Development of Energy and Delay Efficient Protocols for WSAN

Jagadeesh Kakarla

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India

Development of Energy and Delay Efficient Protocols for WSAN

Thesis submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Jagadeesh Kakarla
(Roll: 512CS1010)

under the guidance of

Prof. Banshidhar Majhi

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India
June 2016

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

June 24, 2016

Certificate of Examination

Roll Number: 512CS1010
Name: Jagadeesh Kakarla
Title of Dissertation: Development of Energy and Delay Efficient Protocols for WSAN

We the below signed, after checking the dissertation mentioned above and the official record book (s) of the student, hereby state our approval of the dissertation submitted in partial fulfillment of the requirements of the degree of **Doctor of Philosophy** in **Computer Science and Engineering** at **National Institute of Technology Rourkela**. We are satisfied with the volume, quality, correctness, and originality of the work.

Banshidhar Majhi
Principal Supervisor

Sanjay Kumar Jena
Member, DSC

Suchismita Chinara
Member, DSC

Dipti Patra
Member, DSC

Bheemarjuna Reddy Tamma
External Examiner

Chairperson, DSC

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

June 24, 2016

Supervisor Certificate

This is to certify that the work in the thesis entitled **Development of Energy and Delay Efficient Protocols for WSAN** by **Jagadeesh Kakarla**, bearing roll number 512CS1010, is a record of an original research work carried out by him under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of **Doctor of Philosophy** in **Computer Science and Engineering**. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Banshidhar Majhi
Principal Supervisor

Dedicated To My Family

Acknowledgment

If God brings you to it, he will bring you through it…

Thank you God for showing me the path.

I take this opportunity to thank all those who have contributed in this journey.

Foremost, I would like to express sincere gratitude to my advisor, Prof. Banshidhar Majhi for providing motivation, enthusiasm, and critical atmosphere at the workplace. His profound insights and attention to details have been true inspirations to my research. Prof. Majhi has taught me to handle difficult situations with confidence and courage.

It was indeed a privilege to be associated with Prof. Ramesh Babu Battula for research collaboration. He made my stay at MNIT Jaipur very comfortable. I have learned a lot from his knowledge and enthusiasm to achieve excellence. The kind of research discussions we had, has helped me a lot to shape up this dissertation.

My sincere thanks to Prof. S.K. Rath, Prof. S.K. Jena, Prof. S. Chinara, and Prof. Dipti Patra for their continuous encouragement and valuable advice.

I would like to thank my friends and colleagues at NIT Rourkela for the help they have offered during the entire period of my stay.

Finally, I owe the heartfelt thanks to my parents and in-laws for their unconditional love, support, and patience. Special thanks go to my mother who has supported me a lot to finish this piece of work. Thank you Pradeep for always being there when I wanted you the most. Words fall short to express gratitude to my wife, Siri, who has been the constant source of inspiration to me. I am indeed grateful to you for your support and understanding.

Jagadeesh Kakarla

Abstract

Wireless sensor-actor network (WSAN) is a collection of resource conservative sensors and few resource-rich actors. It is widely used in various applications such as environmental monitoring, battlefield surveillance, industrial process control, and home applications. In these real-time applications, data should be delivered with minimum delay and energy. In this thesis, delay and energy efficient protocols are designed to achieve these objectives. The first contribution proposes a delay and energy aware coordination protocol (DEACP) to improve the network performance. It consists of two-level hierarchical *K-hop* clustering and backup cluster head (BCH) selection mechanism to provide coordination among sensors and actors. Further, a priority based event forwarding mechanism has also been proposed to forward the maximum number of packets within the bounded delay. The simulation results demonstrate the effectiveness of DEACP over existing protocols. In the second work, an interference aware multi-channel MAC protocol (IAMMAC) has been suggested to assign channels for the communication among nodes in the DEACP. An actor assigns the static channels to all of its cluster members for sensor-sensor and sensor-actor coordination. Subsequently, a throughput based dynamic channel selection mechanism has been developed for actor-actor coordination. It is inferred from the simulation results that the proposed IAMMAC protocol outperforms its competitive protocols. Even though its performance is superior, it is susceptible to be attacked because it uses a single static channel between two sensors in the entire communication.

To overcome this problem, a lightweight dynamic multi-channel MAC protocol (DM-MAC) has been designed for sensor-sensor coordination. Each sensor dynamically selects a channel which provides maximum packet reception ratio among the available channels with the destination. The comparative analysis shows that DM-MAC protocol performs better than the existing MAC protocols in terms of different performance parameters. WSAN is designed to operate in remote and hostile environments and hence, sensors and actors are vulnerable to various attacks. The fourth contribution proposes a secure coordination mechanism (SCM) to handle the data forwarding attacks in DEACP. In the SCM, each sensor computes the trust level of its neighboring sensors based on the experience, recommendation, and knowledge. The actor analyzes the trust values of all its cluster members to identify the malicious node. Secure hash algorithm-3 is used to compute the message authentication code for the data. The sensor selects a neighbor sensor which has the highest trust value among its *1-hop* sensors to transfer data to the actor. The SCM approach outperforms the existing security mechanisms.

Keywords: DEACP, Delay, WSAN, Energy, IAMMAC, DM-MAC, Channel, SCM.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

ADC Analog to digital converter

ATIM Ad-hoc traffic indication message

BCH Backup cluster head

CTS Clear to send

DAC Digital to analog converter

DEACP Delay and energy aware coordination protocol

DMMA Dynamic multi-radio and multi-channel MAC

EMI Expected maximum idle time

GPS Global positioning system

HEROP Hierarchical, reliable, and energy efficient routing protocol

HGCP Hierarchical geographic clustering protocol

IAMMAC Interference aware multi-channel MAC

IDS Intrusion detection system

MAC Medium access control

MANET Mobile ad-hoc network

MISS Material for intersection of suspicious sets

MMIMO Multi-channel cooperative multiple-input multiple-output

PRR Packet reception ratio

QoS Quality of service

RTS Ready to send

S-MAC Sensor MAC

SAMBA Suspicious area mark a black hole attack

SCM Secure coordination mechanism

SHA-3 Secure hash algorithm-3

WLAN Wireless local area network

WSN Wireless sensor network

Chapter 1

Introduction

Wireless sensor network (WSN) is a collection of autonomous sensors to monitor the environmental conditions [1]. These sensors coordinate among themselves to collect information from the deployed area and transfer it to a sink/base station. Usually, the sink has higher communication and computation capabilities as compared to the sensors. WSN plays a significant role in various real-time applications such as battlefield surveillance, environmental monitoring, industrial process control, health care monitoring and many more [2]. A typical WSN architecture is shown in Figure 1.1.

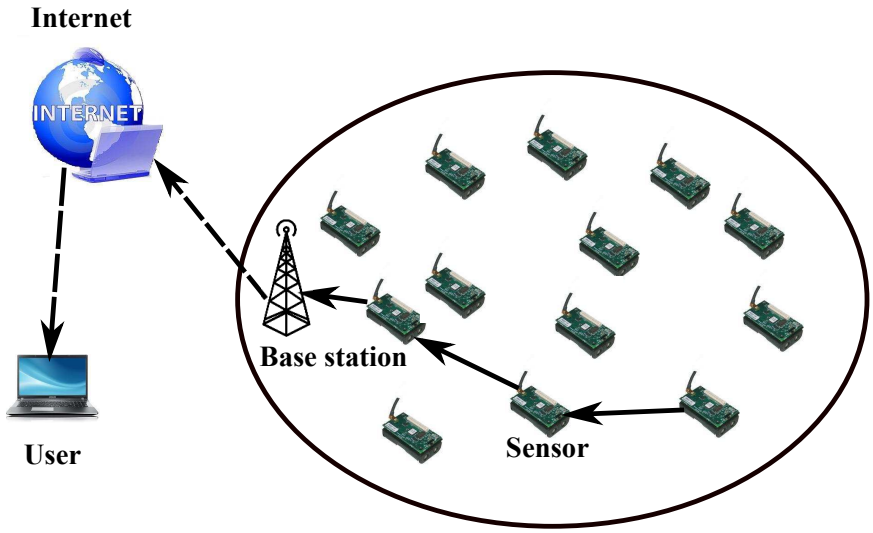


Figure 1.1: Architecture of wireless sensor network

WSN has unique characteristics to discriminate from wireless networks such as mobile ad-hoc networks (MANET) and cellular networks. In WSN, the number of nodes is more as compared to MANET. Path lifetime is also less in WSN due to channel fading, energy depletion, node failure, node addition, and node deletion. Sensors are usually deployed randomly and they configure themselves into a network. In WSN, it is not feasible to have a global addressing mechanism due to high node density. Most of the WSN applications are

data-centric, so data flow in the network exhibits many-to-one traffic pattern [3]. In WSN, sensors collect environmental information and transfer it to the sink, however, they can not perform any actions in the deployed area.

To alleviate this limitation, an expansion of WSN has evolved as wireless sensor-actor network (WSAN) which has actors in addition to the sensors to perform an action in the deployed area [4]. Usually, an actor has higher communication, battery, and computation capabilities as compared to a sensor. It participates in multi-hop communication to transfer and receive data, and typical examples of actors in a WSAN may be water sprinklers, robots, and electrical motors.

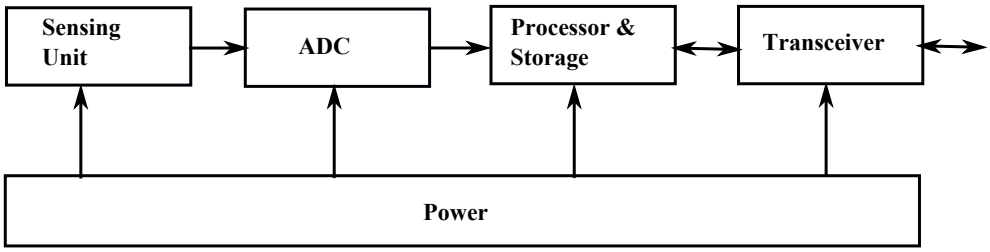


Figure 1.2: Sensor node architecture

A sensor node normally consists of five different components such as sensing unit, analog to digital converter (ADC), processor & storage, transceiver, and power unit (Figure 1.2). A sensor generates an analog signal by sensing the physical area, which is converted into a digital signal using ADC. The digital signal is transmitted to a processor, which in turn consists of micro-controller that performs computing operations. A sensor transfers its data to the destination using a transceiver. The power unit supplies power to all the components in a sensor node [5].

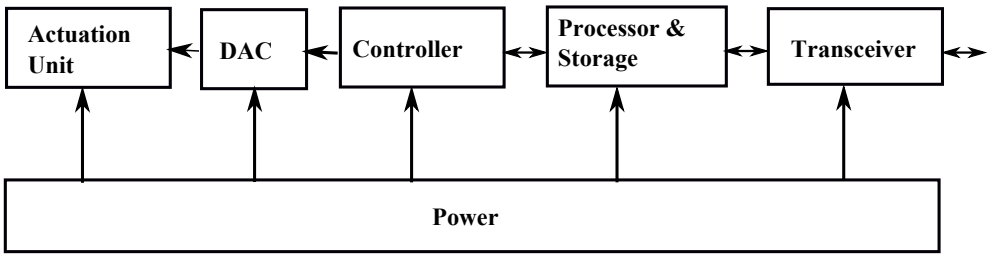


Figure 1.3: Actor node architecture

An actor node consists of six different components: actuation unit, digital to analog converter (DAC), controller, processor & storage, transceiver, and power unit (Figure 1.3). The working principle of power unit, processor & storage, and transceiver is similar to that

of sensor node. The controller unit controls all the components in an actor. The DAC unit converts the digital signal into an analog signal. The actuation unit performs actions in the physical area [6].

1.1 WSAN Applications

WSAN supports various applications. Few of them are described below [7, 8, 9].

- *Environmental Monitoring*: The sensors are used to detect environment conditions such as habitat, air or water quality, hazard, and disaster monitoring. The actor performs an action, if any abnormal event happens in the monitoring area.
- *Military Applications*: In military applications, image sensors are used to detect the presence of enemy targets and tasks. The smart weapons and ambulance can be considered as actors for destroying the targets and rescuing the injured soldiers.
- *Health Care Applications*: Sensors are used to monitor the patient behavior. An actor can take necessary actions based on the patient's health condition.
- *Industrial Process Control*: In industry, sensors are usually deployed to detect any type of faults in the machine. An actor rectifies the faults in a machine.
- *Security and Surveillance*: Video and acoustic sensors are installed in the airports, buildings, and subways to recognize abnormal events. If any abnormal event happens in the monitoring area, then the actor performs actions.
- *Home Intelligence*: WSAN is also used to offer a convenient living environment for human beings.

1.2 WSAN Architecture and Working Principles

It describes how the nodes are organized and communicated with each other to perform network activities efficiently. WSAN consists of automated and semi-automated architectures [10, 11]. In an automated architecture, sensors sense the environmental conditions of the deployed area. The sensed information is directly transferred to an actor in a multi-hop fashion, and the actor performs rapid actions in the target location. The automated architecture improves the network lifetime and delay as information is transferred directly to an actor as shown in Figure 1.4. In a semi-automated architecture, initially sensors send their data to a sink, and the sink processes the collected information.

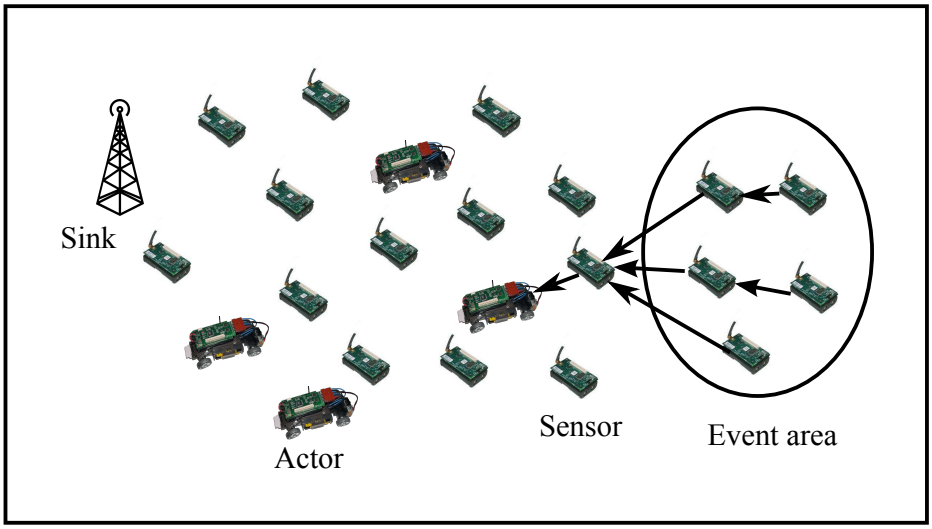


Figure 1.4: Automated architecture of WSAN

Subsequently, it issues the commands to an actor which is nearest to the target location to perform actions. Figure 1.5 shows the semi-automated architecture and its working principle is identical to the traditional WSN architecture. The automated architecture performs well as compared to the semi-automated architecture with respect to network lifetime and delay parameters. Due to inherent advantages of WSAN automated architecture over semi-automated one, more propositions have been made on automated architecture [12]. In this thesis, we have worked in the same direction to design energy and delay efficient protocols in WSAN.

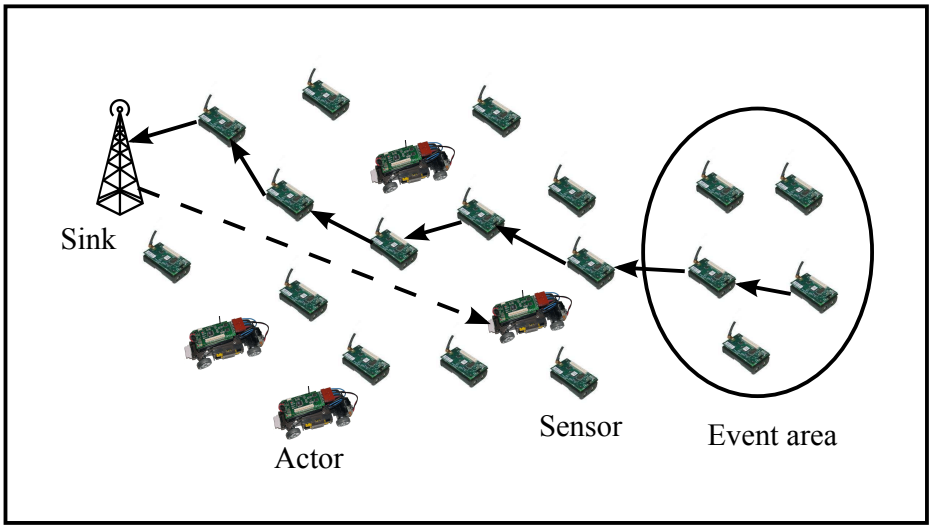


Figure 1.5: Semi-automated architecture of WSAN

WSAN supports three types of data communication modes such as event-driven, periodic, and on-demand [13, 14, 15]. In the event-driven mode, when an event occurs the

sensor transfers its data either to a sink or an actor based on the WSAN architecture. In the remaining time, sensors do not send any information to the sink or actor. Hence, sink/actor does not know whether the sensors are alive or not. The data transmission latency is an important parameter in the event-driven mode. In the periodic mode, sensors periodically transfer their data either to an actor or a sink based on the WSAN architecture. Data gathered in periodic mode does not require quick delivery to the destination. This mode consumes a lot of energy from the sensors as they have to send data periodically. In the on-demand mode, users gather the event information based on their interest. They send instructions to the sink as per their requirements in a specified format. Based on the merits and demerits of the event-driven and periodic mode of data transmission, Manjeshwar *et al.* have proposed a hybrid protocol for efficient information retrieval in sensor networks. It combines the features of event-driven and periodic mode of data transmission [16].

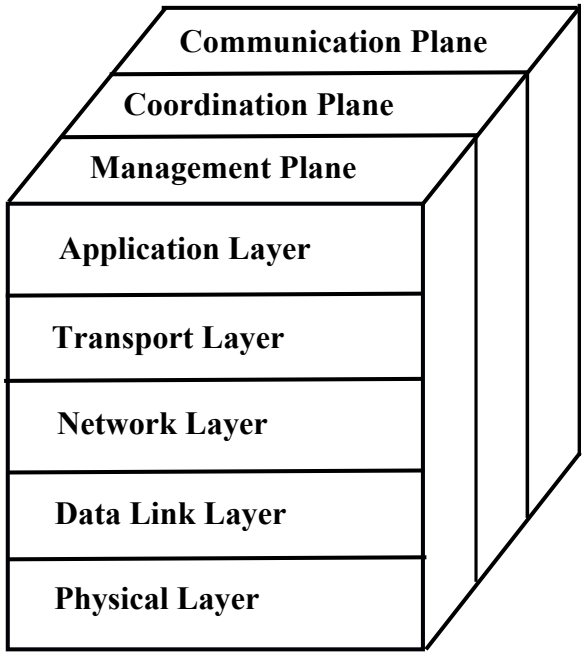


Figure 1.6: WSAN protocol stack

The protocol stack of WSAN consists of five different layers such as physical, data link, network, transport, and application as shown in Figure 1.6. The functionality of each layer is similar to the layers of wireless ad-hoc networks. The application layer provides more operations such as in-network operations, data aggregation, and external query processing. WSAN protocol stack also consists of three planes: management, coordination, and communication. The management plane is responsible for managing the power, actor mobility, and node failure problems. Coordination plane handles coordination among nodes in WSAN and issues instructions to the communication plane for establishing communication in the network [17].

Due to distinctive characteristics of WSAN, existing protocols of wireless sensor networks and ad-hoc networks may not perform well in WSAN [18]. The unique characteristics of WSAN are:

- *Heterogeneity*: The sensors have limited communication resources and battery power. However, an actor has high transmission range, computation, and battery capabilities. Thus, researchers do not give much significance to the energy parameter of actors while designing protocols.
- *Deployment*: A vast number of sensors are thrown in the target area with the help of a helicopter or truck. In addition, few actors with large transmission range and longer battery life are also deployed. The failure of few sensor nodes do not affect the network performance, but the failure of actors are costly.
- *Coordination*: Unlike WSN, WSAN comprises of heterogeneous nodes i.e., sensors and actors. The coordination needs to be three-fold, between sensor-sensor, between sensor-actor, and between actor-actor. Further the coordinations need to be efficient for performing the desired action in the area of deployment.

1.3 WSAN Design Objectives

The unique characteristics of sensor-actor networks and the demand of real-time applications have created a lot of challenges on protocols design in WSAN. The design objectives of sensor-actor networks are [19]:

- *Small Node Size*: Keeping the sensor node size smaller improves the network cost and lifetime.
- *Self Configurability*: In WSAN, nodes to be self configurable to manage effective communication with less power consumption.
- *Adaptability*: In WSAN, path lifetime is less as compared to WSN due to actors mobility and changes in the network density. The protocols of WSAN should be adaptive to network density and actors mobility.
- *Reliability*: To achieve reliability the protocols must support error control mechanisms.
- *Fault Tolerance*: In WSAN, sensors and actors are deployed in a harsh environment. The nodes should hence be fault tolerant.

- *Security*: In real-time applications, sensor and actor nodes perform operations in an unattended area. The adversaries may capture important data from nodes. So, secure protocols are required in WSAN to prevent from active and passive attacks.
- *Quality of service (QoS) Support*: The communication protocols of WSAN should provide QoS support to have high packet delivery ratio and minimum delay for real-time applications.

1.4 Research Challenges and Objectives

Considering the design objectives of a WSAN it reveals that major thrusts need to be given to the coordination mechanisms, medium access control (MAC) protocol design to achieve better QoS parameters, security issues for reliable data delivery etc. It has been observed from the literature that several propositions have already been made [20, 21], however there exists a scope to improve the performance of WSAN by designing improved protocols. Keeping this in mind, the research objectives of the thesis are laid down to

(a) design an energy and delay aware coordination and communication approach to perform reliable actions in an event area, which includes
   - coordination mechanism among sensors and actors to reduce the burden on sensors.
   - a priority based event forwarding mechanism to deliver the maximum number of data packets within the bounded delay.

(b) design an energy efficient multi-channel MAC protocol to improve the network lifetime and channel contention, which contains
   - sleep/wake-up algorithm to reduce energy dissipation in the network.
   - contention based protocol to improve the packet delivery ratio.

(c) design a lightweight distributed multi-channel MAC protocol for sensor-sensor coordination.

(d) design a trust based security model to handle the data forwarding attacks which include black hole, gray hole, and sink hole attacks.

1.5 Thesis Organization

The thesis is organized into six different chapters including introduction and conclusion. The four contributions made out of the thesis are independent and belong to different layers of the WSAN protocol stack. Hence, in place of dedicating a separate chapter for literature survey, the related work is presented separately in each chapter to bring out the motivation for the contribution made.

Chapter 2: A Delay and Energy aware Coordination protocol (DEACP)

A coordination protocol has been proposed to deliver the sensors' information to an actor within the bounded delay. It is a two-level hierarchical *K-hop* clustering algorithm. In the first level, sensors form a *K-hop* cluster by placing actor nodes as cluster heads. In the second level, sink acts as the cluster head and forms a cluster among actors. The sensors which are *1-hop* away from actors are called as *relay* nodes. The actor elects a *relay* node as a backup cluster head (BCH) based on the residual energy and the node degree. The BCH resumes the data gathering process when an actor leaves the cluster to help its neighboring actor. Further, a priority based event forwarding mechanism has been proposed to forward an event information based on its bounded delay. The proposed coordination protocol outperforms its competitive protocols.

Chapter 3: IAMMAC: An Interference aware Multi-channel MAC protocol

The IAMMAC protocol discusses how channels are assigned for the communication among nodes in the DEACP (Chapter 2). An actor acts as a cluster head for *K-hop* sensors and computes the shortest path for all the sensors. An actor partitions the cluster into multiple subtrees and assigns a non-interference channel to each subtree. The actor elects a *relay* node as a backup cluster head (BCH) based on the residual energy and the node degree. An actor broadcasts the BCH information to the remaining *relay* nodes using a common control channel. The *relay* sensors use the same channel of BCH to communicate with it. However, the other cluster members do not change their data channel. Subsequently, an interference and throughput aware multi-channel MAC protocol has been also proposed for actor-actor coordination. The proposed MAC protocol improves the network lifetime, end-to-end delay, packet delivery ratio, and goodput as compared to the existing MAC protocols.

