. Because of the large number and density of nodes in a typical VANET, it is likely that a node can be part of multiple clusters at the same time, making fuzzy clustering algorithms suitable for the scenario. In this section we first discuss the location data collection method, followed by different steps of the proposed technique for Sybil node detection.

The key idea behind the proposed solution comes from a vehicular network phenomenon called *platoon dispersion* [35] as mentioned in Section 5.1. A platoon is a group of vehicles traveling together. If all vehicles in an existing platoon maintain their speeds, a platoon will never break up. However, due to physical factors like road friction, vehicle characteristics and signaling, along with human factors like car following pattern, lane changes, fatigue, there is inherent randomness in driver behavior, and platoons tend to disperse over time. Intuitively, longer the travel time between points, greater is the dispersion, since there is more time for drivers to deviate from current speeds. We use this idea to derive a threshold probability

P_{Th} of two vehicles being within a specified distance after a given time if their initial locations were same.

By the virtue of platoon dispersion, different vehicles in a network are not likely to traverse together for very long. Towards this end, the threshold duration for which vehicles are likely to travel with each other can be estimated theoretically. If any two or more vehicles cross this threshold, they are likely to be the same node faking identities as different nodes. The clustered time series correspond to the identities of the vehicles which are likely to be Sybil nodes.

5.3.1. Location Data Collection by Peer Nodes.

Standard DSRC communication allows vehicles to update its location and other physical parameters using periodic messages at a short, regular interval (usually 20 ms). However, in the scenario considered, any node can be a malicious Sybil node and it can also falsify its own location information to avoid detection. So location data of vehicles over time is collected through peer vehicles through via messages or *report*. In our method, we take into consideration this scenario and involve all peer nodes for location data collection to avoid any possible manipulation by malicious nodes. All nodes send *report* messages to the base station on a periodic basis in a fixed time interval. The purpose of *reports* is to inform the base station about the nodes which V_x has heard communicating in the last time interval. Because only a part of the nodes could be malicious, this collaborative process of reporting assures that the real location of a node is reported. For instance, if node V_x receives message from a node V_y at time t when V_x was at location l, it will incorporate this information in its next *report* to the base station. The location data of V_y collected by peers over time is represented in form of a time series $L_{V_y} = l_{V_y}(0), l_{V_y}(1), ..., l_{V_y}(t)$. It can be noted that the RSUs deployed along the road serve as local base stations that can execute the clustering algorithm and collaborate with each other as needed too. If the communication range of a nodes is R meter, V_y was within a circle with radius R meter from L_{V_y}. Hence the location estimation of V_y at time t has error limit ϵ which is upper-bounded by πR^2. If more than one node reports about the location of V_y at t, the error, decreases.

5.3.2. Preprocessing Collected Data.

After base station collects the location data from nodes in the network, all the following steps are executed by the base station for detection of Sybil nodes. Clustering algorithms are usually used for evenly distributed sampling for time-series, or can handle unevenly sampled data to some extant. But handling the ad hoc nature of data in VANET, specially when the Sybil node uses only a part of it's Sybil ID's at a time, becomes an orthogonal challenge. In simulation based experimentations it is feasible to collect data with regular sampling rates, but it is unlikely to do so in practical scenario. For instance, locations of V_x can be reported by peer nodes time instants t_0, t_1, t_5, t_9, t_{10} and t_{20} whereas locations of V_y can be reported by peer nodes time instants t_0, t_1, t_2, t_3, t_9, t_{11}, t_{12} and t_{15}. Clustering of these two time series becomes due to the irregularity of sampling rate and size. In this section, the effect of linear interpolation in time series clustering of data is studied. Subsequently in Section 5.3.4, a prediction technique is proposed to estimate locations of vehicles when no report is obtained. Although the time-series clustering algorithm used in this section supports clustering of unevenly sampled time-series data, preprocessing of collected data is done for better results.

Linear Interpolation - Referring back to Section 5.3.1, the time series data for V_y can be represented as, $L_{V_y} = l_{V_y}(0), l_{V_y}(1), ..., l_{V_y}(t)$, where, $l_{V_y}(i) = (x_{V_y}(i), y_{V_y}(i))$. The linear interpolation between points $(x_{V_y}(i), y_{V_y}(i))$ and $(x_{V_y}(j), y_{V_y}(j)) \forall (i,j)$ and $(j-i) > 1$, can be given by,

$$y = y_{V_y}(i) + (x - x_{V_y}(i)) \frac{y_{V_y}(j) - y_{V_y}(i)}{x_{V_y}(j) - x_{V_y}(i)} \tag{26}$$

The data points between $l_{V_y}(i)$ and $l_{V_y}(j)$ can be constructed on the line represented by Equation 26 at regular distances $\Delta d = \frac{||(l_{V_y}(i), l_{V_y}(j))||}{p-1}$, where $(j - i) = p$ and $||.||$ refer to Euclidean distance.

5.3.3. Estimation of Number of Sybil Nodes. Association rule mining is used as feature extraction step in this work, in order to have an idea about how many Sybil nodes are likely to be present in a part of network. Association Rule Learning mines relation between multiple attributes of an entity based on their frequency of co-occurrence in a dataset [5]. Let $I = i_1; i_2; i_3, ..., i_r$ be a set of r binary attributes called items. Let $\tau = \tau_1, \tau_2, \tau_3, ..., \tau_s$ be a set of s transactions called a database. Each transaction in τ contains a subset of the items in I. The problem here is to identify association rules in the database, which is an implication of the form $X \implies Y$, where $X, Y \in I$ and $X \bigcap Y = \emptyset$. Reverting back to Sybil detection, consider a Vehicle V_x that has communicated with peers over time. The dataset τ_x of V_x is a row of transactions with each time-stamped row consisting of vehicle ids with which V_x has communicated at that time. Recall from platoon dispersion that a group of vehicles is highly unlikely to be consistently associated geographically (i.e., as a platoon) over a long time period. When a consistent association of two or more vehicles is seen, those vehicles can be suspected to be Sybil. Using this technique, different peer nodes in a network can predict how many Sybil nodes are likely to be present in its vicinity and report to the base station. Also, the base station itself can use this technique to estimate which nodes could possibly be Sybil. However, it is not possible to draw a conclusion from their analysis when the Sybil node uses only a part of forged identities over time and changes them over next time period. This step is only useful for the base station to predict expected number of clusters, w_{ij}, which is an input to the clustering algorithm as discussed in Section 5.4.1.

5.3.4. Fuzzy Time Series Clustering. The basic principle behind short time series based fuzzy clustering is derived from [118]. It can be noted that the proposed short time series based piecewise slope distance clustering seems intuitively appropriate for the application considered in this section. The type of location data obtained from vehicles in a VANET can be enormous in size, but the Sybil detection technique deals with data over a comparatively shorter period of time. However, there are several differences in the two approaches. Firstly, in our work, two dimensional location data is considered for clustering over time. So the time series data considered is three dimensional unlike the two dimensional clustering performed in [118]. Besides, this technique is further extended in Section 5.3.4 to leverage the advantages of estimation techniques in the domain of time series clustering.

Fuzzy short time series (FSTS) technique proposed in [118] is a variation of fuzzy C-means clustering for time series data. The basic idea is to perform a slope distance computation of time series which can be used for clustering the time series in FSTS method. In this work, the distance considered includes the three dimensional data (x and y coordinates of location and time) obtained from VANETs. For time series of vehicle $l_{V_x} = l_{V_x}(0), l_{V_x}(1), ..., l_{V_x}(t_n)$, the linear function between $L_{V_x}(t)$ two consecutive time instants t_k and $t_(k+1)$ are defined as,

$$L_{V_x}(t) = m_k(t) + b_k, \tag{27}$$

where $t_k < t < t_{k+1}$, and

$$m_k = \frac{||l_{V_x}(k+1) - l_{V_x}(k)||}{t_(k+1) - t_k}, \tag{28}$$

$$b_k = \frac{t_(k+1) l_{V_x}(k+1) - t_k l_{V_x}(k)}{t_(k+1) - t_k} \tag{29}$$

Equation set above results in a set of equations as both x and y coordinate as separately considered for difference in Equation 29. The short time-series distance between time series vector of vehicle V_x and prototype vector V_y is computed as below -

$$\begin{aligned} d^2_{STS}(V_x, V_y) = \sum_{k=0}^{n_t-1} & \frac{V_y(k+1) - V_y(k)}{t_{k+1} - t_k} \\ & - \frac{V_x(k+1) - V_x(k)}{(t_{k+1} - t_k)^2}. \end{aligned} \tag{30}$$

Rest of the FSTS algorithm is similar to fuzzy C-means algorithm . The cost function is defined as,

$$J(V_x, V_y, u) = \sum_{i=1}^{n_k} \sum_{i=1}^{n_v} u_{ij}^w d^2(V_x(j), V_y(i)), \tag{31}$$

where n_k is the number of clusters, n_v is the number of vehicles and w is the weight factor. All these values are user-defined. The value of u determines the membership value of the element in the cluster. Updating of the partition matrix is done in the same way as described in [118], where u_{ij}^w is updated as,

$$u_{ij}^w = \frac{1}{\sum_{q=1}^{n_k} (d_{STS_{ij}} / d_{STS_{qj}})^{\frac{1}{w-1}}} \tag{32}$$

Further details of this algorithm can be found in [118].

