DESIGN OF COMMUNICATION STRATEGY FOR WIRELESS SENSORS IN NON-DETERMINISTIC ENVIRONMENT USING MOBILE AGENT

THESIS

submitted in fulfillment of the requirement of the degree of

DOCTOR OF PHILOSHOPHY

to

YMCA UNIVERSITY OF SCIENCE & TECHNOLOGY

by

PREETI SETHI

Registration No: YMCAUST/Ph16/2010

Under the Supervision of

Dr. NARESH CHAUHAN
PROFESSOR

Dr. DIMPLE JUNEJA GUPTA
PROFESSOR

Department of Computer Engineering
Faculty of Engineering and Technology
YMCA University of Science &Technology
Sector-6, Mathura Road, Faridabad, Haryana, INDIA

DECEMBER, 2015

Dedicated

to

My daughters Navya & Tiana

DECLARATION

I hereby declare that this thesis entitled "**DESIGN OF COMMUNICATION STRATEGY FOR WIRELESS SENSORS IN NON-DETERMINISTIC ENVIRONMENT USING MOBILE AGENT"** being submitted in fulfillment of requirement for the award of Degree of Doctor of Philosophy in the Department of Computer Engineering under Faculty of Engineering and Technology of YMCA University of Science and Technology, Faridabad, during the academic year May 2011 to December 2015,is a bonafide record of my original work carried out under the guidance and supervision of **Dr. NARESH CHAUHAN, PROFESSOR & CHAIRMAN, DEPARTMENT OF COMPUTER ENGINEERING, YMCA UNIVERSITY OF SCIENCE AND TECHNOLOGY, FARIDABAD** and **Dr. DIMPLE JUNEJA GUPTA, PROFESSOR AND DIRECTOR, DRONACHARYA INSTITUTE OF MANAGEMENT & TECHNOLOGY, KURUKSHETRA** and has not been presented elsewhere.

I further declare that the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

(PREETI SETHI)

Registration No.YMCAUST/Ph16/2010

CERTIFICATE

This is to certify that this thesis entitled **"DESIGN OF COMMUNICATION STRATEGY FOR WIRELESS SENSORS IN NON-DETERMINISTIC ENVIRONMENT USING MOBILE AGENT"** by **PREETI SETHI** submitted in fulfillment of the requirements for the award of Degree of Doctor of Philosophy in Department of Computer Engineering, under Faculty of Engineering and Technology of YMCA University of Science and Technology, Faridabad, during the academic year May 2011 to December 2015, is a bonafide record of work carried out under our guidance and supervision.

We further declare that to the best of our knowledge, the thesis does not contain any part of any work which has been submitted for the award of any degree either in this university or in any other university.

DR. NARESH CHAUHAN
Professor & Chairman
Department of Computer Engineering
Faculty of Engineering and Technology
YMCA University of Science & Technology Faridabad

DR. DIMPLE JUNEJA GUPTA
Professor & Director
Dronacharya Institute of Management & Technology
Kurukshetra

Dated:

ACKNOWLEDGEMENTS

I am thankful to God for making things possible at the right time always.

With the debt of gratitude which cannot be rightly explained in words, I would like to thank my *Gurus* **Dr. Naresh Chauhan,** Professor and Chairman, Department of Computer Engineering, YMCA University of Science and Technology, Faridabad and **Dr. Dimple Juneja Gupta** , Professor and Director, Dronacharya Institute of Management and Technology, Kurukshetra., who made my dream of persuing and completing the research work true. Both of them provided me a perfect vent to learn and explore innovative things and were always there to guide me for improvisation by giving their invaluable suggestions and constructive criticism. Despite being a philoshoper, Dr. Gupta guided me as a friend and helped me to meet the technical glitches related to the work. At times when was I stumbled upon big obstacles, both my mentors encouraged me to look further and keep sailing through tough times.

I am indebted to **Dr. Atul Mishra** and **Dr. Komal Kumar Bhatia**, Associate Professors, Department of Computer Engineering, YMCAUST for their support as Ph.D coordinators. I am also thankful to all my students, faculty and staff members of the department who helped me directly or indirectly in completing my work.

A word of special thanks is due to my family for their help and support. I express thanks to my in-laws **Mr.A.K.Sethi** and **Mrs. Kusum Sethi** for providing me a peaceful environment at home. I would also like to thank my parents **Mr. Suresh Kumar Arora** and **Mrs. Harsh Lata Arora** and sister **Ms. Ashima** for being a constant source of motivation.

Words fail to express my heartfelt thanks to my better half **Mr. Vikas Sethi** for his technological assistance and unparalled availability at all times during the course of my work. Without his moral support this thesis would not have been a reality. My special thanks to my daughters **Navya and Tiana** for tolerating me whenever I had been annoyed or angry while working. Their endless love, priceless and perpetual support made all this possible. They have given me more than I can say.

Thank you all**!** **(Preeti Sethi)**

ABSTRACT

Wireless Sensor Network(WSN) is an assimilation of numerous tiny sensing units' called sensors which communicate with each other wirelessly. These low cost devices have the ability to monitor physical, chemical or biological properties and are thus used in huge plethora of applications like urban monitoring, health care, habitat monitoring, surveillance etc. Despite this fact, petite work has been in the field of non-deterministic environment of WSN. As the name suggests, non-deterministic environment (for example, under-sea, dense jungles etc.) are the ones which are not easily reachable and in which sensors are deployed randomly. Unlike its counterpart, the sensors in such an environment once deployed remain unattended throughout their lifetime. Saving the energy of such untethered units is thus very crucial as otherwise the failure of sensor(s) may lead to the failure of the network

The thrust of this thesis is to design an efficient communication strategy for non-deterministic WSN using mobile agents. As the name suggests, a mobile agent is a self-contained piece of software that can migrate and execute on different machines in a dynamic networked environment, and that senses and (re) acts autonomously and proactively in this environment to realize a set of goals or tasks. Use of mobile agents in place of the conventional client-server paradigm provides an energy efficient means of communication in such a resource constrained network. The work is targeted to amalgamate the characteristics of mobile agents with WSN with an aim to achieve better results.

The multifold contributions of this thesis include: 1) ***M**obility **C**ontrolled **C**ommunication of randomly deployed sensors* 2) *Energy aware **C**lustering and itinerary determination* 3) *Efficient information processing at the nodes level using **F**iltering* 4) ***F**usion of the filtered data.* The work is being carried out in 4 phases and is collectively termed as MC3F2.

The work has initially put forward a solution to the major challenge of providing an energy efficient means for communication among the randomly deployed sensors in the non-deterministic area.It has limited the mobility of sensor nodes and made them

function both static as well as mobile sensor nodes by making the pause time between the sensors random instead of constant. While simulating the same, it was observed that elongated random pause time made the nodes almost static while short pause times saved their mobile nature. Further, the task of dynamic mobility is achieved with the help of mobile agents embedded within the sensor nodes. The nodes are henceforth termed as Intelligent Sensing Units (ISU).

The thesis next provides an intelligent approach for clustering the sensors into unique clusters with the help of mobile agents. It has proposed an event driven approach for clustering which elects cluster heads amongst the sensors on the basis of residual energy of sensors and the reliability values of the agents. Thereafter it employs a novel approach to determine the itinerary of the mobile agent which is traversing within the cluster. The implementation of the same offered competitive results when compared with other clustering mechanisms in the allied fields.

The third phase of filtering is motivated by the fact that lot of raw data is being sensed with proportionate amount of noise resulting into very less useful information. Out of the various filtering algorithms available, Extended Kalman Filter(EKF) is being chosen , primarily because it is able to minimize the variance of estimation error i.e. filters noise from the actual signal more accurately. Simulation results show that considerable noise is being filtered and ample energy is saved by using the proposed approach.

During the course of this research, the need of in-network processing was identified. In view of this context, the thesis presents a novel approach for fusion which forms the last and the fourth phase of the work. In this phase the plain data is first encrypted and then fused at each ISU being visited by the mobile agent.

In summary, the research work has given an *agent based* solution for communication in non-deterministic WSN. In fact, by incorporating mobile agents within the sensors in the areas like dense jungles, underwater etc, the work attempts to introduce intelligence & decision-making capability in every sensor so that they become competent enough to deliver meaningful information instead of raw data to the base station. This information can then be used as per the application requirements.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Description
BS	Base Station
CH	Cluster Head
CSMA	Carrier Sense Multiple Access
CSWSN	Client Server Based WSN
DAI	Distributed Artificial Intelligence
DARPA	Defense Advanced Research Project Agency
DC	Data Collectors
DD	Directed Diffusion
DSN	Distributed Sensor Network
EEGCF	Energy Efficient Global Closest First
EKF	Extended Kalman Filter
FFSS	Forest Fire Surveillance System
GAF	Geograpahic Adaptive Fidelity
GEAR	Geographic Energy Aware Routing
GPS	Geographical Positioning System
IP	Internet Protocol
KF	Kalman Filter
LEACH	Low Energy Adaptive Cluster Hierarchy
MA	Mobile Agent
MAC	Medium Access Control
MANET	Mobile Adhoc Network
MAS	Multi Agent System
MAWSN	Mobile Agent Based Wireless Sensor Network
MCFA	Minimum Cost Forwarding Algorithm
MC3F2	Mobility Controlled Communication, Clustering, Filtering and Fusion
MIP	Multi Agent Itinerary Planning
PEGASIS	Power Efficient Gathering In Sensor Informative Systems
REAR	Reliable Energy Aware Routing

RPC	Remote Procedure Call
RR	Rumour Routing
RRP	Robust Routing Protocol
SAR	Sequential Access Routing
SIP	Single Agent Itinerary Planning
SOSVS	Sound Surveillance System
SPIN	Sensor Protocol for Information via Negotiation
TDMA	Time Division Multiple Access
TEEN	Threshold Energy Efficient Network
WBA	Wireless Broadcast Advantage
WMSNS	Wireless Medical Sensor Networks
WSN	Wireless Sensor Network

Chapter I

INTRODUCTION

1.1 WIRELESS SENSOR NETWORKS

Wireless Sensor Networks (WSNs) are special class of adhoc networks [54,56,61] which has helped humans to extract information even from the insurmountable areas of the globe. These networks are equipped with small sensing devices called sensors which sense the information from the environment and transmit the same to the sink module depending upon the application. This unique combination of software and hardware together has revolutionized the telecommunication industry in the last 2 decades.

WSNs are deployed in large groups and these deployed nodes collect, process and cooperatively pass this collected information to a central location. Because of these inimitable features of nodes, WSNs are finding applications in variety of domains such as traffic monitoring, target tracking, observing environmental changes in real time and so on.

The potential of WSN was realised by researchers and academicians decades ago and as a result several protocols leading to substantial improvements in terms of energy, routing and lifetime of WSN have been exclusively developed for this integrated technology. However, despite the above listed developments, literature indicates that energy supply and communication bandwidth limit the life of sensor nodes. Therefore the need of strategies overcoming these limitations resulting into improvements of the lifetime of the network and efficient use of limited bandwidth are highly apparent.

1.2 SOFTWARE AGENTS

The term agent comes from greek '*agein*', which means to drive or to lead [40]. In the realm of computer science, it is defined as *"an intelligent software unit capable of* *transferring code and data to the processing end"* [74]. It is placed within an environment and is able to recognize the environment through sensors and react upon it with the help of effectors. To define them more precisely, *"software agents are intelligent software units that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in doing so, employ some knowledge or representation of the user's goals or desires"* [104].

Infact, the fundamental characteristics of software agents which includes both the basic features of autonomy, reactivity, pro activity and advanced , human-like features like beliefs, desires, intentions and commitments have fascinated researchers to explore and integrate these units into WSN.

A special class of these intelligent software units, called ***mobile agents*** [75,139] are able to migrate from one machine to another under its own control and can suspend execution any time. They are capable of executing on heterogeneous machines situated at different locations, sense and take decisions of their own to achieve the delegated targets. These mobile entities are a promising candidate for communication in WSNs. They are a natural extension to remote procedure call (RPC) [142] approach used for communication in WSN and offers various advantages like asynchronous interaction, robustness, fault tolerance, dynamic adaptation and efficiency to name a few.

1.3 MOTIVATION AND GOAL

WSNs are finding applications in both deterministic and non-deterministic environments [54] where, deterministic environment is the one which is reachable and in which the sensors are deployed in a pre-defined manner. The data and/or information in such cases are routed through pre-defined paths. On contrary, non-deterministic environments are not easily reachable and the sensors are usually deployed randomly in such environments. Though extensive research has been done in the field of deterministic WSN, the researchers have been silent in the case of non-deterministic environment. There are several challenges to be dealt with in this domain like improving the lifetime of network, energy efficient node deployment, connectivity, coverage, sink hole problem etc. [24,47,135]. In addition to the various challenges listed above, one of the major challenges is to find an energy efficient alternative to the conventional routing paradigm i.e. the client server approach because this model is tagged with several problems like excessive use of bandwidth, longer delay in transmission and most importantly the transfer of raw and redundant data to the sink. This disproportionate amount of sensory data if delivered can cause unpunctual delivery, immense energy consumption and load maladjustment among sensors. The problem gets alleviated when the sensors are deployed in non-deterministic environments where once deployed, sensors remain unattended during their whole lifetime. The above stated limitations of client server computing model applied in WSN thus demand for the inevitability of exploring and designing intelligent frameworks and hence the motivation.

The work aimed to design an efficient communication strategy for improving the lifetime of the network. In order to achieve this objective, following goals have been identified which formulated the base of research work:

- ***To design a framework for communication in event driven applications of non-deterministic WSN.***

- ***To perform information processing on the sensed data at the source nodes.***

- ***Evaluation & comparison of proposed framework with its agent based counterparts.***

1.4 DESIGN CHALLENGES

A grave study of the existing propositions in the realm of WSN reveals that the following issues need to be addressed in order to meet the above stated goals:

- **Optimal Clustering**: Though clustering is used in hierarchical WSN, most of the approaches are spatial based approaches which can result in messier and longer routes.

Solution*: In order to perform clustering in an efficient manner, a mobile agent based event driven clustering approach has been proposed which uses reliability of participating agents and the residual energy of the motes to make clusters.*

- **Efficient Communication of Randomly Deployed Sensor Nodes:** In a non-deterministic WSN where nodes are randomly deployed, mobility of the nodes can lead to dynamically changing links and unpredictable random topology. A need of mobility controlled communication is thus apparent.

  Solution: *Though mobility of sensing units (regular nodes or sink) leads to above mentioned problems, it is inevitable for certain applications. The work thus makes use of mobile agents for moving within the network, thereby restricting the mobility of sensors.*

- **Filtering at Source Node:** The data being sensed at the source node is usually corrupted with noise. Transmission of such raw data to the sink causes wastage of bandwidth.

  Solution: *The work makes use of Extended Kalman Filter (EKF) for filtering the data at the place where it is sensed so that only relevant information is communicated over the network.*

- **Redundant Data**: As sensors are deployed randomly, they are spatially correlated and the probability of sensing and transmitting the redundant data is very high.

  Solution*: The work has proposed an agent based fusion approach for in-network processing. This approach ensures that only significant information is being given to the sink.*

- **Security of Data:** Most of the data which is transmitted through the wireless channel is unencrypted and thus prone to channel attacks. There is a strong need of adding a layer of security to data transmission.

  ***Solution**: The work has proposed a security mechanism in which it the transmitted data is encrypted before transmission and decrypted for aggregation/fusion.*

1.5 ORGANIZATION OF THESIS

The thesis is principally carved up into six chapters as listed below:

Chapter 2 provides the details about the background study that was carried out to pursue this research work. It begins by presenting a detailed study of WSN ranging from its origin to real time applications and routing protocols. This chapter later throws light on the details of exploring the feasibility and deploying mobile agents in the area under consideration i.e. WSN.

Chapter 3 provides an insight into the literature review which motivated this research work. The very nascent idea of associating agents with WSN has emerged because of a thorough study of the available literature which indicated that research should be carried forward in four different phases i.e. mobility controlled communication, clustering, filtering and fusion. This chapter provides the backdrop of existing works pertaining the mentioned phases and further explores the possibility of improvements.

Chapter 4 furnishes a four phased novel approach which is presented in the light of drawbacks in the existing work. This chapter discusses the first two phases of the proposed approach. It begins by discussing the improved energy efficient approach for the randomly deployed sensors which considers random pause time of nodes at a given location. It further discusses the implementation and analysis done to compare the proposition. In the second phase, an agent based clustering approach for clustering the randomly deployed sensors is given which forms the second phase of the work. This phase also presents a novel itinerary determination approach of the mobile agent within the cluster. Phases 3 and 4 of the proposed work are being described in depth in the next chapter.

Chapter 5 initially presents a unique application of Extended Kalman Filter (EKF) for filtering sensitive information available with the sensors. The motivation for carrying out this work is imprecise sensed data owing to real time fluctuations in environment and also the need for energy efficient computations in addition to efficient communication. After filtering the data at each of the sensor node individually, it is being encrypted and fused at the intra cluster level and henceforth transmitted to the clusters above in the hierarchy for onwards routing. This accounts for the fourth and last phase of the proposed work. The chapter concludes by presenting a case study of a non-deterministic application and evaluation of the proposed multi agent framework with its counterparts.

Chapter 6 concludes the outcome of the work. It summarises the major achievements of the research work and elucidates the scope for future work in this domain.

Chapter II

WIRELESS SENSOR NETWORKS AND SOFTWARE AGENTS: A PREFACE

2.1 INTRODUCTION

The promising field of WSNs amalgamates sensing, computation, and communication into a device more commonly known as mote. Numerous such motes are connected to form a marine that extends the reach of cyberspace out into the real world. This revolutionary technology finds its way in the huge plethora of applications like disaster relief, habitat monitoring, health care, home networks, detecting chemical, biological, radiological, nuclear, and explosive materials, just to name a few [30,37,98,128,141,143].

Software agents on the other hand are proactive intelligent objects capable of performing the assigned task. Mobile agents are unique subset of software agents that have the capability to migrate from machine to machine in a heterogeneous network. They can defer their execution at an arbitrary point, migrate themselves to a different machine and may resume execution at a new location. A collection of such autonomous entities is being referred to as Multi Agent System (MAS) [26,74]. Available literature reflects that MAS is growing exponentially and is being applied in personnel management, e-commerce, search engines, electronic gadgets, manufacturing and production processes etc. [41, 43, 104].

Owing to the huge number of applications that software agents and distinctively mobile agents cater, it is feasible to exploit and use the power of mobile agents into these self configuring, self healing networks. This thesis contributes towards integrating mobile agents into WSNs so as to increase their lifetime and to provide an intelligent substitute to the conventional client server paradigm for routing the data from source to destination.

The upcoming section presents a meticulous study of wireless sensor networks and its applications in various domains. A portrayal of software agents is provided in the subsequent sections.

2.2 WIRELESS SENSOR NETWORKS: THE BACKGROUND

WSN is a special class of adhoc networks which has boomed up as a result of current advances in networking, semiconductor, and material science technologies. The heart of these self deployed networks is a dedicated diminutive entity known as sensor.

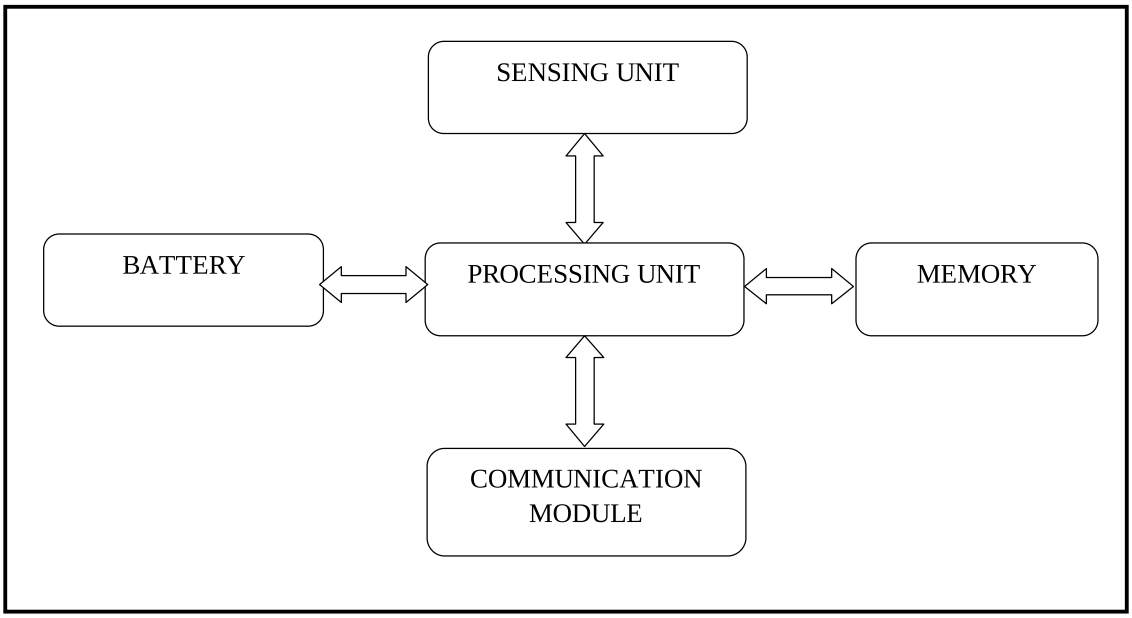

Figure 2.1: Components of a Sensor Node

As shown in Figure 2.1, the sensing unit is capable of monitoring various kinds of data like acoustic, visual, seismic and temperature data. The communication module is a kind of radio unit capable of short range communication (up to tens of meters), while the processing unit contains small memory and a processor with limited size and processing speed. A sensor node functions using a non-rechargeable battery contributing to one of the major drawbacks of sensors [56, 64]. In order to address the problem of inability of recharging, developers are working towards a mass production of nodes, which will significantly lower the per device cost, and to deploy them liberally as disposable devices.

Before further discussing the architecture and applications of WSNs, a short description related to its evolution is given in the next section

2.2.1 Origin of WSN

As stated above, WSN is an association of compact micro sensors with wireless communication capabilities. Like many advance technologies, WSN owe its root in military and heavy industrial applications. The first wireless network that is inline with the latest WSN is the Sound Surveillance System (SOSUS) developed on submerged acoustic sensors. Sensors in SOSUS were distributed in the Atlantic and Pacific oceans.

Stimulated by the developments pertaining to Internet in 1960s and 1970s to develop the hardware for today's Internet, Defence Advanced Research Projects Agency (DARPA) initiated the Network (DSN) program in 1980[21]. The motive was to explore the design challenges related to WSN. With the birth of DSN and its penetration into education through Carnegie Mellon University and the Massachusetts Institute of Technology, WSN technology could find its base in household, education and civilian scientific research.

Very soon, public and private communities started deploying sensors to monitor air quality, detect forest fire, forecast weather, prevent natural disaster etc. The sensors however at that time were bulky, expensive and made use of proprietary protocols. The use of such WSNs thus weighed down the industry which used it. This disproportionate relation of high cost with low volume of sensors declined their pervasive use.

Realising the potential of the network, industry and academia joined hands to solve the engineering challenges associated with sensors and lead to the production of modern sensors: low cost miniature size sensors, having simplified development and maintenance tasks.

2.2.2 Communication Architecture of WSN

Communication between the nodes of WSN is made possible when number of motes are deployed covering a given geographical area. Larger is the number of sensors covering the geographical region, higher would be the accuracy. Figure 2.2 shows a schematic diagram of the communication architecture of WSN. Here, each sensor senses the data (usually sensors are application specific) related to the event. These dispersed sensor nodes collect and transmit data to base station routed through the number of hops. A base station may be a fixed or mobile node capable of connecting this network to an existing communications infrastructure offering the services to the end user [30, 56, 64].

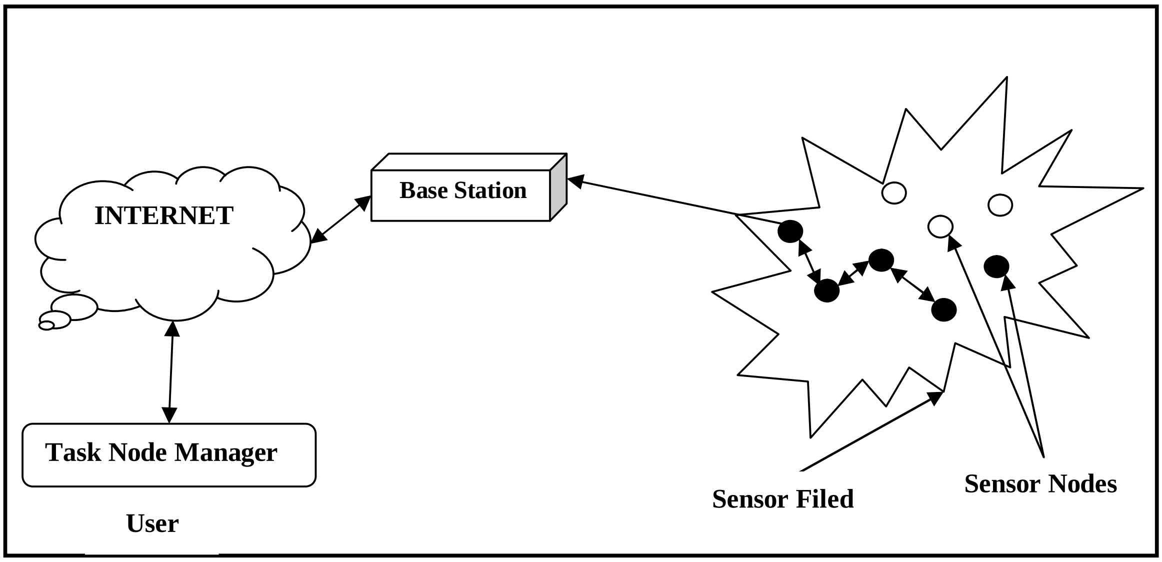

Figure 2.2 : Communication Architecture of WSN

Though the above described architecture is well suited to almost all the applications of WSN, communication in WSN provides an interplay between the energy which is spent on communication vis a vis on computation. The goal is to understand the impact of these constraints on the overall usefulness of the sensor networks.

Since past few years, rigorous research that addresses the potential of collaboration among sensors in data gathering, processing, coordination and management of the sensing activity is being conducted. In most applications, sensor nodes are inhibited in terms of energy and communication bandwidth. Thus, innovative techniques to eliminate energy inefficiencies that shorten the lifetime of the network and efficient use of the limited bandwidth are vital especially in non-deterministic areas like remote jungles, undersea areas etc. Such constraints united with a typical deployment of large number of sensor nodes pose many challenges to the design and management of WSNs and necessitate energy-awareness at all levels. For example, one of the main objective is to find ways and means for energy-efficient routing and reliable transmitting of data from the sensor nodes to the sink so that the lifetime of the network is maximized [35].There is a definite need to enhance the power of each sensor individually so that they can carry out the task of sensing dedicatedly and for a much longer span of time.

The core challenge is thus to synthesize the capacity of the sensor without causing any extra burden on either the hardware or network as a whole.

2.2.3 Applications of WSN

Research in WSN supports abundant applications in almost all domains ranging from household appliances to military surveillance. The huge plethora of such applications is majorly classified into three main categories depending upon the initiative taking entity. Each of these categories is described as follows:

- ***Event Driven Model***

  The applications in this model are initiated by source and the source reports to the base station immediately on occurrence of an event. Event driven applications are generally delay intolerant and interactive. This model is majorly applicable to mission critical applications [18,24,28]. The interaction is between the sink and the group of nodes deployed to monitor the event in an unattended area. The data thus sensed is likely to be highly correlated and redundant. Also, the data traffic may be either of low or high intensity depending on it being sensed by a single sensor or by a set of sensors. This phenomenon of more than one sensor sensing the same event is known as event showers. Examples of such applications include earthquake, forest fire, military based applications etc.[7,19,100,125].

- ***Query Driven Model***

  Also termed as sink initiated model, applications in this model reports on the status of an area when requested. They have similar characteristics as that of event driven applications except that the data is pulled by the sink in the former case. For example, wind stream during storm [128].

- ***Periodic Model***

  Applications in this model periodically monitor the event and then report to the base station. The model is also referred to as continuous data delivery model. Applications include traffic control, weather tracking etc.[107].

The subsection below describes some of the applications of WSN belonging to one or more of the category listed above.

A) Military based Applications

As described earlier, US military sowed the seeds of the ubiquitous WSNs which we experience in today's world. Military based applications are so closely related to WSN that it is very difficult to ensure that whether motes were developed because of military and air-defence needs or whether they were invented independently and were subsequently applied to army services.WSN finds its usage in military in numerous ways like military situation awareness, detection of enemy unit movements on land and sea and battle field surveillance (target classification) etc.

Sensors have replaced the mines in the army areas as mines are obsolete and can be hazardous to civilians. These sensors, being connected wirelessly form a network, detect any intrusion of hostile units and alarm the army. Then, the prevention of intrusion will be the response of the army. The use of WSN ensures complete visibility of the field and makes communication over the long radius possible. Several applications related to this scenario of detection and classifications of objects have been developed. One such application is being demonstrated by Ohio State

University. The name of this project is *"A line in the Sand"* [117] and refers to the deployment of ninety nodes which are capable of detecting metallic objects. The ultimate objective is the tracking and classification of moving objects with significant metallic content and specifically the tracking of vehicles and armed soldiers. Other beings (for example, civilians) were ignored by the system.

A comprehensive study done in the field of military based applications reveals that most of the research efforts has been undertaken with reference to wartime scenarios. Peacetime applications such as homeland security, property protection and surveillance, border patrol, etc. are activities that perhaps in future sensors networks will undertake.

(B) Forest Fire Detection

Detection of forest fire is one of the prime examples of event driven applications which can be handled using WSNs. This application is of prime concern because in addition to causing the tragic loss of lives and a great hazard to ecologically healthy grown forests, they cause an irreparable damage to the atmosphere and environment(30% of carbon dioxide in the atmosphere comes from forest fires)[15].

The main source of fire in these abandoned/unmanaged areas is that they are filled with leaves, dry and parching wood etc. which together form an extremely combustible material and represent the perfect context for initial-fire ignition and act as fuel for later stages of the fire. The use of WSN for forest fire detection has gone beyond all its previous approaches(like manned observation towers, camera surveillance systems, satellite imaging technologies etc.). WSN Technology integrated with other technologies or networks has been used for detection and prevention of wood fire systems. Lloret et al. [7] has deployed a mesh network of sensors provided with internet protocol (IP) cameras where sensors detect the fire at the beginning and send an alarm signal to the base station. Son et al. [15] proposed a project for fire detection in South Korean Forest Fire Surveillance System (FFSS) which makes use of Minimum Cost Forwarding(MCFA) protocol [36] to sense humidity, temperature, and illumination to forward it to the base station node and then to the gateway. Hartung et al. [19] have used FireWxNet which is a multi-tiered portable wireless system for monitoring weather conditions in rugged wild-land fire environments. FireWxNet provides the fire fighters with the ability to measure fire and weather conditions. Conrad et al. [7] have given a business case for the Enhanced Forest Fire Detection System with a GPS project in Pennsylvania. They proposed to create a fire detection system using fire sensors and GPS devices.

(C) Traffic Control

Wireless Sensor networks are being extensively used for vehicle traffic monitoring and control. These applications make use of either overhead or buried sensors to detect vehicles and control traffic lights. Furthermore, video cameras are frequently used to monitor road segments with heavy traffic, with the video sent to human operators at central locations. The wireless capability of sensors clubbed with their low cost can contribute to revolutionise the way in which traffic monitoring and patrolling can be done. Cheap sensors with embedded networking capability can be deployed at every road intersection to detect and count vehicle traffic and estimate its speed. The sensors will communicate with neighbouring nodes to eventually develop a “global traffic picture” which can be queried by human operators or automatic controllers to generate control signals. Another more radical concept [101] is to attach the sensors to each vehicle. As the vehicles pass each other, they exchange summary information on the location of traffic jams and the speed and density of traffic, information that may be generated by ground sensors. These summaries propagate from vehicle to vehicle and can be used by drivers to avoid traffic jams and plan alternative routes.

(D)Environmental Monitoring

Another major area of research is monitoring and control of the environment. The most beneficial aspect of this application is the ability to produce a big picture of the environment being monitored. For example, use of WSN to track the mating habits of seabirds. Similar to vehicles, sensors are directly attached to animals particularly to large mammals. This type of arrangement allows sensors to exchange information when animals are near to each other. Two sensor applications which have used this approach are the SWIM project for monitoring whales and the ZebraNet project for monitoring Zebras[117]. Another application is monitoring river currents. The flow of currents in a river depends in part on the quantities and temperatures of water flowing from and to different tributaries. Positioning sensor nodes throughout the river can give the detailed information of the river currents and flow and mixtures of water from different tributaries. The information gained can also be used to track information about water ways.

(E) Medical Applications/HealthCare Systems

The use of Wireless Sensors in the arena of Medicine has brought a revolution in the diagnosis system as a whole.WSN specifically designed for medical applications are often referred to as Wireless Medical Sensor Networks (WMSN)[119]. They are being envisioned as medical devices that are implanted on a patient’s body and can be used to closely monitor the physiological condition of patients. They monitor the patient’s vital body signs (for example, temperature, heart rate, blood pressure, oxygen saturation etc.) and transmit the data in a timely fashion to some remote location without human intervention. A doctor can interpret the sensor readings to assess a patient’s condition. The application of the WSN in healthcare systems basically deals with monitoring of patients in clinical settings, home & elderly care center monitoring for chronic and elderly patients, collection of long-term databases of clinical data.

Table 2.1 presents the classification of each of the above explained application in the models described above.

Table 2.1: Nomenclature of WSN Applications

Sensor Network Application	Event Driven Model	Periodic Model	Query based Model
Forest Fire Detection	✓ (alarm for sudden fire detection)		
Military Based Applications	✓ (for sudden danger awareness as enemy enters in the territory)	✓ (monitoring or tracking the enemies)	✓ (monitoring or tracking the enemies)
Medical Applications	✓ (Alarms for sudden issues in patient's health eg: high blood pressure or low blood pressure alarm, Sudden increase in heart beat)	✓ (Patient monitoring to collect data periodically or constantly and send to the doctor when patient is at home)	✓ (Patient monitoring to collect data periodically or constantly and send to the doctor when patient is at home)
Traffic Control Applications	✓ (traffic rule violation alert , eg: vehicle crossing the red light)	✓	✓
Environmental Applications	✓ (sudden rainfall alert or temperature raise exceeds limit in any factory)	✓ (to forecast about floods , volcanic eruption ,rainfall etc)	✓ (to check temperature , humidity pressure at any time)

2.2.4 Taxonomy of Routing in WSN

Routing is the process of identifying a route from a source to a destination node for transmitting the sensed data and is achieved either by computing all routes before and restoring them or computing them when needed [33,55,120,140]. During this process, at least one intermediate node within the inter-network is encountered. The main design objective at the network layer is to develop energy efficient routing protocols which can contribute to prolong the lifetime of the network and prevent connectivity degradation by employing aggressive energy management techniques. Figure 2.3 illustrates the taxonomy of routing protocols developed and categorised on the basis of network structure and protocol operation and each one of these is described in short as follows.

- ***Flat Routing Protocols***

Flat routing protocols make use of a huge number of sensor nodes which collaborate to perform sensing. Because of this huge number, these nodes are not assigned anyparticular identification (id) and hence each node plays the same role. This leads to the usage of data-centric routing approach [56] which follows the request response concept[64,30].The protocols belonging to this category are SPIN[58], Directed Diffusion[39], Rumour Routing[29], MCFA,GBR, COUGAR and ACQUIRE[56].

Sensor Protocols for Information via Negotiation (SPIN) [58] protocol is amongst the early work which overcomes the disadvantages of traditional flooding and gossiping mechanisms by negotiation and resource adaptation. It works by negotiating for meta-data with its neighbouring motes instead of data itself. There are three messages defined in SPIN to exchange data between nodes namely ADV, REQ and DATA. ADV allows a sensor to advertise a particular meta-data, REQ requests the specific data and DATA message carries the actual data.In addition to the conventional SPIN protocol, several variations have been proposed in literature like SPIN with energy consumption awareness (SPIN-EC) [58], SPIN for broadcast networks (SPIN-BC), and SPIN with reliability (SPIN-RL) [68].

Directed Diffusion [39] is a highly energy efficient protocol. It uses the naming scheme to diffuse the data. This naming scheme defines attribute-value pairs for the data and queries the sensors when demanded by using those pairs only. In order to create a query, an interest is defined using a list of attribute-value pairs such as name of objects, interval, duration, geographical area, etc. It works in four phases namely interest propagation phase, gradient setup phase, reinforcement phase and data delivery phase to construct routes between the sink and the sensors of interest to the sink’s request.