Chapter 4: A Dynamic Multi-channel MAC (DM-MAC) protocol for Sensor-Sensor Coordination

In IAMMAC protocol, a static channel is assigned between two sensors for entire communication to transfer data to the actor (Chapter 3). Even though its performance is

superior, it is susceptible to be attacked because it uses a single static channel between two sensors in the entire communication. To overcome this problem, a lightweight dynamic channel selection mechanism has been proposed for sensor-sensor coordination. Each sensor dynamically selects a channel that has the maximum packet reception ratio among the available channels with the destination. The comparative analysis shows that DM-MAC protocol performs better than the existing MAC protocols in terms of different performance parameters.

Chapter 5: A Secure Coordination Mechanism for Data Forwarding Attacks

A secure coordination mechanism (SCM) has been suggested to handle data forwarding attacks in the DEACP (Chapter 2). Each sensor computes the message authentication code for data using the secure hash algorithm-3 (SHA-3) and shared key (between sensor and actor). The message authentication code is appended to the data and transferred to the actor. The trust value of each sensor is computed based on the three parameters such as experience, recommendation, and knowledge. The sensor selects a *1-hop* sensor which has the highest trust value among its neighbors to deliver the data to an actor. The SCM approach outperforms the existing security mechanisms.

Chapter 2

A Delay and Energy Aware Coordination Protocol

In WSAN, coordination among nodes is required to perform reliable actions in the environment [22, 23]. Coordination is defined as the organization of the different elements of a complex body or activity so as to enable them to work together effectively. In WSAN, coordination among the nodes is divided into three categories: sensor-sensor, sensor-actor, actor-actor coordination. The primary objective of a sensor-sensor coordination is to gather event information in the deployed area with minimum energy usage. Sensor sleep/active mechanism is the primary technique to minimize the number of active sensors in the deployed area. The sensors periodically go to sleep state to reduce the data redundancy and improve the sensors' lifetime. Coordination between a sensor and actor helps the sensor to transfer its data with minimum energy to the nearest actor. Various authors have used cluster based techniques to achieve this objective [24, 25]. Clustering is the process of dividing the nodes into groups, where each group agrees on a central node called as the cluster head. The cluster head gathers the data from all its group members, aggregates the data and sends it to a sink. Further, an actor-actor coordination manages to perform reliable actions in the event area. A single actor can not perform actions independently in the event area, due to its energy and transmission range constraints. Hence, actors coordinate among themselves to perform actions by optimally allocating tasks to each other. The actor-actor coordination has been divided into action-first and decision-first coordination mechanisms. In the action-first coordination, an actor begins the action and then informs it to other actors. The actors are allowed to take their decisions independently whether to join in the action or not. On the other hand, in decision-first coordination, the actor communicates with its neighbor actors before performing any actions in the event area assuming its own constraints.

The rest of the chapter is organized as follows. Section 2.1 describes related work on routing protocols in WSAN to list out their merits and demerits. The proposed delay and energy aware coordination protocol is discussed in Section 2.2. Section 2.3 presents simulation results and analysis. Finally, Section 2.4 summarizes the chapter.

2.1 Related Work on Routing Protocols in WSAN

The essential function of a network layer is to forward the information to the destination [26]. In WSAN, sensors monitor the environment and deliver the data to an actor. An actor processes the sensors' data and performs efficient actions in the deployed area. The design goal of any routing protocol in WSAN needs to be

(a) *Simple*: The routing protocol should be simple and memory efficient because of small sized sensors.

(b) *Energy-efficient*: The routing protocol must consume less energy and should utilize resource-rich actors properly to reduce the communication overhead on sensors.

(c) *Self-organizing and Scalable*: In WSAN, nodes are deployed in a physical area without proper planning. Hence, the routing protocol should be self-organizing. It should be scalable to adapt the changes in node density.

(d) *Distributed*: In large scale sensor networks, distributed routing protocols perform well as compared to centralized mechanisms. Single point failure in a centralized control system reduces the network reliability.

The existing routing protocols of WSAN are broadly classified into cluster based and non-cluster based protocols. The cluster based protocols virtually divide the nodes into groups using their physical properties. The key idea of these protocols is to use the features of actor to minimize the overhead on sensors. The non-cluster based protocols use flooding mechanism to learn about their neighbors. These protocols do not structure the physical network into virtual groups. The working principles of the cluster as well as non-cluster based protocols are discussed below along with their comparative analysis.

2.1.1 Cluster based Routing Protocols

Clustering is defined as the virtual partitioning of the nodes into various groups based on the distance between them [27]. In WSAN, cluster head manages its members in inter-cluster

and intra-cluster routing for proper utilization of resources. The gateway node works as an intermediate node for two cluster heads. The process of clustering is a combination of two phases namely, cluster formation and maintenance. In the cluster formation phase, the sensors are segregated into groups based on their properties. In each group, a sensor acts as a cluster head to manage its group members. The maintenance phase tries to maintain the cluster as long as possible. Different cluster based protocols are described below with their working principles to analyze their relative merits and demerits.

Eduardo *et al.* have designed a hierarchical, reliable, and energy efficient routing protocol (HEROP) [28]. It uses meta-data to create energy efficient clusters. HEROP is a scalable approach which considers sensors energy while transmitting data to them. Hence, it is an energy efficient mechanism. It also provides fault tolerance routing and reliable data transmission in the network. However, HEROP does not consider the node heterogeneity property. The actors mobility control, coordination among actors and sensors are also not addressed properly. A hierarchical geographic clustering protocol (HGCP) has been proposed in WSAN [29]. In HGCP, an area is segregated into virtual grids. The grids are used to distribute the workload optimally among actors. In each grid, a sensor which has the highest residual energy acts as a cluster head. It performs data aggregation and forwards to the closest actor. The reduction in grid area leads to the formation of more clusters and degrades the network lifetime. HGCP does not address the delay parameter properly which is important in real-time applications of WSAN. Finally, it assumes that both the sensors and actors are static.

A quality of service (QoS) aware routing protocol (QARP) has been suggested for WSAN [20]. In QARP, whenever a sensor identifies an event then it checks the subscription table to find out whether any interest on the event is registered or not. If any node is registered for it, then the sensor selects a path to transfer the packet based on its priority. A queuing model has been designed to transfer low priority packets in a less-expensive path to reduce energy consumption in the network. It uses direct diffusion technique to transfer the event information to the actors. QARP considers that both the sensors and actors are static, which is a non-realistic assumption for many WSAN applications. It does not utilize resource-rich actors properly, which causes extra communication burden on sensors and degrades the network lifetime. Tommaso *et al.* have designed an event driven clustering protocol (EDCP) [30], where clusters are generated around an event as it occurs. In the sensor-actor coordination, the actor constructs an aggregation tree for the sensors in its transmission range. A real-time auction protocol has been designed for actor-actor coordination. In the overlapping area, an actor which has the highest residual

energy and also takes less completion time for an action wins the auction. EDCP utilizes actors properly in data communication to reduce the burden on sensors. It also uses greedy routing scheme to improve packet delay in sensor-actor coordination. EDCP does not perform well where multiple events occur concurrently.

Fei *et al.* have proposed a hierarchical energy efficient routing protocol (HEERP) to improve the network lifetime [31]. The network area is divided into domains and each domain has an actor and a set of sensors. A master is selected randomly among the sensors to perform data aggregation. HEERP constructs virtual domains and zones around an actor, which is similar to the hierarchical geographic clustering protocol (HGCP). In HEERP, sensors perform data aggregation process, which degrades the network lifetime. To improve the network lifetime, weighted bi-partite matching protocol (WBMP) employs resource-rich actors as cluster heads [32]. An actor collects the event information from its associated cluster members and performs reliable actions in the event area. To reduce the latency between sensing and acting tasks, the actor maximizes its coverage area based on the sensors density. Further, WBMP does not address the delay parameter effectively. Shahzad *et al.* have suggested a delay and throughput aware protocol (DTAP) to improve the network performance [33]. It consists of static and mobile actors. The network area is segregated into grids and each grid consists a set of static sensors and actors. It tries to find the proper placement of actors to improve the network performance.

Zhiceng *et al.* have developed a sensor-actor coordination protocol (SCP) [34]. In SCP, an actor acts as a cluster head and sends its residual energy to the sink. The sink constructs a weighted actor Voronoi diagram and sends back to the actor. Finally, every actor informs its Voronoi region information to its cluster members. Sensors transmit their data to the actor using shortest path tree to reduce the packet delay. It requires complete topological information and also consumes a lot of energy to calculate the shortest path tree. SCP does not consider sensor-sensor and actor-actor coordination. It assumes that both the sensors and actors are static in nature. A distributed actor positioning and clustering protocol (DAPCP) has been proposed in WSAN [35]. In DAPCP, actors act as cluster heads to minimize the communication burden on sensors. The *k-hop* independent dominating set is used to find the actor's position. It also uses node degree parameter while selecting a cluster head to improve the packet delay. A complete network topological information is essential to compute *k-hop* independent dominating set. It is an energy efficient mechanism as actors are utilized properly in the communication.

2.1.2 Comparative Analysis of Cluster based Routing Protocols

In the previous section, cluster based routing protocols have been discussed with their relative merits and demerits. To derive an overall inference, all the cluster based protocols under consideration have been simulated in a common platform using NS-2 simulator. A radio model has been considered to compute the energy consumption while transmitting and receiving the data as shown in Figure 2.1.

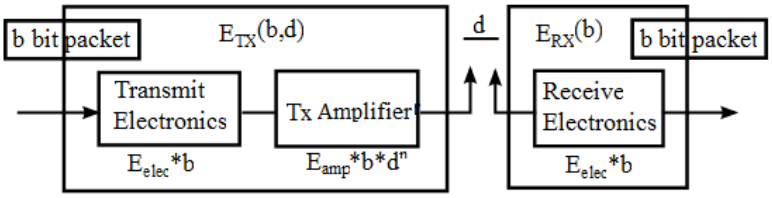


Figure 2.1: Radio energy dissipation model

The free space (E_{fs}) and multi-path fading (E_{mp}) channel models have been utilized depending on the distance between the transmitter and receiver. The free space channel model has been utilized, if the distance between transmitter and receiver is less than threshold d_o, otherwise multi-path channel model has been utilized for communication. The energy required to transmit a $b-bit$ message over the distance d ($E_{TX}(b)$) and to receive the message ($E_{RX}(b)$) are represented as,

$$\begin{aligned} E_{TX}(b) &= E_{TX-elec}(b) + E_{TX-amp}(b,d) \\ &= \begin{cases} bE_{elec} + bE_{fs}d^2, d < d_0 \\ bE_{elec} + bE_{mp}d^4, d \geq d_0 \end{cases} \end{aligned} \tag{2.1}$$

$$E_{RX}(b) = E_{RX-elec}(b) = bE_{elec} \tag{2.2}$$

where, $d_0 = \sqrt{E_{fs}/E_{mp}}$. Electrical energy ($E_{elec}$) depends on digital coding, modulation, and filtering mechanism of the signal. The amplifier energy and $E_{fs}d^2$ or $E_{mp}d^4$ depend on the distance between transmitter and receiver and the acceptable bit-error rate. The simulation parameters like duration of simulation, traffic flow, etc. are listed in Table 2.1, which are used in all protocols. Various performance metrics like average end-to-end delay, average energy dissipation, and packet delivery ratio are used to analyze the performance of the cluster based protocols.

The comparative analysis for these metrics are shown in Figures 2.2 - 2.4. It can be observed that HEROP dominates the other cluster based routing protocols in terms of superior performance in all the three metrics. Even through HEROP is scalable, fault

tolerant, and energy efficient, it does not consider node heterogeneity and actors mobility. Hence, there exists a scope to design new energy efficient cluster based routing protocols in WSAN.

Table 2.1: Simulation parameters for analyzing cluster and non-cluster based routing protocols

Parameters	**Values**
Network area	$1000 \times 1000\ m^2$
Simulation duration	200 s
Traffic flow	CBR
MAC layer	IEEE 802.15.4
CBR packet interval	0.05 s
Number of sensors	100 - 1000
Number of actors	3 - 12
Seed value	0
Actor's mobility speed	0 - 16 m/s
Mobility pattern	Random waypoint
Transmission range of a sensor	100 m
Transmission range of an actor	300 m
Packet size	64 B
Initial energy of a sensor	2J
E_{elec}	$50nJ/bit$
E_{fs}	$10pJ/bit/m^2$
E_{mp}	$0.0013pJ/bit/m^4$

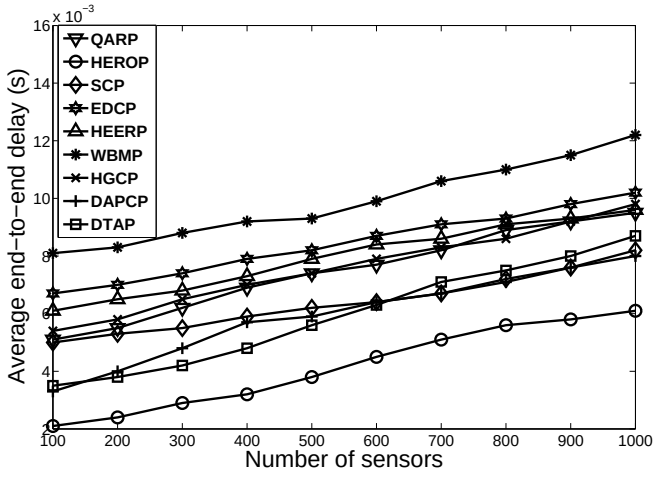


Figure 2.2: Average end-to-end delay for cluster based routing protocols

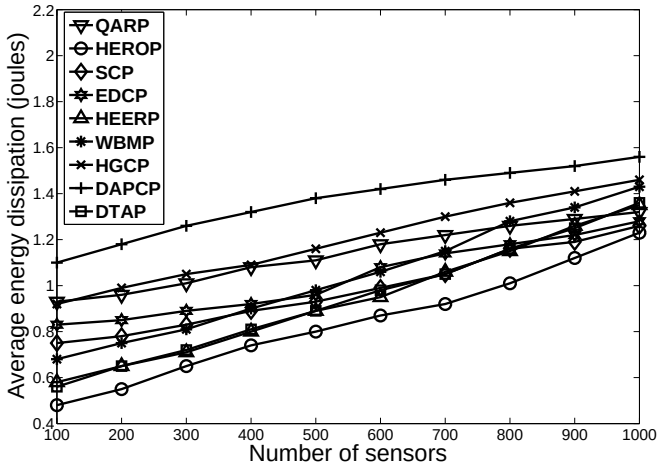


Figure 2.3: Average energy dissipation for cluster based routing protocols

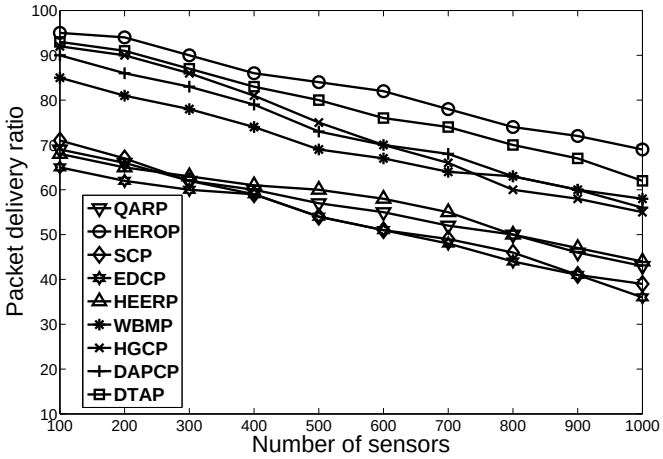


Figure 2.4: Packet delivery ratio for cluster based routing protocols

2.1.3 Non-cluster based Routing Protocols

The non-cluster based routing protocols either use flooding or broadcast mechanisms for communication. They do not structure the physical network into virtual groups. Different non-cluster based routing protocols are described with their working principles to analyze their relative merits and demerits. Durresi *et al.* have proposed a delay and energy aware routing protocol (DEARP) to improve the network performance [36]. DEARP consists of random wake-up scheme and geographic routing. The primary objective of random wake-up scheme is to wake-up a sensor for a specific duration in every time slot. In the geographical routing phase, it uses a greedy mechanism to transfer data to the forwarding candidate set. It provides a loop-free path to the destination for transferring the data, but data may not reach the destination if holes exist in the network. Since WSAN is a dense

network, there is less scope for the existence of holes in a network.

Anycast tree based communication mechanism (ATCM) constructs an anycast tree with its root at the sensor [37]. A sink can dynamically join as well as it can leave the sink tree. In ATCM, every sensor forms an anycast tree. If a sink joins in the network, a new branch is added to the anycast tree. A sensor uses its anycast table for transferring data to the nearest sink. Every sink periodically sends a beacon packet to refresh anycast table entries. ATCM approach is similar to the direct diffusion routing protocol. The anycast table size is controlled by storing only nearest sink information. It performs well when the updates from a sink are not frequent. ATCM mechanism has been simulated using IEEE 802.11 MAC protocol, which has been designed explicitly for WLANs. IEEE 802.11 MAC protocol is not suitable for energy constrained networks namely, WSN and WSAN. Ngai *et al.* have suggested a delay sensitive routing protocol (DSRP) for reliable communication in the network [22]. The network area is segregated into virtual grids for event monitoring. DSRP is a reliability centric framework and uses fault tolerant data aggregation mechanism to eliminate the faulty sensors in the network. DSRP has been simulated using IEEE 802.11 MAC protocol and considered both the sensors and actors are static. The actors are not used properly in the network establishment and data transmission phases. Hence, DSRP creates a lot of communication burden on resource conservative sensors and thus reduces the network lifetime.

Durresi *et al.* have designed a geometric broadcast routing protocol (GBRP) to provide energy efficient packet broadcasting in the network [38]. In GBRP, nodes take local decisions while forwarding data to the destination. It provides low communication overhead as it does not require neighborhood information. The actors are utilized properly to reduce energy consumption in the sensors. GBRP uses separate protocols to handle the broadcast mechanisms among sensors and actors. GBRP broadcasts packets in the entire network area instead of concentrating on a specific region. Power aware routing protocol (PARP) [39] has two versions and in the first version, every node transmits data using same transmission power. In the second one, a sensor can dynamically adjust its transmission power for data transmission. PARP requires a lot of space to store the large size routing table. It chooses a route which requires less energy while forwarding the data. However, it leads to the degradation of the delay parameter. PARP is not feasible for a dense network as the routing table size increases with the increase in network size.

Power controlled routing protocol (PCRP) forwards the packets in a stateless manner [40]. Each sensor sets its power level based on the distance to the intended

neighbor. In PCRP, the sensor selects a neighbor according to the packet delay deadline and energy required to forward the packet. PCRP needs *2* to *3-hop* neighbors information to compute the packet delay, that causes control packet overhead in dynamic networks. Due to the transmitter power control, a sensor uses small transmitting power to the nearest node. This information may not be sensed by other neighbors that are far away and want to send the packets at the same time. It causes a lot of packet collisions and degrades the network performance. PCRP has been simulated using IEEE 802.11 MAC protocol which has specifically designed for WLAN. IEEE 802.11 MAC protocol is not feasible for energy constrained sensor-actor networks. Fuhrmann has proposed a scalable source routing protocol (SSRP) for sensor-actor networks [41]. SSRP is a reactive protocol and uses a proactive mechanism for the virtual ring construction. In SSRP, the source selects an intermediate node that is nearest to the destination. This type of routing may not always produce shortest paths and also increases the packet end-to-end delay.

Fei has suggested a routing protocol for light monitoring and control application (LMCA) [42]. In LMCA, sensor-sensor coordination and actor-actor coordination is performed in separate channels with different capacity, cost, and reliability. The backhaul nodes are resource-rich and they act as mediators between sensor and actor networks. The sensor network uses a data-centric routing architecture. On the other hand, the actor network uses point-to-point communication to improve the network performance. LMCA uses semi-automated architecture for communication, where the sink collects all the sensor data and takes a decision. The semi-automated architecture incurs high end-to-end delay and rapid energy depletion on the sensors. The inclusion of backhaul nodes also increases the network design complexity.

2.1.4 Comparative Analysis of Non-cluster based Routing Protocols

To derive an overall inference, all the non-cluster based protocols under consideration are simulated using same parameters (Table 2.1) which are used for cluster based protocols. Figures 2.5 - 2.7 show the average end-to-end delay, average energy dissipation, and packet delivery ratio, respectively for all the non-cluster based protocols. DEARP uses a greedy mechanism and assures a loop-free path selection while transferring the data. It provides reliable data transmission, and each sensor uses a periodic wake-up mechanism to improve the network lifetime. The PCRP, DSRP, and ATCM protocols have been simulated using IEEE 802.11 MAC protocol. It has specifically designed for wireless local area network (WLAN) and does not give much emphasis to the energy efficient mechanisms as compared to the sensor networks.

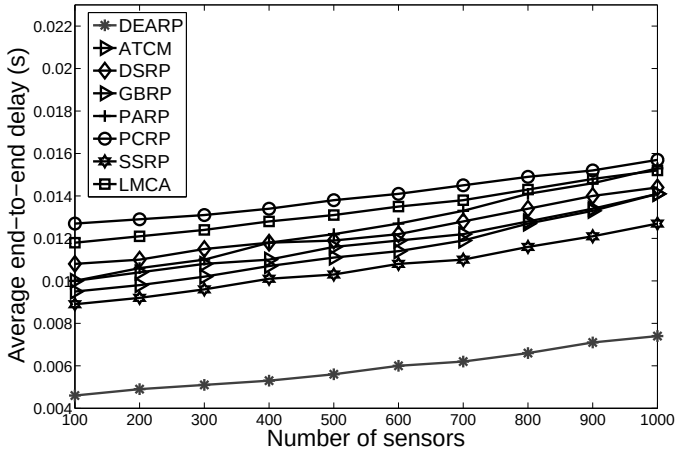


Figure 2.5: Average end-to-end delay for non-cluster based routing protocols

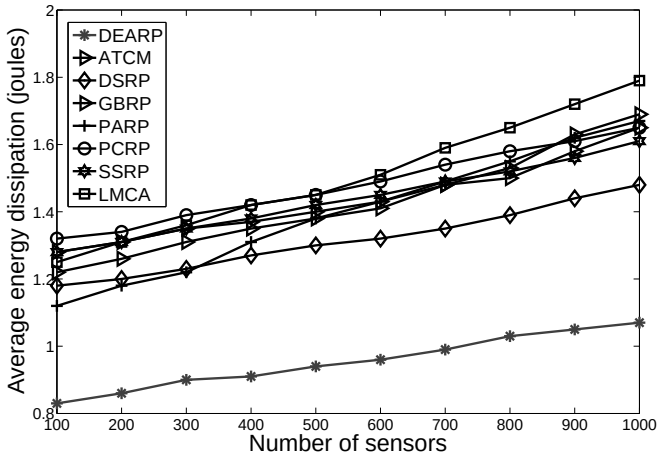


Figure 2.6: Average energy dissipation for non-cluster based routing protocols

SSRP may not always produce the shortest paths and requires complete network topological information. It does not select destination actor properly, which may cause a delay in the data transmission. GBRP is useful for only query-based applications. However, it biases the energy consumption and delay as it uses the broadcast mechanism to transfer the data. LMCA uses a semi-automated architecture, which produces a high delay in the network. DEARP does not specify how to select a destination for border sensors. It requires MAC layer information for calculating the sleep schedule of a sensor and actors mobility is also not considered properly. It can be observed that with respect to all the three metrics under consideration, DEARP outperforms other non-cluster based routing protocols.

