WEIGHTED RENDEZVOUS PLANNING FOR ENERGY EFFICIENT MOBILE-SINK PATH SELECTION IN WIRELESS SENSOR NETWORKS

PROJECT REPORT

Submitted by

PREETHA.S

Register No: 13MCO15

in partial fulfillment for the requirement of award of the degree

of

MASTER OF ENGINEERING

in

COMMUNICATION SYSTEMS

Department of Electronics and Communication Engineering

KUMARAGURU COLLEGE OF TECHNOLOGY

(An autonomous institution affiliated to Anna University, Chennai)

COIMBATORE - 641 049

ANNA UNIVERSITY: CHENNAI 600 025

APRIL-2014

BONAFIDE CERTIFICATE

Certified that this project report titled **"WEIGHTED RENDEZVOUS PLANNING FOR ENERGY EFFICIENT MOBILE-SINK PATH SELECTION IN WIRELESS SENSOR NETWORKS"** is the bonafide work of **S.PREETHA [Reg. No. 13MCO15]** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Ms.S.NAGARATHINAM
ASSISTANT PROFESSOR (SRG)
Department of ECE
Kumaraguru College of Technology
Coimbatore-641 049

SIGNATURE
Dr. RAJESWARI MARIAPPAN
HEAD OF THE DEPARTMENT
Department of ECE
Kumaraguru College of Technology
Coimbatore-641 049

The Candidate with university **Register No. 13MCO15** was examined by us in the project viva –voice examination held on............................

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

First I would like to express my praise and gratitude to the Lord, who has showered his grace and blessing enabling me to complete this project in an excellent manner. He has made all things in beautiful in his time.

I express my sincere thanks to our beloved Joint Correspondent, **Shri. Shankar Vanavarayar** for his kind support and for providing necessary facilities to carry out the project work.

I would like to express my sincere thanks to our beloved Principal **Dr. R.S.Kumar M.E., Ph.D.,** who encouraged me with his valuable thoughts.

I would like to express my sincere thanks and deep sense of gratitude to our HOD, **Dr. Rajeswari Mariappan M.E., Ph.D.,** for her valuable suggestions and encouragement which paved way for the successful completion of the project.

I am greatly privileged to express my deep sense of gratitude to the Project Coordinator **Ms.R.Hemalatha M.E.,** Associate Professor, Department of Electronics and Communication Engineering for her continuous support throughout the course.

In particular, I wish to thank and express my everlasting gratitude to the Supervisor **Ms.S.Nagarathinam M.E.,** Assistant Professor (SRG), Department of Electronics and Communication Engineering**,** for her expert counselling in each and every steps of project work and I wish to convey my deep sense of gratitude to all teaching and non-teaching staff members of ECE Department for their help and cooperation.

Finally, I thank my parents and my family members for giving me the moral support in all of my activities and my dear friends who helped me to endure my difficult times with their unfailing support and warm wishes.

ABSTRACT

Wireless sensor networks (WSNs) uses mobile sink to reduce the energy consumption of nodes and to prevent the formation of energy holes. Within a given time constraint all sensed data should be collected. Here mobile-sink node only visits rendezvous points (RPs). If Sensor nodes are not RPs then forward its sensed data via multi hopping to the nearest RP. The basic problem is computing a tour that visits all these RPs within a given delay bound. The NP-hard problem is identifying the optimal tour. In order to overcome this, a heuristic called weighted rendezvous planning (WRP) has been used, and then each sensor node is assigned a weight depending on the hop distance from the tour and the number of data packets. WRP is calculated using extensive computer simulation and results gives that WRP enables a mobile sink to get all sensed data within a given constraint while conserving the energy expenditure of sensor nodes. As compared with existing algorithms WRP reduces energy consumption and increases network lifetime by using two mobile sink than a single mobile sink.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

WRP	Weighted Rendezvous Planning
RP	Rendezvous Point
WSN	Wireless Sensor Network
QOS	Quality of Service
DEETP	Delay-aware energy efficient path
CB	Cluster Based
RD-VT	Rendezvous Design for Variable Tracks
RP-UG	Rendezvous Planning Utility based Greedy
SMT	Steiner Minimum Tree
TDMA	Time division multiple access
NS	Network Simulator
WAN	Wireless Area Network
ME	Mobile Element
RREP	Route Reply
RREQ	Route Request
ADOV	Ad hoc On-Demand Distance Vector

CHAPTER 1
INTRODUCTION

1.1 WIRELESS SENSOR NETWORKS

Wireless Sensor Network (WSN) is a collection of spatially deployed wireless sensors by which to monitor various changes of environmental conditions (e.g., forest fire, air pollutant concentration, and object moving) in a collaborative manner without relying on any underlying infrastructure support. Recently, a number of research efforts have been made to develop sensor hardware and network architectures in order to effectively deploy WSNs for a variety of applications.

Due to a wide diversity of WSN application requirements, however, a general-purpose WSN design cannot fulfill the needs of all applications. Many network parameters such as sensing range, transmission range, and node density have to be carefully considered at the network design stage, according to specific applications. To achieve this, it is critical to capture the impacts of network parameters on network performance with respect to application specifications.

Wireless sensor networks are deployed to monitor the sensing field and gather information from it. Traditionally, two approaches can be adopted to accomplish the data collection task: through direct communication, and through multi-hop forwarding. In the first case, sensor nodes upload data directly to the sink through one-hop wireless communication, which may result in long communication distances and degrade the energy efficiency of sensor nodes. On the other hand, with multi-hop forwarding, data are reported to the sink through multiple relays, and the communication distance is reduced. However, since nodes near the sink generally have a much heavier forwarding load, their energy may be depleted very fast, which degrades the network performance.

1.2 ROUTING

Routing is the process of selecting paths in a network to forward the data packets. In packet switching networks, routing directs packet forwarding, the transit of logically addressed packets from their source toward their ultimate destination through intermediate nodes; typically hardware devices called routers, bridges, gateways, firewalls, or switches. Most routing algorithms use only one network path at a time, but multipath routing techniques enable the use of multiple alternative paths.

Routing Protocol

The routing protocol [2] specifies how routers in a network share information with each other and report changes. The routing protocol enables a network to make dynamic adjustments to its conditions, so routing decisions do not have to be predetermined and static. A routing protocol shares this information first among immediate neighbors, and then throughout the network. The different ad hoc routing protocols are listed below.

AD HOC ROUTING PROTOCOL

An ad-hoc routing protocol [11] is a standard that controls how nodes decide which way to route packets between computing devices in a mobile ad hoc network. Some of the ad-hoc network routing protocols are

1. Pro-active (table-driven) routing

2. Reactive (on-demand) routing

3. Hybrid (both pro-active and reactive) routing

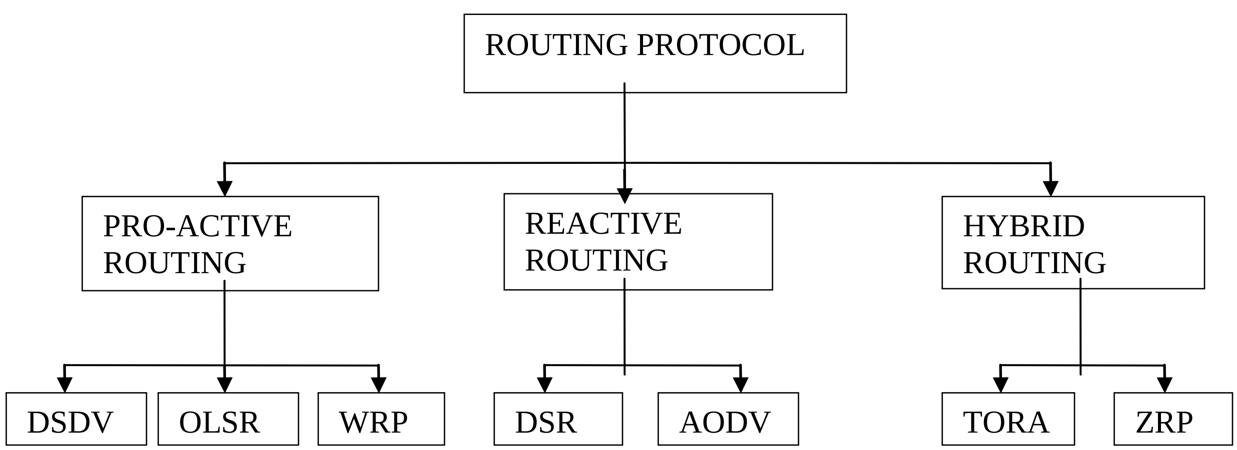

Fig 1.1: Classification of Ad hoc routing Protocols

1.3 NATURES OF ROUTING

Since a distributed network has multiple nodes and services many messages, and each node is a shared resource, many decisions must be made. There may be multiple paths from the source to the destination. Therefore, message routing is an important topic. The main performance measures affected by the routing scheme are throughput (quantity of service) and average packet delay (quality of service). Routing schemes should also avoid both deadlock and live lock (see below). Routing methods can be fixed (i.e. pre-planned), adaptive, centralized, distributed, broadcast, etc. Perhaps the simplest routing scheme is the token ring [Smythe 1999]. Here, a simple topology and a straightforward fixed protocol result in very good reliability and precomputable QoS. A token passes continuously around a ring topology. When a node desires to transmit, it captures the token and attaches the message. As the token passes, the destination reads the header, and captures the message. In some schemes, it attaches a 'message received' signal to the token, which is then received by the original source node. Then, the token is released and can accept further messages. The token ring is a completely decentralized scheme that effectively uses TDMA. Though this scheme is very reliable, one can see that it results in a waste of network capacity. The token must pass once around the ring for each message. Therefore, there are various modifications of this scheme, including using several tokens, etc.

Fixed routing schemes often use Routing Tables that dictate the next node to be routed to, given the current message location and the destination node. Routing tables can be very large for large networks, and cannot take into account real-time effects such as failed links, nodes with backed up queues, or congested links.

Adaptive routing schemes depend on the current network status and can take into account various performance measures, including cost of transmission over a given link, congestion of a given link, reliability of a path, and time of transmission. They can also account for link or node failures.

Routing has two main functions: route discovery and packet forwarding. The former is concerned with discovering routes between nodes, whereas the latter is about sending data packets through the previously discovered routes. There are different types of ad hoc routing protocols. One can distinguish proactive and reactive protocols. Protocols of the latter category are also called on-demand protocols. Another type of classification distinguishes routing table based protocols (e.g., DSDV) and source routing protocols (e.g., DSR).

The major requirements of a routing protocol

- Minimum route acquisition delay
- Quick route reconfiguration in the case of path breaks.
- Loop-free routing
- Distributed routing protocol
- Low control over-head
- Scalability with network size
- QoS support as demanded by the application
- Support of time-sensitive traffic and
- Security and privacy

Proactive Routing Protocols (Table-driven)

- Nodes exchange routing information periodically in order to maintain consistent and accurate information.
- To transmit data to a destination, path can be computed rapidly based on the updated information available in the routing table.
- The disadvantage of using a proactive protocol is high overhead needed to dynamic topology that might require a large number of routing updates.
- Each node maintains a routing table, with an entry for each possible destination address, next hop on the shortest path to that destination, shortest known distance to this destination, and a destination sequence number that is created by the destination itself.

This type of protocols maintains fresh lists of destinations and their routes by periodically distributing routing tables throughout the network.