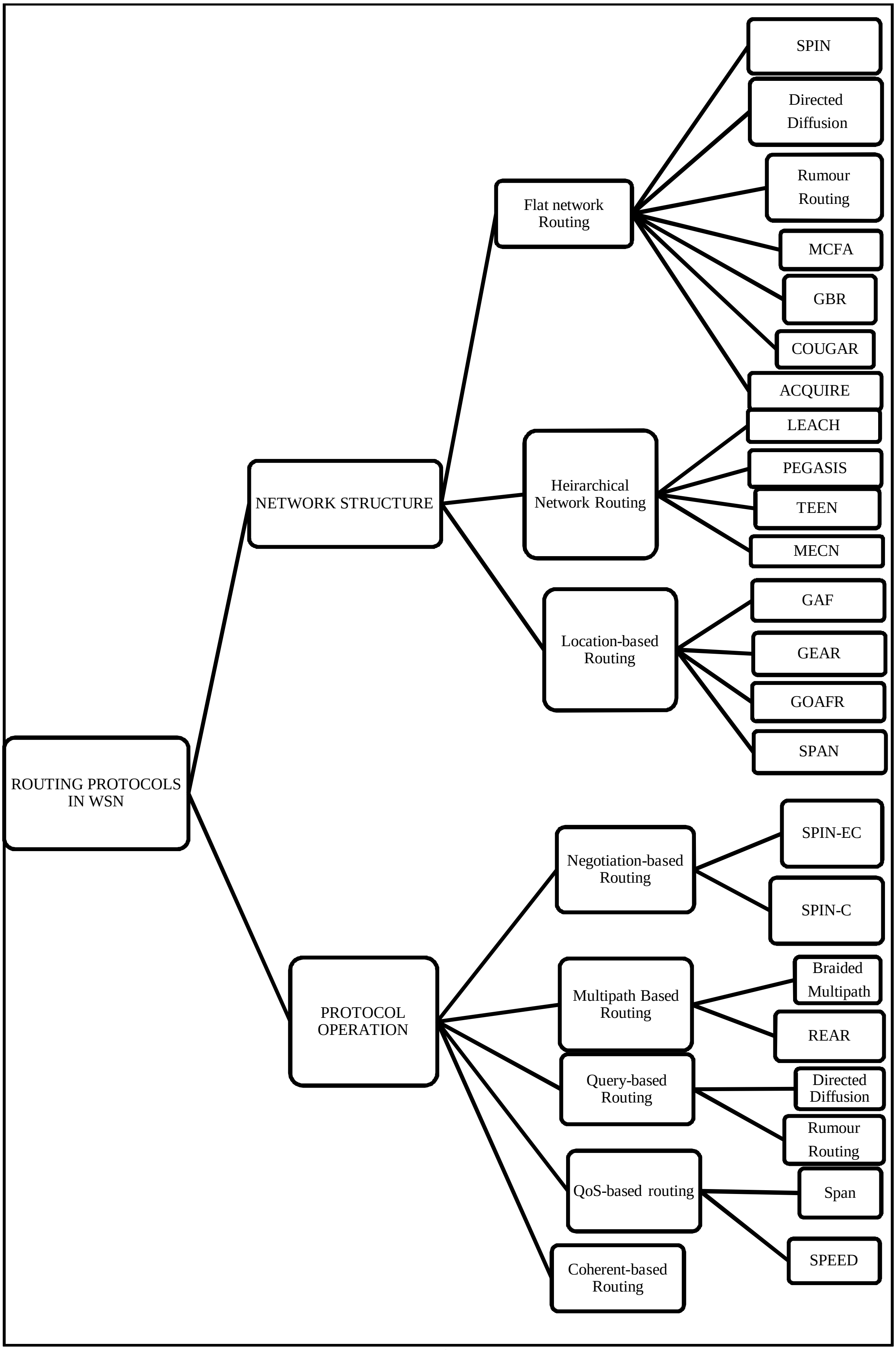

Figure 2.3: Taxonomy of Routing Protocols

Rumour Routing [29] is a variation of directed diffusion which mainly works for contexts in which geographic routing criteria are not applicable. It stands between event flooding and query flooding and proceeds by routing the queries to the nodes that have observed a particular event rather than flooding the entire network to retrieve information about the occurring events. Simulation results have shown that this form of routing results in substantial savings of energy over event flooding and is also capable of handling node failure.

- ***Hierarchical Routing Protocols***

The need of hierarchical routing emerges from the fact that single-gateway networks are not capable of long-haul communication. These networks cannot support additional loads and cannot cover a larger area of interest [35]. Hierarchical or cluster based routing methods, originally proposed in wire line networks, operate by assigning special tasks to high energy nodes.These higher-energy nodes are used to process and send the information, while low-energy nodes are used to perform only the task of sensing in the proximity of the target. This task of clustering can greatly contribute to overall system scalability, lifetime, and energy efficiency by performing the tasks of data aggregation and fusion in order to decrease the number of transmitted messages to the sink node[68].LEACH [86,135],PEGASIS[113], TEEN[2] and MECN[30] are some of the protocols following this strategy.

LEACH stands for Low Energy Adaptive Clustering Hierarchy Protocol[96,135]. It minimises energy consumption by cluster based operation. It dynamically selects sensor nodes as cluster heads based on incoming signal strength and forms clusters in the network..Cluster heads then directly communicate with the sink to relay the collected information from each cluster and saving the energy of rest of its cluster members which would otherwise have been used for communication. Cluster heads change randomly over time in order to balance the energy consumption of nodes. Selection of CH nodes is made by choosing a random number between 0 and 1.LEACH achieves as much as 70% reduction in energy dissipation as compared to direct communication and about 40%–80% as compared to the minimum transmission energy routing protocol. Alike SPIN, many variations of this protocol such as[58,61] have also been proposed and implemented by researchers.

Another protocol, called Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [113], is a near optimal chain-based protocol. It is basically an enhancement of LEACH protocol and was proposed to improve it by addressing the overhead caused by cluster formation in LEACH.It works by constructing chains of nodes instead of clusters according to a greedy algorithm, where nodes select their closest neighbors as next hops in the chain. It is assumed that the nodes have a global knowledge of the network and the chain construction starts from the nodes that are farthest from the sink. As a result of this chain operation, instead of maintaining cluster formation and membership, each node only keeps track of its previous and next neighbour in the chain. PEGASIS achieves over a factor of 3 reductions in energy consumption in comparison to LEACH for different sizes and topologies. On the other hand, it introduces excessive delay for distant node on the chain and bottleneck because of single leader. A variation of PEGASIS, called Hierarchical PEGASIS[113] has also been proposed to reduce the delay incurred for packets during transmission to the base station.

Threshold Sensitive Energy Efficient Sensor Network (TEEN) [2] is another hierarchical protocol designed for event driven applications such as forest fire detection.As the name implies, multi-hop routes are generated according to a threshold related to sensory data, which is set by the application. This protocol works by organizing the sensor nodes into multiple levels of hierarchy including sensor nodes and cluster heads. In order to evenly distribute the energy consumption, the cluster heads are periodically changed within the cluster.

- ***Location Based Routing Protocols***

In location based routing, sensor nodes are addressed depending on their locations. The distance between neighbouring nodes can be estimated on the basis of incoming signal strengths. Relative coordinates of neighboring nodes is obtained either by exchanging information between neighbor nodes or by directly communicating with a Global Positioning System (GPS) [146]. To save energy, some location-based schemes demand that nodes should go to sleep if there is no activity. Protocols belonging to this family include GAF(Geographic Adaptive Fidelity),GEAR(Geographic Energy Aware Routing Protocol) and Span[54,61,68].

GAF was initially designed for MANETs but works equally well for WSNs[68]. It divides the given area into zones called grids. Sensors associate themselves with a point in the virtual grid with the help of GPS and collaborate with other sensors in that zone.GAF works in three states. The discovery state determines the neighbors in the grid; the active state reflects the participation in routing; and the sleep state describes when the radio is to be turned off. In order to handle mobility, each node in the grid estimates its time of leaving the grid and sends this to its neighbors. The sleeping neighbors adjust their sleeping time accordingly in order to keep routing fidelity. Before the leaving time of the active node expires, sleeping nodes wake up and one of them becomes active.GAF is implemented both for non-mobile (GAF-basic) and mobile(GAF-mobility adaptation) nodes.

Geographic Energy Aware Routing(GEAR) [47] is an enhancement of the traditional directed diffusion protocol as already described.It uses energy-aware and geographically informed neighbor selection heuristics to route a packet toward the destination region. The protocol works by determining the user’s interest as in directed diffusion but restricts them to only a certain region rather than to the whole network resulting into conserving more energy as compared to Directed Diffusion.

Span is a position based algorithm which works by selecting some sensor nodes to act as coordinators based on their positions. A particular sensor node can become a coordinator if two neighbors of this non- coordinator node are unreachable either through a direct link or via one or two coordinators (three-hop reachability). These coordinators then form a network backbone used to forward messages. New and existing coordinators are not necessarily neighbors in [30], which in effect makes the design less energy-efficient because of the need to maintain the positions of two or three-hop neighbors in the complicated Span algorithm.

- ***Negotiation Based Routing Protocols***

This category of protocols makes use of high level descriptors in order to eliminate redundant data transmissions through negotiation. The prime advantage of using negotiation-based routing in WSNs is to restrain duplicate information and prevent redundant data from being sent to the next sensor or to the base station by conducting a series of negotiation messages before the real data transmission begins. The SPIN family of protocols (SPIN-P, SPIN-BC,SPIN-EC etc.) [58] are few negotiation based protocols.

- ***Multi-path Based Routing Protocols***

This type of routing uses multiple paths rather than a single path in order to enhance network performance. Multiple path leads to increased fault tolerance, improved network reliability between the source and destination at the expense of increased energy consumption, traffic generation and cost of maintaining alternate paths[3,80].Alternate paths are kept alive by sending periodic messages. This type of routing is able to meet three objectives: path discovery, path distribution and path maintainenance. Directed Diffusion [30] is an example of Multipath routing Protocol. Few other multipath routing protocols being described below:

Braided Path Routing [3] works by initially computing a primary path for the packets to be routed. Then, for each sensor node on this primary path, the best path from a source sensor to the sink that does not include that node is computed. The best alternate paths may not be disjoint from the primary path and are called idealized braided multipaths. Moreover, the links of each of the alternate paths lie either on or geographically close to the primary path. Therefore, the energy consumption on the primary and alternate paths is likely to be comparable as opposed to the scenario of mutually ternate and primary paths. The braided multipath can also be constructed in a localized manner in which case the sink sends out a primary-path reinforcement to its first preferred neighbor and alternate-path reinforcement to its second preferred neighbor**.**

Reliable Energy Aware Routing(REAR) [119] considers residual energy capacity of each sensor node in establishing routing paths and supporting multi-path routing protocol for reliable data transmission. In addition, REAR allows each sensor node to confirm the success of data transmission to other sensor nodes by supporting the DATA-ACK oriented packet transmission [30].

- ***Query Based Routing Protocols***

As the name suggests, in these protocols, destination node(s) forwards a query for data to a sensing node through the network [56,64]. The node which has the data matching the query then sends the data back to the node that initiated the query. Usually these queries are described in natural language or high-level query languages.

Directed Diffusion and Rumour Routing protocols described above fall under the category of Query based Protocols.

- ***QoS Based Routing Protocols***

This category of routing protocols ensure that while delivering data to the base station, the network satisfies certain QoS metrics like delay, energy, bandwidth etc. Some of the protocols which fall under this category are described in subsequent paragraphs.

Sequential Access Routing(SAR) [30] is one of the first routing protocol which introduced the concept of QoS into routing decisions. SAR works by calculating a weighted QoS metric as the product of the additive QoS metric and a weight coefficient associated with the priority level of the packet.SAR minimizes the average weighted QoS metric throughout the lifetime of the network. To meet the problem of changing topology, base station pre computes the triggered path periodically.

SPEED protocol [127] is particularly used for real time communication.It requires each node to maintain information about its neighbours and uses geographic forwarding to find the paths. In addition, SPEED strives to ensure a certain speed for each packet in the network so that each application can estimate the end-to-end delay for the packets by dividing the distance to the sink by the speed of the packet before making the admission decision.

- ***Coherent Based Routing Protocols***

This category of routing protocols takes into consideration the node processing ability of the sensor units. In coherent routing, the sensor nodes which sense the data only timestamp it and forward the raw data to the aggregators. On the other hand in case of non-coherent routing[43],the data is first processed locally are forwarded later.

However due to both wireless communication effects and the peculiarities of sensor networks, the process of routing has become a complicated issue. Further, routing protocols in WSNs also differ depending on the application and network architecture. Next section presents the issues and prominent challenges that are still prevailing in WSN especially in routing domain.

2.2.5 Challenges in Routing in WSN

Despite the fact that researchers have been putting efforts to improve the routing protocols, still study of literature revealed that there are many unfolded challenges dominating in routing in WSN. Few important ones are discussed below.

- ***Deployment of Nodes***

The topological deployment of sensor nodes is application specific which in turn substantially affects the performance of routing protocol. In case of deterministic deployment, the motes are manually placed and data is routed through pre-determined paths. However; in non-deterministic deployment, nodes are scattered randomly creating an adhoc infrastructure [54,61]. In such an infrastructure, the position of the sink or the cluster-head is also crucial in terms of energy efficiency as inter sensor communication is between short transmission ranges. When the distribution of nodes is not uniform, optimal clustering becomes a vital issue to enable energy efficient network operation. Due to bandwidth limitations a multiple hop route is preferred.

- ***Energy Preservation***

One of the major limitations of sensor nodes is the limited energy supply. The major portion of residual energy of these resource constraint miniature devices is consumed in discovering the neighboring mote and performing the tasks of both computations and transmitting information in a wireless environment. Further, as the transmission power of a wireless radio is proportional to distance squared or even higher order in the presence of obstacles, multi-hop routing consumes less energy than direct communication. However, multi-hop routing introduces significant overhead pertaining to topology management and medium access control.As such, formulating energy-conserving forms of communication and computation are essential especially in a multi hop environment [16,46].

- ***Node/ Link Heterogeneity***

Heterogeneous nodes such as temperature sensors, pressure sensors, and oxidant sensors are often deployed in WSN to meet the application's needs. This heterogeneous set of sensors can generate readings and report data at different rates, lead to diverse quality of service constraints and even follow multiple data reporting models. Dealing with such a diverse mix of sensors makes the process of routing more complex and challenging [51,63]**.**

- ***Fault Tolerance***

The routing protocol designed for WSN should be robust enough to handle the worst case functioning of the network. The sensor units can fail due to various reasons such as malfunctioning hardware, software glitches, dislocation or environmental hazards, for example, fire or flood. In such a case, medium access control (MAC) and routing protocols together must accommodate formation of new links and route to the data collection BSs. This may require rerouting packets through regions of the network where more energy is available or actively adjusting transmitting powers and signaling rates on the existing links to reduce energy consumption. Therefore, multiple levels of redundancy may be needed in a fault-tolerant sensor network [118]**.**

- ***Scalability***

Usually, in a non-deterministic environment, sensor motes are required to monitor the whole environment. For example, for habitat monitoring or forest fire detection, order of hundreds or thousands or more number of sensors are deployed. The routing protocols should be scalable enough to support the growing size of network [56]**.**

- ***Network Dynamics***

Network dynamics refers to the study of topological changes in the network. For few applications, static sensor nodes and hence simple routing may suffice however; for applications that require mobile sinks or cluster-heads, routing messages amongst the moving nodes is challenging in addition to other factors such as energy, bandwidth etc. Monitoring static events allows the network to work in a reactive mode i.e. simply generating traffic when reporting while dynamic events require periodic reporting and consequently generate significant traffic to be routed to the sink [46].

- ***Connectivity***

Connectivity implies ensuring that all the nodes within the network are connected in such a manner that the data can traverse from source to destination. Now since a WSN is usually a dense sensor network therefore it is apparent that most of the nodes stay connected. These connections should further allow dynamic network topology and should pose no shrinking affect due to node failures. Further a routing protocol is expected to ensure maximum connectivity even when distribution of nodes is random [24].

- ***Coverage***

  Because of the hardware constraints, a typical sensor is limited in its sensing range and can thus cover only a limited physical area of the given environment. This property of sensor nodes makes optimal coverage a design parameter for any of the routing protocol[106].

- ***Quality of Service***

  In certain applications, timely delivery of the sensed data is as crucial as its accurate delivery. Data which is not delivered in a stipulated period of time from the moment it is sensed becomes useless. Therefore, bounded latency for data delivery is another parameter for time-constrained applications. On the other hand, in many applications, conservation of energy, which is directly related to network lifetime, is considered relatively more important than the quality of data sent. As energy is depleted, the network may be required to reduce the quality of results in order to lengthen the total network lifetime. Hence, energy-aware routing protocols are required to ensure quality of service [31].

An in-depth analysis of the routing protocols and their applicability in various application scenarios reveal the fact that though extensive research has been done in the field of deterministic WSN, the researchers have almost remained silent in the case of non-deterministic environment. Further, researchers have been spending time in exploring and exploiting an energy efficient alternative to the conventional client server paradigm used for routing but to the best of our knowledge, none of the above listed protocols is found to be best suitable for all applications associated with non-deterministic environment. *The deficiencies of this traditional paradigm like excessive* *consumption of bandwidth, traffic on the network channel, packet delays etc. are of utmost concern especially for mission critical applications.*

A special class of software agents called mobile agents seems to be a promising solution to the above stated problems because of their properties like autonomy, pro-activity, reactivity, intelligence etc. and their adaptability to a distributed environment.

The subsequent section presents an in-depth study of software agents in general and mobile agents in particular.

2.3 SOFTWARE AGENTS

The term agent comes from greek '*agein*', which means to drive or to lead [40]. Though the term has a very broad scope, it has been explored by computer science community to describe current trends in computer science and develop programming techniques and software that enable a more active role of the computer. In the domain of computer science, it is thus termed as software agent. Defining it more precisely, "Software Agent is *an object of the environment that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators"* [74]**.**These special software entities carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires [104].

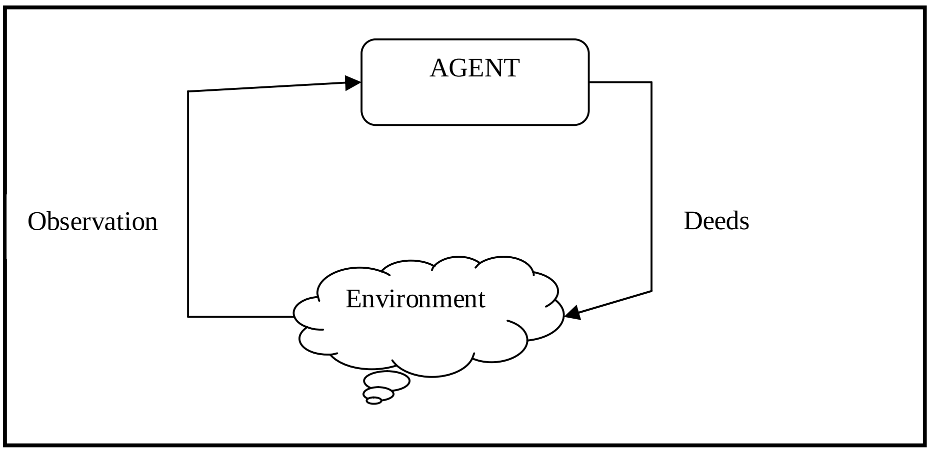

Figure 2.4 : Agent Interaction Environment

Though the concept of Software Agent looks much like the conventional objects of OOP, it differs from them in terms of the degree of autonomy or the *self-initiating-property*.

2.3.1 Evolution and Behaviour

The concept of software agents owe its roots in the Distributed Artificial Intelligence(DAI) research which was conducted about 3 decades ago. Carl Hewitt proposed an Actor system [74] where each Actor had an explicit internal state and had the capability to respond to the messages of various other Actors. The subsequent years focused on the more theoretical aspect of bringing intelligence to software agents. The last two decades have seen a huge expansion of systems to solve practical problems drawing on changes in distributed processing, object-oriented programming, the Internet, the Web and the increased digitization of information and services.Study in this domain reveals that the agent paradigm is an extension rather than a replacement of conventional systems that are either object-oriented or component-based. The software Group at MIT compares and contrasts software agents to conventional software and highlights the differences to be as "*Software agents differ from conventional software in that they are long-lived, semi-autonomous, proactive, and adaptive"*[104].The realm of software agents has fascinated the researchers to a great extent because of its vast set of primary and secondary properties which make it stand ahead of its counterparts. This section describes in detail the various features of agents.

- ***Social Ability***

  It is the ability of software agents to communicate with other agents that constitute a part of its environment [74]. The communicating agents may work towards a single global goal or separate individual goals.

- ***Autonomy***

Autonomy refers to self-governance[1]. According to this property, an agent can function on its own without the need of human guidance or any external elements. They have control on their own actions and internal states.

- ***Reactivity***

Agents can distinguish their environment and respond in a timely fashion to the changes that occur. Pure reactive agents do not have any internal symbolic models of their environment and they act using a response type of behavior by responding to the present state of the environment in which they are embedded. The agents have no goals and even no internal states.

- ***Adaptability***

Agents can adapt to changing environment and can set up their own goals based on their implicit purpose[26]. They attain and process situation information, both spatially and temporally.

- ***Intelligence***

Intelligence is the property when a software agent is able to incorporate knowledge-based technology and act proactively on perceiving the dynamic state of its environment [74]. It is this ability of software agents which make them unique and an energy efficient solution in a given distributed environment.

- ***Learning***

The property of intelligence helps software agents to learn by their experience and adjust their future action sequences and behavior so as to avoid future mistakes [26]. Hysteretic agents act based on perceptions and also past experience.

- ***Pro-activity***

  Pro-activeness or self-starting capability refers to the ability of agents to take the initiative rather than acting simply in response to their environment.

- ***Goal-oriented***

  Agents should exhibit goal driven behavior that their action will cause beneficial changes to the environment.

- ***Mobility***

  A mobile agent can migrate between various machines to perform assigned tasks. Mobility is neither a necessary nor a sufficient condition for agent-hood.

The above listed properties of software agents distinguish them from regular objects and expert systems. Table 2.2 gives a tabular comparison of agents, objects and expert systems.

Table2.2 : Comparison of Agents, Objects And Expert Systems

Entities / Properties	Agent	Object	Expert system
Social	✓	✓	
Autonomous	✓		✓
Reactive	✓	✓	
Adaptable	✓		
Intelligent	✓		✓
Goal Oriented	✓	✓	✓
Mobile	✓		✓
Learning Ability	✓		✓
Pro-Activity	✓		✓

However, an agent may or may not possess all the above-mentioned characteristics for acting in an agency. They can in fact, function efficiently even by possessing some of the features. The taxonomy of agents, depending upon the properties they possess is given in the next section.

2.3.2 Taxonomy of Software Agents

This section describes the taxonomy of software agents on the basis of the properties which they possess. The various types of agents are shown in Figure 2.5

- ***Collaborative Agents***

  These agents are autonomous, reactive, social and pro-active in nature. They can rationally act in a given open and time-constrained multi-agent environments in an autonomous manner. They are static and large coarse-grained agents which can together create a system that interconnects separately developed collaborative agents, thus permitting the group to function beyond the capabilities of any of its member.

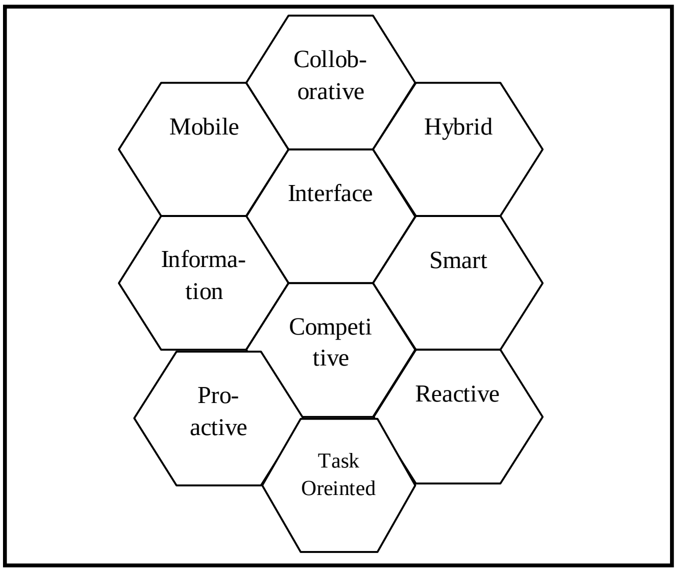

Figure 2.5 : Taxonomy of Software Agents

- ***Interface Agents***

  These agents are autonomous and good learners. They also have communication ability and communicate with the user instead of other agents. Their cooperation with other agents, if any, is limited typically to asking for advice. They learn to better assist their user by observing and imitating the user, receiving feedback and explicit instructions from user and asking other agents for advice. They try to perform some direct manipulation tasks in order to accommodate novice users. The major motivation behind the design of such agents is to eliminate the tedium of humans in performing several manual sub-operations.

- ***Mobile Agents***

Mobile agents are capable of executing on different machines in a dynamic networked environment, and sense and (re)act autonomously and proactively in this environment to realize a set of goals or tasks. They are thus autonomous, social and adaptable mobile entities. A Mobile agent approach trades server computation and cost for savings in network bandwidth and client computation. This approach is advantageous when the server's CPU is not a bottleneck. These types of agents provide a natural development environment for implementing free market trading services. The flexible distributed computing architecture and mobile agents provides a radical and attractive rethinking of the design process[65].

- ***Information Agents***

Also known as *Internet agents or Internet Softbots*, these agents act as tool to manage information explosion. They collect, manage, and manipulate information from various distributed sources. These agents have the primary properties of being autonomous and social. In addition, they also have varying characteristics: they may be static or mobile, non-cooperative or social, may or may not learn.

- ***Reactive Agents***

These agents react in a stimulus-response manner to the present state of the environment in which they are embedded. A reactive agent is viewed as a collection of modules, which operate autonomously and are responsible for specific tasks. Communication between the modules is minimized when using these agents. They tend to operate on representation, which are closed to raw sensor data. They are autonomous and social but unintelligent.

- ***Proactive Agents***

Unlike reactive agents who just react to situation as directed, proactive agents are able to reason on the changes in environment, hence can change their intentions and beliefs, can change plan of action and execute the actions.

- ***Hybrid Agents***

Hybrid agents refer to agents which consist of two or more agent philosophies. Hybridism usually translates to ad hoc or unprincipled designs. Many hybrid architectures tend to be very application-specific.

- ***Smart Agents***

As the name suggests, these types of agents possess almost all the features of agents including autonomy, social ability, proactivity, adaptability, intelligence, ability to learn, goal oriented and ability to travel from one environment to the other. However, no agent today belongs to this class of agents. Agents of such form offer many open research issues such as standardization of agent oriented technology, infrastructure and technology needs to be established before real smart agents can be developed and deployed.

- ***Competitive Agents***

Competitive agents are autonomous, social, reactive and proactive. They are quite similar to collaborative agents except that they may compete with other agents in order to perform tasks for their owners. They may not perform task for other agent request if it may be detrimental to its objectives.

Considering the above significant types of agents, it is evident that over the years the big community of information have been restructured into smaller elements called agents that are now possessed to have many abilities well suited to handle complex applications. In other words, a group of software agents can be treated as a dynamic collection of simple agents with self-describing interfaces who while progressing through their life span continue to collect the information in a “knowledge soup,” and same is presented to the user according to the application in which these agents operate [108]. In such an environment, individual agents do not manage the entire system to achieve the target where as they would interact and collaborate actively forming a multiagent system (MAS) [74]. In such a system agents integrate to provide the needed information as discussed in detail in the next section.

2.3.3 MultiAgent Systems (MAS)

A system in which a community of software agents works in collaboration and coordination with each other to achieve a common goal is called a Multi-Agent System. It has evolved as a result of sociological relationships among agents [1]. In MAS, at least one agent has an ascribed set of goals and other agents in the system may adopt the goal. The goal describing agent(s) is referred as the Coordinator or parent agent and the goal adopting agents are called as *Sociological Agents* as shown in Figure 2.6.

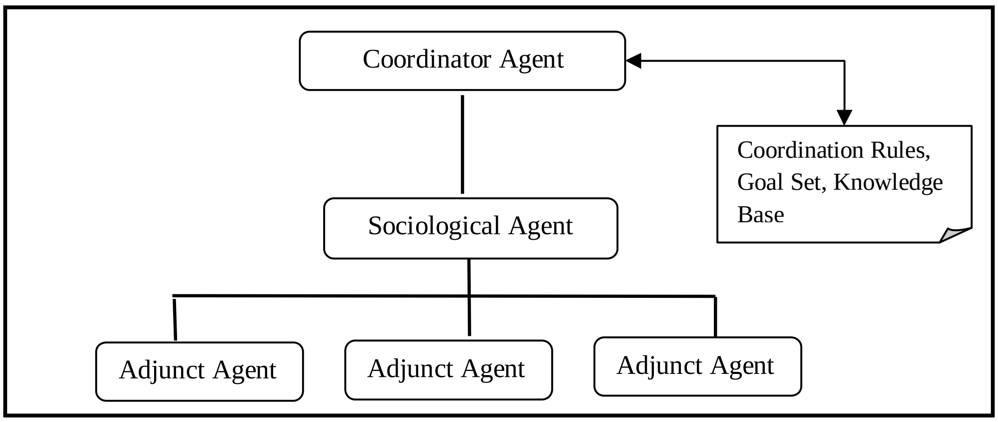

Figure.2.6: Multi-Agent System

Once an agent adopts the goal of goal describing agent, a relationship is created between the two. If the goal cannot be achieved at this level it may again be forwarded to different homogeneous or heterogeneous adjunct agents. More precisely, MAS can be defined as a loosely coupled network of problem solvers that work together to solve problems that are beyond the individual capabilities of knowledge of each problem solver.

A typical MAS inherits most of the advantages of distributed intelligence over centralized and sequential processing. Table 2.3 delineates the distinguishing characteristics of MAS.

A MAS comprises of variety of homogeneous as well as heterogeneous agents having different behavioral aspects. In a MAS, agents are usually classified on the basis of their behavior and locomotive ability [105]. Since the behavior of software agents is observed within MAS, the agent community refers the former group of agents as *internal agents*.

Table 2.3: Characteristics of a MultiAgent System

Characteristic	Description
Dynamic	Agents in MAS form dynamic groups to solve specific problems, pool together resources and disband after the problems are solved releasing resources to local usage.
Robust	MAS is more reliable and fault-tolerant as compared to individual agents serving a goal.
Concurrent	A MAS can make member agents reason and perform system tasks in parallel and asynchronously, resulting in faster and flexible execution of tasks.
Adaptive	Agents in MAS can re-configure themselves to suit system changes such as noise, resource allocation and faults without disturbing the entire system.
Scalable	Agents can be added or deleted without greatly disrupting the system.

In the same manner, since the locomotive ability is noticed when the agents communicate outside the system, later group of agents is referred to as *external agents*. The communication between internal and external agents is made possible through *mediator agents.* The categories of agents listed above are described in short as follows:

A) *Internal Agents*

Internal Agents are local to a particular architecture and are classified according to the role being played within the environment i.e. how they behave inside the system and with each other. Figure 2.7 illustrates the classification of internal agents in a multiagent system. Within a MAS, agents can be cooperative (collaborate with each other and share some common goals), self-interested (with distinct goals), competitive (possess mutually exclusive goals), destructive (intentionally provide wrong information), interface (takes the input and ultimately deliver the output by delegating the task to other agents), task-oriented Agents (the lowest level of agents), reactive (reacts as directed) and proactive (may decide on its own to improve the probability of success).

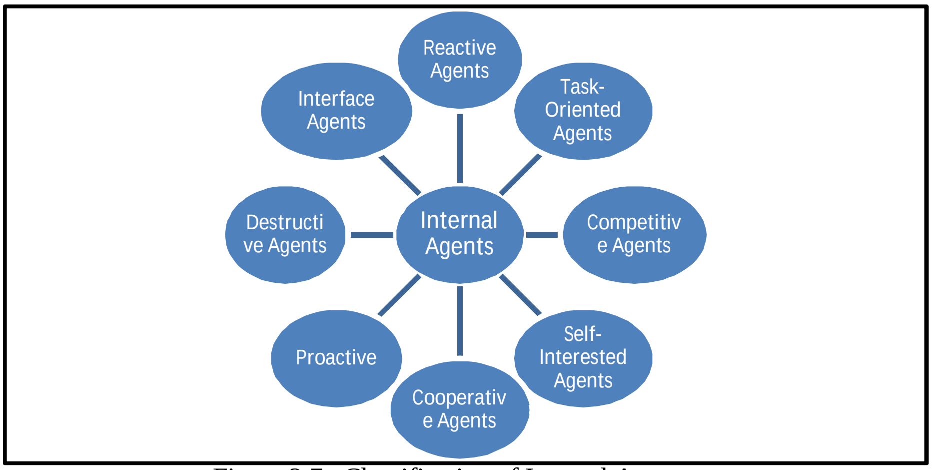

Figure 2.7 : Classification of Internal Agents

B) *External agents*

External Agents are also referred to as dynamic agents as these possess the ability to change their residing locations. These agents move out of a system to perform a task and these may or may not return to the initiating point. External agents fall into two categories namely mobile agents and ants. The agents which move out to different locations to gather the desired information for carrying out a task and then returning to originating node are termed as mobile agents [104]. Mobile agents spread intelligence across networks. Ongoing research in the domain of mobile agents reveals the fact that these agents have the potential to provide an energy efficient solution in a networked environment. In contrast to mobile agents ants are special external agents which originate at one node, keep on changing their locations and may die on any other node [74]. Ants follow the principle of "STIGMERGY". It is a form of indirect communication through an environment. The insect ants when move in search of food stimulate a hormone named as pheromones, which attracts other surrounding ants. In the same fashion, the routing of an ant-based agent is pheromone distribution dependent where pheromone distribution depends upon the environment in which the agent's properties are utilized. Swarm intelligence stems from the work of ants in which unintelligent internal and external agents possibly belonging to heterogeneous platforms, work independently or with relatively small amount of collaboration to achieve a greater goal that requires intelligence.

C) Mediator Agents

As the name suggest, mediator agents provide an interface between internal agents and external agents. These agents come into picture when one category of agents needs support from the other category of agents. In other words, mediator agents are also called as matchmaking agents as these facilitate the communication between service requester and provider.

Current research work more dominantly employs the mobile agents to improve the efficiency of WSN. Therefore the next section provides an overview of mobile agents in detail.

2.3.4 Mobile Agents

Mobile agents are a distributed computing paradigm. As already described in section above, a special class of software agent which is mobile in nature i.e. which can migrate and execute on different machines in a dynamic networked environment is known as Mobile Agent[43]**.** Like any other agent, it senses and (re)acts autonomously and proactively in a given environment to realize a set of goals or tasks. A typical mobile agent can migrate from one machine to another under its own control and can suspend execution any time.

The inception of mobile agent paradigm for communication is a result of limitations/weaknesses in its predecessor paradigms. One of the first paradigm proposed in this light was the message passing paradigm [74] which lets processes communicate by explicitly sending and receiving messages. Both asynchronous and synchronous message passing is being used for communication. However, literature survey reveals that developing distributed applications based on message passing primitives is a very complex and error-prone task, and programs are hard to analyse and debug.

Next paradigm that came into existence was remote procedure call (RPC).Through this concept , the research community successfully introduced the concept of mobility of data in a given network. In this form of communication, all processes call remote procedures rather than explicitly sending and receiving messages. An RPC supports client/server-style of interactions in which clients issue requests to servers, which execute the requested procedure and then return the results. Like message passing paradigm, many RPC mechanisms support both asynchronous and synchronous calls. The success of RPC inspired the researchers to move code along with the data. A piece of program was sent to another machine and executed there. This is called remote evaluation (RE) if the sender starts the action, while it is called code on demand (COD) if the receiver does it**.**

While remote evaluation only allows for ‘code mobility’, the concept of a mobile agent moved a step ahead and provided a support for ‘process mobility’, i.e., program executions may migrate from node to node of a computer network[99].Obviously, for migrating agents not only code but also the state information of the agent has to be transferred to the destination.

An agent’s state is further subdivided into data state and execution state. While the first includes the agent’s global variables and instance variables, the later comprises the local variables and the active threads. These two types of migration of the mobile agent are termed as weak migration and strong migration respectively and are shown in the Figure 2.8.

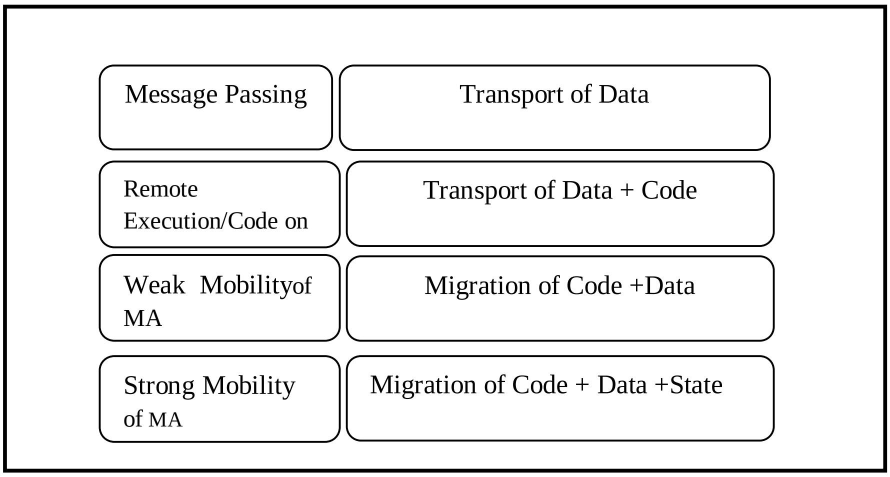

Figure 2.8. : Degrees of Mobility

In case of strong migration, the underlying system captures the entire agent state (consisting of data and execution state) and transfers it together with the code to the subsequent location. Once the agent is received at its new location, its state is automatically restored. Though this type of migration captures, transfers and restorates the complete agent state completely transparently by the underlying system, providing this degree of transparency in heterogeneous environments at least requires a global model of agent state as well as transfer syntax for this information. Moreover, the given agent system must support functions to externalize and internalize agent state. Only few languages such as Facile and Tycoon [74] allow externalizing state at such a high level. It was also realized that transferring the complete agent state can be cumbersome, particularly for multi-threaded agents and thus strong migration might be a very time-consuming and expensive operation. These difficulties thus led to the development of the so-called weak migration scheme, where only data state information is transferred. The size of the transferred state information can be limited even more by letting the programmer select the variables making up the agent state. As a consequence, the programmer is responsible for encoding the agent's relevant execution states in the program variables. While this method may substantially reduce the amount of state to be communicated, it puts additional burden on the programmer and makes agent programs more complex. Table 2.4 compares and contrasts the various entities described above on the basis of their mobile units.

Table 2.4 :Mobile Units in Different Paradigms

	Message Passing	RPC	Remote Evaluation	Code on Demand	Mobile Agents
Data		✓	✓	✓	✓
Code			✓	✓	✓
Execution State					✓
Examples	Smalltalk	Java	Java servlets	Java applets	Aglets, D'Agents

2.3.5 Benefits of Mobile Agents

The use of mobile agents has simplified the implementation of many applications in a networking environment. The various advantages incurred by using them are as follows:

- ***Reduction of Communication***

  The use of mobile agents reduces communication with respect to latency, bandwidth and connection time at the expense of minimal overhead for sending agent code and execution state across the network. Communication latency is reduced by sending an agent with a sequence of service requests across the network rather than issuing each service request by a separate remote procedure call. Similarly, communication bandwidth is controlled by migrating the agent across the network in order to deliver instructions for the generation of data on a remote host. It also gets reduced by moving the agent across the network to the source of data in order to reduce the data before transmission. An example for the reduction of communication by mobile code is the NeWS window system[144] where clients communicate with the display server by sending brief PostScript programs instead of drawing a grid by sending several thousand messages for individual points.

