

1
2003
54803935



This is to certify that the
thesis entitled

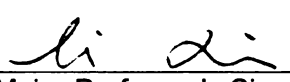
Building A Location-Based Prioritized Overlay Multicast in Ad-hoc Environments

presented by

Yunhao Liu

has been accepted towards fulfillment
of the requirements for the

M.S degree in CSE



Major Professor's Signature

6/27/2003

Date

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.
MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

**BUILDING A LOCATION-BASED PRIORITIZED OVERLAY MULTICAST IN AD-
HOC ENVIRONMENTS**

BY

Yunhao Liu

A THESIS

**Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of**

MASTER OF SCIENCE

Department of Computer Science and Engineering

2003

ABSTRACT
BUILDING A LOCATION-BASED OVERLAY MULTICAST IN AD-HOC ENVIRONMENTS

BY

Yunhao Liu

Overlay multicast in mobile ad-hoc environments are finding newer applications everyday. Although overlay multicast is not as efficient as IP-based multicast, they have the advantage of being easy to implement. In many applications, some participating nodes might be members of more than one overlay tree or may wish to build a temporary tree in order to perform certain important tasks. The priority of these trees can be defined by the importance of the service. For the success of such applications, it is necessary that nodes belonging to more than one tree are smart enough to ignore incoming messages from members in low priority trees while they are listening to member from a higher priority tree. In this thesis, we present a Location-based Prioritized Overlay Multicast (L-POM) that builds priority trees with certain nodes carrying important tasks in overlay networks, and rearranges low priority trees using node location information whenever some nodes temporarily move to a high priority network. To obtain accurate location information, we design and implement an indoor location sensing system, LANDMARC. LANDMARC can improve the overall accuracy of locating objects by utilizing the concept of reference tags. Our experimental analysis demonstrates that active RFID is a viable and cost-effective candidate for indoor location sensing, and can be an effective support to L-POM. We further study the feasibility of L-POM by identifying a suitable unicast (ad-hoc) routing protocol, exploring to use location information to build more efficient trees, and studying the impact of density of wireless nodes, packet size, and fragmentation on performance.

to my grandmother

ACKNOWLEDGMENT

First and foremost, I would like to deeply thank my thesis advisors Dr. Lionel Ni and Dr. Li Xiao for their continual guidance and support during this work. I would like to thank my grandmother and my parents for their unconditional love and encouragement. I am also grateful to the members of the ELANS lab especially Abhishek for exchanging ideas and helping me throughout.

TABLE OF CONTENTS

1	INTRODUCTION.....	1
1.1	MOTIVATION.....	1
1.2	OBJECTIVES	5
1.3	ORGANIZATION.....	6
2	RELATED WORK.....	7
2.1	RELATED WORK IN OVERLAY MULTICAST IN AD-HOC ENVIRONMENTS.....	7
2.2	RELATED WORK IN LOCATION SENSING SYSTEMS.....	9
3	LANDMARC: LOCATION-SENSING USING RFID	14
3.1	RFID TECHNOLOGY AND OUR FIRST ATTEMPT.....	15
3.1.1	<i>Overview</i>	15
3.1.2	<i>Range and Power Levels</i>	17
3.1.3	<i>A Triangulation Approach</i>	18
3.1.4	<i>Our first attempt</i>	20
3.2	LANDMARC APPROACH.....	23
3.2.1	<i>System Setup</i>	24
3.2.2	<i>Methodology</i>	25
3.3	EXPERIMENTAL RESULTS AND PERFORMANCE EVALUATION	27
3.3.1	<i>Effect of the Number of Nearest Neighbors</i>	29
3.3.2	<i>Influence of the Environmental Factors</i>	30
3.3.3	<i>Effect of the Number of Readers</i>	32
3.3.4	<i>Effect of Placement of Reference Tags</i>	34
4	LOCATION-BASED PRIORITIZED OVERLAY MULTICAST	41
5	OVERLAY MULTICAST PERFORMANCE EVALUATION.....	47
5.1	UNICAST PROTOCOL IDENTIFICATION	48
5.2	LOCATION-BASED MULTICAST TREE	56
5.3	DENSITY OF MOBILE NODES.....	60
5.4	FILE FRAGMENTATION	62
6	CONCLUSION AND FUTURE WORK	65
6.1	CONCLUSION.....	65
6.2	FUTURE RESEARCH FOR LOCATION SENSING	66
6.3	FUTURE WORK FOR LOCATION BASED OVERLAY MULTICAST	67
7	BIBLIOGRAPHY	69

LIST OF TABLES

Table1: Location Sensing System Criteria	12
Table 2: Simulation Parameters for protocol comparison	50
Table 3: Simulation parameters for testing location aware tree.	57
Table 4: Simulation parameters for testing effect of density.	60
Table 5: Simulation parameters for testing effect of density.	62

LIST OF FIGURES

Figure 1.1: An inefficient overlay causes a lot of unnecessary traffic.....	2
Figure 2.1: Signal Strength: 802.11b	10
Figure 2.2: Active Bat [30]	11
Figure 2.3: SpotON [31]	12
Figure 3.1: RFID System Components.....	16
Figure3.2: Active tag System.....	17
Figure 3.3: A Triangulation Approach.....	19
Figure 3.4: Power Level in Different Environment.....	20
Figure 3.5: The RFID reader and tag used in our prototype system.....	21
Figure 3.6: Placement of 9 readers with two different ranges and the sub-regions.....	22
Figure 3.7: Placements of RF Readers and Tags	28
Figure 3.8: Cumulative percentile of error distance for k from 2 to 5.....	29
Figure 3.9: Cumulative percentile of error distance in the daytime and at night.....	30
Figure 3.10: Placement of RF readers and tags (placement configuration 2).....	31
Figure 3.11: Cumulative percentile of error distance between two tracking tag placement configurations in Fig 3.7 and Fig 3.10.....	32
Figure 3.12: Cumulative percentile of error distance for 3 and 4 RF readers.	33
Figure 3.13: A physical partition to separate reference tags c and f from others	34
Figure 3.14: More reference tags are used.....	36
Figure 3.15: Two higher density, comparing with those in Fig. 4, placements of reference tags	37
Figure 3.16: Cumulative percentile of error distance with a higher reference tag density	37

Figure 3.17: Two lower density, comparing with those in Fig. 4, placements of reference tags	38
Figure 3.18: Cumulative percentile of error distance with a lower reference tag density	39
Figure 4.1: Formation of high priority tree and rearrangement of the old tree.....	43
Figure 4.2: Role-based partitioning of overlay network.....	46
Figure 5.1: A Changed Physical topology	49
Figure 5.2: Topologies used for testing different protocols.....	51
Figure 5.3: Comparison of average completion time for packet size of 512.....	51
Figure 5.4: Comparison of average completion time for packet size of 1024.....	52
Figure 5.5: Comparison between AODV & DSR (Average Completion time)	52
Figure 5.6: Performance in terms of Drop Ratio	53
Figure 5.7: Performance in terms of Protocol Ratio	54
Figure 5.8: Comparison between DSR & AODV (FTP Completion time).....	54
Figure 5.9: Drop Ratio comparison for DSR & AODV (Twin Topology).....	55
Figure 5.10: Performance in terms of Protocol Ratio for twin topology	55
Figure 5.11: Overlay tree without any location information.	57
Figure 5.12: Location aware overlay tree.	57
Figure 5.13: Actual Physical positions of nodes in one particular (random) Scenario.....	58
Figure 5.14: Performance of location aware overlay tree and an overlay tree without any location information about member nodes.	59
Figure 5.15: Worst case scenarios	59
Figure 5.16: Performance Comparison for different number of nodes and areas.....	61
Figure 5.17: Performance Comparison for different number of nodes and areas.....	61

Figure 5.18: 200K file transferred in fragments. 63

Figure 5.19: 100KB file transferred in fragments..... 64

1 Introduction

1.1 Motivation

The concept of multicast communication has been around for more than a decade. Video conferencing, many peer-to-peer networks, web cast, etc employ multicast communication. Multicast networks can be implemented at the network (IP) layer or at the application layer (Overlay). The current implementation of IP networks provides very limited provision for forming multicast networks. The scene in the ad-hoc wireless domain is further convoluted. Due to its wireless nature, the mobile nodes have a limited coverage range and hence, multiple network hops may be needed for one node to exchange information with another mobile node. Many network multicast routing protocols have been proposed for MANET [5][6][7][21]. These protocols require that member and non-member nodes maintain route information for providing multicast service. To make things more challenging, node movement and the random MANET membership of nodes, changes the state information from time to time. Even a fast moving non-member node can trigger updates in the routing tables at several other nodes. Also, since the network is dynamic, any node can enter or leave the network at random times triggering a state information update at the nodes. Thus the multicast protocols

have to respond to network dynamics in addition to the group dynamics. As a result, multicast in MANET suffers from high overhead and robustness. Network multicast has not been widely employed by most Internet service providers and hence, major portions of the internet are still not capable to even perform basic multicast functionality.

Overlay multicast can overcome most of the setback described above. In application layer (overlay) multicast, unlike IP multicast where the routers in the network perform the task of duplication and forwarding, the participating hosts perform the multicast functionality. While this approach may sound pretty straightforward, it does have its own set of drawbacks. The system does not scale well when the member group is very large. Data exchange between member nodes requires traversing other member nodes hence it increases the latency. And, without an efficient mechanism on building the overlay tree, a lot of unnecessary traffic could be incurred, which will further increase the latency.

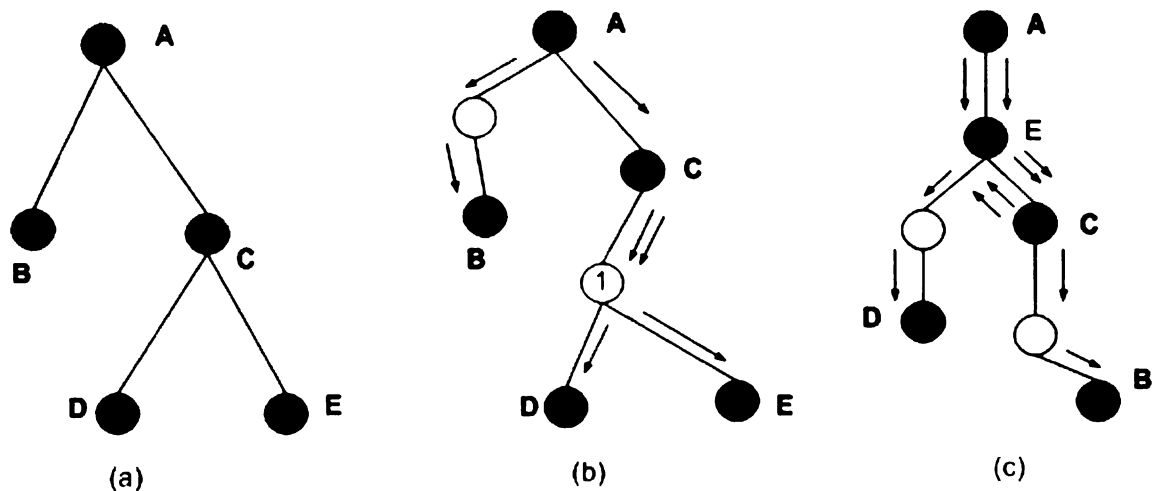


Figure 1.1: An inefficient overlay causes a lot of unnecessary traffic

In the Figure 1.1 (a), it can be seen that data from node A has to traverse through node C in order to reach node D and E. As the number of members' increases, the latency

would increase. Further, it is possible that more than one edge of the overlay tree maps on to the same physical link thus making it possible for traffic to get duplicated over the same link. Referring to Figure 2.1 (b), node C will make separate unicast connections to node D and E. As a result, the link between node C and non-member node 1 would be carrying the same data twice. Even worse, since all the nodes in the multicast group are moving, the logical overlay tree may be mismatching the physical position at some time as shown in Figure 2.1 (c). In this case, the overlay shown in Figure 2.1(a) is far from efficient. We can see the same packets would traverse the E-C link for four times!

However, these weaknesses have not reduced the attention received by overlay networks in MANET. Due to its ease of implementation and flexibility to adapt, overlay networks are finding many practical applications. In some situations, participating nodes may be able to carry out several different functions and as a result can be associated with more than one overlay tree. There may be times when some member nodes may decide to form a short-term network (tree) to perform certain important tasks. For example, a hospital may have an overlay networks for doctors, nurses and other medical personnel. During a medical emergency, certain doctors and medical attendants may wish to form their own temporary network to attend the emergency situation. The different trees can have different levels of priority depending on the importance of the service they perform. In case of our hospital example, the members of the emergency network would give higher priority to messages from members belonging to the emergency network while ignoring messages from other low-priority networks.

In this thesis, we propose a prototype of Location-based Priority Overlay Multicast (L-POM) for MANET, which can be used in a class of applications discussed above. Our proposed L-POM builds priority trees with certain nodes carrying important tasks in overlay networks, and rearranges low priority trees using node location information whenever some nodes temporarily move to a high priority network. To obtain accurate location information, we design and implement an indoor location sensing system: LANDMARC. LANDMARC can improve the overall accuracy of locating objects by utilizing the concept of reference tags. Our experimental analysis demonstrates that active RFID is a viable and cost-effective candidate for indoor location sensing, and can be an effective support to L-POM. In a location-aware overlay trees, the logical topology would tend to resemble the physical topology. As a result, neighboring member nodes in the overlay tree would be physically close to each other thus reducing the number of hops between them. Our simulation results show the effectiveness of L-POM with overlay trees that are built by using location information. L-POM can be applied to a wide spectrum of applications where setting up infrastructure based systems – wireless access points with wired connection to the network – is difficult and where the organizers would like to have a role-based partition in their network. Here is another application example. In public events, such as in a big-ten basketball game and Olympic games, some of the security persons may need to form a priority network to handle an emergency situation (see Figure 1.2).

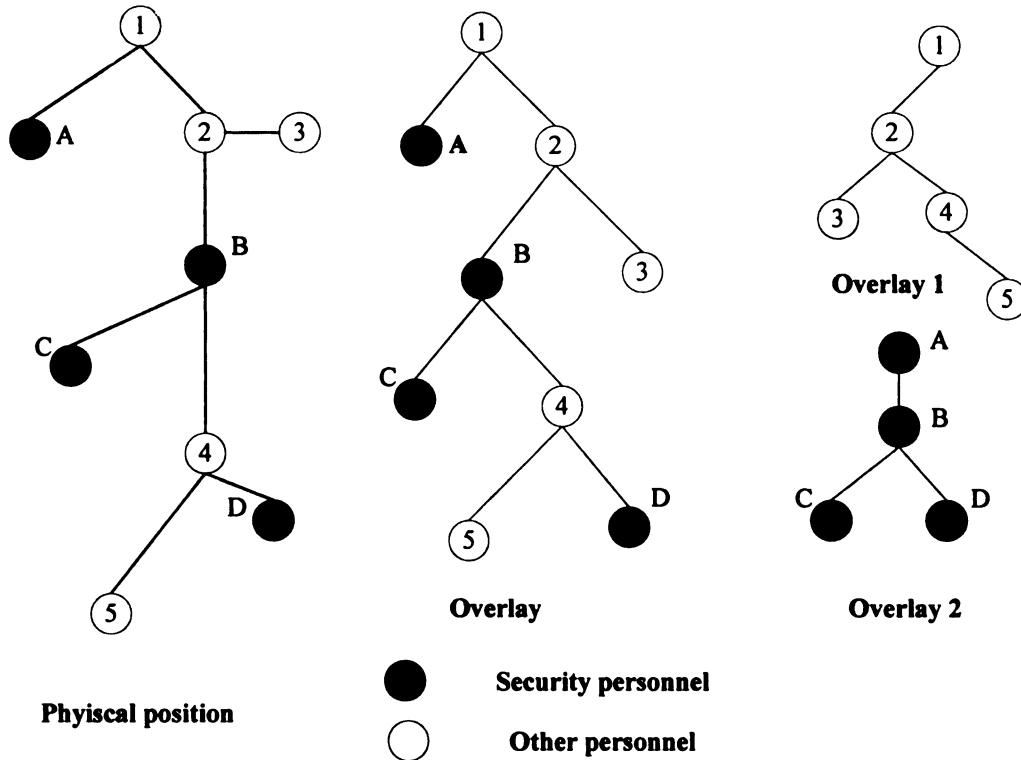


Figure 1.2: Prioritized overlay network

1.2 Objectives

We have three objectives in this thesis. Firstly, we need to evaluate a practical location sensing approach to provide location information of each member node so as to help the system building an efficient overlay multicast tree. Secondly, we examine the feasibility of the location-based prioritized overlay multicast approach to build priority trees with certain nodes carrying important tasks in overlay networks, and rearranges low priority trees whenever some nodes temporarily move to a high priority network. Thirdly, we experimentally study the feasibility of L-POM by identifying a suitable unicast (ad-hoc) routing protocol, exploring to use location information to build more efficient trees, and studying the impact of density of wireless nodes, packet size, and fragmentation on performance.