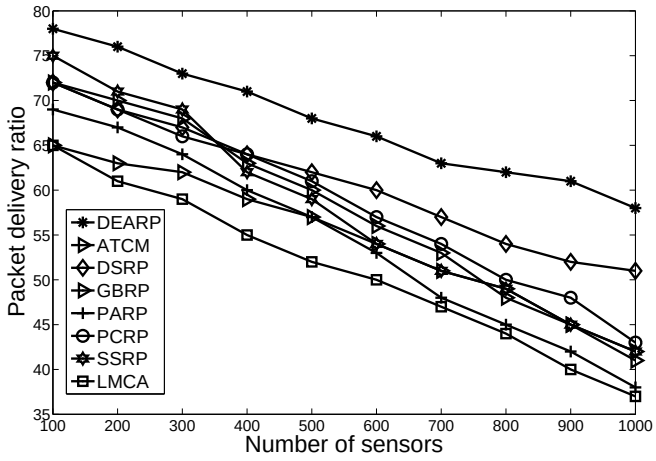


Figure 2.7: Packet delivery ratio for non-cluster based routing protocols

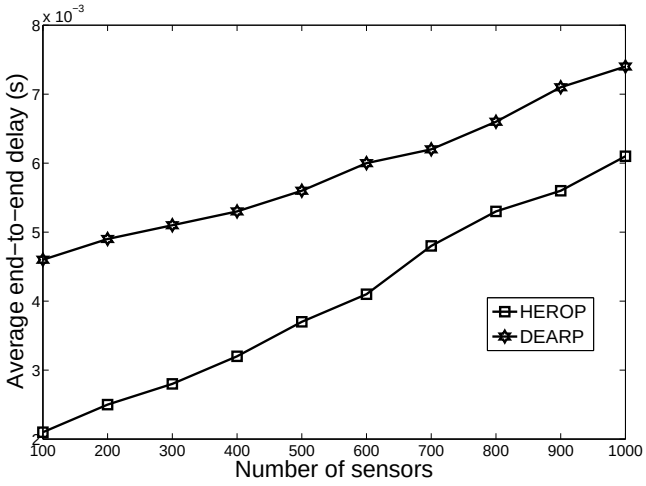


Figure 2.8: Average end-to-end delay of HEROP and DEARP

2.1.5 Comparison of HEROP and DEARP

Amongst the cluster based routing protocols HEROP outperforms others with respect to all the three metrics under consideration. Similarly, DEARP is observed to have superior performance among non-cluster based routing protocols. The two best protocols HEROP from cluster based protocols and DEARP from non-cluster ones are compared to derive an overall inference regarding their performance.

All the three metrics average end-to-end delay, average energy dissipation, and packet delivery ratio performance comparison are shown in Figure 2.8, Figure 2.9, and Figure 2.10. It can be observed that HEROP performs better as compared to DEARP. Hence, cluster based routing protocols have a better scope in WSAN due to their own merits and the

present research directions are witness to it. In this chapter, a cluster based delay and energy aware coordination protocol has been proposed to improve the network lifetime and to deliver the maximum number of packets within the bounded delay.

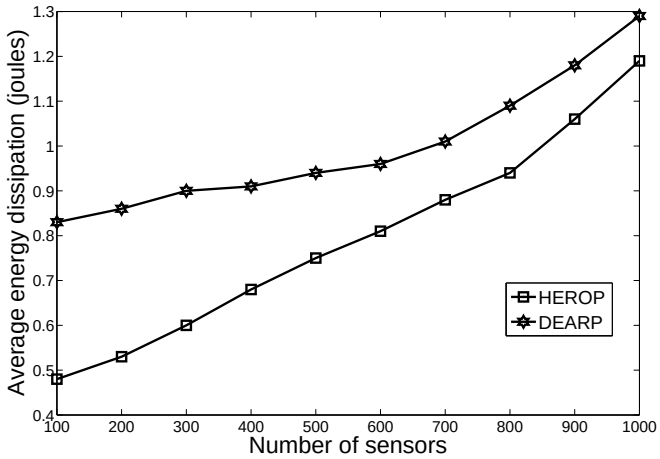


Figure 2.9: Average energy dissipation of HEROP and DEARP

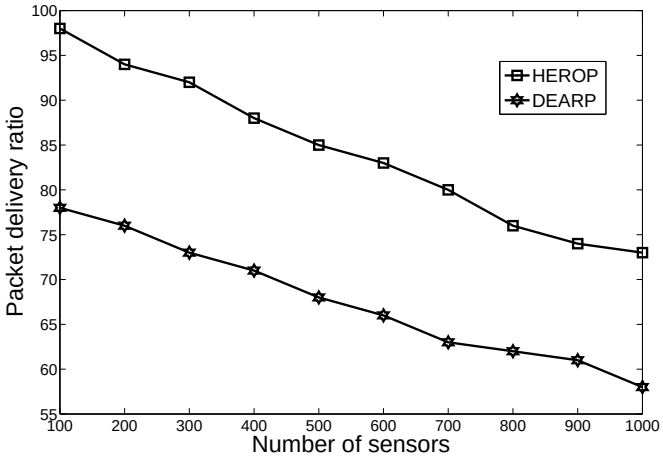


Figure 2.10: Packet delivery ratio of HEROP and DEARP

2.2 Proposed Scheme

The proposed delay and energy aware coordination protocol (DEACP) is a two-level hierarchical *K-hop* clustering. In the first level, sensors form a *K-hop* cluster by placing actors as cluster heads and in the second level, sink acts as the cluster head and forms a cluster among the actors. The sensors which are *1-hop* away from an actor are called as *relay* nodes. The actor elects a *relay* node as a backup cluster head (BCH) based on

the residual energy and node degree. BCH resumes data gathering process when an actor performs the actions or leaves the cluster to help its neighbor actor. Each sensor reports data to the cluster head based on the attribute set defined by the cluster head. The priority based event forwarding mechanism is used to transfer an event information within the bounded delay to improve the packet reliability ratio, average event waiting time, and average energy dissipation in the network.

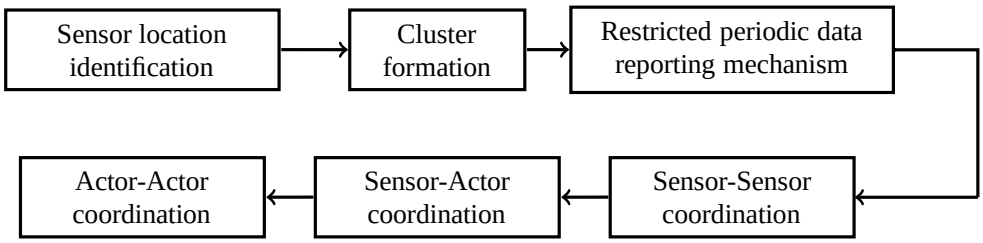


Figure 2.11: DEACP framework

DEACP framework consists of six phases: sensor location identification, cluster formation, restricted periodic data reporting mechanism, sensor-sensor coordination, sensor-actor coordination, and actor-actor coordination as shown in Figure 2.11. A sensor location identification phase is used to estimate the location of sensors based on the received signal strength. The cluster formation phase describes a two-level hierarchical clustering algorithm and backup cluster head (BCH) selection mechanism. BCH selects a cluster head from the *relay* nodes based on the residual energy and node degree. A restricted periodic data reporting mechanism describes when a sensor has to report an event information to the cluster head. The coordination mechanisms deal with effective communication in sensor-sensor, sensor-actor, and actor-actor to fulfill the objective of WSAN.

2.2.1 Sensor Location Identification

In DEACP, a set of static sensors $S = \{S_1, S_2, ..., S_{sn}\}$ are uniformly deployed in an area to detect and track the events. An optimal number of mobile actors $A = \{A_1, A_2, ..., A_{an}\}$ are also deployed at proper positions to improve their coverage area using $k - hop$ independent dominant set algorithm [32]. The sensor location can be obtained by embedding a global positioning system (GPS) device in each sensor, but it consumes a lot of energy. Hence, a GPS device is embedded only in the resource-rich actors. Initially, every actor broadcasts its position and *id* to the sensors in its transmission range. An actor computes the distance to the sensor in its transmission range based on the received signal strength of a reply message from the sensors [29]. The received power at a distance d in free space model is computed

as,

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi^2 d^2 L)} \tag{2.3}$$

where, P_t is the transmission power and λ is wave length. L is system loss factor, G_t and G_r denote transmit and receiver antenna power gains, respectively. In the simulation G_t, G_r, and λ values are defined as 1. The trilateration estimation method is used to compute the locations of the sensors. There are three possible scenarios when computing the location of all the sensors in the proposed network architecture.

1. The sensor node can able to communicate with three actors.
2. The sensor node can able to communicate with at most two actors.
3. The sensor node cannot communicate with any actor.

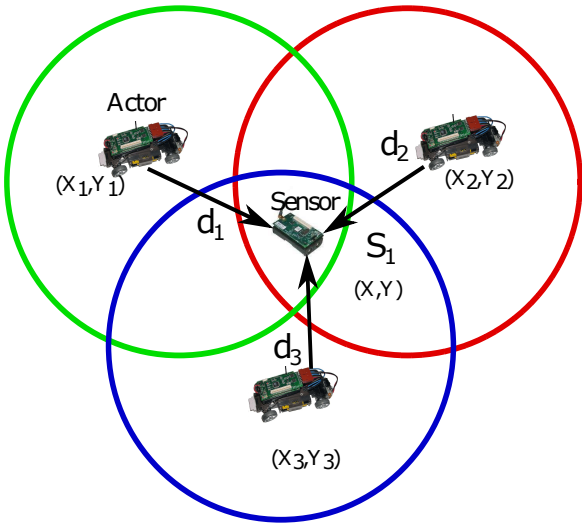


Figure 2.12: Sensor location estimation scenario with three actors

In the first situation, a sensor node can communicate with three actors then the location of the target sensor can be obtained directly using trilateration method. In the other two scenarios, iterative localization mode is used to compute the sensors location. In Figure 2.12, the actors are used to estimate the location of a sensor. The distance between an actor and a sensor is computed (d_1, d_2, d_3) using the received signal strength indication (RSSI) method. It computes the distance between an actor and a sensor based on the received received power of the signal. The distance d is calculated using the Equation

2.3. The location (x, y) of the target sensor can be estimated as,

$$\begin{aligned} d_1^2 &= (x_1 - x)^2 + (y_1 - y)^2 \\ d_2^2 &= (x_2 - x)^2 + (y_2 - y)^2 \\ d_3^2 &= (x_3 - x)^2 + (y_3 - y)^2 \end{aligned} \tag{2.4}$$

$$\begin{aligned} x &= \frac{F_1 y_{32} + F_2 y_{13} + F_3 y_{21}}{2(x_1 y_{32} + x_2 y_{13} + x_3 y_{21})} \\ y &= \frac{F_1 x_{32} + F_2 x_{13} + F_3 x_{21}}{2(y_1 x_{32} + y_2 x_{13} + y_3 x_{21})} \end{aligned} \tag{2.5}$$

where,

$$\begin{aligned} F_1 &= x_1^2 + y_1^2 - d_1^2 \\ F_2 &= x_2^2 + y_2^2 - d_2^2 \\ F_3 &= x_3^2 + y_3^2 - d_3^2 \end{aligned} \tag{2.6}$$

and

$$\begin{aligned} x_{32} &= (x_3 - x_2) \\ x_{13} &= (x_1 - x_3) \\ x_{21} &= (x_2 - x_1) \end{aligned} \tag{2.7}$$

$$\begin{aligned} y_{32} &= (y_3 - y_2) \\ y_{13} &= (y_1 - y_3) \\ y_{21} &= (y_2 - y_1) \end{aligned} \tag{2.8}$$

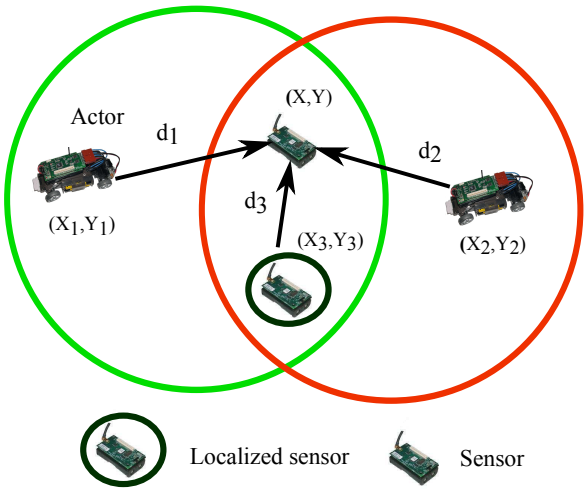


Figure 2.13: Iterative trilateration estimation scenario with at most two actors

Figures 2.13 and 2.14 show the sensor can able to communicate with at most two actors and cannot communicate with any actor scenarios, respectively. In these scenarios, iterative

localization is used to estimate the location of a sensor. In this scheme, the sensors whose location are computed in the first scenario are referred as localized sensors. These localized sensors are used to estimate the location of the sensors that are not reachable to at least three actors by using trilateration technique. This process repeats to compute the location of all the sensors in the network.

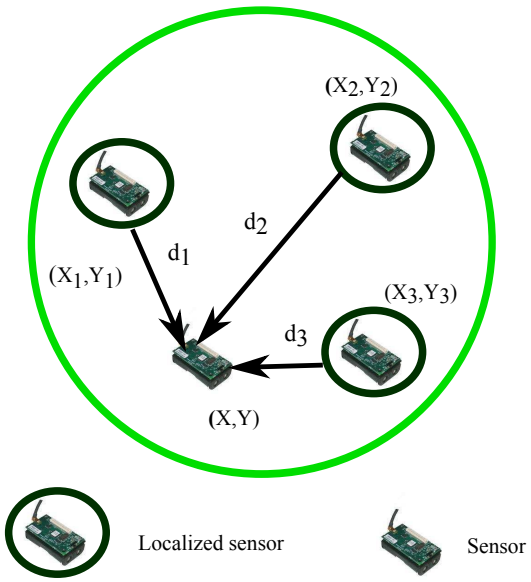


Figure 2.14: Iterative trilateration estimation scenario with localized sensors

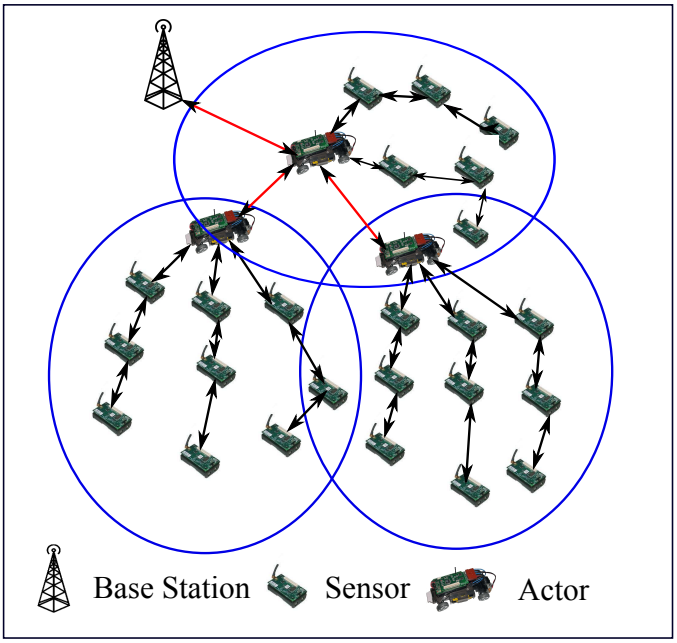


Figure 2.15: DEACP network architecture

2.2.2 Cluster Formation

WSAN is a collection of resource conservative sensors and few resource-rich actors. A two-level hierarchical clustering model has been designed to utilize actors properly and reduce the burden on sensors as shown in Figure 2.15. In the first level, sensors are organized into clusters, where the actor acts as a cluster head and in the second level, the sink acts as a cluster head and forms a cluster among the actors. In every cluster, sensors send their event information to the corresponding cluster head (actor). The actor performs an appropriate action in the event area based on the sensor information.

It is feasible to assign a unique *id* to every actor, as few actors are deployed in the network area. Every actor broadcasts a *Hello* packet to its *K-hop* neighbors that consist of its *id* and *K* value. The value of *K* is defined as a ratio of actor and sensor transmission range. When a sensor node, which is in *1-hop* distance to an actor receives a *Hello* packet, then it stores the actor address as its neighboring address. The sensor sends a *Join* packet to the actor and forwards the *Hello* packet to its *1-hop* sensors. This process repeats until a packet reaches the actors *K-hop* neighbors. If a sensor does not receive a *Hello* packet from any of the actors, then announces itself as a cluster head to the neighboring sensors and transfers the event information to the nearest actor. If a sensor node receives a *Hello* packet from multiple actors, then it sends a *Join* packet to the nearest actor. In our approach, a sensor may be in the communication range of more than one cluster, and such sensors are called as *gateway* nodes. The *gateway* nodes are used to forward the event information from one cluster to another. To handle the mobility of actors, every actor should periodically send their mobility information to the neighbor sensors. The steps followed are given in **Algorithm** 1.

In the second level, the actors form a cluster, where the sink acts as the cluster head. The sink initiates the cluster formation process by forwarding a *Hello* packet to the actors. If an actor receives a *Hello* packet, then it stores the sink address as its neighbor address. The actor sends a *Join* packet to the sink and forwards the *Hello* packet to its *1-hop* actors. This process is repeated until a packet reaches the sink *H-hop* neighboring actors. The value of *H* is defined as a ratio of sink and actor transmission range. In our simulation, the sink transmission range is considered as 1000 m. Whenever an actor is performing an action in the event area, it will forward the information to the neighboring actors and sink. The actor cluster formation process is given in **Algorithm** 2.

The backup cluster head (BCH) selection phase will be enabled, whenever an actor wants to perform action in the event area or leaves the cluster to help its neighboring actors.

Algorithm 1: Sensor cluster formation

```
if node_id ∈ A then
    A_i → S_i : broadcast_Hello(id, K)
    while hop_distance ≤ K do
        accept ← 1
        S_i → forward_to(id, K − −)
    end
end
if node_id ∈ S then
    while hop_distance ≤ K do
        accept ← 1
        S_i → A_i : join_cluster(id, hop_distance)
        S_i → forward_to(id, K − −)
    end
end
```

Algorithm 2: Actor cluster formation

```
if node_id = Sink then
    Sink → A_i : broadcast_Hello(id, 2)
    while hop_distance ≤ H do
        accept ← 1
        A_i → forward_to(id, H − −)
    end
end
if node_id ∈ A then
    while hop_distance ≤ H do
        accept ← 1
        A_i → Sink : join_cluster(id, hop_distance)
        A_i → forward_to(id, H − −)
    end
end
```

The objective of BCH selection phase is to minimize the overall energy consumption and packet drops in the network. In a cluster, the sensors which are *1-hop* away from an actor are called as *relay* nodes $R = \{RS_1, RS_2, ..., RS_{rn}\}$. Before selecting any *relay* node as BCH, the average residual energy (E_{min}) of all the *relay* nodes in a cluster is computed as,

$$E_{\min} = \frac{1}{rn} \sum_{i=1}^{rn} RE_i \tag{2.9}$$

where, *rn* is the number of relay nodes and RE_i is the residual energy of relay node *i*.

The *relay* nodes which have more residual energy than E_{min} are eligible to act as a BCH.

The backup cluster head suitability score (BCH_Score) for a relay node is depends on the residual energy of the relay node (RS_i) and node degree (ND_{RS_i}). The (BCH_Score) is computed as,

$$BCH\_Score_{RS_i} = RE_{RS_i} * ND_{RS_i} \tag{2.10}$$

Among the eligible *relay* nodes, the node which has the highest backup cluster head suitability score is selected as the BCH. Newly elected BCH takes over the role of cluster head and forwards its aggregated data to the actor. The BCH periodically compares its residual energy with E_{min}. If residual energy of the BCH is less than E_{min}, then it leaves the BCH role. The remaining relay nodes perform local election among themselves to elect a BCH. In a worst case, if all the relay nodes are not eligible to act as a BCH; then a sensor which has the highest backup cluster head suitability score has selected as the BCH. The cluster head switching operation takes place in one cluster does not affect regular network operations in other clusters. When the primary cluster head (actor) comes back to its original location then it acts as a cluster head. The BCH selection process is described in the **Algorithm** 3.

Algorithm 3: Backup cluster head selection mechanism

```
1  if node_id ∈ A then
2      A_i → R_j : broadcast_Hello(id)
3      RE_{S_i} ← Residualenergy(S_i)
4      ND_{S_i} ← Nodedegree(S_i)
5      R_j → A_i : (RE_{S_i}, ND_{S_i})
6      Ac_i → r_i : (ACK)
7      foreach Relay node R_j do
8          BCH_Score_{S_i} = RE_{S_i} * ND_{S_i}
9          max ← BCH_Score_{S_i}
10         if max < BCH_Score_{S_{i+1}} then
11             max ← BCH_Score_{S_{i+1}}
12         end
13     end
14     foreach Relay node R_j do
15         I ← max_BCH_Score(id)
16         A_i → R_j : broadcast_Hello(I)
17     end
18 end
```

2.2.3 Restricted Periodic Data Reporting Mechanism

The sensors sense their environment continuously and forward a report to the cluster head in a periodic mode. This mode of data transmission increases data redundancy and consumes a lot of energy from sensors. To overcome this drawback, a restricted periodic data reporting mechanism has been proposed. It is a combination of periodic and event-driven data forwarding modes. In this mechanism, after the cluster formation phase, each actor broadcasts an attribute set which consists of minimum value, minimum difference value, and expected maximum idle time. The attribute set defines when a sensor should forward its data to the cluster head. The parameters are defined as,

Minimum Value (MV): It is the minimum threshold value for a sensed attribute. If the sensed data value is greater than the minimum value then the sensor switch on its transmitter and report the data to its cluster head.
Minimum Difference Value (MDV): It is defined as the minimum difference between two sensed data values. If the difference between two sensed data values is more than MDV then the sensor switch on its transmitter to forward the data to its cluster head.
Expected Maximum Idle (EMI) time : It is the maximum threshold time for which a sensor can be idle.

The MV parameter minimizes the number of transmissions. A sensor reports the data when the sensed attribute is greater than the minimum threshold value. The MDV parameter further reduces the number of transmissions by removing the duplicate data. It allows the sensor to transmit data when the difference between sensed data is more than MDV. If the threshold values of MV and MDV parameters are not reached, then the sensors will never communicate to the cluster head. It creates some confusion to the cluster head whether its cluster member is alive or not. Hence, in this approach, EMI parameter is used to force the sensors to forward the data packet after a minimum threshold period. All these three parameters under consideration are application specific. In WSAN, various sensors may detect a similar event. Hence, it is essential to perform data aggregation before sending it to the actor. The intermediate sensors compute the mean of the received data and forwards to the actor.

2.2.4 Sensor-Sensor Coordination

It is important to gather event information with minimum delay. Hence, the sensor sleep mechanism has been proposed to achieve this objective. A sensor sleep duration depends on its queue utilization. Whenever a sensor goes to the sleep state, it should forward its

sleep period information to the neighboring nodes. The sleep information is useful for the source sensor to identify which sensor is in active state among its *1-hop* neighbors. Every sensor calculates its active duration as,

$$AP_p = \begin{cases} AP_{p-1} * \alpha & if \ \ q_{S_i} > 0 \\ \frac{AP_{p-1}}{\beta} & if \ \ q_{S_i} = 0 \end{cases} \tag{2.11}$$

where, q_{S_i} is the queue size of the sensor S_i. AP_p is the current active period, and AP_{p-1} is the previous cycle active period. The α and β parameters determine the active duration. In the proposed scheme, the value of α and β values are taken as 2 to change the active period linearly. Each sensor maintains destination information, *1-hop* neighbors, sleep/active status, number of packets waiting in the queue, and the residual energy in the routing table as shown in Table 2.2. Each sensor periodically forwards its routing table to the *1-hop* neighbors to update the information about its neighbors.