- ***Asynchronous tasks***

  With mobile agent technology, the client part of the application can be transferred from the mobile device to stationary servers in the network. From an end user's perspective, not only individual requests but the entire task is moved to the network, where it is performed asynchronously.

- ***Dynamic Protocols and Intelligent Data***

  Mobile agents permit dynamic protocols, i.e. new protocols to be installed automatically and only as needed for a particular interaction. To receive an agent initially, the client and server must share some standard protocol. Once the agent is running, though, it can use a specialized protocol for communication back to its home server. Further, an executing agent can communicate repeatedly with the server without intervention from the user, allowing the construction of dynamic services. For example a news service could transmit news updates to agents on distributed clients by using a special multicast protocol. A recent example of intelligent data is the MPEG4 compression standard for video, where the decompression algorithm is bundled with the data. This approach makes the standard highly flexible and allows the upgrade to use improved compression techniques.

- ***Software Deployment***

  Mobile agents can contribute to automate the software installation and updating process. These mobile entities are capable gathering information about the environment, query the user for installation preferences, configure the system, create directories and uncompress and compile the software. However, this approach to software deployment has its limitations since it might not be possible to capture every special case and error condition of the installation process and the programming of suitable deployment agents might become very difficult. A better approach to software deployment would be to use the agent language itself, since the agent language is in particular designed to prevent such damage.

- ***Temporary Applications***

In addition to deploying software packages, the agent could be the application itself. In many of the cases, an application-agent might be self-contained and has no communication or migratory needs at all. It is much smaller than a stand-alone application since it could exploit the infrastructure provided by the mobile agent system. Examples of such temporary applications can be travel guides and route planners downloaded on a mobile computer for a particular trip and discarded afterwards. Upon arrival at a new location, the user might temporarily download services that are specific to the new environment. Java based applets are also the examples of applications of mobile agents.

- ***Distributed and Heterogeneous Computing***

Mobile agents can also serve as the basis for general-purpose distributed and heterogeneous computing. They provide the necessary infrastructure for communication between the tasks in a heterogeneous environment. The agent system furthermore supports the independent compilation and initiation of agents so that further agents can be assigned to a task at runtime. Prospective applications for agent-based distributed computing are parallel algorithms with a reasonable low communication overhead compared to its computation requirement and particle or object based simulations.

2.3.6 Application Areas of Mobile Agents

The benefits of using mobile agents in any networked environment have attracted researchers to implement them in various application domains.The section discusses the key application areas of mobile agents such as information retrieval, E-commerce, network management, load management, just to list a few.

- ***Information Retrieval***

In information retrieval applications, mobile agents typically visit several nodes of the network in search for given information. The number of sites to be visited, often called itinerary, can be either statically defined at the agent creation or dynamically built from the information the agent collects during its travel. For example, when one agent visits several Web servers following the found links, its itinerary is dynamically generated by an agent-based solution which provides one or more agents that visit WWW servers searching for interesting pages [104].

- ***Electronic Commerce***

One of the most attractive applications of the mobile agent technology is electronic commerce. In such cases, the network nodes model virtual marketplaces and mobile agents well suit to model buyers and sellers that roam through a network to carry out exchanges of goods, services and money. A buyer (mobile)agent can travel from site to site to act in behalf of a user who wants to buy goods. The "intelligence" of the agent can be used to compare determine the cheapest service available in the marketplace and also to coordinate different services when required.

- ***Network Management***

The network today is a complex set of resources and services available for the applications. The use of mobile agents has provides a more scalable and reliable management model, because the execution of an agent occurs autonomously and locally to the devices/resources it is devoted to manage, without being affected by the network latency and by possibly intermitting network connections. Also, mobile agents suit both low-level and high-level network management issues [79]. These mobile entities can be executed onto specific devices to monitor, control and program and are also enriched with the necessary "intelligence" to deal with high-level application issues required by today’s networks, i.e., the management of services and data available to applications.

- ***Load Management***

  Mobile Agent technology provides a cost effective solution to the problem of distributing the load in a set of computational entities. In such cases, multi-agents are used to decentralize the distribution of the computational load [75]. In fact, a complex application can be divided into autonomous parts, each of which delegated to a mobile agent. Each mobile agent is in charge of searching for the most convenient node of the network, where to execute its own part of code. During the execution, agents can move to other nodes where more computational resources are available, in order to better distribute the load[144]. The case of load balancing applications is however quite different from other applications because it requires that the application is restored exactly as it was before the movement of agents, because it must be transparent to the application itself. It thus makes use of strong mobility mechanism for the same, which grants that also the execution state is transferred and resumed at the destination node.

Considering the above listed applications of mobile agents in various domains, a survey was carried to explore the feasibility of employing mobile agents in WSN so that current research work could proceed. Next section hence explores the same.

2.3.7 Exploring the Feasibility of Mobile Agents in WSN

The applicability of mobile agents in multiple domains attracted researchers working in WSN. One of the initial works was that given by Chen et al.[75,78] in which mobile agents were used for dissemination. The authors also gave various design issues for using mobile agents in WSN[77]. Thereafter the research community expoited these distributed autonomous mobile entities for aggregation in addition to data dissemination[76].Over the years, mobile agents have been used for energy balancing, extending the lifetime of the network etc. The use of mobile agents for one or the other aforementioned tasks in WSN also depends upon the application. With its exclusive set of features, mobile agent technology was thus found to be a promising solution to the resource constrained network of wireless sensors, especially when deployed in non-deterministic environment.The main factor which encourages the integration of mobile agents with wireless sensors in a given network is that they reduce communication cost. Mobile agents have increased flexibility provided by mobility and the agent itself can be send to the server for direct computation. Large amount of raw information transferred in order to determine their relevance can be very time consuming and clog of the networks. Mobile agent approach trades server computation and cost for savings in network bandwidth and client computation. Extensive simulation-based comparisons between Mobile agent based WSN (MAWSN) [75] and client/server based WSN (CSWSN) has revealed that, depending on the parameters, MAWSN considerably reduces the energy consumption while conditionally improving the end-to-end delay. Majority of the work exploited the ability of mobile agents (MA) to carry processing codes that allow the computation and communication resources at the sensor nodes to be efficiently harnessed in an undefined area. These intelligent units are capable of adjusting their behaviors depending on quality of service needs (e.g. data delivery latency) and the network characteristics to increase network lifetime while still meeting those quality of service needs. The basic advantage(s) of mobile agent technology in comparison to the client/server technology are given below:

- ***Scalability***

  The use of mobile agents supports scalability of sensor nodes in a network. Agent architectures that support adaptive network load balancing could do much of a redesign automatically [79,80].

- ***Reliability***

  Mobile agents can be sent when the network connection is alive and return results when the connection is reestablished. This asynchronous working of mobile-agent-based computing model is not affected much by the reliability of the network [110].

- ***Extensibility and Task Adaptivity***

  Mobile agents can be programmed to carry different task-specific integration processes which extend the functionality of the network.

- ***Energy Awareness***

  The itinerary of the mobile agent is dynamically determined based on both the information gain and energy constraints. It is tightly integrated in to the application and is energy efficient.

- ***Progressive accuracy***

  A mobile agent always carries a partially integrated result generated by nodes it already visited. As the mobile agent migrates from node to node, the accuracy of the integrated result is constantly improved assuming the agent follows the path.

The survey presented above unveiled the fact that there are numerous software agents which contribute to achieve energy efficient data dissemination in sensor networks. The current body of research focuses on mobile agents to achieve a secure and efficient communication. Mobile agents are mainly characterized by autonomy, adaptability and mobility. The main factor which encourages the development of mobile agents is that they reduce communication cost. Mobile agents have increased flexibility provided by mobility and the agent itself can be send to the server for direct computation. Large amount of raw information transferred in order to determine their relevance can be very time consuming and clog the networks. Mobile agent approach trades server computation and cost for savings in network bandwidth and client computation. The approach is advantageous when the server's CPU is not a bottleneck. It gives performance optimization for distributed operations that involve heavy network delays and/or weak connectivity; extended autonomy in terms of existing support for asynchronous execution and disconnected operations. It provides a natural development environment for implementing free market trading services. The flexible distributed computing architecture and mobile agents provides a radical and attractive rethinking of the design process.

2.4 CONCLUSION

The chapter provided the motivation behind the research work being carried out in this thesis. It initially detailed the basic concepts of wireless sensor networks and thereafter focussed on presenting the properties of software agents and mobile agents, in particular that make them suitable to be employed in WSN. Next chapter presents the literature survey by exploring the contribution of eminent researchers and the drawbacks of fundamentals protocols deployed in WSN laying the foundation for current research work.

Chapter III

MOBILE AGENTS IN WIRELESS SENSOR NETWORKS: A LITERATURE REVIEW

3.1 INTRODUCTION

Wireless sensor networks have seen the exponential growth and have also received considerable attention [13,24,54,56,128] as these small and inexpensive entities can sense the environment, process data and help scientists to take decisions based on the inputs. WSNs have been applicable in various sensitive domains such as target detection, surveillance and environmental monitoring. Exponential developments towards reducing the size of motes and improvements in technical designs including hardware and software have led to advancements in this domain. It is evident from the architecture of sensor node presented in the previous chapter that lifetime of a network is primarily dependent on the battery of the node. Hence, it is of utmost importance to employ energy efficient protocols for primary tasks i.e. sensing, networking and communication. In the light of current body of research, sensors have been considered to be deployed in non-deterministic area, hence the deployment, networking and communication shall be carried out efficiently.

An in-depth study of literature revealed that lot many system architectures, communication protocols and data aggregation algorithms are available addressing the need of energy efficiency [35,45,70,107,118]. In a non-deterministic environment, wireless sensors are deployed randomly and they form an adhoc network. Nonetheless, various algorithms supporting the hierarchical clustering for efficient coverage and connectivity and information processing are available in literature and are being discussed in the upcoming sections. However, very few researchers have thought of employing mobile agents to improve the efficiency of these highly useful tiny motes. The chapter focuses on exploring the feasibility of mobile agents in WSN by citing the work of eminent researchers in the related field. Also, since the focus of the current research is to propose energy efficient protocols, therefore related with respect to communication, clustering in WSN and information processing is also being presented in different sections.

3.2 MOBILE AGENTS BASED WIRELESS SENSOR NETWORKS

Although mobile agents have gained a lot of attention in late nineties but very few proposals are available exploiting their capabilities in wireless sensor networks. Owing to the high cost of deploying hundreds and thousands of sensor nodes in any non-deterministic environment (like dense forests, under sea etc.) the deployment is done with an aim to achieve the trend of "*One deployment, multiple applications*"[35,81]. However, this trend requires sensor nodes to have various capabilities to handle multiple applications. But, it is impossible to store the programs required to run every possible application in the local memory of embedded sensors, as these devices generally have stiff memory constraints. Because of this reason, use of mobile agents to dynamically deploy new applications in WSNs appears to be an efficient approach to address this challenge.

Earlier work which used mobile agents in the field of WSN dates back to the year 2005 when the first architecture of Mobile Agent Based Wireless Sensor Network (MAWSN) was given by Chen and his team[75,76,77,78]. The work used these special mobile entities in a planar WSN, where mobile agents were exploited at three levels (i.e. node level, task level, and combined task level).The framework employs the mobile agent's ability to carry processing codes that allow the computation and communication resources at the sensor nodes to be efficiently harnessed in an application specific fashion. Owing to their inbuilt features, mobile agents adjust their behaviors depending on quality of service needs (e.g. data delivery, latency) and the network characteristics to increase network lifetime while still meeting those quality of service needs. The authors in their subsequent works [77] have also highlighted various applications and design issues for using mobile agents in wireless senor networks. Authors identify that mobile agents in WSN offer twin-fold advantages. First of all, mobile agents would move data processing to sensed location resulting into conservation of bandwidth which otherwise would consume lot of energy of sensor nodes, Secondly, mobile agents facilitate collaborative signal and information processing resulting into flexibility of data.

Chong and his team [21] has presented recent trends in algorithms and routing in wireless adhoc networks, and distributed classification using local agents. The paper identifies that the initial applications of sensors was limited to large military systems only and the recent developments of MEMS [131] have resulted into embedded processing which in turn can be used in variety of applications.

Authors [14] have devised UbiMASS, a mobile agent system for dynamic service distribution. In UbiMASS, mobile agents offering services are loaded on sensor nodes dynamically and hence a single node could serve variety of applications.

SENMA [65], sensor networks with mobile agents is based on node redundancies that communicate opportunistically with a large field of sensors. Authors have proved that the addition of mobile agents shifts computationally intensive tasks away which in turn offer energy efficient operations. The work has been compared with a flat adhoc network and a substantial gain in energy efficiency has been observed.

Work in [22] presented a mobile agent middleware(called Agilla) that facilitates the rapid deployment of adaptive applications in WSNs. Agilla injects mobile agents in a sensor network which can then intelligently move or clone themselves to desired locations in response to changing environment. The work presents a case study for using mobile agents for fire tracking application. The inherent disadvantages of WSN like limited memory, limited provision of flexible application development forced the need of developing a mobile middleware solution for WSN applications.

A project named Contiki [6] uses code mobility. Although it is not an agent based system but it offers dynamic reprogramming of sensor nodes. The code lacks proactive behavior. Work presented in [87] presents the feasibility of employing mobile agent in wireless environments. Authors in [46,106] have leveraged mobile agent technology to study the issue of how to balance the energy consumption during data collection in WSNs. The work is motivated by the fact that there is uneven energy dissipation in the network which can be balanced using mobile agents. To achieve this energy balancing, the authors have premeditated an energy prediction strategy, which can determine the position of mobile agents to lessen the uneven energy dissipation problem. Finally, a clustering protocol called Energy Balance Cluster Routing Based on Mobile Agent(EBMA)[45] routing is proposed. In this form of routing the cluster structure is formed based on a cellular topology with the consideration of the energy balancing of inter-clusters and intra-clusters.

Work presented in [115] has focused on maximizing network lifetime of a heterogeneous Wireless Sensor Network (WSN) using mobile Data Collectors (DCs) without compromising on the reliability requirements. The notion of reliability is achieved by moving DCs via alternate routes to the sink by avoiding hotspot region formations. The work has given both a centralized and distributed solution for the problem defined.

Works by Aielloet. al [38] has considered agents for signal processing in WSN. Work presented in [41] has shown the efficiency of the multi-agent technology for WSN based structural health monitoring (SHM) applications on the large aircraft structures. Kallapur along with his co-author has described various research challenges for using mobile agents for the task of aggregation[88].

The literature presented above clearly advocates deploying mobile agents for efficient processing in sensor networks. It is evident from the works already available that in client/server-based sensor network, sensors only collect data and send to a sink node. However, the mobile agent is a new computing paradigm that offers data and code mobility. A mobile agent visits the network either periodically or on demand and performs data processing autonomously. On contrary to various advantages that a mobile agent based sensor network offer, it also contributes certain disadvantages such as code caching, safety, and security. Mobile agents are making their space in e-commerce, military situation awareness, just to list a few. Recent studies indicate that scientists have been using mobile agents for data fusion in distributed sensor networks.

A critical look at the above literature indicates that though extensive work has been done in the field of routing in WSN and mobile agents have also been introduced but none of the researchers have focused their works towards event-driven route discovery in wireless sensors deployed in non-deterministic environment. Additionally, use of MAS to deal with critical issues like formation of energy holes near the sink, finite on-boards capacity of sensor nodes and QoS constraints are still in its infancy.

Upcoming section presents an exhaustive literature survey done to exploit the potential of mobile agents done particularly in the domain of communication and information processing in non-deterministic environments

3.3 MOBILITY IN WSN

Majority of the WSN applications assume that the deployed sensor nodes are static in nature. However, certain applications of non-deterministic environment like sea exploration, wildlife protection, and traffic congestion control[16,110,122] have the need of mobile sensor nodes to be deployed in a network. In such applications, either the sink node or an ordinary sensor node(or both) can be mobile in nature. Though this mobility of sensor nodes give more accurate results in terms of security, k-clustered connectivity, sensing reliability etc. [44,105,130], it causes frequent topology changes resulting into high packet loss. For this reason, mobility becomes an important issue that must be considered in the design of WSNs.

Several network and MAC layer protocols[16,69,116] have been proposed and implemented which handle the mobility of sensor nodes. As WSN are a special class of MANETs, the AODV protocol used for routing in case of MANET was initially applied to deal with MWSNs(Mobile WSNs).However, simulation reveals that the conventional AODV routing protocol cannot perform in MWSN as good as in static WSN because the protocol is not good enough to detect broken routes and react to topology change fast enough in mobile environment.

A distributed protocol called Robust Routing Protocol has been proposed in [140] which handle the mobility issue of sensors along with other parameters. In this protocol, nodes work cooperatively to enhance the robustness of routing against path breakage due to mobile sensors. The work has exploited the Wireless Broadcast Advantage (WBA) property of the wireless medium in which the neighboring nodes of the transmitting node can overhear the packet. Because of this cooperative caching in neighborhood, the nodes are able to deal with dynamically changing topology.

As evident from the description of LEACH protocol in previous chapter, the basic LEACH(and even LEACH-C) cannot deal with mobile sensor nodes. To meet the mobility requirement of sensor nodes, a variation of LEACH known as LEACH-Mobile has been given in [110]. The protocol is based on the assumptions that the sensor network is considered to be homogenous, the initial energy of the sensors is same, every sensor knows its velocity and location, the base station is stationary and all sensors are time synchronized and each sensor node should be able to estimate the time it takes to transmit a packet. Like basic LEACH this protocol also works in two phases. However, LEACH-Mobile protocol works by confirming whether a mobile node is capable of communicating with specific cluster head within the time slot allotted in TDMA schedule. If a similar node does not respond to the request of the CH in successive time slots twice, it is considered to be removed from the cluster. The time slot of this removed cluster is then allotted to the newly arriving node. At the cost of higher energy consumption this protocol outweighs its parent protocol in terms of reduced packet loss and increased successful packet consumption.

A protocol called Cluster Based Routing Protocol for Mobile Nodes in Wireless Sensor Network(CBR-Mobile)[112] has been designed to further reduce the packet loss. This approach minimizes the effect of mobility of motes by decreasing packet loss by changing the TDMA scheduling adaptively and using round free cluster head. This protocol is also energy aware, since it reduces the energy consumption by transmitting with low transmission with minimal amount of energy power based on the received signal strength of data request message. The CBR Mobile-WSN protocol moves a step ahead in data transfer success rate and energy consumption in mobility environment in comparison to LEACH-Mobile protocol.

Mobility models are used to simulate the displacement patterns of such mobile sensor nodes. They are designed to describe the movement pattern of the nodes, and how their locations, velocity and acceleration change over time. There exist many mobility models in literature which attempt to represent realistically the behavior of mobile nodes[99]. Figure 3.1 presents the nomenclature of routing mobility models used in WSN and are being described in short in subsequent sections:

3.3.1 Entity/ Individual Models

As the name suggests, nodes in these types of models move independently of each other. Random Waypoint Model, Random Walk Model and Random Direction Model are examples of entity mobility models used in WSN and are described as follows.

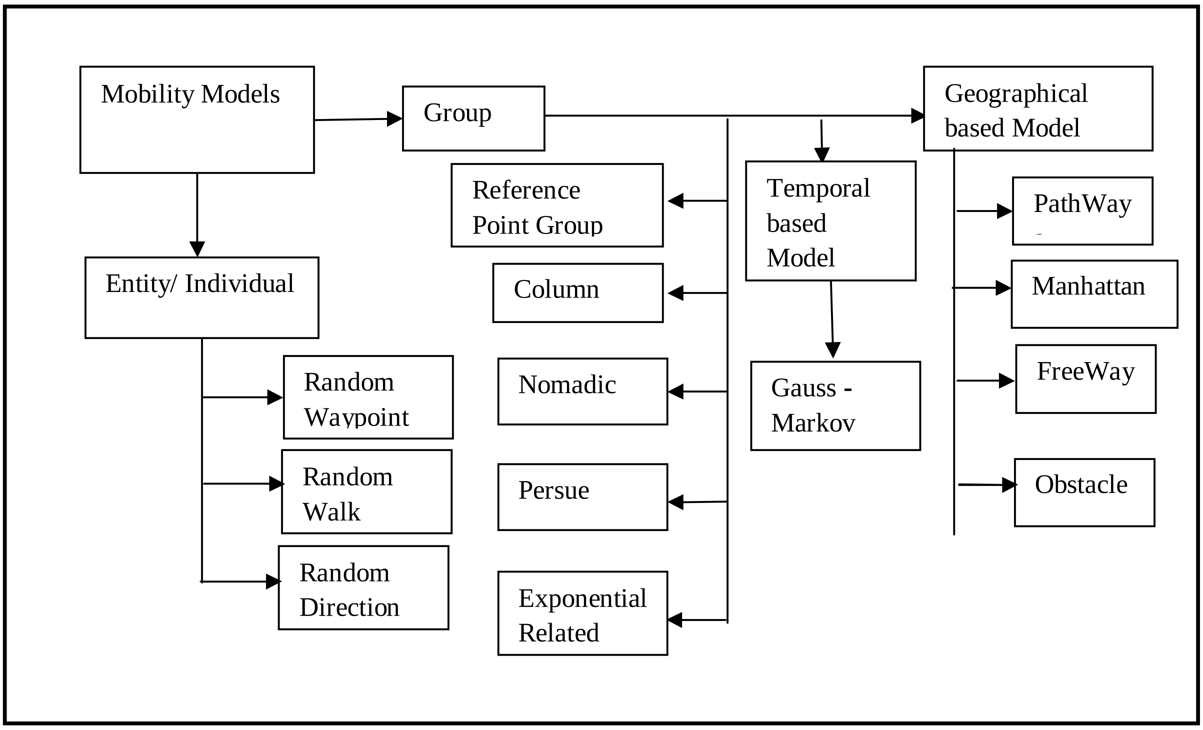

Figure 3.1 : Mobility Models in WSN

\

- ***Random Waypoint Model***

It is one of the most commonly used mobility model in WSN in which a travel path consists of a series of trips**.** In this model**,** the probability of a mobile node choosing a new destination that is located in the center of the simulation area, or a destination which requires travelling through the middle of the simulation area, is high. Hence, in RWP, nodes are clustered near the center of the simulation area, distance between nodes is shorter, performance is better. Literature survey related to simulations using RWP reveals that the model produces unexpected results for poor velocity distribution [125].

- ***Random Walk Model & Random Direction Model***

Both these models change their direction after very time slot. New Directions are chosen randomly between (0,2Π]. The speed is selected using Gaussian distribution. In this model node reaches boundary it bounces back with (π-θ) [23].

3.3.2 Group based Mobility Models

In these types of models, sensor nodes move in groups. The nodes are grouped generally either geographically or on the basis of temporal property. Also in certain models the mobile nodes are dependent on each other like Reference Point Group Model, Column Model, Nomadic Model, Persue Model and Exponential Correlated Model. Each of the categories and their sub-categories are explained below:

- ***Geographical based Models***

The movement of nodes in this category of mobility models is restricted based on the location. Common examples of geographical based mobility models [23,99] are Pathway Model, FreeWay Model, Manhattan Grid Model.

Pathway Model integrates geographic constraints into the mobility model is to restrict the node movement to the pathways in the map. The map is predefined in the simulation field which can either be randomly generated or carefully defined based on certain map of a real city. The vertices of the graph correspond to the buildings of the city, and the edges model the streets and freeways between those buildings. Initially, the nodes are placed randomly on the edge. Then for each node a destination is randomly chosen and the node moves towards it through the shortest path along the edges. FreeWay Model emulates the motion behaviour of mobile nodes in a FreeWay. The model works by restricting mobile node is restricted to its lane on the freeway. The velocity of mobile node is temporally dependent on its previous velocity. If two mobile nodes on the same freeway lane are within the safety distance (SD), the velocity of the following node cannot exceed the velocity of preceding node. Manhattan Grid Model uses the grid road topology. The mobile nodes in this model can move only in horizontal and vertical direction. At each intersection of a horizontal and a vertical edge, the mobile node can turn left, right or go straight with certain probability.

- ***Temporal based Models***

  In this model, each node is initialized with a speed and direction. Gauss –Markov Model is one such model in which after regular intervals of time movement occurs to updating the speed and direction of each node.

There are certain other models in which the mobile nodes move in groups. The Column Mobility Model represents a set of mobile nodes (e.g., robots) which can move in a certain fixed direction. This mobility model can be used in searching and scanning activity, such as destroying mines by military robots. When the mobile node is about to travel beyond the boundary of a simulation field, the movement direction is then flipped 180 degree. Thus, the mobile node is able to move towards the center of simulation field in the new direction**.** The Nomadic Mobility Model represents the mobility scenarios where a group of mobile nodes moves randomly from one location to another. It determines the reference point of each group based its general movement. This model could be applied in mobile communication in a conference or military application. In Persue model, several nodes attempt to capture single mobile node ahead. This mobility model can be used in target tracking and law enforcement. The node being pursued (target node) moves freely according to the Random Waypoint model by directing the velocity towards the position of the targeted node and the pursuer nodes (seeker nodes) try to intercept the target node.

The research work targeted various deployment approaches to determine the most commonly used simulation approach. Optimized approaches deploying minimum number of sensors are available in [27,138] and these aim to provide sufficient grid coverage of the sensor field specifically deal with the context of uncertainty in sensor locations subsequent to airdropping.

Taking into consideration the limited energy supply of the randomly dropped sensors, the authors in [146] have proposed the deployment of multiple sinks in place of one. The work has presented a mathematical model which determines the location of the sink which minimizes the sensors' average distance. For this it presents two algorithms, one which determines the location on the basis of global information about the network and the other which carries out the sink deployment based only on the location information of the neighboring nodes while the location of the distant nodes is being approximated.

An application specific approach has been given in [28, 53] where the authors have presented two deployment algorithms for underwater acoustic WSN. The work aimed to minimize the number of sensors needed to be deployed to achieve the optimal sensing and communication, coverage, which are dictated by the application.

A fuzzy optimization algorithm (FOA) [121] that efficiently adjusts the sensor placement after an initial random deployment is also available. It makes use of fuzzy logic theory to handle the uncertainty in sensor deployment problem. Literature review reveals that though mobility of a sensor unit is an inevitable requirement, it can cause dynamically changing links and unpredictable random topology. It is therefore essential to *control the mobility* of these intelligent units because reduction in mobile variance ensures smooth traffic in WSN especially when spatial correlation is high.

3.4 CLUSTERING & ITINERARY PLANNING IN WSN

Clustering is defined as the task of grouping sensors on the basis of some parameter (distance, logical organization etc.).The special sensor nodes elected as cluster head then transmit the information of its respective cluster to its immediate parent thereby reducing the network traffic[12,25,124]. Clustering reduces the number of nodes taking part in communication, ensures scalability for large number of nodes and reduces the communication overhead for both single hop and multi hop. Grouping sensor nodes into clusters has been widely pursued by the research community in order to achieve one or the other of the following objectives.

- ***Load Balancing***

  One of the major objectives of clustering is to evenly distribute the randomly deployed sensors so as to achieve load balancing. Even distribution of sensors can also leverage data delay [12]. When Cluster Heads of such equal sized clusters perform data aggregation, it is imperative that the combined data report becomes ready almost at the same time for further processing at the base-station or at the next tier in the network.

- ***Fault Tolerance***

  As the selected cluster heads are nothing but sensor nodes, they are prone to malfunction or fail. In order to prevent the loss of information due to the failure of sensor node in a given cluster, achieving fault tolerance is a desirable property especially in harsh and non-reachable environments [69]. To achieve fault tolerance dynamic clustering or (re)clustering is proposed in literature. Though dynamic clustering helps to achieve fault tolerance, it causes additional burden on nodes and distruption to the on-going operation. Therefore, back up cluster heads or rotation of CHs is done to achieve fault tolerance

- ***Increased Connectivity and Reduced Delay***

  To ensure maximum benefits as a result of clustering, inter-cluster communication is a major requirement in many applications. The goal of connectivity can be just limited to ensuring the availability of a path from every cluster head to the base-station [24] or be more restrictive by imposing a bound on the length of the path [35].

- ***Minimal Cluster Count***

  It is desired to minimize the count of elected cluster heads especially when they are resource rich nodes and there is inherent complexity in deploying such nodes. Reducing their count is also required when their size is large and their visibility is unwanted in applications like border protection, military reconnaissance and infrastructure security [82].

- ***Maximal Network Longevity***

  When CHs are richer in resources than regular sensors, it is crucial to minimize the energy for intra-cluster communication[35]. The CH in such a case should be placed as closed to the sensors. However, when CHs are regular sensors their lifetime can be extended by limiting their load.

Various clustering algorithms have been proposed in literature each of which meets one or more of the above stated objectives.

One of the most common and oldest clustering algorithm is Low Energy Adaptive Clustering Heirarchy(LEACH)[56,135].As described in previous chapter, it is a hierarchical protocol which converges in a fixed number of iterations and uses Cluster Heads to do the task of data aggregation for a given cluster. LEACH forms clusters by using a distributed algorithm, where nodes make autonomous decisions without any centralized control. When a node becomes a cluster head it broadcasts its decision and the non CH nodes then join the cluster which can be reached by them using minimum communication energy. The role of CH is being evenly distributed among sensors in order to achieve load balancing. However, the random selection of CH can at times proves to be disadvantageous also because if a low energy sensor gets selected as CH , it may die out quickly because of which robustness and life time of the network gets affected. Also the number of clusters is of uneven size with certain clusters having comparatively more sensors than others.

To overcome the limitation of unequal clusters, a protocol called Balance Cluster, Balance Energy(BCBE)has been proposed by Zhang et al.[111].This protocol extends LEACH as it also divides the whole process of clustering in rounds. The advantage of this protocol is that it requires knowing the location information of two nodes in order to locate other nodes.

Another protocol known as Hybrid Energy Efficient Distributed Clustering (HEED) [84] is a distributed clustering protocol which selects CHs from amongst the deployed sensors. It uses a hybrid value of residual energy and the node degree in order to determine the cluster among the given set of sensors. In this approach only the high residual energy nodes can become CHs. This algorithm ensures that the probability that two nodes within each other's transmission range becoming CHs is small and for a given sensor's transmission range, the probability of CH selection can be adjusted to ensure inter-CH connectivity. The disadvantage of this protocol is that since it takes several iterations to elect a CH, energy dissipation is high. It has also been observed that some CH, particularly near the sink dies because they have huge workload.

Authors [16] have proposed a Linked cluster Algorithm(LCA)that handles the mobility of nodes. This distributed algorithm forms clusters in a manner that a cluster head is usually directly connected to all nodes in its cluster, thereby maximizing network connectivity. In this algorithm, initially each node broadcasts its ID and listens to transmission of other nodes. In the next round, a node broadcast the set of neighbors that it heard from and thus every node will eventually know its 1-hop and 2-hop neighbours. A given node can becomes a CH if it has the highest ID among its neighbours or does not have the highest ID in its 1-hop neighborhood, but there exists at least one neighboring node y such that the given node is the highest ID node in the 1-hop neighbourhood of y.

Another algorithm in this category has been given by [17]. It is termed as Random Competition based Clustering (RCC)and was initially designed for MANETs but works equally good for WSNs. It applies the First Declaration Wins rule, in which any node can ‘‘govern’’ the rest of the nodes in its radio coverage if it is the first to claim being a CH. After hearing the claim which is broadcasted by the first node, neighboring nodes join its cluster as member and give up their right to be a CH. To overcome the problem of delay incurred in broadcasting and receiving a CH claim packet , a random timer and node ID number are used.

Karim et al. [68] proposed a clustering approach which takes into consideration case of fault tolerance i.e. when either the CH or an ordinary sensor node fails. It extends the Dynamic Static Clustering Protocol proposed in [17] and adds the notion of fault tolerance to it. To achieve the same the work ensures that non-CH nodes send notify messages in the time slot when there is no subscribed event to report. Similarly, the base station monitors the presence of each CH by setting a timer and sending a *"hello"*message to that CH. If the Base Station (BS) does not get the response before the timer expires it assumes that the CH has failed. In that case the BS will assign the sensor with maximum residual energy to act as CH. Simulations reveal that these extra messages consume energy far less than the data messages and thus do not cause any additional burden as compared to the parent protocol.

A Mobile Agent Based LEACH in WSN [110] has been proposed in practical. This paper exploits the autonomy and intelligence features of software agents to sense and disseminate data to the sink node. Though it results in conserving the energy of sensor nodes, agents by no means contribute for the task of clustering in this work. There exist model which allows the mobile nodes to be deployed in non-deterministic environment but restricts the mobility by keeping the constant pause time. Although this idea of constant pause time was initially implemented in order to conserve energy by avoiding thrashing but on contrary, the current aims to consider dynamic pause time and investigate the effect of same on proposed protocol.

Literature Review done till the date of listing reveals that though various approaches have been proposed for the task of clustering, the use of software agents for the same is still in its infancy. Further, no work till date has given the notion of reliability of software agents which is an essential aspect of communication specially when the data transmitted and aggregated pertains to mission critical applications.

Once the data is grouped into clusters, the next task in the series to achieve processed information is to determine the itinerary in which the mobile agent visits various sensor units in a given cluster. This work of determining the itinerary is considered as a sub-task of clustering in the domain of hierarchical WSN.

The task of itinerary planning of mobile agent can be solved by using two approaches namely Single-agent Itinerary Planning (SIP) and Multi-agent Itinerary Planning(MIP) [49,73]. As the name suggests, SIP makes use of a single agent to visit the nodes. Assigning the complete task to single agent however makes the application unscalable. For large scale WSNs, single agent data dissemination leads to problems of large delays, unbalanced load and reduced reliability.

MIP on the other hand is based on the assumption that primary itinerary design algorithms are executed at the sink which is rich in resources in terms of energy and computation.

Xu and Qi [50,100] have given static, dynamic and predictive dynamic approaches to determine the itinerary of a mobile agent for target tracking application. It provides an energy efficient and fault tolerant itinerary solution for collaborative processing in WSN. For the dynamic itinerary planning, the work makes use of information driven approach to determine the next node to be visited. The work has also compared its proposed approaches on the metrics of energy consumption, network lifetime and number of hops.

Rajagoplan along with his co-authors [108] has formulated mobile agent routing problem as multi-objective optimization problem where the authors have maximized the total detected signal energy while minimizing the energy consumption and path loss. The authors have used two famous evolutionary algorithms namely EMOCA and NSGA-II to obtain mobile agent routes.

Chen et al.[76] has given an energy efficient itinerary determination approach. It proposes separate algorithms for first node selection and rest of the nodes thereafter. The algorithm called IEMF (Itinerary Energy Minimum for First-source-selection) which extends LCF[48] by considering the estimated communication cost. Then an Itinerary Energy Minimum Algorithm (IEMA) is given, which is an iterative version of IEMF. During each iteration , IEMA selects the best node according to IEMF as the next source to visit among the remaining set of source nodes. It has been found analytically that with more iterations, the suboptimal itinerary can be progressively improved and that the major reduction in average energy consumption is achieved for the first few iterations.

Another major proposition for itinerary determination is the work by Konstantopoulos et al.[20]. The authors have proposed a Tree based Itinerary Design(TBID) algorithm that employs multiple mobile agent for data gathering task in WSNs. In this method, it is assumed that sink knows the geographical location of all the sensor nodes. It works by building a spanning forest of binary trees rooted at sink in network and calculates itineraries by post order traversal of binary trees. At the end, it assigns these itineraries to individual mobile agents. In this scheme, each mobile agent carries the pre-computed itinerary that determines the order of sensor nodes to be visited.

Work in [46] have used mobile agents for energy efficient solution to the sink hole problem in flat WSNs. The protocol, termed as Energy Balanced Mobile Agent based Data Dissemination (EBMADD) protocol has used MAs for data aggregation and calculated the itinerary for the same. The sensors in this approach are first being grouped into equiangular wedges. The sensors of each wedge are then organized into layered BFS using Dijkstra's Algorithm. The itinerary of these nodes is then computed using level order traversal of these balance trees. The work has been compared with its predecessor TBID for a given number of metrics

Wu et al.[103] gave a Genetic Algorithm (GA) for computing the itinerary of an MA that incrementally aggregates data as it visits nodes in a WSN. Though this approach has better performance than its predecessor approaches, it implies a time-expensive itinerary calculation (genetic algorithms typically start their execution with a random solution "vector," which is improved as the execution progresses), which is an open challenge for time-critical applications, e.g., in target location and tracking.

Qi gave two approaches [111] for optimal itinerary determination of mobile agents in WSN. They are LCF and GCF respectively.LCF searches for the next node with the shortest distance to the current node, while GCF searches for the next closest node to the cluster center.

The study [69,76] in the domain of routing of MA reveals that a mobile agent can be modeled as an entity of four attributes namely *identification, itinerary, data space,* and *method.* Identity refers to the identification of the mobile agent. Itinerary is the route of its migration. Data space refers to the buffer of the MA in which it carries the partially integrated result and method is the execution code carried by the agent.

3.5 FILTERING IN WSN

Filtering is defined as the task of extracting relevant information from the sensed data and removing the unwanted noise signals. The data gathered by the sensors is usually corrupted with noise [50,66,83]. This data if transferred in its raw form is bound to cause unnecessary consumption of bandwidth during the phase of data transfer and would lead to accumulation of redundant data at the cluster head/sink as the case may be. In order to prevent sensors from transmitting unwanted and unfruitful data, filtering at the source nodes is necessary. Filtering is one form of information processing (the other being fusion, which is discussed in next section)

Filtering is desirable, especially in non-deterministic environments in order to smooth out the fluctuations that otherwise would shorten the lifespan of sensors**.** Moreover, since, sensors are constrained in terms of energy, therefore, in order to be efficiently benefited from such systems, information that is actually useful shall only be routed so as to avoid useless energy drain and reduced lifespan of WSN.

To the best of our knowledge, the concept of filtering of data at sensor nodes till the time of listing has been explored by various researchers mainly to prevent the false data injection attacks. Young along with his co-authors[52] has given a commutative cipher-based en-route filtering scheme (CCEF) which establishes a secure session between a sink node and a cluster head (CH) based on the commutative cipher. However, due to the probabilistic approach, it is difficult to adapt to the change of false traffic ratio in the network which may in turn lead to energy inefficiency

Moon & Cho[121] have given a deterministic approach to assign filtering nodes to a given session. This approach is an enhancement over CCEF in terms of energy and also does not sacrifice security. Sheybani gave a special wavelet-based approach to suppress the effect of noise and data order[34].