5.3.5. Derivation of P_{Th}. In this section, the objective is to derive P_{Th}, the probability of two vehicles traveling in each other's vicinity so that the expected time of observation for Sybil node detection can be estimated. Towards this end, first theoretical analysis is performed to determine $P_T h$ and then the outcome is tested using simulation studies.

Let us consider that two vehicles are moving on a straight road. They are initially (time $t = 0$) at a distance d_0 apart. In a time interval δt, the vehicles can move any distance within a range of D_H and D_L on the road. The range is represented as D_{range}. At every time instance the vehicles update their velocities based on past velocities and thus the distances to be covered (denoted by D_1 and D_2) in next time interval, δt. D_1 and D_2 are chosen from D_{range} using uniform distribution. Our initial objective is to figure out the probability that the two vehicles are within a distance α of each other after a time interval $n\delta t$.

As mentioned above, we assume uniform distribution for D_1 and D_2. For simplicity of computation, we assume $d_0 = 0$ throughout this derivation. Now, using normal approximation of uniform distribution, if $D_1 \sim Unif(D_H, D_L)$ and $D_2 \sim Unif(D_H, D_L)$, then
$\sum_{i=1}^{n} D_{1i} \sim N(\frac{n(D_L+D_H)}{2}, \frac{n(D_H-D_L)^2}{12})$ and $\sum_{i=1}^{n} D_{2i} \sim N(\frac{n(D_L+D_H)}{2}, \frac{n(D_H-D_L)^2}{12})$.
So, $(\sum_{i=1}^{n} D_{1i} - \sum_{i=1}^{n} D_{2i}) \sim N(0, \frac{2n(D_H-D_L)^2}{12})$.

Now, the probability that the condition $|\sum_{i=1}^{n} D_{1i} - \sum_{i=1}^{n} D_{2i}| \leq \alpha$ holds true can be written as

$$\begin{aligned}
&P(|\sum_{i=1}^{n} D_{1i} - \sum_{i=1}^{n} D_{2i}|) \qquad (33)\\
&= P(-\alpha \leq \sum_{i=1}^{n} D_{1i} - \sum_{i=1}^{n} D_{2i} \leq \alpha)\\
&= P(\frac{-\alpha - 0}{\sqrt{\frac{2n(D_H-D_L)^2}{12}}} \leq Z \leq \frac{\alpha - 0}{\sqrt{\frac{2n(D_H-D_L)^2}{12}}})\\
&[where\ Z = \sum_{i=1}^{n} D_{1i} - \sum_{i=1}^{n} D_{2i}]
\end{aligned}$$

$$= P(-z \leq Z \leq z) \qquad (34)$$

$$[where\ z = \frac{\alpha}{\sqrt{\frac{2n(D_H - D_L)^2}{12}}}]$$

Using standard normal distribution of Z, i.e., $\Phi(Z)$, it is evident that, $\Phi(Z) = P(Z \leq z)$. So our probability expression, (in equation 34) = 2 $\Phi(Z)$-1.

Using the standard normal CDF table, the probability for different values of D_H, D_L, n and α can be found out. From this derivation, it is straight forward to derive the expected time, t_{exp}, that two vehicles will take to reach a threshold probability P_{th} that they are traveling in each other's vicinity. It can be noted that in real life, based on several physical and human factors, any other distribution other than uniform distribution can be used to model vehicle's distance traveled over a time period. However, similar derivation can be done using other probability distributions too.

A theoretical probability of two nodes moving within a given distance over a time period can be obtained by plugging in values of different input parameters into the expression derived above. In Figure 5.1, probability values derived through theoretical analysis and simulation results are plotted against different values of α where D_H = 50 m, D_L = 0 m, n = 10, δt = 1s. This figure shows a case where simulation data is plotted along with theoretical results to show that the results match closely. Thus from this derivation, for a given time period, the probability of two vehicles being in a same cluster (or within a given distance) for a given time period can be obtained. For different experiments performed with different values of network parameters (like D_H, D_L etc.), we derived the probability threshold for which two nodes can be in a cluster for a given time duration. If the output of FSTS algorithm yielded a higher cluster membership than the probability threshold derived, the node in the cluster are detected as Sybil nodes. Derivation of threshold parameter through this process helped us differentiate among nodes traveling together for long time and malicious Sybil nodes.

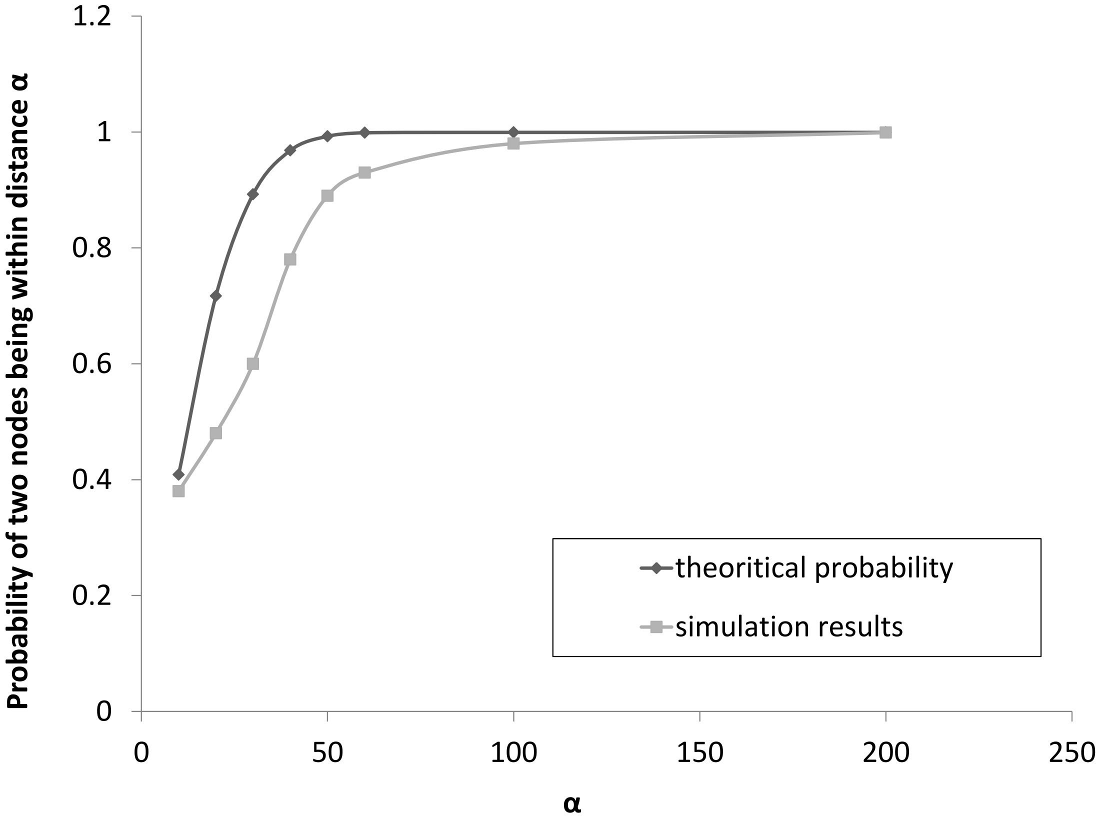


Figure 5.1. Determination of Input Parameters where $D_H = 50$ m, $D_L = 0$ m, $n = 10$, $\delta t = 1$ s

5.4. PERFORMANCE EVALUATION

In this section, the performance of the proposed technique is presented and analyzed.

5.4.1. Experimental Setup. SUMO (Simulator of Urban Mobility) was used to generate mobility traces of nodes and this data was used as input to the network. The simulation was conducted on a real city road network imported from

Open Street Map. A C++ simulator is developed to emulate the vehicular network where the nodes move following the mobility traces from SUMO, thereby mimicking real traffic patterns. The final clustering experiments are done using the C++ simulator. By default, there were 100 vehicles with an average speed of 50mph, and sources and destinations were randomly chosen for each vehicle. Unless mentioned, there were 10 Sybil nodes among them, and each had 10 identities. Each vehicle was assumed to report it's location once every second, and the transmission range was assumed to be 250 meter. The simulation was run for 1000 seconds and the default clustering distance was 400 m. All simulations were conducted 10 times and results were averaged.

Different sets of experiments were run in different phases. First the collected time-series data is preprocessed using linear interpolation using Matlab and then association rule mining is used for feature extraction phase estimating expected number of clusters in the data using Weka. For the first phase of the study with association rule mining, each Sybil node used all its counterfeit identities during query response. Later the cases were studied when only a smaller percentage of identities are used by a node during a time period.