Table 2.2: Sensor routing table

Destination	*1-hop* neighbor	Sleep/Active	Residual energy	No of packets in queue
Actor1	S_2	Active	0.5 J	2
Actor1	S_6	Sleep	1.8 J	3
Actor2	S_3	Sleep	0.73 J	4

2.2.5 Sensor-Actor Coordination

The primary goal of a sensor-actor coordination is to transfer sensor information to an actor with minimum delay. In DEACP, a sensor transfers its data to the cluster head to improve network lifetime and delay. While transferring data to the cluster head, each sensor uses priority based event reporting mechanism. The objective of this mechanism is to maximize the number of reports reaching the destination within the bounded latency. The priority based event reporting mechanism acts as an index for route selection to meet the bounded latency and computing the sensors active period.

In the proposed DEACP, a priority queue is used to serve the event packets based on their delay requirement. The end-to-end delay is the summation of queuing, transmission, propagation, and processing delays. In a dense network, the queuing delay dominates the end-to-end delay. Hence, the proposed mechanism tries to reduce the queuing delay. The queuing delay depends on the number of high priority packets waiting for transmission across the link. The queue delay increases with the increase in network contention and interference among the wireless links.

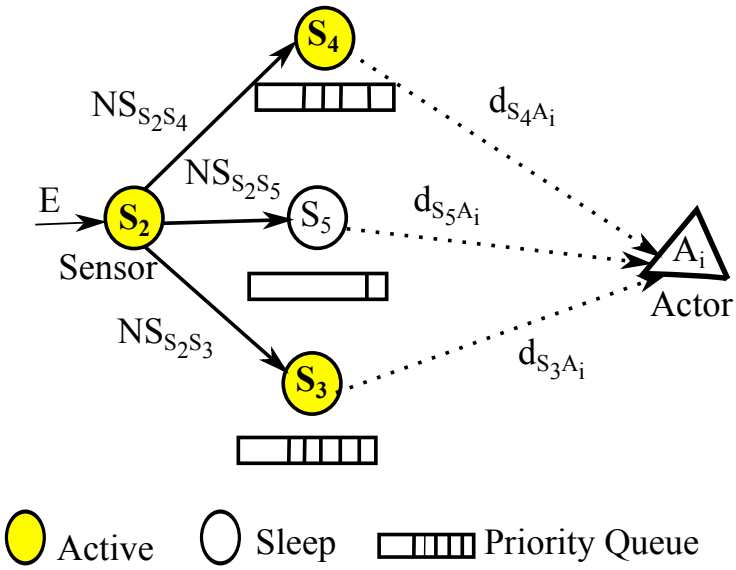


Figure 2.16: Weight graph for sensor-actor coordination

Figure 2.16 shows how sensor S_2 selects a forwarding sensor among its available *1-hop* neighbors; with $d_{S_4A_i}$ being the distance between the sensor S_4 and actor A_i. Each sensor is enabled with one priority queue so that the packets are served based on its delay requirement. Each sensor selects its neighbor which has the maximum neighbor-score (NS) among its *1-hop* neighbors. The neighbor-score between S_i and S_j ($NS_{S_iS_j}$) is computed as,

$$NS_{S_iS_j} = \left(\frac{P(qa_{S_j}) * RE_{S_j}}{d_q(S_j)}\right) \tag{2.12}$$

where, $P(qa_{S_j})$ is the probability of queue availability at sensor S_j, $d_q(S_j)$ is the average waiting time of a packet at sensor S_j, and RE_{S_j} is the residual energy of the sensor S_j. The probability of queue availability at sensor S_j is denoted as,

$$P(qa_{S_j}) = \left(\frac{qs_{S_j} - m_{S_j}}{qs_{S_j}}\right) \tag{2.13}$$

where, qs_{S_j} is the queue size of sensor S_j and m_{S_j} is the number of packets waiting in the sensor S_j queue. Each sensor periodically forwards a control packet which consists of its residual energy, sleep/active status, and the number of packets waiting in a queue. The average waiting time of a packet at sensor S_j is computed as,

$$d_q(S_j) = \bar{S} + R * qs_{S_j} \tag{2.14}$$

where, $\bar{S}$ is the average service time in the sensor S_j and R is the packet arrival rate. The average service time of a packet at a sensor S_j is computed as,

$$\bar{S} = \left(\frac{(\frac{1}{\varepsilon} + \frac{Le}{W})\frac{1}{p_{ac}}}{P_{ac} - 4nI(n)R\frac{Le}{W}} \right) \tag{2.15}$$

where, P_{ac} is the probability that the intended sensor being in active state, Le is the packet size, W is the transmission rate, and ε is the average back-off duration. If any sensor is in the sleep state, then the value of P_{ac}, $\bar{S}$ become zero and infinity, respectively. The neighbor-score of that sensor also becomes zero. The sensors are uniformly distributed in a unit area, the probability that a sensor is in interfering region with its neighbors is computed as,

$$I(n) = \pi.r_{S_j}.(n_{S_j})^2 \tag{2.16}$$

where, r_{S_j} is the radio range of a sensor S_j and n_{S_j} is the number of neighbors for the sensor S_j. Each sensor needs to ensure that the packet end-to-end delay should not be more than the bounded delay. It first calculates the advancement h_{S_i,S_j} towards the actor A_i from S_i to S_j.

$$h_{S_i,S_j} = \frac{d(A_i,S_i) - d(A_i,S_j)}{d(A_i,S_i)} \tag{2.17}$$

where, $d(A_i,S_j)$ is the distance between sensor S_j to actor A_i. The maximum hop-to-hop delay ($delay_{S_i,S_j}$) from S_i to S_j is denoted as,

$$delay_{S_i,S_j} \leq BD_E * h_{S_i,S_j} \tag{2.18}$$

Each intermediate sensor updates the bounded delay of the event E (BD_E) before forwarding the data to the next hop, using the following equation:

$$BD_E = BD_E - (t_{dept} - t_{arr}) - d_{trans} - d_{prop} \tag{2.19}$$

where, $(t_{dept} - t_{arr})$ is the elapse time of the packet in a sensor. d_{trans} can be calculated using transmission rate and d_{prop} is the propagation time. In wireless transmission d_{prop} is in order of microseconds. Packet delay is the summation of queue, transmission, propagation, and processing delays ($delay_{S_i,S_j} = d_q + d_{prop} + d_{trans} + d_{proc}$). The maximum queuing delay d_{q-max} is computed as,

$$d_{q-\max} = BD_E * h_{S_i,S_j} - (d_{trans} + d_{prop} + d_{proc}) \tag{2.20}$$

When the data transmission starts, the sensor updates its $\bar{S}$ and routing table to make sure that the transmission completes in the BD_E. If the bounded delay does not meet, then the sensor has to forward the packets in another route. In worst case, if any alternative route is not found, the sensor informs its previous node to select an alternative route for next transmission.

2.2.6 Actor-Actor Coordination

The actor-actor coordination manages to perform reliable actions in the event area. A single actor can not perform actions independently in the event area, due to its energy and transmission range constraints. Hence, actors coordinate among themselves to perform actions by optimally allocating tasks to each other. The actor-actor coordination mechanisms are divided into action-first and decision-first coordination mechanisms. In the action-first coordination, an actor begins the action and then informs it to other actors. The actors are allowed to take their decisions independently whether to join in the action or not. On the other hand, in decision-first coordination, the actor communicates with its neighbor actors before performing any actions in the event area assuming its own constraints. The action-first scheme performs well as compared to the decision-first with respect to delay parameter. In this work, we have preferred action-first scheme due to its inherent advantage over decision-first scheme.

In the proposed coordination mechanism, whenever an actor receives the event information from its cluster members, it processes the information and updates its event table. The event information is relayed to the sink through its neighboring actors. Each actor and sink maintains an event table as shown in Table 2.3. It consists a list of events, event locations, and actors which are performing actions on the event areas. In a cluster, if multiple events occur simultaneously, then a single actor can not perform actions independently in the event location. In this scenario, the actor seeks the help of its neighboring actors to perform reliable actions in the event area. The actor which is near to primary actor as compared to the available actors will perform actions in the event area. In our proposed mechanism the actor assumes the source sensor location as the event location information.

Table 2.3: Event table

Events	Position	Actor
Event1	(x_{15}, y_{20})	Actor1
Event2	(x_3, y_{12})	Actor2
Event3	(x_{32}, y_6)	Actor3

2.3 Simulation Results and Analysis

To evaluate the performance of the proposed DEACP routing protocol, simulations have been carried out using NS-2 simulator. A radio model is considered to compute the energy

consumption while transmitting and receiving the data which is described in Section 2.1. To perform simulation, 100 - 1000 static sensors are deployed uniformly in a 1000×1000 m^2 network area. IEEE 802.15.4 MAC protocol is used for sensors whereas IEEE 802.11 is utilized for actors. The simulation parameters like data transfer rate, sensors initial energy, sink transmission range are listed in Table 2.4. In the proposed DEACP, it is assumed that sensors are static and actors are semi-mobile. Initially, the actors are deployed at proper positions to improve their coverage area using *k-hop* independent dominant set algorithm [32]. If an event occurs, the actor moves to the target location and performs required action. The actor comes back to its original location after performing actions in the target location. Three simulation scenarios are considered to analyze the performance of the proposed protocol with the two best cluster based routing protocols in WSAN i.e. HEROP [28] and DTAP [33]. Various metrics such as packet reliability ratio, average event waiting time, and average energy consumption in the network are used to investigate the performance of the proposed DEACP protocol, and existing HEROP and DTAP protocols.

Table 2.4: Simulation parameters for DEACP

Parameters	Values
Network Area	1000×1000 m^2
Simulation Duration	200 s
Traffic Flow	CBR
CBR packet interval	0.05 s
Number of Sensors	100 - 1000
Number of Actors	3 - 12
Seed value	0
Sensor's Transmission Range	100 m
Actor's Transmission Range	300 m
Sink transmission range	1000 m
K	3
Packet Size	64 B
Bounded Delay	2 - 3.5 s
Data Transfer Rate	20 pkt/s
Sensor's Initial Energy	2J
E_{elec}	$50nJ/bit$
E_{fs}	$10pJ/bit/m^2$
E_{mp}	$0.0013pJ/bit/m^4$

2.3.1 Simulation Scenario 1

In this scenario, the number of sensors is varied from 100 - 1000 in a step of 100. Each active sensor transfers the data with transfer rate of 20 pkts/s. The performance of the

proposed DEACP is analyzed with packet reliability ratio, average event waiting time, optimal number of actors, and average energy consumption in the network. It is not feasible to deploy a huge number of high cost resource-rich actors in the monitoring area. In the proposed DEACP, an optimal number of actors (A_{opt}) are computed based on the number of sensors and network area [43]. The optimal number of actors in DEACP increases with the increase in number of sensors for fixed size of network area as shown in Figure 2.17. A_{opt} is computed using the following equation.

$$A_{opt} = \left\lceil \sqrt{\frac{N}{2\pi}} \sqrt{\frac{E_{fs}}{E_{mp}}} \frac{M}{d^2{}_{toBS}} \right\rceil \tag{2.21}$$

where, N is the number of sensors and M is the network region. The average distance (d_{toBS}) from a cluster head to the base station is computed as,

$$d_{toBS} = 0.765 \frac{M}{2} \tag{2.22}$$

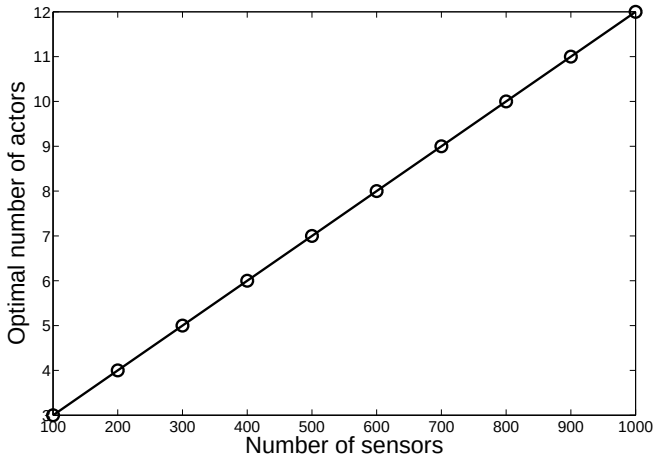


Figure 2.17: Optimal number of actors vs number of sensors for DEACP

Figure 2.18 shows the packet reliability ratio of the proposed DEACP for bounded delay varied from 2 seconds - 3.5 seconds in a step of 0.5. The packet reliability ratio is defined as the ratio of number of packets successfully delivered to an actor within the predefined latency to the total number of packets are forwarded by the sensors. It can be observed that the packet reliability ratio in DEACP increases with the increase in bounded delay and inversely proportional to the network density.

The event waiting time is defined as the difference in time between the occurrence of an event to the starting time of an action performed by an actor. The number of events is

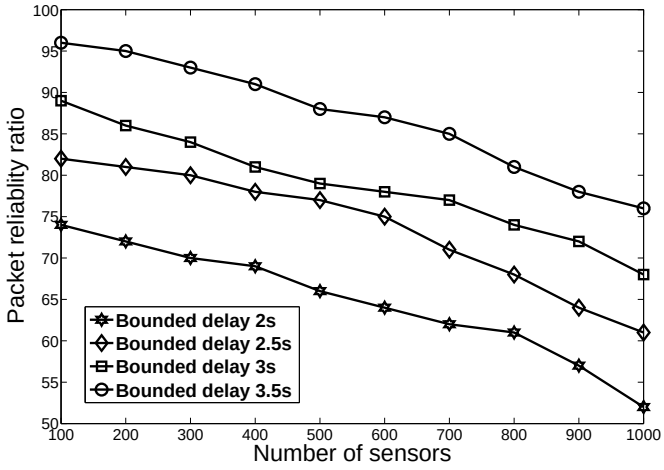


Figure 2.18: Packet reliability ratio of DEACP for various bounded delays

varied from 2 - 6 in a step of 1. The bounded delay for each event is fixed to 2 seconds. The number of actors is varied based on the number of sensors. Figure 2.19 depicts that the average event waiting time in the proposed DEACP is directly proportional to the network density and number of events.

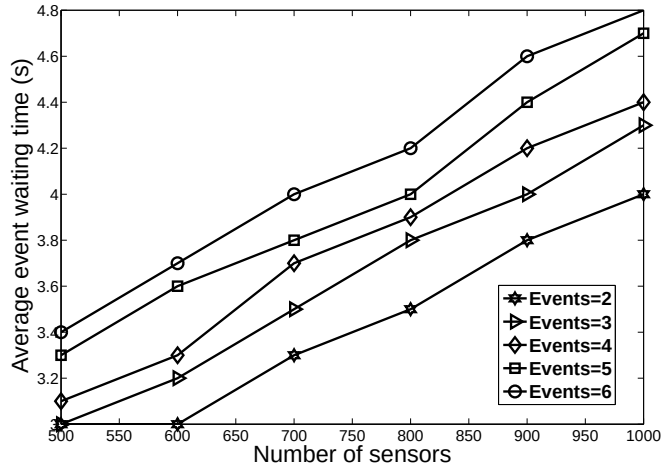


Figure 2.19: Average event waiting time in DEACP with number of events

The energy consumption in the network is defined as the amount of energy consumed to establish the network and to transfer the event information from a source to the destination. Figure 2.20 shows the average energy dissipated by the proposed mechanism in the network under backup cluster head scenario and without backup cluster head scenario. In DEACP, the BCH mechanism reduces the cluster reconfiguration process by switching the cluster head. Whenever the primary cluster head (actor) leaves the cluster, the BCH performs

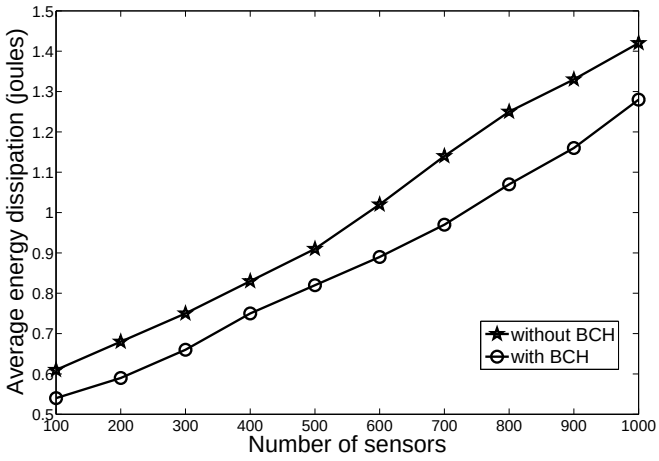


Figure 2.20: Average energy dissipation vs network density for the proposed DEACP

data gathering process from its cluster members. It can be observed from Figure 2.20, that DEACP under backup cluster head scenario consumes less energy as compared to the normal conditions.

2.3.2 Simulation Scenario 2

In this scenario, the performance of the proposed DEACP is compared with the existing HEROP and DTAP cluster based routing protocols. To compare the performance, the number of sensors varied from 100 - 1000 in a step of 100. Each active sensor transfer 20 pkts/s. The optimal number of actors is varied based on the number of sensors. The event bounded delay is fixed to 2 seconds. The network metrics such as packet reliability ratio, average energy dissipation, and average event waiting time are used to analyze the performance of the three protocols under consideration.

Figure 2.21 depicts the packet reliability ratio of the proposed DEACP, and existing HEROP and DTAP protocols for number of sensors varied from 100 - 1000 in a step of 100. In WSAN, a lot of packets will be dropped due to the presence of a large number of sensors, mobility of actors, and network congestion. In DEACP, a restricted periodic data reporting mechanism has been proposed to decrease the data redundancy and traffic flow in the network. The backup cluster head selection mechanism is used to reduce the packet losses in the cluster when a primary cluster head (actor) wants to perform actions in the event area or leaves the cluster to help its neighboring actors. It can be observed from Figure 2.21 that the proposed DEACP achieves 8% more packet reliability ratio as compared to the existing cluster based routing protocols i.e. HEROP and DTAP.

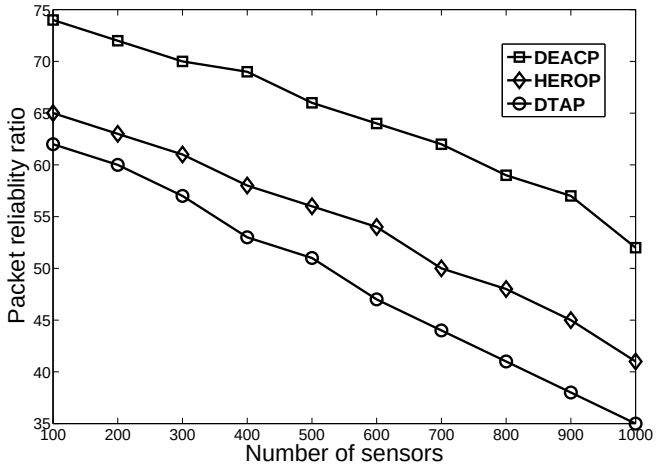


Figure 2.21: Comparative analysis of packet reliability ratio with number of sensors

The average energy dissipation for all the three protocols under consideration is shown in Figure 2.22. Due to a large number of battery constrained sensors, it is crucial to design an energy efficient routing protocol to improve the network lifetime. In DEACP, each sensor goes to sleep state when it does not have any data to send to the neighbors. Further, actor acts a cluster head to reduce the burden on sensors. Hence, the proposed DEACP consumes 27% less energy as compared to the existing routing protocols.

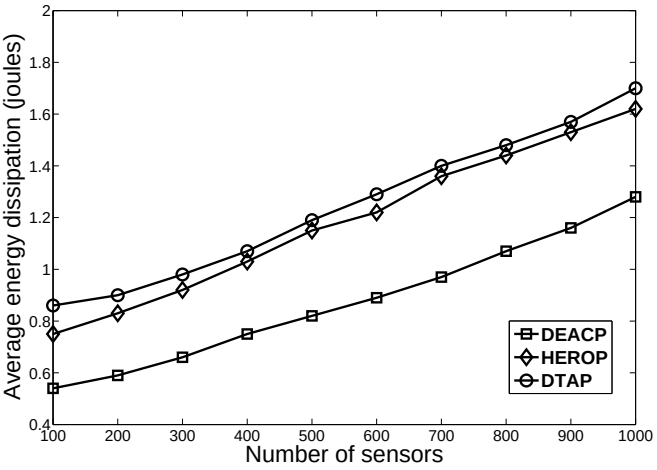


Figure 2.22: Comparative analysis of average energy dissipation with number of sensors

Figure 2.23 depicts the average event waiting time for the proposed DEACP, and existing HEROP and DTAP protocols for number of sensors varied from 100 - 1000 in a step of 100. The number of events is fixed to 2. The average event waiting time increases with the increase in network density for all the three protocols under consideration. In the

proposed DEACP, each sensor transfers its data directly to the cluster members and also considers the delay for transferring data to its *1-hop* sensor before transmitting data to the particular sensor. Hence, the proposed DEACP outperforms achieves 50% less average event waiting time as compared to the existing protocols.

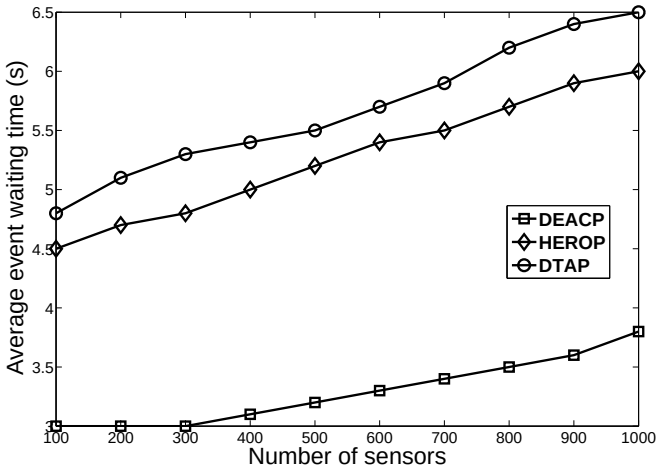


Figure 2.23: Comparative analysis of average event waiting time with number of sensors

2.3.3 Simulation Scenario 3

In this scenario, the data transfer rate varied from 20 - 60 pkts/s in a step of 10 pkts/s. The number of sensors and actors are fixed to 500 and 7, respectively. Two events are randomly generated in the network. The event information should reach to the cluster head (actor) with in the bounded delay of 2 seconds. Figure 2.24 illustrates the average event waiting time for all the three protocols under consideration. It can be observed that the average event waiting time is directly proportional to the data transfer rate and the proposed DEACP performs actions with 25% less delay as compared to the existing HEROP and DTAP protocols.

The comparison of all the three protocols with respect to packet reliability ratio is shown in Figure 2.25. In the proposed DEACP, the backup cluster head mechanism is used to gather the information from cluster members when the primary cluster head (actor) performing actions in the event area or leaves the cluster to help its neighboring clusters. Further, a priority based event reporting mechanism is used to deliver the event information in the bounded delay. Hence, the proposed DEACP achieves 11% high packet delivery ratio compared to its competitive protocols as shown in Figure 2.25.