An in-depth evaluation of the above section indicates that researchers have been putting efforts to meet the challenges in information processing but most of them have remained silent towards the application of Kalman filter in this domain. The thesis contributes a unique application of Extended Kalman Filtering (EKF)[10,32] technique for processing such sensitive information because sensor readings are usually imprecise due to strong variations in environment and also, computation has to be much more energy efficient than communication. Out of the various filtering algorithms available, we have chosen to apply Kalman filter, primarily because it works well both in theory and practice and moreover, it is able to minimize the variance of estimation error i.e. filters noise from the actual signal more accurately.

3.6 DATA FUSION IN WSN

Data Fusion is defined as the process of gathering data from multiple nodes and aggregating it based on certain decision criterion [8,62,114,132,145]. This approach aims to process the information at intermediate levels before sending it to the sink/base station etc by exploiting data-correlation and employing in-network processing. It is a formal framework in which are expressed the means and tools for the alliance of data originating from different sources. Fusion aims at obtaining information of greater quality. It can be divided into two parts as shown in Figure 3.2.

- **Value Fusion:** In value fusion, each sensor sends its measurements to the cluster head, which makes the detection decision based on the received measurements. Value fusion usually has better detection performance than decision fusion.

- **Decision Fusion:** In decision fusion, each sensor makes a local decision based on its measurements and sends its decision to the cluster head, which makes a system decision according to the local decisions.

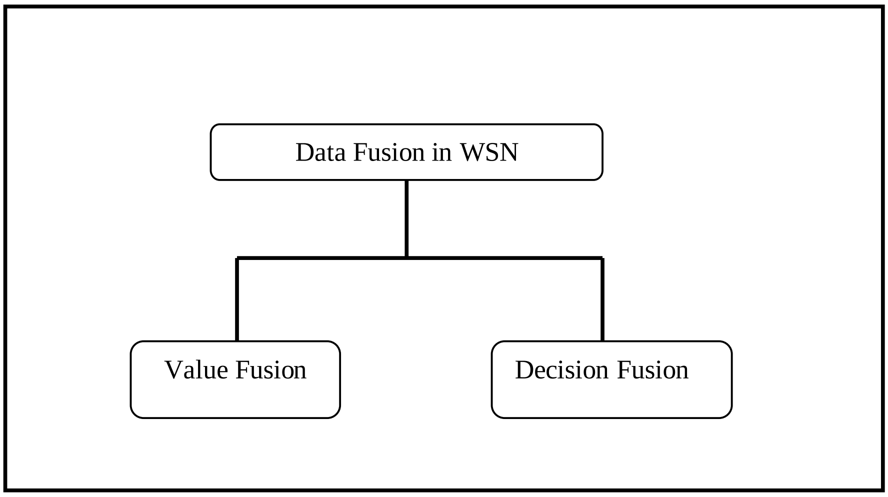

Fig 3.2 : Types of Data Fusion in WSN

Data fusion techniques thus play a crucial role in non-deterministic WSN by reducing the traffic of sensor networks as well as by conserving the energy of the sensors. Though data fusion was achieved even without using mobile agents, the feature of autonomy of mobile agents has compelled them to become an active candidate for data fusion in Wireless Sensor Networks (WSN).Both the computing models(i.e. client/server computing model & mobile-agent based computing model) perform the task of collaborative information processing in sensor networks. They however differ primarily in two aspects: what is transferred over the network and where has the fusion taken place. The authors have explored the domain of data fusion in WSN both with and without mobile agents.

Sung[137] has used BPN technology multi-sensors data fusion in a wireless sensor networks (WSNs) system with a node-sink mobile network structure.

Xing et al. [47,129] in his research work highlights the fact that one of the performance measure characteristic for wireless sensor networks is sensing coverage, which characterizes how well a sensing field is monitored by a network. Although advanced collaborative signal processing algorithms have been adopted by many existing WSNs for efficiently and accurately detecting the target, most of the analytical studies on sensing coverage are conducted based on overly simplistic sensing models (e.g., the disc model) that do not capture the stochastic(random) nature of sensing In their work, attempt is made to bridge this gap by exploring the fundamental limits of coverage based on stochastic data fusion models that fuse noisy measurements of multiple sensors. The work emphasis on the fact that the earlier fusion work had not been taking noise among sensors and thus the false alarm rate into consideration It thus proposes and implements the probabilistic disc model for data fusion (or collaborative processing).Data fusion is shown to significantly improve sensing coverage by exploiting the collaboration among sensors.

Researchers [79] describe the agent oriented programming paradigm for development of intelligent sensor networks. The case study is performed by implementing a test bed using JADE, Java Agent Development Framework, as a basis of testing this proposed approach. It makes use of data fusion for gathering information. However, as the network considered is decentralized, the fusion process takes place without the need of a central fusion node.

Luo et al.[51] in their work have emphasized the fact that by exploring data correlation and employing in-network processing, redundancy among sensed data can be curtailed and hence the network load can be reduced. The objective of sensor routing algorithms is then to jointly explore the data structure and network topology to provide the optimal strategy for data gathering with as minimum energy as possible. However, though in-network data fusion for information processing in WSN can reduce data redundancy and hence curtail network load, the fusion process itself may introduce significant energy consumption for emerging wireless sensor networks. For certain applications, fusion costs indeed are comparable to those of communications. Therefore, fusion-driven routing protocols for sensor networks cannot optimize over communication cost only, the fusion cost must also be accounted for.

Biswas along with his co-authors[89] have proposed a multi-layered, middleware-driven, agent-based architecture for supporting mobile-agent-based collaborative processing in sensor networks. The work has given an energy-efficient, fault-tolerant approach for fusion among multiple sensor nodes using a mobile-agent-based computing model and compared the performance of mobile agent-based approach with that of the traditional client/server-based computing model on the basis of energy consumption and execution time. Though there are certain overheads associated with the use of mobile agents, yet mobile-agent-based collaborative fusion is found to be advantageous over client/server based model due to the amount of sensor nodes deployed and the degree of collaboration needed.

Misra and Thomasinous [118] have given a simple, least-time, energy-efficient routing protocol with one-level data aggregation (LEO) that ensures increased life time for the network. The proposed protocol outperformed many popular adhoc and sensor network routing protocols such as AODV, DSR, DSDV,DD in throughput, latency, average energy consumption and average network lifetime. Besides the success of LEO, challenges such as mobility, security, reliability and fault tolerance still need to be resolved.

Zou et al. [143] have proposed decision fusion rules based on multi-bit knowledge of local sensors. Their work emphasis that fusion rules based on the commonly used weight cannot improve the system performance even with a large number and high SNR.

Rani**[**82**]** called *Tree Based Itinerary Design*(TBID) in which the given area is partitioned around the processing element into concentric zones and MA paths are built from the inner (close to PE) zones to the outer ones. It then makes use of post order traversal for itinerary planning.

Patil and his co-authors[121] have exploited the concept of collaborative signal detection in a hierarchical network to perform serial and parallel fusion at various levels of the network.

The work given in [136] used Ant colony Optimization(ACO) approach to perform the task of aggregation. It works by constructing the aggregation tree by the accumulation of pheromone. After a short transitory period, the amount of pheromone on the aggregation nodes becomes large enough to guide the data packets from different sources (like ants) to meet together at these nodes for data aggregation. The proposed approach has better results in terms of energy consumption than its traditional counterparts like DD.

A critical look at the above literature indicates that though extensive work has been done in the field of data fusion in WSN and mobile agents have also been introduced but none of the researchers have focused their works towards event-driven route discovery in wireless sensors deployed in non-deterministic environment. Moreover, the existing routing algorithms are not effective in supporting the dynamic characteristics of wireless sensor networks (WSNs) and cannot ensure sufficient quality of service in WSN applications [36,77,119]. Therefore, the need of technical solution is apparent and the related work cited above ensures that one of the implicit underlying paradigms is that of multi-agent systems (MAS) and MAS is a promising candidate solution.

3.5 CONCLUSION

A critical look at the above literature points out the need to bridge the gap between existing technologies and current needs. Therefore the need of the hour is to propose an energy efficient solution ranging from the deployment to transmitting the processed data to the sink for visualization purposes. Although a lot of work has been done pertaining to the field of sensor networks, but most of the researchers have remained silent towards a specific solution which can cater the needs of an event driven of WSN in a given non-deterministic environment. In addition to the challenge of controlling the mobility of sensor nodes in such an environment, energy aware clustering and effective information processing in the form of fusion and filtering of data of such an expensive system shall also be considered apriori. The subsequent chapters aim to provide solutions to the issues highlighted in this chapter and hence meet the stated objectives.

Chapter IV

A NOVEL APPROACH FOR COMMUNICATION AND CLUSTERING OF SENSORS USING MOBILE AGENTS IN A NON-DETERMINISTIC AREA

4.1 INTRODUCTION

Literature review presented in previous chapter disclosed that developments in WSN specifically deployed in non-deterministic environments have been a hot area of research. Scientists have been looking for inter-disciplinary solutions to achieve the energy efficient solutions as routing in WSN comes with many design challenges such as efficient deployment of sensors, optimal clustering, ensuring connectivity and coverage metrics, and so on. It is also desired that a good communication strategy would prolong the lifetime of network by avoiding sink hole problem. Also, security of data transmitted is another challenge that should be considered while designing a communication strategy.

In order to counter the pressing need of a good communication strategy in sensor networks, the idea of amalgamating mobile agents in sensor networks is being conceived and a strategy that could address the above stated limitations is being proposed. The proposed communication not only controls the communication among sensor nodes by employing modified random way point model but also offers unique clustering strategy followed by filtering and fusion. The proposed strategy has been named as Mobility Controlled Communication, Clustering, Filtering and Fusion (MC3F2) [95] and hence forth we would refer the same as MC3F2 only.MC3F2 executes in four phases as shown in Figure 4.1.

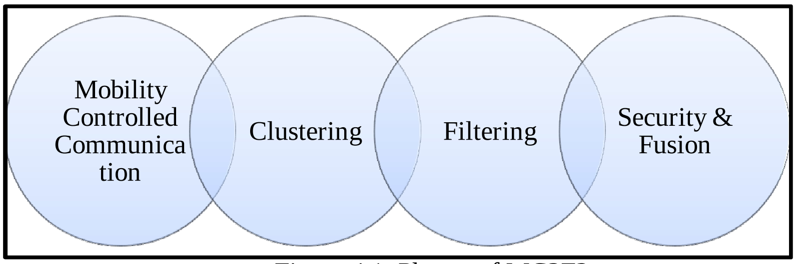

Figure 4.1: Phases of MC3F2

MC3F2 provides a complete solution pertaining to sensor nodes deployed in non-deterministic environment. MC3F2 significantly contributes various algorithms that efficiently controls the mobility of nodes deployed in an unattended environment. It also proved that a clustered network with filtering and fusion performed at source nodes is an effective and an efficient solution for an energy constrained network. During the entire study it was found that assisting a sensor network with mobile agents is a new area of interest [59, 65, 77] and is gaining importance. Mobile agents travel through the sensor network either periodically or when an event occurs, collecting the data from each hop. It may also process the data as directed. MC3F2 is a novel framework which deploys agents to reduce the overhead of sensor nodes. To the best of our knowledge only few researchers [122,133.142] have been focusing on employing intelligent entities in WSN. Mobile agent based routing protocols in WSN [59,78] emphasize that agents can be delegated the task of collecting data from nodes, processing the same and delivering to sink. These protocols advocate that agents can prove to be substantial supporter to improve the efficiency of a WSN. Moreover, the concept of injection of mobile agents introduces fluidity of code and state in the network, thereby making WSN capable of running several autonomous applications at a time [77]**.** In fact, since an agent based framework comprises of multiple agents where each agents is usually designed to be autonomous, pro-active, rational, self-adaptive and most importantly cooperative, the overall framework thus formed is usually an intelligent framework. The inter-disciplinary framework although intelligent but it also pretence various challenges ranging from the route determination to quality and quantity of data to be transmitted so that the overall impact remains positive on the energy-efficacy of WSN. Therefore, a solution which could offer low energy consumption, high accuracy and fewer migration hops is being highly desired and hence MC3F2.

4.2 ABSTRACT VIEW OF MC3F2

Figure 4.2 depicts the abstract view of MC3F2. The detailed working of various components of MC3F2 is presented later in the chapter. As shown in Figure 4.2, a random number of sensors are deployed in a non-deterministic environment. The sensor nodes thus deployed are possessed with mobile agents and are known as Intelligent Sensing Units (ISU). The mobility of ISU is controlled using modified random way point model [95]. ISUs now form the clusters using the strategy defined in agent based clustering algorithm (ABC) [95] and an agent with maximal residual energy (E_{res}) and highest reliability value (RV) is elected as cluster head. The E_{res} and RV values are calculated using EBMADD protocol [46] and RCNTEP protocol [4] respectively.

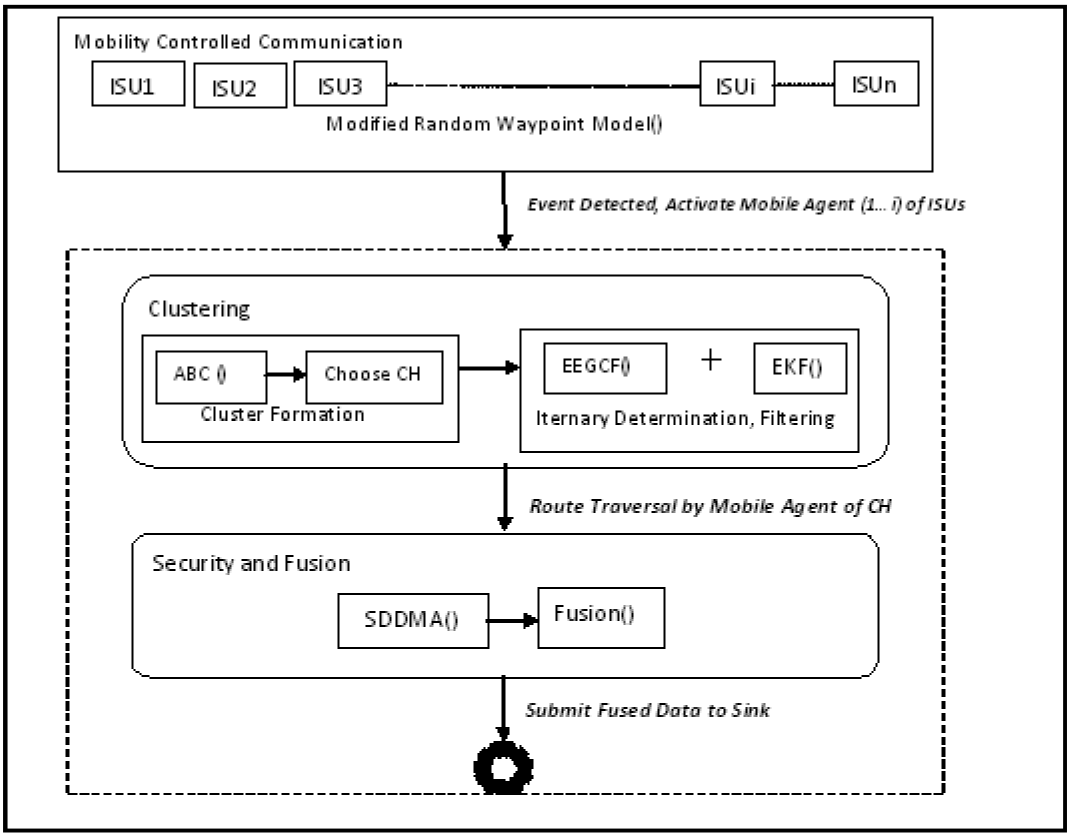

Figure 4.2 : Abstract View of MC3F2

Energy Efficient Global Common Factor (EEGCF) [91] is an itinerary determent approach that decides the route a mobile agent must traverse while visiting various nodes in the network and performing the task of data collection. Further, in order to avoid transferring the noisy data, filtering of data is carried out individually by each source node using EKF (Extended Kalman Filter) [10,32]. The filtered information is encrypted and decrypted using algorithm SDDMA [134], hence the security factor has also been taken into consideration. The processed information is then fused using [92,93] and transferred to the cluster head. This complete cycle of constrained communication, clustering, filtering, securing and fusing the information results in considerable saving of bandwidth in a resource constrained network of sensors. In fact, an overall improvement in the lifetime of the network has been observed as described later. The upcoming section discusses various phases and the associated algorithms in detail.

4.2.1 Mobility Controlled Communication Phase

A lot of research has been carried out to find the feasibility of implementing the idea under focus. It is evident in order to overcome the constraints such as lifetime of network, deployment strategy, communication protocols, just to list a few, various mechanisms have already been proposed and implemented ranging from initial deployment to finally processing the information received. For instance, Random Waypoint Model (RWP) [23,99] is one of the most common mobility models being used. Though this basic model works well in most of the situations, results get fluctuated when the distance between two nodes and the speed at which the nodes move do not match with each other. The proposed deployment scheme takes the advantage of above listed fact.

Basically, this phase puts forward an energy efficient means for communication among the randomly deployed sensors in the nondeterministic area. While deploying the sensors in an unattended environment, an underlying network model is considered and various assumptions pertaining to the same are listed as follows:

- Sensor nodes are heterogeneous.
- The energy consumption is not uniform for all the nodes.
- All the links are symmetric
- Nodes are not uniformly distributed and remain unattended once they have been deployed

On the basis of above assumptions, initially nodes are being deployed using modified random way point mobility model. The basic model of RWP assumes that the sensor nodes are mobile and they either move individually or in groups. In either of the cases, their constant pause time becomes the major bottleneck in the practical implementation of the existing model. In light of this limitation, we propose that the mobile nodes shall function as almost static sensor nodes by making the pause time between the sensors random instead of constant. It is noteworthy that during the practical implementation of the work, the interval of random pause time is found to be greater than constant pause time. Further, the task of mobility is achieved with the help of mobile agents embedded within them. The work thereafter simulates the aforesaid proposition in MATLAB. The network topology generated using random pause time is given in Figure4.3. A comparison of both existing and new deployment strategy revealed that significant energy could be conserved by avoiding unnecessary movements of mobile nodes and also, since mobile agents also travelled less frequently(random pause time is greater than constant pause time), a lot of energy could be saved. As can be seen from results shown in Figure4.4(a) and Figure 4.4(b) respectively, with the increase in the number of sensor nodes there is steep fall in the residual energy (energy dissipated) in case of Basic RWP . This fall is however gradual in case of Modified RWP.

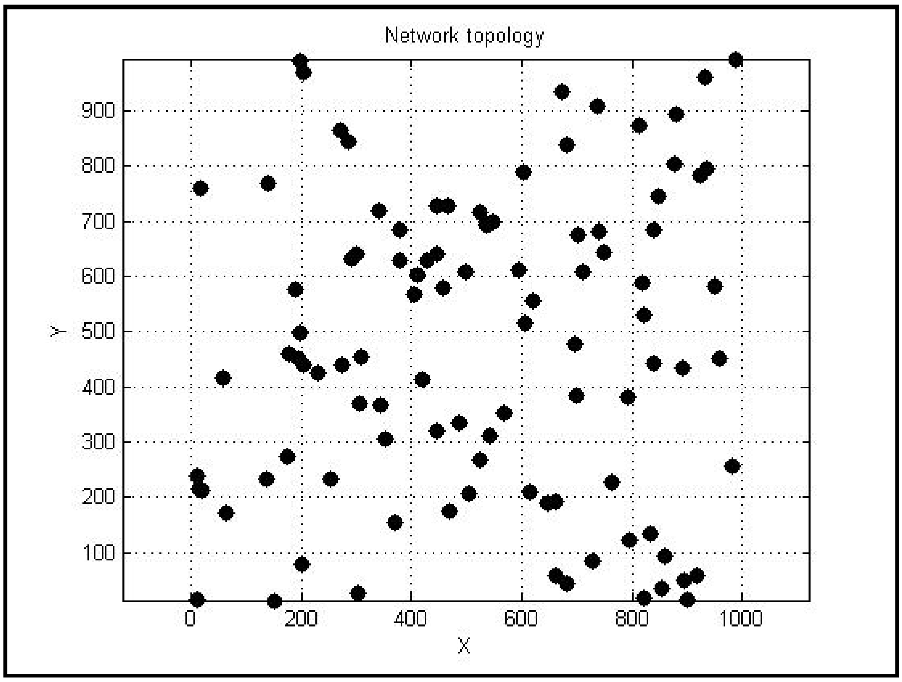

Figure 4.3: Network Topology Generated

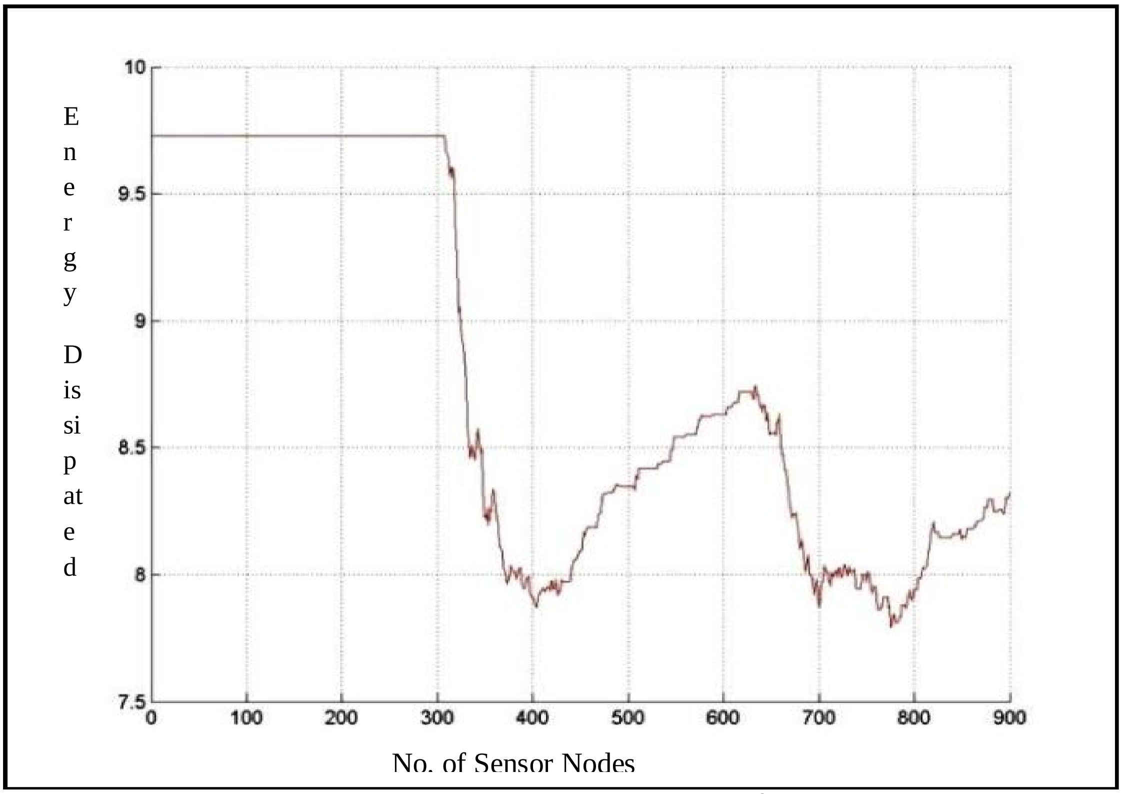

Figure 4.4(a) : Energy Dissipation due to Modified RWP

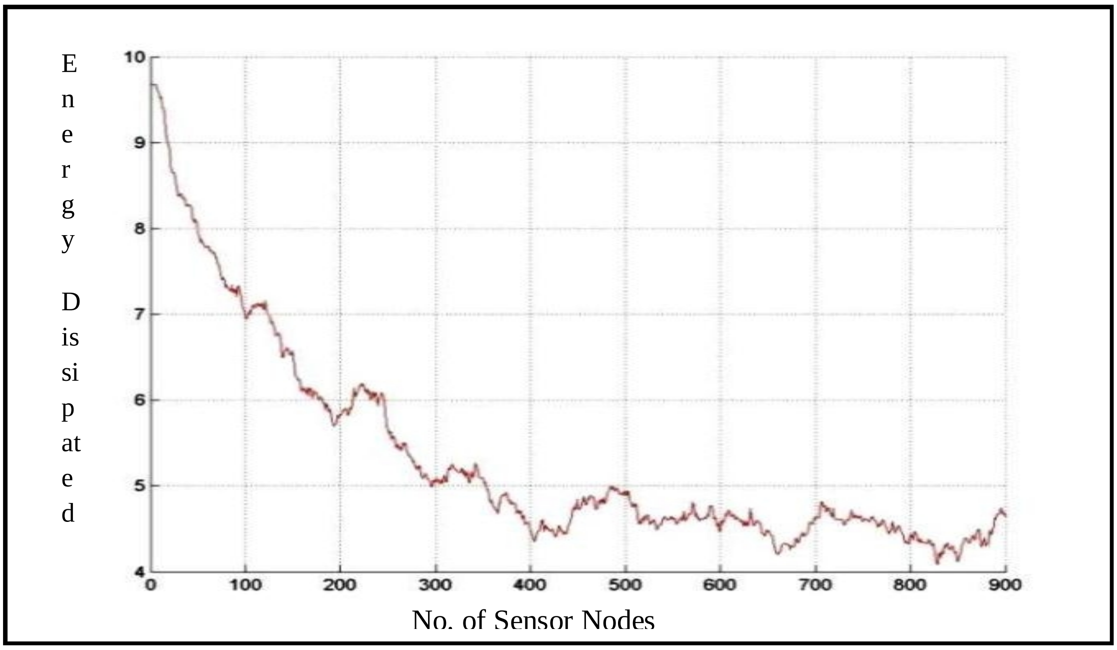

Figure 4.4(b) : Energy Dissipation due to Basic RWP

4.2.2 Clustering Phase

Routing and gathering the sensed information is an expensive task especially when number of nodes in a given network is enormous. In such cases, clustering becomes inevitable. Though researchers have exploited this segment of routing in WSN, use of mobile agents for the same is still in its infancy. The authors have proposed an agent based clustering approach which is well suited to event driven applications especially in a non-deterministic area. Formation of clusters may appear to be an uncomplicated task, but during the practical implementation, the clustering phase encountered the design challenges listed below.

- Mapping of all nodes to atleast one cluster is a sufficient condition. Although, cluster head is only responsible for upward transmission of information but there may exist nodes which do not belong to any cluster and intend to transmit the information.

- Designing a clustering approach with limited complexity and less intercommunication cost is another design challenge in existing clustering approaches.

- Mapping of each node with atmost one cluster head is a necessary and sufficient condition. Handling the redundant links will remain one of the primary challenges.

- In order to have efficient clustering strategy, it should be distributed and energy efficient.

- The current phase is an extension of our work Agent Based Event Driven Route Discovery Protocol (AERDP) [94](explained in the upcoming subsection) and hence adding more functionality to the same may increase the complexity of the overall network.

- The major requirement is to maintain the existing complexity of AERDP by proposing an efficient clustering strategy which not only reduces the interagent communication cost but also the processing times at each node.

Before explaining the protocols associated with clustering phase, it is important to elaborate AERDP.

4.2.2.1 Agent Based Event Driven Route Discovery Protocol (AERDP)

This section presents an agent-based event driven route discovery protocol (AERDP). The work is motivated by the fact that various sensor nodes in the same vicinity often come across spatial contention i.e. will sense the same event and would transmit the same information. Taking into consideration the battery constraints of sensors, it is desirable that only the necessary subset of all nodes (observing the same event)shall transmit the data and rest should remain in an active mode. Further, with time the density of sensor nodes in a particular geographical domain varies. Therefore, *the four major reasons that motivate this work are event-driven nature of sensor network, spatially correlated contention, need of event-reporting by a subset of nodes and variation in the density of sensor nodes.*

A) High Level View of AERDP

AERDP is an event-driven protocol which presents a hierarchical solution in a clustered sensor network. Clustering is achieved using Agent Based Clustering (ABC) protocol and is being discussed in the upcoming section. The high-level view of the proposed work is depicted in Figure 4.5.

B) Working of AERDP

As proposed in [90], the fundamental backbone remains same i.e. ISUs have been deployed. These agents can adopt the role of a master agent and the child agent. The sensor network in an environment is divided into various clusters and each cluster can elect a monitor (head) to which all other nodes within the same cluster report about the events happening in the vicinity. Each node is now assisted with two mobile agents which are basically replicas of each other having potential to run concurrently and independently. Further, the agent assisting the monitor is termed as *Master Agent* and rest of the agents in a cluster are termed as *Child Agent.* Basically, rather than each child agent directly reporting to the sink, it reports to its monitor's agent about the event.

The master agent is now responsible for the discovery of route for gathering the information from other monitors. The master agent on detecting an event of interest initially time-stamps itself and the event and then immediately sends a replica of its own with a message '*call for any event*' to the immediate neighbors. This originating master agent i.e. the one with the oldest timestamp acts as the root and is responsible for sending the data to sink node. The listening nodes i.e. master agents of neighboring clusters further generate a similar message for their next immediate neighbors. Within a particular periphery, each master agent fuses the data received from its child agents and other neighboring master agents and forwards the same to the ancestor master agent who had actually originated the call.

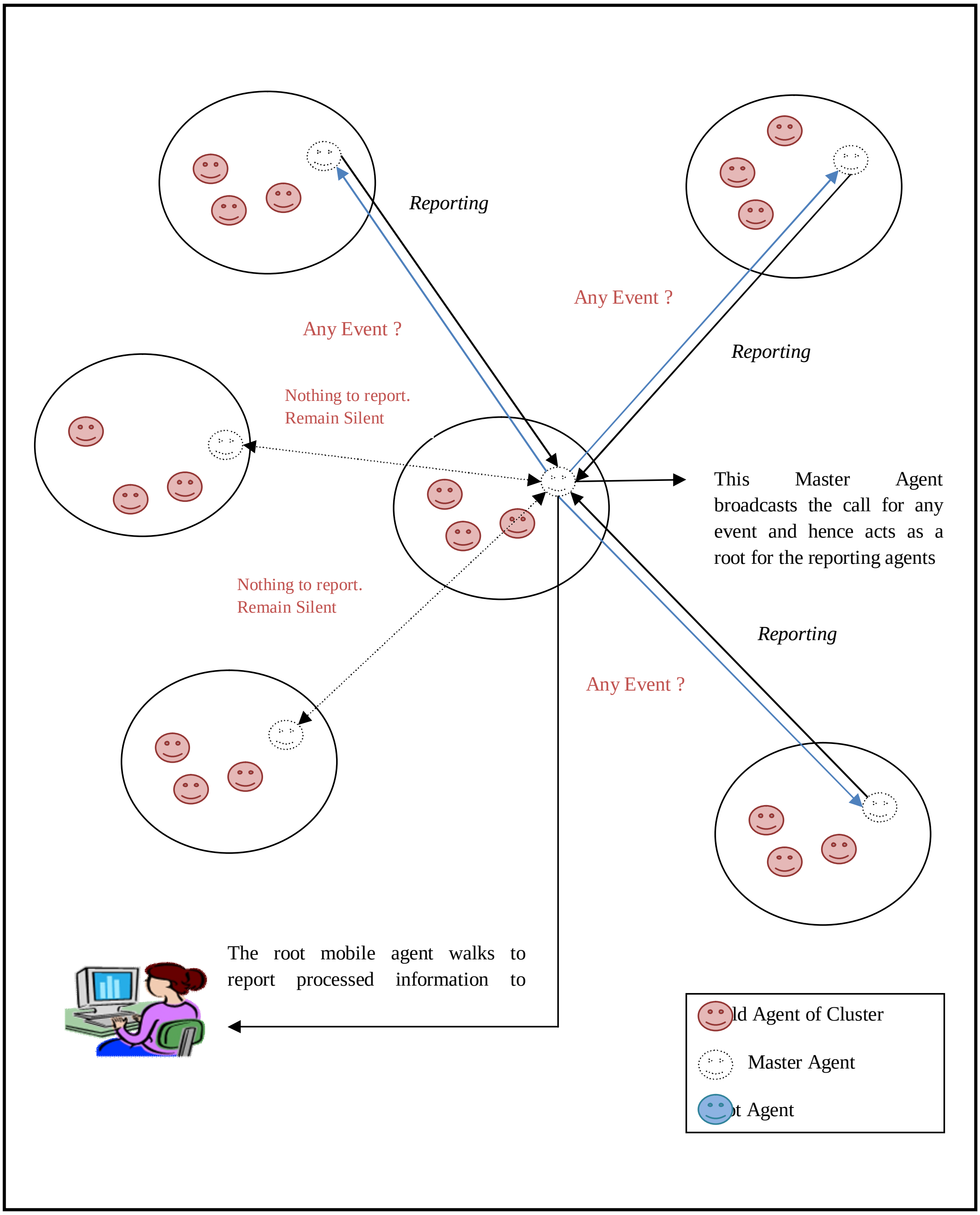

Figure 4.5 : High Level View of AERDP

In contrast, if the neighboring master agent has nothing to report, it remains in silent/sleep mode and the ancestor assumes that there is no further information that is to be collected. In contrast to Agilla [22], which addresses the agents by their geographic location, AERDP addresses agents by their ids as nodes deployed in a non-deterministic environment are prone to drift away from their locations and further a mobile agent would move from one location to another, it would then be difficult to identify the parent node of migrating agent. In order to facilitate inter-agent coordination, AERDP supports both global and local name space. The local name space maintains the list of all agents, the agent id and their responsibilities within the same cluster whereas the global name space maintains the list of all immediate monitors and the assisting mobile agents and their replicas. Now, since only master agent can interact with master agents of different clusters, therefore global name space is usually referred by other master agents while local name space is meant for child agents. Both the namespaces are remotely accessible. This procedure of detecting the event is being gradually carried out in the clusters pertaining to the area where the event has occurred. As shown in Figure 4.6, each master agent in its own cluster operates in three different modes, i.e. *observation mode, negotiation mode and execution mode.* In observation mode, it senses the event from the environment and refers the same to its global and local name space to gather the information about the location and ids of it one-hop neighbors & communicates its own requirements to its immediate neighbors.

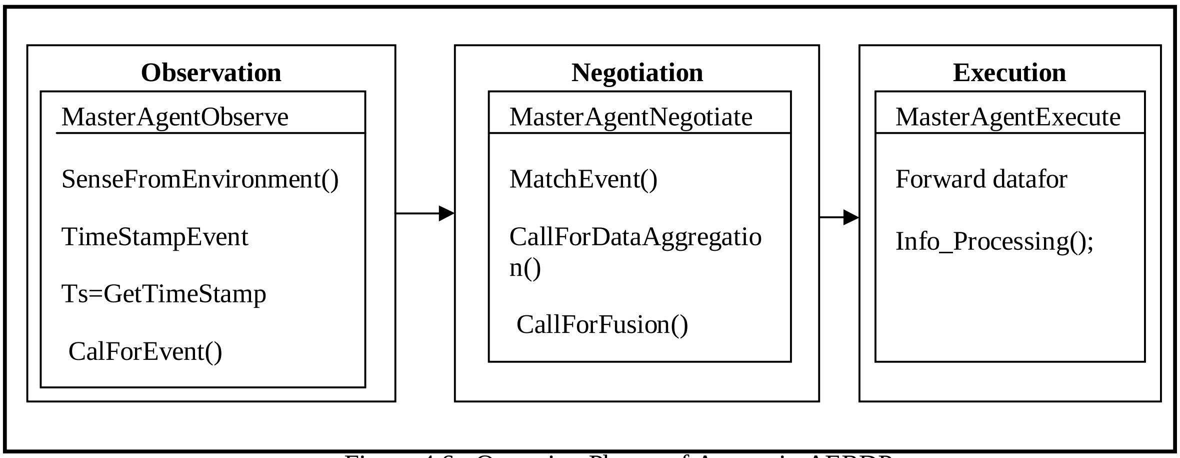

Figure 4.6 : Operating Phases of Agents in AERDP

The output of this mode of the agent is being fed to peer agents. For the observed event, the master agent negotiates the requirements with the child agents and its peer agents. The later phase is termed as negotiation mode. In this mode, master agent gathers the information from all its peers and is responsible for data aggregation and data fusion. The issues related to data aggregation and data fusion has been considered in the fourth and the last phase of MC3F2.In the third phase i.e. execution mode, all agents on receiving the request from the ancestor agent execute the request in the pre-determined interval of time otherwise no information available is presumed. Since their aim is to generate a response to the query by the ancestor, these are said to be functioning in the *execution mode***.** Working algorithm of the observation phase of AERDP and its flowchart are shown in Figure 4.7 and Figure 4.8 respectively.

```
Algorithm : AERDP()
            Input :Event in the environment
            Output : Fused Data to the sink
Begin
          MasterAgentObservation()
{Timestamp ts;
If (Action=SenseFromEnvironment)
                    { Activate MasterAgent;
Ts=GetTimeStamp(Event);
CallForEvent();}
If (EventbyOtherAgent!={})
                { EventsSensed=EventbyOtherAgent∪ SenseFromEnvironment;
MatchEventName(EventsSensed);}
If (EventbyOtherAgent SenseFromEnvironment!={})
               { AggregatedData=DataAggregation();
FusedData=DataFusion(AggregatedData);
ForwardtoSink(FusedData);}
End;
```

Figure 4.7 : Working Algorithm of AERDP

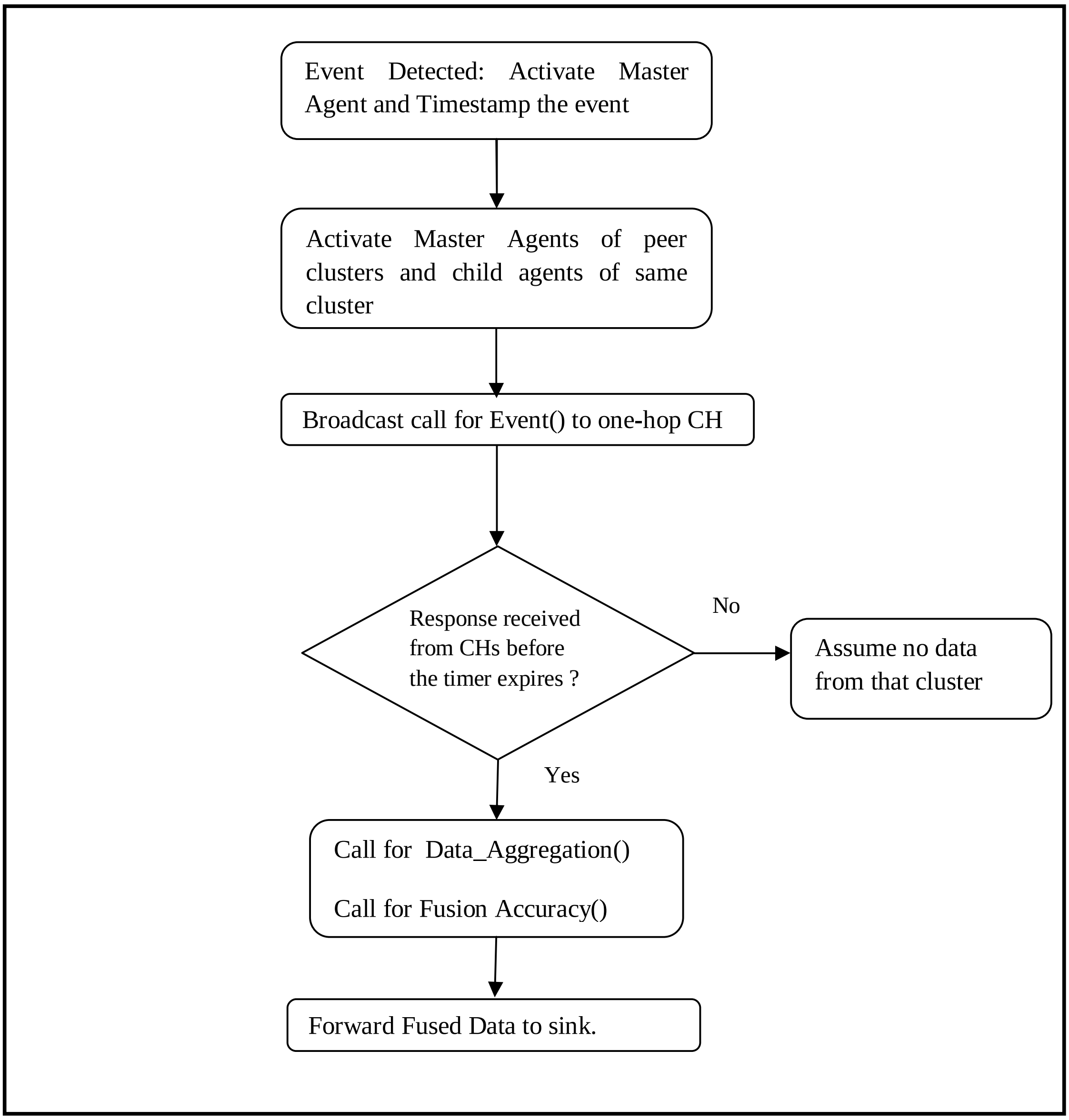

Figure 4.8: Flowchart of AERDP

AERDP fully exploits the clustering approach and the potential of agents.Now, since, AERDP is already employing mobile agents for collecting and transmitting the information, clustering would also be carried out by mobile agents, thereby alleviating the sensors from the burden of processing. In fact, since the proposed clustering approach (Phase 2 of MC3F2) is an addition in the basic AERDP, therefore it is also event-driven. It has been termed as *Agent Based Clustering approach*.