Destination-Sequenced Distance Vector routing

Destination-Sequenced Distance-Vector Routing (DSDV) is a table-driven routing scheme for ad hoc mobile networks. Each entry in the routing table contains a sequence number, the sequence numbers are generally “even number” if a link is present; else, an odd number is used. The sequence number is generated by the destination. Routing information is distributed between nodes by sending full dumps infrequently and smaller incremental updates more frequently.

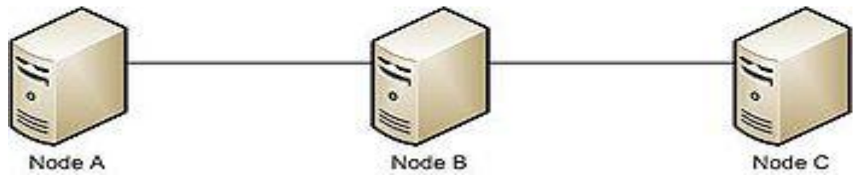

Fig 1.2 Destination-Sequenced Distance Vector routing

Optimized Link State Routing Protocol

Optimized Link State Routing Protocol (OLSR) is an IP routing protocol which is optimized for mobile ad-hoc networks but can also be used on other wireless ad-hoc networks. OLSR is a proactive link-state routing protocol which uses Hello and Topology Control (TC) messages to discover and then disseminate link state information throughout the mobile ad-hoc network. Individual nodes use this topology information to compute next hop destinations for all nodes in the network using shortest hop forwarding paths. Using Hello messages the OLSR protocol at each node discovers 2-hop neighbor information and performs a distributed election of a set of multipoint distribution relays.

Wireless Routing Protocol

The Wireless Routing Protocol (WRP) is a proactive unicast routing protocol for mobile ad-hoc networks (MANETs). WRP uses an enhanced version of the distance-vector routing protocol, which uses the Bellman-Ford algorithm to calculate paths. Because of the mobile nature of the nodes within the MANET, the protocol introduces mechanisms which reduce route loops and ensure reliable message exchange.

Reactive Routing Protocols (On-demand)

- Route discovery mechanism is initiated only when a node does not know the path to a destination it wants to communicate with.

- In case of mobile ad hoc network, reactive routing protocols have been demonstrated to perform better with significantly lower changes that may occur in node connectivity, and yet are able to reduce/eliminate routing overhead in periods or areas of the network in which changes are less frequent.
- A reactive routing has two main operations. Route discovery (usually broadcasting using a form of controlled flooding) and route maintenance. Various reactive protocols have been proposed in literature such as Ad Hoc On-demand vector (AODV), Dynamic source routing (DSR),
- Temporary Ordered Routing Algorithm (TORA), etc

This type of protocols finds a route on demand [8] by flooding the network with Route Request packets. The main disadvantages of such algorithms are:

1. High latency time in route finding.
2. Excessive flooding can lead to network clogging.

Distance-vector routing protocol

1. A distance-vector routing protocol is a reactive protocol for mobile ad-hoc networks (MANETs). A distance-vector routing protocol requires a router to inform its neighbor when a change is detected in the topology of a network. The distance-vector routing protocols have less computational complexity and message overhead. Distance Vector means that routers are advertised as vector of distance and Direction. Routers using distance vector protocol do not have knowledge of the entire path to a destination. As the name suggests the DV protocol is based on calculating the direction and distance to any link in a network. The cost of reaching a destination is calculated using various route metrics. RIP uses the hop count of the destination whereas IGRP takes into account other information such as node delay and available bandwidth. Updates are performed periodically in a distance-vector protocol where all or part of a router's routing table is sent to all its neighbors that are configured to use the same distance-vector routing protocol.

Hybrid (both pro-active and reactive) routing

This type of protocols combines the advantages of proactive and of reactive routing. The routing is initially established with some proactively prospected routes and then serves the demand from additionally activated nodes through reactive flooding. The choice for one or the other method requires predetermination for typical cases. The main disadvantages of such algorithms are:

1. Advantage depends on amount of nodes activated.
2. Reaction to traffic demand depends on gradient of traffic volume.

Temporally-ordered routing algorithm

The TORA attempts to achieve a high degree of scalability using a "flat", non-hierarchical routing algorithm. TORA builds and maintains a Directed Acyclic Graph rooted at a destination. No three nodes may have the same height. Information may flow from nodes with higher heights to nodes with lower heights. Information can therefore be thought of as a fluid that may only flow downhill. By maintaining a set of totally-ordered heights at all times, TORA achieves loop-free multipath routing, as information cannot 'flow uphill' and so cross back on itself. The key design concept of TORA is localization of control messages to a very small set of nodes near the occurrence of a topological change. The protocol performs three basic functions: Route creation, Route maintenance, Route erasure. During the route creation and maintenance phases, nodes use a height metric to establish a directed acyclic graph (DAG) rooted at destination.

Zone Routing Protocol

Zone Routing Protocol or ZRP was the first hybrid routing protocol with both a proactive and a reactive routing component. ZRP was first introduced by Haas in 1997. ZRP is proposed to reduce the control overhead of proactive routing protocols and decrease the latency caused by routing discover in reactive routing protocols. ZRP defines a zone around each node consisting of its k-neighborhood (e. g. k=3). In ZRP, the distance and a node, all nodes within -hop distance from node belongs to the routing zone of node.

1.4 AD HOC ON-DEMAND DISTANCE VECTOR ROUTING

In ad hoc mobile [5] networks, routes are mainly multihop because of the limited radio propagation range and topology changes frequently and unpredictably since each network host moves randomly. Hence on-demand routing protocol [8] has been proposed for ad hoc networks. Network hosts maintain route table entries only to destinations that they communicate with.

Ad hoc On-Demand Distance Vector (AODV) Routing is a routing protocol for mobile ad hoc networks (MANETs) and other wireless ad-hoc networks. AODV is capable of both unicast and multicast routing. It is a reactive routing protocol, meaning that it establishes a route to a destination only on demand. It is an “on demand algorithm”, meaning that it builds routes between nodes only as desired by source nodes. It maintains these routes as long as they are needed by the sources. Additionally, AODV forms trees which connect multicast group members. The trees are composed of the group members and the nodes needed to connect the members. AODV uses sequence numbers to ensure the freshness of routes [6]. It is loop-free, self-starting, and scales to large numbers of mobile nodes. AODV builds routes using a route request / route reply query cycle. When a source node desires a route to a destination for which it does not already have a route, it broadcasts a route request (RREQ) packet across the network. Nodes receiving this packet update their information for the source node and set up backwards pointers to the source node in the route tables. In addition to the source node's IP address, current sequence number, and broadcast ID, the RREQ also contains the most recent sequence number for the destination of which the source node is aware. A node receiving the RREQ may send a route reply (RREP) if it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the RREQ. If this is the case, it unicasts a RREP back to the source. Otherwise, it rebroadcasts the RREQ. Nodes keep track of the RREQ's source IP address and broadcast ID. If they receive a RREQ which they have already processed, they discard the RREQ and do not forward it. Once the source node receives the RREP, it may begin to forward data packets to the destination. If the source later receives a RREP containing a greater sequence number or contains the same sequence number with a smaller hop count, it may update its routing information for that destination and begin using the better route. As long as the route remains active, it will continue to be maintained [6]. A route is considered active as long as there are data packets periodically travelling from the source to the destination along that path. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. If a link break occurs while the route is active, the node upstream of the break propagates a route error (RERR) message to the source node to inform it of the now unreachable destination(s). After receiving the RERR, if the source node still desires the route, it can reinitiate route discovery. Multicast routes are set up in a similar manner.

A node wishing to join a multicast group broadcasts a RREQ with the destination IP address set to that of the multicast group and with the 'J'(join) flag set to indicate that it would like to join the group. Any node receiving this RREQ that is a member of the multicast tree that has a fresh enough sequence number for the multicast group may send a RREP. As the RREPs propagate back to the source, the nodes forwarding the message set up pointers in their multicast route tables. As the source node receives the RREPs, it keeps track of the route with the freshest sequence number, and beyond that the smallest hop count to the next multicast group member. After the specified discovery period, the source nodes will unicast a Multicast Activation (MACT) message to its selected next hop. This message serves the purpose of activating the route. A node that does not receive this message that had set up a multicast route pointer will timeout and delete the pointer. If the node receiving the MACT was not already a part of the multicast tree, it will also have been keeping track of the best route from the RREPs it received. Hence it must also unicast a MACT to its next hop, and so on until a node that was previously a member of the multicast tree is reached. AODV maintains routes for as long as the route is active. This includes maintaining a multicast tree for the life of the multicast group. Because the network nodes are mobile, it is likely that many link breakages along a route will occur during the lifetime of that route.

CHAPTER – 2

LITERATURE SURVEY

Energy-Quality Tradeoffs for Target Tracking In Wireless Sensor Networks

The tradeoffs involved in the energy-efficient localization and tracking of mobile targets by a wireless sensor network. This work focuses on building a framework for evaluating the fundamental performance of tracking strategies in which only a small portion of the network is activated at any point in time. We first compare naive network operation with random activation and selective activation. In these strategies the gains in energy-savings come at the expense of increased uncertainty in the location of the target, resulting in reduced quality of tracking. We show that selective activation with a good prediction algorithm is a dominating strategy that can yield orders-of-magnitude energy savings with negligible difference in tracking quality. We then consider duty-cycled activation and show that it offers a flexible and dynamic tradeoff between energy expenditure and tracking error when used in conjunction with selective activation.

There is an emerging trend towards the use of sophisticated wireless networks of unattended sensor devices for intelligence gathering and environmental monitoring. One \ canonical application of sensor networks that has received considerable attention in the literature is the tracking of a mobile target (point source) by the network. In a tracking scenario, information obtained from nodes far away from the region of activity is of little or no use. For a typical sensor network with a large number of nodes, a major portion of these falls in the above category. In addition, if the nodes are densely deployed information obtained from some sensors close to the region of activity might be redundant. An obvious way to save energy is to switch on only a subset of the sensor nodes. We discuss in this paper various possible activation strategies: (1) naive activation, (2) randomized activation (3) selective activation based on trajectory prediction and (4) duty-cycled activation. In these sensor activation strategies, energy savings come at the expense of a reduction in the quality of tracking. In other words, relying on the information provided by a small subset of the sensor nodes results in an increased uncertainty in the sensed location of the mobile. In this paper we study the energy-quality tradeoffs involved by building a model to quantify both the energy expenditure and the quality of tracking. Also for a particular strategy, we study the impact of the following: a) deployed/activated density of sensors

b) their sensing range c) capabilities of activated and un-activated nodes d) the target's mobility model.

Our efforts are not directed at proposing new techniques for mobile tracking. Rather the focus is on the evaluation and analysis of general strategies which may be incorporated into a real system. We start with a simple model for tracking and substantiate the intuition that it is possible to obtain orders of magnitude savings in energy while keeping the uncertainty within acceptable limits. We also discuss the extensions of the model to relate closely with real life scenarios. The results in this work are a first step in our attempt to understand the fundamental bounds on the the tracking quality that can be obtained under various energy constraints and sensor models.

Advantages

- Energy-efficient localization and tracking of mobile targets.

Drawbacks

- Reduced quality of tracking performance.