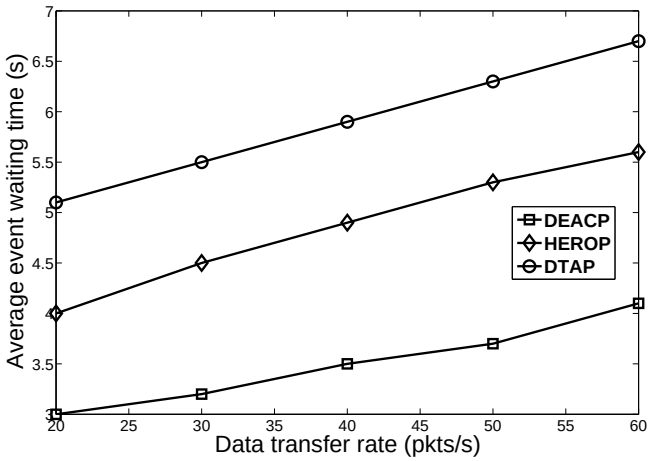


Figure 2.24: Comparative analysis of average event waiting time with data transfer rates

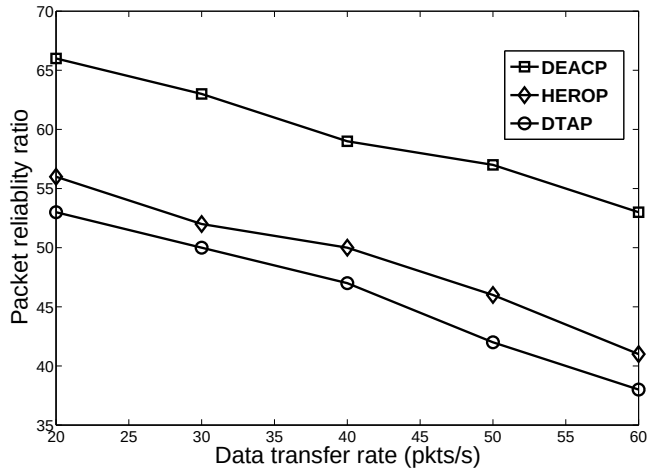


Figure 2.25: Comparative analysis of packet reliability ratio with data transfer rates

Figure 2.26 shows the average energy dissipation in the network for DEACP, HEROP, and DTAP. In DEACP, sensor residual energy is considered in the data forwarding and backup cluster head mechanisms. The actor acts as a primary cluster head to reduce the burden on resource conservative sensors. Each sensor goes to sleep state when it does not have any packets to transfer, which improves the sensors' lifetime. It can be observed that the proposed DEACP consumes 13% less energy as compared to the existing routing protocols.

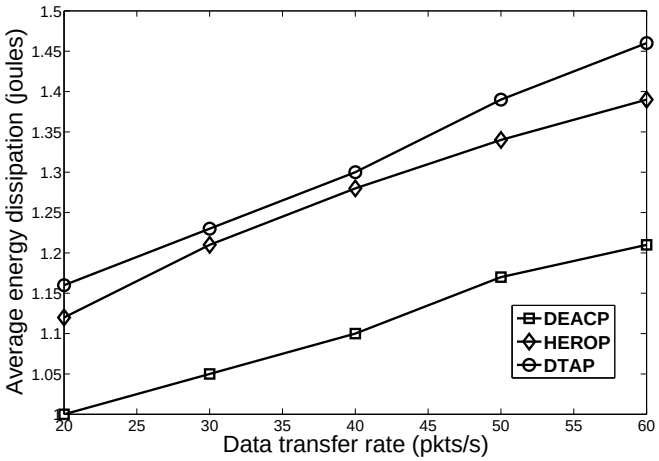


Figure 2.26: Comparative analysis of average energy dissipation with data transfer rates

2.4 Summary

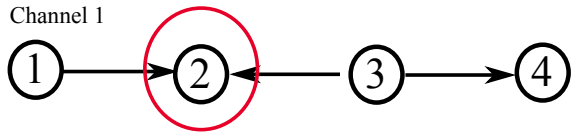
In this chapter, a delay and energy aware coordination protocol (DEACP) has been proposed in WSAN to improve the network performance. The network reliability is closely related to the data freshness and energy efficient data reporting mechanism. Hence, they should be optimized together. In the proposed DEACP, initially an optimal number of actors is calculated based on the number of sensors and network area. An energy efficient two-level hierarchical *K-hop* clustering algorithm has been proposed in WSAN. Whenever an actor leaves its cluster, the backup cluster head (BCH) gathers the event information from its cluster members. The restricted periodic data reporting mechanism reports only few data packets using an attribute set defined by the actor to reduce the energy usage and network congestion. A priority based event forwarding mechanism has been also proposed to deliver the maximum number of packets within the bounded delay.

The performance of the proposed coordination protocol has been evaluated through simulations in NS-2. The results are analyzed using various metrics such as average energy dissipation in the network, packet reliability ratio, and average event waiting time. The simulation results reveal that the proposed coordination protocol outperforms the existing HEROP and DTAP.

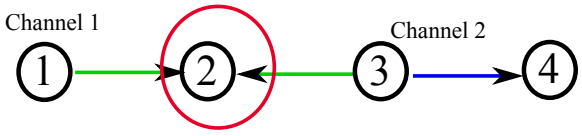
Chapter 3

IAMMAC: An Interference Aware Multi-channel MAC Protocol

Wireless sensor-actor network (WSAN) is a collection of resource conservative sensors and resource-rich actors. Each active sensor traces events in the network area and transfers it to the actor, where actor processes the data and executes efficient actions in the event area. WSAN supports IEEE 802.15.4 medium access control (MAC) standard to provide communication among nodes. IEEE 802.15.4 MAC standard provides 16 non-orthogonal channels, but the existing MAC protocols do not utilize these channels to achieve better performance [44].



(a) Single channel communication



(b) Multi-channel communication

Figure 3.1: Data transmission using single channel and multi-channel

In Figure 3.1(a), nodes 3 and 4 cannot communicate with each other, when nodes 1 and 2 are already in communication mode. According to the IEEE 802.11 standard, the ready to send (RTS)/clear to send (CTS) messages between nodes 1 and 2 block the node 3 from transferring data to node 4. This problem occurs due to the use of a single channel in the

communication [45]. To overcome this problem, various researchers have suggested the use of multiple channels for communication among the nodes [46].

In a multi-channel communication, as depicted in Figure 3.1(b), nodes 3 and 4 can communicate with each other using channel 2, whereas nodes 1 and 2 can communicate with non-interfering channel 1. By using multiple channels, one can achieve a higher throughput in the network than the single channel because multiple transmissions can take place in parallel without any interference [47]. Existing single channel MAC protocols may not perform well in a multi-channel environment because they may create a multi-channel hidden terminal problem in WSAN [48]. This problem occurs where nodes may listen to different channels, that makes it difficult to use virtual carrier sensing mechanism to avoid the hidden terminal problem.

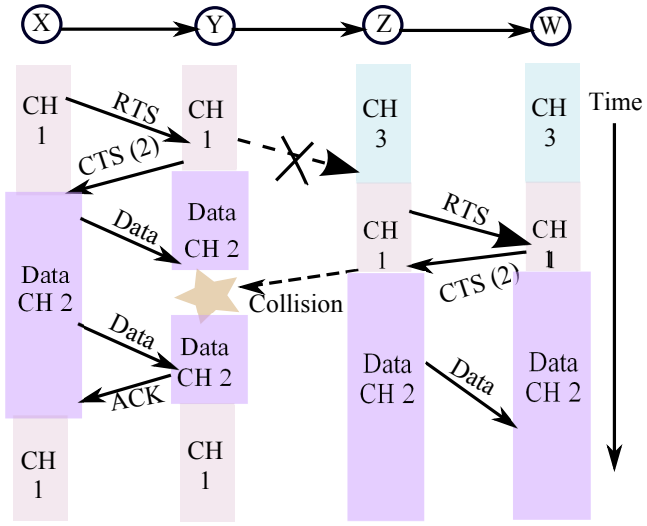


Figure 3.2: Multi-channel hidden terminal problem scenario

To understand the multi-channel hidden terminal problem, let us consider four nodes X, Y, Z and W and three channels CH1, CH2 and CH3 are available for communication among them (Figure 3.2). If node X wants to communicate with Y, then X sends an RTS packet using the channel CH1. Y chooses channel CH2 for transferring the data, and sends a CTS packet to X. These control messages reserve channel CH2 in the transmission ranges of X and Y. However, when node Y sends a CTS packet to X, node Z is busy in listening channel CH3, so it does not hear the CTS packet of node Y. Hence, it does not notice the communication taking place between X and Y on channel CH2. At the same time, if Z initiates the communication with W and selects channel CH2, then a collision will occur at node Y. This problem occurs when a node has a single transceiver and can listen only

to one channel at a given instant of time. To overcome this drawback, various researchers have worked in the direction using multiple transceivers in a sensor [49]. These protocols use a common channel to negotiate for a data channel. Enabling multiple transceivers on a sensor reduces the network lifetime.

The type of antenna chosen for communication also plays a vital role in the performance of a MAC protocol [50]. Generally, antennas are categorized into omnidirectional and directional antennas based on their coverage. The omnidirectional antenna radiates radio wave power uniformly in all the directions. On the other hand, the directional antenna radiates more power in one direction and reduces the interference from unwanted sources. A multi-channel MAC protocol should address issues in channel assignment and medium access mechanism. The channel assignment mechanism decides which channel is used by the node to communicate with its neighbor. The medium access mechanism resolves the collisions using a particular channel [51]. The state-of-the-art research in WSAN reports about receiver and link based channel allocation mechanisms. In a receiver based channel allocation, each node is assigned with a channel to receive packets from its neighbors. In link based channel allocation, every link is assigned to a channel to transfer data along the link. It allows better spatial reuse, due to the flexibility in assigning different channels to different senders [52].

A delay and energy aware coordination protocol (DEACP) has been discussed in the Chapter 2 to deliver the maximum number of packets within the bounded delay. In this chapter, an interference aware multi-channel MAC protocol has been proposed to assign channels for the communication in DEACP. In IAMMAC protocol, the cluster head (actor) divides the cluster into multiple vertex disjoint subtrees and assigns a static channel to each subtree. An actor selects a maximum throughput channel to communicate with its neighboring actor.

The rest of the chapter is organized as follows. Section 3.1 describes various multi-channel MAC protocols available in the literature to list out their merits and demerits. The proposed interference aware multi-channel MAC protocol for DEACP is discussed in Section 3.2. Section 3.3 presents simulation results and analysis. Finally, Section 3.4 summarizes the chapter.

3.1 Related Work

Maximizing the network lifetime is a common objective in sensor networks as sensors are resource conservative. However in WSAN, both packet delay and network lifetime should be considered while designing a MAC protocol. The packet delay can highly impact the performance of WSAN applications. On the other hand, due to the existence of a large number of resource conservative sensor nodes, it is important to design delay and energy efficient MAC protocols. In WSAN, there are four major energy consuming sources at the MAC layer such as collisions, overhearing, control overhead, and idle listening. An efficient MAC protocol should consider these factors to improve the network lifetime.

The existing MAC protocols can be classified into a single channel and multi-channel MAC protocols, based on the number of channels accessible by each node. The single channel MAC protocols suffer from high collisions, network congestion, and hidden terminal problems. These problems degrade the network performance. In the multi-channel MAC protocol, the overall bandwidth is equally divided to n channels. Further, the multi-channel MAC protocols are classified into single transceiver and multi-transceiver multi-channel MAC protocols. In the single transceiver multi-channel MAC protocols, each node can transmit or listen on a single channel at any given instant of time. These protocols may also face the multi-channel hidden terminal problem. Carley *et al.* have proposed a single channel MAC protocol for WSAN [53]. It uses a packet scheduler to provide priority for every node in accessing the channel. Jungmin *et al.* have proposed a multi-channel MAC protocol for ad-hoc networks (MMAC) [54]. The time duration is segregated into slots and each slot is further divided into ad-hoc traffic indication message (ATIM) window and data transmission phase. In the ATIM window, each node transfers its channel negotiation messages in the default channel. In the data transmission phase, a sender transfers its data to the destination using the assigned data channel. Chen *et al.* have proposed a MAC protocol for ad-hoc networks [55]. It is similar to MMAC protocol. However, the time duration slot is variable.

A quality of service (QoS) aware multi-channel MAC protocol has been proposed in sensor networks [46]. It supports dynamic channel assignment mechanism and each sensor node is equipped with a directional antenna. It is suitable for short packet transmission under low traffic networks. A multiple channel reservation MAC protocol has been proposed to tackle the channel conflict problem [56]. It is a fully distributed MAC protocol and does not require time synchronization. Diab *et al.* have proposed a channel allocation mechanism for hybrid multi-channel MAC protocol to improve network

performance in sensor networks [45]. Each sensor uses *3-hop* neighborhood information to select an interference free channel. Computing *3-hop* neighborhood information causes control packet overhead and reduces the network lifetime. Gong *et al.* have proposed a multi-channel cooperative multiple-input multiple-output (MMIMO) MAC protocol in sensor networks [57]. The sensors are organized into clusters, and each cluster head selects few cooperative nodes to forward data to other clusters. For intra-cluster communication, different channels are assigned to adjacent clusters to reduce collisions. In inter-cluster communication, cooperative MIMO links are used to improve the network lifetime and throughput [58]. The multiple transceiver protocols consume a lot of energy. Hence, these protocols do not perform well in the energy constrained sensor networks.

In the multi-radio model, each node consists of two radios to transmit/receive data independently. It improves network performance at the cost of energy consumption. Bahl *et al.* have analyzed the impact of a multi-radio communication model in the network performance [59]. They reveal that a multi-radio platform offers significant benefits for wireless systems. Wang *et al.* have proposed an energy efficient protocol for wireless LAN [60]. An interference aware channel assignment has been proposed for multi-radio wireless mesh networks [61]. It uses a multi-radio conflict graph to model the interference between routers. It is simulated using IEEE 802.11 MAC protocol, which is not suitable for sensor networks. Diab *et al.* have proposed a multi-channel MAC protocol with multi-interface sink in sensor networks [62]. It is an extension to hybrid multi-channel MAC (HMC-MAC) protocol and considers interference caused by the technologies [63]. Liu *et al.* have proposed a dynamic multi-radio and multi-channel MAC (DMMA) protocol for sensor networks [64]. Each sensor dynamically selects a channel based on the spectrum availability. DMMA uses a multi-radio sleeping mechanism to improve the network lifetime.

The multi-radio multi-channel MAC protocols improve the network performance compared to single radio mechanisms but reduces the sensors' lifetime. All the existing MAC protocols do not consider the channel utilization, interference, and capacity. These protocols do not perform well in WSAN, because of its unique characteristics. Hence, there exists a scope to design new multi-channel MAC protocols for WSAN. To address all these issues, in this chapter, an interference aware multi-channel protocol has been proposed to assign channels for sensor-sensor, sensor-actor, and actor-actor coordination in WSAN.

3.2 Interference Aware Multi-channel MAC Protocol

In the proposed interference aware multi-channel MAC (IAMMAC) protocol, the cluster head (actor) divides the cluster into multiple vertex disjoint subtrees and assigns a static channel to each subtree for sensor-actor coordination. In actor-actor coordination, an actor selects a maximum throughput channel to communicate with its neighboring actor. This proposed IAMMAC protocol improves the average packet delay, goodput, packet delivery ratio, and average energy dissipation in the network. The network assumptions for IAMMAC protocol and protocol framework are discussed below in detail.

3.2.1 Network Assumptions

The following assumptions are considered while designing the IAMMAC protocol.

(*a*) Let there be C number of non-orthogonal channels with same bandwidths are available. Out of C channels one channel is used as control channel and $C-1$ channels are used as data channels. The control and data channels are used to transfer control and data messages, respectively.

(*b*) Each sensor node is equipped with a half-duplex transceiver and directional antenna. Hence, a sensor can either transmit or receive data only on a single channel.

(*c*) The actor node is equipped with multiple radios and on each radio T number of channels are available.

(*d*) The sensors are static, but actors are semi-mobile nodes. Initially, the actors are placed in fixed positions. If an event occurs, they move to the event locations, perform the required actions, and come back to their original locations.

3.2.2 IAMMAC Protocol Framework

The IAMMAC protocol framework (Figure 3.3) consists of three phases: channel assignment for sensor-sensor and sensor-actor coordination, a contention based MAC protocol, and channel selection mechanism for actor-actor coordination. The channel assignment for sensor-sensor and actor-actor coordination phase decides which channel is used by the sensor to communicate with its *1-hop* sensor in the vertex-disjoint subtree for transferring the event information to the cluster head (actor). The contention based MAC protocol resolves the collisions while using a particular channel in the vertex-disjoint subtree. In the channel selection mechanism for actor-actor coordination, an actor selects a maximum throughput channel to communicate with its neighboring actor.

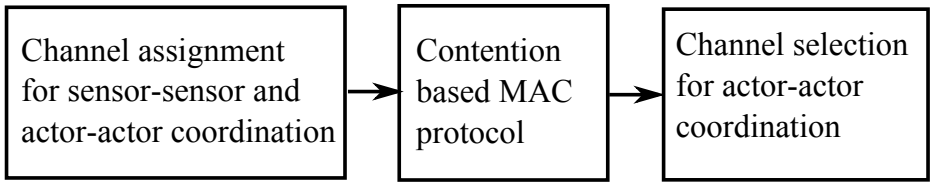


Figure 3.3: IAMMAC protocol framework

Channel Assignment for Sensor-Sensor and Actor-Actor Coordination

A multi-channel MAC protocol should address the problems in channel assignment and medium access mechanism. The channel assignment mechanism decides which channel is used by the node to communicate with its neighbor. The medium access mechanism resolves the collisions when using a particular channel. The proposed IAMMAC protocol uses the link based channel access mechanism and contention based MAC protocol. Two sensors in a cluster are said to interfere each other, if a sensor transmission interferes with another sensor. To eliminate the interference among sensors, each sensor should use a channel, which is different from its interfering sensors. In our proposed channel assignment mechanism, an actor calculates the shortest path to all of its cluster members (sensors) in a cluster using Dijkstra's algorithm.

This mechanism reduces the burden on sensors. For calculating the shortest path, every link is assigned a weight (W_{S_i,S_j}) using the sensor remaining energy. The W_{S_i,S_j} represents the link weight between sensor S_i and S_j. It is computed using the residual energy of sensor S_j. After calculating the shortest path, an actor divides the cluster into multiple vertex-disjoint subtrees all rooted at the actor. Then, it allocates a non-interference channel to each subtree. It is similar to the link based channel allocation mechanism. The actor assigns a non-interference channel to its *1-hop relay* nodes using greedy based mechanism. The actor checks whether the distance between the two *relay* nodes is more than the sensor interference range, while assigning the same channel to its any other *relay* node. The channel assignment information is transferred using a common control channel. The *1-hop* sensor assigns the same channel to its *2-hop* child sensors in the subtree as shown in Figure 3.4.

The proposed channel assignment algorithm reduces the channel interference in the inter-subtrees, but still interference exists in the intra-subtree. In the link based channel allocation, a non-interference channel is assigned to every link. So, the data is transmitted on that link using the assigned channel. A channel is assigned for every sensor except the actor. Hence, the receiver should use the same channel on which the sender is transmitting

the data.

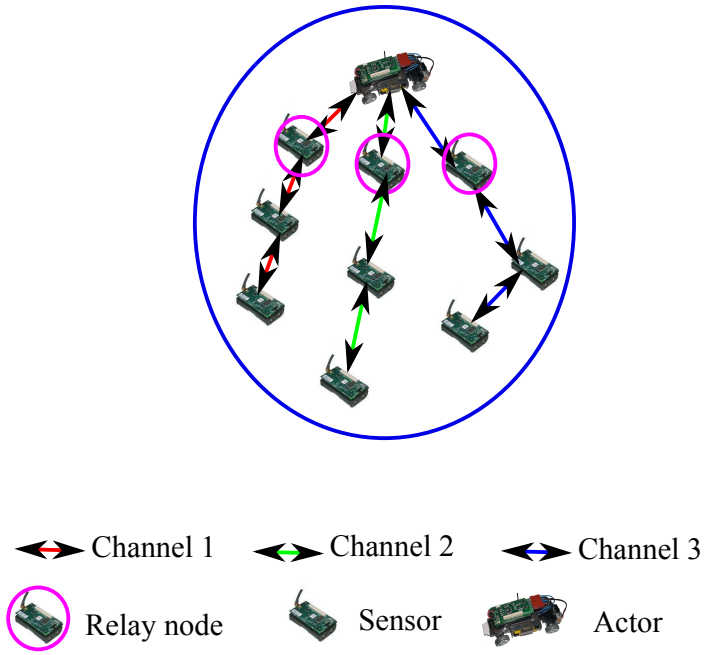


Figure 3.4: Channel assignment in a cluster

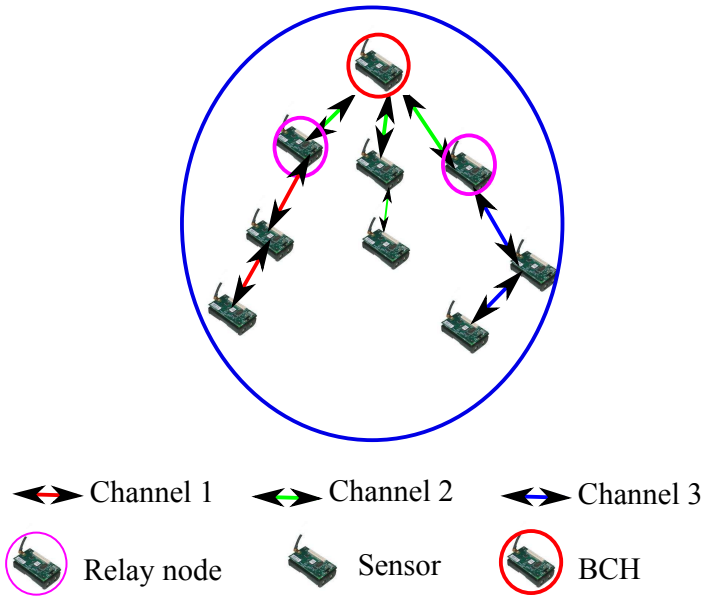


Figure 3.5: Channel assignment in a cluster under backup cluster head scenario

Figure 3.5 depicts the channel assignment under the backup cluster head (BCH) in a cluster. Among the *relay* (sensors which are *1-hop* away from actor) nodes, the actor selects a *relay* node as the BCH based on the backup cluster head score which is described in Section 2.2.2 (Chapter 2). After selecting a BCH from the *relay* nodes, the actor broadcasts

this information to the remaining *relay* nodes using the common control channel. All the *relay* nodes communicate with the BCH using the same channel, but the corresponding child sensors channels are not disturbed. The leaf nodes transfer data to the *relay* nodes using multi-hop communication, then the *relay* nodes forward the received data to BCH. An actor acts as the cluster head when it comes back to its original location.

Contention Based MAC Protocol

Only channel assignment mechanism can not resolve the interference caused by the child sensors in the subtree. It should be further reduced by using contention free or contention based MAC protocols. The contention free MAC protocol requires tight time synchronization which creates a lot of burden on resource conservative sensors and provides less throughput under low traffic conditions. Hence, the proposed IAMMAC protocol uses a contention based MAC protocol. If two sensors want to communicate with a common parent, then the sensor who wins in the contention phase transfers its data to the parent node. The carrier sense multiple access/collision avoidance (CSMA/CA) mechanism is to used in the contention phase. The control messages are transferred using the common control channel to improve network throughput. If a sensor does not have data to transmit, then it will go to sleep state and forward its sleep duration to its *1-hop* neighbors. The sleep period reduces the energy consumption and idle listening time in the network.

Channel Selection Mechanism for Actor-Actor Coordination

A delay aware MAC protocol is required for actor-actor coordination in WSAN. Energy is not an important parameter while designing a MAC protocol for actor-actor coordination because an actor is a resource-rich node. A throughput based multi-channel MAC protocol has been designed for actor-actor coordination. Each actor is embedded with two radios for sensor-actor and actor-actor coordination. So, the data transmission in a sensor-actor phase does not interfere with the actor-actor coordination. The interference is considered while computing the channel throughput. The proposed multi-channel MAC protocol selects a channel which provides maximum throughput among the available channels [65]. This leads to finding a better channel from source to destination and increases the network performance. In this protocol, time is segregated into beacon intervals. Each beacon interval is further divided into ad-hoc traffic indication message (ATIM) window and data transmission phase as shown in Figure 3.6. During ATIM window, the actors negotiate for maximum throughput channel with the destination to transfer the data. The channel negotiation between source and destination is done via a common control channel. In the ATIM window, each actor should listen to the control channel and sends its control

messages.