4.2.2.2 Agent Based Clustering (ABC)

In order to meet the above stated challenges, the following assumptions have been made:

- Each node knows its position and residual energy level. A node can obtain its location information with the help of GPS or any other Localisation System[97], which is already available due to the need of sensor net applications.

- The link between the nodes is bi-directional i.e. if a node can hear from a neighbour, then its transmission range can reach it.

ABC approach for clustering runs every time an event is detected by an ISU. When an ISU detects an event, it initiates its corresponding agent to form cluster. This mobile agent(termed as initiator agent) considering itself to be a tentative cluster head generates a call for proposal(*cfp*) for all the ISUs which are within its communication range of its ISU. The communication range of a particular sensor is calculated using equation (4.1)

$$\boldsymbol{c = 2 \times r} \tag{4.1}$$

$$where\ \ c = communication\, range\ of\ ISU,$$

$$r = sensing\ range\ of\ ISU$$

Each ISU which receives the *cfp*, respond with its residual energy and reliability value through its embedded mobile agent. The residual energy at any interval(say t) is given by equation (4.2)

$$\boldsymbol{E_{residual} = E_{maximum} - \left(E_{sensing} + E_{processing} + E_{transmission}\right)} \tag{4.2}$$

The reliability value(RV) of an agent refers to its credibility to perform an assigned task. When two or more agents compete for a similar task, the one with the higher RV value is chosen to perform that specific task. The RV value of an agent has been calculated using RCNTEP[4].The initiator agent logs both these values from every responding sensor and compares its residual energy with each of the incoming residual energy values logged in as :$\boldsymbol{E_{residual[1....n]}}$

Three possibilities arise:

a) The residual energy value of the initiator agent is larger than any of the respondent's residual energy. In such a case, the initiator agent considers its sensor to be a Cluster Head and broadcasts an "*election_won*" message to all the neighbors for which it generated a *cfp*.

b) The residual energy value of the initiator agent is less than any of the respondent's residual energy. In such a case, the initiator agent determines the sensor with the maximum energy and communicates it to carry out the task of clustering and generate a *cfp* for the same.

c) The residual energy value of the initiator agent is equal to any of the respondent's residual energy. This situation normally arises when clustering is being carried out for the first time in the network. In such a case, every sensor has maximum energyi.e. $E_{residual} = E_{max}$. The initiator agent in this case thus uses the RV value of an agent to determine the cluster head.

Figure 4.9 and Figure 4.10 depicts the working algorithm and flowchart of ABC.

Algorithm :Form_clusters()

Input :

- **n : no. of nodes in the network**
- **Eresidual[1…..n] : Residual energies of all the nodes in the network**
- ***RVagent[1……...n] : Reiability value of the software agents of all the sensors in the network***

Output : A Clustered Network

Begin

. Detect the event

For (i=1; i<=n; i++) // n is the no. of sensors which detected the event

***a**. Sensor_Agent[i] = Tentative CH*

***b**. Sensor_Agent[i] broadcasts its sensing range(r) value to rest of the n-1 sensors in the network and ask for their E_{res} value based on the equation : c = 2r*

***c**. Sensor_Agent[i] asks for Reliability value of the agent of each responding sensor*

***d**. Log in the incoming E_{res} value and Reliability Value of the sensors (say j sensors, where j<=n) and compare it with the E_{res} values of the parent/receiver sensor*

For (k=1; k<=j; k++)

If any(E_{res} (k)) >E_{res} (parent sensor)

cluster_head = k

else if E_{res} (k) <Eres (parent sensor)

cluster_head = parent sensor

else if Eres (j) = Eres (parent sensor) = Emax

cluster_head = sensor whose agent has max(RV)

Figure 4.9 : Working Algorithm of ABC

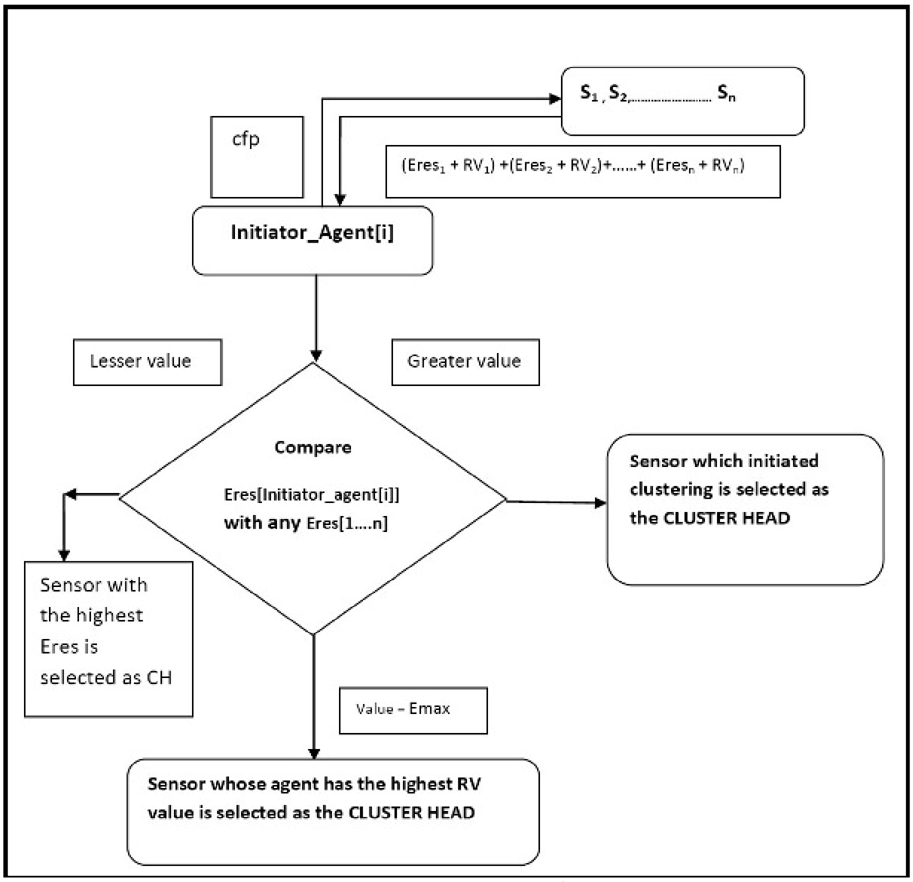

Figure 4.10 : Flowchart of ABC

ABC protocol was simulated using MATLAB and the snapshots of the simulation of agent based clustering is shown in Figure 4.11(a) and Figure 4.11(b).

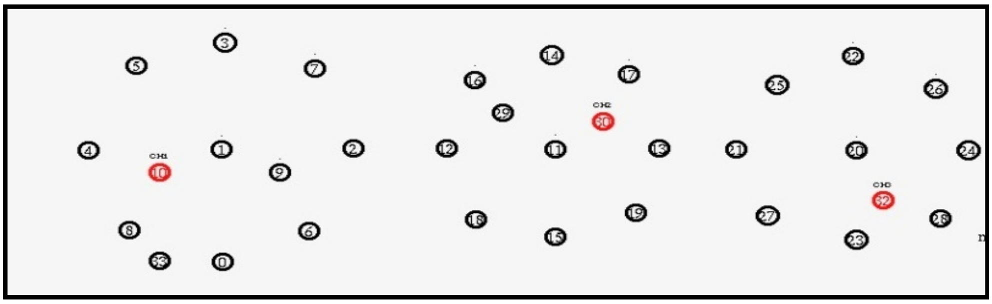

Figure 4.11(a): Randomly Deployed Clusters with Elected Cluster Heads

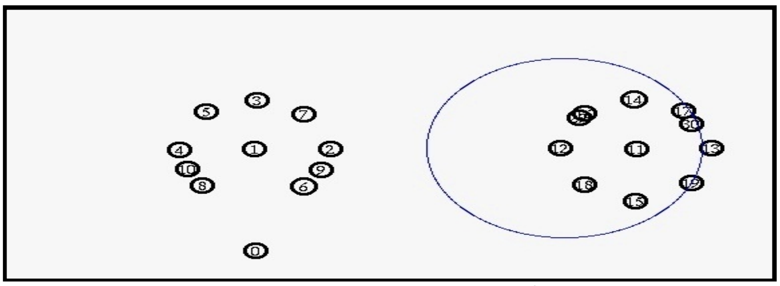

Figure 4.11 (b) : Intermediate Phase of Clustering

Once the ISUs get grouped into optimal number of clusters using the ABC approach, the next challenge is to route the mobile agent of the cluster head to visit its ISUs within the cluster. For this the authors have exploited the cooperative nature of mobile agents and put forward an agent driven approach for itinerary determination.The work extends the concept of Global Closest First(GCF)[49] to suggest an energy efficient approach ,termed as Energy Efficient Global Closest First(EEGCF) . The succeeding sections explain the working of EEGCF in detail.

4.2.2.3Itinerary Determination using Energy Efficient Global Closest First(EEGCF)

As stated in previous chapter, GCF or Global Closest First looks for the next node closest to the sink.The main drawback of this algorithm is that it computes the next node to be visited by considering only spatial distance between the nodes[20]. As a result, it produces messier routes because of repeated computations. The work overcomes this drawback by adding the concept of residual energy of the ISU to be considered in addition to the spatial distance. Once the ISUs get clustered, each cluster head becomes responsible for gathering and disseminating the data of various ISUs in the cluster. The cluster head achieves this task by dispatching an agent on occurrence of a particular event. It is significant here that the work caters event driven application of sensors in particular.

When an event occurs, it is being sensed by, say, n sensors in a given cluster. Whichever sensor detects the event first i.e. a CH or a regular sensor it can initiate the event of itinerary determination so that the agent can be dispatched from the CH to perform data fusion. Consider the case when an event is first detected by some members of a cluster. It is reasonable that a particular event is detected by more than one sensor because they are randomly deployed and can be very close to each other. They dispatch the “hello” msg to the cluster head indicating that they are awake and have sensed the event. The CH on receiving the message from k (k<=n) sensors concludes that it has to route the agent to these k sensors only for aggregation and fusion. It thus clones its agent to all these k sensors to gather their residual energy and maintains a table of the same. To route the agent, it takes the help of GCF approach. If more than one sensor is found to be suited for routing, agent is routed to the one having less residual energy.

A) Working Algorithm and State Diagram

Figure 4.12 and Figure 4.13 presents the algorithm and state diagram of EEGCF respectively.

Algorithm : EEGCF()

Input :

- ***a cluster of a network with n ISUs***
- ***a special node(out of these n ISUs) acting as CH***

Output: an energy efficient itinerary of a mobile agent

Algorithm at the sensor's end

For (i=1;i<=k;i++) // k(<=n)is the no. of sensors which detect the event

Begin

1. *Timestamp the event detected*
2. *Initiate the child agent and communicate a hello msg to master agent located at CH*
3. *delay()*
4. *Respond with the details of residual energy to the cluster head*
5. *Sleep()*

End

Algorithm at the CH end

Begin

1. *Log in the values of k ISUs*
2. *Dispatch a clone to each of the k ISUs to inquire about their residual energy*
3. *Log in their Residual energy.*
4. *Use GCF to determine the closest agent to the cluster head out of these k ISUs*
5. *If more than one ISUs is equidistant from the CH then*
    a. *Route the agent to the sensor which is having relatively less residual energy to fuse the data.*
6. *Repeat steps 1-5 for all the k ISUs*

End

Figure 4.12 : Algorithm for Itinerary Determination

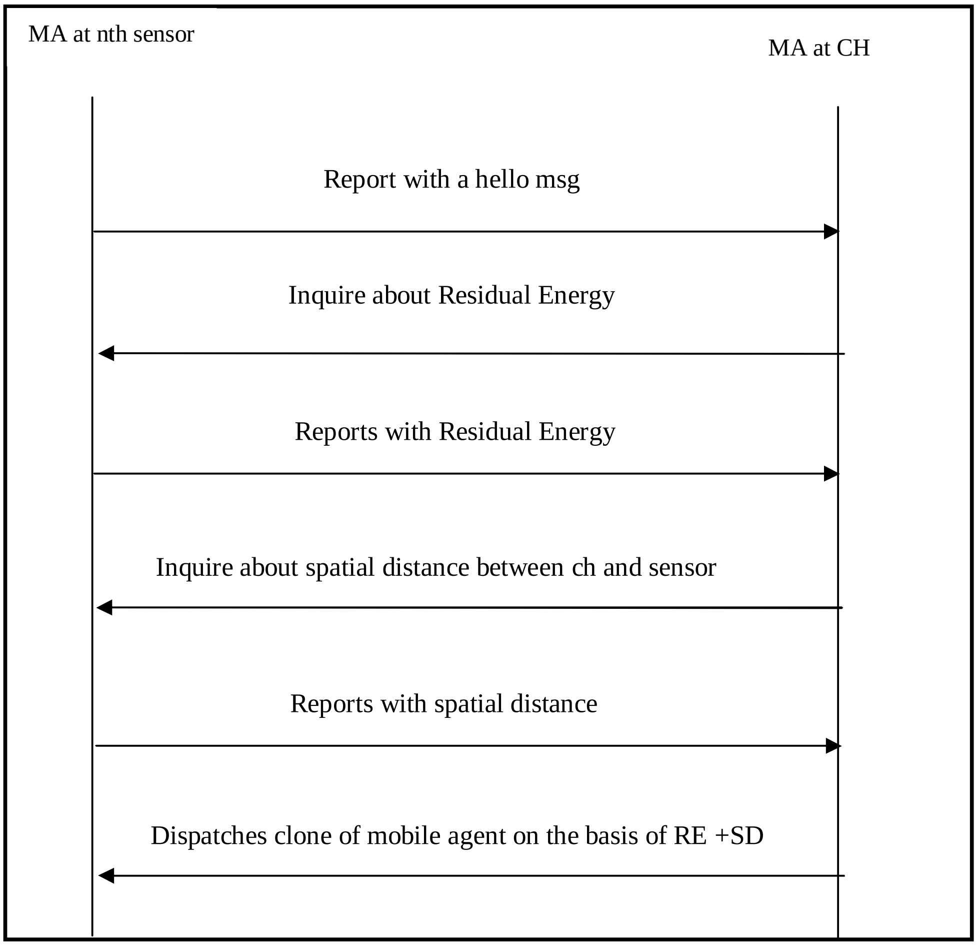

Figure 4.13 : State Diagram of EEGCF

The proposed EEGCF approach has been implemented and is being compared with its successor algorithm i.e. GCF. Table 4.1lists the possible differences between the two.

Table 4.1 : Comparison of GCF and EEGCF

GCF	**EEGCF**
Uses only spatial distance to determine the next node to be visited	Uses both spatial distance and residual energy to determine the next node to be visited
Produces messier routes in a large or dense network	Does not produce messier routes
Involves simple comparison for determining the next hop	Involves relatively complex comparisons for determining the next hop

The comparison study has also being carried out with one of its peer agent based protocol i.e. TBID(Tree Based Itinerary Design)[49,73]. It takes into consideration two parameters namely *energy consumption* and *network lifetime* to evaluate the performance of the protocols. Before comparing the protocols, these parameters are briefly described below:

- ***Energy Consumption***

In any mobile agent based data dissemination protocol, energy consumption at a given node (say i) is the sum total of the energy consumed by an agent for data aggregation purpose, energy spent to receive the mobile agent and the energy spent to transmit it to any other node or the sink as the case may be[49]. Representing the same analytically, it can be said that

$$\boldsymbol{E_{Total_i} = E_{RMA} + E_{agg_i} + E_{TMA}} \tag{4.3}$$

$here \quad E_{Total_i} = total\ enegy\ consumed\ at\ a\ node\ i$

$E_{RMA} = energy\ spent\ to\ receive\ a\ mobile\ agent$

$E_{agg_i} = energy\ spent\ for\ aggregating\ the\ data\ at\ nodei$

$E_{TMA} = energy\ spent\ to\ transmit\ a\ mobile\ agent$

Normally, $E_{agg_i}$is constant for all source nodes visited by the mobile agent. $E_{TMA}$and E_{RMA}depends on size of mobile agent received and transmit respectively.

- ***Network Lifetime***

The network lifetime (NL) of a sensor network is defined as the period of time until the first node dies due to energy depletion.

Comparison Results

Simulation results show that EEGCF outperforms TBID especially when the network size grows. This is because length of itinerary of mobile agent is relatively shorter in EEGCF as compared to TBID .

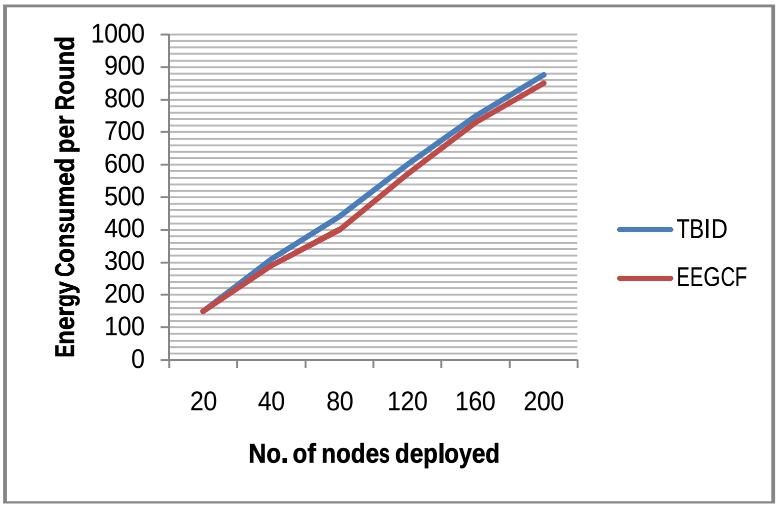

Figure 4.14 : Energy consumption Per Round (TBID vs EEGCF)

As stated above, another parameter being analysed for both the protocols is network lifetime. EEGCF gives better results for this parameter also. The reason is that in TBID, mobile agent takes forward and backward move to complete its itinerary, resulting in unnecessary energy consumption. Because of this 2-way movement, energy consumption among nodes is not balanced. However, in EEGCF, mobile agent does not take backward movement. Also, it visits those sensors first which have less residual energy so that it can gather maximum data before network failure.

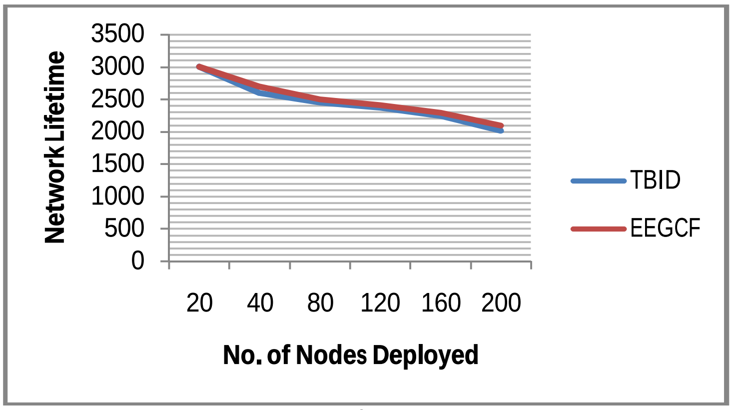

Figure4.15: Network Lifetime (TBID vs EEGCF)

Considering the lifetime of the network as the prime factor in a given non-deterministic area, the work has given an energy efficient approach for wireless motes. The work has been analysed analytically and performs better than its counterpart for the given set of parameters.

4.3 CONCLUSION

The chapter began by justifying the need of proposed strategy MC3F2. MC3F2 comprised of four phases and only first two phases i.e. Mobility controlled communication and clustering had been explained in the chapter. The two phases uniquely contributed efficient deployment and communication approach, and a clustering protocol along with the itinerary determination strategy. Both phases were implemented separately and the results achieved outperforms their competitive counterparts, hence the work made a significant contribution. Remaining two phases pertaining to information processing is being presented in the next chapter.

Chapter V

A NOVEL APPROACH FOR INFORMATION PROCESSING USING FILTERING AND DATA FUSION IN A NON-DETERMINISTIC AREA

5.1 INTRODUCTION

The chapter uniquely contributes algorithms pertaining to information processing as bulk of data which is being sensed comes with loads of noise resulting into very less useful information. For instance, data sensed by various pollution measuring sensors is often corrupted and contains ambiguous information. Additionally, lot of energy of sensors is being used in transmitting this data which actually is of no use directly unless filtered. Also, in order to conserve the energy of sensors and increase the lifetime of network, the requirement to filter the sensed data at source level is highly desired. Section 5.2 presents a solution for the same.

Once the filtered data is available with the sensors, it is ready to be submitted to the sink for further analysis. However, this data is raw and redundant as sensors are deployed in close vicinity. Transmission of such redundant data is futile at the cost of bandwidth of the network. Meticulous study done in this area reveals that till the time of listing no agent based solution has been proposed to the above stated setback. For that reason, the authors have come up with a novel approach of fusing the data at intermediate nodes using mobile agents as discussed in section 5.3. This phase intelligently fuses the data within the cluster once an event is detected by the cluster. Further, security of data being transmitted is another challenge. Hence, we further propose to encrypt the data during transmission and the same may be decrypted by mobile agent while performing the task of fusion. The proposed architecture is henceforth evaluated in section 5.4.

5.2 FILTERING PHASE

Filtering is defined as the task of extracting the actual sensor readings from effects of acquisition noise, channel noise, fading etc. Filtering is desirable in WSNs in order to smooth out such fluctuations that would otherwise lead to energy drain and shorten the lifespan of sensors. In order to execute this phase efficiently, study was carried out and it was discovered that Extended Kalman Filter (EKF) [10,32] is the most suitable tool to filter the sensed data. In fact, EKF filter is an extension of traditional Kalman Filter and it has always served as an important tool when it comes to modelling a system with unknown precise nature especially to model a non-linear environment. It can be used to estimate the state of system efficiently with minimum mean of the squared error. The prime intent of this phase is to develop a mathematical model using EKF for filtering the data sensed by pollution sensors, in particular. The mathematical model presented in the upcoming subsections has been tested and simulated and results obtained have shown remarkable gap between actually measured values and transmitted values

Now, since EKF is an extension of KF, therefore an overview of the same is being provided in the section presented next.

5.2.1 Kalman Filter and Extended Kalman Filter: An Overview

The Kalman Filter [9,110] is a mathematical model which describes equations to estimate the state of a process in a recursive computational manner. Named after Rudolf E. Kálmán, the great success of Kalman filter is due to its small computational requirement and elegant recursive properties. Out of the all the available filters, Kalman Filter [132] is found to be the most suitable filter in the context of current research work as it works well both in theory and practice. It is a linear optimal filtering approach which has been often used in embedded control systems for estimating the accurate range of process variables. This filtering approach is extremely efficient for minimizing the mean of squared error, estimations of past, present and future states.

This section aims to provide an insight to application of Kalman Filter for filtering noise within data sensed by WSN deployed in a given deterministic area. The basic Kalman filter demands that the process to be measured shall be such that it can be described as a linear system. A linear system can be expressed by a difference equation and a measurement equation as given by equation (5.1) and equation (5.2) respectively.

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{5.1}$$

$$z_k = Hx_{k+1} + v_k. \tag{5.2}$$

Here,

x : the state of the system

A : $n \times n$ matrix relates the state at the previous time step k –1 to the state at the current step k, in the absence of either a driving function or process noise. Note that in practice, A, might change with each time step, but here we assume it is constant.

B : $n \times 1$ matrix relates the optional control input u to the state x.

H : $m \times n$ matrix in the measurement equation relates the state to the measurement $\boldsymbol{z_k}$. In practice, H might change with each time step or measurement, but here we assume it is constant.

k : time index

u : a known input to the system

z : measured output

w : process noise

v : measurement noise

Given these two equations, we calculate z, which is a function of x that is corrupted by noise v as we cannot measure x directly. However, it is noteworthy here that z can only be used to obtain an estimate of x as the information in z is also corrupted by noise. Also, the above two equations indicates that Kalman Filter is recursive in nature i.e. it primarily estimates the state of the process at some arbitrary time and the feedback in the form of noise is observed. It is evident that using Kalman filter, one is able to estimate the state of a discrete-time controlled process, which is governed by a linear stochastic difference equation.

Although KF is a widely used method for filtering in linear systems due to its simplicity, optimality, tractability and robustness [9,109], it is not suited to non-linear ones. For non-linear sytems, a variant of KF called Extended Kalman Filter (EKF) [10] is being used. It converts a non-linear system into a linear model so that conventional KF can be applied. Let us assume that the process under consideration is a non-linear process and for a state vector *x*, the process is now governed by the two non-linear stochastic difference equations given by equation 5.3 and equation 5.4 known as time-update and measurement-update equations respectively.

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, w_k) \tag{5.3}$$

$$\hat{z}_{k+1} = h(\hat{x}(k+1), v_k) \tag{5.4}$$

where, the random variables w and v again represent the process and measurement noise. f and h are two non-linear functions where former relates the state at the previous time step to the state at the current time step and later relates the state x_k to the measurement z_k. Here, matrices A, B and H (as mentioned in equations above) are partial derivatives of functions f and q respectively. In practice, of course one does not know the individual values of the noise w_k and v_k at each time step. However, one can approximate the state and measurement vector without them as given in equation 5.5 and equation 5.6 respectively.

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, 0) \tag{5.5}$$

$$\hat{z}_{k+1} = h(\hat{x}(k+1), 0) \tag{5.6}$$

where, $\hat{x}_{k+1}$ is some a posteriori estimate of the state (from a previous time step k).

The current research work has considered estimation of air pollutants and as the relationship between parameters such as volume of air flow, mass concentration of various pollutants in a particular vicinity etc., is non-linear, this work aims to apply an

EKF that linearizes the estimation around the current estimate using the partial derivatives of the process and measurement functions to compute estimates even in the face of non-linear relationships The proposed work employs EKF technique for processing information sensed by sensor motes because sensor readings are usually imprecise due to strong variations in environment and also, computation has to be equally energy efficient as that of communication. Moreover, it is able to minimize the variance of estimation error i.e. filters noise from the actual signal more accurately.

5.2.2 Problem Statement & Proposed Mathematical Model

The problem of air pollution is not confined to one particular locality or region but is a world wide spread problem. The section aims to initially present the existing approach for measuring pollution in air and later proposes the filtering model (based on EKF) to filter the noise so as to get accurate information. As concluded from the literature review [60], the five main air pollutants that are usually estimated are sulphur dioxide, carbon monoxide, oxides of nitrogen and oxidants. Further, it is stated that methods for sampling and measuring air pollution should be chosen carefully. However, all methods including very sensitive ones are prone to measuring noise. Therefore filtering of this undesired information is strongly desired.

Now, since measuring all pollutants and further filtering is out of scope of this work, therefore the current work only considers estimating suspended particulates in air. Suspended particulates are a composite group of substances i.e. liquids or solids which are dispersed in the atmosphere. These substances can have severe effect on an individual's health and vision. At present, the gravimetric principles are used which involves selection of suspended particulate matter often for a period of 24 hours.

Analytically, equation (5.7) is used to compute the mass concentration of suspended particulate:

$$sp = \frac{w_f - w_i}{v_m} \times 10^{6} \tag{5.7}$$

where

sp : $mass\ concentration\ of\ suspended\ particulates\ in$ µg/ms;

wf : $final\ weight\ of\ filters\ in\ g/ms$;

wi : $initial\ weight\ of\ filters\ in\ g/ms$;

vm : $total\ volume\ of\ air\ sampled\ in\ ms$;

At any time interval$i+1$, sp_{i+1} represents the mass concentration of suspended particulates measured w.r.t. sp_i measured at time i. In fact, equation (5.7) represents the difference equation while equation (5.8) is known as measurement error as specified on the basis of EKF for representing a non-linear state of a system:

$$sp_{i+1} = Asp_i + Bu_i + w_i \tag{5.8}$$

$$y_i = Csp_i + z_i \tag{5.9}$$

where,

wi : $process\ sampling\ error$

ui : $\frac{w_f - w_i}{v_m}$, $Input\ to\ System$

zi : $measurement\ error$

A : n x n *matrix that relates the state of* spi *at the previous time step i to the state at the current step* spi+1, *i+1, in the absence of either a driving function or process noise*

B : n x l $matrix\ that\ relates\ the\ optional\ control\ input\ u\ to\ the\ state$

C : m x n matrix in the measurement equation relates the state to the measurement y .

yi : $the\ measured\ oputput$

Similarly, standard air volume or air volume at standard conditions is given by equation (5.10):

$$v_s = v_m \left(\frac{p_a - p_m}{760}\ X\ \frac{298}{t_a}\right) \tag{5.10}$$

where,

v_s : $air\ volume\ at\ std\ conditions$

v_m : $air\ volume\ being\ measure$d

p_a : $pressure\ drop\ at\ inlet\ of\ positive\ displacement\ mtr$

p_m : $barometric\ pressure\ at\ atmospheric\ conditions$

t_a : $absolute\ temperature\ of\ ambient\ air$

Similar to equation (5.8) and equation (5.9), equation (5.10) can be represented by equation (5.11) and equation (5.12) given as follows:

$$v_{s_{i+1}} = Av_{s_i} + BI_i + \widetilde{v_{s_i}} \tag{5.11}$$

$$y_i = Cv_{s_i} + z_i \tag{5.12}$$

Here ,I_i is the known input to the equipment measured in terms of air volume, pressure and absolute temperature. $\widetilde{v_{s_i}}$ is the process error.

5.2.3 Implementation & Results

This system has the capability to measure and filter information about various hazardous substances but measuring and using all of them could increase the complexity of the problem under focus. Therefore the scope of work has been limited to suspended particulates. Above proposed environment was simulated and the results produced using sigmaplot are shown in Figures 5.1(a) and Figure 5.1(b).

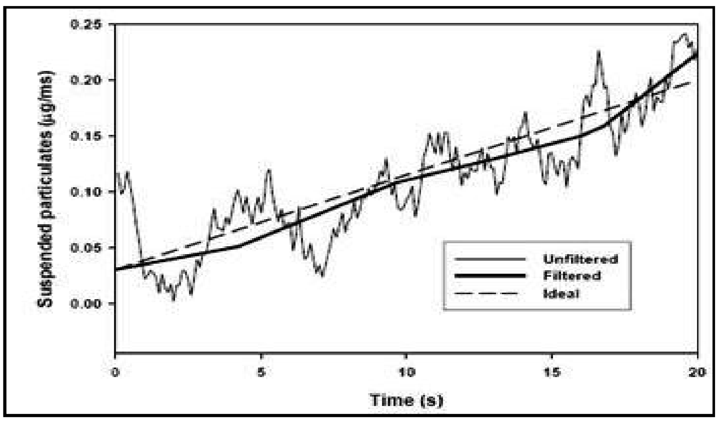

Figure 5.1(a) : Variation of Measured Suspended Particulates with time

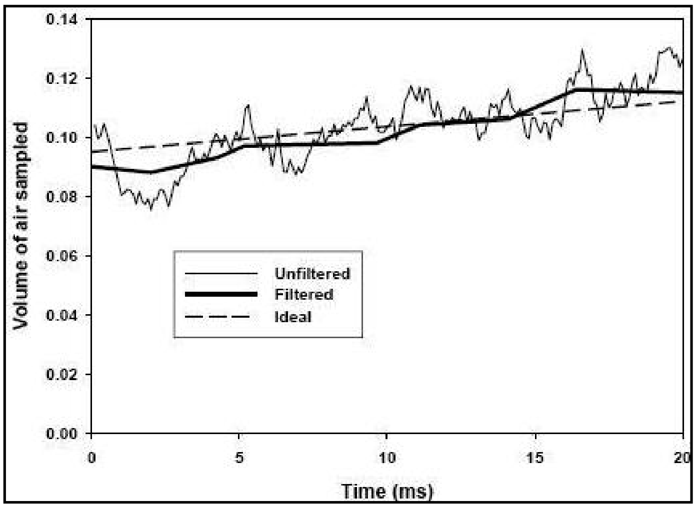

Figure 5.1(b) : Variation of Air Volume Measured at Standard Conditions with Time

It is clear from the graphs that unfiltered signals (shown by thin black solid lines in both the Figures presented above) varies randomly and significantly from the ideal condition (shown as dashed lines). The proposed filtering strategy is able to reduce the noise significantly and results obtained are competitive enough with ideal condition.

5.3 FUSION PHASE

Data fusion deals with collaborative in-network processing and gathers relatively accurate information about the events in the environment. Conventional data fusion algorithms when assisted with mobile agent technology shifts computationally intensive tasks to these intelligent units thereby increasing the lifetime of the network. There exists mobile agent based event driven protocols for accumulating and forwarding the information to the sink (base station) for example, Tree-based Itinerary Design [82,102]. Usually, such protocols either deploy value-based fusion or decision-based fusion but very few are using both at the same time. Moreover, use of multi-agent systems in such protocols is still in its infancy. The focus of this work is thus to propose a multi-agent hybrid protocol [93] exploiting the benefits of both value and decision fusion by performing aggregation at the source level in a clustered WSN. Moreover, when a typical WSN is being deployed in a non-deterministic area, energy is the main constraint of this network due to the fact that each sensor node has limited battery power and it cannot be recharged. So it is necessary to incorporate energy awareness into every stage of the network design and operation. In order to maximize the lifetime of sensor networks, the system needs aggressive energy optimization techniques, ensuring that energy awareness is incorporated not only into individual sensor nodes but also into groups of cooperating nodes and into an entire sensor network. Data fusion techniques thus play a very important role by reducing the traffic of sensor networks as well as by conserving the energy of the sensors.

Fusion in WSN is done in order to exploit data-correlation and employ in-network processing. There exist two basic data fusion schemes, namely, decision fusion and value fusion [48,85]. In decision fusion, each sensor makes a local decision based on its measurements and sends its decision to the cluster head, which makes a system decision according to the local decisions. In value fusion, each sensor sends its measurements to the cluster head, which makes the detection decision based on the received measurements. Value fusion usually has better detection performance than decision fusion.

5.3.1 The Proposed Fusion Algorithms

This section proposes two approaches pertaining to fusion. It begins with presenting an algorithm for removing the redundant data from the filtered information [93] and a hybrid approach for data fusion exploiting the significant characteristics of fusion at the cluster level[94].The details for the data aggregation focussing on removal of redundant data are presented below.

5.3.1.1 Data Aggregation

This section proposes an agent-based protocol for data aggregation in non-deterministic WSN. The proposed protocol performs aggregation to eliminate redundant data before getting transferred to the sink/base station, in particular. It applies a novel data-centric approach to replace the traditional address-centric approach in data forwarding. It uses AERDP [95] as the base protocol which is already defined in the last chapter. The sensor network for this protocol is divided into various clusters and each cluster can elect a monitor (head) to which all other nodes within the same cluster report about the events happening in the vicinity. Each node is assisted with two mobile agents which are basically replicas of each other having potential to run concurrently and independently. Further, the agent assisting the monitor is termed as Master Agent and rest of the agents in a cluster are termed as Child Agents. Basically, rather than each child agent directly reporting to the sink, reports to its monitor's agent about the event. The various operating modes of AERDP have already been elucidated in the last chapter. The task of data aggregation which takes place in negotiation phase) is being discussed in the following subsection.