1.3 Organization

Chapter 2 gives problem statement and discusses related work. In Chapter 3 we present the prototype of the supporting system, LANDMARC, which is a location sensing approach using RFID. We then describe the design of the location-based prioritized overlay multicast (L-POM) in Chapter 4. Chapter 5 shows the experimental methodology and simulation results to evaluate L-POM. Chapter 6 presents the conclusion and future work.

Related Work

2.1 Related Work in Overlay Multicast in Ad-hoc Environments

Overlay multicast in wired networks has been a popular area of research for the last few years. There have been several papers addressing the issue of forming an efficient overlay multicast tree. NICE [2] presents an application layer multicast protocol that arranges the end host into a hierarchy (layers) which defines the multicast overlay data path. Layers are numbered sequentially; starting from zero for the lowest layer. Adjacent hosts in each layer form a cluster and each cluster has a leader. The cluster leaders together form the next higher layer. Again repeating the same algorithm the next higher layer would be created until there is only one leader host left, which would act as the root of the tree. Two metrics are used to analyze the goodness of data paths in an overlay multicast – Stress and Stretch. The Stress at a link is defined as the number of identical copies of a packet carried by that link. Stretch measures the relative increase in delay incurred by the overlay path between pairs of members with respect to direct unicast path. However, the tradeoff for Stress and Stretch for NICE [2] has been studied for wired networks only.

Ad hoc multicast routing protocol (AMRoute[11]) makes use of user-multicast trees and dynamic logical cores to build a robust multicast network. It creates per group multicast distribution tree using unicast tunnels between group members. The bidirectional tunnels are created between member nodes that are close to each other (neighbors in the multicast tree) to form a virtual mesh. From this mesh, a subset of links

is used to create a multicast distribution tree. AMRoute can run seamlessly over several separate domains that may be using different unicast protocols. A packet sent between logical neighboring nodes is physically sent on a unicast tunnel and may possibly pass through several intermediate nodes (routers). The path taken by the unicast tunnel can change with the changing network topology without affecting the user multicast tree.

PAST-DM [1], Progressively Adaptive Sub-Tree in Dynamic Mesh, is an overlay multicast protocol defined for mobile ad-hoc networks. PAST-DM tries to eliminate redundant physical links so that the overall bandwidth consumption of the multicast session is reduced. Unlike AMRoute [11], which builds its shared tree route using static virtual mesh, the virtual mesh in PAST-DM constantly adapts to the changes in the underlying network topology. PAST-DM yields a stable tree quality as the update period is increased. This may lead to higher overhead for maintaining a good stable tree.

Location Guided Tree (LGT) [8] construction scheme builds overlay multicast tree using geometric distance between member nodes as the heuristic of link costs. Two tree construction algorithms are proposed: greedy k-ary tree construction (LGK) and Steiner tree construction (LGS). With LGK, source node selects k nearest neighbors as its children, and partitions the remaining nodes according to their distance to the children nodes. LGS constructs the Steiner tree using link costs as their geometric lengths. Each child node is then responsible for packet delivery to its own subgroup using the same algorithm. The paper proposes a novel idea of building location-aware overlay trees. However, the frequency of update messages is not clearly mentioned. Since nodes are under constant motion, high mobility would drastically increase the overhead of update messages. Our approach tries to be more optimal and simple to implement. We propose

to update the location information between new sessions. A session can either be defined by a fixed amount of time or by the amount of data transferred by the source node.

2.2 Related Work in Location Sensing Systems

A number of wireless technologies have been used for indoor location sensing.

Infrared. Active Badge, developed at Olivetti Research Laboratory (now AT&T Cambridge), used diffuse infrared technology [26] to realize indoor location positioning. The line-of-sight requirement and short-range signal transmission are two major limitations that suggest it to be less than effective in practice for indoor location sensing.

IEEE 802.11. RADAR is an RF based system for locating and tracking users inside buildings [27], using a standard 802.11 network adapter to measure signal strengths at multiple base stations positioned to provide overlapping coverage in a given area. This system combines empirical measurements and signal propagation modeling in order to determine user location thereby enabling location-aware services and applications. The major strengths of this system are that it is easy to set up, requires few base stations, and uses the same infrastructure that provides general wireless networking in the building. The difficulty is that the object being tracked must be supported by a Wave LAN NIC, which may be impractical on small or power constrained devices. In most cases to date, the overall accuracy of the systems, using 802.11 technologies, is not as optimal as desired. For example, RADAR's implementation can place objects to within about 3

The difficulty is that the object being tracked must be supported by a Wave LAN NIC, which may be impractical on small or power constrained devices. In most cases to date, the overall accuracy of the systems, using 802.11 technologies, is not as optimal as desired. For example, RADAR's implementation can place objects to within about 3 meters of their actual position with 50 percent probability, while the signal strength lateration implementation has 4.3-meter accuracy at the same probability level [28].

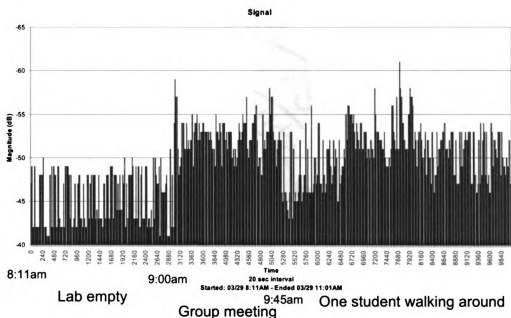


Figure 2.1 Signal Strength: 802.11b

Ultrasonic. The Cricket Location Support System [29] and Active Bat location system [30] are two primary examples that use the ultrasonic technology. Normally, these systems use an ultrasound time-of-flight measurement technique to provide location information. Most of them share a significant advantage, which is the overall accuracy. Cricket for example can accurately delineate 4x4 square-foot regions within a room while Active Bat can locate Bats to within 9cm of their true position for 95 percent of the

measurements. However, the use of ultrasonic this way requires a great deal of infrastructure in order to be highly effective and accurate, yet the cost is so exorbitant that it is inaccessible to most users.

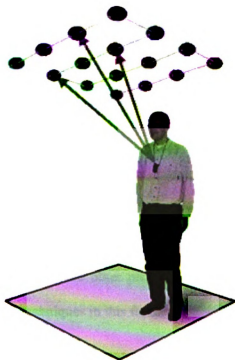


Figure 2.2 Active Bat [30]

RFID. One well-known location sensing systems using the RFID technology is SpotON [31]. SpotON uses an aggregation algorithm for three dimensional location sensing based on radio signal strength analysis. SpotON researchers have designed and built hardware that will serve as object location tags. In the SpotON approach, objects are located by homogenous sensor nodes without central control. SpotON tags use received radio signal strength information as a sensor measurement for estimating inter-tag distance. However, a complete system has not been made available as of yet.

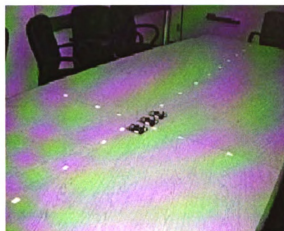


Figure 2.3 SpotON [31]

The above are popular technologies for indoor location sensing. Some other technologies, such as ultra-wideband [32], are also being investigated. The choice of technique and technology significantly affects the granularity and accuracy of the location information. There are some other projects using the above technologies. In Table 1 we list some typical techniques in this field.

Table1: Location Sensing System Criteria

System Name	Technology	Accuracy	Cost	Limitations and Disadvantages
Active Badges	Diffuse infrared Cellular proximity	Room Size	Administration costs, cheap tags and bases	Sunlight and fluorescent interference with infrared
Active Bat	Ultrasonic Time-	9cm(95%)	Administration	Required

	of-flight measurement		costs, cheap tags and bases	ceiling sensor grids, costly
Cricket	Ultrasonic Time-of-flight measurement and proximity	4x4ft. Regions (approximately 100%)	\$10 beacons and receivers	Each antenna has a narrow cone of influence, expensive
RADAR	802.11 RF scene analysis and triangulation	3-4.3m(50%)	802.11 network installation, ≈\$100 wireless NICs	Wireless NICs required
SpotON	RFID	3 meters (using commercial components)	\$30 per tag, no infrastructure	SpotON Product not available

Due to the lack of availability of cost-effective indoor location sensing products, we have tried both infrared and 802.11b products. Neither was satisfactory for the above reasons. We do not intend to build our own devices due to cost constraint. We selected commercially available RFID devices as our prototyping technology, which is described below.

3 LANDMARC: Location-Sensing Using RFID

The proliferation of wireless technologies, mobile computing devices, and the Internet has fostered a growing interest in location-aware systems and services. Many applications need to know the physical location of objects. Over the years, many systems have addressed the problem of automatic location-sensing. Triangulation, scene analysis, and proximity are the three principal techniques for automatic location-sensing [28]. One of the most well known location-based systems is the Global Positioning System (GPS), a satellite-based navigation system made up of a network of 24 satellites placed into orbit [33]. GPS is widely used to track moving objects located outdoors. However, GPS, as it is satellite dependent, has an inherent problem of accurately determining the location of objects located inside buildings. Different approaches have been proposed and tested for their effectiveness and utilities in order to achieve the ability to locate object whatever in or out of the buildings.

At present, there are several types of location-sensing systems, each having their own strengths as well as limitations. Infrared, 802.11, ultrasonic, and RFID are some

examples of these systems. We are interested in using commodity off-the-shelf products. The results of our comparative studies reveal that there are several advantages of the RFID technology. The no contact and non-line-of-sight nature of this technology are significant advantages common among all types of RFID systems. All RF tags can be read despite extreme environmental factors, such as snow, fog, ice, paint, and other visually and environmentally challenging conditions. They can also work at remarkable speeds. In some cases, tags can be read in less than a 100 milliseconds. The other advantages are their promising transmission range and cost-effectiveness.

3.1 RFID Technology and Our First Attempt

RFID is a means of storing and retrieving data through electromagnetic transmission to a RF compatible integrated circuit, and is now being seen as a radical means of enhancing data handling processes [34].

3.1.1 Overview

A RFID system has several basic components or technical characteristics including a reader with an antenna, a tag, and the communication between them (Figure 3.1)

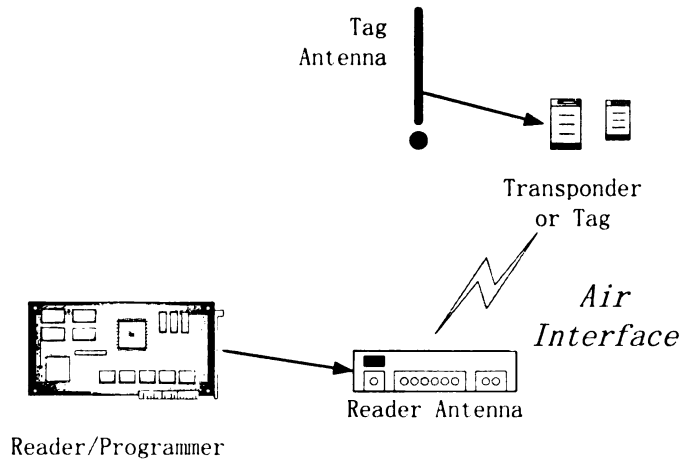


Figure 3.1: RFID System Components

The reader can read and write data to RFID tags, and the tags transmit data to a reader. The whole system uses a defined radio frequency and protocol to transmit and receive data.

RFID tags are categorized as either passive or active. Passive RFID tags may operate either with or without a battery. They reflect the RF signal transmitted to them from a reader and add information by modulating the reflected signal. Passive tags are much lighter and less expensive than active tags, and offer a virtually unlimited operational lifetime. However, their read ranges are very limited. Active tags contain both a radio transceiver and battery to power the transceiver (Figure 3.2). Since there is an onboard radio on the tag, active tags have more range than passive tags. For instance, the products we use in the experiment are active tags, which have a read range of 150 feet. If necessary, this range can be increased to 1000 feet with special antenna.

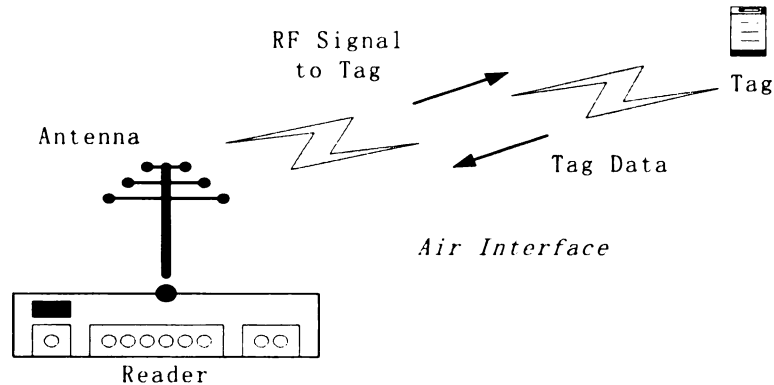


Figure3.2: Active tag System.

3.1.2 Range and Power Levels

The range that can be achieved in an RFID system is essentially determined by [35]:

- The power available at the reader/interrogator to communicate with the tag(s)
- The power available within the tag to respond
- The environmental conditions and structures, the former being more significant at higher frequencies including signal to noise ratio

The field or wave delivered from an antenna extends into the space surrounding it and its strength diminishes with respect to distance. The antenna design will determine the shape of the field or propagation wave delivered, so that range will also be influenced by the angle subtended between the tag and antenna. In space free of any obstructions or absorption mechanisms the strength of the field reduces in inverse proportion to the square of the distance. For example, consider an isotropic point source fed by a transmitter of P_t watts. At an arbitrary, large distance d from the source, the radiated power is uniformly distributed over the surface area of a sphere of radius. Thus, the received signal power at distance is given by [36]

$$P_r = \frac{A_e G_t P_t}{4\pi d^2} \quad (3.1)$$

where A_e is effective area and G_t is the transmitting antenna gain. Since the receiving antenna gain G_r can be given by

$$G_r = \frac{4\pi A_e}{\lambda^2} \quad (3.2)$$

where λ is the wavelength of the electromagnetic wave. By substituting A_e of equation (3.2) into equation (3.1), we obtain

$$P_r = \frac{G_r G_t P_t}{4\pi d^2}$$

When $G_t = G_r = 1$, free space loss L_f is given by

$$L_f = \frac{P_t}{P_r} = \frac{1}{G_r G_t} \left(\frac{4\pi d}{\lambda} \right)^2 = \left(\frac{4\pi d}{\lambda} \right)^2 = \left(\frac{4\pi f_c d}{c} \right)^2$$

where c is the speed of light ($= 2.998 \times 10^8$ m/s) and f_c is carrier frequency.

3.1.3 A Triangulation Approach

After the previous discussion on the features of RFID technology, it is easy to get the idea of using a triangulation approach to realize location sensing with RFID. Basically, in this idea, we want to make use of the signal strengths of the tags. However, the product currently we can use does not provide this parameter. To each tag, the only direct information we can get is in which range the tag is detected, that is, the tags are reported by the reader in power level 1 to 8. We need to find a way to get the tags' signal

strengths by analyzing their reported power levels. Intuitively, it is a good way to use the triangulation approach to compute the position of each tag like shown in Figure 3.3.

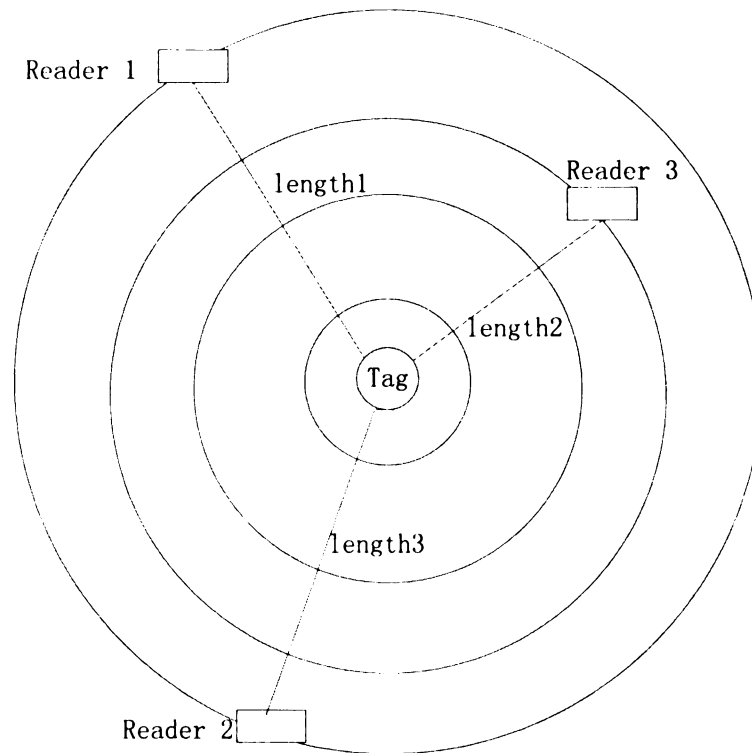


Figure 3.3: A Triangulation Approach

If we can get the accurate value of Length 1, Length 2 and Length 3, since the position of readers are known, it is not difficult for us to compute the position of the tag. Here, we can make use of the equation discussed in section 2.2. In fact, some researchers did use this kind of approach.