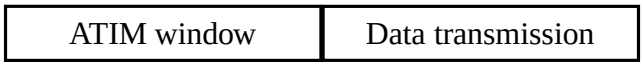


Figure 3.6: Channel architecture for actor-actor coordination

Let us consider a scenario where actor A_i wants to transfer data to the actor A_j. Actor A_i senses the control channel, if it is idle for distributed interframe spacing (DIFS) time, then it generates a random back-off time in the range [0, $cw - 1$]. Where, cw is the size of the contention window. When the back-off timer reaches to zero, the actor A_i sends a ready to send (RTS) packet. If it is successfully received, the actor A_j waits for a short interframe spacing (SIFS) time and sends a clear to send (CTS) packet. The actor A_i sends a probe packet to actor A_j that consists of maximum throughput channel among the available channels. After receiving a probe packet, the actor A_j checks the channel with its neighbors. If it does not provide interference then it sends a confirmation packet else sends an invalid message. This MAC protocol tries to reduce the collisions and selects a channel which provides highest throughput among the available channels. The steps followed are given in **Algorithm** 4.

Algorithm 4: Channel selection in actor-actor coordination

Channel(N_B, SN, M_R**)**
  foreach *Channel* C_i **do**
    $\Psi_{C_i} = \frac{AG_{a_i}\phi_{C_{ck}} Z_{C_{ck}}(1-Q_{C_{ck}})}{\sum_{T=0}^{n} n\phi_T Z_T(1-Q_T)}$
    $max \leftarrow \Psi_{C_0}$
    if $max < \Psi_{C_i}$ **then**
      $max \leftarrow \Psi_{C_i}$ $BC \leftarrow C_i$
    end
  end
  $A_j \rightarrow A_k : RTS(BC)$
  $A_k \rightarrow A_j$: CTS

Let us consider an actor A_i sends data to the actor A_j over channel C_{ck}. The throughput for channel C_{ck} from actor A_i to A_j is calculated as,

$$\Psi_{A_iA_j}^{C_{ck}}(t) = \frac{AG_{A_i}\phi_{C_{ck}} Z_{C_{ck}}(1 - Q_{C_{ck}})}{\sum_{T=0}^{n} \phi_T Z_T(1 - Q_T)} \tag{3.1}$$

where, $\Psi_{A_iA_j}^{C_{ck}}(t)$ denotes the throughput of channel C_{ck} between actors A_i and A_j at time t. AG_{A_i} denotes aggregated throughput of the actor A_i , and it is the sum of the data rates that are delivered to actor A_i. The $\phi_{C_{ck}}$ represents the service probability of channel C_{ck}. $Q_{C_{ck}}$ represents the channel loss probability and it is the ratio of number of packets dropped and number of packets successfully transferred. $Z_{C_{ck}}$ defines the channel (C_{ck}) service rate. The $Z_{C_{ck}}$ is the sum number of packets that are successfully transmitted and number of packets that are dropped. The service probability ($\phi_{C_{ck}}$) of the channel C_{ck} is calculated as,

$$\phi_{C_{ck}} = \frac{\omega_{C_{ck}} MC_{C_{ck}}}{\sum_{C_{ck}=0}^{n} \omega_{C_{ck}} MC_{C_{ck}}} \tag{3.2}$$

The $\omega_{C_{ck}}$ provides the window size at the back-off time t and $MC_{C_{ck}}$ calculates the maximum capacity of channel C_{ck}. According to Shannon's theorem, the channel capacity not only depends on its bandwidth, but also depends on the received signal strength and interference [66]. The maximum capacity ($MC_{C_{ck}}$) that a channel C_{ck} can provide between actor A_i and A_j can be computed as,

$$MC_{C_{ck}} = B\log_2\left[1 + \frac{R_{y,A_j}^{C_{ck}}}{GN + R_{I,A_j}^{C_{ck}}}\right] \tag{3.3}$$

where, GN is the white Gaussian noise power, B is the bandwidth of the channel C_{ck}, and $R_{y,A_j}^{C_{ck}}$ is the received signal power by the sensor A_j. The $R_{y,A_j}^{C_{ck}}$ value depends on the node density and probability of a node in active state. The $R_{I,A_j}^{C_{ck}}$ provides the interference information at sensor A_j in channel C_{ck}. The channel interference is estimated as,

$$R_{I,A_j}^{C_{ck}} = \frac{1}{ns_{A_j} * M_{A_i,A_j}} \sum_{C_{ck} \in C} D_{C_{ck}} + \sum_{C_{ck} \in C} CS_{C_{ck}} \tag{3.4}$$

where, $R_{I,A_j}^{C_{ck}}$ value is close to zero then it indicates the channel C_{ck} has less interference from its neighbors. ns_{A_j} represents neighbor set of actor A_j , which is useful to calculate the interfering actors with A_j during data transmission on channel C_{ck}. M_{A_i,A_j} is the expected transmission time (ETT) between A_i and A_j, $D_{C_{ck}}$ represents the interference-aware resources for channel C_{ck}, and $CS_{C_{ck}}$ defines channel switching cost. In our simulation, the channel switching cost is fixed to 224 μs. The ETT between actor A_i and A_j is calculated as,

$$M_{A_i,A_j} = \frac{1}{(1-p)} * \frac{PS}{B} \tag{3.5}$$

where, p denotes the probability of an unsuccessful transmission. PS and B represent probe packet size and bandwidth of the channel C_{ck}, respectively. The probability p can be

computed as,

$$p = 1 - (1 - p_{fd})(1 - p_{rd}) \tag{3.6}$$

where, p_{fd} and p_{rd} define the probability of packet loss in the forward and reverse directions, respectively. The interference aware resources for channel C_{ck} is estimated as,

$$D_{C_{ck}} = M_{A_i,A_j} * ns_{A_j} \tag{3.7}$$

In the IAMMAC protocol, an actor assigns a set of channels to its cluster members. The sensors transfer data to their corresponding parent nodes using assigned data channels. It is a centralized approach and reduces burden on the sensor nodes. The actor performs reliable and timely actions in the event area based on the sensors' information. If an actor alone can not perform appropriate actions in the event area, then it can seek the help of neighboring actors. An actor selects a maximum throughput channel among the available channels to communicate with its neighboring actors. The channel selection mechanism for actor-actor coordination calculates the channel interference level using ETT parameter. The ETT calculation consumes a lot of energy but gives accurate results in the channel interference level. Hence, it is used in the actor-actor coordination, because actors are resource-rich nodes.

3.3 Simulation Results and Analysis

The performance of IAMMAC protocol is evaluated using NS2 simulator. Each sensor is enabled with single radio and directional antenna whereas an actor is embedded with two radios for sensor-actor and actor-actor coordination. Multiple channels and omnidirectional antenna are enabled on each radio of an actor. In simulation, the size of the data packet is defined as 64 bytes, beacon interval is 100 ms, and the ATIM window size is 20 ms. The number of channels is varied from 3 to 4. 100 - 1000 static sensors are placed uniformly deployed in the $1000 \times 1000\ m^2$ area.

In the proposed IAMMAC, we have assumed that actors are semi-mobile. Initially, the actors are deployed at proper positions to improve their coverage area using *k-hop* independent dominant set algorithm [32]. If an event occurs, the actor moves to the target location and perform required actions. The actor comes back to its original location after performing actions in the target location. The simulation parameters are listed in Table 3.1. A radio model is considered to compute the energy consumption while transmitting and receiving the data which is described in Section 2.1 (Chapter 2). Three simulation scenarios

are considered to analyze the performance of the proposed IAMMAC protocol with its competitive MAC protocols.

Table 3.1: Simulation parameters for IAMMAC

Parameters	Values
Simulation Duration	200 s
Traffic Flow	CBR
CBR packet interval	0.05 - 0.016 s
Routing protocol	DEACP
Sensor's Transmission Range	100 m
Actor's Transmission Range	300 m
Optimal number of actors	3 - 12
K	3
Seed value	0
Channel Switching Cost	224 μs
Number of sensors	100 - 1000
Sensor's Initial Energy	2J
Packet Size	64 B
ATIM window size	20 ms
Beacon interval	100 ms
Data Transfer Rate	20 - 60 pkts/s
Number of channels	3 - 4
E_{elec}	$50nJ/bit$
E_{fs}	$10pJ/bit/m^2$
E_{mp}	$0.0013pJ/bit/m^4$

3.3.1 Simulation Scenario 1

The simulation has been carried out by varying the number of channels as either three or four. The number of sensors being varied from 100 - 1000 in a step of 100. Based on the number of sensors, an optimal number of actors varied from 3 to 12. Each active sensor transfers 20 pkts/s. Along with the proposed IAMMAC protocol, the existing protocols like DMMA [64] and MMIMO [57] are also simulated using same parameters for performance comparison. The network metrics such as average end-to-end delay, packet delivery ratio, average goodput, and average energy consumption are used to analyze the performance of all the three protocols under consideration.

Figure 3.7 depicts the average end-to-end delay for three channels and similar results are shown in Figure 3.8 for four channels. The simulation results indicate that the average end-to-end delay increases with the increase in network density and it is inversely proportional to the number of channels. In the proposed IAMMAC protocol, the contention

between intra-subtree sensors are minimal. However, the inter-subtree contention still exists. A contention based MAC protocol has been used to further reduce the contention. The proposed IAMMAC protocol performs well as compared to the existing DMMA, MMIMO MAC protocols. Further, the proposed IAMMAC delivers the data with 8% and 11% less time as compared to the existing mechanisms for three channels and four channels, respectively .

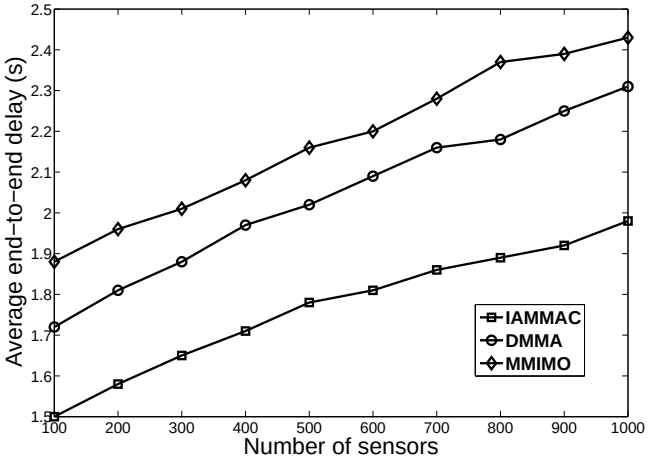


Figure 3.7: Comparative analysis of average end-to-end delay with number of sensors (number of channels = 3)

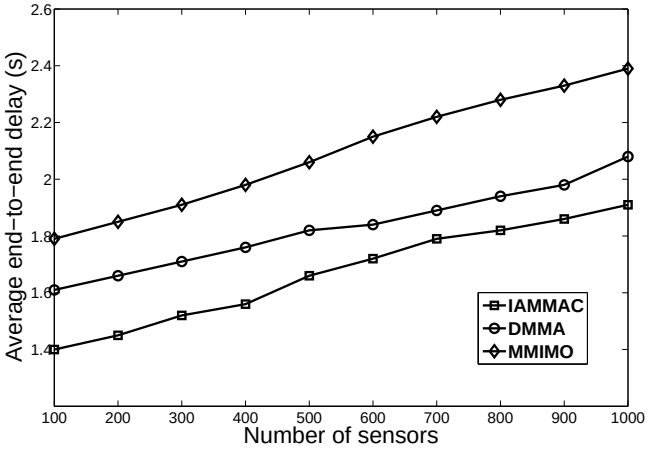


Figure 3.8: Comparative analysis of average end-to-end delay with number of sensors (number of channels = 4)

In WSAN, the packet delivery ratio depends on the link lifetime and congestion in the network. The IAMMAC protocol reduces the network congestion by transferring data

through multiple channels. The control and data packets are transferred using control channel and assigned data channel, respectively. Each actor is enabled with multiple channels to improve the network performance. Figures 3.9 and 3.10 show that the proposed IAMMAC protocol achieves 12% and 11% more packet delivery ratio as compared to the existing DMMA and MMIMO MAC protocols for three and four channels, respectively.

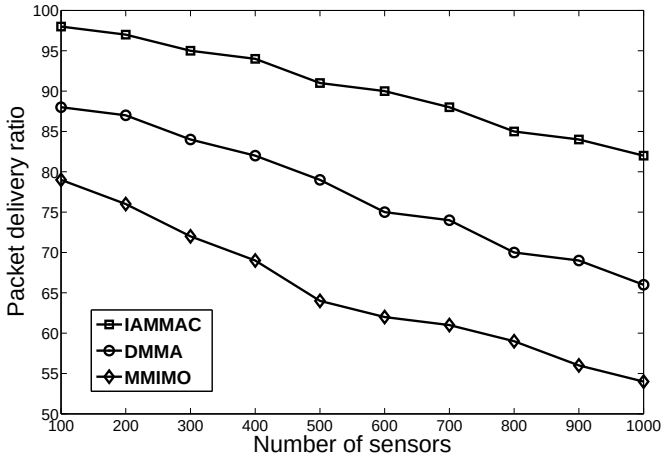


Figure 3.9: Comparative analysis of packet delivery ratio with number of sensors (number of channels = 3)

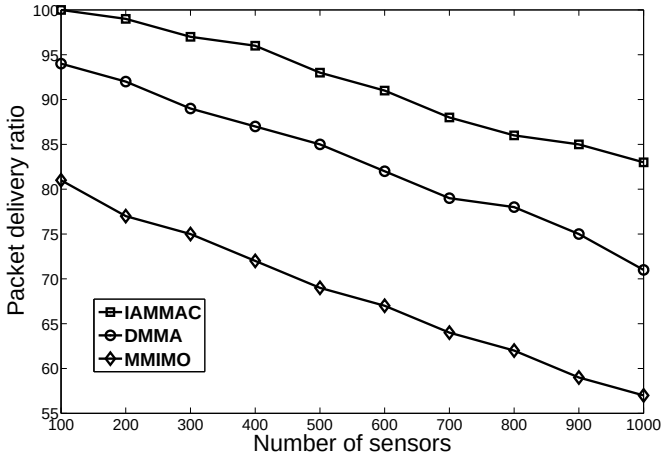


Figure 3.10: Comparative analysis of packet delivery ratio with number of sensors (number of channels = 4)

WSAN consists of a vast number of battery constrained sensors, so it is important to design an energy efficient MAC protocol. In IAMMAC protocol, a sensor goes to sleep state whenever it does not have any data to send. An actor reduces the burden on sensors by performing energy consuming tasks namely, shortest path calculation and

channel allocation for all the sensors. Hence, average energy consumption in the network for IAMMAC protocol is less as compared to the existing DMMA and MMIMO MAC protocols. Figures 3.11 and 3.12 depict that the average energy consumption in the network increases with the increase in network density and inversely proportional to the number of channels for a constant data transfer rate.

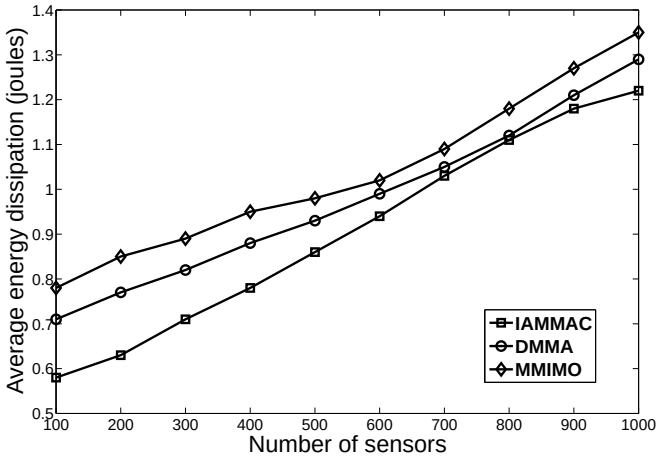


Figure 3.11: Comparative analysis of average energy dissipation with number of sensors (number of channels = 3)

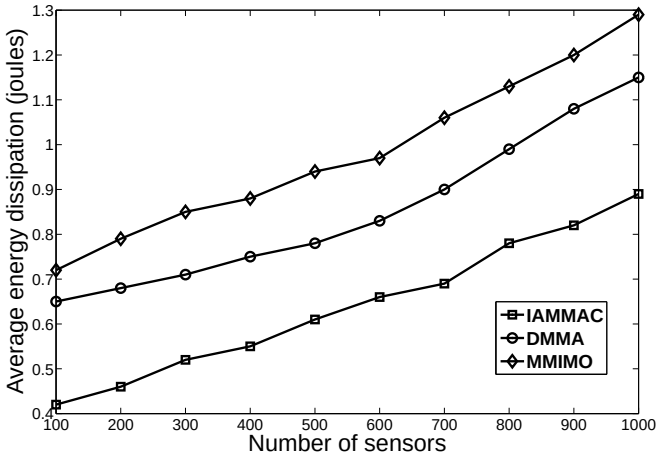


Figure 3.12: Comparative analysis of average energy dissipation with number of sensors (number of channels = 4)

Goodput is an application level throughput and can be defined as the number of useful bits that have been delivered to the destination per unit time. It excludes protocol overhead bits and retransmitted data packets. Figures 3.13 and 3.14 illustrate the performance of the proposed IAMMAC, existing DMMA and MMIMO MAC protocols with respect to average

goodput in the network. The average goodput increases with the increase in number of sensors and number of channels. The goodput depends on the data transfer delay and packet delivery ratio. As, IAMMAC protocol achieves less transmission delay and more packet delivery ratio as compared to DMMA and MMIMO MAC protocols. Hence, IAMMAC protocol produces 13% and 11% more average goodput as compared to the existing MAC protocols for three and four channels, respectively.

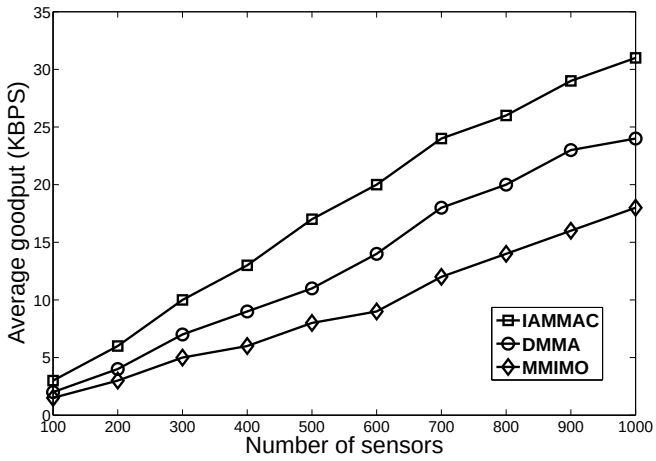


Figure 3.13: Comparative analysis of average goodput with number of sensors (number of channels = 3)

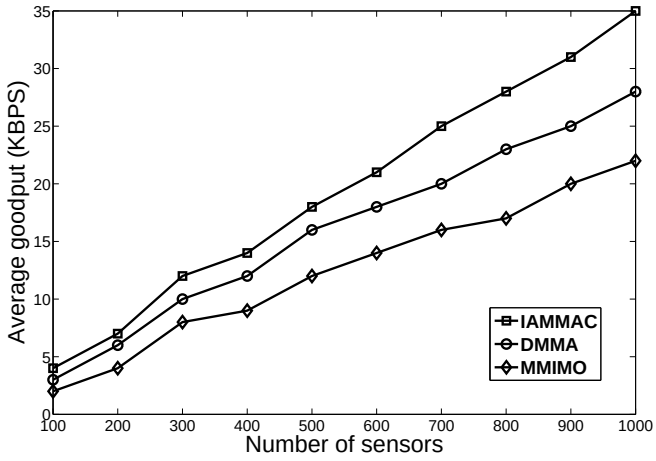


Figure 3.14: Comparative analysis of average goodput with number of sensors (number of channels = 4)

3.3.2 Simulation Scenario 2

In this scenario, the data transfer rate varied from 20 pkts/s to 60 pkts/s in a step of 10 pkts/s. The number of sensors and actors are fixed to 500 and 7, respectively. Three channels are used to transfer the information in the network. Figure 3.15 depicts the average end-to-end delay for data transfer rate from 20 - 60 pkts/s. The average end-to-end delay increases with the increase in data transfer rate for a constant number of sensors and channels. In the proposed IAMMAC protocol, a sensor transfer its data to the actor using its assigned non-interference channel and the actor can receive from multiple sensors as it uses multi channel communication. Hence, our proposed IAMMAC protocol achieves 9% less average end-to-end delay as compared to its competitive protocols.

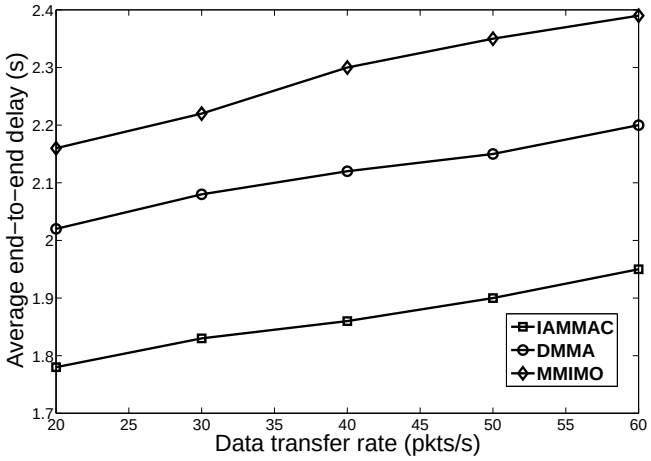


Figure 3.15: Comparative analysis of average end-to-end delay with data transfer rates

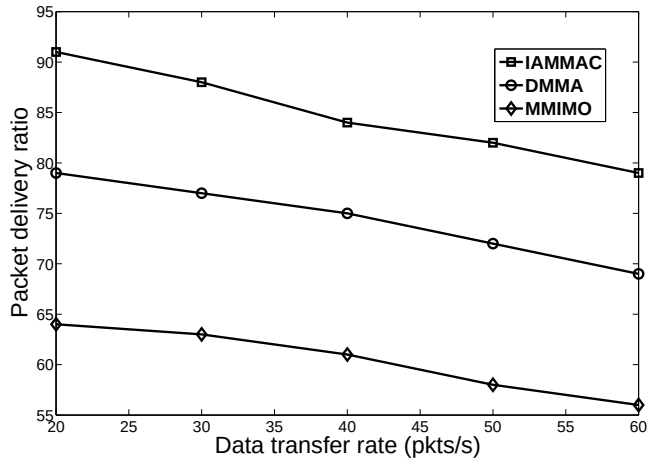


Figure 3.16: Comparative analysis of packet delivery ratio with data transfer rates

The performance of the proposed IAMMAC protocol, existing DMMA and MMIMO MAC protocols with respect to packet delivery ratio for 20 - 60 pkts/s is shown in Figure 3.16. The simulation results indicate that the packet delivery ratio is inversely proportional to the data transfer rate for a constant number of sensors. The network congestion increases with the increase in data transfer rate for a fixed network resources, it leads to degrade the packet delivery ratio. It can be observed that the proposed IAMMAC protocol achieves 10% more packet delivery ratio as compared to the existing MAC protocols.