Apriori algorithm implemented in WEKA (Waikato Environment for Knowledge Analysis) tool was used as a feature extraction technique to identify abnormally repeating associations in the dataset of each vehicle independently as it completes its run. The success rate was 100% in Sybil vehicle detection without false positives. However when the percentage of ID's used by the Sybil node varied, only 60% of the Sybil nodes were detected and equal number non-Sybil nodes were detected as Sybil nodes. It means that the false positive and true positive rates were equal, which is not a desired performance. Clearly there is need of further analysis which is conducted subsequently. However, several association rule experiments help get an *feel* or estimate of how many clusters to look for and the probable number of

Sybil nodes in a set of nodes. For instance, in the case mentioned above, the results of feature extraction show that there are likely to be 12 clusters. In reality, there were 10 clusters that had Sybil nodes in them in that case. So in our experiment, we put the input number of clusters between 9 and 15, getting the best results when the number of clusters was 10. It can be noted that usually all clustering algorithm (including FSTS) require preprocessing and feature extraction of data or some sort of prior knowledge to estimate number of clusters. However, the results from association rule mining are not conclusive, warranting further experiments using the FSTS technique to determine the Sybil nodes from past location traces.

5.4.2. Clustering of Data. Recall from Section 5.3.5, theoretical analysis can be used to derive P_{Th} for different input parameters and the output can be used to determine whether the concerned nodes are Sybil or not based on their cluster membership values determined using FSTS. In the clustering process, firstly the binary connection metric is clustered using the FSTS algorithm. Figure 5.2 shows the detected number of false positives and false negatives averaged over 10 runs of simulation each. The X axis represents the percentage of available fake IDs that a Sybil node is using at a time instant. If all of the available IDs are used for transmission at every time instant, the false positive and false negative rates are both zero, indicating that all the Sybil nodes are identified. However, as the percentage decreases, both false positives and false negatives increase, although a major part of the Sybil nodes are detected over time. This figure demonstrates the effectiveness of the proposed technique in detecting Sybil nodes in VANETs.

Figure 5.3 plots the time required to reach 100% true positive rate (that is, detects all Sybil nodes) for varying percentage of ID's used by the Sybil nodes at a time instant. With increasing percentage of ID's used, the detection is faster. But

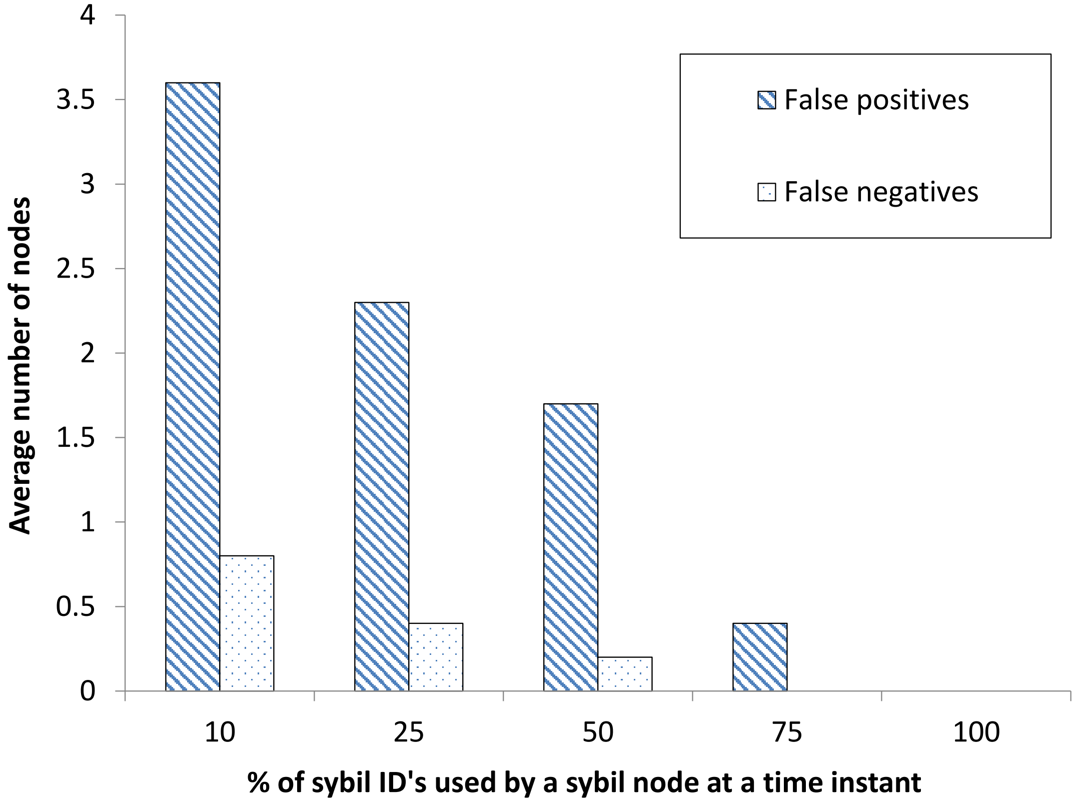


Figure 5.2. False positive and false negative rate for varying percentage of Sybil ID's used by a Sybil node at a time instant

as very less percentage of ID's are used by a Sybil node at a time instant, it still reaches 100% true positive rate in longer time.

5.5. FINAL REMARKS

This section proposes a technique for Sybil attack detection in VANETs, based on fuzzy time series clustering. The method leverages the principle of dispersion of vehicles in a VANET and detects the nodes clustered with each other for longer than expected. Theoretical analysis has been conducted to derive input parameter to the algorithm and simulation results are presented to evaluate performance of the proposed method. The proposed method has achieved very low false positive

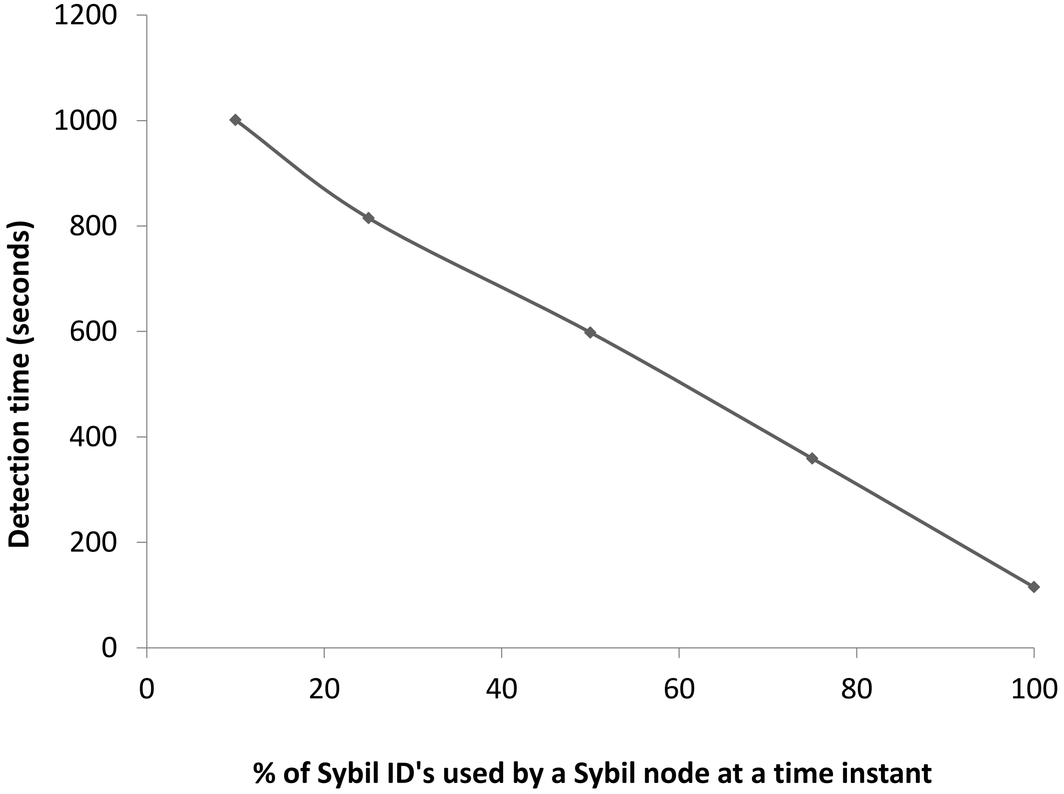


Figure 5.3. Detection time in seconds for varying percentage of Sybil ID's used by a Sybil node at a time instant

and false negative rates even when the Sybil nodes use a small percentage of the counterfeit identities at a time instant.