A. Negotiation Mode of AERDP

As stated earlier, this mode is responsible for data aggregation at the cluster head(s) i.e. at all the master agents of the sensor nodes (cluster heads) which are being intimated by the monitor node to perform aggregation at the cluster level. For this purpose, the master agent (agent at the cluster head) aggregates the data sequentially. The agent, however, before adding the data in its packet, compares the timestamp of the event being sensed & brought to it by the child agent with its own. If the timestamp is found to be older, data is being rejected. This task of comparison is being done for all the data packets which come to it. From the second submission onwards, the master agent in addition to checking the timestamp also compares the data bits of the child agent with the data it has already collected. If the data is redundant, it is discarded; else it is added to the data packet. In this manner, the monitor agent collects data only from first 'n'(i.e. a specified) sensors reporting to it. Data of only a subset of nodes (sensing the same event) is considered because the sensors are correlated spatially.

The rest of the sensors in the cluster remain in the dormant state, thereby increasing the lifetime of the network. This task of gathering the information in the respective cluster is being done at all the peer clusters of the root node. In addition to gather the data from its child agent(s), each master agent also act as a parent for its immediate one-hop neighbor thereby maintaining the hierarchy for collecting the data.

Once the data at all the clusters has been aggregated (using value fusion), it needs to be further forwarded to the agent of the root node which actually sensed the event. At this place decision fusion is used to arrive at a consensus by fusing all the decisions submitted by master agents of various cluster head and report the same to the sink/base station.

In the observation phase of AERDP, when the event is being sensed by a particular sensor, it is communicated to its agent. The agent then timestamps this event and negotiates with its peers in order to determine the root node (i.e. one with the oldest timestamp). All the sensors which detect the event then form their respective clusters. Using ABC clusters are formed, the agent of the root node instructs all its one-hop neighbors to gather information from within their clusters within the given time & also communicate the same to other sensors in their vicinity. This phase of AERDP thus explores the fact that sensors are spatially correlated and a series of multiple short hops is more cost-effective as compared to one long hop. In the negotiation phase, each master agent thus aggregates the information from the children in its respective cluster using the algorithm and flowchart given in Figure 5.2 & Figure 5.3 respectively.

The same procedure is being carried out at every cluster which is being initiated by the root node. The cluster heads of the respective clusters, in addition to aggregating data in their cluster also refer to the global tuple space to inform about the root node to their immediate one-hop neighbors. The collected data is submitted to the root node where the agent further fuses all the data and submits the processed information to the sink. The agent at this level performs decision fusion i.e. if two (or more) cluster heads have submitted the same information, only one is accepted. If an intermediate cluster head fails, then an election algorithm [12,17] is executed to choose the new cluster head.

Algorithm :Aggregation_Agent()

Input :

- ***node[1…n] :A cluster of WSN consisting of n nodes***
- ***Ts: Time at which the master agent sensed the event***

Output : Aggregated Data at each cluster head

```
Begin
Function EvaluateTimestamp(Event_of_ChildAgent)
        { Activate ChildAgent    // the child node is chosen randomly from n
        nodes using local tuple space
        Ts1 = GetTimeStamp(Event_of_ChildAgent)
        return Ts1
        }
Master_Agent_Negotiation()
        {
        Ts2= Call EvaluateTimestamp(Event_of_Ist_Node_VIsited)
        If (Ts2>Ts) // event being sensed is a previous one
                Action = InformationDiscarded
        Else
                Action = Add_Data_to_Buffer(Data)
        For ( i =2; i<=x; i++) // x<=n .It is the number of the sensor node
        which is visited by the master agent till it ends its itinerary.
                {
                Ts3=CallEvauluateTimeStamp(Event_of_Next_Node_VIsited)
                If(Ts3<Ts)
                        Action = InformationDiscarded
                Else
                        Action = CompareDataBits() // data is rejected if it is
                        redundant or else added to buffer
                AggregatedData = DataAggregation()
                Add_Data_to_Buffer(AggregatedData)
                }
end
```

Figure 5.2: Algorithm for Data Aggregation

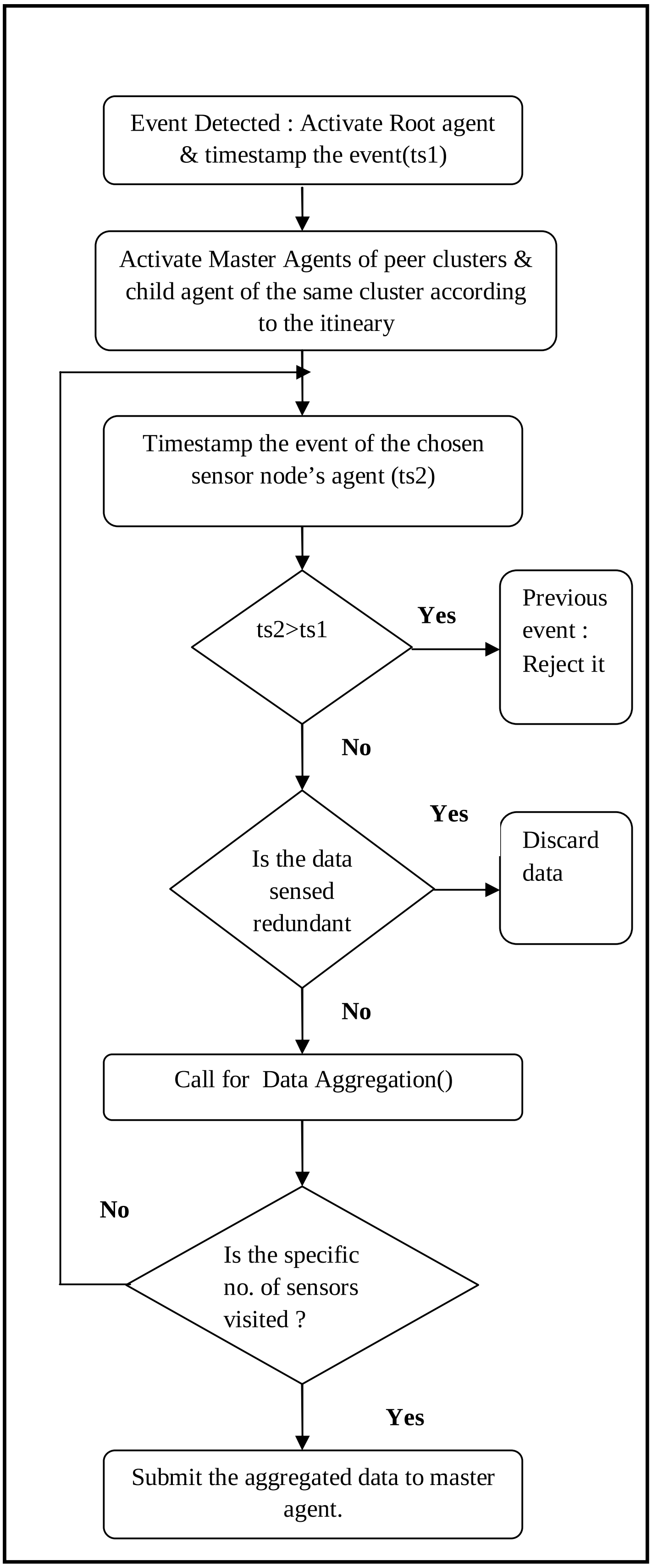

Figure 5.3 : Flowchart for Data Aggregation

This *two level agent aggregation approach* performs distributed in-network data aggregation in an efficient and cost effective manner as the task of aggregation is being carried out by agents and not by sensors thereby alleviating them from the complex task of performing aggregation in addition to routing process.

5.3.1.2 Hybrid Data Fusion

As already described, the root agent activates it peer master agents to gather the information in their own domain which can then be forwarded to fusion center or sink. The peer cluster heads are desired to report within the threshold time about the event being sensed, failing which it is considered that the particular cluster do not have any event to report. The current work considers sensing range of a sensor as one of the prime contributing factor in sensing and forwarding the information. Here, the sensing range ‘r’ of a sensor implies that the sensor can sense within the radii ‘r’ only. In addition to many other details [94], the master agent now maintains a data structure, known as Sensing Range Table (SRT) its packet recording the ‘r’ value of all visited nodes. Table 5.1 delineates the modified packet format of the master agent and is explained below.

Table 5.1: Modified Packet Format of Master Agent

Agent’s Identification (i,j)	Agent’s Itinerary (a,b,c)	Sensing Range Table (SRT) (m, r_m)	Data Buffer	Processing Code

- ***Agent’s Identification***

  It is a quadratic tuple (i,j) where ‘i’ indicates the id of the dispatcher agent (cluster head in this case) and ‘j’ is the time-stamp assigned to the agent.

- ***Agent’s Itinerary***

  It is a cubic tuple (a,b,c) where ‘a’ is the id of the first node to be visited, ‘b’ is the id of the destination node and ‘c’ represents the base address of the list containing the ids of all the nodes to be visited in the order. When an agent visits a node or is unable to visit the node due to link failure, it deletes the respective node from the list.

- ***Sensing Range table (SRT)***

  It is a quadratic tuple (m,r_m) , where ‘m’ indicates the sensor-id and r_m is sensing range of m^{th} sensor . With each visited sensor, master agent updates this parameter.

- ***Data Buffer***

  It represents the agent’s buffer which carries the partially integrated (rather fused) results.

- ***Processing Code***

  It is the execution code carried by the agent.

On reaching a particular node, the master agent compares the r value of the current node with all the previously stored values. If this value is found to be less than or equal to any of the previously visited ISUs, it is assumed that the sensing range of the current sensor is same as that of any of the previous ISU and thus it is concluded that the data being sensed is redundant and hence rejected. On the other hand, if the value is greater than the previous ones, it is fused in the data buffer. Assuming that the sensors can detect noise in addition or in place of the desired sensor reading, the data fusion algorithm applied within the cluster of ISUs is described in Figure 5.4. In this way, only non-redundant data is being considered by the agents for data fusion. Figure 5.5 describes the working of the above proposed algorithm which runs simultaneously in all the clusters made in the network.

The section proposed a hybrid protocol that makes use of mobile agents for performing fusion at the cluster level. Since the overhead of communicating the

```
Algorithm : FUSE_INFORMATION_CLUSTER
    Input:
        • n:no.of ISUs
        • MA : master agent embedded in the cluster head
        • d: data in the data buffer
    Output : Aggregated Data at each cluster head
Begin
for (i=1;i<=n;i++)
        { //determine the Sensing range of ith sensor
        SRT[i] = Sensing range[ith sensor] // insert in the  SRT tuple
        for(j=1;j<=i-1;j++)
                { if (SRT[i]<= SRT[j])
                { neglect (SRT[i])
                 break;
                }
        }
call _Aggregation_Agent()
End
```

Figure 5.4: Algorithm for Hybrid Data Fusion

information is being taken over by agents, it could greatly enhance the energy-efficacy and lifetime of network. However, at this level, mobile agents could carry unencrypted data and therefore there exist scope of improvement. In order to secure the data at the time of transmission, a security mechanism is being proposed in the upcoming section.

The security algorithm named as SDDMA [134] executes parallel to fusion algorithm presented above as the data which is to be fused is now handed over to mobile agent in encrypted mode and mobile agent in turn is required to decrypt the same before fusion.

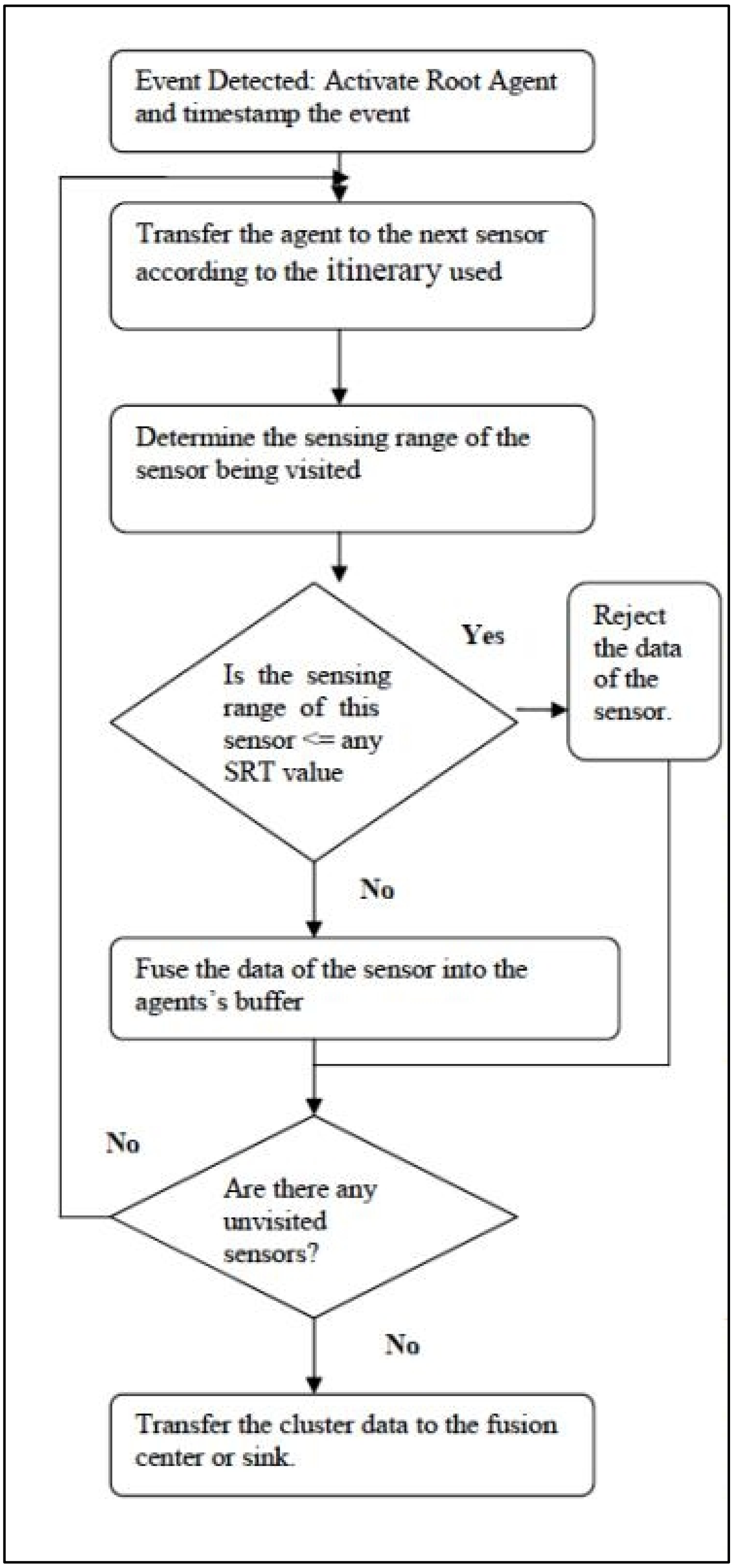

Figure 5.5 : Flowchart for Hybrid Data Fusion

5.3.2 SECURE DATA DIFFUSION USING MOBILE AGENTS (SDDMA)

Due to unfriendly environments and unique capabilities of wireless sensor networks, it is an adventurous task to protect sensitive information transmitted by wireless sensor networks [11]. In addition, wireless sensor networks encounter security problems which traditional networks generally never face. Therefore, security is of prime importance for wireless sensor networks and there are many security considerations that should be looked into. In this section, we present the essential security requirements that are raised in a wireless sensor network environment and have elaborated these requirements related to data aggregation process [8].

5.3.2.1 Security Related Issues in WSN

- ***Data Confidentiality***

  Data confidentiality is probably the most common aspect of information security. In wireless sensor networks, data confidentiality ensures that secrecy of sensed data is never disclosed to unauthorized parties and it is an issue of utmost importance in mission critical applications. It is expected that a sensor node should not pass on its readings to neighbouring nodes. Moreover, in many applications, sensor nodes transmit highly sensitive data, e.g., secret keys. In the military, concealment of sensitive information is major concern and therefore it is absolutely important to build secure channels among sensor nodes [124].

- ***Data Integrity***

  Although data confidentiality ensures that only intended parties obtain the un-encrypted plain data, it hardly protects data from being altered. Data integrity guarantees that a message being transferred is never disturbed. A malicious node may corrupt messages to prevent network from functioning properly [11]. In fact, due to untrustworthy communication channels, data may be altered without the presence of an intruder.

- ***Source Authentication***

Since wireless sensor networks work on a common wireless medium, any unauthorized entity can intrude the sensor nodes. So, the sensor nodes need authentication mechanisms to detect maliciously injected or spoofed packets. Source authentication enables a sensor node to ensure the identification of the peer node it is in communication with. Without source authentication, an opponent could masquerade a node, thus gaining illegal access to resource, crucial information and interfering in the operations of other nodes. Moreover, a compromised node may send data to its data aggregator under several fake identities so that the integrity of the aggregated data is doubtful. Faking multiple sensor node identities is called Sybil attack [5].

- ***Availability***

Availability ensures the workability of network services against Denial-of-Service (DOS) attacks. A DOS attack can be started at any layer of a wireless sensor network and it may disable the victim node(s) permanently. In addition to DOS attacks, increase communication or computation may finish off battery charge of a sensor node. Consequences of availability loss may be drastic.

5.3.2.2 Working of SDDMA

Wireless sensor networks generally comprises of a huge number of cost efficient sensor nodes that have restricted sensing, computation and communication capabilities. Due to restriction of sensor nodes resources, minimized data to be transmitted is preferable as it improves the life of sensor and its overall bandwidth utilization is improved [54,123]. Data aggregation is the process of accumulating and reorganizing sensor data so as to lower the amount of data transmission in the network. Data aggregation/fusion protocols in wireless sensor network must satisfy the security requirements explained in above section. However, the resource constraints of sensor nodes and necessity of plain data for aggregation process poses great challenge when security and data aggregation together are put to work [63,126]. Security requirements of wireless sensor networks can be satisfied using either symmetric key or asymmetric key cryptography. Security and data aggregation are together achieved in a hop-by-hop fashion. However, data aggregation protocols usually cannot aggregate encrypted data. Therefore, such data aggregation protocols must decrypt the sensor data to perform data aggregation and encrypt the aggregated data before transmitting it. That is, data aggregators must decrypt every message they receive, aggregate the message according to the corresponding aggregation function, and encrypt the aggregation result before forwarding it. This entire process of data aggregation and providing security is done by the agent in the agent based computing system.

The author aims to propose a mechanism for securely sending the data to cluster head using agents. Since, sensed data should be secured while travelling on a wireless medium; it is suggested by providing security to sensed data using asymmetric key cryptography. The work is an extension of a data centric protocol called IDDMA (Improved Directed Diffusion Protocol using Mobile Agents) [39]. The SDDMA mechanism is divided into three phases: *Controlled gradients setup phase, Target region setup phase and Secured mobile agent action phase.*

- ***Controlled Gradients Setup Phase***

  In the first phase a CH node floods an interest message to its neighbours to setup initial gradient values of sensors taking both optimal path choosing and load balancing into consideration.

- ***Target Region Setup Phase***

In the second phase of SDDMA, two steps are put forth. In the first step source node send their identity to the sink along with their public keys. In the second step of this phase, each source node floods all its neighbours with the data for the second time focussing on setting up of path which is maintained at the initiator node.

- ***Secured Mobile Agent Action Phase***

In the third phase, a mobile agent will be created by sink and dispatched to the target region to visit the source nodes. When the task is finished, the mobile agent will return to the sink with the aggregated sensory results.

Table 5.2 depicts the modified packet format of the mobile agent for this approach.

Table 5.2 : Packet format of the Mobile Agent for SDDMA

Time_Stamp	SinkId + PUKey	Fst_Src	Lst_Src	Next_Src	Srclist + PUkey	SrcNum	Processing Code	Data

Where

Time_Stamp : Time at which the event occurs

Sink_Id + PU Key : Public Key of the cluster head

Fst_Src : First node to be visited according to the itinerary

Lst_Src : Last node to be visited according to the itinerary

Next_Src : The next source to be visited according to the itinerary. The id of this node is deleted from the buffer once it is visited.

SrcList +PUKey : A quadratic tuple describing each source node to be visited along with its public key.

SrcNum : No. of nodes to be visited in a given cluster

Data : Partially Integrated Data

Processing Code : Code contained in the mobile agent

The cluster head encrypts mobile agent with the first source's public key and then dispatches the encrypted mobile agent to the target region. First source on receiving the mobile agent decrypts it with its private key. This source then copies the mobile agent processing code into its memory and the mobile agent starts functioning. Mobile agent migrates among target nodes to collect sensory data and also copies its processing code into the memory of each source node. Now, the agent processes the sensed data stored at source node and aggregates it. After processing, source node will encrypt the mobile agent with the next source's public key. Next source on receiving the agent decrypts the mobile agent with its private key and starts the function of the corresponding mobile agents again. Mobile agent aggregates this processed data with the data stored in agent data field. On the last source node, agent processes the data and aggregates it with the data stored in agent data field. Now last source will encrypt the agent with sink's public key. After visiting the last node according to the itinerary, the mobile agent discards the processing code and returns to sink with the collected result. After the mobile agent leaves each source node, all the relevant code stored in each source node will be discarded. When the mobile agent leaves *Lst_Src*, it will return to the cluster head as shown in the Figure 5.6

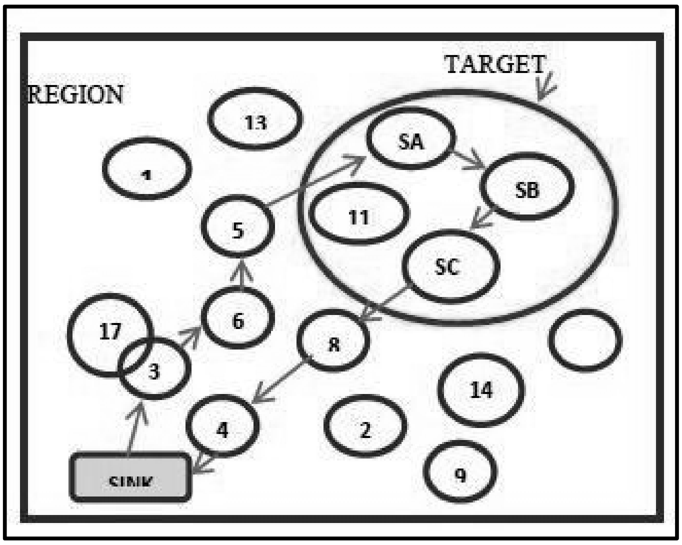

Figure 5.6 : Detailed Mobile Agent Action Phase

Figure 5.7 and Figure 5.8 depicts the algorithm and flowchart for SDDMA.

Algorithm : SDDMA (src_lst, fst_src, lst_src, nxt_src)

Input : Unencrypted data wih ISUs

Output : Encrption of data at ISUs

```
Begin // at the cluster head end
  Encrypt(Pu_fst_src(MA))
 //After creating an agent, Sink will encrypt the agent with the public key of fst_src
      Repeat until ( src_node !=lst_src)
            {
             Decrypt (Pr_Src_node(MA)) //at each ISU being visited
              Call _Aggregate_data()
              Encrypt(Pu_nxt_src(MA))
    // src node will encrypt the agent with nxt_src public key and send the agent to next node
```

Figure 5.7: Algorithm for SDDMA

The aim of developing Secured Directed Diffusion using Mobile Agent(SDDMA) is to provide security to the data sent to the cluster head. A mechanism is being proposed for securely sending the data from source nodes to sink using Mobile Agents. An asymmetric key encryption technique is used for the same. For generating the public key and private key pair, RSA algorithm is used. Using this technique mobile agent is able to process the data, confidentially. Also aggregation takes place at each node while maintaining the confidentiality of the data.

Source node energy is utilized in encrypting and decrypting the agent. All processing and aggregation work is done by the agent which results in saving the energy of the nodes. In addition, the nodes do not send the sensed data to other nodes, as the agent is taking the data with it. As a result it is also saving much energy of the source nodes.

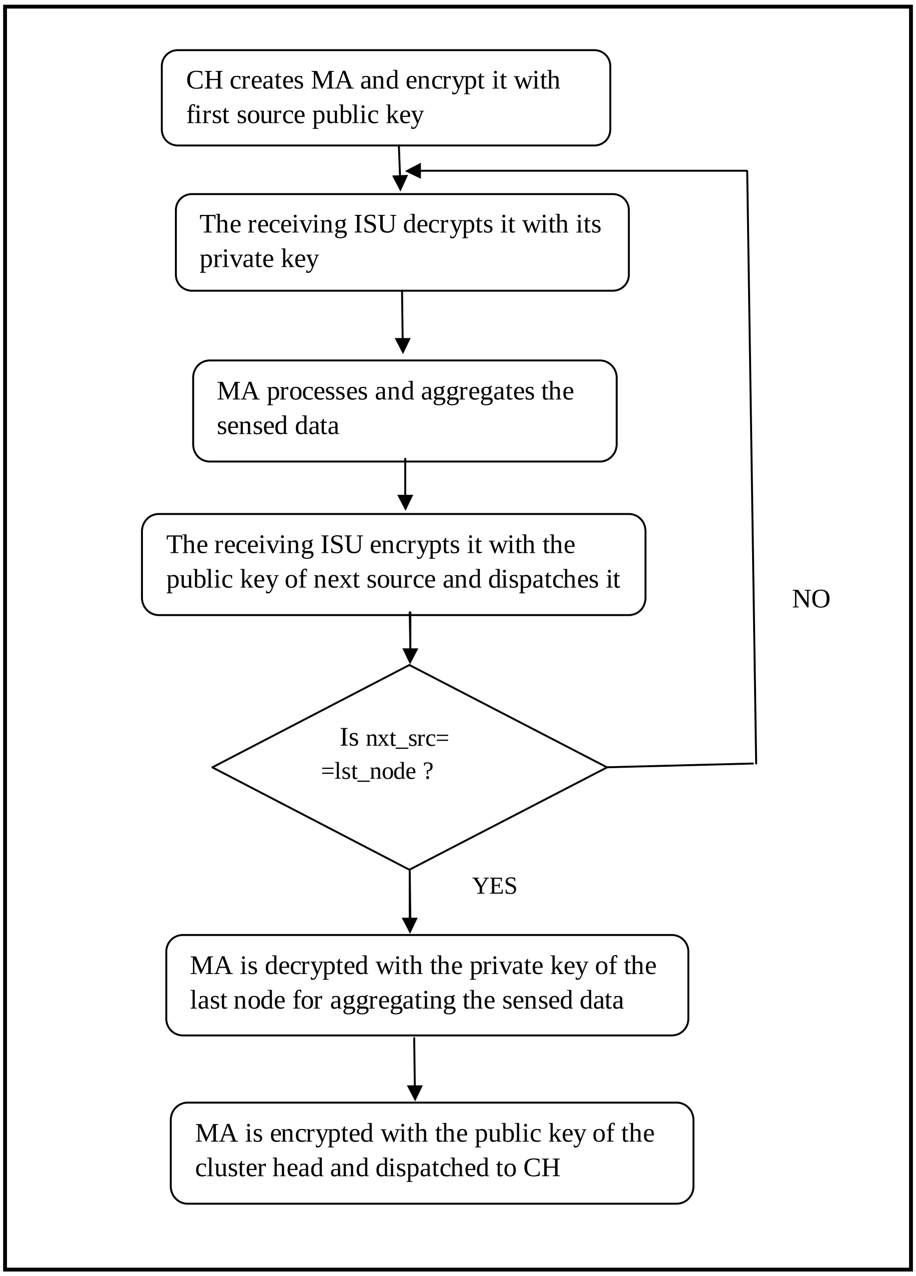

Figure 5.8 : Flowchart for SDDMA

5.3.3 Forest Fire Detection: A Non-Deterministic Application

The section presents a case study of an event driven application for non-deterministic environment namely Forest Fire Detection. The work incorporates the four phases of MC3F2 to detect and prevent fire in a forest.

As it is already mentioned that the foundation of this work is AERDP, hence the network which is designed to detect fire has ISUs deployed in it. The case study is rightly termed as Software Agent Based Forest Fire Detection (SAFFD).According to the work, when fire is detected by such a network, sensors relay this information through the means of mobile agents to the appropriate actors in their respective cluster where actors are cluster heads having decision making capabilities along with a hardware unit attached to them called water sprinklers. These sprinklers then, immediately come into action and extinguish the fire in that particular cluster where the collected and sensed data is still be valid at time of taking actions by actor nodes.

In this case study, there are two classes of components- Sensors and Actor nodes. Sensors are regular, low cost, low power devices with limited sensing, computation and wireless communication capabilities. They are intended for sensing the environment and local processing can be done on it. On the other hand, actor nodes are special resource rich nodes which have been made intelligent by embedding mobile agents in them. Multiple actor nodes can coordinate to decide on appropriate actions based on information received from multiple sensors deployed all over the network in random fashion. They have attached water sprinklers to put off fire as and when required.

The proposed approach works in two phases namely data collection phase and risk analysis phase (Figure 5.9).

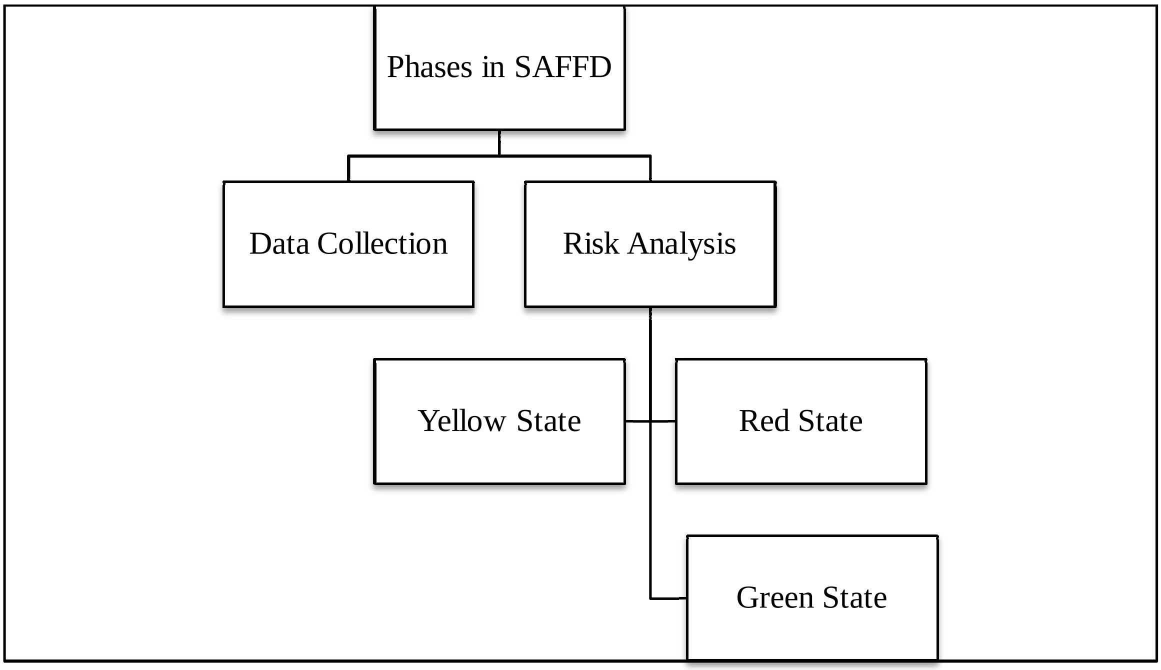

Figure 5.9 : Phases of SAFFD

The description of each phase is given below:

A. Data collection phase

In this phase, mobile agent is dispatched to first node to be visited. When mobile agent arrives at a sensor node, it first compares the id of current node with that of last node to decide whether it has arrived on destination or not. If not, mobile agent continues migrating towards and specific sources. Otherwise, it operates as follows:

- Collects locally processed sensed data
- Deletes id of current target from source_list maintained in the buffer.
- Choose next source to move on.
- If current source is *last_source*, the mobile agent will return to actor.

B. Risk Analysis Phase

This phase deals with analyzing the severity and risks associated because of intensity of fire. The severity of fire being detected is categorized in 3 levels i.e. green state, yellow state and red state. Here, *Green State* represents the lowest risk state for possible fire danger. In this state, sensor's sensing period is longer as compared to other states' period. Further, in contrast to *Red State* which is the highest risk state and water sprinklers get activated in this state to put off the fire, *Yellow State* represents the medium risk state. This state may have transition to red or green state.

In data collection phase, actor dispatches mobile agent from first_source to the last_source according to the itinerary determined using EEGCF [91]. At each sensor node, it collects sensed data and processes it locally. At last, the aggregated and processed temperature value is passed to actor. In risk analysis phase, actor decides whether the risk state is green, yellow or red using some predefined thresholds to specify the severity level of fire. The green state is least severe while yellow state is moderate risk state. But red state means fire sprinklers need to be initiated to put off the fire before it spreads in the whole forest area.

The working algorithms are being listed in Figure 5.10(a) and Figure 5.10(b) respectively.

5.3.3.1 Implementation and Results

The work has been implemented using Aglets. Aglet API is an agent development kit consisting of a set of Java classes and interfaces that allows creation of mobile Java agents i.e. "write once, go anywhere" as once an aglet is written, it will run on every machine that supports the Aglet API [67]. Aglet API mirrors the applet model in Java and hosted by an Aglet Server i.e. Tahiti Server.

Algorithm:Data_Collection_Phase_SAFFD

Input:

- ***source_list //sequence of all sensor nodes in the cluster***
- ***first_source //first source where MA is dispatched initially***
- ***last_source //destination node in cluster.***

Output: T_agg_data // Aggregated and processed temperature value

Begin

Step 1:Timestamp the Event

Step 2: Dispatch MA to first_source from actor for a round in cluster

2.1 if current_source==first_source then, MA migrates towards first_source node

2.2 else if current_source==next_source and next_source!=last_source then,

(i) MA collects locally processed sensed data

(ii) delete the id of current_source from source_list in MA

(iii) Among the sources in source_list, select the one with Maximum gradient as next_source

(iv)Set next_source in MA packet

(v)MA migrates towards next_source

(vi)Make status of current_source sleep=true

2.3 else if current_source==last_source then, MA collects locally processed sensed data.

MA migrates back to actor node.

2.4 else if next_source==first_source then, MA migrates between source nodes.

Step 3: Receive T_agg_data i.e. aggregated processed data from sensor nodes at Actor

Step 4: Analyze risk state and take appropriate action accordingly. Dispatch MA to Cluster Head.

End

Figure 5.10(a) : Algorithm for Data Collection Phase

```
Algorithm:Risk_Analysis_Phase_SAFFD

Inputs:
        Tgreen _threshold
        Tgreen _MAX,
        Tgreen_Avg,
        Tyellow_max,
        Tyellow_Avg
        T_agg_data // received processed data in above round.
        C // Error value constant for precision.
Output: risk state
Begin
        Step 1: Green State
                if T_agg_data>=Tgreen _threshold +C then,
                        enter yellow state
        Step 2: Yellow State
                If T_agg_data>=Tgreen _MAX+C then,
                        enter RED State
                if T_agg_data<Tgreen _Avg +C then,
                        enter Green state
        Step 3: Red State
                if T_agg_data<Tyellow _Avg +C then,
                        enter Yellow state.
                if T_agg_data>=Tyellow_max+C then,
                initiate water sprinklers.
End
```

Figure 5.10(b) : Algorithm for Risk Analyis Phase

The implementation has been carried out on a single machine with three ports (the work however is scalable to three different machines on LAN). The first port acts as an actor which dispatches agents to other two ports acting as sensors in a predefined sequence. These sensors then, sends aggregated and processed temperature value from all the sensors back to actor. The execution of code is shown in Figures 5.11(a) to 5.11(d).