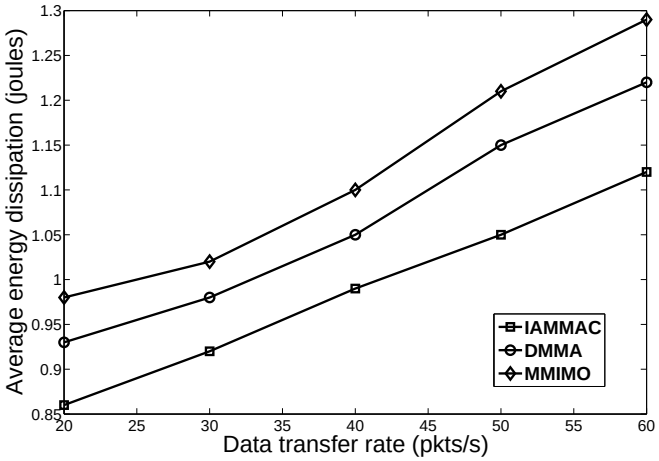


Figure 3.17: Comparative analysis of average energy dissipation with data transfer rates

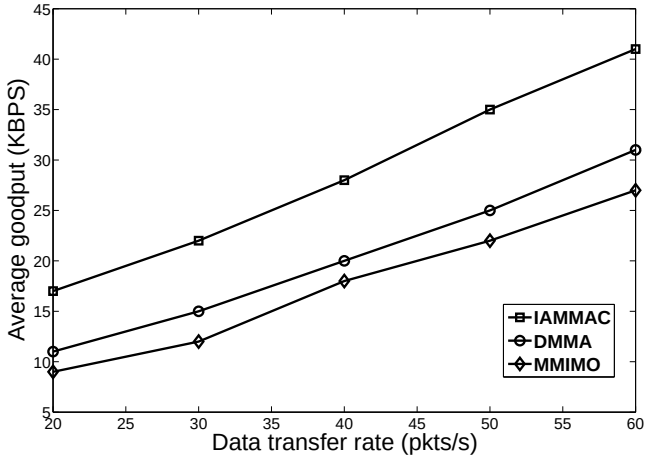


Figure 3.18: Comparative analysis of average goodput with data transfer rates

Figure 3.17 shows the average energy consumption in the network for 500 sensors with data transfer rate varied from 20 pkts/s - 60 pkts/s. In the proposed IAMMAC protocol, the actor acts as cluster head and assigns static channel to its cluster members. The sleep

mechanism is introduced in the sensors to improve the network lifetime. A sensor goes to sleep state, whenever it does not have any data to send. Thus, IAMMAC protocol consumes 6% less average energy as compared to the existing DMMA and MMIMO MAC protocols.

The average goodput of the proposed IAMMAC protocol, existing DMMA and MMIMO MAC protocols for data transfer rate of 20 - 60 pkts/s is shown in Figure 3.18. The results indicate that the average goodput increases with the increase in data transfer rate for a constant number of sensors. It can be observed that the IAMMAC protocol achieves 35% more average goodput as compared to the existing MAC protocols.

3.3.3 Simulation Scenario 3

In this scenario, the number of sensors is varied from 100 - 1000 in a step of 100. Three number of channels are used to transfer the information in the network. The performance of the proposed IAMMAC protocol is analyzed for 20 - 60 pkts/s using metrics such as packet delivery ratio, average end-to-end delay, and average energy dissipation in the network.

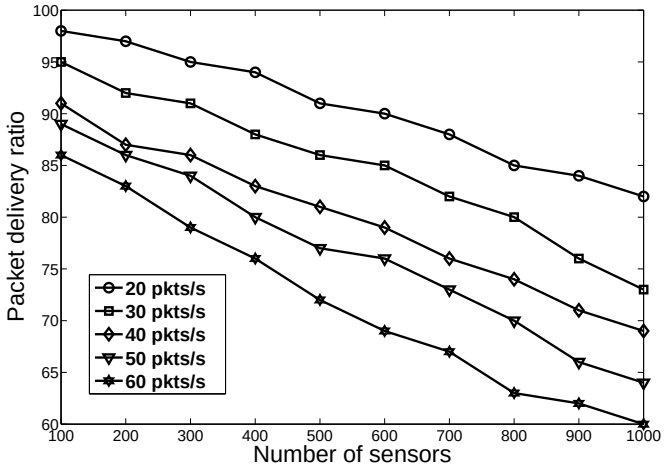


Figure 3.19: IAMMAC protocol packet delivery ratio with number of sensors

The packet delivery ratio of the proposed IAMMAC protocol for three channels with data transfer rate is varied from 20 - 60 pkts/s in a step of 10 pkts/s. Similarly, the number of sensors are also varied from 100 - 1000. Figure 3.19 depicts that the packet delivery ratio decreases with the increase in data transfer rate and number of sensors.

Figures 3.20 and 3.21 illustrate the performance of the proposed IAMMAC protocol with respect average end-to-end delay and average energy dissipation in the network for data transfer rate of 20 - 60 pkts/s. It can be observed that the average end-to-end delay

and average energy consumption of the IAMMAC protocol are directly proportional to the number of sensors and data transfer rate.

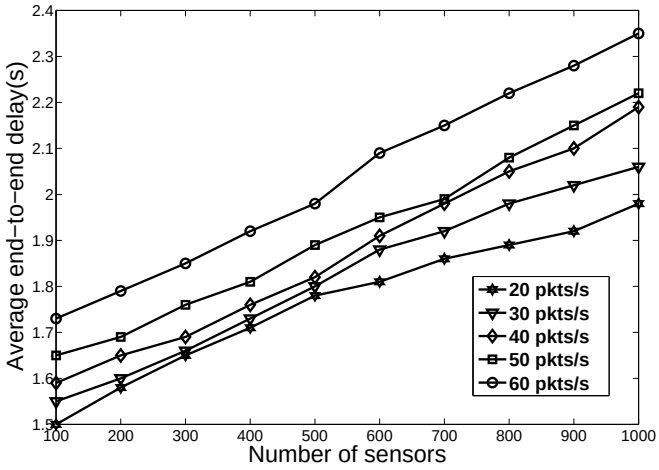


Figure 3.20: IAMMAC protocol average end-to-end delay with number of sensors

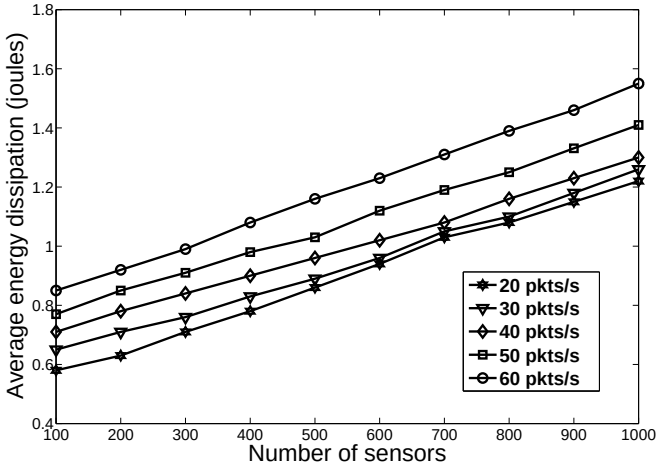


Figure 3.21: IAMMAC protocol average energy dissipation with number of sensors

3.4 Summary

A delay and energy aware coordination protocol (DEACP) has been proposed in Chapter 2 to deliver the maximum number of packets with in the bounded delay. In this chapter, an interference aware multi-channel MAC protocol has been proposed to assign channels for sensor-sensor, sensor-actor, and actor-actor coordination in DEACP. In the proposed IAMMAC protocol, an actor acts as a cluster head for *K-hop* sensors and computes the

shortest path for all the sensors. Then, the actor partitions the cluster into multiple subtrees and assigns a non-interference channel to each subtree. The sensors which are *1-hop* away from an actor are represented as *relay* nodes. The actor selects a *relay* node as a backup cluster head (BCH) based on the residual energy and the node degree. After selecting a BCH from the *relay* nodes, the actor broadcast this information to the remaining *relay* nodes using the common control channel. All the *relay* nodes communicate with the BCH using the same channel, but the corresponding child sensors channels are not disturbed. The leaf nodes transfer data to the *relay* nodes using multi-hop communication, then the *relay* nodes forward the received data to BCH. Further, a throughput aware dynamic multi-channel MAC protocol is also proposed for actor-actor coordination. The performance of the proposed IAMMAC protocol has been analyzed using metrics such as packet delivery ratio, average end-to-end delay, average goodput, and average energy dissipation in the network. The obtained simulation results indicate that the proposed IAMMAC protocol has superior performance as compared to the existing DMMA and MMIMO MAC protocols.

Chapter 4

A Dynamic Multi-channel MAC Protocol for Sensor-Sensor Coordination

Wireless sensor-actor networks (WSAN) is a collection of sensors and actors. Generally, these networks are deployed in an unprotected environment to sense the physical world and perform reliable actions on it. These networks are always susceptible to various kinds of attacks. Our objective is to design an energy efficient MAC protocol which can protect sensors' data from the attackers. An interference aware multi-channel MAC protocol (IAMMAC) has been proposed in Chapter 3. In IAMMAC protocol, an actor acts as a cluster head for *K-hop* sensors and computes the shortest path for all the sensors. The actor partitions a cluster into multiple subtrees and assigns a non-interference channel to each subtree. Thus, a static channel is assigned between two sensors for entire communication to transfer data to the cluster head (actor). Even though its performance is superior, it is susceptible to be attacked because it uses a single static channel between two sensors in the entire communication.

To overcome this problem, various authors have designed dynamic channel selection mechanisms in sensor networks [67]. In dynamic channel selection mechanisms, each sensor selects the best channel dynamically based on the metrics such as channel capacity, throughput, and packet delivery ratio among the available channels. Due to the ample amount of resource constrained sensors, it is also important to design a lightweight MAC protocol for WSAN. To achieve these objectives, in this chapter, a dynamic multi-channel MAC protocol (DM-MAC) has been designed for sensor-sensor coordination. Each sensor dynamically selects a channel which has the maximum packet reception ratio among the available channels with the destination.

The rest of the chapter is organized as follows. Section 4.1 describes various existing lightweight MAC protocols for sensor networks and ad-hoc networks. The proposed

dynamic multi-channel MAC protocol for sensor-sensor coordination is discussed in Section 4.2. Section 4.3 presents simulation results and analysis. Finally, Section 4.4 summarizes the chapter.

4.1 Related Work

In this section, some of the existing lightweight MAC protocols for sensor networks and ad-hoc networks are analyzed to list out their merits and demerits. Sensor-MAC (S-MAC) is a contention based protocol [68]. Each sensor uses a periodic wake-up mechanism to improve its lifetime. The sensor active period is based on its radio characteristics. However, a sensor sleep duration depends on the application requirement. In S-MAC, sensors periodically exchange their schedule with neighbors for synchronization. Due to high latency in the packet delivery, S-MAC does not meet the requirements of WSAN.

To overcome these drawbacks of S-MAC protocol, Lin *et al.* uses a dynamic duty cycle for sensors [69]. Each sensor dynamically adjusts its wake-up period based on its average packet delay. Initially, a common duty cycle is adopted for all the sensor nodes. If a receiver experiences an intolerable packet delay, then it will double its duty cycle by reducing the sleep period. Hence, it achieves less packet delay under heavy traffic conditions. Later, Polastre *et al.* have designed a MAC protocol for sensor networks to reduce the delay in data transmission [70]. It uses a noise floor estimation mechanism for finding an accurate active channel for data transmission. Pham *et al.* have proposed a MAC protocol to handle the sensors' mobility [71]. In a static sensors scenario, it adopts S-MAC protocol to conserve energy; otherwise, it adopts the IEEE 802.11 mechanism. The change in received control message signal strength indicates the node mobility.

Lu *et al.* have proposed an energy aware MAC protocol for sensor networks [72]. It overcomes data forwarding interruption problem that exists in S-MAC. It uses a staggered wake-up scheme to transfer data in the network. A sensor wake-up duration depends on its level in the data aggregation tree. Langendoen *et al.* have proposed a T-MAC protocol for sensor networks [73]. It reduces sensor idle period through the dynamic duty cycle mechanism. In the active period, packets are transmitted in burst of variable size. T-MAC protocol uses a handshake (RTS-CTS-Data-ACK) mechanism to reduce the number of collisions. Initially, a common duty cycle is adopted for all the sensor nodes. The distributed energy protocol has been proposed to reduce energy dissipation in the network [74]. It dynamically assigns a wake-up period based on the sensor residual energy. Chatterjea *et al.* have proposed a lightweight MAC protocol [75]. The number of data

slots depends on the network load. The slot information is stored in a data distribution table (DDT) and it is periodically updated among all the nodes. The maintenance of DDT causes an extra overhead and high energy consumption in the network. Hence, it is not suitable in sensor networks. An energy efficient multi-token based MAC protocol has been proposed to improve the network lifetime [76]. It provides fault tolerant and reliable data transmission.

A schedule based multi-channel MAC protocol has been proposed in sensor networks [67]. The sink disseminates the global time to the rest of nodes in a network using hierarchical structured tree. The control packets with timing information are sent before data transmissions so that the time synchronization accuracy may depend on traffic flow in the network. Munir *et al.* have designed a distributed MAC protocol for WSAN [77]. It considers the actors are static, which is not an appropriate assumption in real-time applications. It uses a single channel for data communication in the network. In our proposed DM-MAC protocol, a multi-channel MAC protocol has been suggested to improve the network performance. Shih *et al.* have proposed on-demand multi-channel MAC protocol for ad-hoc networks [78]. This protocol divides the entire bandwidth into one control channel and *n* data channels. Each node consists of two half-duplex transceivers to operate on the control and data channels separately. Enabling multiple transceivers on sensor consumes a lot of energy and also decreases the network lifetime. In our proposed DM-MAC protocol, a single transceiver is used in sensors to improve the network lifetime.

Tie *et al.* have proposed a cooperative asynchronous multi-channel MAC protocol for ad-hoc networks [79]. This protocol causes control packet overhead and consumes a lot of energy to broadcast the control information to all of its neighbors. Fucai *et al.* have described a multi-channel MAC protocol for ad-hoc networks (MMACCW) [80]. It uses the channel width adaption technique to improve the network performance. In MMACCW, the sender and receiver get more bandwidth channel, if the traffic between them is more as compared to other nodes. Tsuen *et al.* have proposed a MAC protocol for mobile ad-hoc network (MANET) [51]. It uses a single transceiver and divides the beacon interval into channel negotiation and data transmission phases. However, the fixed length of channel negotiation interval limits the channel utilization.

Ian *et al.* have analyzed the impact of channel selection mechanisms such as random, lowest channel first, and soft channel reservation on bidirectional multi-channel MAC protocol [81]. The random channel selection technique randomly selects a channel from the available channels. The lowest channel first technique selects a lower numbered channel

from the available channels. The soft channel reservation technique selects a channel which has previously transmitted the data successfully. If that channel is not available, then the sender may select a channel randomly or using lowest channel first technique. The soft channel reservation mechanism reduces the multi-channel hidden terminal problem as compared to the random and lowest channel first techniques. In our proposed DM-MAC protocol, each sensor selects a channel based on the channel packet reception ratio (PRR).

Ozlem *et al.* have described a multi-channel MAC protocol for sensor networks (MC-LMAC) [82]. It is an extension to the single channel based MAC (LMAC) protocol for sensor networks. MC-LMAC selects the interference and contention-free channels to transmit data in parallel on different channels. MC-LMAC is a scheduled based channel access mechanism and requires tight synchronization among the nodes. Thus, it consumes a lot of energy from sensors and also increases the control packets overhead in the network. A dynamic multi-channel energy efficient MAC protocol has been designed for sensor networks [83]. It uses an adaptive receiver initiated multi-channel rendezvous and predictive wake-up scheduling mechanism. It substantially enhances the channel utilization and transmission capability by dynamically selecting a minimum interference channel. It selects a sensor as a cluster head to reduce the network lifetime. All the existing lightweight MAC protocols for sensor networks may not perform well in WSAN due to its unique characteristics.

An interference aware multi-channel MAC protocol (IAMMAC) is discussed in the Chapter 3. In IAMMAC protocol, an actor acts as a cluster head for *K-hop* sensors and computes the shortest path for all the sensors. The actor partitions a cluster into multiple subtrees and assigns a non-interference channel to each subtree. Thus, a static channel is assigned between two sensors for entire communication in a cluster. In many applications of WSAN, sensors and actors are deployed in an unprotected environment to sense the physical world and perform reliable actions on it. These networks are always susceptible to various kinds of passive and active attacks by malicious nodes. If a static channel is used between two sensors in the entire communication, then the attacker can easily attack the network. In this chapter, a dynamic multi-channel MAC protocol (DM-MAC) has been proposed for sensor-sensor coordination to overcome these drawbacks. Each sensor dynamically selects a channel which has the maximum packet reception ratio among the available channels with the destination.

4.2 Proposed Dynamic Multi-channel MAC Protocol

In wireless networks, interference plays an important role in degrading the network performance. Due to the broadcast medium, data transmission from a node interfere with its neighboring nodes resulting in lower throughput and higher data latency. In WSAN, interference is very high, due to the dense node deployment and limited bandwidth. In DM-MAC protocol, every sensor selects the channel for data communication that provides highest packet reception ratio (PRR) with respect to the destination among its available channels to improve the network performance.

4.2.1 Channel Selection Mechanism for Sensor-Sensor Coordination

Each sensor can transmit data using single channel (among its available multiple channels) because it is embedded with only one half-duplex transceiver. So, each sensor selects a best channel among the available channels based on the channel packet reception ratio. In communication theory, the bit error rate (BER) is defined as the probability that a receiver fails to receive an incoming bit, because of signal to interference plus noise ratio (SINR). Unfortunately, the BER-SINR cannot be measured directly on radio transceivers [84]. Hence, recent studies have used a PRR with SINR model [85, 86]. Packet reception ratio (PRR) is defined as the probability that a receiver successfully receives all bits in an incoming packet on a particular channel and it is computed as,

$$PRR_{S_j}(sp) = pr_{S_j}(sp)^{x(sp)} \tag{4.1}$$

where, $pr_{S_j}(sp)$ is the probability that sensor S_j receives an incoming bit of packet (sp) of size $x(sp)$ on channel C_{ck}. The $pr_{S_j}(sp)$ depends on the signal energy E, and the two-sided power spectral noise density $ND/2$. The $pr_{S_j}(sp)$ is computed as,

$$pr_{S_j}(sp) = 1 - Z\left(\sqrt{\frac{2E}{ND}}\right) \tag{4.2}$$

$$Z(x) = \frac{1}{\sqrt{2\pi}} \int_{y}^{\infty} e^{\frac{-t^2}{2}} dt = \frac{1}{2}(1 - gef(y/\sqrt{2})) \tag{4.3}$$

where, $gef()$ function is the Gaussian error function. The SINR at the receiver of packet sp is computed as,

$$SN = \frac{E}{GN}\frac{M_R}{N_B} \tag{4.4}$$

where, M_R is the modulation rate and N_B is the noise bandwidth. Eq. 4.5 is derived by

substituting Eq. 4.2 to Eq. 4.4 in Eq. 4.1.

$$PRR_{S_j}(sp) = \left(\frac{1}{2} + \frac{1}{2}\left(gef\left(\sqrt{\frac{N_B * SN}{M_R}}\right)\right)\right)^{x(p)} \tag{4.5}$$

The link throughput also depends on the channel utilization. Hence, each sensor selects the best channel to improve the network performance. In DM-MAC protocol, each sensor selects a maximum PRR channel to transfer its data to the intermediate sensor. DM-MAC protocol not only finds the better channel from the source sensor to destination actor and also increases the network performance.

In the proposed DM-MAC protocol, time is divided into beacon intervals. Each beacon interval is further divided into ad-hoc traffic indication message (ATIM) window and data transmission phase. During ATIM window, the sensor that has packets to transmit negotiates maximum PRR channel with the destination. The channel negotiation between source and destination is performed in the common control channel. During ATIM window, each sensor should listen the control channel to send its control messages. When a sensor S_i wants to transfer data to S_j, it senses the control channel. If the channel is idle for a distributed interframe spacing (DIFS) time, then the sensor S_i generates a random backoff time from the range [0, $cw - 1$], where cw is the size of the contention window. When the backoff timer reaches to zero, the sensor S_i sends a ready to send (RTS) packet. In the RTS phase, the sensor S_i sends information about the channel that consists of maximum PRR channel with respect to the destination S_j among the available channels. After receiving the channel information, sensor S_j sends a clear to send (CTS) packet to sensor S_i and switches to the selected channel to receive data from sensor S_i. This contention based mechanism reduces the number of collisions and selects a maximum PRR channel among the available set of channels. The steps followed are given in **Algorithm** 5.

The objective of sensor-actor coordination is to deliver the sensor data to the nearest actor with minimum energy and delay. In our proposed architecture, an actor acts as a cluster head for K-hop sensors. Thus in a cluster, sensors send their data in a multi-hop fashion to the cluster head (actor). The sensors which are *1-hop* away from an actor are denoted as *relay* nodes. If an actor leaves the cluster to help its neighboring actor, then a *relay* node acts as a backup cluster head based on its residual energy and node degree. The sensors transfer the data to the *relay* node using dynamic channel selection mechanism. The *relay* node selects a highest packet reception ratio channel to transfer data to the cluster head (actor).

Algorithm 5: Channel selection in sensor-sensor coordination

Channel(N_B, SN, M_R**)**
  foreach *Channel* C_i **do**
    $PRR_{C_i} = \left(\frac{1}{2} + \frac{1}{2}\left(gef\left(\sqrt{\frac{N_B * SN}{M_R}}\right)\right)\right)^{x(p)}$
    $max \leftarrow PRR_{C_0}$
    if $max < PRR_{C_i}$ **then**
      $max \leftarrow PRR_{C_i}$
      $BC \leftarrow C_i$
    end
  end
  $S_j \rightarrow S_k : RTS(BC)$
  $S_k \rightarrow S_j$: CTS

An actor-actor coordination manages to perform reliable actions in an event area. Single actor can not perform actions independently in the event area, due to its energy and transmission range constraints. Hence, actors coordinate among themselves to perform actions by optimally allocating tasks to the actors. In Section 3.2.2 (Chapter 3), a dynamic channel selection mechanism for actor-actor coordination has been proposed to deliver the data from one actor to another actor with minimum delay. Each actor dynamically selects a channel, which provides highest throughput among the available channels. The dynamic channel assignment mechanism for actor-actor coordination is used in this chapter to analyze the DM-MAC protocol with our previously proposed IAMMAC protocol, and the existing DMMA and MMIMO MAC protocols.

4.3 Simulation Results and Analysis

To evaluate the performance of the proposed DM-MAC protocol, simulation has been performed in NS2 simulator. Each sensor is enabled with single radio and directional antenna whereas an actor is embedded with two radios for sensor-actor and actor-actor coordination. Multiple channels and omnidirectional antenna are enabled on each radio for an actor. In simulation, the length of the data packet is defined as 64 bytes, beacon interval is 100 ms, and the ATIM window size is 20 ms. The number of channels is varied from 3 to 4. 100 - 1000 static sensors are placed uniformly in the 1000×1000 m^2 area. In the proposed IAMMAC protocol, we have assumed that actors are semi-mobile. Initially, the actors are deployed at proper positions to improve their coverage area using *k-hop* independent dominant set algorithm [32]. If an event occurs, the actor moves to the target location and performs required actions. The actor comes back to its original location after

performing actions in the target location. The simulation parameters are listed in Table 3.1. A radio model has been considered to compute the energy consumption while transmitting and receiving the data as described in Section 2.1 (Chapter 2). Two simulation scenarios are used to compare the performance analysis of the proposed DM-MAC protocol with its competitive MAC protocols such as IAMMAC, DMMA, and MMIMO. Standard metrics like average end-to-end delay, packet delivery ratio, average energy dissipation, and average goodput are used to analyze the protocols under consideration.

Table 4.1: Simulation parameters for DM-MAC

Parameters	Values
Network Area	$1000 \times 1000\ m^2$
Simulation Duration	200 s
Traffic Flow	CBR
CBR packet interval	0.05 s
Routing protocol	DEACP
Seed value	0
Number of Sensors	100 - 1000
Number of Actors	3 - 12
Number of Channels	3 - 4
Channel Switching Time	$224\ \mu s$
Sensor's Transmission Range	100 m
Actor's Transmission Range	300 m
ATIM Window Size	20 ms
Packet Size	64 B

4.3.1 Simulation Scenario 1

The simulation has been carried out by varying the number of channels as either three or four. The number of sensors is varied from 100 - 1000 in a step of 100. Based on the number of sensors, an optimal number of actors is varied from 3 to 12. Each active sensor transfers 20 pkts/s. Along with the proposed DM-MAC protocol, its competitive protocols like IAMMAC, DMMA [64], and MMIMO [57] are also simulated using the same parameters for performance comparison.