6. SUMMARY OF THE DISSERTATION

In this dissertation, we have investigated four research problems in the realm of Location Based Services in Wireless ad hoc networks. Broadly speaking, the research directions we have taken span the four dimensions of privacy, security, trust and performance enhancement. Towards this end, we have addressed the following problems in specific: i) providing Location privacy in wireless sensor networks; ii) providing end-to-end secure communications in wireless sensor networks; iii) enhancing content retrieval in mobile ad hoc networks and iv) defending against Sybil attacks in vehicular networks. We believe that with the wide spread demands for improved location based services, our work in this dissertation provides important contributions in this area.

Firstly, we address the issue of providing location privacy in wireless sensor networks. We define a practical wireless sensor network problem wherein an adversary that is not cooperating with the wireless sensor network attempts to surreptitiously discover locations of sensors in the network. The adversary (or localizer) leverages from analyzing raw wireless signals emanated by the sensors. Our objective in this chapter is to formally define and analyze this attack model and subsequently preserve location privacy of the sensor nodes under such attack. Although localization in wireless sensor networks is a widely researched topic, not many work address localization in scenarios where the nodes do not cooperate with the entity attempting to localize sensors. In this dissertation, we first propose a new method for localization of sensors in a non-cooperative environment by a mobile localizer, wherein the localizer receives no cooperation from the sensor nodes that constitute the sensor network. The localizer localizes the sensors using physical properties of the sensor communication messages: Angle of Arrival (AoA) and Received Signal Strength

Indicator (RSSI). Using the proposed method, the localizer can determine the presence of sensor node at a certain location with some error margin. This work shows how an external entity can invade in the location privacy of sensors in a network without being localized by the sensors. We call this kind of attack as adversarial localization. In other words, adversarial localization refers to passive attacks where an adversary attempts to disclose physical locations of sensors in the network by physically moving in the network while eavesdropping on communication messages exchanged by sensors. Our next contribution towards location privacy in wireless sensor networks is in designing a novel solution for defending against adversarial localization using a location privacy preserving tracking algorithm. The principle of the proposed approach is to allow sensors intelligently predict their own importance in light of two conflicting goals they have - preserving location privacy and tracking the adversary. The proposed algorithms ensures high degree of adversary localization, while also protecting location privacy of many sensors. Theoretical analysis and extensive simulations are conducted to demonstrate the performance of both the attack and defense models.

We next investigate the problem of secure end-to-end communication in randomly deployed wireless sensor networks, where one of the most fundamental challenges stems from lack of control where sensors are located in the network post deployment, especially under larger scale deployments. As a consequence, pre-establishing neighbor proximity information is not feasible, and so pre-fixing pairwise keys between sensors is not possible either. Beyond this challenge to securing communications in wireless sensor networks, energy limitations of sensor nodes clearly imply that complex cryptographic operations like public key based schemes are harder to implement in wireless sensor networks. Finally, while existing work focuses primarily on securing node-to-node communications, the issues of end-to-end secure communications (i.e., between a node to base station) is mostly ignored considering the significant location disparities between nodes and the base station in large scale sensor networks. In this dissertation, we propose an algorithm for end-to-end secure communication taking into consideration the issues originating from random deployment of sensors. We introduce a new principle called differentiated key predistribution wherein different number of secure communication keys are deployed to different sensor nodes. The links associated with nodes with higher number of keys have enhanced security and are called high resilience links which are subsequently used in routing secure messages between nodes and the base station. Subsequently, we couple the secure communication algorithm with data-centric and location-centric routing algorithms.

The next part of this dissertation focuses on quality versus latency in content retrieval in mobile ad hoc networks, from the perspective of improving performance of content retrieval. In mobile ad hoc networks, the rapidly changing location of nodes leads to the challenge in addressing a fundamental trade-off between accurate searches for queries and associated latencies. When a query is issued by a user in a mobile ad hoc environment, peer nodes can often provide a better quality of response compared to the local node. Hence in content retrieval applications, usually queries are forwarded to peer nodes for content retrieval. In this scenario, the core challenge comes from returning requested content to the node that requested it, as nodes change their locations rapidly over time. A fast response retrieving the relevant content can serve as a feasible solution to this problem as the response can reach the requester before it moves far away. Hence optimizing search latency is a critical factor while designing peer-to-peer based content retrieval algorithm in mobile ad hoc networks. However, there is a clear trade-off between accuracy of response and search latency wherein longer searches are usually expected to yield more accurate responses to queries. Towards this end, we perform a detailed study on the quality or accuracy of responses versus search latency and show that there is a distinct relationship between these two parameters in mobile ad hoc networks. Our investigation reveals that content retrieval in mobile ad hoc networks follows the trend of a logistic curve in terms of accuracy of response versus length of query. Simulation results proves our initial conjecture to be valid. We use this result to train our peer-to-peer search algorithm so that it learns the expected accuracy of response based on length of query. Thus the peer-to-peer search algorithm that we design optimizes search latency in mobile ad hoc environment while yielding high accuracy. It can be noted that accuracy or relevance of response is defined in terms of the match or similarity that a retrieved content has with the query. The entire content retrieval framework is encapsulated in a two-tiered architecture. The first tier deals with optimized peer-to-peer search and routing of contents. We use flooding based technique for query and response routing among peer nodes. The second tier entails another contribution of our research: searching the local database for relevant and popular contents based on past similar queries. We propose a chained bloom filter based technique for fast retrieval of popular contents that are relevant to the current query in the local node database. The chained bloom filter links queries searched previously at a local node with relevant popular content present in that node for subsequent retrieval. When a new query comes in the local node, the chained bloom filter can be searched in order to retrieve highly popular relevant contents that have been retrieved in past. We describe different operations available on the chained bloom filter, such as insertion, search and update.

The final contribution of this dissertation is in providing a trust based service, that is, Sybil attack detection in vehicular ad hoc networks by designing a location clustering based algorithm for anomaly detection. Vehicular ad hoc networks an emerging class of network systems, where vehicles traveling on roads communicate with peers and selected infrastructures to enhance quality of driving experience via improved congestion avoidance, route planning, real-time accident warning, commercial information disseminations etc. With such applications, the issue of trust among nodes becomes paramount. In other words, unless vehicles can trust information from their peers, such services will have limited utility. In this dissertation, we investigate solutions to one of the most practical and potent attacks in the realm of trust in vehicular networks, namely Sybil attack. In simple terms, a Sybil attack is one where an attacker creates and uses multiple counterfeit identities risking trust and reputation of a peer-to-peer system. The fundamental reason for the potency of Sybil attack in vehicular networks stems from the fact that due to mobility, an attacker can easily create and use multiple fake identities, and exploit node mobility to exit the location of the attack. Consequently, detecting the presence of Sybil attack and identifying the Sybil nodes become a challenge considering the dynamic nature of vehicular networks, ephemeral neighborhood proximities and ad hoc mobility. Existing techniques for detection of Sybil attack primarily use additional hardware or complex cryptographic solutions, which are quite cumbersome to deploy. In this dissertation, we propose a location based clustering of nodes using fuzzy time-series clustering based approach that does not require any additional hardware or infrastructural support for Sybil attack detection in VANETs. The proposed technique introduces a novel paradigm wherein dispersion of vehicle platoons over time in a network is leveraged to detect Sybil attack. Such dispersion is well studied in transportation engineering using a theory called *Platoon Dispersion*, that models vehicle locations in the form of neighborhood proximities over time and space as vehicles travel. The underlying principle behind our approach is that *fake* vehicles (represented by the counterfeited identities) will generally cluster around the Sybil vehicle that uses the identities leading these vehicles to violate normal road dispersion dynamics. In our protocol, we leverage clustering techniques to detect abnormal clusters (i.e., platoons). Abnormal clusters are indicative of multiple vehicles traveling closely for an unusually long time without following the normal principle of dispersion platoons, which is indicative of Sybil attack. To the best of our knowledge, this is the first application of a well established theory in transportation engineering to address a trust problem in vehicular networks.

7. CONCLUSIONS

In light of all the findings and contributions of this dissertation, we would like to conclude that we have shown the performances of few important application among versatile range of location based services in various wireless ad hoc networks from the perspective of several network parameters and scenarios. As a part of our study in location privacy in wireless sensor networks, we first introduce a grid-based non-cooperative localization method. This method can be used to locate sensors deployed in a network without any assistance or input from the sensor network with very high efficiency and accuracy. Then we propose a novel location based defense mechanism for this attack that allows the sensor network to track the adversarial localizer while preserving high degree of location privacy. In the next part of the dissertation, we investigate security issues in wireless networks through studying secure end-to-end communication in randomly deployed wireless sensor networks. We propose differentiated key pre-distribution and demonstrate the efficiency of our secure communication algorithm when location-centric and data-centric routing are performed. Next part of this dissertation presents our research in the arena of mobile ad hoc networks. We design a novel multi-tiered solution to address quality versus latency issues for content retrieval in mobile environment. This architecture provides a solution for location-centric query and response routing in content retrieval in peer to peer based mobile environment without compromising quality of response. The last part of this dissertation is on a popular trust-based service in vehicular ad hoc networks, namely, detection of Sybil attack. Detection of Sybil nodes in the highly dynamic environment of vehicular networks is a critical problem towards preventing performance and reputation system from subverting. As the final research contribution of this dissertation, we propose location clustering based technique for detection of Sybil attack in vehicular ad hoc leveraging vehicle platoon dispersions which to the best of our knowledge is the first attempt to apply transportation engineering theory into the computer science perspective of vehicular ad hoc networks.