DCTC: Dynamic Convoy Tree-Based Collaboration for Target Tracking In Sensor Networks

Most existing work on sensor networks concentrates on finding efficient ways to forward data from the information source to the data centers, and not much work has been done on collecting local data and generating the data report. This paper studies this issue by proposing techniques to detect and track a mobile target. We introduce the concept of dynamic convoy tree-based collaboration, and formalize it as a multiple objective optimization problem which needs to find a convoy tree sequence with high tree coverage and low energy consumption. We propose an optimal solution which achieves 100% coverage and minimizes the energy consumption under certain ideal situations. Considering the real constraints of a sensor network, we propose several practical implementations: the conservative scheme and the prediction-based scheme for tree expansion and pruning; the sequential and the localized reconfiguration schemes for tree reconfiguration. Extensive experiments are conducted to compare the practical implementations and the optimal solution. The results show that the prediction-based scheme outperforms the conservative scheme and it can achieve similar coverage and energy consumption to the optimal solution.

The experiments also show that the localized reconfiguration scheme outperforms the sequential reconfiguration scheme when the node density is high, and the trend is reversed when the node density is low. Most existing researches in sensor networks, e.g., the directed diffusion, LEACH, and two-tier data dissemination (TTDD), concentrate on finding efficient ways to forward the data report to the data center, and not much work has been done on how to detect the mobile target and generate robust and reliable reports in an energy efficient way. Recently, Chu et al. studied the problem of tracking a mobile target using an information-driven approach. However, their approach assumed that a single node close to a target can detect the status of the target, and did not consider the collaboration among nodes that can detect the target at the same time. Since sensor nodes deployed in current sensor networks do not have a large sensing distance, or a high level of sensing accuracy and node reliability, Cerpa et al suggested that multiple nodes surrounding the target should collaborate to make the collected information more complete, reliable, and accurate. However, no concrete algorithm was given.

A big challenge of implementing the DCTC framework is how to reconfigure the convoy tree in an energy efficient way as the target moves. To address this problem, we first formalize it as an optimization problem of finding a min-cost convoy tree sequence with high tree coverage, and give an optimal solution (o-DCTC) based on dynamic programming. Considering the constraints of sensor networks, we propose some practical solutions. Specifically, we propose two tree expansion and pruning schemes: the conservative scheme and the prediction- based scheme; and two tree reconfiguration schemes: the sequential reconfiguration and the localized reconfiguration. We also evaluate the performance of the optimal solution and the practical implementations through extensive simulations. Based on the simulation results, when the same reconfiguration scheme is used, the prediction-based scheme outperforms the conservative scheme and it can achieve a similar coverage and energy consumption to the optimal solution. When using the same scheme for tree expansion and pruning, the localized reconfiguration scheme outperforms the sequential reconfiguration scheme when the node density is high, and the trend is reversed when the node density is low.

Advantages

- It should promptly provide robust and reliable status information about the mobile target and the region around it in an energy efficient way, and the network should forward this information to the sinks in a fast and energy efficient way.

Drawbacks

- The performance metrics are not good when the node density is low.

Coverage and Connectivity Issues in Wireless Sensor Networks

Wireless sensor networks have inspired tremendous research interest in since the mid-1990s. Advancement in wireless communication and micro electromechanical systems (MEMSs) have enabled the development of low-cost, low power, multifunctional, tiny sensor nodes that can sense the environment, perform data processing, and communicate with each other untethered over short distances. A typical wireless sensor network consists of thousands of sensor nodes, deployed either randomly or according to some predefined statistical distribution, over a geographic region of interest. A sensor node by itself has severe resource constraints, such as low battery power, limited signal processing, limited computation and communication capabilities, and a small amount of memory; hence it can sense only a limited portion of the environment. However, when a group of sensor nodes collaborate with each other, they can accomplish a much bigger task efficiently. One of the primary advantages of deploying a wireless sensor network is its low deployment cost and freedom from requiring a messy wired communication backbone, which is often infeasible or economically inconvenient. Wireless sensor networks ensure a wide range of applications, starting from security surveillance in military and battlefields, monitoring previously unobserved environmental phenomena, smart homes and offices, improved healthcare, industrial diagnosis, and many more.

Optimal resource management and assuring reliable QoS are two of the most fundamental requirements in ad hoc wireless sensor networks. Sensor deployment strategies play a very important role in providing better QoS, which relates to the issue of how well each point in the sensing field is covered. However, due to severe resource constraints and hostile environmental conditions, it is nontrivial to design an efficient deployment strategy that would minimize cost, reduce computation, minimize node-to-node communication, and provide a high degree of area coverage, while at the same time maintaining a globally connected network is nontrivial.

Challenges also arise because topological information about a sensing field is rarely available and such information may change over time in the presence of obstacles. Many wireless sensor network applications require one to perform certain functions that can be measured in terms of area coverage. In these applications, it is necessary to define precise measures of efficient coverage that will impact overall system performance.

Historically, three types of coverage have been defined by Gage:

1. Blanket coverage — to achieve a static arrangement of sensor nodes that maximizes the detection rate of targets appearing in the sensing field

2. Barrier coverage — to achieve a static arrangement of sensor nodes that minimizes the probability of undetected penetration through the barrier

3. Sweep coverage — to move a number of sensor nodes across a sensing field, such that it addresses a specified balance between maximizing the detection rate and minimizing the number of missed detections per unit area

Advantages

- This service facilitated by a better coverage is application specific.
- Reduced computation and communication cost.

Drawbacks

- Still require better solutions to overcome challenges to the coverage–connectivity problem.

Wireless Sensor Network Energy-Adaptive Mac Protocol

Wireless Sensor Networks (WSNs) provide a valuable capability to autonomously monitor remote activities. Their limited resources challenge WSN medium access control (MAC) layer designers to adequately support network services while conserving limited battery power. This paper presents an energy adaptive WSN MAC protocol, Gateway MAC (GMAC), which implements a new cluster-centric paradigm to effectively distribute cluster energy resources and extend network lifetime. G-MAC's centralized cluster management function offers significant energy savings by leveraging the advantages of both contention and contention-free protocols. A centralized gateway node collects all transmission requirements during a contention period and then schedules their distributions during a reservation-based, contention-free period. With minimal overhead, the gateway duties are efficiently rotated based upon available resources to distribute the increased network management energy requirements among all of the nodes.

The G-MAC protocol's innovative architecture is motivated by the necessity for resource-challenged WSN mote sensor platforms to minimize the time radios spend in both the idle and the receive modes. Research shows that wireless platform transceivers expend a significant amount of energy receiving on an idle channel, and many of the WSN mote platform radios expend more energy in receive than in transmit mode. G-MAC provides effective network control mechanisms to maximize sleep durations, minimize idle listening, and limit the amount of cluster control traffic overhead. G-MAC dynamically rotates point coordination duties among all the nodes to distribute the management energy costs, to allow other nodes to sleep longer, and to extend the network's lifetime.

Advantages

- Significantly reduces energy consumption.
- Reduced overhearing or idle listening overhead and increases the network lifetime

Drawbacks

- Still networks require delicate tradeoffs in energy, latency, and throughput.

Energy Efficient Sleep Schedule for Achieving Minimum Latency In Query Based Sensor Networks

Energy management in sensor networks is crucial to prolong the network lifetime. Though existing sleep scheduling algorithms save energy, they lead to a large increase in end-to-end latency. We propose a new Sleep schedule (Q-MAC) for Query based sensor networks that provide minimum end-to-end latency with energy efficient data transmission. Whenever there is no query, the radios of the nodes sleep more using a static schedule. Whenever a query is initiated, the sleep schedule is changed dynamically.

Based on the destination's location and packet transmission time, we predict the data arrival time and retain the radio of a particular node, which has forwarded the query packet, in the active state until the data packets are forwarded. Since our dynamic schedule alters the active period of the intermediate nodes in advance by predicting the packet arrival time, data is transmitted to the sink with low end-to-end latency. The objectives of our protocol are to (1) minimize the end-to-end latency by alerting the intermediate nodes in advance using the dynamic schedule (2) reduce energy consumption by activating the neighbor nodes only when packets (query and data) are transmitted. Simulation results show that Q-MAC performs better than S-MAC by reducing the latency up to 80% with minimum energy consumption.

In query based sensor networks, the sensors report their results in response to an explicit request from the user. Users input the queries at the sink that describes the data they wish to collect. In a home network, user may send a query for eg. “Whether the gas tank should be refilled or the lights are ‘on’”. Based on the query for a particular detail, data can be collected from the corresponding subset of nodes in the complete network. The flow of data packets from the sensors to sink can be classified as broadcast, unicast or multicast based on the queries.

Advantages

- It provides minimum end-to-end latency in query based sensor networks with low energy consumption.

Drawbacks

- Data collection rate is not achievable i.e. low throughput ratio.

CHAPTER-3
EXISTING METHOD

3.1 INTRODUCTION

Existing methods on using a mobile sink in WSNs can be grouped into two categories: 1) direct, where a mobile sink visits each sensor node and collects data via a single hop; and 2) rendezvous, where a mobile sink only visits nodes designated as RPs. The main goal of protocols in category 1 is to minimize data collection delays,

Direct

Initial studies used a mobile sink that visits sensor nodes randomly and transport collected data back to a fixed sink node. An example is the use of animals as mobile-sink nodes to assist in data collection from sensor nodes scattered on a large farm. To reduce the latency of visiting each sensor node randomly, researchers have proposed TSP-based data collection methods. In essence, the problem is reduced to finding the shortest traveling path that visits each sensor node For example, TSP with neighborhood involves finding the shortest traveling tour for a mobile-sink node that passes through the communication range of all sensor nodes. Another TSP-based algorithm called label-covering considers a WSN as a complete graph. For each edge, it calculates a cost and associates a label set. The cost of an edge is the Euclidean distance between nodes, whereas the label set contains sensor nodes whose transmission range intersects with the given edge. The label-covering algorithm selects the minimum number of edges where their associated label set covers all sensor nodes. Rendezvous The problem with collecting data directly from sensor nodes is that it becomes impractical when there are a large number of sensor nodes. Visiting each sensor node increases the mobile sink's traveling path length and results in sensor nodes experiencing buffer overflow due to data collection delays. To address this problem, researchers have proposed a rendezvous-based model, in which a mobile sink only visits a subset of sensor nodes called RPs.

Fixed

In the studies conducted in the path of the mobile sink is fixed, and sensor nodes are randomly deployed near the sink's traveling path. Sensor nodes that are inside a mobile sink's communication range play the role of RPs and collect data from other sensor nodes. An example application is a traffic management system where mobile sinks are public buses that roam a city to collect data from sensor nodes placed on buildings. In these approaches, the length of the traveling path is not dependent on the buffer size of sensor nodes or application deadline. Hence, the buffer of RPs may overflow or packets may expire before they are collected by the sink.

Unconstrained: In a WSN with a static sink node and a mobile element (ME) is assumed to collect data from RPs. Moreover, RPs performs data aggregation. An algorithm called RD-VT is proposed with the objective of identifying a traveling path that is shorter in duration than the packet delivery time. The algorithm first constructs a Steiner minimum tree (SMT) rooted at the sink node. RD-VT then starts from the sink's position and traverses the SMT in preorder until the shortest distance between visited nodes is equal to the required packet delivery time. Since, in an SMT, a Steiner point may be a physical position and does not correspond to the position of a sensor node, RD-VT[16] replaces these virtual RPs with the closest sensor nodes. A major limitation of RD-VT is that traversing the SMT in preorder leads to the selection of RPs that in turn results in long data forwarding paths to sensor nodes located in different parts of the SMT. As a result, RD-VT causes nodes to have an unbalanced data forwarding load and energy consumption. Also RP-UG[17] does not balance the energy consumption and reduces network lifetime.