However, this approach suffers two problems. First, we cannot collect the signal strength of the tag directly from the reader. Readers only report the power level they detected. In order to make use of the equation, we need to do a preliminary test to know which power level denotes what kind of distance. In fact, the equation works only in free

space. Like shown in Figure 3.4, the power levels distribution of a tag maybe not circles in a complicated indoor environment, that is to say, physical distance cannot be computed accurately by using power levels directly.

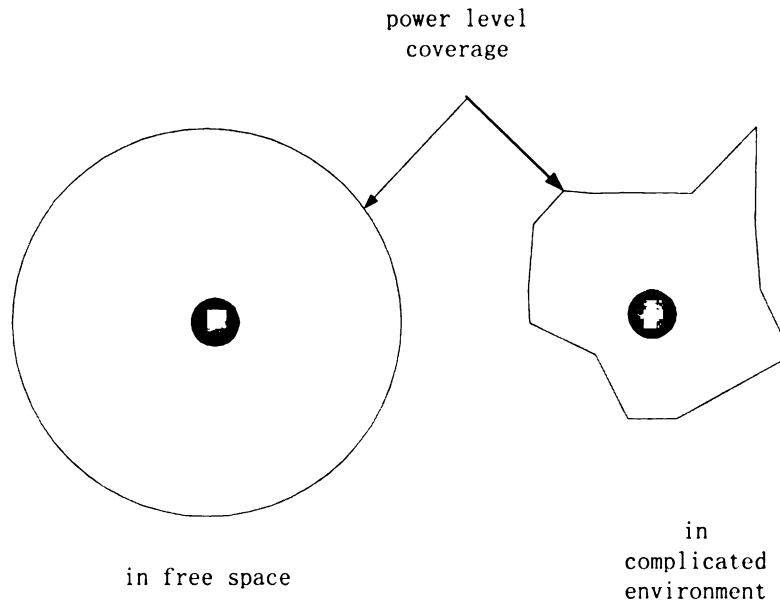


Figure 3.4: Power Level in Different Environment

Secondly, there are always many unpredictable moving things in an indoor environment. Due to these dynamic interferences, even a static object could be reported in different power ranges from time to time, i.e., the power level could be a function of time. In order to offset these dynamic interferences, the only way solving this problem in this approach is to place more readers around the tracking tags to get more data. But the overall cost will be increased greatly with more readers.

3.1.4 Our first attempt

After looking into the specifications of different available systems, we have chosen the Spider System manufactured by RF Code [8] to implement the prototype framework.

Their active tags have a read range of 150 feet. If necessary, this range can be increased to 1000 feet with the addition of a special antenna. Figure 3.5 shows the RFID readers and tags used in our system and their relative size compared with a US quarter.



Figure 3.5: The RFID reader and tag used in our prototype system.

In our system, the RFID Reader's operating frequency is 308 MHz. It also has an 802.11b interface to communicate with other machines. The detection range is 150 feet. The reader provides digital control of read range via providing configuration software and API with 8 incremental read ranges. Each reader can detect up to 500 tags in 7.5 seconds. Each RFID tag is pre-programmed with a unique 7-character ID for identification by readers. Its battery life is 3-5 years. Tags send their unique ID signal in random with an average of 7.5 seconds. Note that the RFID reader has 8 different power levels. Based on the signal strength received by the RFID reader, the reader will report or ignore the received ID, where power level 1 has the shortest range and level 8 has the longest range.

Our first attempt is to install a number of readers as shown in Fig. 3.6 Each reader has a pre-determined power level, thus defining a certain range in which it can detect RFID tags. By properly placing the readers in known locations, the whole region can be divided into a number of sub-regions, where each sub-region can be uniquely identified

by the subset of readers that cover that sub-region. Given an RFID tag, based on the subset of readers that can detect it, we should be able to associate that tag with a known sub-region. The accuracy of this approach is then determined by the number of readers required, the placement of these readers, and the power level of each reader. Such a nicely formulated optimization problem turns out to be useless because the range in which a reader can detect a tagged object is not just due to the power level (similar to signal strength). There are many factors that will affect the range including both static obstructions and dynamic human movement. Due to these dynamic interferences, even a static object could be reported in different sub-regions from time to time. This is the same reason why approaches based on the signal strength of 802.11b are not very useful.

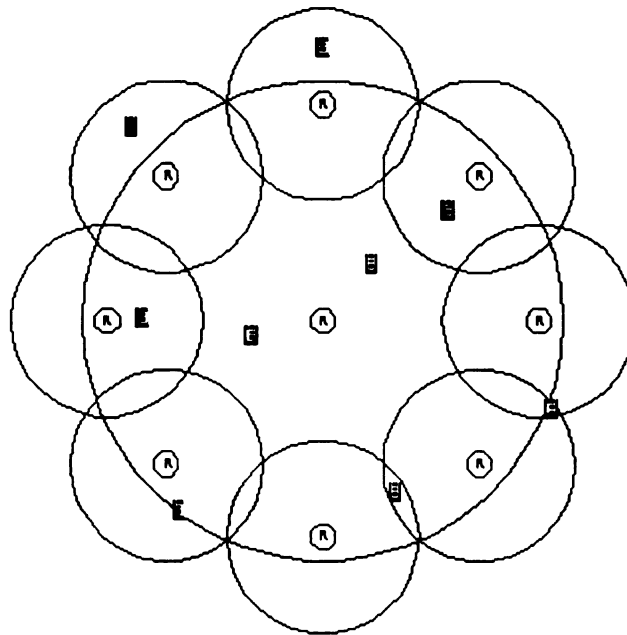


Figure 3.6: Placement of 9 readers with two different ranges and the sub-regions.

3.2 LANDMARC Approach

In order to increase accuracy without placing more readers, the LANDARC (Location Identification based on Dynamic Active RFID Calibration) system employs the idea of having extra fixed location reference tags to help location calibration. These reference tags serve as reference points in the system (like landmarks in our daily life). The proposed approach has three major advantages. First, there is no need for a large number of expensive RFID readers. Instead we use extra, cheaper RFID tags. Second, the environmental dynamics can easily be accommodated. Our approach helps offset many environmental factors that contribute to the variations in detected range because the reference tags are subject to the same effect in the environment as the tags to be located. Thus, we can dynamically update the reference information for lookup based on the detected range from the reference tags in real-time. Third, the location information is more accurate and reliable. The LANDMARC approach is more flexible and dynamic and can achieve much more accurate and close to real-time location sensing. Obviously, the placement of readers and reference tags is very important to the overall accuracy of the system.

The LANDMARC approach does require signal strength information from each tag to readers, if it is within the detectable range. However, the current RFID system does not provide the signal strength of tags directly to readers. Readers only report the power level (1 to 8 in our system) of the tag detected. We might do a preliminary measurement to know which power level corresponds to what distance. However, this may work only in free space. As indicated earlier, the power level distribution is dynamic in a complicated indoor environment. Thus, the physical distance cannot be computed accurately by using

power levels directly. We have to develop an algorithm to reflect the relations of signal strengths by power levels.

3.2.1 System Setup

The prototype environment consists of a sensing network that helps the location tracking of mobile users/objects within certain granularity and accuracy, and a wireless network that enables the communication between mobile devices and the Internet. The sensing network primarily includes the RF readers and RF Tags as mentioned earlier. The other major part of the infrastructure is the wireless network that allows wireless communication between mobile devices like PDAs and the Internet. In addition, it also acts as a bridge between the sensing network and the other part of the system. As the reader is equipped with the capability of communicating wirelessly using IEEE 802.11b wireless network, all the tag information gathered from readers is sent over to the supplied API sitting on a specific server (the location server). This feature does not have the problem of having a wire-connection to the readers, thus reducing the possible restrictions of where the readers could be placed. In addition, the wireless network will serve as the fundamental framework of all the communications in the infrastructure.

To be able to track an object's location, the location information received from the RF Readers has to be processed before being useful. The following is a brief explanation of some of the major configuration values in the API software:

- **Device (RF Readers) Setup:** Used for configuring the IP addresses of the RF Readers.
- **Range:** Used for specifying what range for tags is to be scanned.

- **Mode (Exception versus Continuous):** (1) Exception mode: The reader will report the tag when it is inside the detected range while it will not report again until the reader realizes the tag has gone out of range. (2) Continuous mode: The reader will continuously report the tag ID as long as it was in the configured range
- **Time/tag limit per log file:** Used to configure how long and how much tag events recorded before the API will start a new log file. This is in fact somewhat critical to the configuration in the sense of its effect on efficiency.

After the signal is received by the RF readers, the readers then report the information to “TagTracker Concentrator LI” (a software program/API provided by RF Code, Inc.) via a wired or wireless network. Moreover, the software also acts as a central configuration interface for the RF readers. For example, it can be used to adjust the detection range and rate of the readers. After the information from the readers is processed by the TagTracker Concentrator LI, the processed location information can be buffered locally as a file on the same machine or transmitted via a network socket (configurable in the API).

3.2.2 Methodology

Suppose we have n RF readers along with m tags as reference tags and u tracking tags as objects being tracked. The readers are all configured with continuous mode (continuously reporting the tags that are within the specified range) and a detection-range of 1-8 (meaning the reader will scan from range 1 to 8 and keep repeating the cycle with a rate of 30 seconds per range). We define the Signal Strength Vector of a tracking/moving tag as $\vec{S} = (S_1, S_2, \dots, S_n)$ where S_i denotes the signal strength of the

tracking tag perceived on reader i , where $i \in (1, n)$. For the reference tags, we denote the corresponding Signal Strength vector as $\bar{\theta} = (\theta_1, \theta_2, \dots, \theta_n)$ where θ_i denotes the signal strength. We introduce the Euclidian distance in signal strengths. For each individual

tracking tag p , where $p \in (1, u)$, we define: $E_j = \sqrt{\sum_{i=1}^n (\theta_i - S_i)^2}$ where $j \in (1, m)$, as the

Euclidian distance in signal strength between a tracking tag and a reference tag r_j . Let E denotes the location relationship between the reference tags and the tracking tag, i.e., the nearer reference tag to the tracking tag is supposed to have a smaller E value. When there are m reference tags, a tracking tag has its E vector as $\bar{E} = (E_1, E_2, \dots, E_m)$

This algorithm is to find the unknown tracking tags' nearest neighbors by comparing different E values. Since these E values are only used to reflect the relations of the tags, we use the reported value of the power level to take the place of the value of signal strength in the equation.

There are three key issues we examine through the process of locating the unknown tracking tags. The first issue is the placement of the reference tags. Since the unknown tag is ultimately located in a cell surrounded by some reference tags, the layout of reference tags may significantly affect the location accuracy of an algorithm. The second issue is to determine the number of reference tags in a reference cell that are used in obtaining the most accurate approximate coordinate for each unknown tracking tag. For example, the simplest way to find the nearest reference tag to the tracking tag is to use the coordinate of the reference tag with the smallest E value as the unknown tag's coordinate.

We call this as 1-nearest neighbor algorithm. Or, we can choose a tracking tag's two nearest neighbors and call it 2-nearest neighbor algorithm. When we use k nearest reference tags' coordinates to locate one unknown tag, we call it k -nearest neighbor algorithm. The unknown tracking tag' coordinate (x, y) is obtained by:

$$(x, y) = \sum_{i=1}^k w_i (x_i, y_i)$$

where w_i is the weighting factor to the i -th neighboring reference tag. The choice of these weighting factors is another design parameter. Giving all k nearest neighbors with the same weight (i.e., $w_i = 1/k$) would make a lot of errors. Thus, the third issue is to determine the weights assigned to different neighbors. Intuitively, w_i should depend on the E value of each reference tag in the cell, i.e., w_i is a function of the E values of k -nearest neighbors. Empirically, in LANDMARC, weight is given by:

$$w_j = \frac{\frac{1}{E_i^2}}{\sum_{i=1}^k \frac{1}{E_i^2}}$$

This means the reference tag with the smallest E value has the largest weight. Note that our approach can be easily extended to a 3-dimensional coordinate.

3.3 Experimental Results and Performance Evaluation

We conduct a series of experiments to evaluate performance of the positioning of the LANDMARC System. In the standard setup, we place 4 RF readers ($n=4$) in our lab and 16 tags ($m=16$) as reference tags while the other 8 tags ($u=8$) as objects being tracked, as illustrated in Figure 3.7.

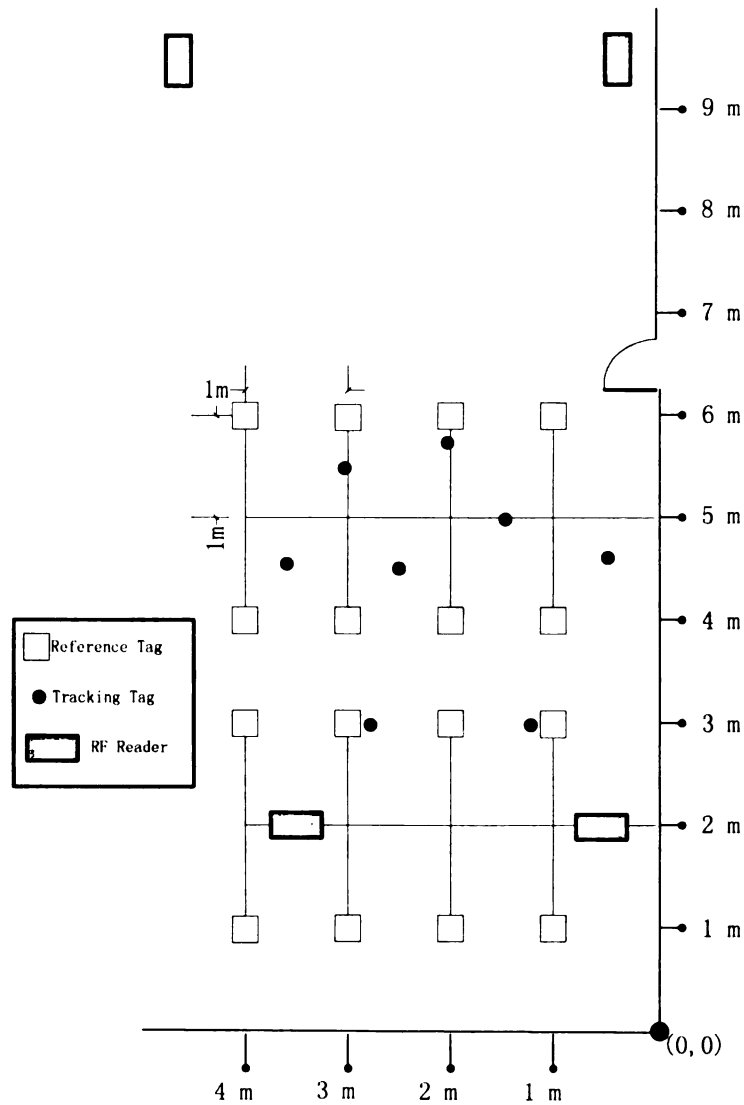


Figure 3.7: Placements of RF Readers and Tags

With the setup, the data are collected via the socket from the TagTracker Concentrator LI in groups of a one-hour period and the system will compute the coordinates of the tracking tags based on each group of data. To quantify how well the LANDMARC system performs, the error distance is used as the basis for the accuracy of the system. We define the location estimation error, e , to be the linear distance between the tracking tag's real coordinates and the computed coordinates, given by

$$e = \sqrt{(x - x_0)^2 + (y - y_0)^2}$$

With the placement of the reference tags and the tracking tags shown in Figure 3.7 for over 48 hours, we keep collecting data of the power levels from 4 RF readers continuously. Thus we obtain 48 groups of one-hour data. For each of 8 tracking tags per hour, the system computes the coordinates of this tag by using the algorithm discussed in Section 3.2.2. We then compute the location error e for each tracking tag. Thus, we have 48 groups of 8 e values. We may examine the location accuracy by analyzing the distribution of these e values under different conditions. Note that we have repeated the experiments many times to avoid statistical errors.

3.3.1 Effect of the Number of Nearest Neighbors

One of the key issues is to find a best k value in the algorithm. We choose different k values as $k=1, 2, 3, 4,$ and 5 and compute the coordinates of the tracking tags, respectively. Figure 3.8 shows the results of using different k values in the formula.

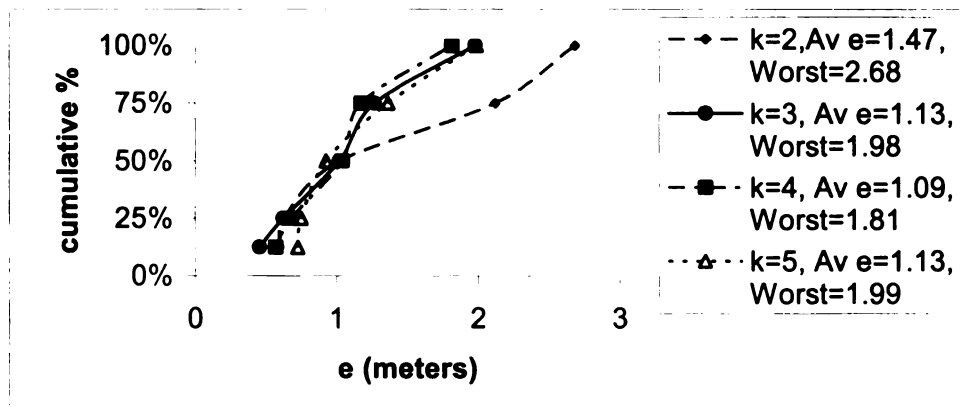


Figure 3.8: Cumulative percentile of error distance for k from 2 to 5.