There are a number of open research issues to investigate in location based services in wireless ad hoc networks. We identify some below here that we think are particularly timely. With the widespread acceptance of wireless sensor networks for many military and civilian applications, there are many researchers investigating and designing sensor services via the cloud. Naturally, when information from sensors (which is location specific in many cases) is integrated via a cloud, there are many emerging trust and security issues to consider. For instance, i) How can we guarantee that sensor information comes from a location where a user wants it to come from; ii) When information propagates via a mixture of wireless and wired infrastructures, how to guarantee security, trust and privacy of information are critical emerging issues here. In the realm of mobile ad hoc networks, there is an emerging paradigm of delay tolerant networking and routing among social (typically) mobile groups. How to exploit location information (based on past and future predictions) to improve routing, caching and replication performance, along with how to design new services like content sharing and emergency information propagation (for instance in a campus environment) when there are randomly moving mobile users are practical open problems in mobile ad hoc networks. With the dramatic increases in vehicular communication technologies, there are many open issues in the realm of routing protocols, improved reliability of wireless communications, design of new services to enhance driver experience etc. Furthermore, since vehicular networks integrate cyber, physical and (at times) control components, they are also instances of cyber physical systems. In vehicular networks though, mobility and consequent dynamics in node locations impose new challenges in analyzing and designing technologies and services for such networks. We believe these challenges also provide new opportunities to integrate diverse theories and disciplines (like we did in this dissertation), and we also believe that such integration among Computer Scientists, Transportation Engineers and Social/ Economic Scientists will be the norm of the near future.

BIBLIOGRAPHY

[1] Abdelzaher, T., Blum, B., Cao, Q., Evans, D., George, J., George, S., He, T., Luo, L., Son, S., Stoleru, R., et al. Envirotrack: An environmental programming model for tracking applications in distributed sensor networks. In *ICDCS04* (2004).

[2] Abowd, G. D., Dey, A. K., Brown, P. J., Davies, N., Smith, M. E., and Steggles, P. Towards a better understanding of context and context-awareness. *CHI 2000 workshop on the what who where when and how of contextawareness 4* (1999), 16.

[3] Advanced Network Technologies Division, National Institute of Standards and Technology. Mobile ad hoc networks. Retrieved February, 2012, from `http://w3.antd.nist.gov/wahn_mahn.shtml`.

[4] Afzal, S. A review of localization techniques for wireless sensor networks. *Journal of Basic and Applied Scientific Research 2*, 8 (2012), 7795–7801.

[5] Agrawal, R., and Imielinski, T. Association rules between sets of items in large databases. In *Proceedings of ACM SIGMOD* (1993), pp. 207 –216.

[6] Akbani, R., Korkmaz, T., and Raju, G. Mobile ad-hoc networks security. In *Recent Advances in Computer Science and Information Engineering*, vol. 127 of *Lecture Notes in Electrical Engineering*. Springer Berlin Heidelberg, 2012, pp. 659–666.

[7] Al-Abri, D., McNair, J., and Ekici, E. Location verification using communication range variation for wireless sensor networks. In *Proceedings of IEEE Military Communications Conference (Milcom)* (Washington D.C., October 2007).

[8] Al-Karaki, J., and Kamal, A. Routing techniques in wireless sensor networks: a survey. *IEEE Wireless Communications 11*, 6 (2004), 6–28.

[9] Alemdar, A., and Ibnkahla, M. Wireless sensor networks: Applications and challenges. In *Proceedings of the 9th International Symposium on Signal Processing and Its Applications (ISSPA)* (Sharjah, 2007), pp. 1–6.

[10] Bai, X., Ye, X., Jiang, H., and Li, J. A novel traffic information system for vanet based on location service. In *Proceedings of IEEE Conference on Networks* (India, 2008).

[11] Basu, P., Khan, N., and Little, T. A mobility based metric for clustering in mobile ad hoc networks. *icdcsw* (2001), 0413.

[12] Blyer, J. Location-based services are positioned for growth. *Wireless Systems Design* (2003), 16–20.

[13] Bo, Y., and Hurson, A. Ad hoc image retrieval using hierarchical semantic-based index. In *Advanced Information Networking and Applications, (AINA) 19th International Conference on* (2005), vol. 1, pp. 629–634.

[14] Bonomi, F., Mitzenmacher, M., Panigrahy, R., Singh, S., and Varghese, G. An improved construction for counting bloom filters. *Algorithms–ESA 2006* (2006), 684–695.

[15] Bouassida, M. S., Guette, G., Shawky, M., and Ducourthial, B. Sybil nodes detection based on received signal strength variations within vanet. *International Journal on Network Security 9*, 1 (2009), 22–33.

[16] Broder, A., and Mitzenmacher, M. Network applications of bloom filters: A survey. *Internet Mathematics 1*, 4 (2004), 485–509.

[17] Brooks, R., Ramanathan, P., and Sayeed, A. Distributed target classification and tracking in sensor networks. *Proceedings of the IEEE 91*, 8 (2003), 1163–1171.

[18] Buchenscheit, A., Schaub, F., Kargl, F., and Weber, M. A vanet-based emergency vehicle warning system. In *Proceedings of IEEE Vehicular Networking Conference (VNC)* (2009).

[19] Bulusu, N., Estrin, D., and Heidemann, J. Adaptive beacon placement. In *icdcs* (2001), Published by the IEEE Computer Society, p. 0489.

[20] Bulusu, N., Heidemann, J., and Estrin, D. Adaptive beacon placement. In *Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS)* (Phoenix, AZ, April 2001).

[21] Chan, H., Perrig, A., and Song, D. Random key predistribution schemes for sensor networks.

[22] Chan, H., Perrig, A., and Song, D. Random key predistribution schemes for sensor networks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy* (Washington, DC, USA, 2003), IEEE Computer Society.

[23] Chan, H., Perrig, A., and Song, D. Random key predistribution schemes for sensor networks. In *Proceedings of IEEE Symposium on Research in Security and Privacy* (May 2003).

[24] Chang, S., Qi, Y., Zhu, H., Zhao, J., and Shen, X. Footprint: Detecting sybil attacks in urban vehicular networks. *Parallel and Distributed Systems, IEEE Transactions on 23*, 6 (2012), 1103–1114.

[25] Cheikhrouhou, O., Kouba, A., Dini, G., and Abid, M. Riseg: a ring based secure group communication protocol for resource-constrained wireless sensor networks. *Personal and Ubiquitous Computing 15*, 8 (2011), 783–797.

[26] Chellappan, S., and Dutta, N. *Mobility in Wireless Sensor Networks.* to appear in Advances in Computers, Academic Press, 2013.

[27] Chellappan, S., Paruchuri, V., McDonald, D., and Durresi, A. Localizing sensor networks in un-friendly environments. In *IEEE Military Communications Conference (MILCOM)* (San Diego, November 2008).

[28] Chellappan, S., Paruchuri, V., McDonald, D., and Durresi, A. Localizing sensor networks in un-friendly environments. In *Military Communications Conference, 2008. MILCOM 2008. IEEE* (2008), IEEE, pp. 1–7.

[29] Chen, L., Cui, B., Shen, H., Lu, W., and Zhou, X. Efficient information retrieval in mobile peer-to-peer networks. In *Proceedings of ACM CIKM* (2009), pp. 967–976.

[30] Cheng, L., and Shakya, R. Vanet worm spreading from traffic modeling. In *Radio and Wireless Symposium (RWS), 2010 IEEE* (2010), IEEE, pp. 669–672.

[31] Chidambaram, L. M., Madria, S. K., Linderman, M., and Hara, T. Meloc: Memory and location optimized caching model for small mobile ad hoc networks. In *Proceedings of the IEEE 13th International Conference on Mobile Data Management (MDM)* (2012).

[32] Chintalapudi, K., Dhariwal, A., Govindan, R., and Sukhatme, G. Ad-hoc localization using ranging and sectoring. In *Proceedings of the Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (2004).

[33] Chlamtac, I., Conti, M., and Liu, J. J.-N. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks 1*, 1 (2003), 13–64.

[34] Chung, W., Kim, S., and Choi, J. High speed navigation of a mobile robot based on robot's experiences. In *Proc. of the JSME Annual Conference on Robotics and Mechatronics* (2006).

[35] Denney Jr, R. Traffic platoon dispersion modeling. *Journal of transportation engineering 115* (1989), 193.

[36] Dikaiakos, M., Florides, A., Nadeem, T., and Iftode, L. Location-aware services over vehicular ad-hoc networks using car-to-car communication. *IEEE Journal on Selected Areas in Communications 25(8)* (2007), 1590–1602.