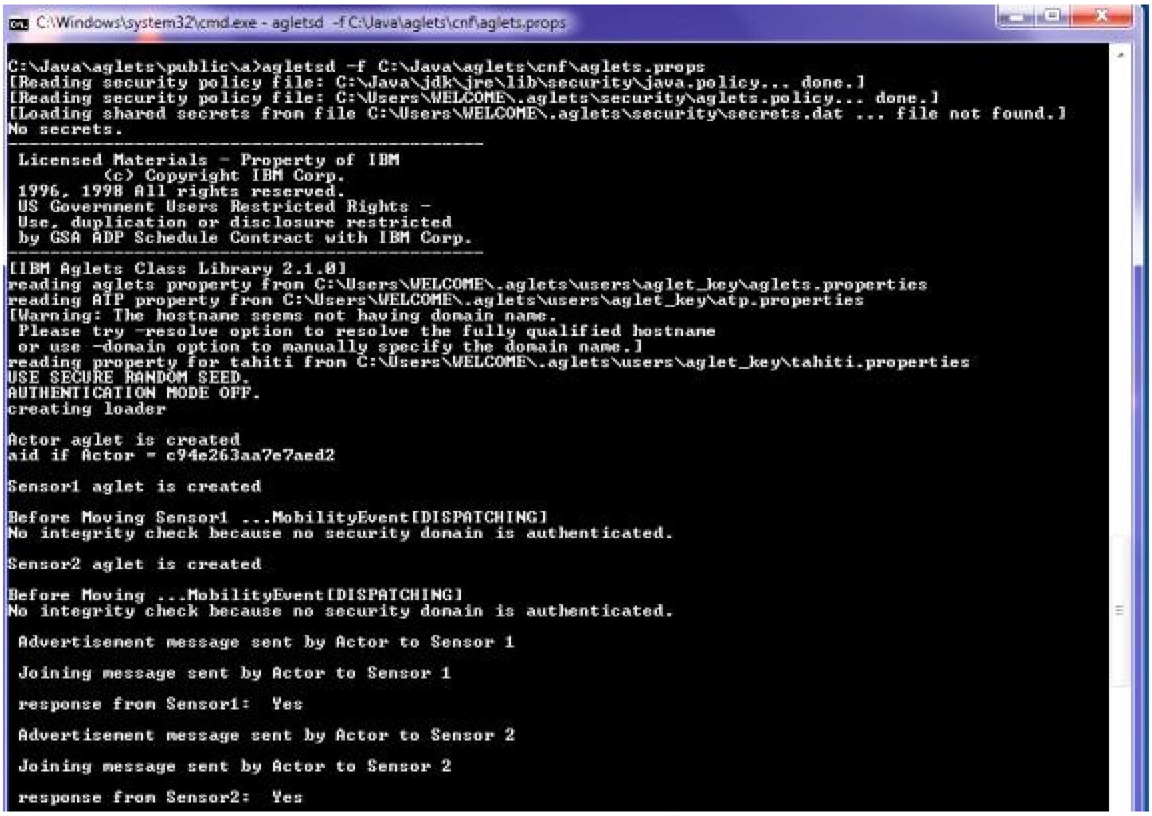

Figure 5.11(a) : Command Window with for Actor(port 4434)

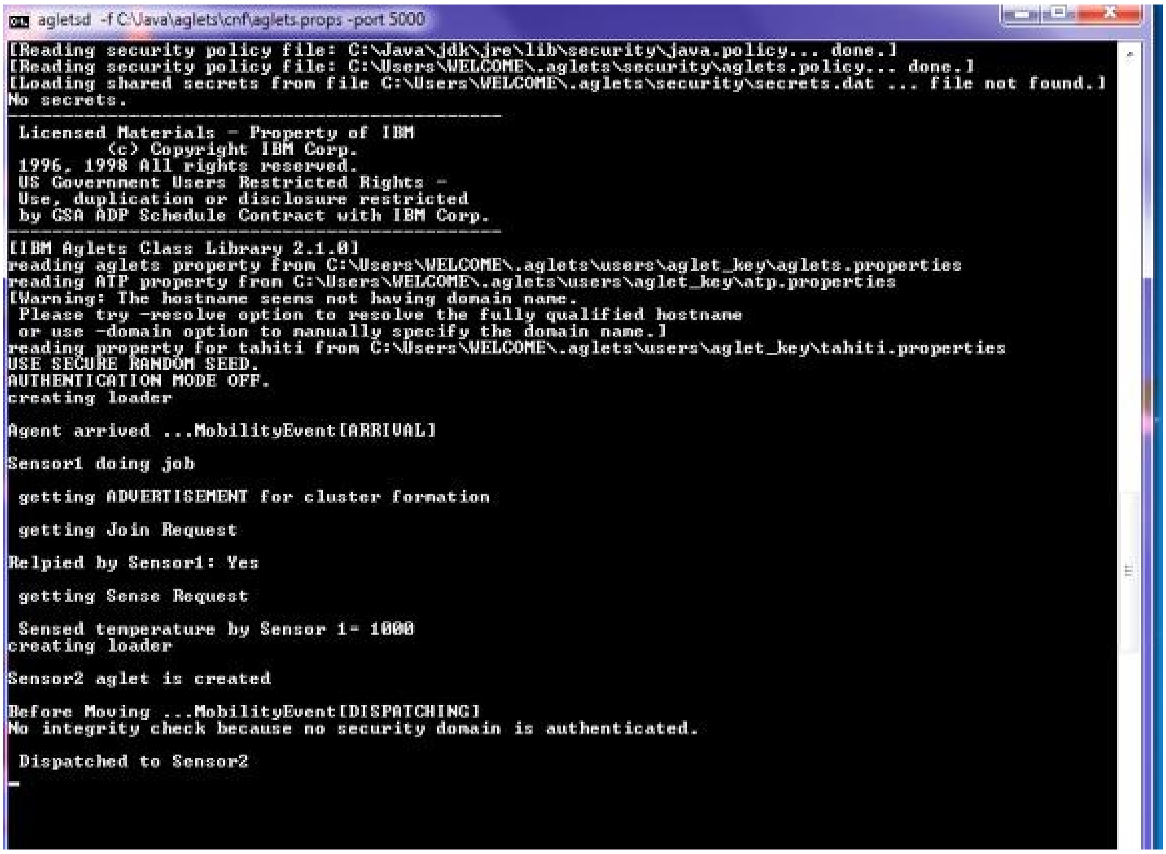

Figure 5.11(b) : Command Window for Sensor1(port5000)

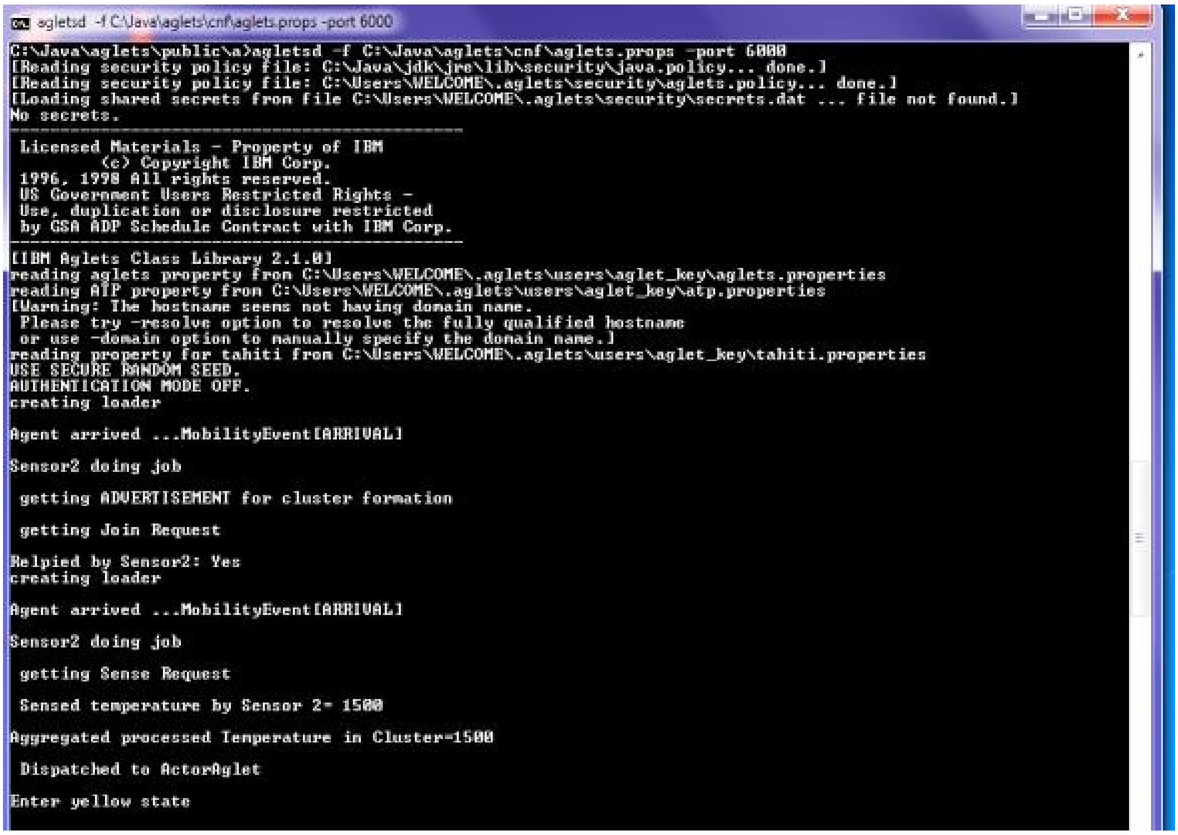

Figure 5.11(c) : Command Window with for Sensor2(port 6000)

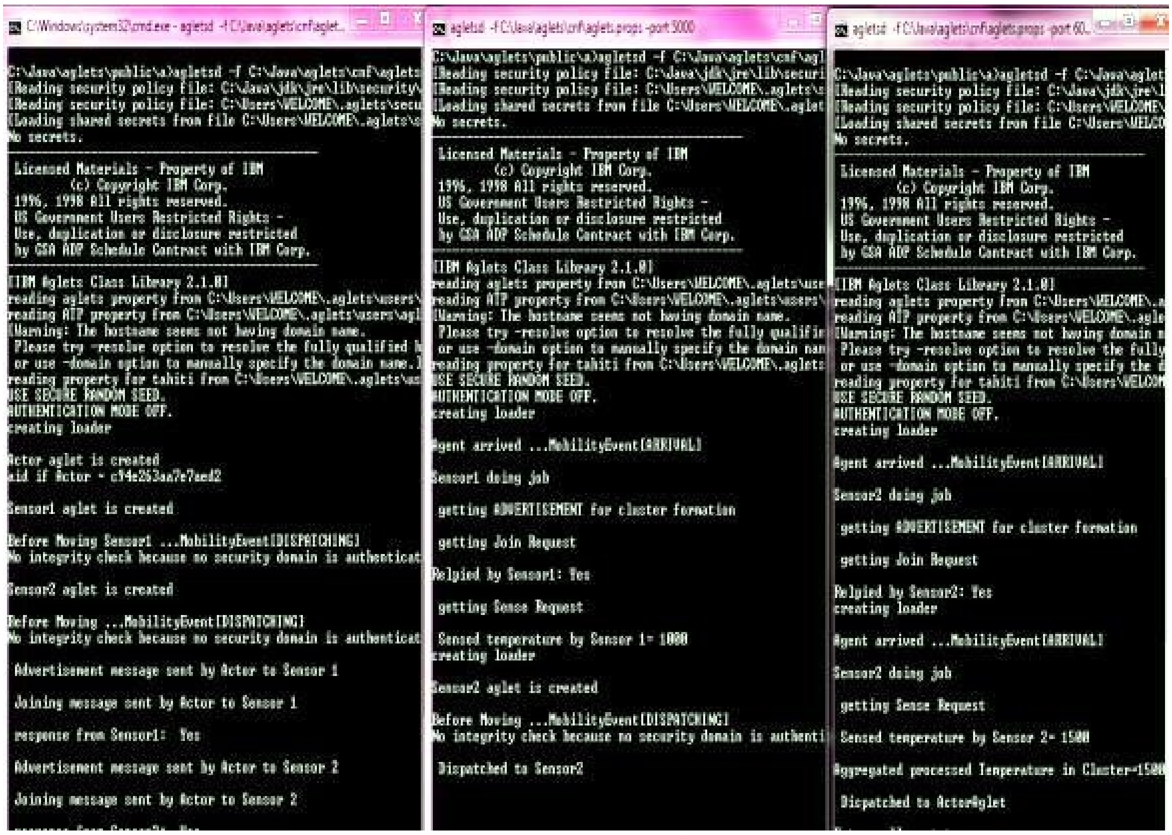

Figure 5.11(d) : Integrated results for all ports

The implementation results above convey that the use of mobile agents drastically increase the lifetime of network as number of packets transmitted over the network have been reduced to a great extent. It thus overcomes the shortcomings of Client/Server based methodology. In an environment where source nodes are close to each other, and considerable redundancy exists in the sensed data, the source nodes generate a large amount of traffic on the wireless channel, which not only wastes the scarce wireless bandwidth, but also consumes a lot of battery energy. Instead of each source node sending sensed data to the sink node, as typically occurs in client/server-based computing, the work proposed a software mobile agent based paradigm for data processing/aggregating/ concatenating in a wireless sensor actor network architecture. Though the work has been carried out for detecting forest fires but it can be extended to any event driven application. The work is scalable i.e. it can be extended to *n* number of nodes in the cluster. Though the use of software agents causes extra burden on the network, but the increased cost is compensated when the end results increase the lifetime of the network.

5.4 EVALUATION OF MC3F2

Considering the algorithms proposed in current as well as previous chapter, Figure 5.12 presents the compiled view of MC3F2. In order to evaluate performance of MC3F2 with respect to mobile agents exclusively, literature [5,71,72] was heavily grilled to find out the metrics of evaluation and it was discovered that there are no generic metrics for evaluating any agent based framework. To the best of our knowledge, such evaluations had been application specific. Works of various researchers suggests that an agent based framework may be evaluated on the basis of intra and inter-cluster agent communication cost, cost of maintenance of agent based system, scalability of framework, reliability of live agents, security of data packets being transferred through the agents and also the overall effect of injecting agents on the life time of network, just to list a few. Parameters such as trusted agents, autonomy, robustness etc. [15] are also prevailing in the literature but these are more or less application dependent. However, current work aims to evaluate the performance of MC3F2 on the basis of above listed metrics described below:

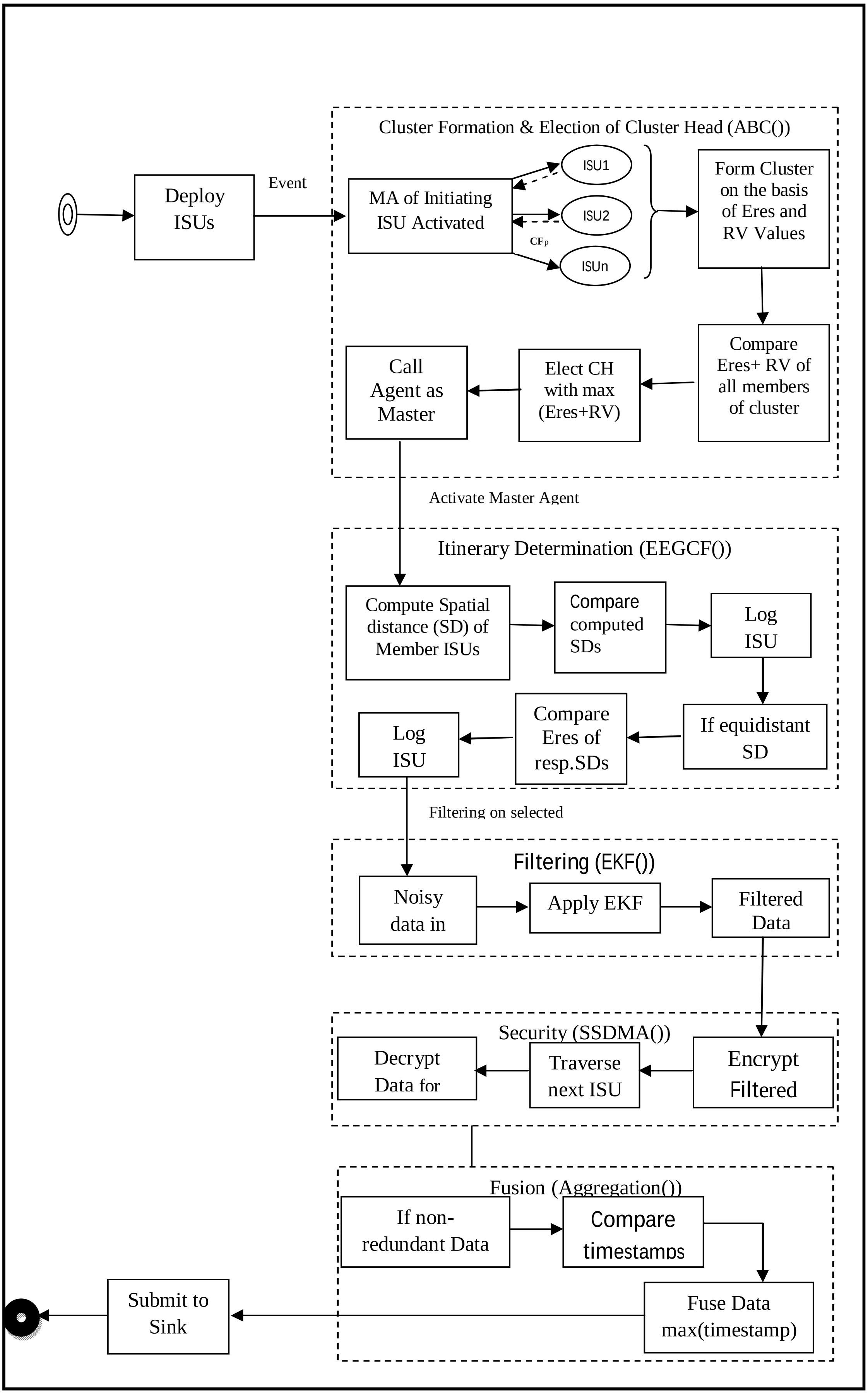

Figure 5.12 : Detailed Flow Diagram of MC3F2

The communication cost is the cost incurred due to the communication established between agents during requests reception, aggregation of the data and transmission of the data to the next agent in the cluster (intra-cluster communication) and to executive cluster-head (inter-cluster communication) as well.

It initially appeared that intra and inter-cluster communication would add an overhead cost to the entire system. While simulating MC3F2, for the sake of simplicity, we took intra and inter-cluster communication cost similar values and henceforth we would be referring both intra as well as inter-cluster communication costs as communication cost (ć) only. Accordingly, equation (5.13) gives the communication cost as

$$\acute{c} = + \acute{c}_{\boldsymbol{rec}} + \acute{c}_{\boldsymbol{agg}} + \acute{c}_{\boldsymbol{tran}} \tag{5.13}$$

In MC3F2, the communication among mobile agents was constrained using modified random way point model and also RCNTEP was employed to establish communication. The agents only having some data to transmit could only communicate leading to lower communication cost which in turn saved the potential amount of energy of sensor nodes.

Further, the performance of any agent based system is also majorly affected by the high cost of maintaining the agent post its dispatch in a non-deterministic environment such as WSN where, maintenance cost is the cost incurred to maintain and manipulate the complex data structures and the knowledge base associated with each cluster. However; since data structures of agents once deployed in WSN could not be altered in this case, therefore it is the cost of updating the knowledge base only which was further updated dynamically so as to provide the real-time information to peer members of the cluster.

Therefore, the cost of maintaining the deployed agents μ depends upon the updates done in knowledge base. Hence, equation (5.14) and subsequently equation (5.15) governs the variation in the same.

$$\mu \ \alpha \ \text{kbupdates}$$

$$\mu = k \times \text{kbupdates} \tag{5.14}$$

where k is a constant ($k \geq 0$). Hence,

$$\mu = \begin{cases} 0 & \text{for } k = 0 \\ k & \text{for } k \geq 1 \end{cases} \tag{5.15}$$

The decision to use mobile agents in place of conventional client/server technology is a trade off between the cost incurred to use mobile agents and the efficiency achieved by using them. Figure 5.13 shows that that with number of agents and knowledge updates observed a almost linear relationship implying more number of agents leading to more knowledge updates. Therefore more updates lead to higher maintenance cost.

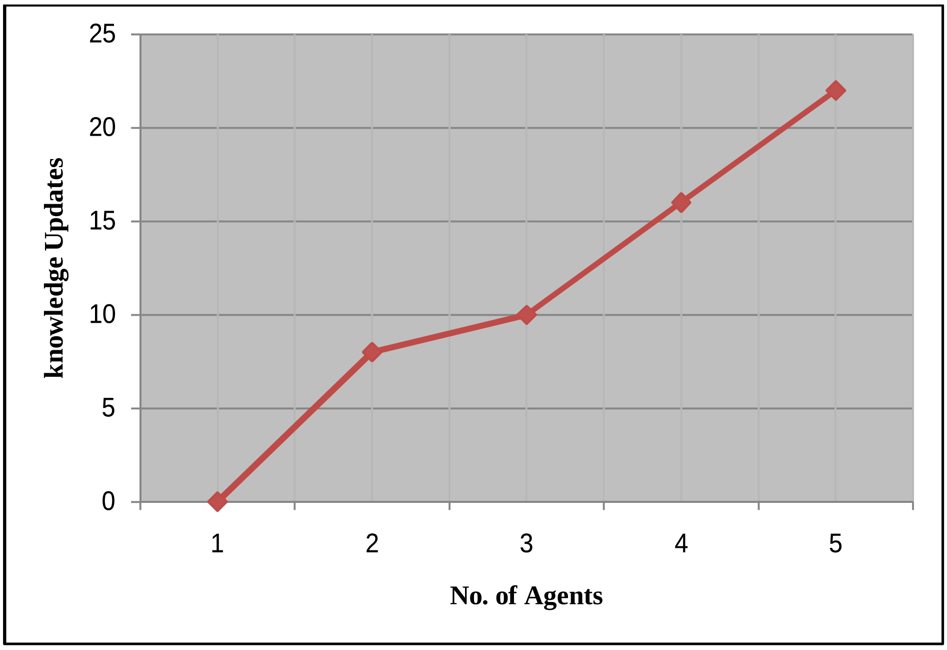

Figure 5.13 : Maintenance Cost of MC3F2

Scalability is one of the prominent factors that are highly desired in a multiagent framework. Scalability here deals with increasing the number of agents to handle the same number of sensors. Defining more formally, "*Scalability of MASs is the average measure of the degree of performance degradation of individual agents in the framework caused by the expansion of size of agent's society*" [16]. Since, mobile agents are reusable entities and could also adopt the goal of other agents such as role of cluster head, therefore number of agents deployed was limited and that also could exchange the role of each other, therefore the need of injecting new agents did not evolve. Moreover, MC3F2 is based on the concept of ISU in which agents are embedded within the sensors. Therefore, increasing the number of agents implied increasing the number of sensors in the network i.e. the density of the network (see Figure 5.14).

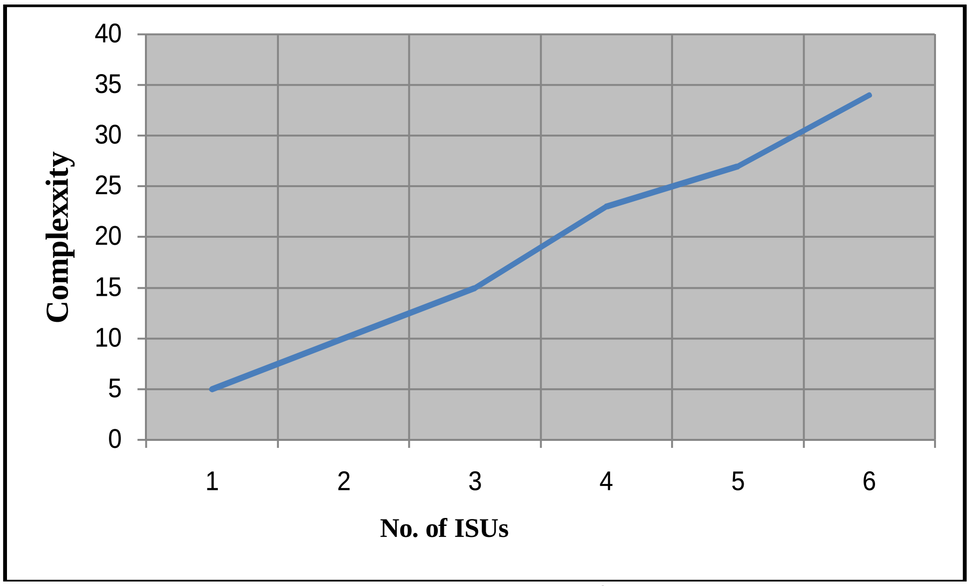

Figure 5.14 : Scalability of MC3F2

As the work primarily focused on non-deterministic environments where the sensors once deployed remained untethered and the challenge was to ensure coverage and connectivity, the issue of increasing agents in the network did not occur. Further, while evaluating MC3F2 analytically it is found that like any other Multi Agent Framework (MAF), scalability represents the algorithmic complexity i.e. whether the worst-case performance of the system is bounded by polynomial function of the load.

Thus the analysis of MC3F2 is confined to computational time as a major performance indicator. In the second and third phase of MC3F2, the increase in number of agents linearly effected the time taken to elect the cluster head, further to route the agent in a given cluster and finally filter the data at each ISU node itself. In the fourth phase however, where in-network processing or data fusion takes place within each cluster, the increase in number of participants is restricted as the buffer size of the agent is directly affected by the number of ISUs it visits in a given round.

Thus it can be concluded that cumulatively the time complexity of MC3F2 increases logarithmically with the increase in the number of participants.

As mobile agents are responsible for carrying the code and data over the network pro-actively, security of the data being transported/ migrated is thus an issue of concern.MC3F2 is secure in the sense that it makes use of encryption mechanism for transfer of data between various ISUs. The packet format of the mobile agent has been modified to contain the key- value pairs for encryption and decryption. The details of the same are given in [134]. Since the additional cost of performing encryption/decryption is being achieved by software, it doesn't require any additional installation of new infrastructure but can be implemented on the existing one. Since, the security of the content was considered at later stage and we could initially evaluate the system without encrypting the data, a comparison is available depicting the number of malicious transfers and encryption done at later stage.

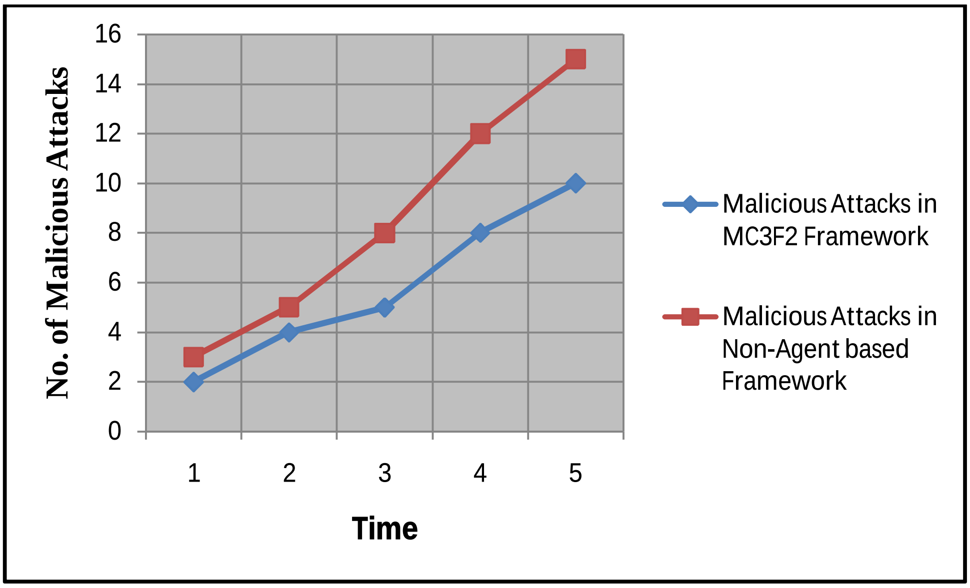

Figure 5.15 : Security in MC3F2

Reliability of a mobile agent refers to its credibility value which in turn is calculated based on its previous performance. Though mobile agents through its various properties (like autonomy and intelligence) provide energy aware communication, reliability of these proactive units also needs to be considered specially in mission critical applications. MC3F2 uniquely contributes towards ensuring reliability of mobile agents prior to dispatching it for data fusion in the respective cluster. It makes use of RCNTEP [4] protocol to calculate the same in its clustering phase. This calculation of reliability value ensures that the communication between sensor motes is between reliable communicating parties i.e. agents. This feature of MC3F2 is essential for hard real time applications. Although an overhead in calculating in the reliability value was observed but on the other hand the overall performance was found to improve.

One of the prime reasons for using /embedding mobile agents in WSN is to increase the lifetime of otherwise unattended sensor nodes. Each travelling agent thus packages a given set of instructions, moves to the destination and processes the instructions locally. Further, each agent is provided with the message ID, so that message is conveyed to the desired agent only. Once the agent has been dispatched, it can operate asynchronously of its sender, thereby contributing to saving of bandwidth of the network. Information was also filtered and fused, which in turn also improved the network lifetime indirectly.

5.5 CONCLUSION

The chapter proposed two important phases out of four phases carried out during this thesis. A new dimension of using EKF has been projected and implemented as this technique is suitable for critical measurements which are not only accurate but also filtered data. The electronically acquired data can be distributed to the scientists for better analysis and it also gives an opportunity to view the data in real time. Similarly the filtered data at each source node is fused so as to achieve in-network processing. The work has also successfully introduced encryption mechanism to prevent security attacks. The work concludes by successfully evaluating MC3F2 on various parameters and it has been found that empirical results obtained are consistent to the expectations and intentions behind the design of framework.

Chapter VI

CONCLUSIONS AND FUTURE SCOPE

"Writing is hard, but even harder is to discover when it is time to stop."
*Peter Ustinov (*1921)*

6.1 CONCLUSIONS

This chapter recapitulates the major achievements of this research work and provides an outlook to further research that may lead to the ultimate goal of harnessing the energy of tiny and very low cost energy-self-sufficient wireless sensor nodes in its best possible manner.

The research work has given a novel, reliable, energy efficient event-driven approach called **MC3F2** which is a Multi Agent System (MAS) based solution for communication in non-deterministic WSN. It has in-depth explored and dealt with the challenges related to communication and energy matters of WSN and has successfully attempted to introduce strategies to increase the lifetime of unattended sensors.The data thus delivered to sink shall be of high utility to scientific community and can be analyzed for future predictions. The proposed approach has not only resulted into a proficient communication among Intelligent Sensor Units (ISUs) but also since it finally delivers filtered and fused information i.e. only relevant information without noise, it owes an edge over existing protocols.Once the tiny low cost sensors are clustered intelligently; they can be used for a multitude of applications. Setting the sensor network within any non-deterministic environment (a case study has been done taking forest fire as an example) is not only distinctive with respect to ongoing developments in this domain, but it also brings a range of features from the sensor world that can aid development and ease the monitoring of such unapproachable regions. In fact, by incorporating mobile agents within the sensors in any of the areas like dense jungles, underwater etc, the work attempts to introduce intelligence & decision-making capability in every sensor so that they become competent enough to deliver meaningful information instead of raw data to the base station. This information can then be used as per the application requirements.

An in-depth literature survey was carried out and the critical analysis of the same raised the following major shortcomings in the domain of non-deterministic environment of WSN which formulated the base of research work:

- To propose a novel and efficient communication strategy of sensors in the non-deterministic area.

- To propose intelligent clustering and itinerary strategy to establish an effective inter-networking within the deployed collection of nodes.

- To propose an efficient strategy for information processing by filtering of noise from the sensed data volumes.

- To propose a fusion technique within a cluster which can perform in-network aggregation and help to transfer non-redundant meaningful information instead of raw data to the sink or cluster head

- Evaluation & comparison of proposed framework with its agent based counterparts.

In the light of the objectives identified, the research work has given the following novel algorithms/concepts which can be used in non-deterministic environments of WSN to improve their lifetime:

- Architecture for event driven applications of non-deterministic environments of WSN has been given in **“A Mobile Agent Based Event Driven Route Discovery Protocol in Wireless Sensor Networks: AERDP”.**

- An intelligent itinerary determination method has been introduced which produces relatively less messy routes than its descendent protocols in the paper -titled **“Computing An Agent’s Itinerary in a Clustered Network: An Energy Efficient Approach”.**

- A case study for introducing security in mobile agents has been carried using Directed diffusion Protocol in "**Secured Directed Diffusion Using Mobile Agents (SDDMA)".** With petite changes, the same can be applied to any data centric or event driven protocols.

- A novel agent based approach for clustering the randomly deployed sensors has been given which uses only reliable software agents for communication in "**Design of Communication Strategy for Wireless Sensor Networks in Non-deterministic Environments Using Mobile Agents"** This paper also contains an application of Extended Kalman Filter to process the information locally so as to save the energy of deployed nodes has been given which proves to be a leverage over regular analytical methods of filtering the noise from data.

- The filtered data which is gathered by the mobile agent in every cluster is intelligently fused both at intra cluster level and inter cluster level so that only processed information is transferred to the sink/base station. This approach consumes relatively less energy of each sensor, thereby contributing to increase the lifetime of the network. The work has been presented in **"A Multi-agent Hybrid Protocol for Data Fusion and Data Aggregation in Non-deterministic Wireless Sensor Networks"** and **"Hybrid Data Fusion Using Software Agents for Event Driven Application of Sensor Networks".**

- A non-deterministic event driven application of Forest Fire Detection has been implemented in **"Software Agent Based Forest fire Detection(SAFFD): A Novel Approach".** It detects fire and takes necessary action depending upon the severity of the situation.

- The proposed framework has been evaluated in the **"Evaluation of MC3F2 Framework for Non-Deterministic Environment of Wireless Sensors".**

6.2 BENEFITS OF PROPOSED FRAMEWORK

The research work has successfully given a communication strategy which is designed for sensors having mobile agents embedded in them. When operational, the network of such intelligent sensors will incur the following benefits to the domain of routing in Wireless Sensor Networks:

- **Reduction in Bandwidth Consumption:** Since the work makes use of mobile agents for transporting data, it will result in sufficient saving of bandwidth as tasks will be encoded into mobile agents and then dispatched. The mobile agent will then operate asynchronously and independent of the sending program.

- **Reliable Event-based Clustering:** The work performs clustering anew for each event. It further introduces the concept of reliability of software agents for clustering ISUs which is an essential aspect of communication especially when the data transmitted and aggregated pertains to mission critical applications.

- **Route Optimization in the Cluster:** The mobile agent travelling in the cluster makes use of an energy efficient route to visit the optimal number of ISUs in a given cluster, thereby contributing to saving of time in gathering and processing the information about the event.

- **Secure Transmission of Processed Information:** The work filters and fuses the information sensed by the ISUs at the source node itself. This makes only processed information to be transmitted to the sink/gateway. The data being transmitted is encrypted making it immune to various network attacks.

6.3 FUTURE SCOPE

The work contained in the thesis made an attempt to answer the questions which came forward as a result of literature survey. However, while designing the solutions for these questions, some new issues that are still open and can become the subject of further research in the near future were identified and are mentioned below:

- **Fault Tolerance:** Though the four phased approach proposed is self-sufficient to increase the life of the network and ensure optimal clustering, the fact that at any time a certain node can malfunction (due to an unknown and unexplained reason) cannot be ignored. Therefore, fault tolerance should be added to the MC3F2 to increase its productivity.

- **QoS-Aware Communication Protocols:** In order to efficiently support QoS in WSNs, communication protocols need to be designed keeping in mind the platform heterogeneity, specifically the heterogeneity between sensors that are involved in the communication. Therefore, the communication protocols for WSNs should be designed to perceive the service requirement of each type of traffic so that it can be guaranteed a specific service level.

- **Mobile Sensor Nodes:** The research work takes into consideration only static sensor nodes. However, the same can be extended to mobile sensor nodes. Issues and challenges related to mobility of motes account for the future scope of this work.

- **Various Data Models:** The work has primarily been designed and implemented for event driven application of WSNs. In the future, it can be extended to meet the requirements of poll driven applications and query driven applications also.

Last but not the least, "Sensors are here to stay and Mobile Agents are here to assist".

REFERENCES

[1] A.K.Sharma, Dimple Juneja, "Evaluation of The Intelligent Multi-Agent Framework of Cellular Networks", Indian Journal of Computing Technology, Vol. 2, No.1, May 2007, pp.9-19.

[2] A.Manjeshwar and D.P.Agarwal, "TEEN: A Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks", in *1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing*, April 2001, pp. 2009-2015.

[3] A.V.Sutagundar and S. S. Manvi. "Location Aware Event Driven Multipath Routing in Wireless Sensor Networks: Agent Based Approach", Egyptian Informatics Journal, Vol.14, No.1,2013, pp. 55-65.

[4] Aarti Singh and Dimple Juneja, "An Improved Design of Contract Net Trust Establishment Protocol", ACEEE International Journal on Communications, Vol. 4, No.1, 2013, pp. 19-25.

[5] Aarti Singh, Dimple Juneja and A. K. Sharma, "Evaluation of Intelligent and Adaptive Multi-Agent Framework for Semantic Web", Global Journal of Computer Science and Technology, Vol. 11, No. 11, July 2011, pp. 41-46.

[6] Adam Dunkels, Bjorn Gronvall, and Thiemo Voigt, " Contiki-A Lightweight and Flexible Operating System for Tiny Networked Sensors", in *29th Annual IEEE International Conference on Local Computer Networks*, 2004,pp. 455-462.

[7] Ahmad A. A. Alkhatib, "A Review on Forest Fire Detection Techniques," International Journal of Distributed Sensor Networks, Vol. 2014, Article ID 597368, 12 pages, 2014. doi:10.1155/2014/597368

[8] Ahmed Abdelgawad and Magdy Bayoumi, "Data Fusion in WSN", Resource-Aware Data Fusion Algorithms for Wireless Sensor Networks, Lecture Notes in Electrical Engineering, , Springer, 2012, pp.17-35.

[9] Alejandro Ribeiro, Ioannis D. Schizas, S. Roumeliotis, and Georgios B. Giannakis, "Kalman Filtering in Wireless Sensor Networks", Control Systems, IEEE, Vol. 30, No.2, 2010, pp. 66-86.

[10] Ali Shareef and Yifeng Zhu, “Localization using Extended Kalman filters in Wireless Sensor Networks”, www.intechopen.org.

[11] Al-Sakib Khan Pathan, Hyung-Woo Lee, and Choong Seon Hong, “Security in Wireless Sensor Networks: Issues And Challenges”, in *8th IEEE International Conference on Advanced Communication Technology (ICACT 2006)*, Vol. 2, 2006, pp. 1043-1048.

[12] Ameer Ahmed Abbasi and Mohamed Younis, “A Survey on Clustering Algorithms for Wireless Sensor Networks”, Journal on Computer Communications, Elseiver, Vol. 30, No.14, 2007, pp. 2826-2841.

[13] Anwar Al Yatama, Kassem Saleh and Nidal Nasser, “A Novel Hierarchical and Heterogeneous Mobile Agent-Based Wireless Sensor Networks (HHMA-WSN) Architecture”, Journal of Next Generation Information Technology(JNIT), Vol.3, No.4, 2012, pp. 1-9.

[14] Aristides Mpitziopoulos, Damianos Gavalas, Charalampos Konstantopoulos and Grammati Pantziou , “Mobile Agent Middleware for Autonomic Data Fusion in Wireless Sensor Networks”, Autonomic Computing and Networking, Springer US, 2009,pp.57-81.

[15] B. Son, Y. Her, and K. Kim, “A Design and Implementation of Forest-Fires Surveillance System based on Wireless Sensor Networks for South Korea Mountains”, International Journal of Computer Science and Network Security, Vol. 6, No. 9, 2006, pp.124–130.

[16] B.Srikanth, M. Harish, and R. Bhattacharjee, “An Energy Efficient Hybrid MAC Protocol For WSN Containing Mobile Nodes”, in *8th IEEE International Conference on Information, Communications and Signal Processing (ICICS)* ,2011,pp.1-5.

[17] Bajaber, Fuad, and Irfan Awan, “Dynamic/Static Clustering Protocol for Wireless Sensor Network”, in *second IEEE UKSIM European Symposium on Computer Modeling and Simulation*(EMS'08),2008, pp.524-529.

[18] Bharat Sundararaman, Ugo Buy and Ajay D. Kshemkalyani, “Clock Synchronization for Wireless Sensor Networks: A Survey”, Journal of Ad Hoc Networks , Vol.3,2005, pp. 281-323.