As shown in Fig. 3.8, $k=4$ works the best and the positioning accuracy does not improve as the k value further increases. Keeping the same placement, we repeat the process for another 48 hours. Though the positioning error distribution changes, $k=4$ still gives the best location information. In fact, in all the later experiments except on a few occasions that $k=3$ and $k=5$ worked better, in most cases $k=4$ is the best choice. Hence, we set 4 as the value of k in our formula in the following experiments.

Based on the statistics, it can be seen that the 50 percentile has an error distance of around 1 meter while the maximum error distances are less than 2 meters. This is very promising because the 50 percentile of the RADAR project is around 2.37m-2.65m and its 90 percentile is around 5.93m-5.97m [9].

3.3.2 Influence of the Environmental Factors

In order to see how well the LANDMARC approach works in different environments, we collect 10 groups of data from midnight to early morning (during which time there is little movement) and another 10 groups of data from 10 AM to 3:00 PM (at which time varying level of activities that would result in variations in transmission of the tags). Figure 3.9 shows the comparison.

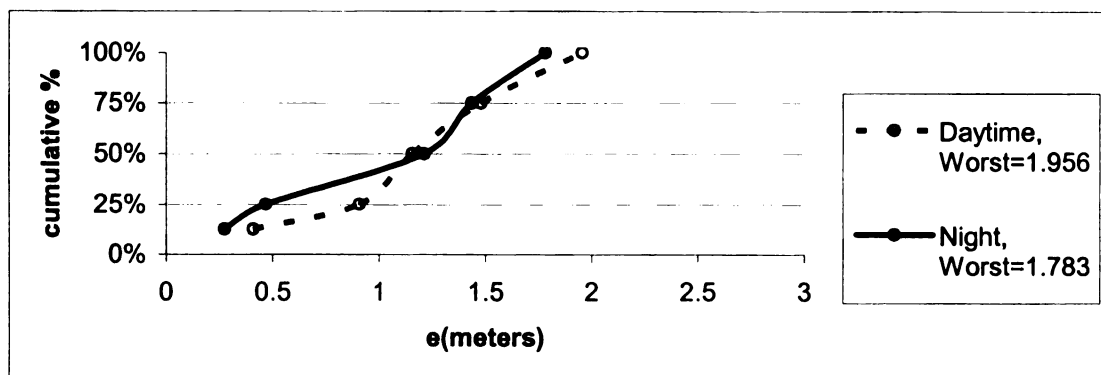


Figure 3.9: Cumulative percentile of error distance in the daytime and at night.

We know that during the daytime, the lab is very busy with many people so there is more interference than at night. From the results, we do not see much difference in the overall accuracy. This shows that our reference tag approach can successfully offset the dynamics of interference.

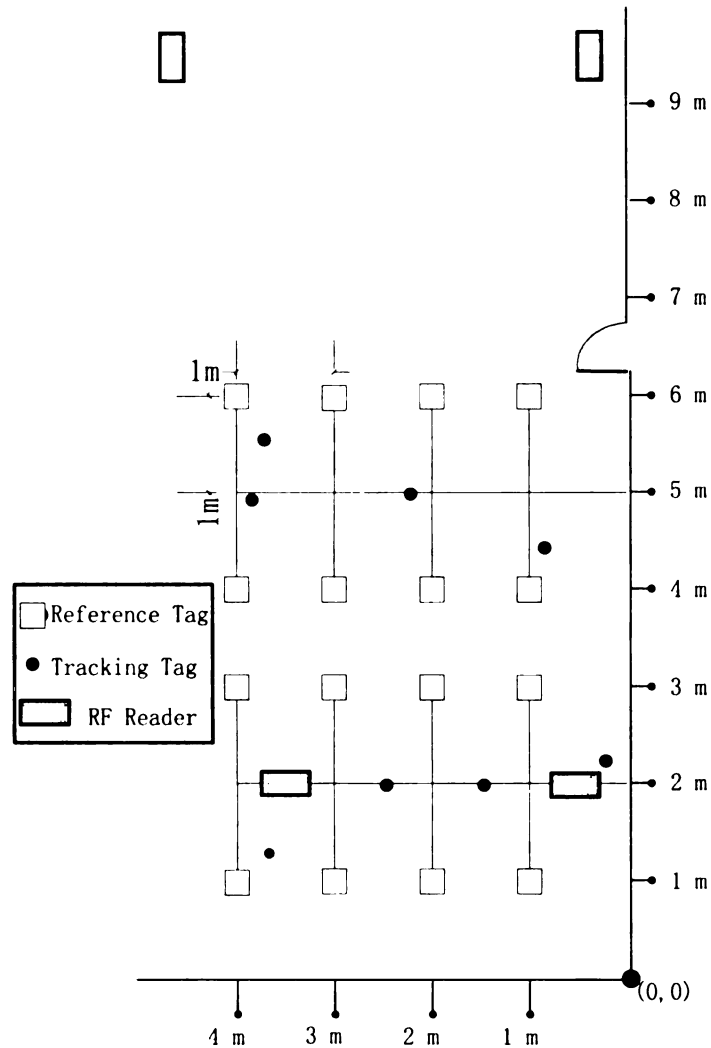


Figure 3.10: Placement of RF readers and tags (placement configuration 2)

As the positions of tracking tags in the real world would be unpredictable, we change the placements of tracking tags randomly and expect the distribution of e could be

changed but the accuracy of the system should be at the same level. We change the placement of the tracking tags as shown in Fig. 3.10 with the reference tags' placement unchanged and repeat the process.

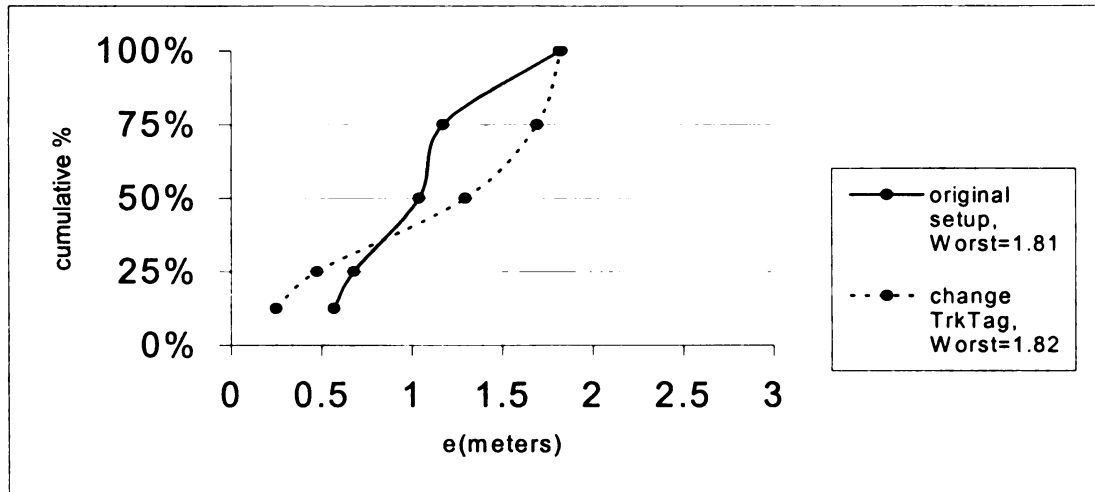


Figure 3.11: Cumulative percentile of error distance between two tracking tag placement configurations in Fig 3.7 and Fig 3.10.

Figure 3.11 is the comparison of the results between the two tracking tag placements shown in Fig. 3.7 and Fig. 3.10. As we expected, the distribution is changed but the overall accuracy is at the same level. Figures 3.9 and 3.11 show that the approach of using reference tags effectively helps offset some of the environmental factors that contribute to the variations in a detected range. Since the reference tags are subject to the same effect in the environment as the tags to be located, we can dynamically update the reference information for lookup, based on the detected range from the reference tags in real-time.

3.3.3 Effect of the Number of Readers

One of the problems of using RF to locate objects is the inconsistency of the signal strength reception. This can primarily be due to the environment and the device itself. In

most cases, the environmental factors always have the most impact on the accuracy and maximum detectable range. These include issues like furniture placement, people's movement, and so on. Besides, non-line of sight (NLOS) is another source of reducing the location sensing accuracy. Even NLOS does not prohibit RF transmission as that of infrared, it does create the multi-path problem, meaning the signal can possibly take different paths to reach the receiver and result in interference among the received signals.

To better deal with the problem, we can use more RF readers to improve the accuracy. With more RF readers, a better decision can be made for location sensing because more data can be gathered by having extra readers to do the sensing as shown in Fig. 8. However, the RF readers are usually quite expensive so placing more readers means extra costs for the whole system. Due to budget constraints, we have only four RF readers. Adding more readers may not necessary significantly increase the accuracy. It does increase the processing overhead.

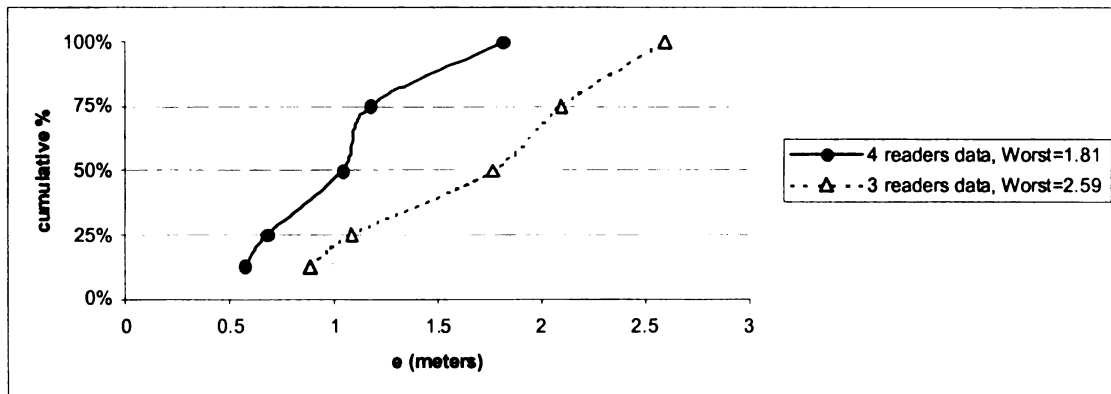


Figure 3.12: Cumulative percentile of error distance for 3 and 4 RF readers.

3.3.4 Effect of Placement of Reference Tags

Intuitively, placement of reference tags should have an effect on the measurement accuracy. Consider the two configurations shown in Fig. 3.13.

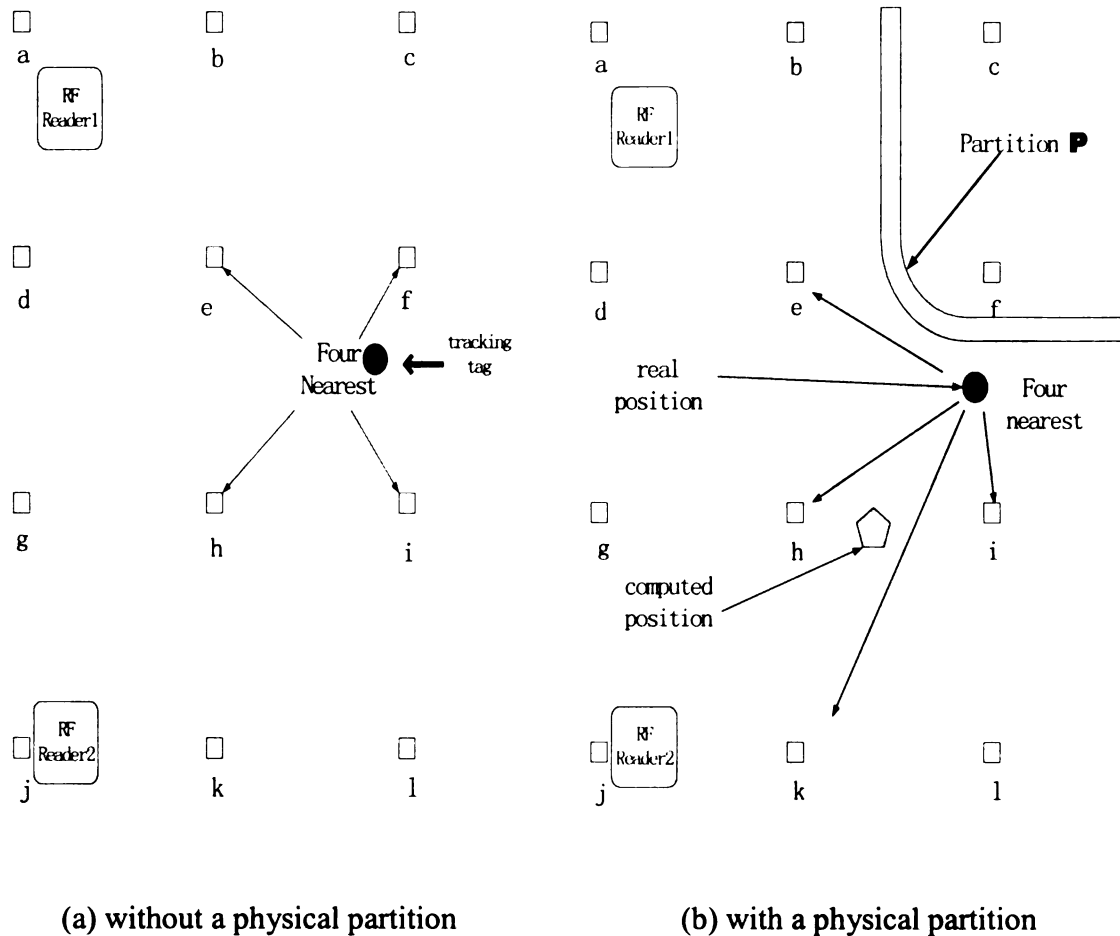


Figure 3.13: A physical partition to separate reference tags c and f from others

In the case of Fig 3.13a, where there is not any obstacle, it is probable that the system can easily find the tracking tag's four nearest neighbors which are tags e, f, h, i by comparing the reported signal strengths and E_f could be the smallest in this tracking tag's E vector. Thus, the tracking tag could be located among the four reference tags. However, sometimes the environment could be more complicated. Suppose there is a partition (or sometimes even a person standing like the partition) as shown in Fig. 3.13 b.

Under these circumstances, it is possible that the reception of the signal strength from the reference tag f is influenced by the partition (or the unexpected people). Consequently, the readers will report a weaker signal strength from tag f so that tag f could fail to be included in the four neighbors of the tracking tag. Instead, tag k may become one of the four reported nearest neighbors to the tracking tag as shown in Fig. 3.13 b. Using e, h, i, k as the four reported neighbors, the position of the tracking tag is likely to be computed as indicated in Fig. 3.13b. Thus, more error occurs.

Things will change if we place more reference tags as illustrated in Figure 3.14. Around the tracking tag, now we have placed more reference tags (the green ones in the Figure). Together with the tag i , tags m, n, o could be included in the four reported nearest neighbors of the tracking tag. Thus, better location information will be provided.

In our next experiment, we place all of the reference tags with a higher density as shown in Figure 3.15. In the first 48 hours we keep the original positions of all the tracking tags unchanged (case Near 1 in Fig. 3.15). In the next 48 hours we move the positions of the tracking tags as indicated in the case of Near 2 in Fig. 3.15.