In spite of the diverse applications, WSNs face a number of unique technical challenges due to their inherent energy and bandwidth limitations, ad hoc deployment, and unattended operation, etc.,. Unfortunately, very little previous works on distributed systems can be applied to WSNs, since the underlying assumptions have changed dramatically. Therefore, innovative energy-aware, scalable, and robust algorithms for distributed signal processing in WSNs are highly required. A problem that is closely related is the localized topology control, which maintains energy-efficient network connectivity by controlling the transmission power at each node, or selecting a small subset of the local links of a node.

Since nodes often run on batteries that are generally difficult to be recharged once deployed, energy efficiency is a critical feature of WSNs for the purpose of extending the network lifetime. Target tracking in WSNs has been studied extensively. Due to the limited sensing capability and limited resources for communications and computation, collaborative resource management is required to trade-off between the tracking accuracy. Therefore, energy-efficient target tracking should improve the tradeoff between energy efficiency and tracking performance—e.g., by improving energy efficiency at the expense of a relatively small loss on tracking performance. For target tracking applications, idle listening is a major source of energy waste. To reduce the energy consumption during idle listening, duty cycling is one of the most commonly used approaches. The idea of duty cycling is to put nodes in the sleep state for most of the time, and only wake them up periodically. In certain cases, the sleep pattern of nodes may also be explicitly scheduled, i.e., forced to sleep or awakened on demand. This is usually called sleep scheduling.

As a compensation for tracking performance loss caused by duty cycling and sleep scheduling, proactive wake up has been studied for awakening nodes proactively to prepare for the approaching target. However, most existing efforts about proactive wake up simply awaken all the neighbor nodes in the area, where the target is expected to arrive, without any differentiation. Based on target prediction, it is possible to sleep-schedule nodes precisely, so as to reduce the energy consumption for proactive wake up. For example, if nodes know the exact route of a target, it will be sufficient to awaken those nodes that cover the route during the time when the target is expected to traverse their sensing areas but not achieve that much target performance.

Drawbacks

- However, if energy efficiency is enhanced, the quality of service (QoS) of target tracking is highly likely to be negatively influenced. For example, forcing nodes to sleep may result in missing the passing target and lowering the tracking coverage.
- Sleep scheduling inevitably increases the probability of losing track of the object when the sensor nodes that should be active are asleep.

Thus, the established connection routes between senders and receivers are likely to be broken during video transmission. It causes interruptions, freezes, or jerkiness in the received video signal. An end-to end connection route in wireless ad hoc networks generally consists of multiple wireless links, resulting in higher random packet loss than single hop wireless connections in wireless networks with infrastructure, such as base stations. The delay and loss sensitive nature of interactive video applications, make video communication over wireless ad hoc networks a challenging proposition [16]. Multicast over wireless networks is an important and challenging goal, but several issues must be addressed before many group applications can be deployed on a large scale. Multicasting is a more efficient method of supporting group communication than unicasting or broadcasting, as it allows transmission and routing of packets to multiple destinations using fewer network resources. The shortest path is to be determined by implementing the Ad hoc on Demand Distance Vector routing protocol in the wireless simulation environment for AODV in wireless environment [9]. The existing method deals with 100 nodes in the network with the packet size of 2000Bytes. The video signal is splitted into five parts and transmitted via different paths.

Algorithm

1. Source broadcast the RREQ packet to all the nodes in the network in order to discover the paths for data transmission.
2. Once the source receives the RREP packet, wait for a certain amount of time, then from received paths choose those ones that have the least active neighbors and starts load balancing data transfer on these paths.
3. The video packets are splitted into multiple parts and are transmitted via different paths.
4. Multicasting method is employed to transmit the video packets.

For eg: Consider an ad hoc architecture with 11 nodes is shown in Fig 3.1 as SR- Source Node, D- Destination Node

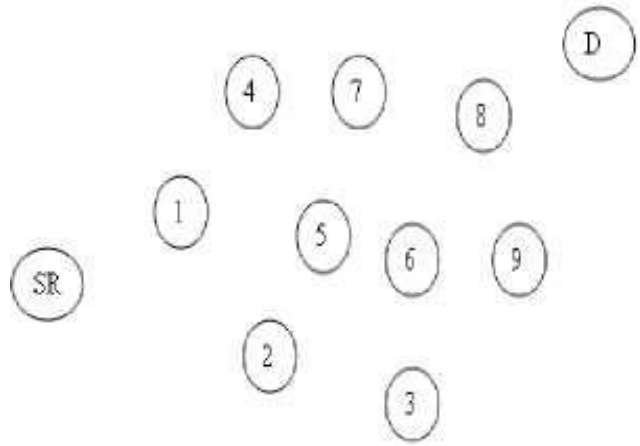

Fig 3.1 Basic WSN architecture

In the WSN the nodes undergo change in their positions the source should be continuously tracking their positions. By implementing the AODV protocol in the simulation scenario it transmits the first part of the video. For example the path is taken as S-1-4-17-7-8-D

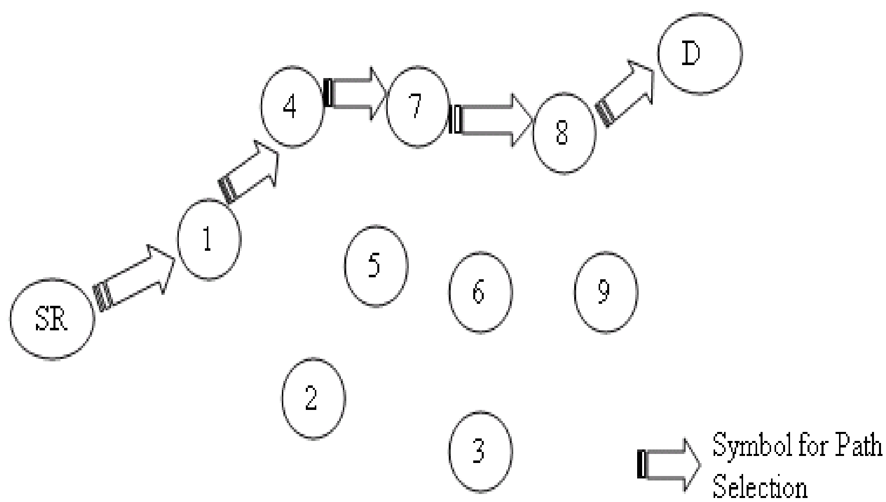

Fig 3.2 Data Transmission

In the existing method, the network is considered here as lightly loaded. Due to the mobility and poor channel condition, congestion and packet drop may occur. QoS parameters like Throughput, Delay, Delivery ratio and Efficiency and their performance will be discussed later.

CHAPTER 4
PROPOSED SYSTEM

4.1 MULTIPATH ROUTING

Multipath routing [6] is the routing technique of leveraging multiple alternative paths through a network, which can yield a variety of benefits such as fault tolerance, increased bandwidth, improved delay, increased throughput or improved security. The ability of creating multiple routes from a source to a destination is used to provide a backup route. When the primary route fails to deliver the packets in some way, the backup is used.

4.2 MULTICAST ROUTING

Multicast communication [7] is an efficient solution for group applications in the Internet. Multicast conserves the network bandwidth by constructing a spanning tree between sources and receivers. A single copy of the data is sent to all the receivers through the multicast tree. Applications such as videoconferencing, distant learning or network games use video, audio and data traffic.

4.3 MULTIPATH MULTICASTING

Multipath Multicasting is based on three aspects:

1. Multipath selection and establishment
2. Multipath route maintenance
3. Load distribution for distributing traffic among multiple paths

4.4 MULTIPATH SELECTION AND ESTABLISHMENT

In AODV, when a node broadcasts a RREQ message, it is often likely to receive more than one response message since any node in the multicast tree can respond to the message. If the source node receives one or more RREP messages in this time, it queries the multicast table and check if the route is activated to confirm which one is the first arrival.

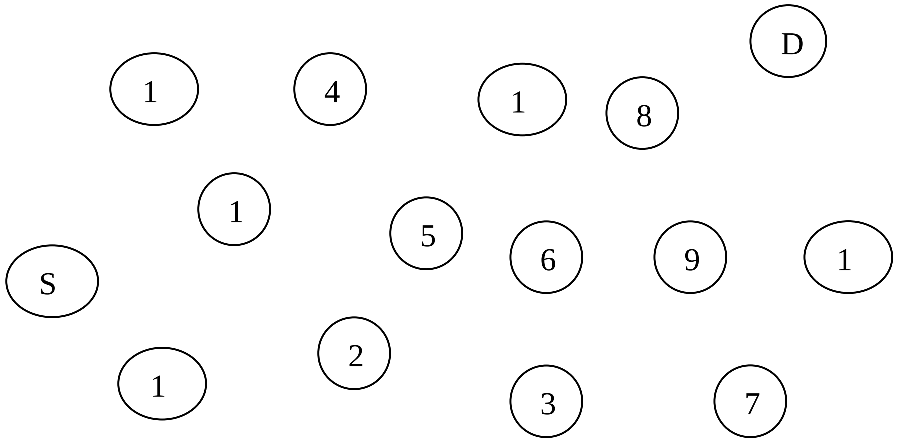

Figure 4.1 Basic ad hoc architecture

The source node unicasts a MACT (Multicast Activation) to the node which RREP is the first arrival for activating the route and sends packets through the path due to the first path has the shortest latency.

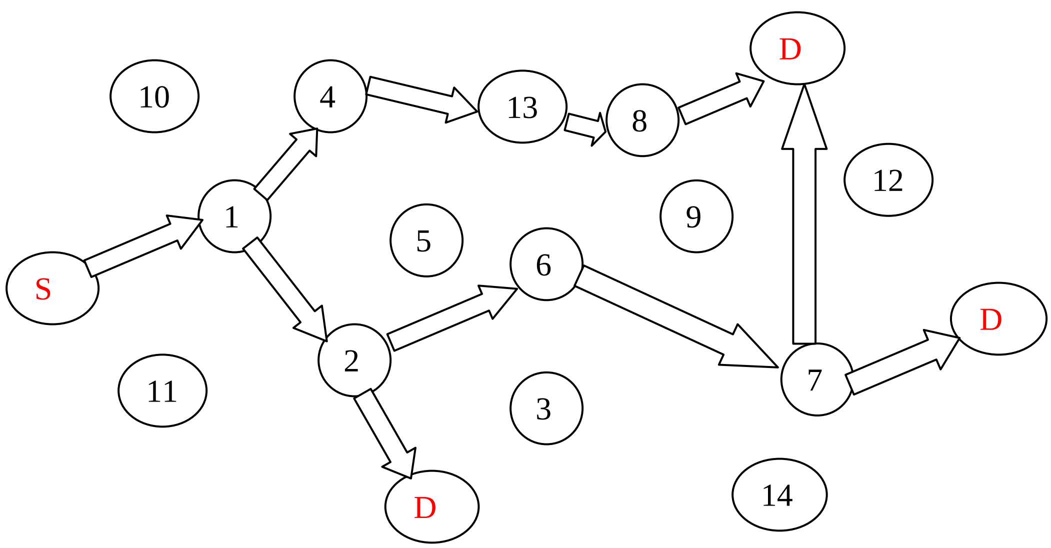

Figure 4.2 Multipath Multicasting

Multipath Route Maintenance

The intermediate nodes, which received MACT, activate the related entry in multicast table, and then forward the MACT to next hop until one group member receives MACT.