[37] Dornbush, S., and Joshi, A. Streetsmart traffic: Discovering and disseminating automobile congestion using vanet's. In *Proceedings of IEEE 65th Vehicular Technology Conference, (VTC)* (2007).

[38] Douceur, J. The sybil attack. *Peer-to-peer Systems* (2002), 251–260.

[39] Du, W., Deng, J., Han, Y. S., and Varshney, P. K. A pairwise key pre-distribution scheme for wireless sensor networks. In *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)* (October 2003).

[40] Dutta, N. A peer to peer based information sharing scheme in vehicular ad hoc networks. In *IEEE 2010 Eleventh International Conference on Mobile Data Management (MDM)* (2010), pp. 309–310.

[41] Dutta, N., and Chellappan, S. Nclocs: Non-cooperative localization of sensors. *submitted to Computer Communication* (2013).

[42] Dutta, N., and Chellappan, S. A time-series clustering approach for sybil attack detection in vanets. In *Accepted in Proceedings of Advances in Vehicular Systems, Technologies and Applications (Vehicular)* (2013).

[43] Dutta, N., Kotikalapudi, R., and Bhonsle, M. A formal analysis of protocol-independent security threats in vanets. In *IEEE Students' Technology Symposium (TechSym)* (2011).

[44] Dutta, N., Kotikalapudi, R., Saxena, A., and Chellappan, S. A multi-tiered architecture for content retrieval in mobile peer-to-peer networks. In *Mobile Data Management (MDM), 2011 12th IEEE International Conference on* (2011), vol. 1, pp. 104–109.

[45] Dutta, N., Saxena, A., and Chellappan, S. Defending wireless sensor networks against adversarial localization. In *Workshop on Mobile P2P Data Management, Security and Trust in Conjunction with International Conference on Mobile Data Management (MDM)* (2010), IEEE, pp. 336–341.

[46] Dutta, N., Saxena, A., and Chellappan, S. Defending Wireless Sensor Networks Against Adversarial Localization. In *Invited Paper in International Workshop on Mobile P2P Data Management, Security and Trust (MP-DMST) in conjunction with International Conference on Mobile Data Management (MDM)* (2010).

[47] Ekici, E., McNair, J., and Al-Abrii, D. A probabilistic approach to location verification in wireless sensor networks. In *Proceedings of IEEE International Conference on Communications (ICC)* (Istanbul, June 2006).

[48] Ekici, E., Vural, S., McNair, J., and Al-Abri, D. Secure probabilistic location verification in randomly deployed wireless sensor networks. In *Ad Hoc Networks Journal (Elsevier)* (January 2007).

[49] El Defrawy, K., and Holland, G. Secure and privacy-preserving querying of content in manets. In *Proceedings of the 2012 IEEE Conference on Technologies for Homeland Security (HST)* (2012), pp. 603–608.

[50] El Defrawy, K., and Tsudik, G. Privacy-preserving location-based on-demand routing in manets. *Selected Areas in Communications, IEEE Journal on 29*, 10 (2011), 1926–1934.

[51] Enyang, X., Ding, Z., and Dasgupta, S. Source localization in wireless sensor networks from signal time-of-arrival measurements. *Signal Processing, IEEE Transactions on 59*, 6 (2011), 2887–2897.

[52] Erik, G. N., and Ketil, S. Ad hoc networks and mobile devices in emergency response - a perfect match? In *ADHOCNETS* (2010), pp. 17–33.

[53] Erkan, G., and Radev, D. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research 22*, 1 (2004), 457–479.

[54] Eschenauer, L., and Gligor, V. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security* (2002), ACM, pp. 41–47.

[55] Eschenauer, L., and Gligor, V. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security* (2002), ACM, pp. 41–47.

[56] Eschenauer, L., and Gligor, V. D. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS)* (November 2002).

[57] Farkhatdinov, I., and Ryu, J. Hybrid position-position and position-speed command strategy for the bilateral teleoperation of a mobile robot. In *Control, Automation and Systems, 2007. ICCAS'07. International Conference on* (2007), IEEE, pp. 2442–2447.

[58] Fiore, M., Casetti, C., and Chiasserini, C. Efficient retrieval of user contents in MANETs. In *IEEE Infocom* (2007).

[59] Giordano, A., Borkowski, D., and Kelley, D. Location enhanced cellular information services. In *5th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication* (1994).

[60] Giordano, A., Chan, M., and Habal, H. A novel location-based service and architecture. In *Sixth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)* (1995).

[61] Golle, P., Greene, D. H., and Staddon, J. Detecting and correcting malicious data in vanets. In *Proceedings of the First International Workshop on Vehicular Ad Hoc Networks* (October 2004), K. P. Laberteaux, R. Sengupta, C.-N. Chuah, and D. Jiang, Eds., pp. 29–37.

[62] Goyal, P., Parmar, V., and Rishi, R. Manet: Vulnerabilites, attacks, application. *International Journal of Computational Engineering and Management 11* (Jan, 2011).

[63] Grover, J., Gaur, M. S., Laxmi, V., and Prajapati, N. A sybil attack detection approach using neighboring vehicles in vanet. In *Proceedings of the 4th international conference on Security of information and networks* (2011), pp. 151–158.

[64] Gu, W., Bai, X., Chellappan, S., and Xuan, D. Network decoupling for secure communications in wireless sensor networks. In *Proceedings of IWQoS* (New Haven, June 2006).

[65] Gu, W., Dutta, N., Chellappan, S., and Bai, X. Providing end-to-end secure communications in wireless sensor networks. *IEEE Transactions on Network and Service Management (TNSM)r 8* (2011), 1–14.

[66] Gui, C., and Mohapatra, P. Power conservation and quality of surveillance in target tracking sensor networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking* (2004), ACM, pp. 129–143.

[67] Hao, Y., Tang, J., and Cheng, Y. Cooperative sybil attack detection for position based applications in privacy preserved vanets. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE* (2011), pp. 1–5.

[68] Hasegawa, T., Sekine, S., and Grishman, R. Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics* (2004), Association for Computational Linguistics, p. 415.

[69] He, T., Huang, C., Blum, B., Stankovic, J., and Abdelzaher, T. Range-free localization schemes for large scale sensor networks. In *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)* (San Diego, August 2003).

[70] He, T., Krishnamurthy, S., Stankovic, J. A., Abdelzaher, T., and et.al. Vigilnet:an integrated sensor network system for energy-efficient surveillance. In *In submission to ACM Transaction on Sensor Networks (ToSN)* (2004).

[71] Heinzelman, W., Chandrakasan, A., and Balakrishnan, H. Energy-efficient communication protocol for wireless microsensor networks. In *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on* (2002), IEEE, pp. 10–pp.

[72] Hoebeke, J., Moerman, I., Dhoedt, B., and Demeester, P. An Overview of Mobile Ad Hoc Networks: Applications and Challenges. *Journal of the Communications Network 3* (2004), 60–66.

[73] Howard, A., Mataric, M., and Sukhatme, G. Relaxation on a mesh: a formalism for generalized localization. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on* (2001), vol. 2, IEEE, pp. 1055–1060.

[74] Huang, Q., Cukier, J., Kobayashi, H., Liu, B., and Zhang, J. Fast authenticated key establishment protocols for wireless sensor networks. In *Proceedings of ACM international conference on Wireless Sensor Networks and Applications (WSNA)* (2003).

[75] Hull, B., Bychkovsky, V., Zhang, Y., Chen, K., Michel, G., Miu, A., Shih, E., Balakrishnan, H., and Samuel, M. Cartel: a distributed mobile sensor computing system, 2006.

[76] Hussain, R., Kim, ., and Oh, H. Privacy-aware vanet security: Putting data-centric misbehavior and sybil attack detection schemes into practice. In *Information Security Applications*, vol. 7690 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 296–311.

[77] I. Akyildiz, I. K. Wireless sensor and actor networks: research challenges. In *AdHoc Networks Journal - Elsevier, vol. 2, pp. 351-367* (2004).

[78] Indiveri, G., Nuchter, A., and Lingemann, K. High speed differential drive mobile robot path following control with bounded wheel speed commands. In *Robotics and Automation, 2007 IEEE International Conference on* (2007), IEEE, pp. 2202–2207.

[79] Isaac, J. T., Zeadally, S., and Camara, J. Security attacks and solutions for vehicular ad hoc networks. *Communications, IET 4*, 7 (April, 2010), 894–903.

[80] Jakubiak, J., and Koucheryavy, Y. State of the art and research challenges for vanets. In *Proceedings of the 5th IEEE Consumer Communications and Networking Conference (CCNC)* (2008).

[81] Jiang, R., Luo, J., and Wang, X. An attack tree based risk assessment for location privacy in wireless sensor networks. In *Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference on* (2012), pp. 1–4.

[82] Johnson, D., Maltz, D., Hu, Y., and Jetcheva, J. The dynamic source routing protocol for mobile ad hoc networks (DSR), 2002.