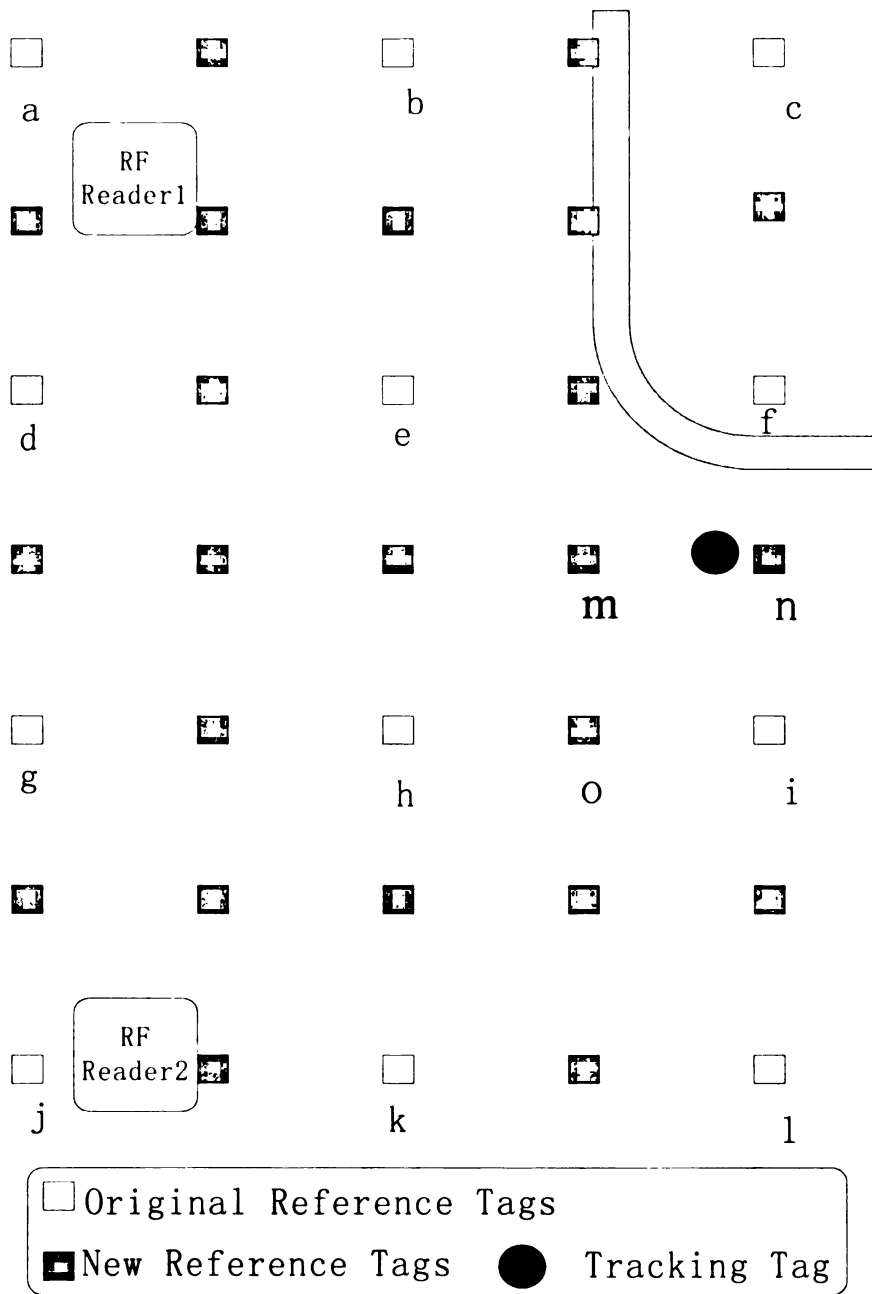


Figure 3.14: More reference tags are used.

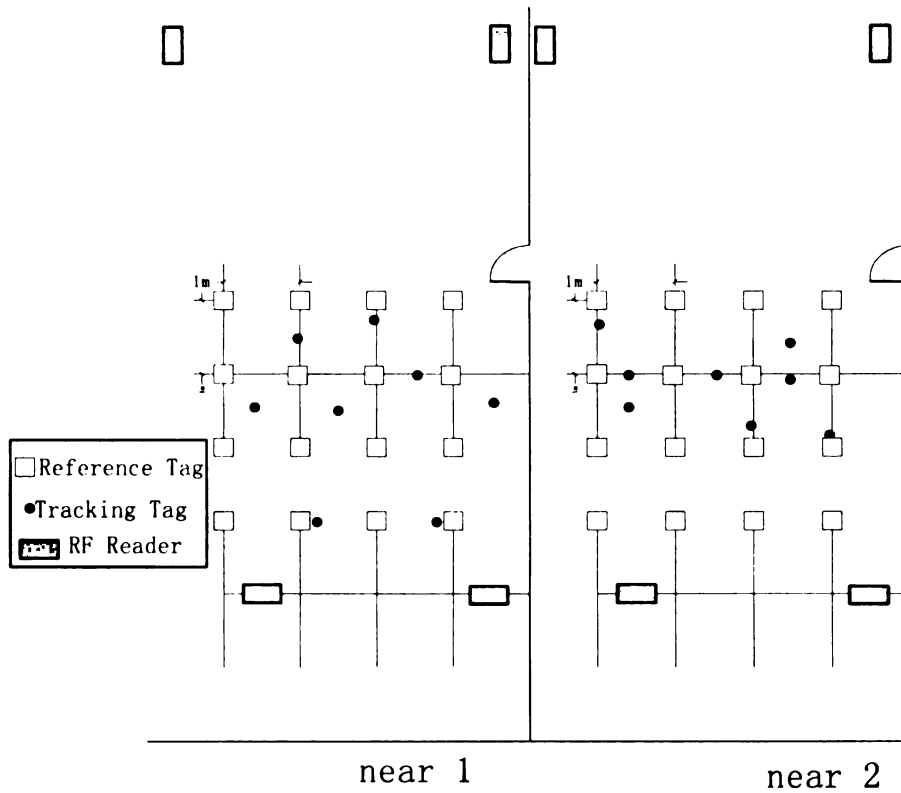


Figure 3.15: Two higher density, comparing with those in Fig. 4, placements of reference tags

It can be seen in Fig. 3.16 that the accuracy of the LANDMARC System is improved with a higher reference tag density, as we have discussed. However, the improvement is not as great as we expected. We will discuss this point later.

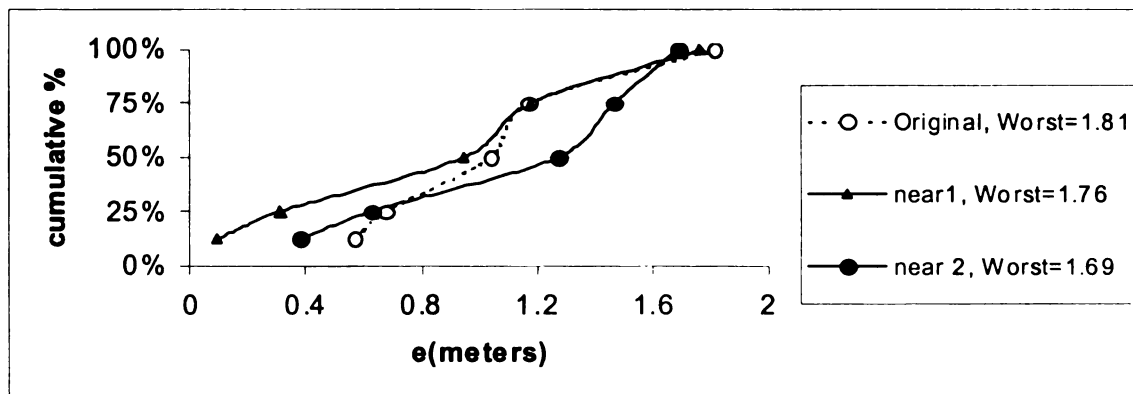


Figure 3.16: Cumulative percentile of error distance with a higher reference tag density

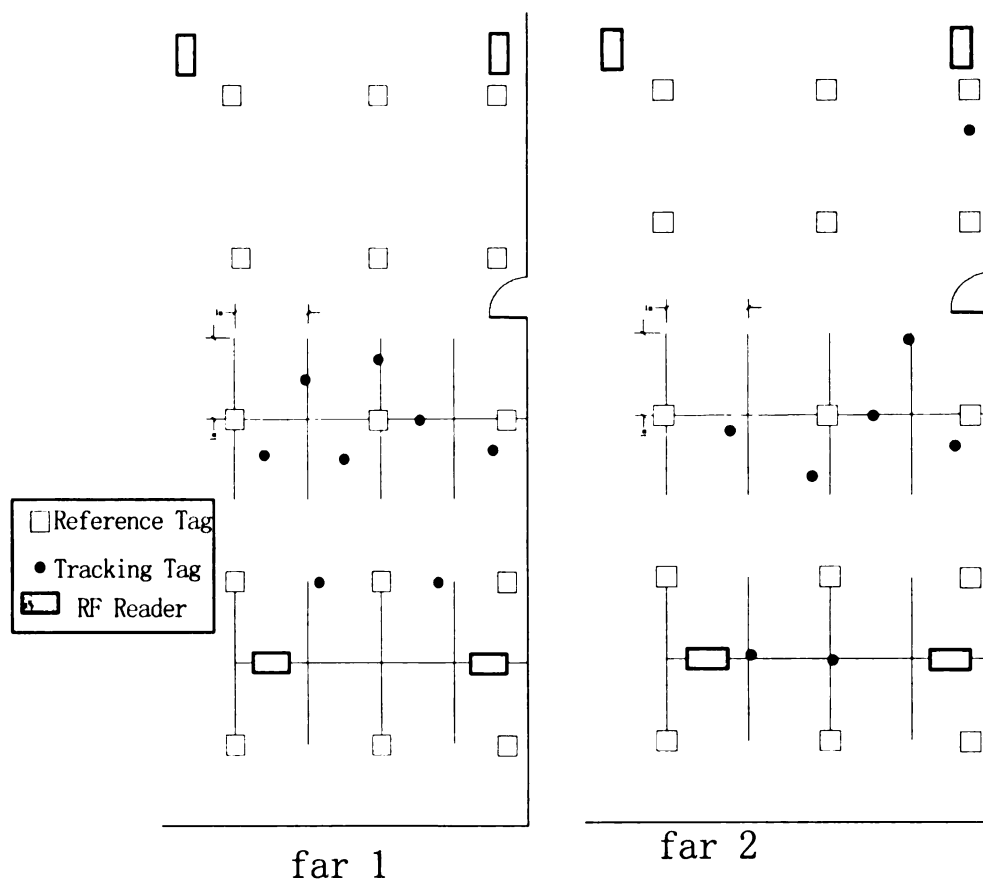


Figure 3.17: Two lower density, comparing with those in Fig. 4, placements of reference tags

Figure 3.17 shows two configurations of a lower reference tag density. The corresponding distribution of error distance is shown in Fig. 3.18. As expected, the accuracy has dropped quite significantly.

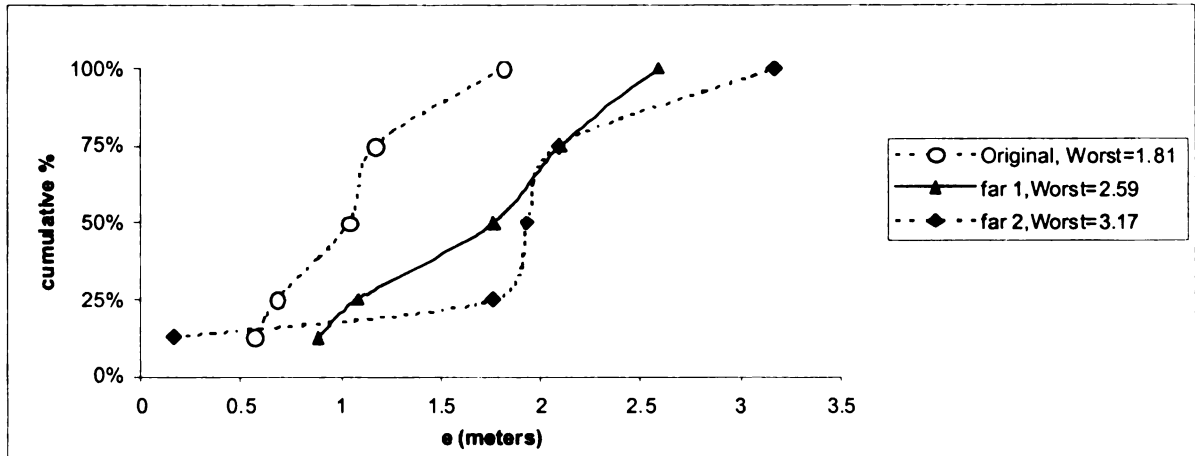


Figure 3.18: Cumulative percentile of error distance with a lower reference tag density

It is obvious that the accuracy of the LANDMARC approach decreases greatly with a lower density of reference tags. Thus, there is a tradeoff between the accuracy and the number of reference tags. Conceivably, we can improve the accuracy of the LANDMARC System by placing as many reference tags as we can, for example, even in every cubic square centimeter of the space where we want to locate objects. This does not make much sense due to the increased complexity, overheads, inherent device error, and measurement error. We believe that the accuracy can be greatly improved if RFID vendors can make some design changes to be discussed in chapter 6.

In this chapter we have presented a prototype indoor location sensing system using active RFID. Although RFID is not designed for indoor location sensing, the proposed LANDMARC approach does show that active RFID is a viable cost-effective candidate for indoor location sensing, and LANDMARC is a good candidate approach to support the location-based prioritized overlay multicast system.

We decided to use the LANDMARC location sensing system to supply the location based prioritized overlay multicast with all member's current space location and position, which are required for building a efficient tree. LANDMARC is a scalable, robust and easy to deploy system. The experimental results have indicated that LANDMARC System works well. Using 4 RF readers in our lab, we roughly need one reference tag per square meter to accurately locate the objects within error distance such that the largest error is 2 meters and the average is about 1 meter, which meets the accuracy requirements for the location based overlay multicast system in ad-hoc environments.

4 Location-based Prioritized Overlay Multicast

It is quite evident that IP-based multicast would always be more efficient compared to overlay multicast for a given topology. However, overlay networks have the advantage of being easy to implement. Since the topology is built at the application layer, it provides greater flexibility in the design. Building role based priority trees is one such advantage of overlay networks. Different priority trees can be built in the same environment to provide different kinds of services. The priority of the trees would depend on the importance of the service they provide. With multiple priority trees, nodes belonging to different trees can switch between networks depending on what functionality they want to provide. At any given time, a node would associate itself only with one priority tree (the highest priority tree in its set) and it would have to ignore incoming messages/data from members in low priority trees. Priority tokens can be supplied by a node that initials the formation of a new priority tree. The value of the priority token can determine the priority of the tree. Thus if a node is currently a member of priority tree 'i', it would not listen to message/data from member nodes belonging to 'i-1' or lower priority tree. Once the 'i' token tree is dissolved, the member nodes will return to their original tree that held a lower priority than 'i'.

When a node decided to form a high priority tree or receives an invitation to join a high priority network, it may leave behind several 'orphan' nodes. These orphan nodes would now have to connect to another node in the original network so that they are able to receive data from the source node. One way to tackle this problem would be for the departing nodes to send a control message to their children informing them that it is leaving the network. Along with this message, the node will give them its parent address. The child node can thus contact its grandparent (parent's parent) node and start getting data from it. Figure 4.1 (a) presents one such scenario. It should be noted that the Figure 4.1 refers to the logical topology. The physical topology of the above tree can be quite different. The member nodes use multi-hop means of communicating with each other. Non-member nodes in the network may act as intermediate hops. Thus referring to Figure 4.1 (a) again, nodes F and I may not be close to each other and yet would be neighbors to each other in the logical topology. Later sections of the paper describe the idea of location aware overlay trees by possible using LANDMARC. With such trees, the logical topology would try to match the physical topology and there is a good possibility that F and I are indeed close to each other in the physical topology too.

Figure 4.1 (b) shows the formation of the new (high priority) tree. At the beginning, the external node M contacts nodes D, E & G of the original tree and another external node L to form a high priority network. In the first step, M asserts its priority by sending a unicast token message to each of the desired nodes. In step two, M exchanges information about the formation of the tree topology. The topology information can be

based on the location information of nodes with respect to each other. Step 3 is the final formation of the tree. The number of step can vary depending on the implementation or the algorithm used for tree formation. The location discovery can be implemented by the source node and hence it decides on the topology and informs the other member nodes.