Multiple paths are selected to reduce resource consumption and improve calculation efficiency. The wireless link is easy to break because of nodes mobility or other reasons. When a node doesn't receive any message from the adjacent node or can't send any packet to the next hop, it thinks the link is broken. When the intermediate nodes in this path receive RERR, they delete the entry in the route table, and continue to forwarding RERR until the source node receives RERR message. When the source node receives the RERR, it deletes the related entry in the route table, searches backup route table [6] and checks whether both paths are invalid. If the two paths are broken at the same time, the source node broadcasts RREQ to initiate a new route discovery.

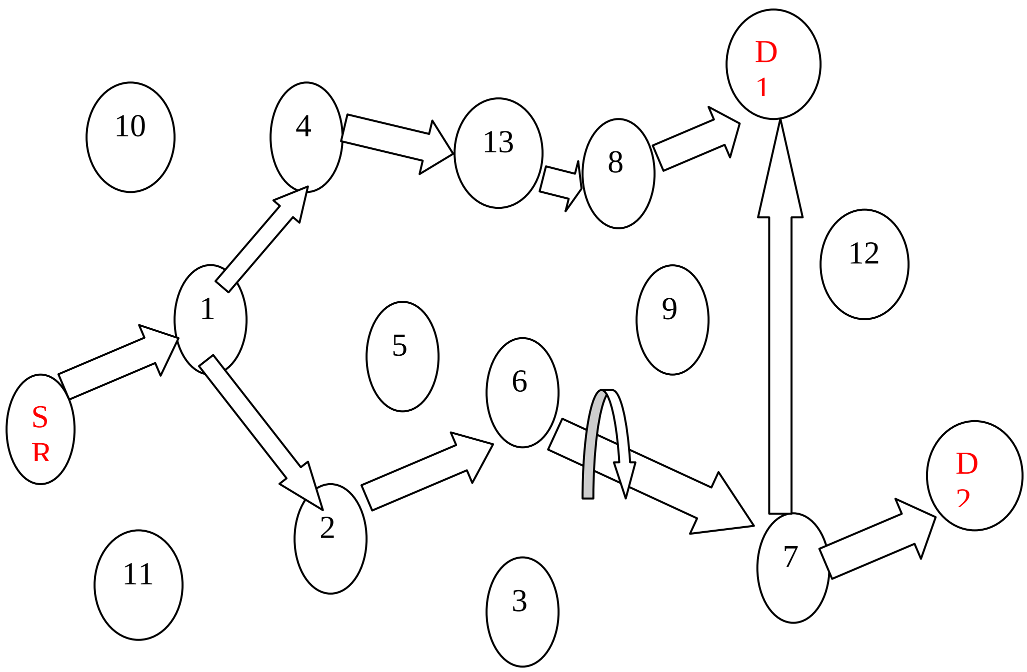

Figure 4.3 Link Failure

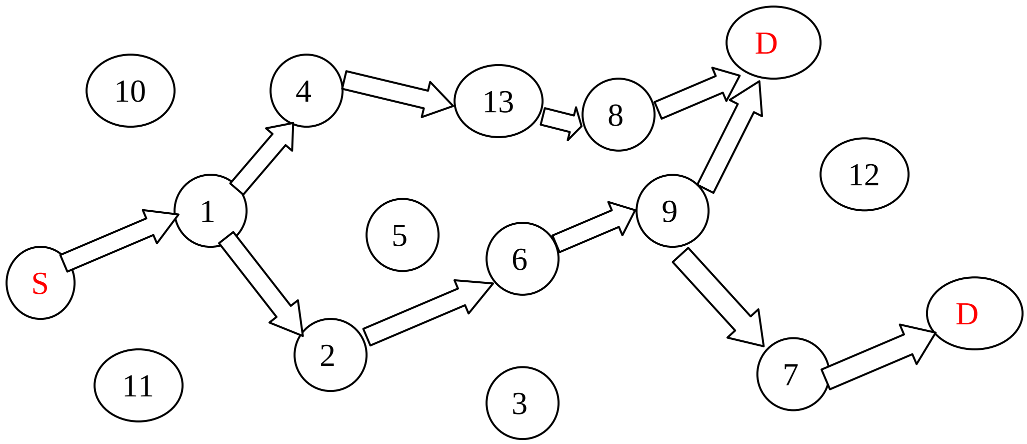

Figure 4.4 choosing an alternate path

Load Distribution

Once the path has been established, the source node starts to send packets [4] through multiple paths. It sends all packets through these paths in order to reduce latency caused by route discovery. This method can balance the network load and relieve the network congestion. In order to improve the QoS parameters like Throughput, Delay, Efficiency, Delivery ratio the “Efficient Resource allocated improve QoS AODV” algorithm has been designed.

EFFICIENT RESOURCE ALLOCATED IMPROVE QoS AODV

Intermediate nodes are unable to send the data packets due to heavy traffic and congestion. This will result in packet drops eventually resulting in degradation of packet delivery ratio. This situation will result to localized congestion [7]. If congestion happens at any point of time between source and destination nodes on primary route, concerned node warns its previous node about congestion.

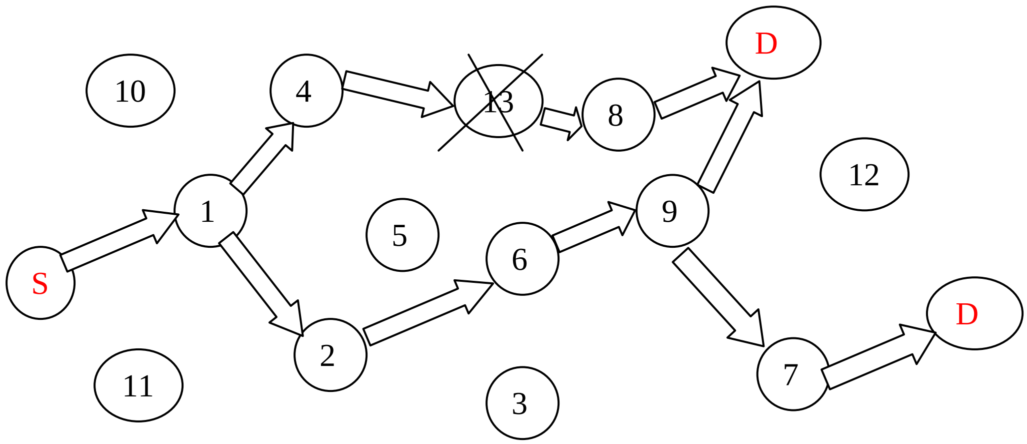

Figure 4.5 Congestion occurs at node 13

The previous node uses a non congested alternate route to the destination node. And if the path chosen are equally congested, then the data is transmitted through resource allocation method. The local route repair in AODV [18] exploits the by-pass route to the destination as soon as the route is received from the neighbor to the destination. These results to an improved performance in the packet delivery fraction normalized routing load and end to end delay. Each node preserves a local congestion level by calculating ratio which the current cache buffer queue length to the maximum length (called as congestion state: CS). The congestion state of nodes will be divided into three grades based upon queue length are no congestion, mild congestion, serious congestion.

IF (Queue- Average) < min

THEN no action is taken

IF (min > Queue- Average) > max

THEN with probability, the packet is dropped due to congestion

IF (Queue- Average) > min

THEN packet is dropped

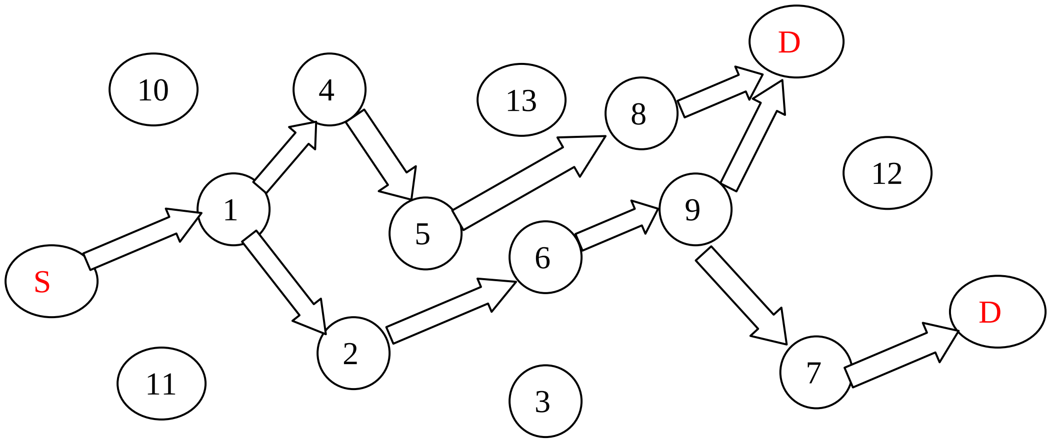

Figure 4.6 choosing a non congested path

Algorithm

- Source broadcast the RREQ packet to all the nodes in the network in order to discover the paths for data transmission.
- Once the source receives the RREP packet, wait for a certain amount of time, then from received paths choose those ones that have the least active neighbors and starts load balancing data transfer on these paths.
- The data packets are transmitted to different receivers via chosen paths.
- If congestion occurs in the specified path, the AODV protocol itself chooses the alternate path.
- If all the paths are congested, then based on resource allocation method the path is chosen.

In the proposed method, resource allocation method is implemented to increase the network throughput, delivery ratio, efficiency and to decrease the delay. These performances are analyzed by using the tool Network Simulator 2. The parameters that are configured in the network are Packet Size of 2000bytes, 100 nodes with the dimension of 1000*1000. The input given to the network is signal. To evaluate the performance, the Channel Type used here is Wireless channel. The Mobility of the node is 10 m/s. The delay for the proposed method is reduced to 3% and throughput is increased to 0.5% by varying the number of nodes in the network. Therefore the network efficiency is improved by 7% when compared to that of the existing case. With respect to mobility, the throughput of the proposed method is increased to 2% and the efficiency by 12%.

Besides, a cluster-based scheme is proposed, where sensors are statically divided into clusters, and each cluster consists of a single Cluster Head (CH) and a bunch of slave sensors. At every sampling instant, only one cluster of sensors is triggered to track the target. Resource consumption of the network is thus restricted to the activated cluster, where intra cluster communication is dramatically reduced so achieves optimization based sleep scheduling. Therefore, the cluster activation phase has a great importance not only in minimizing resource consumption but also in tracking accuracy. First, all the CHs need to measure the distances between the target and themselves at every sampling instant; then, a comparison among them is required to choose the nearest one. When a target enters the wireless sensor network, the CH that detects the target becomes active while other nodes are in sleep mode. Then the active CH selects three sensor nodes of its members for tracking in which one node is selected as Leader node. The selected nodes sense the target and current target location is calculated.