[83] Karlof, C., and Wagner, D. Secure routing in wireless sensor networks: Attacks and countermeasures. In *Proc. of 1st IEEE International Workshop on Sensor Network Protocols and Applications* (May 2003).

[84] Karp, B., and Kung, H. Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM)* (August 2000).

[85] Knuth, D. *The art of computer programming*, third ed., vol. 2. Addison-Wesley Longman Publishing Co. Inc., 1997.

[86] Ko, Y., and Vaidya, N. Location-aided routing (LAR) in mobile ad hoc networks. *Wireless Networks 6*, 4 (2000), 321.

[87] Korpipaa, P., Mantyjarvi, J., Kela, J., Keranen, H., and Malm, E. J. Managing context information in mobile devices, 2003.

[88] Kulkarni, R., Venayagamoorthy, G., and Cheng, M. Bio-inspired node localization in wireless sensor networks. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on* (2009), pp. 205–210.

[89] Kung, H., and Vlah, D. Efficient location tracking using sensor networks. In *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE* (2003), vol. 3, IEEE, pp. 1954–1961.

[90] Lab, U. V. Cvet. accessed online at http://cvet.cs.ucla.edu/index.php, March 9, 2012.

[91] Lazos, L., and Poovendran, R. Serloc: Secure range-independent localization for wireless sensor networks. In *Proceedings of ACM Workshop on Wireless Security (WiSe)* (Philadelphia, October 2004).

[92] Lazos, L., and Poovendran, R. Rope: Robust position estimation in wireless sensor networks. In *Proceedings of International symposium on Information processing in sensor networks (IPSN)* (2005).

[93] Lazos, L., and Poovendran, R. Serloc: Robust localization for wireless sensor networks. In *ACM Transactions on Sensor Networks (ToSN)* (August 2005).

[94] Lee, J., and Stinson, D. R. Deterministic key predistribution schemes for distributed sensor networks. In *Proceedings of the 11th workshop on Selected Areas in Cryptography (SAC)* (August 2004).

[95] Lee, J., and Stinson, D. R. A combinatorial approach to key predistribution for distributed sensor networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)* (March 2005).

[96] Lee, K. C., Lee, U., and Gerla, M. Survey of routing protocols in vehicular ad hoc networks. In *Proceedings of Car2Car communication consortium* (2010).

[97] Lee, W., Tseng, S., and Wang, C. Design and implementation of electronic toll collection system based on vehicle positioning system techniques. *Computer Communications 31*, 12 (2008), 2925 – 2933.

[98] Leveque, A., Pecheux, F., Louerat, M., Aboushady, H., and Vasilevski, M. Systemc-ams models for low-power heterogeneous designs: Application to a wsn for the detection of seismic perturbations. In *Proceedings of the 23rd International Conference on Architecture of Computing Systems (ARCS)* (Hannover, Germany, 2010), pp. 1–6.

[99] Li, D., Wong, K., Hu, Y., and Sayeed, A. Detection, classification and tracking of targets in distributed sensor networks. *IEEE signal processing magazine 19*, 2 (2002), 17–29.

[100] Li, F., and Wang, Y. Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine 2*, 2 (june 2007), 12 –22.

[101] LI, S., Li, X., and Wang, J. Sensor network localization based on distance reconstruction. In *Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference on* (2012), pp. 1–4.

[102] Liu, D., and Ning, P. Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In *Network and Distributed System Security Symposium (NDSS)* (San Diego, February 2003).

[103] Liu, D., and Ning, P. Establishing pairwise keys in distributed sensor networks. In *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)* (October 2003).

[104] Liu, D., Ning, P., et al. Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks. In *Proceedings of the 10th Annual Network and Distributed System Security Symposium* (2003), vol. 276, Citeseer.

[105] Liu, K., Abu-Ghazaleh, N., and Kang, K. Location verification and trust management for resilient geographic routing. In *Journal of Parallel and Distributed Computing (JPDC)* (February 2007).

[106] Lorincz, K., Malan, D., Fulford-Jones, T., Nawoj, A., Clavel, A., Shnayder, V., Mainland, G., Moulton, S., and Welsh, M. Sensor networks for emergency response: Challenges and opportunities. In *IEEE Pervasive Computing, Special Issue on Pervasive Computing for First Response* (October 2004).

[107] Lynch, J. P., and Koh, K. J. A summary review of wireless sensors and sensor networks for structural health monitoring. *Shock and Vibration Digest 38(2)* (2006), 91–128.

[108] Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., and Anderson, J. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless sensor networks and applications* (Atlanta, Georgia, USA, 2002), pp. 88–97.

[109] Marti, S., Giuli, T., Lai, K., and Baker, M. Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking* (2000), ACM, pp. 255–265.

[110] Mehta, K., Liu, D., and Wright, M. Protecting location privacy in sensor networks against a global eavesdropper. *Mobile Computing, IEEE Transactions on 11*, 2 (2012), 320–336.

[111] Melodia, T., Pompili, D., Gungor, V., and Akyildiz, I. A distributed coordination framework for wireless sensor and actor networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)* (2005).

[112] Mi, Q., Stankovic, J. A., and Stoleru, R. Practical and secure localization and key distribution for wireless sensor networks. *Ad Hoc Networks 10*, 6 (2012), 946 – 961.

[113] Mihalcea, R., Corley, C., and Strapparava, C. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the National Conference on Artificial Intelligence* (2006), vol. 21(1), p. 775.

[114] Milenkovi, A., Otto, C., and Jovanov, E. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Computer Communications 29(13 - 14)* (2006), 2521–2533.

[115] Misra, S., Bhardwaj, S., and Xue, G. ROSETTA: Robust and secure mobile target tracking in a wireless ad hoc environment. In *Proceeding of the Military Communications Conference (MILCOM)* (2006), pp. 1–7.

[116] MIT. Cartel. accessed online at http://cartel.csail.mit.edu/doku.php, March 10, 2012.

[117] Mitra, P., and Poellabauer, C. Routing in asymmetric wireless ad-hoc networks. *Next Generation Mobile Networks and Ubiquitous Computing* (2010).

[118] Mller-Levet, C., Klawonn, F., Cho, K., and Wolkenhauer, O. Fuzzy clustering of short time-series and unevenly distributed sampling points. *Advances in Intelligent Data Analysis V, Lecture Notes in Computer Science 2810* (2003), 330–340.

[119] Mondal, A., and Kitsuregawa, M. Privacy, security and trust in p2p environments: A perspective. In *Proceedings of 17th International Conference on Database and Expert Systems Applications* (Krakow, Poland, 2006).

[120] Ngai, E.-H., and Rodhe, I. On providing location privacy for mobile sinks in wireless sensor networks. *Wireless Networks 19*, 1 (2013), 115–130.

[121] Niculescu, D., and Nath, B. Ad hoc positioning system (aps) using aoa. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)* (April, San Francisco 2003).

[122] of Transportation, U. D., and Hamilton, B. A. Vehicle infrastructure integration, November, 2006.

[123] Panja, B., and Madria, S. K. An energy and communication efficient group key in sensor networks using elliptic curve polynomial. In *Proceedings of the 6th international conference on Ad-hoc, mobile and wireless networks* (2007).

[124] Parno, B., Perrig, A., and Gligor, V. Distributed detection of node replication attacks in sensor networks. In *Proceedings of IEEE Symposium on Security and Privacy* (Oakland, May 2005).

[125] Pathak, V., Yao, D., and Iftode, L. Securing location aware services over vanet using geographical secure path routing. In *Proceedings of IEEE International Conference on Vehicular Electronics and Safety* (2008), pp. 346–353.

[126] Pathirana, P., Bulusu, N., Savkin, A. V., and Jha, S. Node localization using mobile robots in delay-tolerant sensor networks. In *IEEE Transactions on Mobile Computing (TMC) vol. 4, no.3, 2005, pp. 285-296* (May 2005).

[127] Pelusi, L., Passarella, A., and Conti, M. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. *Communications Magazine, IEEE 44*, 11 (2006), 134–141.

[128] Portugal, C. Drive-in: Content delivered to your car. accessed online at http://drive-in.cmuportugal.org/, March 9, 2012.

[129] Priyantha, N., Miu, A., Balakrishnan, H., and Teller, S. The cricket compass for context-aware mobile applications. In *Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom)* (July 2001).

[130] Priyantha, N. B., Balakrishnan, H., Demaine, E., and S.Teller. Mobile-assisted localization in wireless sensor networks. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)* (Miami, March 2005).

[131] R., Z. Adaptive vehicle alarm detection and reporting system, 09 1993.

[132] Rahbari, M., and Jamali, M. Efficient detection of sybil attack based on cryptography in vanet. *International Journal of Network Security & Its Applications 3* (2011).

[133] Rai, V., and Mahapatra, R. Lifetime modeling of a sensor network. In *Proceedings of the Design, Automation and Test in Europe 2005 (DATE)* (2005).