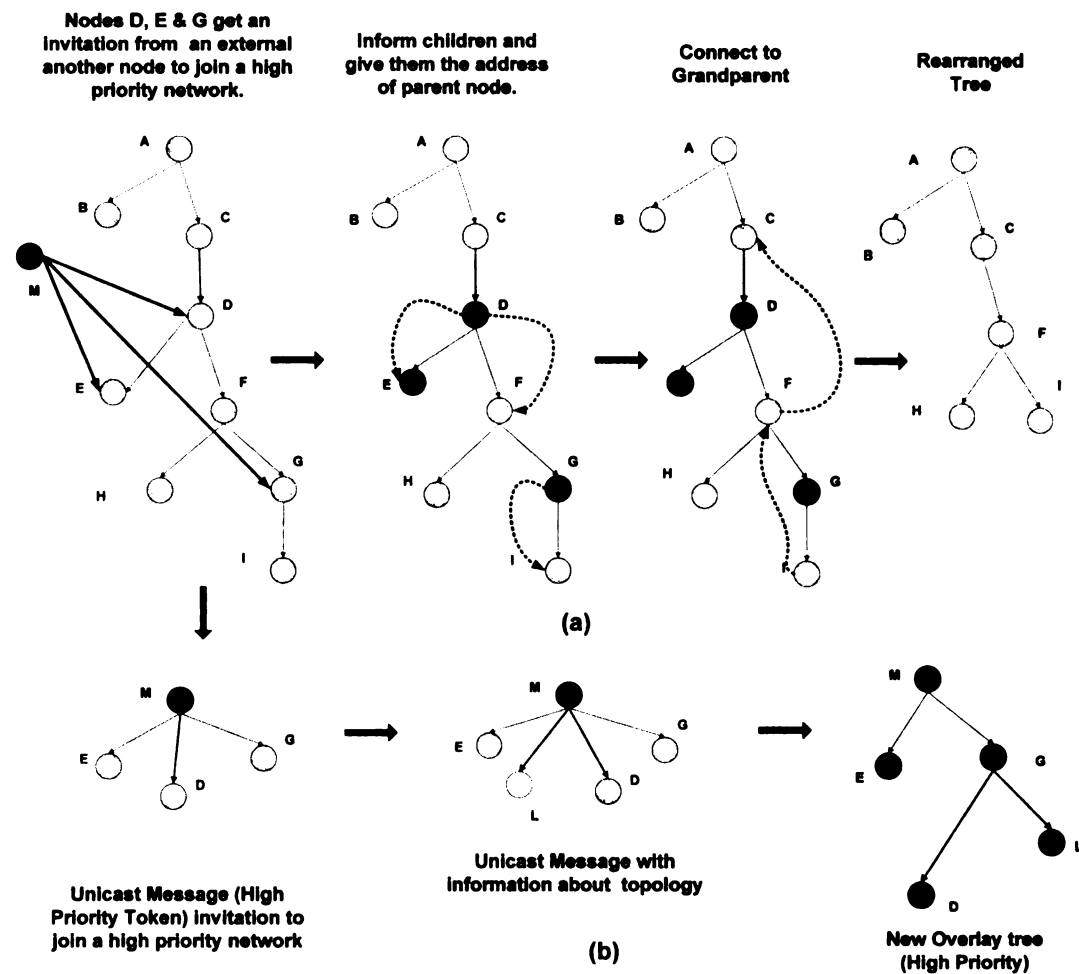


Figure 4.1: Formation of high priority tree and rearrangement of the old tree

Imagine that in a basketball arena there are thousands of spectators. Policing a large crowd has always been a challenge. It is unwise to have all the security personnel at one place waiting for something to go wrong. Instead we could have security personnel scattered across the stadium to monitor any suspicious activity in the crowd. At anytime, a security guard can request assistance from nearby guards if he observes any suspicious activity in the crowd. Some of the information, such as the color, height and facial description of suspects, depends on an individual's perception and/or his point-of-view with respect to the suspect. In most of the cases, this information is conveyed as audio over a walkie-talkie, which has some static noise. Besides, if the crowd is cheering, the listener would have great difficulty in understanding the message. Thus, with the present approach, the information can be insufficient and create ambiguity in identifying suspects when monitoring a large crowd. Recently, security systems have started using more advanced (multimedia) ways to describe suspects. Images of the suspect or video clips (live or recorded) can be used to accurately identify suspects. This multimedia information will be exchanged between security personnel's wireless handheld devices using overlay multicast networks in our proposed location-based prioritized overlay multicast model (L-POM). L-POM builds priority trees with certain nodes carrying important tasks in overlay networks, and rearranges low priority trees whenever some nodes temporarily move to a high priority network.

L-POM also utilizes location information while building the overlay trees. Mobile devices carried by the event organizers and the security personnel would be tuned to communicate over the same channel using the same SSID for the wireless network.

Under normal conditions, security nodes and nodes belonging to event organizers and stadium management would all belong to the same tree. However, during suspicious activities, a security node can initiate the formation of a high priority security consisting of only security nodes. The device carried by the security guard would implement some form of location (neighbor) discovery protocol to identify mobile devices carried by nearby security personnel. In Chapter 2 we have compared several means to obtain location information. The LANDMARC proposed in Chapter 3 can be used to obtain accurate location information. The neighbor discovery protocol can be made proactive where it periodically checks for neighboring devices or user initiated where the user sets off the search. The alert message received by the participating security nodes will cause them to ignore messages from other non-security nodes since they (non-security nodes) would now be considered as belonging to a low priority network. The application layer at the mobile receivers can filter messages depending on who the source is or the message/group priority. For instance, a mobile device carried by the security personnel would be smart enough to ignore messages from an event organizer when it is receiving a video clip from another security guard or when it is assigned the task of frequently report the activities of a suspect. Figure 4.2 shows such a scenario. During such activities, the security nodes would be part of a high priority tree. It should be noted here that since the network is partitioned at the application layer, members of lower priority tree would be considered as non-members by participants in the higher priority tree. Having the network partitioned at the application layer has a subtle advantage. The advantage stems from the fact that having a high density of nodes in a given area helps in multi-hop communication in an ad-hoc wireless environment. With higher density of nodes there is

more overlapping between the coverage areas of nodes, which result in better multi-hop communication between member nodes resulting in lower latency. We will present simulation results showing the effect of density of nodes to the latency in simulation part.

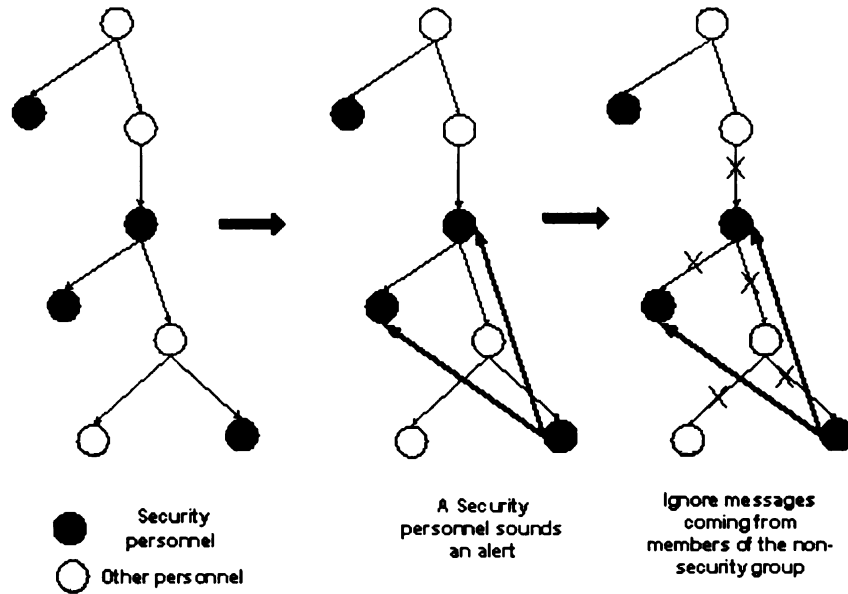


Figure 4.2: Role-based partitioning of overlay network

5 Overlay Multicast Performance Evaluation

The simulations were carried out on *ns* (2.1b7a) simulator using CMU extension (for Ad-Hoc Network) on a P4 Dell Precision 330 running RedHat Linux 7.3. As of this writing, *ns* does not have any extension for simulating overlay multicast, although, it does have an extension to simulate network layer multicast (ODMRP [3] and ADMR [4]). With the help of *bash* scripting, the traffic pattern generated by CMU's *cbrgen* utility was modified to simulate overlay multicast for different movement scenarios. CMU's *setdest* utility was used to generate different movement patterns for mobile nodes. The parameters considered were the number of nodes, pause time, speed, time of simulation, and the movement area. The nodes in the simulation move according to the 'random waypoint' model [9]. When the simulation starts, each node is stationary at a particular location in the specified area for a time equal to the pause time. After the pause time expires, the nodes select a random destination within the given area and start to move with a maximum speed specified during the creation of the scenario file. After reaching the destination, the nodes stay stationary for a time equal to the pause time and select another destination and proceed towards it as described before. Since the performance of protocols is very sensitive to the movement pattern, all simulations were test for 10

different movement patterns for every combination considered. The result values are average of 10. In the rest of this chapter, we will present the simulation results of identifying a suitable unicast (ad-hoc) routing protocol, exploring to use location information to build more efficient trees, and studying the impact of density of wireless nodes, packet size, and fragmentation on performance

5.1 Unicast Protocol Identification

Since overlay network forms a logical network consisting of multicast member nodes, the underlying network looks at the data exchange between member nodes as a unicast communication. This unicast communication can make use of the various ad-hoc routing protocols DSR [5], AODV [6], DSDV [7] TORA [21]. DSDV is a hop-by-hop distance vector routing protocol requiring each node to periodically broadcast routing updates. The key advantage of DSDV over traditional distance vector protocols is that it guarantees loop-freedom. DSR uses source routing rather than hop-by-hop routing, with each packet to be routed carrying in its header the complete, ordered list of nodes through which the packet must pass. The key advantage of source routing is that intermediate nodes do not need to maintain up-to-date routing information in order to route the packets they forward, since the packets themselves already contain all the routing decisions. AODV is essentially a combination of both DSR and DSDV. It borrows the basic on-demand mechanism of Route Discovery and Route Maintenance from DSR, plus the use of hop-by-hop routing, sequence numbers, and periodic beacons from DSDV. TORA is a distributed routing protocol based on a “link reversal” algorithm. It is designed to discover routes on demand, provide multiple routes to a destination, establish routes quickly, and minimize communication overhead by localizing algorithmic reaction to

topological changes when possible. Route optimality (shortest-path routing) is considered of secondary importance, and longer routes are often used to avoid the overhead of discovering newer routes.

In most cases, the logical network remains more or less static even though the underlying physical topology is changing with time. The ad-hoc routing protocols are designed to adapt to the changing network dynamics. The member nodes are required to provide multicast functionality and keep state information only. They do not have to worry about the dynamics of the underlying network or get any assistance from non-member nodes. This conforms very well with the stateless architecture for network protocols.

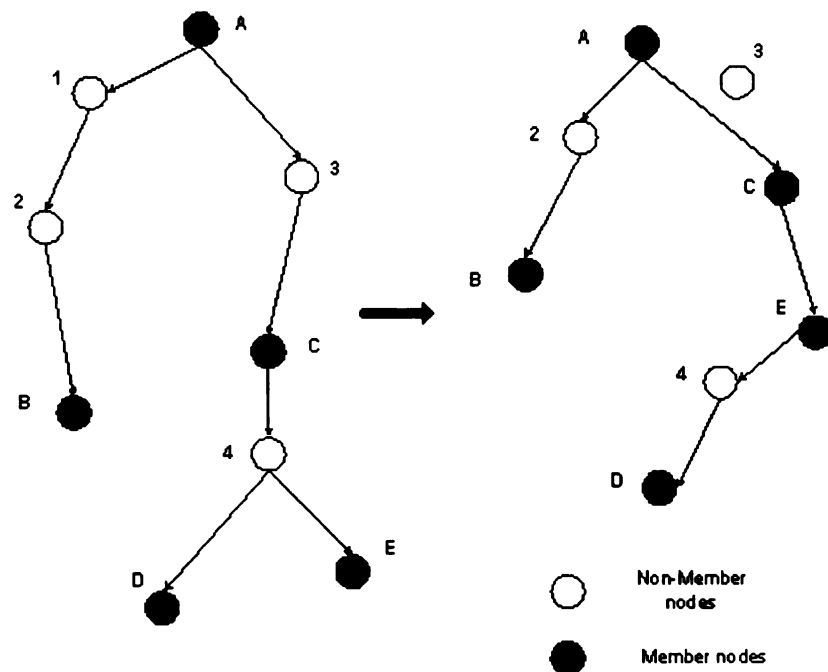


Figure 5.1: A Changed Physical topology

Figure 5.1 depicts a snapshot of a situation where the nodes in the have either moved or left the network. The overlay tree still remains the same although in some cases, it might even change. In response to the change in the physical topology, the unicast routes at almost all the nodes would have changed. However, this change in physical topology is transparent to the overlay members. This paper examines different unicast ad-hoc routing protocols in an attempt to find an efficient protocol with low latency, less drop rate and minimal overhead.

In order to identify a good ad-hoc routing protocol, simulation results for two overlay trees in Figure 5.2 (a) and Figure 5.2 (b) were analyzed. Table 2 shows the different simulation parameters that were considered. In this case, the mobile nodes are carried by humans (security personnel), so the speeds considered in the simulation were 1 m/sec (human walking) and 5 m/sec (human running). The average time to complete the transfer of a 100KB file to all the member nodes along with the average drop ratio (ratio of total number of packets dropped to the total number of packets sent) and average protocol ratio (ratio of total number of protocol message packets to the total number of sent packets) was observed and analyzed for 10 movement scenarios.

Table 2: Simulation Parameters for protocol comparison

Simulation Area (m ²)	500x500
Total number of nodes	25
Total number of member nodes	12 (Figure 5.2(a)), and 15 (Figure 5.2(b))
Speed of nodes (m/sec)	1, 5
Pause Times (Sec)	0, 5, 10, 15, 20
Simulation Time (Sec)	400
File Size (KB)	100
Packet Size (bytes)	512, 1024
Routing protocols ¹	DSR, DSDV, AODV
Number of scenario patters tested	10

¹ TORA [21] was one of the test candidates, however, due to its very poor performance it was eliminated in the initial rounds of simulations.

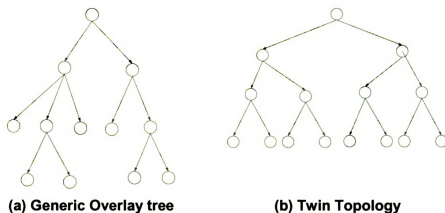


Figure 5.2: Topologies used for testing different protocols

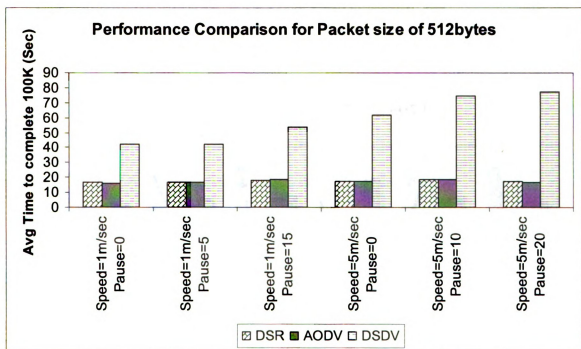


Figure 5.3: Comparison of average completion time for packet size of 512

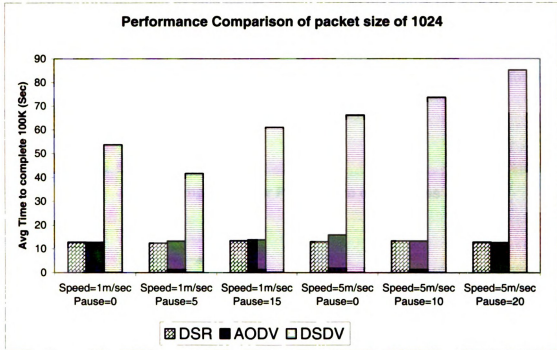


Figure 5.4: Comparison of average completion time for packet size of 1024

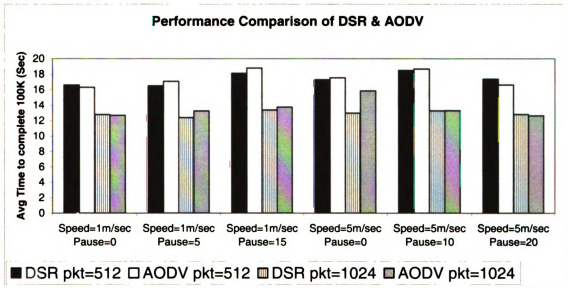


Figure 5.5: Comparison between AODV & DSR (Average Completion time)

In Figure 5.3, Figure 5.4 and Figure 5.5, we compare the performance of the three protocols for different speeds, pause times and packet size. It is clear that DSDV shows a

very poor performance and its high values overshadowed the values for AODV and DSR. Henceforth, the protocol comparison has been made between AODV and DSR only. It is also evident that completion times for packet size of 1024B is less than that for 512B, indicating that larger packet sizes can result in better performance.

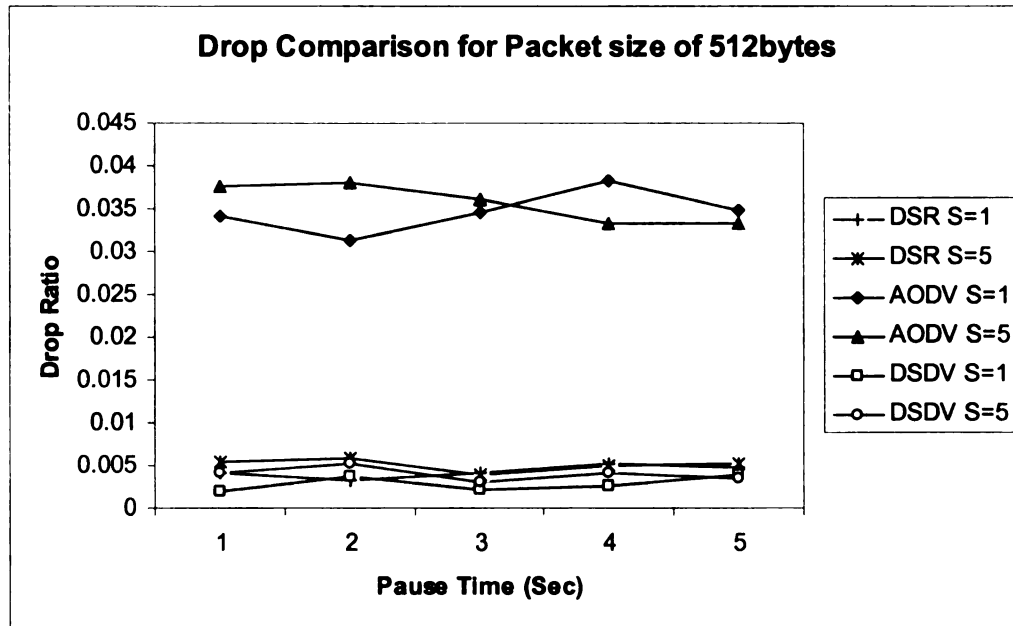


Figure 5.6: Performance in terms of Drop Ratio

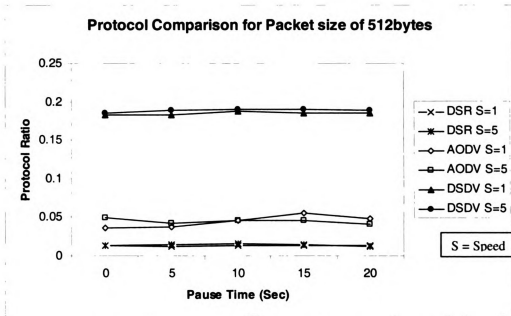


Figure 5.7: Performance in terms of Protocol Ratio (Lower is better)

The drop ratio graph shows that AODV has a very high drop ratio compared with other two protocols while the protocol ratio comparison graph indicated that DSDV has a very high protocol overhead.