In this approach three sensor nodes are selected each time in which two nodes calculates its distance from the moving object and sends the data to the leader node. The localization of the moving object is done by leader node whereas in previous methods it's done by CH. Using prediction based clustering method energy consumed in the network will be reduced since the transmission power of the nodes is directly proportional to the distances. The three nodes selected for tracking are close to each other, thus the energy consumed for sending a data between the nodes is lower than sending a data from one of the selected nodes to its CH.

In this work, a system is developed in such a way that target tracking in WSN is done in efficient way using an energy efficient prediction- based clustering algorithm. Energy efficient prediction based Clustering algorithm, reduces the average energy consumed by sensor nodes and thereby increase the lifetime of the network. The tracking of the moving object is accurately done.

CHAPTER 5

SYSTEM SPECIFICATION

HARDWARE REQUIREMENTS

Processor	:	Intel Pentium IV
Processor Speed	:	1.4 GHz
Memory (RAM)	:	Default
Hard disk	:	Default
Monitor	:	Default
Input Device	:	Keyboard (104)

SOFTWARE REQUIREMENTS

Operating system	:	Fedora core 8 / Redhat 5 Linux
Software	:	NS2
Language	:	C (parsec), Java
Development Kit	:	JDK1.7
Platform	:	Independent

CHAPTER 6
NETWORK SIMULATOR

6.1 INTRODUCTION

A network simulator is a software program that imitates the working of a computer network. In simulators, the computer network is typically modeled with devices, traffic etc and the performance is analyzed. Typically, users can then customize the simulator to fulfill their specific analysis needs. Simulators typically come with support for the most popular protocols in the use today, such as Wireless LAN, Wi-Max, UDP, and TCP. A network simulator is a piece of software or hardware that predicts the behavior of a network, without an actual network being present. NS is an object oriented simulator, written in C++, with an OTcl interpreter as a frontend.

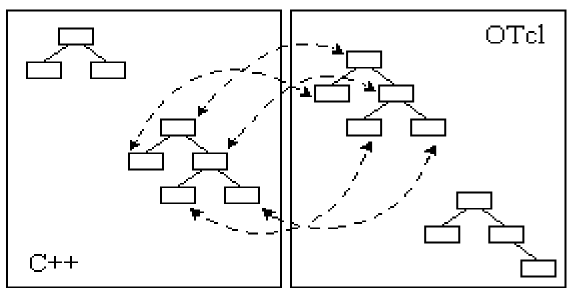

Figure 6.1 Network Simulator

The simulator supports a class hierarchy in C++ and a similar class hierarchy within the OTcl interpreter. The two hierarchies are closely related to each other; from the user's perspective, there is one-to-one correspondence between a class in the interpreted hierarchy and one in the compiled hierarchy. The root of this hierarchy is the class Tcl object. Users create a new simulator objects through the interpreter; these objects are instantiated within the hierarchy. The interpreted class hierarchy is automatically established through methods defined in the class Tcl object. There are other hierarchies in the C++ code and OTcl scripts; these other hierarchies are not mirrored in the manner of Tcl object.

Network simulators serve a variety of needs. Compared to the cost and time involved in setting up an entire test bed containing multiple networked computers, routers and data links, network simulators are relatively fast and inexpensive. They allow engineers to test scenarios that might be particularly difficult or expensive to emulate using real hardware- for instance, simulating the effects of sudden bursts in the traffic or a Dos attack on a network service. Networking simulators are particularly useful in allowing designers to test new networking protocols or changed to existing protocols in a controlled and reproducible environment. Network simulators simulate and then analyze the effect of various parameters on the network performance.

Typical network simulators encompasses a wide range of networking technologies and help the users to build complex networks from basic building blocks like variety of nodes and links. With the help of simulators one can design hierarchical networks using various types of nodes like computers, hubs, bridges, routers, optical cross-connects, multicast routers, mobile units, etc. various types of Wide Area Network (WAN) technologies like TCP, ATM, IP etc and Local Area Network (LAN) technologies like Ethernet, token rings etc, can all be simulated with the typical simulator and the user can test, analyze various routing etc. There are a wide variety of network simulators, ranging from the very simple to very complex. Minimally a network simulator must a user to represent a network topology, specifying the nodes of the network, the links between the nodes and the traffic between the nodes. More complicated systems may allow the user to specify everything about the protocols used to handle network traffic. Graphical applications allow users to easily visualize the working of their simulated environment. Text based applications may provide a less intuitive interface, but may permit more advanced forms of customization. Others, such as GTNets, are programming- oriented, providing a programming framework that the user then customizes to create an application that simulates the networking environment to be tested.

6.2 NETWORK SIMULATOR 2 (NS2):

NS2 is an open- source simulation tool that runs on Linux. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP over wired and wireless (local and satellite) networks. It has many advantages that make it useful tool, such as support for multiple protocols and the capability of graphically detailing network traffic. Additionally, NS2 supports several algorithms in routing and queuing. Queuing algorithms include fair queuing, deficit round-robin and FIFO. REAL is a network simulator originally intended for studying the dynamic behavior of flow and congestion control schemes in packet switched data network. NS2 is available on several platforms such as FreeBSD, Linux, Sim OS and Solaris. NS2 also builds and runs under Windows

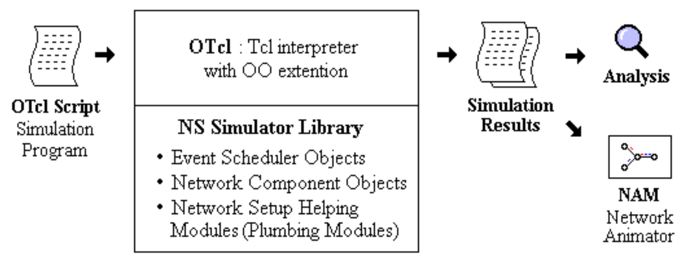

Figure 6.2 Simplified user's view of NS

Installing Ns

There are two ways to install NS: either independently by pieces or all at once (recommended one). Install all pieces at once using ns-allionexxx package. The following lines have to be changed in .bash _ profile file or in the corresponding file at our system:

export PATH=$PATH:..

export LD_ LIBRARY_ PATH=$LD_ LIBRARY_ PATH:…

export TCL_ LIBRARY=$TCL_ LIBRARY:…..

6.3 NODE BASICS

The basic primitive for creating a node is

set ns[new Simulator]

$ns node

The instance procedure node constructs a node out of simpler classifier objects. All nodes contain the following components:

1. An address or id_, monotonically increasing by 1 (from initial value o) across the simulation namespace as nodes are created
2. A list of neighbors (neighbor_)
3. A list of agents (agent_)
4. A node type identifier (node type_)

5. A routing module

Address and port number management

1)The procedure **$node id** returns the node number of the node. This number is automatically incremented and assigned to each node at creation by class Simulator method, $ns node.

2)The procedure **$node agent <port>**returns the handle of the agent at the specified port. If no agent at the specified port number is available, the procedure returns the null string.

3)The procedure **alloc_ port** returns the next available port number. Is uses an instance variable, np_, to track the next unallocated port number.The procedures, **add_ route** and add-routes, are used by unicast routing to add routes to populate the classifier.

Creating a Tcl scenario

To define trace files with the data that needs to be collected from the simulation, we have to create these files using the command open:

#open the trace file

set traceFile [open out.tr w]

$ns trace-all $traceFile

#open the Nam trace file

set namFile [openout.nam w]

$ns namtrace-all $namFile

#define the TCP agent

Set tcp [new Agent/TCP]

$ns attach-agent $n(0) $tcp

Set sink [new Agent/TCPSink]

$ns attach-agent $n(1) $sink

$ns connect $tcp $sink

Node Configuration

Node configuration essentially consists of defining the different node characteristics before creating them. They may consists of the type of addressing structure used in the simulation, defining the network components for mobile nodes, turning on or off the trace options at Agent/Router/MAC levels, selecting the type of adhoc routing protocol for wireless nodes or defining their energy model

6.4 SIMULATION PROCEDURE

Run the script by typing at the Konsole

Ns filename.tcl

On completion of the run, Trace output file “filename-out.tr” and nam output file “filename-out.nam” are created. Running filename-out.nam, the mobile nodes moving in the nam window can be seen. The active senders start informing the network about its presence and begin sending data according to the random progress method.

The finish procedure is given as

```
proc finish{} {
$ns flush-trace
close $r
close $nf
exec nam –r filename. nam &
exit 0
}
```

In the finish procedure, the trace file buffer is cleared and the graphs are generated in the terminal in a pipelined manner. $ns is used to close the trace field. Now the animator field is generated using command

exec nam filename.nam

To run the file **$ns run** command is used and the tcl script is executed.

To execute the graph **exec ns graph.tcl command is used.**

CHAPTER 7

EXPERIMENTAL RESULTS

7.1 Performance Analysis

Various parameters are calculated in order to evaluate the performance of total WSN. The graph is plotted for many parameters and comparative analysis is done. Here Network simulator 2 (ns2) is used for simulation. By using multiple mobile sink the improved WRP (I-WRP) shows better results than compared with existing WRP and CB with single mobile sink. The comparisons are made with parameters such as packet drop, end to end delay, throughput, packet delivery ratio, energy consumption.

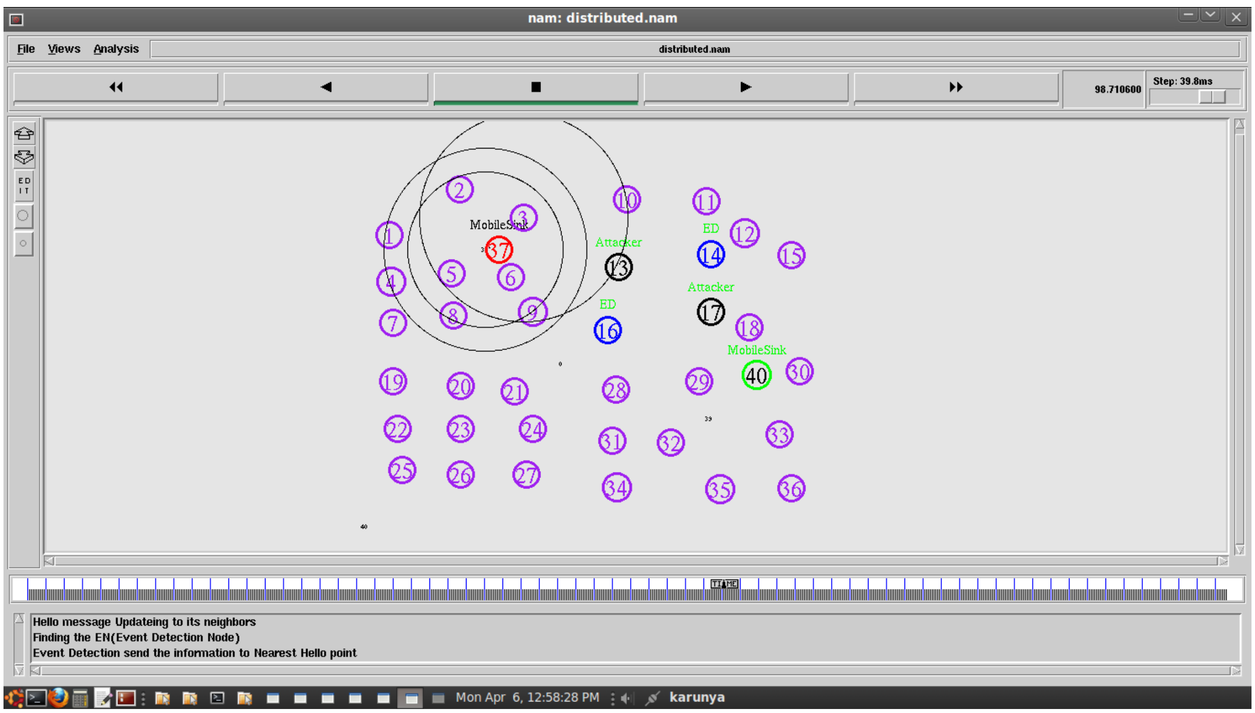

Figure 7.1 using two mobile sink

The above figure shows how the connectivity and establishment is made using two mobile sink. It moves randomly and collects the data. Since two mobile sink is used the consumption of energy is less and increases network lifetime.