[19] C. Hartung, R. Han, C. Seielstad and S. Holbrook, “FireWxNet: A Multi-Tiered Portable Wireless System for Monitoring Weather Conditions In Wildland Fire Environments,” in *4th ACM International Conference on*

Mobile Systems, Applications and Services (MobiSys '06), Uppsala, Sweden, June 2006, pp.28–41.

[20] Charalampos Konstantopoulos, Aristides Mpitziopoulos, Damianos Gavalas, and Grammati Pantziou, "Effective Determination of Mobile Agent Itineraries for Data Aggregation on Sensor Networks", IEEE Transactions on Knowledge and Data Engineering, Vol. 22, No.12, 2010, pp. 1679-1693.

[21] Chee-Yee Chong, and Srikanta P. Kumar. "Sensor Networks: Evolution, Opportunities, and Challenges" in *Proceedings of the IEEE* , Vol.91, Issue.8 , 2003, pp.1247-1256.

[22] Chien-Liang Fok, Gruia-Catalin Roman and Chenyang Lu, "Mobile Agent Middleware for Sensor Networks: An Application Case Study" , in *Fourth IEEE International Symposium on Information Processing in Sensor Networks,* 2005, pp. 382-387.

[23] Christian Bettstetter, Hannes Hartenstein, and Xavier Pérez-Costa, "Stochastic Properties of the Random Waypoint Mobility Model", Journal of Wireless Networks,Springer, Vol. 10,No.5 ,2004 , pp. 555-567.

[24] Chuan Zhu, Chunlin Zheng, Lei Shu, and Guangjie Han, "A Survey on Coverage and Connectivity Issues in Wireless Sensor Networks", Journal of Network and Computer Applications Vol. 35, No.2,2012, pp. 619-632.

[25] Chuan-Ming Liu, Chuan-Hsiu Lee, and Li-Chun Wang, "Distributed Clustering Algorithms for Data-Gathering in Wireless Mobile Sensor Networks", Journal of Parallel And Distributed Computing, Vol. 67, No.11, 2007,pp. 1187-1200.

[26] Costin Badaica, Zoran Budimac, Hans-Dieter Burkhard, and Mirjana Ivanovic, "Software agents: Languages, Tools, Platforms", Computer Science and Information Systems, Vol. 8, 2011, pp. 255-298.

[27] Daniel Massaguer, "Multi Mobile Agent Deployment in Wireless Sensor Networks". Dissertation, University of California, Irvine, 2005.

[28] Dario Pompili, Tommaso Melodia, and Ian F. Akyildiz, "Deployment Analysis in Underwater Acoustic Wireless Sensor Networks", in *1st ACM International Workshop on Underwater Networks,*2006, pp. 48-55.

[29] David Braginsky and Deborah Estrin, “Rumor Routing Algorthim For Sensor Networks.” in *1st ACM international workshop on Wireless sensor networks and applications*,2002,pp. 22-31.

[30] David Culler, Deborah Estrin, and Mani Srivastava, “Guest Editors' Introduction: Overview of Sensor Networks”, Journal of Computer, Vol. 37, No.8,2004, pp. 41-49.

[31] Dazhi Chen and Pramod K. Varshney, “QoS Support in Wireless Sensor Networks: A Survey”, in *International Conference on Wireless Networks*. Vol. 233,2004, pp. 1-7.

[32] Dimple Juneja, Atul Sharma, and A.K.Sharma, “A Novel Application of Extended Kalman Filter for Efficient Information Processing In Subsurfaces”, International Journal of Computer Applications(IJCA) Vol. 17, No.2 ,March 2011, pp. 28-32.

[33] Dimple Juneja, Atul Sharma, and A.K.Sharma, “A Query Driven Routing Protocol for Wireless Sensor Nodes in Subsurface”, International Journal of Engineering Science and Technology (IJEST), Vol. 2, No. 6, pp. 1836-1843.

[34] Ehsan Sheybani, “An Algorithm for Denoising and Compression in Wireless Sensor Networks”, http://dx.doi.org/10.5772/48518.

[35] Elahmadi Cheikh, Chakkor Saad,Baghori Moustafa, Hajraoui Abderrahmane, “A New Approach to Improving Lifetime in Heterogeneous Wireless Sensor Networks Based on Clustering Energy Efficiency Algorithm”, Journal of Theoretical and Applied Information Technology, Vol. 61, No. 2, March 2014,pp. 405-412.

[36] F. Ye, A. Chen, S. Liu, L. Zhang, “A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks.” in *Tenth International Conference on Computer Communications and Networks (ICCCN)*, 2001, pp.304-309.

[37] Fadi M Al-Turjman, Hossam S. Hassanein, and Mohamed A. Ibnkahla. “Efficient Deployment of Wireless Sensor Networks Targeting Environment Monitoring Applications”, Journal of Computer Communications, Elseiver, Vol. 36, No.2, 2013, pp. 135-148.

[38] Faruk Bagci Julian Wolf, Theo Ungerer, and Nader Bagherzadeh, “Mobile Agents for Wireless Sensor Networks” in *International Conference on Wireless Networks (ICWN)*, 2009,pp. 502-508.

[39] Fei Jiang, Haoshan Shi, Zhiyan Xu, and Xiangjun Dong "Improved Directed Diffusion-Based Mobile Agent Mechanism For Wireless Sensor Networks" in *fourth IEEE International Conference on Communications and Networking in China, (ChinaCOM 2009),*2009, pp.1-5.

[40] Francesco Aiello, Alessio Carbone, Giancarlo Fortino, and Stefano Galzarano, "Java-Based Mobile Agent Platforms for Wireless Sensor Networks." *in IEEE International Multiconference on Computer Science and Information Technology (IMCSIT),2010*, pp. 165-172.

[41] Francisco Aiello, Fabio Luigi Bellifemine, Giancarlo Fortino, Stefano Galzarano, and Raffaele Gravina ,"An Agent-Based Signal Processing In-Node Environment For Real-Time Human Activity Monitoring based on Wireless Body Sensor Networks", Engineering Applications of Artificial Intelligence Vol.24,No.7,2011, pp. 1147-1161.

[42] Franco Cicirelli, Angelo Furfaro, and Libero Nigro, " Exploiting Agents for Modelling and Simulation of Coverage Control Protocols in Large Sensor Networks", Journal of Systems and Software, Vol. 80, 2007, pp. 1817-1832.

[43] Ghassan Beydoun, Graham Low, and Paul Bogg, " Suitability Assessment Framework of Agent-Based Software Architectures",Journal of Information and Software Technology, Elseiver,Vol.55, No.4, 2013, pp. 673-689.

[44] Giacomo Cabri, Letizia Leonardi, and Franco Zambonelli. "Weak And Strong Mobility In Mobile Agent Applications" in *2nd International Conference and Exhibition on The Practical Application of Java (PA JAVA 2000),* Manchester (UK), 2000.

[45] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. "Energy Conservation in Wireless Sensor Networks: A Survey" Journal of Ad hoc Networks, Elseiver, Vol.7, No.3, 2009, pp. 537-568.

[46] Govind Gupta, Manoj Misra, and Kumkum Garg. "An Energy Balanced Mobile Agents Based Data Dissemination Protocol for Wireless Sensor Networks" in *1st ACM International Conference on Wireless Technologies for Humanitarian Relief*. 2011,pp. 89-95.

[47] Ganesan D. Cerpa and Estrin, D., "Networking issues in Wireless Sensor Networks", Journal of Parallel and Distributed Computing 64.7 (2004): 799-814.

[48] Hairong Dong, “Data-Fusion Techniques and Its Applications” in *Fourth IEEE International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007),* Vol. 2, 2007, pp 442-445.

[49] Hairong Qi, and Feiyi Wang. “Optimal Itinerary Analysis for Mobile Agents in Ad Hoc Wireless Sensor Networks.” in *Proceedings of the IEEE , 2001* , pp. 147-153.

[50] Hairong Qi, Yingyue Xu and Xiaoling Wang. “Mobile-Agent-Based Collaborative Signal and Information Processing in Sensor Networks”, in *Proceedings of the IEEE,* Vol. 91, No.8, 2003 , pp. 1172-1183.

[51] Henry Leung, Sandeep Chandana, and Shuang Wei, “ Distributed Sensing Based on Intelligent Sensor Networks”, in *Circuits and Systems Magazine, IEEE* , Vol 8,No. 2, 2008 pp. 38-52.

[52] H. Yang and S. Lu., “Commutative Cipher Based En- Route Filtering in Wireless Sensor Networks,” Vehicular Technology Conference, 26-29 September 2004, pp. 1223- 1227.

[53] Ian F.Akyildiz,Dario Pompili and Tommaso Melodia, “Underwater Acoustic Sensor Networks: Research Challenges”, Journal of Adhoc Networks,Elseiver, Vol. 3, 2005, pp. 257-279.

[54] Jamal N. Al-Karaki and Ahmed E. Kamal, “Routing Techniques in Wireless Sensor Networks: A Survey.” Wireless communications Magazine, IEEE, Vol.11, No. 6, 2004, pp. 6-28.

[55] Javed Aslam, Qun Li, and Daniela Rus, “Three Power Aware Routing Algorithms For Sensor Networks”, Journal on Wireless Communications and Mobile Computing, Vol.3, No. 2, March 2003, pp. 187-208.

[56] Jennifer Yick, Biswanath Mukherjee and Dipak Ghosal. “Wireless Sensor Network Survey”, Journal of Computer Networks, Elseiver,Vol. 52, No.12, 2008, pp. 2292-2330.

[57] Jeong Hee-Jin, Nam Choon-Sung, Jeong Yi-Seokand Shin Dong-Ryeol, “A Mobile Agent Based Leach in Wireless Sensor Networks.” in *10th IEEE International Conference on Advanced Communication Technology (ICACT 2008),* Vol. 1,2008,pp. 75-78.

[58] Joanna Kulik, Wendi Heinzelman and Hari Balakrishnan, “Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks”, Journal of Wireless networks , Vol. 8 , No.2, 2002, pp. 169-185

[59] Kai Lin, Min Chen, Sherali Zeadally, and Joel JPC Rodrigues, “Balancing Energy Consumption with Mobile Agents in Wireless Sensor Networks”, Journal of Future Generation Computer Systems ,Elseiver, Vol.28 , 2012, pp. 446-456.

[60] Kavi K.Khedo, Rajiv Perseedoss, and Avinash Mungur, “A Wireless Sensor Network Air Pollution Monitoring System”, International Journal of Wireless & Mobile Network(IJWMN), Vol.2, No. 2, May 2010, pp. 31-45.

[61] Kemal Akkaya and Mohamed Younis, “A Survey on Routing Protocols for Wireless Sensor Networks”, Journal of Ad hoc Networks,Elseiver,Vol. 3,No. 3, 2005, pp. 325-349.

[62] Kiran Maraiya, Kamal Kant, and Nitin Gupta. “Study of Data Fusion in Wireless Sensor Network” in *International Conference on Advanced Computing and Communication Technologies (ACCT 2011),* 2011,pp 535-539.

[63] Konstantinos Kalpakis, Koustuv Dasgupta, and Parag Namjoshi, “ Efficient Algorithms for Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks”, Journal of Computer Networks, Elseiver, Vol. 42, 2003, pp. 697-716.

[64] Lei Shu and Min Chen, “Multimedia Over Sensor Networks”, *IEEE COMSOC MMTC E-Letter* 5.5 (2010) pp. 59-61.

[65] Lang Tong, Qing Zhao, and Srihari Adireddy. “Sensor Networks with Mobile Agents” in *IEEE Military Communications Conference, MILCOM'03*, Vol. 1, 2003, pp. 688-693.

[66] Li Wenshuang, Zhu Shanying, Chen Cailian, and Guan Xinping. “Distributed Consensus Filtering Based on Event-Driven Transmission for Wireless Sensor Networks”, in *31st IEEE Chinese Control Conference (CCC),* Heifi, China, July 25-27 2012.

[67] Luca Ferrari “The aglets 2.0. 2 user's manual.” 2010-04. http://aglets. sourceforge. net (2004).

[68] García Villalba, Luis Javier, Ana Lucila Sandoval Orozco, Alicia Triviño Cabrera, and Claudia Jacy Barenco Abbas , “ Routing Protocols In Wireless Sensor Networks”,Journal of Sensors, Vol. 9, No.11, 2009, pp. 8399-8421.

[69] Lutful Karim, Nidal Nasser, and Tarek Sheltami "A Fault Tolerant Dynamic Clustering Protocol of Wireless Sensor Networks" in *IEEE Global Telecommunications Conference(GLOBECOM 2009)*,2009, pp.1-6.

[70] Mahdi Zareei, A. K. M. Muzahidul Islam, Sabariah Baharun, and Shozo Komaki , " Energy-Efficient and Mobility-Aware MAC Protocol for Wireless Sensor Networks" , in *19th IEEE Asia-Pacific Conference on Communications (APCC)*,Bali,Indonesia, 2013,pp. 110-114.

[71] J. Todd McDonald, Alec Yasinac, Willard C. Thompson III, " A Survey of Mobile Agent Security", www.cs.fsu.edu.

[72] Meriem Zaïter, Salima Hacini, and Zizette Boufaïda, " Trust Metrics Identification for Mobile Agents Protection", www.Researchgate.net

[73] Meritxell Vinyals, Juan A. Rodriguez-Aguilar and Jesus Cerquides, "A Survey on Sensor Networks From a Multiagent Perspective", The Computer Journal ,2010, pp. 455–447.

[74] Michael Wooldridge, *An Introduction to Multiagent Systems,* John Wiley & Sons, 2009.

[75] Min Chen, Laurence T. Yang, Taekyoung Kwon, Liang Zhou, and Minho Jo, " Itinerary Planning for Energy-Efficient Agent Communications in Wireless Sensor Networks", IEEE Transactions On Vehicular Technology, Vol. 60, No. 7, 2011,pp.3290-3299.

[76] Min Chen, Taekyoung Kwon, Yong Yuan, and Victor Leung, " Mobile Agent Based Wireless Sensor Networks", Journal of Computers, Vol.1, No.1,April 2006, pp. 14-21.

[77] Min Chen, Sergio Gonzalez and Victor CM Leung, "Applications and Design Issues for Mobile Agents in Wireless Sensor Networks", Wireless Communications, IEEE ,Vol.14, No.6,2007,pp. 20-26.

[78] Min Chen, Taekyoung Kwon, and Yanghee Choi, "Data Dissemination Based on Mobile Agent in Wireless Sensor Networks" in *IEEE Conference on Local Computer Networks (LCN'05)* 2005, pp.527-529.

[79] Mo Haghighi and Dave Cliff , " Sensomax : An Agent-Based Middleware for Decentralised Data Gathering in Wireless Sensor Networks", in *International Conference on Collaboration Technologies and Systems(CTS),* 20th – 24th May 2013 , pp. 10-114.

[80] Mohammad Masdari and Maryam Tanabi, “ Multipath Routing Protocols in Wireless Sensor Networks: A Survey and Analysis”, International Journal of Future Generation Communication and Networking, Vol. 6, No.6, 2013, pp. 181-192.

[81] Mostapha Bendjima and Mohamed Feham, “Intelligent Wireless Sensor Network Management Based on a Multi-Agent System”, International Journal of Computer Science and Applications (IJCST),Vol.3, No. 2, February 2012 ,pp. 9-14.

[82] N. Sandhya Rani, O. Srinivasa Rao and M. H. Prasad, “Efficient Implementation of Data Aggregation in WSNs by Mobile Agent Paradigm”, International Journal of Computer Science & Engineering (IJCSE) ,Vol. 3, No. 9, September 2011, pp. 3254-3258.

[83] Narayan P. Bhosale and Ramesh R. Manza, “Analysis of Effect of Noise Removal Filters on Noisy Remote Sensing Images”, International Journal of Scientific & Engineering Research (IJSER), Vol.4, Issue 10, 2013, pp. 1511-1514.

[84] Ossama Younis, and Sonia Fahmy, “ HEED: a Hybrid, Energy-Efficient, Distributed Clustering Approach for Adhoc Sensor Networks”, IEEE Transactions on Mobile Computing, Vol. 3, No. 4 , 2004, pp. 366-379.

[85] P.Medagliani, Marco Martalò, and Gianluigi Ferrari, “Clustered Zigbee Networks with Data Fusion: Characterization and Performance Analysis”, Journal of Ad Hoc Networks, Vol. 9, 2011, Elseiver, pp.1083-1103.

[86] Parveen Gupta, Preeti Sethi and Naresh Chauhan “Improved Leach Protocol in Hierarchical Routing of Wireless Sensor Networks “ in *National conference on “Emerging Trends in Mobile Technology & Security” (ETMTS-11),*Rohtak, India.

[87] Per Svensson and Johan Schubert, “FUSION 2004: The Seventh International Conference on Information Fusion” in *7th International Conference on Information Fusion,* 2006, Elsevier, pp. 342-345.

[88] Pranesh V. Kallapur and Geetha V. “Research Challenges in using Mobile Agents for Data Aggregation in Wireless Sensor Networks with Dynamic Deadlines”, International Journal of Computer Applications(IJCA) ,Vol. 30, No.5 ,September 2011, pp. 34-38.

[89] Pratik K.Biswas, Hairong Qi and Yingyue Xu, “Mobile-Agent-Based Collaborative Sensor Fusion”, Journal of Information Fusion, Elseiver, Vol.9, No.3, 2008, pp. 399-411.

[90] Preeti Sethi , Dimple Juneja and Naresh Chauhan, “ Exploring the Feasibility of Mobile Agents in Sensor Networks in Non-Deterministic Environments”, International Journal of Advancements in Technology(IJoAT), Vol. 1, No.2, October 2010, pp. 296-302.

[91] Preeti Sethi, Dimple Juneja , Naresh Chauhan, “ Computing An Agent’s Itinerary in a Clustered Network: An Energy Efficient Approach” In 3rd IEEE *International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions)”*,Amity University, Noida, 2014, pp. 1-5.

[92] Preeti Sethi, Dimple Juneja and Naresh Chauhan, “A Multi-Agent Hybrid Protocol for Data Fusion and Data Aggregation in Non-Deterministic Wireless Sensor Networks”, in *IEEE International Conference on Information Systems and Computer Networks(ISCON),*GLA University, Mathura, March 2013,pp. 211-214.

[93] Preeti Sethi, Dimple Juneja and Naresh Chauhan, “Hybrid Data Fusion using Software Agents for Event Driven Application of Wireless Sensor Networks” , in *IEEE International Conference on Information Systems and Computer Networks (ISCON),*GLA University, Mathura, March 2014, pp. 109-112.

[94] Preeti Sethi, Dimple Juneja and Naresh Chauhan, “A Mobile Agent based Event Driven Route Discovery Protocol in Wireless Sensor Networks:AERDP”, International Journal of Engineering Science and Technology(IJEST), Vol. 3, No. 12, December 2011, pp. 8422-8429.

[95] Preeti Sethi, Dimple Juneja, Naresh Chauhan,“ Design of a Communication Strategy for Wireless Sensors in Non-Deterministic Environments using Mobile Agents”, International Journal of Computing Academic Research(IJCAR) ,Vol. 3, No. 5, October 2014, pp. 104-117.

[96] Preeti Sethi and Parveen Gupta “Improved LEACH Protocol using Information Processing (LEACH-IP)” in *International Conference on Advancements in Computer Science & Electronics Engineering ,Nov 2011.*

[97] Preeti Sethi, Surbhi Gupta, Dimple Juneja, Naresh Chauhan, "Software Agent Based Forest Fire Detection : A Novel Approach ", International Journal of Emerging Technologies in Computational and Applied Sciences(IJETCAS) ,March-May 2014, pp. 476-480.

[98] Preeti Sethi, Simran "A Study of Energy Efficient MAC protocols in WSN" in *International Conference on Universal Trends and Technovations in Computer Science, March 2012,*Sri Ram College of Engineering and Management(SRCEM), Palwal

[99] Pu Wang, and Ian F. Akyildiz. "Effects of different Mobility Models on Traffic Patterns in Wireless Sensor Networks", *in IEEE Global Telecommunications Conference (GLOBECOM 20102010),* pp.1-5.

[100] Qing Ling, Yinfei Fu, and Zhi Tian. "Localized Sensor Management for Multi-Target Tracking in Wireless Sensor Networks", Journal of Information Fusion, Vol.12, 2011, pp. 194-201.

[101] Qing Ling, Yinfie Fu and Zhi Tian, "Localized Sensor Management for Multi-Target Tracking in Wireless Sensor Networks" Journal of Information Fusion, Vol.12, 2011, pp. 194-201, Elseiver.

[102] Qing'an Ren Yunmin Zhu, Xiaojing Shen, and Enbin Song, " Optimal Sensor Rules and Unified Fusion Rules for Multisensor Multi-Hypothesis Network Decision Systems with Channel Errors", Journal of Automatica, Elseiver,Vol. 45, No.7, 2009, pp. 1694-1702.

[103] Qishi Wu, Nageswara SV Rao, Jacob Barhen, S. S. Iyenger, Vijay K. Vaishnavi, Hairong Qi, and Krishnendu Chakrabarty, " On Computing Mobile Agent Routes for Data Fusion in Distributed Sensor Networks", IEEE Transactions on Knowledge and Data Engineering, Vol. 16, No. 6, 2004,pp. 740-753.

[104] Qusay H.Mahmoud " Software Agents: Characteristics and Classification." http://www.cis.uoguelph.ca/~qmahmoud/post/agentsoft.pdf

[105] Radosveta Sokullu, and Cagdas Donertas. "Combined Effects Of Mobility, Congestion And Contention On Network Performance For IEEE 802.15. 4 Based Networks" in *23rd IEEE International Symposium on Computer and Information Sciences(ISCIS'08),*2008, pp. 1-5.

[106] Rajashree V. Biradar, V. C. Patil, S. R. Sawant, and R. R. Mudholkar, "Classification And Comparison of Routing Protocols in Wireless Sensor Networks", UBiCC Journal, Vol. 4, 2009, pp. 704-711.

[107] Rajesh Chaudhary and Sonia Vatta, " Review Paper on Energy-Efficient Protocols in Wireless Sensor Networks" , IOSR Journal of Engineering (IOSRJEN), ISSN (e) 2250-3021, Vol.4, No. 2, February 2014, pp.1-7.

[108] Ramesh Rajagopalan, Chilukuri K. Mohan, Pramod Varshney, and Kishan Mehrotra, "Multi-Objective Mobile Agent Routing In Wireless Sensor Networks", in *The 2005 IEEE Congress on Evolutionary Computation,* Vol. 2, 2005, pp. 1730-1737.

[109] Ramsey Faragher , "Understanding the Basis of The Kalman Filter via a Simple and Intuitive Derivation", IEEE Signal Processing Magazine, Vol. 29,No.5,September 2012,pp. 128-132

[110] RenugaDevi G., Sumitra M.G., "An Analysis on LEACH-Mobile Protocol for Mobile Wireless Sensor Networks", International Journal of Computer Applications(IJCA), Vol.65, No. 21, March 2013

[111] Ruihua Zhang Lin Wang, Shichao Geng, and Zhiping Jia, "A Balanced Cluster Routing Protocol of Wireless Sensor Network", in *IEEE International Conference on Embedded Software and Systems Symposia (ICESS Symposia'08)*,2008,pp. 221-225.

[112] S. Bandyopadhyay and E. Coyle, "An Energy Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks" in *22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003),* San Francisco, California, April 2003, pp. 1713-1723.

[113] S. Lindsey and C. Raghavendra. "PEGASIS: Power-Efficient Gathering in Sensor Information Systems". In IEEE Aerospace Conference 2002, Vol. 3, pp. 1125-1130.

[114] S. Lindsey, C. S. Raghavendra, and K. M. Sivalingam, "Data Gathering Algorithms in Sensor Networks using Energy Metrics", IEEE Transactions on Parallel and Distributed Systems, Vol. 13, No. 9,2002, pp.924-935.

[115] Saamaja Vupp.uturi, Kiran K. Rachuri, and C. Siva Ram Murthy, "Using Mobile Data Collectors to Improve Network Lifetime of Wireless Sensor Networks with Reliability Constraints", Journal of Parallel and Distributed Computing , Elseiver,Vo.70,2010, pp. 767-778.

[116] Sajid Hussain, Elhadi Shakshuki, and Abdul Wasey Matin, "Agent-Based System Architecture for Wireless Sensor Networks", in *20th IEEE International Conference on Advanced Information Networking and Applications(AINA '06)* ,Vol. 2, 2006.

[117] Sandeep K.S. Gupta, Golden Richard, and Loren Schwiebert, *Fundamentals of mobile and pervasive computing*. New York: McGraw-Hill, 2005.

[118] Sartaj Sahni and Xiaochun Xu, "Algorithms for Wireless Sensor Networks", International Journal Of Distributed Sensor Networks Vol.1, Issue 1,2005, pp. 35-56.

[119] Kee-Young Shin, Junkeun Song, JinWon Kim, Misun Yu, and Pyeong Soo Mah., " REAR: Reliable Energy Aware Routing Protocol for Wireless Sensor Networks", in 9th *IEEE International Conference on Advanced Communication Technology*, Vol. 1,2007, pp. 525-530.

[120] Shio Kumar Singh, M. P. Singh, and D. K. Singh, "Routing Protocols in Wireless Sensor Networks–A Survey", International Journal of Computer Science & Engineering Survey (IJCSES),Vol 1, 2010, pp. 63-83.

[121] Soo Young Moon, and Cho Tae Ho.,"Fuzzy Based Assignment Method of Filtering Nodes in Wireless Sensor Networks", Journal of Wireless Sensor Network , 2012, http://dx.doi.org/10.4236/wsn.2012.42006.

[122] Srikanth, B., M. Harish and R. Bhattacharjee "An Energy Efficient Hybrid MAC Protocol for WSN Containing Mobile Nodes", in *8th IEEE International Conference on Information, Communications and Signal Processing (ICICS)* 2011, pp. 1-5.

[123] Suat Ozdemir and Yang Xiao, "Secure Data Aggregation in Wireless Sensor Networks: A Comprehensive Overview", Journal of Computer Networks, Elseiver,Vol. 53, No. 12, 2009, pp. 2022-2037.

[124] Sung-Hwa Hong, Jeong-Min Park, and Joon-Min Gil, "Performance Evaluation of A Simple Cluster-Based Aggregation And Routing In Wireless Sensor Networks", Inter. Journal of Distributed Sensor Networks., 2013: 1-9. DOI: 10.1155/2013/501594

[125] Sunita Jyoti Malik and Suman Mor, " Comprehensive Study of Applications of Wireless Sensor Network", International Journal of

Advanced Research in Computer Science and Software Engineering, Vol.2, Issue 11, November 2012, pp. 56-60.

[126] Suzhen Wang, Jiazhen Wang, and Deyun Peng, “A Method to Evaluate Security Performance of Mobile Agent System” in *IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI'06),2006,* pp. 833-838.

[127] Tian He, He, Tian, John A. Stankovic, Chenyang Lu, and Tarek Abdelzaher “SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks”, in *23rd IEEE International Conference on Distributed Computing Systems*, Providence, RI, May 2003, pp. 46-55.

[128] Th. Arampatzis, John Lygeros, and S. Manesis, “A Survey of Applications of Wireless Sensors and Wireless Sensor Networks” in *13th IEEE Mediterranean Conference on Control and Automation ,* Limassol, Cyprus, June 27-29, 2005,pp.719-724.

[129] Tossaporn Srisooksai, Kamol Keamarungsi, Poonlap Lamsrichan, and Kiyomichi Araki, “Practical Data Compression in Wireless Sensor Networks: A Survey”, Journal of Network and Computer Applications, Elseiver, Vol. 35,2012, pp. 37-59

[130] Vasanthi V., Roman Kumar M. Ajith singh N. And M.Hemalata, “A Detailed Study of Mobility Models in Wireless Sensor Network”, Journal of Theoretical and Applied Information Technology, Vol. 33, No.1, 15th November 2011, pp. 7-14.

[131] Vikrant Vij and Rajesh Mehra. “FPGA based Kalman Filter for Wireless Sensor Networks”, International Journal of Computer Technology and App.lications (IJCTA) ,Vol. 2 ,No.1,2011,pp. 155-159.

[132] Vrinda Gupta and Rajoo Pandey, “Data Fusion and Topology Control In Wireless Sensor Networks” WSEAS Transactions on Signal Processing, Vol.4, No.4, April 2008 pp. 150-172.

[133] Wang Jietai,Xu Jiadong, and Yang Shaojun, “Research on Mechanism of Mobile Agent for Wireless Sensor Networks”, in *IEEE International Conference on Wireless Communications, Networking and Mobile Computing, 2007, (WiCom 2007)* ,2007, pp. 2621-2624.

[134] Waseem Ahmed, Preeti Sethi and Naresh Chauhan, “Secured Directed Diffusion using Mobile Agents”, International Journal of Computer App.lications(IJCA), Vol. 73, No. 21, July 2013, pp. 32-36.

[135] Wei-Bo,Hu Han-Ying, and Fu Wen, “An Improved LEACH Protocol for Data Gathering And Aggregation in Wireless Sensor Networks” in *International Conference on Computer and Electrical Engineering*(ICCEE 2008),2008, pp 398-401.

[136] Wen-Hwa Liao, Yucheng Kao and Chien-Ming Fan, “Data Aggregation in Wireless Sensor Networks Using Ant Colony Algorithm”, Journal of Network and Computer Applications , Elseiver, Vol.31, Issue.4,2008, pp. 387-401.

[137] Wen-Tsai Sung, “Multi-Sensors Data Fusion System for Wireless Sensors Networks of Factory Monitoring Via BPN Technology”,Journal of Expert Systems with Applications, Elseiver, Vol. 37, No.3, 2010,pp. 2124-2131.

[138] Wint Yi Poe and Jens B. Schmitt. “Node Deployment In Large Wireless Sensor Networks: Coverage, Energy Consumption, And Worst-Case Delay” in *Asian Internet Engineering Conference(AINTEC’09),Bangkok, Thailand*, ACM, 2009,pp. 77-84.

[139] Woo-Sung Jung Keun-Woo Lim, Young-Bae Ko, and Sang-Joon Park, “Efficient Clustering-Based Data Aggregation Techniques for Wireless Sensor Networks”, Journal of Wireless Networks, Vol.17,No. 5, 2011, pp. 1387-1400.

[140] Xiaoxia Huang, Hongqiang Zhai, and Yuguang Fang, “Robust Cooperative Routing Protocol in Mobile Wireless Sensor Networks”, IEEE Transactions on Wireless Communications, Vol. 7, No..12, 2008, pp. 5278-5285.

[141] Xin Liu and Prasant Mohapatra, “On the Deployment of Wireless Sensor Nodes”, in *3rd International Workshop on Measurement, Modeling, and Performance Analysis of Wireless Sensor Networks, in Conjunction with the 2nd Annual International Conference on Mobile and Ubiquitous Systems. 2005*., pp: 78-85.

[142] Xinyu Feng, “Design and Analysis of Mobile Agent Communication Protocols”, Doctoral Dissertation Nanjing University, China, 2002.

[143] Yi Zou and Krishnendu Chakrabarty, “Uncertainty-Aware Sensor Deployment Algorithms for Surveillance Applications” in IEEE *Global*

Telecommunications Conference,(GLOBECOM'03),Vol. 5, 2003, pp.2792-2796.

[144] Yousry El-Gamal, Khalid El-Gazzar, and Magdy Saeb, “A Comparative Performance Evaluation Model of Mobile Agent versus Remote Method Invocation for Information Retrieval”, World Academy of Science, Engineering and Technology International Journal of Electrical, Robotics, Electronics and Communication Engineering Vol.1, No.3 2007 pp. 286-291.

[145] Yu-Chee Tseng, Sheng-Po Kuo, Hung-Wei Lee, and Chi-Fu Huang, “Location Tracking In a Wireless Sensor Network by Mobile Agents and Its Data Fusion Strategies”, The Computer Journal, Vol. 47, No.4 ,2004, pp. 448-460.

[146] Zoltan Vincze, Rolland Vida, and Attila Vidacs. “Deploying Multiple Sinks In Multi-Hop Wireless Sensor Networks”, in *IEEE International Conference on. Pervasive Services,2007, pp 55-63.*

BRIEF PROFILE OF RESEARCH SCHOLAR

Preeti Sethi did her M.E.(Computer Science)from Maharishi Dayanand University, Rohtak in 2007 and B.E.(Computer Science) from Maharishi Dayanand University, Rohtak in 2001.She also did her PGDBA(HR) from SCDL, Pune in 2007.Ms. Sethi has over 13 years of experience in teaching B.tech and MCA courses. Her areas of interests include Distributed Operating Systems, Programming Languages(C,C++,Java etc.) and Web Technologies. She has published 15 research papers in various journals and conferences of international fame. She is a recipient of Best paper Award in IEEE Sponsored International Conferences (ISCON'14 & ICRITO'14).Currently, she is working as Assistant Professor in the department of Computer Engineering at YMCAUST,Faridabad.

LIST OF PUBLICATIONS OUT OF THESIS

(i) List of Published Papers

S.No	Title of the paper	Name of Journal where Published	No.	Volume & Issue	Year	Pages
1	Exploring the feasibility of Mobile Agents in Sensor Networks in Non-Deterministic Environments	International Journal of Advancements in Technology (IJoAT)	2	Vol.1	2010	296-302
2	A Mobile Agent Based Event Driven Route discovery Protocol in wireless Sensor Networks : AERDP	International Journal of Engineering Science and Technology (IJEST)	12	Vol . 3	2011	8422-8429
3	Secured Directed Diffusion using Mobile Agents (SDDMA)	International Journal of Computer Applications (IJCA)	21	Vol. 73	2013	32-36
4	Design of a Communication Strategy for Wireless Sensors in Non-Deterministic Environments using Software Agents.	International Journal of Computing Academic Research (IJCAR)	5	Vol. 3	2014	104-117
5	Software Agent Based Forest fire Detection(SAFFD): A Novel Approach”	International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)	3	Vol. 5	2014	476-480

(ii) List of Communicated Papers

S.No	Title of the paper along with volume, Issue No, year of publication	Name of Journal	Present Status	Year
1	Mobility Controlled Communication, Clustering ,Filtering and Fusion Architecture of MC3F2	Indian Journal of science and Technology	Under Processing	2015

LIST OF PUBLICATIONS

International Journals

[1] Preeti Sethi , Dimple Juneja and Naresh Chauhan “Exploring the feasibility of Mobile Agents in Sensor Networks in Non-Deterministic environments” International Journal of Advancements in Technology(IJoAT) Vol 1, No.2 (October 2010) ISSN- 0976-4860.

[2] Preeti Sethi, Dimple Juneja and Naresh Chauhan, “A Mobile Agent Based Event Driven Route discovery Protocol in wireless Sensor Networks : AERDP” International Journal of Engineering Science & Technology (IJEST) VOL-3 No. 12 Dec-2011, ISSN-0975-5462.

[3] Waseem Ahmed, Preeti Sethi and Naresh Chauhan, “Secured Directed Diffusion using Mobile Agents (SDDMA)”, International Journal of Computer Applications(IJCA), Volume 73– No.21, July 2013, ISSN 0975-8887.

[4] Preeti Sethi, Surbhi Gupta, Dimple Juneja and Naresh Chauhan “Software Agent Based Forest fire Detection(SAFFD): A Novel Approach” International Journal of Emerging Technologies in Computational and Applied Sciences, March – May 2014, pp 476-480 ISSN-2279-0055.

[5] Preeti Sethi, Dimple Juneja and Naresh Chauhan, “Design of a Communication Strategy for Wireless Sensors in Non-deterministic Environments using Mobile Agents”, International Journal of Computing Academic Research(IJCAR), Volume 3, Number 5 (October 2014) ,pp. 104-117 ISSN 2305-9184.

[6] Preeti Sethi, Dimple Juneja and Naresh Chauhan “Mobility Controlled Communication, Clustering and Filtering Architecture for Wireless Sensor Networks”, Indian Journal of Science and Technology [Communicated].

[7] Mamta, Preeti Sethi, Dimple Juneja and Naresh Chauhan “Development of Mobile Agents with Aglets(A Java Based Tool)”, International Journal of Innovations and Advancements in Computer Science , Vol. 4,(May 2015), pp.245-251, ISSN 2347-8616.

[8] Mukta, Preeti Sethi, Dimple Juneja and Naresh Chauhan “A Review on Agent Based Data Gathering system in Wireless Sensor Network” International Journal of Innovations and Advancements in Computer Sciences, Vol. 4,(May 2015), pp.85-95, ISSN 2347-8616 .

International Conferences

[1] Preeti Sethi, Parveen Gupta and Naresh Chauhan , "Improved LEACH Protocol using Information Processing (LEACH-IP)" in International Conference on Advancements in Computer Science & Electronics Engineering –Nov 2011.

[2] Preeti Sethi, Dimple Juneja and Naresh Chauhan "A multi-agent hybrid protocol for data fusion and data aggregation in non-deterministic wireless sensor networks " in IEEE conference held on 9th-10th march 2013 at GLA University, Mathura.

[3] Preeti Sethi, Dimple Juneja and Naresh Chauhan "Hybrid Data Fusion using software agents for event driven application of sensor networks " in IEEE conference held on 1st-2nd March 2014 at GLA University, Mathura. *

[4] Preeti Sethi, Dimple Juneja and Naresh Chauhan " Computing An Agent's Itinerary In A Clustered Network: An Energy Efficient Approach" in IEEE International Conference on 8th-10th October 2014(ICRITO-2014) at Amity University Noida *

[5] Mamta, Preeti Sethi, Dimple Juneja and Naresh Chauhan " An Agent Based Solution to Energy sink Hole Problem in Flat Wireless Sensor Networks", in CSI-2015 on 2nd-5th December 2015 at BVICAM New Delhi

[6] Preeti Sethi, Dimple Juneja and Naresh Chauhan "Evaluating Performance of Mobility Controlled Communication, Clustering ,Filtering and Fusion Architecture for WSN" in International Conference on "Recent Trends in Computer and Information Technology Research" on 25th -26th Nov-2015 at BSAITM ,Faridabad.

National Conferences

[1] Parveen Gupta, Preeti Sethi and Naresh Chauhan "Improved LEACH Protocol in Heirarchical Routing of Wireless Sensor Networks" in National conference on "Emerging Trends in Mobile Technology & Security" (ETMTS-11),Rohtak,

**** Best paper award in the track***