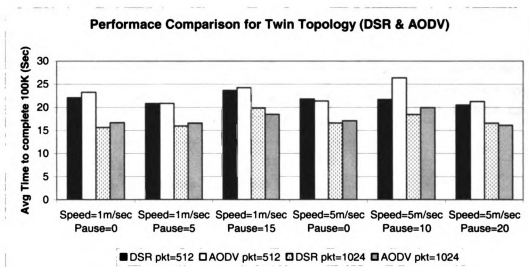


Figure 5.8: Comparison between DSR & AODV (FTP Completion time)

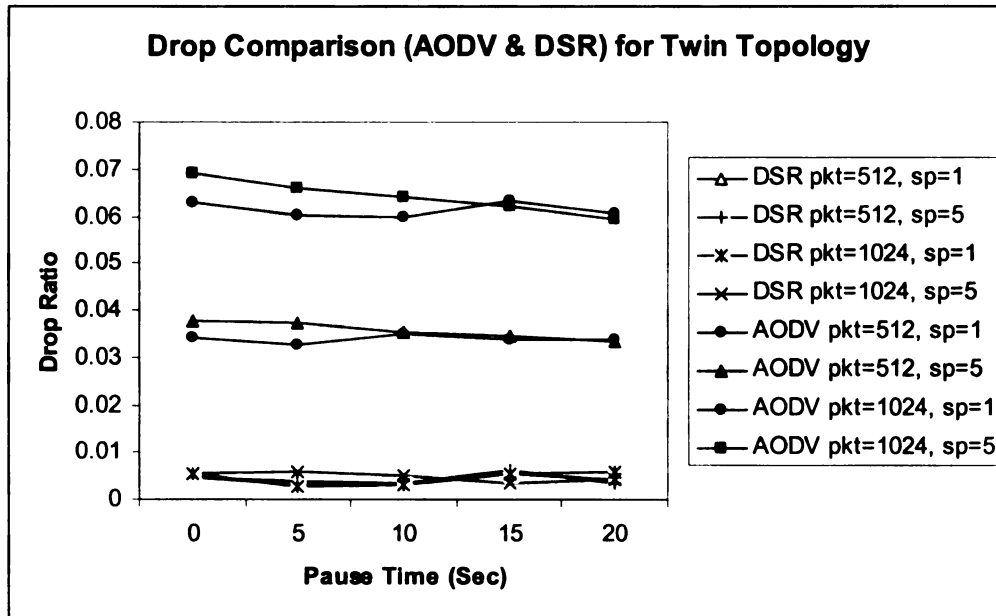


Figure 5.9: Drop Ratio comparison for DSR & AODV (Twin Topology)

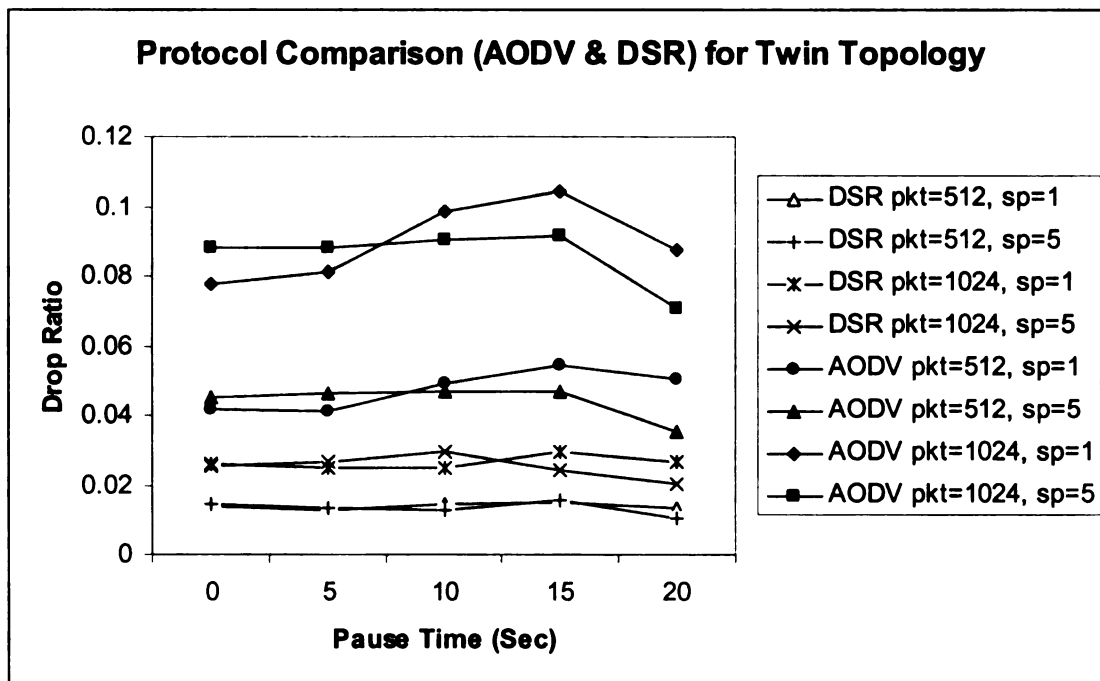


Figure 5.10: Performance in terms of Protocol Ratio for twin topology

The results of simulation for twin topology show the same trends as for the generic tree. DSR and AODV have similar performance in terms of transfer time. However, AODV shows a very high drop ratio and the drop ratio increases with the packet size –

that is, it is high for packet size of 1024B. It is also observed that higher packet size reduces the transfer time. AODV has higher protocol overhead compared to DSR. DSR carries with it the advantage of source routing where the packets carry the information about the route to the destination. As a result, aside from the initial route discovery overhead, DSR does not exhibit a high routing overhead. On the other hand, in case of AODV, each node participating between the source and the destination needs to maintain information about the route. Besides, in order to maintain routes, AODV normally requires periodical transmission of a *hello* message (with a default rate of once per second). With these factors in mind, *packet size of 1024B and DSR (underlying unicast routing protocol) were chosen for further simulations.*

5.2 Location-based Multicast Tree

With improvements in location sensing techniques, it is now possible to use LANDMARC to inexpensively establish the position of a person within 2 meter error in indoor environments [10]. Differential GPS (DGPS) [17] [18] systems have greatly improved the accuracy in location positioning in outdoor environments. Recently several GPS companies have come up with CF cards that can be plugged into handheld devices to enable GPS capability.

Dynamically changing topology is the main characteristics of an ad-hoc networks and an exciting approach would be the use of location sensing technologies to identify near-by member nodes during tree formation. Already an approach to use the geometric distance as heuristic in tree formation in ad-hoc networks has been suggested [8]. Table 3 shows the simulation parameters used to test the idea of building location aware overlay tree. Figure 5.11 shows the example overlay tree that was considered for building an

overlay tree without any location information. In the next step, simulations were repeated by building ‘smart’ overlay trees – trees build with location information (after observing the .nam file generated by ns). One such example is shown in Figure 5.13. Figure 5.12 represents the overlay tree built by making use of information about the location of nodes in that particular scenario shown in Figure 5.13.

Table 3: Simulation parameters for testing location aware tree.

Simulation Area (m ²)	800x800
Total number of nodes	25
Total number of member nodes	10
Speed of nodes (m/sec)	1
Pause Times (Sec)	0 (continuous movement)
Simulation Time (Sec)	400
File Size (KB)	500
Packet Size (bytes)	1024
Routing protocols	DSR
Number of scenario patters tested	7

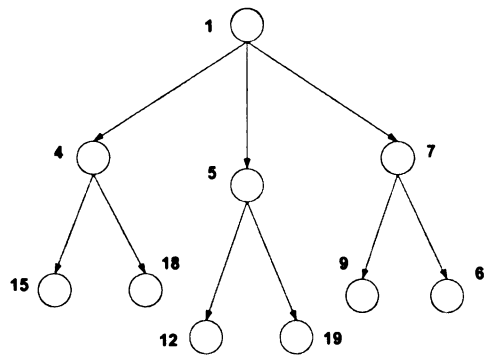


Figure 5.11: Overlay tree without any location information.

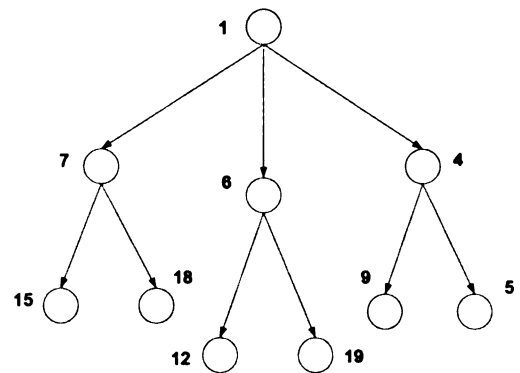


Figure 5.12: Location aware overlay tree.

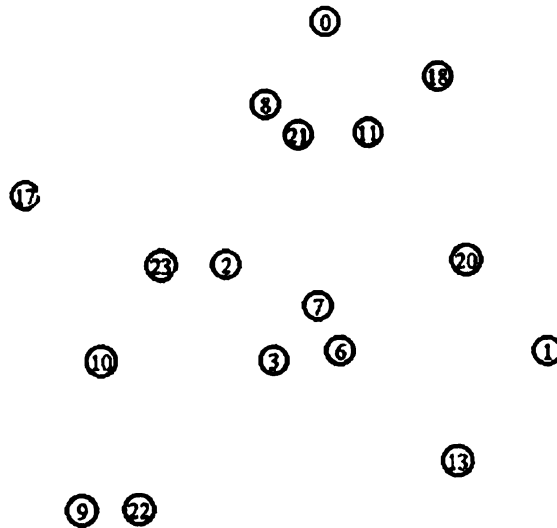


Figure 5.13: Actual Physical positions of nodes in one particular (random) Scenario

As an example, we can see in Figure 5.13 that node 12 is several hops away from node 5 but is closer to node 7. With location-aware tree, node 12 gets data from node 7 instead of 5. Similarly, node 9 is closer to node 4 than node 7. Figure 5.14 compares the performance of the two scenarios. From the graph, it can be seen that location aware overlay trees have lesser latency compared with trees built without any location information. However, it can be noted that this is not always true (such as the rightmost pair of bars in Figure 5.14. There can be scenarios where one or more member nodes are isolated from the group or nodes do not have any overlapping coverage. One such scenario is shown in Figure 5.15.

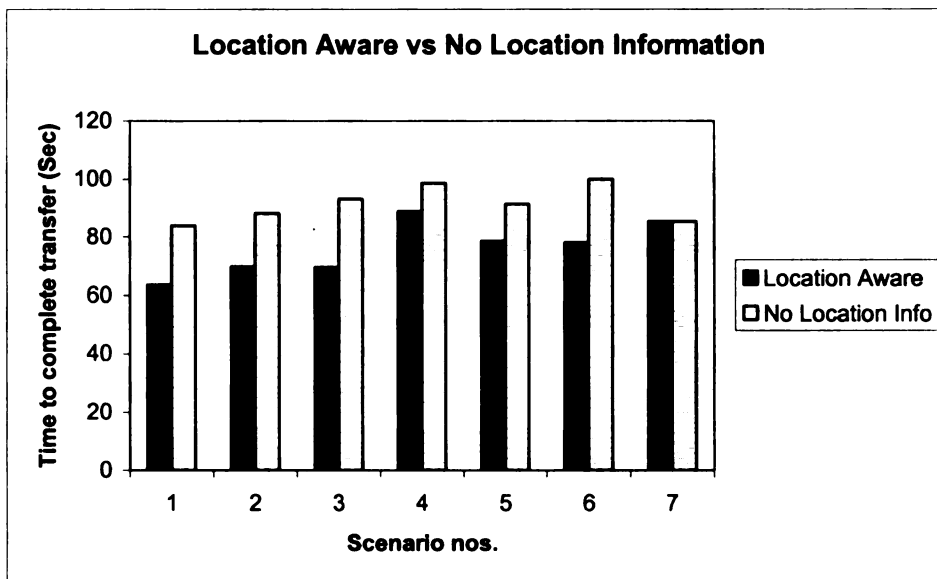


Figure 5.14: Performance of location aware overlay tree and an overlay tree without any location information about member nodes.

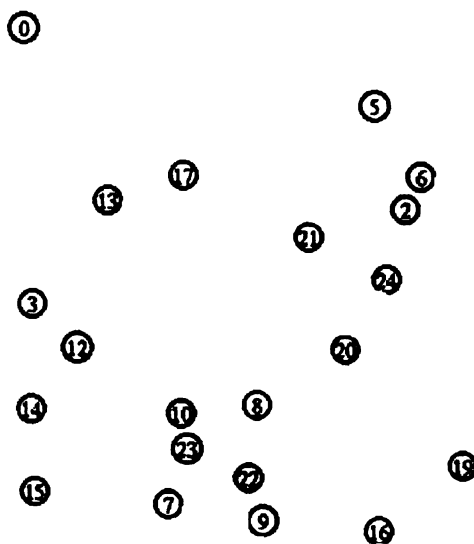


Figure 5.15: Worst case scenarios

Figure 5.15, shows that some of the member nodes (7, 12, 15, 9, 18, and 19) are located far away from the source node (1) and the first tier nodes (4, 5, and 6) that are close to the source.

5.3 Density of Mobile Nodes

Mobile nodes have a limited coverage area. As a result, the density of mobile nodes in a given area greatly influences the performance of the network. With higher density of nodes in a given area, there are more nodes to perform multi-hop forwarding resulting in improvement in the overall performance.

Table 4 shows the simulation parameters used to test this idea. The overlay tree is same as the one shown in Figure 5.11. The area was incrementally changed from $100 \times 100 \text{m}^2$ to $800 \times 800 \text{m}^2$. Also, a total of 15, 25, 40 and 60 nodes were tested with different areas. The results are presented in Figure 5.16 and Figure 5.17.

Table 4: Simulation parameters for testing effect of density.

Total number of member nodes	10
Speed of nodes (m/sec)	5
Pause Times (Sec)	0, 5
Simulation Time (Sec)	400
File Size (KB)	250
Packet Size (bytes)	1024
Routing protocols	DSR
Number of scenario patters tested	10

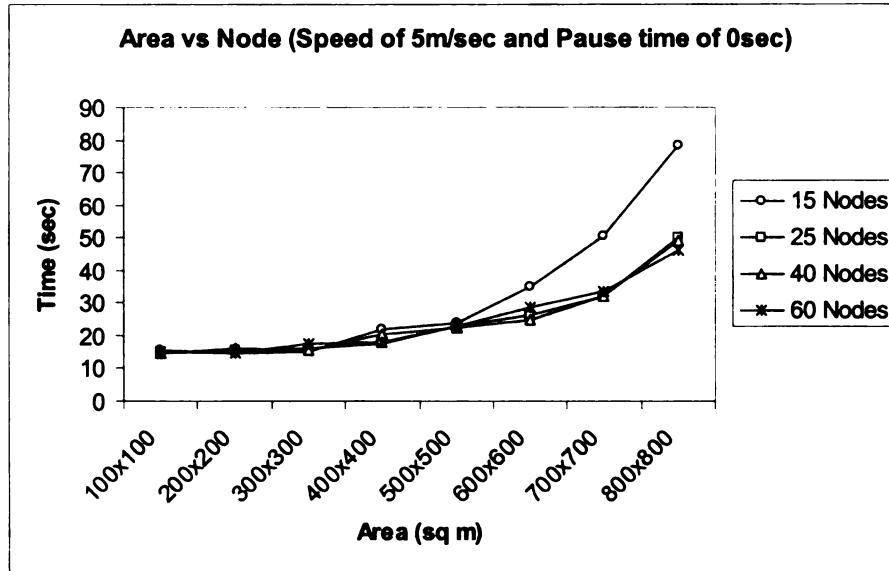


Figure 5.16: Performance Comparison for different number of nodes and areas.