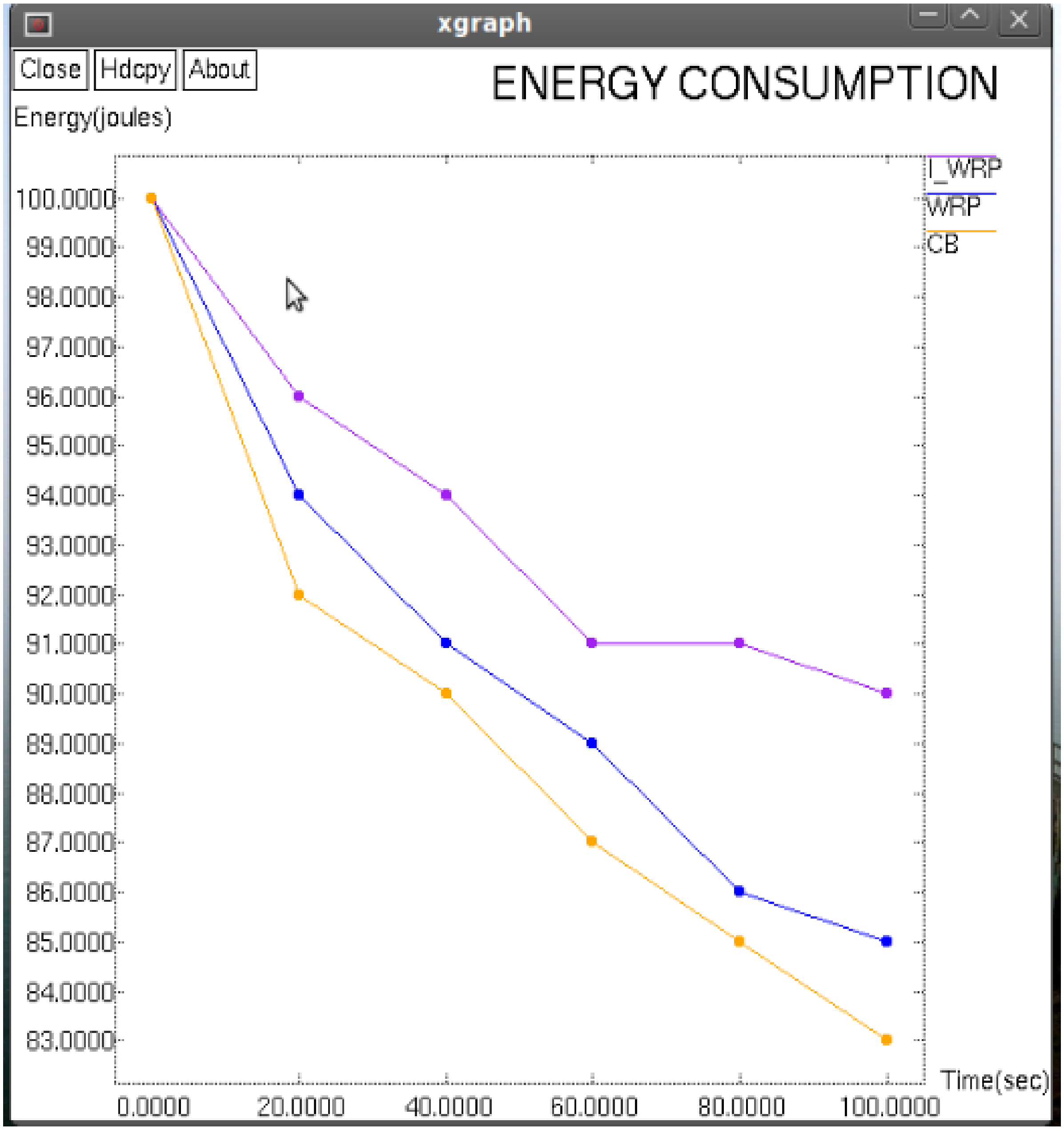

Figure 7.2 Energy consumption

From the above figure we studied the energy in joules with respect to time in seconds. Energy consumption is the amount of energy that is consumed in the network. It shows that energy consumption is less in IWRP than compared to CB and WRP.

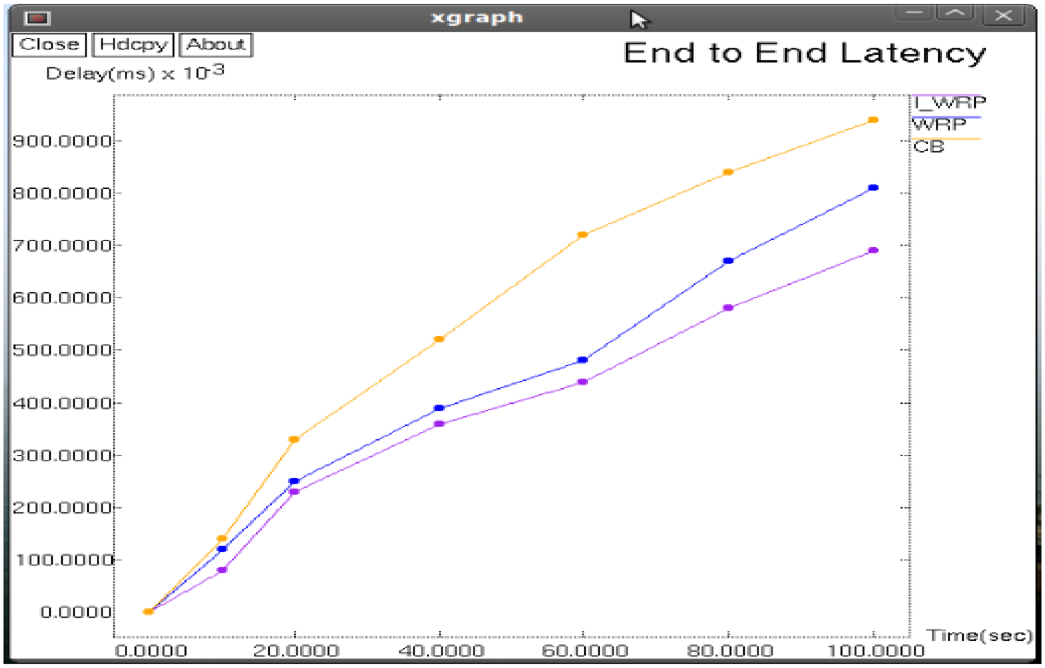

Fig 7.3 End to End Latency

From the above figure we studied the delay with respect to time in seconds. It shows that End to End Latency is less in IWRP than compared to CB and WRP.

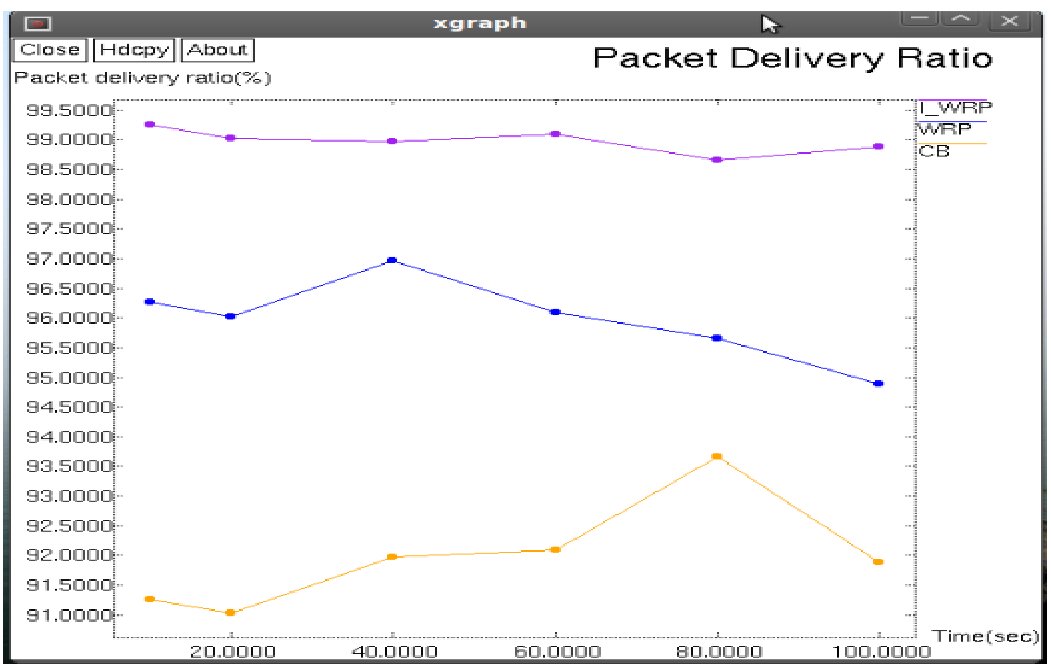

Fig 7.4 Packet Delivery Ratio

From the above figure we studied the packet delivery ratio with respect to time in seconds. The packet delivery ratio is the amount of packet that is received to destination. It shows that packet delivery ratio is high in IWRP than compared to CB and WRP.