[134] Rajabhushanam, C., and Kathirvel, A. Survey of wireless manet application in battlefield operations. *International Journal of Advanced Computer Science and Applications 2*, 1 (2011).

[135] Raya, M., Papadimitratos, P., and Hubaux, J.-P. Securing vehicular communications. *IEEE Wireless Communications 13*, 5 (October, 2006), 8–15.

[136] Reddy, Y., Durand, J., and Kafle, S. Detection of packet dropping in wireless sensor networks. In *Proceedings of the 2010 Seventh International Conference on Information Technology: New Generations (ITNG)* (2010), pp. 879 – 884.

[137] Repantis, T., and Kalogeraki, V. Data dissemination in mobile peer-to-peer networks. In *Proceedings of ACM MDM* (2005).

[138] Rizvi, S., Olariu, S., Weigle, M., and Rizvi, M. A novel approach to reduce traffic chaos in emergency and evacuation scenarios. In *Proceedings of IEEE 66th Vehicular Technology Conference (VTC)* (2007).

[139] Sampigethaya, K., Huang, L., Li, M., Poovendran, R., Matsuura, K., and Sezaki, K. Caravan: Providing location privacy for vanet. In *in Embedded Security in Cars (ESCAR)* (2005).

[140] Sampigethaya, K., Li, M., Huang, L., and Poovendran, R. Amoeba: Robust location privacy scheme for vanet. *IEEE Journal on Selected Areas in Communications 25(8)* (2007), 1569–1589.

[141] Sastry, N., Shankar, U., and Wagner, D. Secure verification of location claims. In *Proceedings of the ACM workshop on Wireless security (WiSe)* (San Diego, 2003).

[142] Savvides, A., Han, C.-C., and Srivastava, M. B. Dynamic fine-grained localization in ad-hoc networks of sensors,. In *Proceedings of ACM MobiCom* (2001).

[143] Schugers, C., and Srivastava, M. Energy efficient routing in wireless sensor networks. In *Proceedings of Milcom* (October 2001).

[144] Schwartz, R., Barbosa, R., Meratnia, N., Heijenk, G., and Scholten, H. A simple and robust dissemination protocol for vanets. In *Proceedings of European Wireless Conference (EW)* (2010).

[145] Sichitiu, M., and Ramadurai, V. Localization of wireless sensor networks with a mobile beacon. In *Proceedings of IEEE International Conference on Mobile AdHoc and Sensor Systems (MASS)* (October, Fort Lauderdale 2004).

[146] Soyoung, P., Aslam, B., Turgut, D., and Zou, C. Defense against sybil attack in vehicular ad hoc network based on roadside unit support. In *Military Communications Conference, 2009. MILCOM 2009. IEEE* (October 2009), pp. 1 – 7.

[147] Strang, T., and Linnhoff-Popien, C. A context modeling survey. *Graphical Models* (2004), 18.

[148] System, I. T., and for Europe, S. Ertico - its europe. accessed online at http://www.ertico.com/, March 10, 2012.

[149] Tangpong, A., Kesidis, G., yuan Hsu, H., and Hurson, A. Robust sybil detection for manets. In *Proceedings of 18th Internatonal Conference on Computer Communications and Networks (ICCCN)* (2009), pp. 1–6.

[150] Testbed, U. D. Dome. accessed online at http://prisms.cs.umass.edu/dome/, March 10, 2012.

[151] Wang, J. Electrochemical sensors for environmental monitoring: A review of recent technology. *A Note from the U.S. Environmental Protection Agency (EPA)* (2004).

[152] Wang, X., Chellappan, S., Gu, W., Yu, W., and Xuan, D. Search-based physical attacks in sensor networks. In *Proceedings of IEEE ICCCN* (October 2005).

[153] White, J., Thompson, C., Turner, H., Dougherty, B., and Schmidt, D. C. Wreckwatch: Automatic traffic accident detection and notification with smartphones. *Mobile Networks and Applications 16*, 3 (2011), 285–303.

[154] wikipedia.org. Location based service. accessed online at http://en.wikipedia.org/wiki/Location-based_service, April 2, 2013.

[155] Xiao, B., Yu, B., and Gao, C. Detection and localization of sybil nodes in vanets. In *Workshop on Dependability issues in wireless ad hoc networks and sensor networks* (2006), pp. 1–8.

[156] Xie, H., Kulik, L., and Tanin, E. Privacy-aware traffic monitoring. *Intelligent Transportation Systems, IEEE Transactions on 11*, 1 (2010), 61–70.

[157] Xu, Y., Heidemann, J., and Estrin, D. Geography-informed energy conservation for ad hoc routing. In *Proceedings of the 7th annual international conference on Mobile computing and networking* (2001), ACM, pp. 70–84.

[158] Yan, P., Jiao, Y., Hurson, A., and Potok, T. E. Semantic-based information retrieval of biomedical data. In *Proceedings of the 2006 ACM symposium on Applied computing* (2006), pp. 1700–1704.

[159] Yang, H., and Sikdar, B. A protocol for tracking mobile targets using sensor networks. In *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on* (2003), IEEE, pp. 71–81.

[160] Yang, Y., and Bagrodia, R. Evaluation of vanet-based advanced intelligent transportation systems. In *Proceedings of the 6th ACM international workshop on VehiculAr InterNETworking (VANET)* (Beijing, China, 2009), pp. 3–12.

[161] Yang, Z., Ekici, E., and Xuan, D. A localization-based anti-sensor network system. In *Proceedings of IEEE INFOCOM 2007 Symposia* (Anchorage, May 2007).

[162] Yick, J., Mukherjee, B., and Ghosal, D. Wireless sensor network survey. *Computer Networks 52*, 12 (2008), 2292 – 2330.

[163] Yousefi, S., Mousavi, M., and Fathy, M. Vehicular ad hoc networks (vanets): Challenges and perspectives. In *Proceedings of the 6th International Conference on ITS Telecommunications* (2006).

[164] Yuanguo, C., and Guohui, L. A Bayesian Framework for Target Tracking in Sensor Networks. In *Control Conference, 2007. CCC 2007. Chinese* (2007), pp. 328–331.

[165] Zhang, Y., Liu, W., and Fang, Y. Secure localization in wireless sensor networks. In *Proceedings of IEEE Military Communication Conference (Milcom)* (October 2005).

[166] Zhang, Y., Liu, W., Fang, Y., and Wu, D. Secure localization and authentication in ultra-wideband sensor networks. *IEEE Journal on Selected Areas in Communications 24*, 4 (2006), 829–835.

[167] Zhang, Y., Zhao, J., and Cao, G. Roadcast: A popularity aware content sharing scheme in vanets. In *Proceedings of IEEE ICDCS* (Canada, June 2009).

[168] Zhao, J., and Cao, G. VADD: Vehicle-assisted data delivery in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology 57*, 3 (2008), 1910–1922.

[169] Zhao, Z., Hu, H., Ahn, G.-J., and Wu, R. Risk-aware mitigation for manet routing attacks. *Dependable and Secure Computing, IEEE Transactions on 9*, 2 (2012), 250–260.

[170] Zheng, Y., and Chen, Y. Adcontrep: A privacy enhanced reputation system for manet content services. In *Ubiquitous Intelligence and Computing*, vol. 6406 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 414–429.

[171] Zhou, G., He, T., Krishnamurthy, S., and Stankovic, J. Models and solutions for radio irregularity in wireless sensor networks. *ACM Transactions on Sensor Networks 2* (2006), 221–262.

[172] Zhou, G., Li, C., and Cheng, P. Unmanned aerial vehicle (uav) real-time video registration for forest fire monitoring. In *Proceedings of IEEE International Geoscience and Remote Sensing Symposium* (2005), pp. 1803–1806.

[173] Zhou, T., Choudhury, R., Ning, P., and Chakrabarty, K. P2DAP-Sybil Attacks Detection in Vehicular Ad Hoc Networks. *Selected Areas in Communications, IEEE Journal on 29*, 3 (March 2011), 582 – 594.

[174] Zhu, S., Xu, S., Setia, S., and Jajodia, S. Establishing pairwise keys for secure communication in ad hoc networks: a probabilistic approach. In *Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP)* (November 2003).

VITA

Neelanjana Dutta was born and brought up in the city of Calcutta, now known as Kolkata, in West Bengal, India. She went to Ramakrishna Sarada Mission Sister Nivedita Girls' School for her primary and secondary schooling. In 2008, she received B.Tech in Information Technology from Institute of Engineering and Management, Kolkata. In the Fall of 2008, she joined Missouri University of Science and Technology to pursue her doctoral studies in Computer Science. She had the privilege of working with Dr. Sriram Chellappan, who advised her during the doctoral research in the field of wireless networks. She received fellowship from University Transportation Center at Missouri University of Science and Technology for her research. In August 2013, she received her PhD in Computer Science from Missouri University of Science and Technology. Her major research interests include security and performance issues in wireless ad hoc networks, specially mobile and vehicular ad hoc networks.