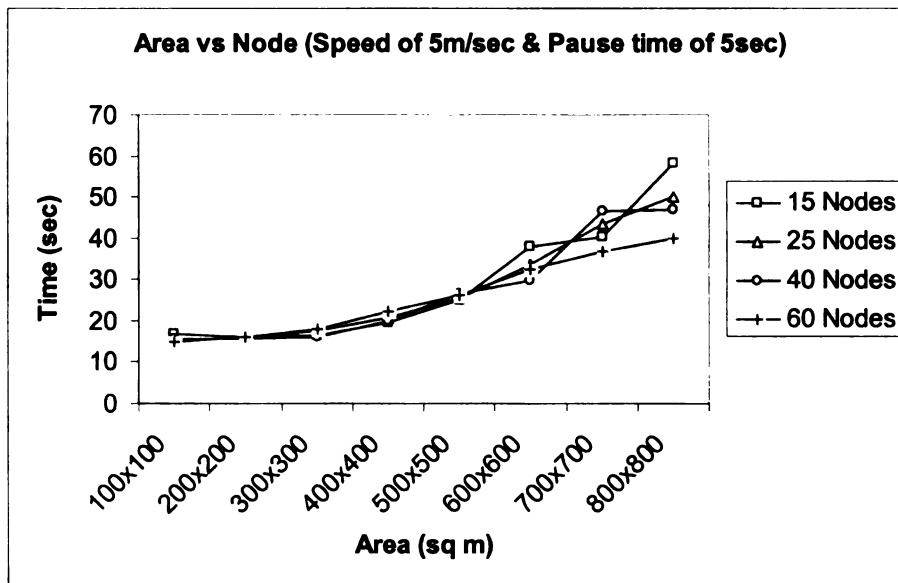


Figure 5.17: Performance Comparison for different number of nodes and areas.

In smaller areas, even with less number of nodes, the coverage region of nodes overlaps and hence the performance is hardly affected by the number of nodes in the environment. However, as the area in the simulation is increased, the nodes are scattered

and the coverage areas have very little overlap. Thus, when the number of hops from source to destination increases, the latency will increase.

5.4 File Fragmentation

The idea of fragmented file transfer has been around for a while. The concept of data buffering in transfers involving live data (eg. real-time video) can be considered analogous to fragmented file transfer. The buffered data can be looked at as if it is part of a larger file. This paper presents some preliminary results of simulations carried out in order to compare the performance of fragment file transfer with that of a complete file transfer. Table 5 shows the simulation parameters for this set. The file size of 100K and 200K were used. The fragments sizes for 200K file were: 100K, 50K, 20K, 10K, 5K and 4K. While for 100K they were: 50K, 20K, 10K, 5K and 2K. Results of the simulation are presented in Figure 5.18 and Figure 5.19. Two graphs do not show any conclusive results. This area needs further research. However, we do see that the transfer time for fragmented transfers lies close to the time it takes for the entire file to be transferred without fragmentation.

Table 5: Simulation parameters for testing effect of density.

Simulation Area (m ²)	500x500, 800x800
Total number of nodes	25
Total number of member nodes	10
Speed of nodes (m/sec)	5
Pause Times (Sec)	0, 5
Simulation Time (Sec)	400
Packet Size (bytes)	1024
Routing protocols	DSR
Number of scenario patters tested	10

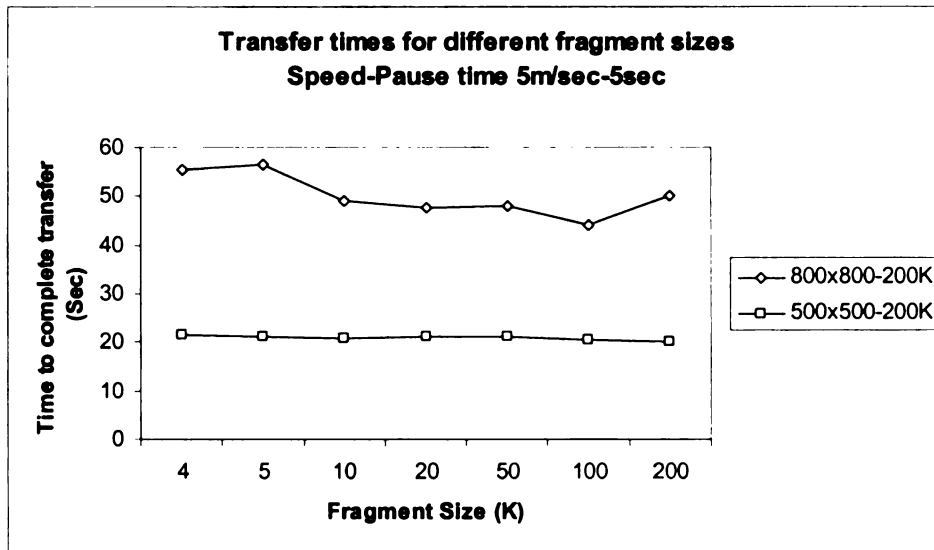


Figure 5.18: 200K file transferred in fragments.

In live mode transfers like real-time data, data can be buffered and then send to child nodes thus making sure that the real-time nature is maintained. A good fragment size would be the one that can carry more data in lesser time. There is always a trade-off between the size of the fragments used and the time it takes for finishing the complete file. Smaller fragments would tend to take more time due to various overhead associated with them. Also, in most of the ad-hoc routing protocols, after the network is initialized, some time is spent in route discovery. In case of small fragments, the effect of this route discovery would be more since the route discovery time would be comparable to the transfer times of smaller fragments. In Figure 5.18, fragment size of 10K seems to be a good trade-off. However, this result was not consistent with other simulations; hence more close examination of this parameter would be carried out in the future.

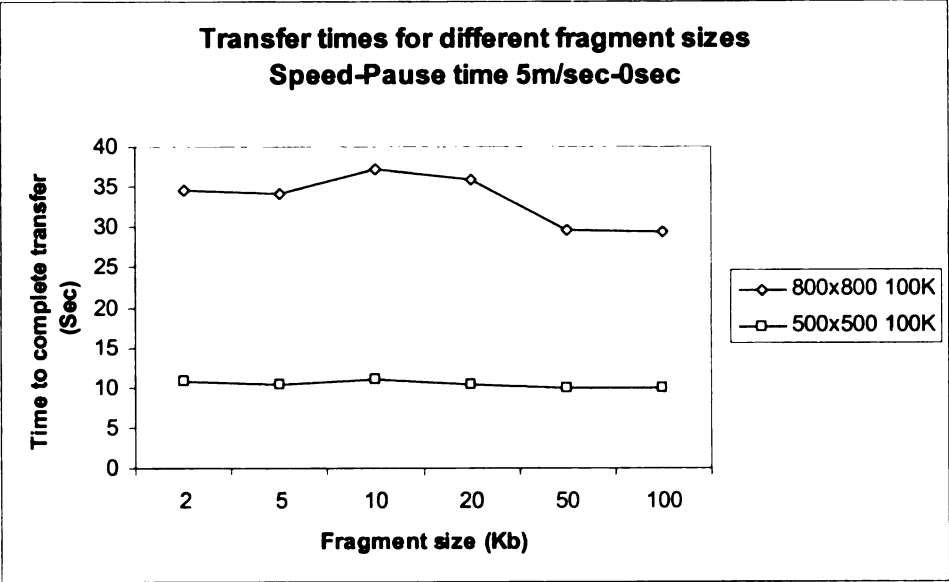


Figure 5.19: 100KB file transferred in fragments.

6 Conclusion and Future Work

This thesis proposes a new infrastructure-less need based security system using location based overlay multicast in ad-hoc environments. The idea of building several role-based priority trees in the same environment can find many interesting application in the future.

6.1 Conclusion

The main contributions of this research are:

- The thesis proposed a location-based prioritized overlay multicast model (L-POM). L-POM builds priority trees with certain nodes carrying important tasks in overlay networks, and rearranges low priority trees whenever some nodes temporarily move to a high priority network in ad hoc environment.
- To build an efficient overlay multicast tree, we design and implement an indoor location sensing system, LANDMARC. We show that active RFID is a viable and cost-effective candidate for indoor location sensing systems, and

LANDMARC may a good support system for the location-based overlay multicast approach.

- We study the feasibility of the approach by identifying a suitable unicast (ad-hoc) routing protocol, exploring to use location information to build more efficient trees, and studying the impact of density of wireless nodes, packet size, and fragmentation on performance

6.2 Future research for location sensing

In future, we have four problems to overcome in LANDMARC approach. The first problem is that none of the currently available RFID products provides the signal strength of tags directly. Instead, the reader reports “detectable” or “not detectable” in a given range. This forces LANDMARC to spend approximately one minute each time to scan the 8 discrete power levels and to estimate the signal strength of tags. By sending the signal strength information directly from readers, it will not only eliminate unnecessary processing time, but also reduce errors. This feature can easily be added as readers do have the signal strength information from tags.

The second problem is the long latency between a tracking tag being physically placed to its location being computed by the location server. There are two factors contributing to this long latency. One is the scanning time of different power levels as indicated above. This factor can be eliminated by sending the signal strength directly. The second factor is the time interval of emitting two consecutive IDs from an active tag. Our product has an average interval of 7.5 seconds in order to avoid signal collision for handling up to 500 tags. Depending on the total number of tags expected in a detectable

area, this time interval can be further reduced. RFID vendors should provide a mechanism to allow users to reconfigure the time interval.

The third problem is the variation of the behavior of tags. When employing the LANDMARC approach, the basic assumption is that all tags have roughly the same signal strength in emitting the RF signal. In our experiments we found that the power level detected by the same reader from two tags in an identical location may be different. A possible explanation for the difference may be due to the variation of the chips and circuits, as well as batteries. In fact, before our experiments, we had conducted a series of pre-tests to classify tags based on their signal strength. Repeated experiments are needed to classify these tags due to the potential signal collision causing a detectable tag not detectable. This is another factor decreasing the accuracy of the system.

Another problem is that all signals are not measured at the same time from neighbors.

If all the above problems can be overcome, the accuracy and latency will be greatly improved.

6.3 Future work for location based overlay multicast

Location information is important while building overlay trees. In the future, some form of location information will be used along with other factors in building efficient overlay trees. The concept of file fragmentation and simultaneous fragment download from multiple hosts has been around in the peer-to-peer world for quite a while. However, this concept has not been tried in wireless environments. The concept of

fragmented file transfer can be used in case of live data for buffered transfers. This thesis did not present a conclusive result for fragmented file transfers. However, it will be an area that would be closely looked at in the future.

7 BIBLIOGRAPHY

- [1] C. Gui and P. Mohapatra, "Efficient Overlay Multicast for Mobile Ad Hoc Networks," Wireless Communications and Networking Conference (WCNC), 2003
- [2] S. Banerjee and B. Bhattacharjee. Analysis of the NICE Application Layer Multicast Protocol. Technical report, UMIACSTR 2002-60 and CS-TR 4380, Department of Computer Science, University of Maryland, College Park, June 2002.
- [3] S.J. Lee, M. Gerla, and C.-C. Chiang, "On Demand Multicast Routing Protocol," In Proceedings of IEEE WCNC'99, pp. 1298-1302, September 1999.
- [4] Jorjeta Jetcheva and David B. Johnson, "Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks", Proceedings of the Second Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2001).
- [5] Johnson, D. and Maltz, D. (1996). "Dynamic source routing in ad hoc wireless networks," in Mobile Computing (ed. T. Imielinski and H. Korth), Kluwer Academic Publishers, Dordrecht, The Netherlands.
- [6] C. E. Perkins, "Ad-hoc on-demand distance vector routing," in MILCOM '97 panel on Ad Hoc Networks, Nov. 1997.
- [7] Charles E. Perkins and Pravin Bhagwat. Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers. In Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, pages 234–244, August 1994.

- [8] K. Chen and K. Nahrstedt, "Effective Location - Guided Tree Construction Algorithm for Small Group Multicast in MANET." In Proceedings of IEEE/Infocom'02, May, 2002.
- [9] David B. Johnson and David A. Maltz, "Dynamic source routing in ad hoc wireless networks". In Mobile Computing, edited by Tomasz Imielinski and Hank Korth, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.
- [10] Lionel M. Ni, Yunhao Liu, Yiu Cho Lau and Abhishek P. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID", IEEE PerCom 2003.
- [11] E. Bommaiah, M. Liu, A. MvAuley, and R. Talpade, "AMRoute: Ad Hoc Multicast Routing Protocol", Internet Draft, draft-manet-amroute-00.txt, (work in progress).
- [12] B. Karp, and H.T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," Proc. of ACM/IEEE MobiCom 2000, August 2000.
- [13] Y.-B. Ko, and N.H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," Proc. of ACM/IEEE MobiCom'98, October 1998.
- [14] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward, "A Distance Routing Effect Algorithm for Mobility (DREAM)," Proc. of ACM/IEEE MobiComm'98, October, 1998.
- [15] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "The case for Resilient Overlay Networks," Proceedings of the 8th Annual Workshop on Hot Topics in Operating Systems (HotOSVIII) , May 2001.
- [16] S. Shi and J. Turner, "Routing in Overlay Multicast Networks," In Proc. of IEEE INFOCOM, June 2002.
- [17] "The Global Surveyor Real-Time DGPS Service," Base Mapping and Geomatic Services and Sigem. Presented at GIS'99, Vancouver, Canada, March 1999.
- [18] G. Dommety and R. Jain, "Potential networking applications of global positioning systems (GPS)," Tech. Rep. TR-24, CS Dept., The Ohio State University, April 1996.
- [19] Yang-Hua Chu, Sanjay Rao, and Hui Zhang, "A Case for End System Multicast," In Proceedings of ACM Sigmetrics, Santa Clara, CA, 2000.

- [20] M. Kwon and S. Fahmy, "Topology-Aware Overlay Networks for Group Communication," In Proceedings of ACM NOSSDAV, May, 2002.
- [21] Vincent D. Park and M. Scott Corson, "Temporally-Ordered Routing Algorithm (TORA)," Internet-Draft, draft-ietf-manet-tora-spec-00.txt, November 1997.
- [22] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," In Proc. of the ACM/IEEE MobiCom, October 1998.
- [23] Y. Xue, and B. Li, "A Location-aided Power-aware Routing Protocol in Mobile Ad Hoc Networks," Proc. of IEEE Globecom 2001, San Antonio, Texas, November, 2001.
- [24] I. Stoica, T.S.E. Ng, and H. Zhang, "REUNITE: A recursive unicast approach to multicast," Proc. of INFOCOM 2000, March 2000.
- [25] E. M. Royer and C. E. Perkins, Multicast operation of the ad-hoc on-demand distance vector routing protocol," in Proc. of ACM/IEEE Intl. Conference on Mobile Computing and Networking (MOBICOM), pp. 207-218, Aug. 1999
- [26] R. Want et al., "The Active Badge Location System," ACM Trans. Information Systems, Jan. 1992, pp. 91-102.
- [27] P. Bahl and V. N. Padmanabhan, RADAR: An In-building RF-based User Location and Tracking System, Proceedings of IEEE INFOCOM 2000, Tel-Aviv, Israel (March 2000), <http://www.research.microsoft.com/~padmanab/papers/infocom2000.pdf>.
- [28] Jeffrey Hightower and Gaetano Borriello, "A Survey and Taxonomy of Location Sensing Systems for Ubiquitous Computing," CSE 01-08-03, University of Washington, Department of Computer Science and Engineering, Seattle, WA, Aug 2001, <http://www.cs.washington.edu/homes/jeffro/pubs/hightower2001survey/hightower2001survey.pdf>.
- [29] Nissanka B. Priyantha, Anit Chakraborty, and Hari Balakrishnan. The cricket location-support system. In Proceedings of MOBICOM 2000, pages32-43, Boston, MA, August 2000. ACM, ACM Press.
- [30] Andy Harter, Andy Hopper, Pete Steggles, Any Ward, and Paul Webster. The anatomy of a context-aware application. In Proceedings of the 5th Annual ACM/IEEE

International Conference on Mobile Computing and Networking (Mobicom 1999), pages 59-68, Seattle, WA, August 1999, ACM Press.

[31] Jeffrey Hightower, Chris Vakili, Caetano Borriello, and Roy Want, "Design and Calibration of the SpotON AD-Hoc Location Sensing System", UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, <http://www.cs.washington.edu/homes/jeffro/pubs/hightower2001design/hightower2001design.pdf>

[32] Aetherwire. <http://www.aetherwire.com/>

[33] Garmin Corporation. About GPS. Website,2001, <http://www.garmin.com/aboutGPS/>

[34] Mario Chiesa, Ryan Genz, Franziska Heubler, Kim Mingo, Chris Noessel, Natasha Sopieva, Dave Slocombe, Jason Tester, "RFID", March 04, 2002, <http://people.interaction-ivrea.it/c.noessel/RFID/research.html>

[35] AIM (Automatic Identification Manufacturers), the trade association for the Automatic Identification and Data Collection (AIDC) industry, Getting Started in RFID - A Step approach, 2002, http://www.aimglobal.org/technologies/rfid/resources/papers/rfid_basics_primer.htm

[36] Dharma Prakash Agrawal, Qing-An Zeng, "Introduction to Wireless and Mobile Systems", University of Cincinnati

MICHIGAN STATE UNIVERSITY LIBRARIES



3 1293 02461 4905