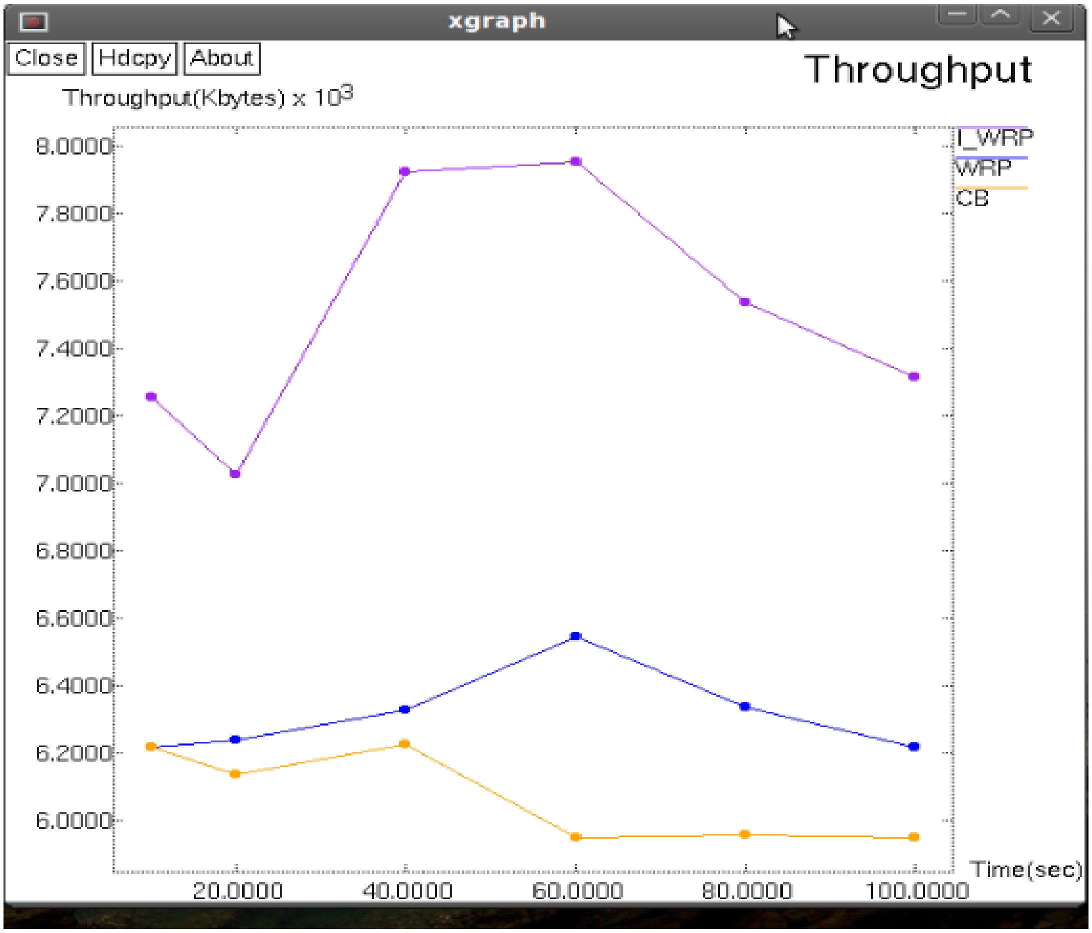

Fig 7.5 Throughput

From the above figure we studied the throughput with respect to time in seconds. The throughput is the rate of successful message delivery over a communication channel. It shows that throughput is high in IWRP than compared to CB and WRP.

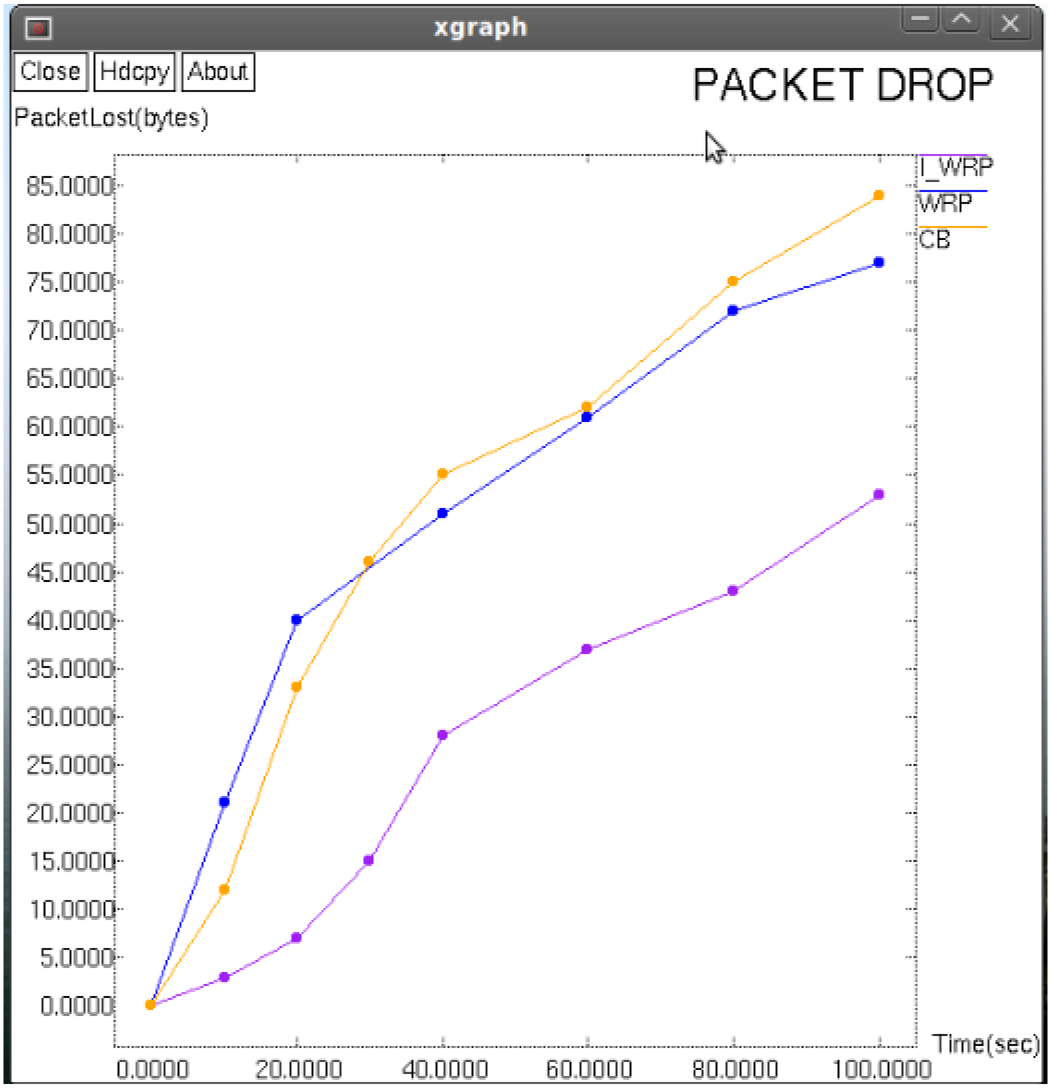

Figure 7.6 Packet drop

From the above figure we studied the packet loss with respect to time in seconds. A packet drop attack or blackhole attack is a type of denial-of-service attack in which a router that is supposed to relay packets instead discards them. This usually occurs from a router becoming compromised from a number of different causes. It shows that packet drop is less in IWRP than compared to CB and WRP.

CHAPTER 8

CONCLUSION

As a result here WRP, which is an algorithm for controlling the movement of a mobile sink in a WSN. WRP selects the set of RPs such that the energy expenditure of sensor nodes is minimized and uniform to prevent the formation of energy holes while ensuring sensed data are collected on time. Thus here by using multiple mobile sink simulation results must show that Improved WRP (IWRP) reduces energy consumption by better than WRP and CB which is having only single mobile sink. Thus the Improved WRP with multiple mobile sink increases network lifetime.

CHAPTER 9
REFERENCES

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless Sensor Networks: A Survey,” Computer Networks, vol. 38, no. 4, pp. 393-422, 2002.

[2] Q. Cao, T. Yan, J. Stankovic, and T. Abdelzaher, “Analysis of Target Detection Performance for Wireless Sensor Networks,” Proc. Int’l Conf. Distributed Computing in Sensor Systems (DCOSS), pp. 276-292, 2005.

[3] G. Wittenburg, N. Dziengel, C. Wartenburger, and J. Schiller, “A System for Distributed Event Detection in Wireless Sensor Networks,” Proc. Ninth ACM/IEEE Int’l Conf. Information Processing in Sensor Networks (IPSN ’10), pp. 94-104, 2010.

[4] T. He, P. Vicaire, T. Yan, L. Luo, L. Gu, G. Zhou, R. Stoleru, Q. Cao, J.A. Stankovic, and T. Abdelzaher, “Achieving Real-Time Target Tracking Using Wireless Sensor Networks,” Proc. 12th IEEE Real-Time and Embedded Technology and Applications Symp. (RTAS ’06), pp. 37-48, 2006.

[5] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, “Towards Optimal Sleep Scheduling in Sensor Networks for Rare Event Detection,” Proc. Fourth Int’l Symp. Information Processing in Sensor Networks, p. 4, 2005.

[6] C. Gui and P. Mohapatra, “Power Conservation and Quality of Surveillance in Target Tracking Sensor Networks,” Proc. 10th Ann. Int’l Conf. Mobile Computing and Networking, pp. 129-143, 2004.

[7] Y. Gu and T. He, “Data Forwarding in Extremely Low Duty-Cycle Sensor Networks with Unreliable Communication Links,” Proc. Fifth Int’l Conf. Embedded Networked Sensor Systems (SenSys ’07), pp. 321-334, 2007.

[8] Y. Gu and T. He, “Data Forwarding in Extremely Low Duty-Cycle Sensor Networks with Unreliable Communication Links,” Proc. Fifth Int’l Conf. Embedded Networked Sensor Systems (SenSys ’07), pp. 321-334, 2007.

[9] Y. Wu, S. Fahmy, and N. Shroff, “Energy Efficient Sleep/Wake Scheduling for Multi-Hop Sensor Networks: Non-Convexity and Approximation Algorithm,” Proc. IEEE INFOCOM, pp. 1568-1576, May 2007.

[10] X. Wang, J.-J. Ma, S. Wang, and D.-W. Bi, "Cluster-Based Dynamic Energy Management for Collaborative Target Tracking in Wireless Sensor Networks," Sensors, vol. 7, pp. 1193-1215, 2007

[11] J. Fuemmeler and V. Veeravalli, "Smart Sleeping Policies for Energy Efficient Tracking in Sensor Networks," IEEE Trans. Signal Processing, vol. 56, no. 5, pp. 2091-2101, May 2008.

[12] J. Jeong, T. Hwang, T. He, and D. Du, "MCTA: Target Tracking Algorithm Based on Minimal Contour in Wireless Sensor Networks," Proc. IEEE INFOCOM, pp. 2371-2375, 2007.

[13] X. Wang, J.-J. Ma, S. Wang, and D.-W. Bi, "Prediction-Based Dynamic Energy Management in Wireless Sensor Networks," Sensors, vol. 7, no. 3, pp. 251-266, 2007.

[14] Y. Xu, J. Winter, and W.-C. Lee, "Prediction-Based Strategies for Energy Saving in Object Tracking Sensor Networks," Proc. IEEE Int'l Conf. Mobile Data Management, pp. 346-357, 2004.

[15] I. Krontiris, T. Giannetsos, and T. Dimitriou, (2008) "Lidea: a distributed lightweight intrusion detection architecture for sensor networks," in SecureComm.

[16] G. Xing, T. Wang, W. Jia, and M. Li, "Rendezvous design algorithms for wireless sensor networks with a mobile base station," in *Proc. 9th ACM Int. Symp. Mobile ad hoc Netw. Comput., Hong Kong, China, May 2008,*pp. 231–240.

[17] G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous planning in wireless sensor networks with mobile elements," *IEEE Trans. Mobile Comput.,*vol. 7, no. 12, pp. 1430–1443, Dec. 2008.

[18] K. Almi'ani, A. Viglas, and L. Libman, "Energy-efficient data gathering with tour length-constrained mobile elements in wireless sensor networks," in *Proc. 35th IEEE Conf. LCN, Denver, CO, USA,* Oct. 2010, pp. 582–589.

PUBLICATIONS

- Presented a paper entitled **"WEIGHTED RENDEZVOUS PLANNING FOR ENERGY EFFICIENT MOBILE-SINK PATH SELECTION IN WIRELESS SENSOR NETWORKS"** in 2nd International conference on Electronics and Communication Systems on 26th and 27th February, 2015 organized by Department of Electronics and Telecommunication Engineering.