Figure 4.1 depicts the packet delivery ratio for three channels and similar results are shown in Figure 4.2 for four channels. It can be observed that packet delivery ratio is inversely proportional to the number of sensors. In WSAN, the packet delivery ratio depends on the link lifetime and congestion in the network. The DM-MAC protocol uses multiple channels to reduce the network congestion. The control and data packets are

transferred using control and assigned data channel, respectively. In DM-MAC protocol, each sensor selects a channel which has the highest packet reception ratio among the available channels. Hence, DM-MAC protocol achieves 2% and 3% more packet delivery ratio as compared to its competitive protocols for three and four channels, respectively.

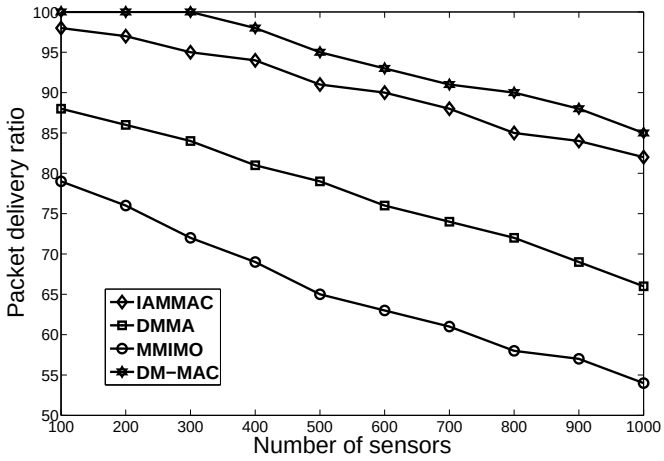


Figure 4.1: Packet delivery ratio vs number of sensors (number of channels =3)

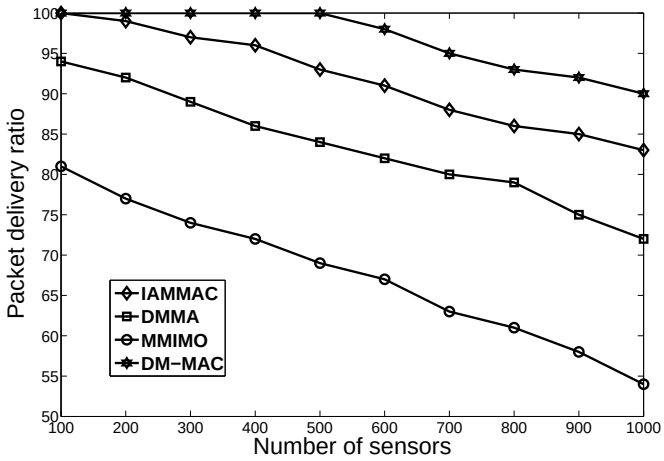


Figure 4.2: Packet delivery ratio vs number of sensors (number of channels = 4)

The average energy dissipation in the network for the proposed DM-MAC protocol, existing IAMMAC, DMMA and MMIMO MAC protocols for three channels is shown in Figure 4.3. Similar results are illustrated in Figure 4.4 for four channels. WSAN consists of a large number of sensors, so it is important to design an energy efficient MAC protocol. In IAMMAC protocol, an actor reduces the burden of sensors by performing energy consumed tasks such as shortest path calculation and channel allocation for all the sensors. However, in DM-MAC protocol, each sensor selects the channel dynamically which provides highest

packet reception ratio among the available channels. Hence, the proposed DM-MAC protocol consumes more energy as compared to IAMMAC protocol. It can be observed that IAMMAC protocol consumes 3% and 5% less average energy compared to the proposed DM-MAC protocol and existing DMMA and MMIMO MAC protocols for three and four channels, respectively.

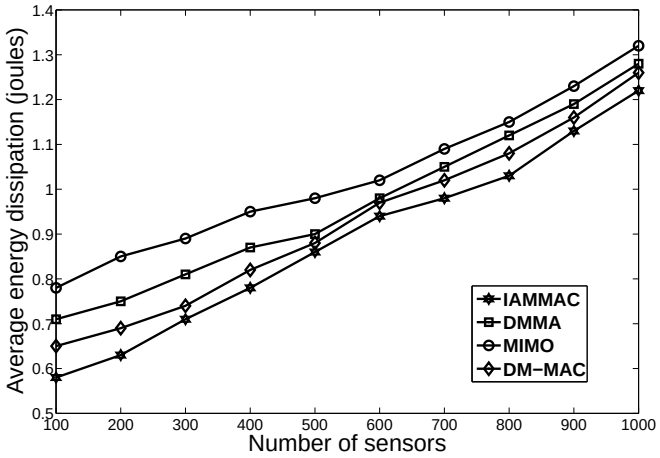


Figure 4.3: Average energy dissipation vs number of sensors (number of channels = 3)

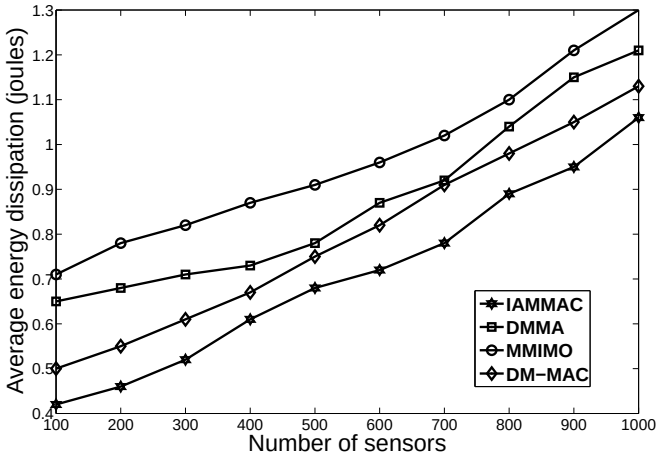


Figure 4.4: Average energy dissipation vs number of sensors (number of channels = 4)

In the proposed DM-MAC protocol, each sensor dynamically selects a channel which has the highest packet reception ratio among the available channels in a multi-hop fashion to transfer the data to its cluster head (actor). The actor dynamically selects a channel which has the highest throughput among the available channels to communicate with its neighboring actors. In the proposed DM-MAC protocol, dynamic channel assignment causes extra communication overhead and delay in selecting the channel. However, in

IAMMAC protocol the contention between intra-subtree sensors are minimal as static channels are assigned to sensors by the cluster head (actor) for communication. Hence, IAMMAC protocol delivers data with 1% and 1.5% less time as compared to the DM-MAC, DMMA, and MMIMO MAC protocols for three and four channels, respectively as shown in Figures 4.5 and 4.6.

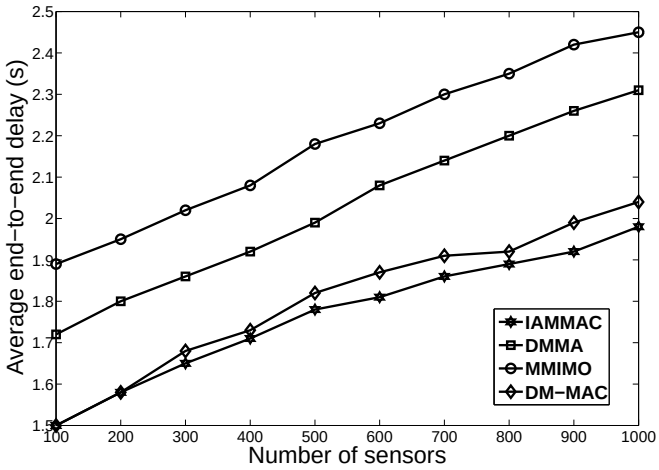


Figure 4.5: Average end-to-end delay vs number of sensors (number of channels = 3)

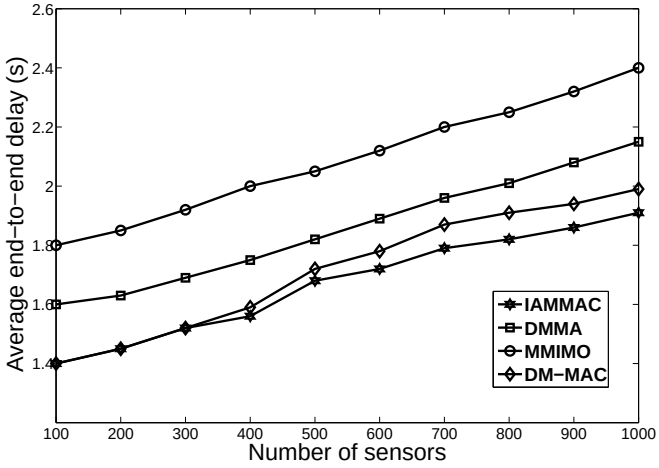


Figure 4.6: Average end-to-end delay vs number of sensors (number of channels = 4)

Figures 4.7 and 4.8 illustrate the performance of proposed DM-MAC, and existing IAMMAC, DMMA and MMIMO MAC protocols with respect to average goodput in the network for three and four channels, respectively. It can be observed that average goodput increases with the increase in number of sensors and number of channels. The goodput depends on the data transfer delay and packet delivery ratio. The proposed DM-MAC

protocol achieves more packet delivery ratio as compared to its competitive MAC protocols. Hence, the proposed DM-MAC protocol achieves 1% and 2% more average goodput as compared to the existing IAMMAC, DMMA, and MMIMO MAC protocols for three and four channels, respectively.

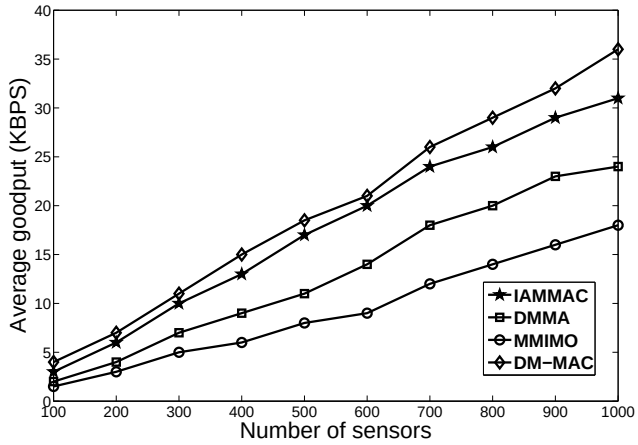


Figure 4.7: Average goodput vs number of sensors (number of channels =3)

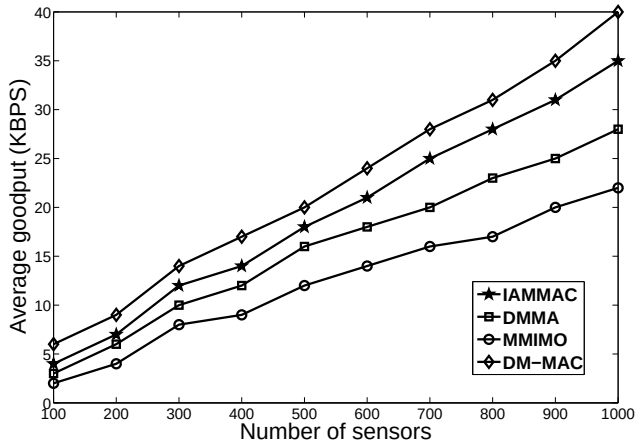


Figure 4.8: Average goodput vs number of sensors (number of channels =4)

4.3.2 Simulation Scenario 2

In this scenario, the data transfer rate is varied from 20 pkts/s to 60 pkts/s in a step of 10 pkts/s. The number of sensors and actors are fixed to 500 and 7, respectively. Three channels are used to transfer the information in the network. The standard network metrics such as packet delivery ratio, average energy dissipation, average end-to-end delay, and

average goodput are used to analyze the performance of the proposed DM-MAC protocol and existing IAMMAC, DMMA, and MMIMO MAC protocols.

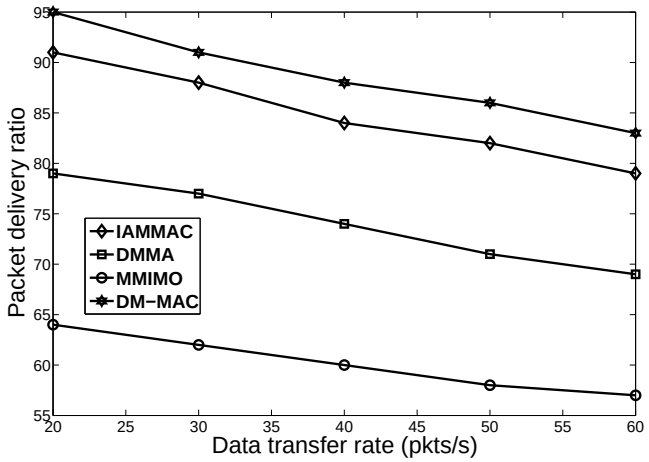


Figure 4.9: Packet delivery ratio vs data transfer rate

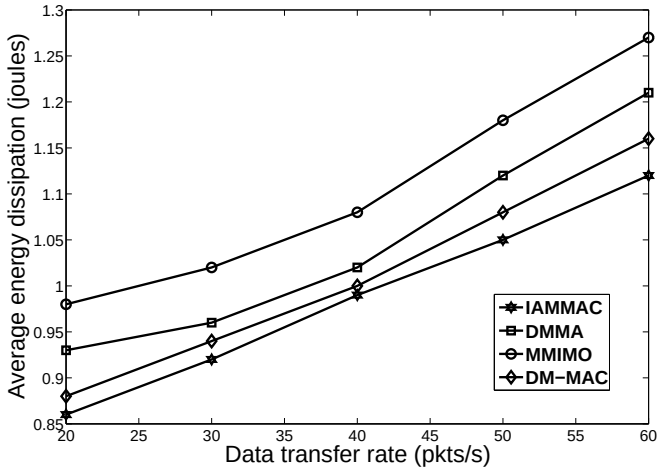


Figure 4.10: Average energy dissipation vs data transfer rate

Figure 4.9 depicts the packet delivery ratio for data transfer rate from 20 - 60 pkts/s of all the four protocols under consideration. It indicates that the packet delivery ratio is inversely proportional to the data transfer rate for a constant number of sensors. It can be observed that the proposed DM-MAC protocol achieves 5% more packet delivery ratio as compared to the existing IAMMAC, DMMA and MMIMO MAC protocols.

Figure 4.10 shows the average energy consumption in the network for 500 sensors with data transfer rate is varied from 20 - 60 pkts/s. In IAMMAC protocol, the actor acts as a

cluster head and assigns static channel to its cluster members. However, in the proposed DM-MAC protocol, each sensor dynamically selects a channel which has the highest packet reception ratio among the available channels. It creates burden on the sensors and degrades the network lifetime. Thus, IAMMAC protocol consumes 1% less average energy as compared to the proposed DM-MAC protocol and existing DMMA and MMIMO MAC protocols.

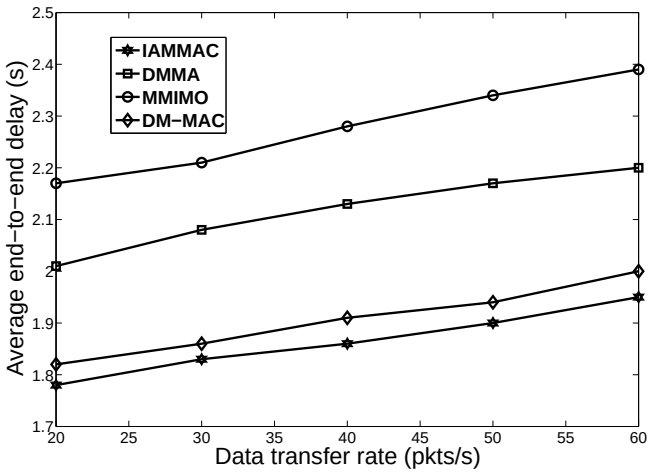


Figure 4.11: Average end-to-end delay vs data transfer rate

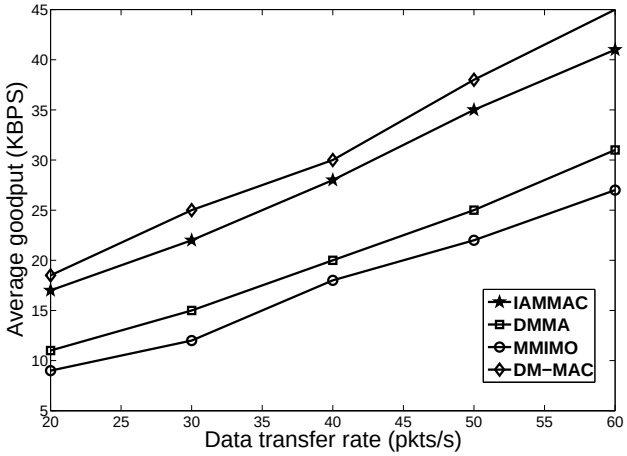


Figure 4.12: Average goodput vs data transfer rate

The performance of the proposed DM-MAC protocol, existing IAMMAC, DMMA and MMIMO MAC protocols with respect to average end-to-end delay for 20 - 60 pkts/s have been compared as shown in Figure 4.11. It indicates that the average end-to-end delay increases with the increase in data transfer rate for 500 sensors and 3 channels. It can be

observed that IAMMAC protocol delivers data with 0.5% less delay as compared to the DM-MAC, DMMA and MMIMO MAC protocols.

The average goodput of the proposed IAMMAC protocol, existing DMMA and MMIMO MAC protocols for data transfer rate of 20 - 60 pkts/s is shown in Figure 4.12. It indicates that the average goodput increases with the increase in data transfer rate for a constant number of sensors. The proposed DM-MAC protocol achieves 2% more average goodput as compared to the existing IAMMAC, DMMA, and MMIMO MAC protocols.

4.4 Summary

In this chapter, a dynamic multi-channel MAC (DM-MAC) protocol has been proposed for sensor-sensor coordination. The proposed protocol is simulated in NS2 along with other competent protocols. The comparative analysis shows that the proposed DM-MAC protocol is energy efficient with actors as the cluster heads. Each sensor selects a maximum packet reception ratio channel to communicate with the neighboring sensor among the available channels. It results in achieving higher goodput and packet delivery ratio as compared to the DMMA, IAMMAC, and MMIMO MAC protocols. In DM-MAC protocol, increase in the average goodput is directly proportional to the number of channels, since it considers the channel interference during data transfer. In IAMMAC protocol, an actor reduces the burden of sensors by performing energy consumed tasks such as shortest path calculation and channel allocation for all the sensors. However, in DM-MAC protocol, each sensor selects the channel based on the channel packet reception ratio. Hence, DM-MAC protocol consumes much energy as compared to the IAMMAC protocol.

Further, IAMMAC protocol has less average end-to-end delay as compared to DM-MAC protocol. In IAMMAC protocol, channel assignment for sensor-sensor coordination is static. Static channel assignment causes lower overhead and delay as compared to dynamic channel assignment mechanisms. However, an attacker can easily induce attacks on static channel assignment mechanisms as sensor always uses the same channel for transferring its data. In many applications of WSAN, sensors and actors are deployed in an unprotected environment to sense the physical world, and perform reliable actions on it. Security is also an important parameter in WSAN, as sensors and actors are always susceptible to various kinds of attacks. The suggested DM-MAC protocol is an alternative solution to IAMMAC protocol to achieve higher security.

Chapter 5

A Secure Coordination Mechanism for Data Forwarding Attacks

Wireless sensor-actor network (WSAN) plays a crucial role in civilian and military applications such as disaster monitoring, battlefield monitoring, medical monitoring, and home intelligence. Security mechanisms are required to achieve data and node protection in these applications. However, designing security protocols is an arduous task in sensor-actor networks because of the following challenges:

1. In the unlicensed frequency band, anyone can monitor the channel. The attackers can attack the network by eavesdropping or by modifying the data transferring through the channel.

2. WSAN is designed to operate in remote and hostile environments. Hence, sensors and actors are prone to failures and vulnerable to various attacks.

3. The sensors are resource constrained nodes. The robust security protocols which consume more energy can not be applied in the sensor nodes. The attackers can easily break the weak security mechanisms. Thus, energy efficient secured mechanism is required in these networks.

WSAN requires an energy efficient and lightweight security mechanisms to protect the network from the attackers. The following requirements should consider while designing any security mechanism in WSAN.

(a) Confidentiality: It is an assurance of authorized access to data. The data should not be revealed to the eavesdropper.

(b) Integrity: It ensures that the data has not been modified during transmission.

(c) Availability: The network should always provide services to authorized parties.

(d) Non-repudiation: In a successful data transmission from a source to the destination, both of them should not deny their participation in the activity.

(e) Authentication: A node verifies the identity of the peer node with which it participates in the data communication.

(f) Freshness: Data and key should always be fresh. Data freshness ensures that the adversary did not replay the old messages. On the other hand, key freshness provides security to the communication.

The attacks in WSAN can be broadly classified into passive and active attacks. In a passive attack, the aim of a malicious node is to observe data flow in the network, but it does not modify or tamper the data. However, these kind of attacks may harm the source and destination. In an active attack, the attacker tries to modify the data or sometimes it does not forward the data packets to the destination. Usually, the active attacks are easier to detect than preventing them because the malicious nodes can launch these attacks in various ways. Hence, the active attacks are more dangerous than passive attacks. In WSAN, the data forwarding attacks such as the sink hole, black hole, and selective forwarding attack (gray hole) attacks are few active attacks [87]. These attacks can also be denoted as the denial of service attacks. In the black hole attack, the attacker drops all the packets in bulk instead of forwarding them to the destination. If a sensor acts as a black hole node in the network, then the actor is unable to identify the event information sensed by the sensor and it may lead to various problems. In a sink hole attack, the attacker advertises as a powerful node and attracts all the traffic. Then, it either drops all the packets or selectively drops few packets. In selective forwarding (gray hole) attack, the attacker selectively drops the packets either from a particular source or some specific type of data [88].

Data encryption and authentication are the main defense mechanisms against various attacks in wireless networks. Many authentication and encryption techniques have been proposed in wireless sensor networks (WSN) [89]. Due to the unique characteristics in WSAN, the existing authentication and encryption based protocols of WSN can not be applied directly and needs substantial modification to devise schemes with less computational and communication overhead. The state-of-the-art research in sensor networks reveals that data encryption techniques consume a lot of energy and degrade the sensors' lifetime as compared to the node authentication techniques. Hence, in sensor networks, authentication techniques are preferred to data encryption techniques.

In Chapter 2, a delay and energy aware coordination protocol (DEACP) has been suggested to improve the network performance. In this chapter, a secure coordination mechanism has been designed to handle data forwarding attacks in DEACP. Each sensor computes the message authentication code for data using the secure hash algorithm-3 (SHA-3) and appends it to the data. The sensor selects a *1-hop* sensor which has the highest trust value among its neighbors to deliver the data to the cluster head (actor).

The rest of the chapter is organized as follows. Section 5.1 describes various mitigation techniques available in the literature for black hole, gray hole, and sink hole attacks. The proposed secure coordination mechanism for DEACP is discussed in Section 5.2. Section 5.3 presents simulation results and analysis. Finally, Section 5.4 summarizes the chapter.

5.1 Related Work

The typical data forwarding attacks in sensor networks and ad-hoc networks are categorized into black hole, sink hole, and gray hole attack. Various researchers have suggested protocols to mitigate these attacks. These are discussed below in sequel.

5.1.1 Mitigation Techniques for Black Hole Attacks

A black hole attack is a kind of denial of service attack accomplished by dropping packets. The attacker drops all the packets in bulk instead of forwarding them to the destination. If a sensor acts as a black hole node in the network, then an actor is unable to identify the event information sensed by the sensor and it may lead to various problems. Karakehayov has proposed a routing algorithm to identify collaborative black hole attack in sensor networks [90]. It uses two broadcast messages: material for intersection of suspicious sets (MISS) and suspicious area mark a black hole attack (SAMBA) to detect the black hole attack. Identification of malicious node working in the ID space can be done with the help of MISS message. Location of the detected black hole attacks has provided by SAMBA message that is related to the physical space. It consumes a lot of energy from the sensors and degrades the network lifetime. A multi-path routing technique has been proposed to handle black hole attacks in sensor networks [91]. Each sensor uses a randomized route to the base station instead of deterministic multi-path routes. Managing the multiple paths increases the network overhead and degrades the sensors' lifetime. A symmetric key cryptography based technique has been proposed to mitigate the black hole attacks [92]. The public key cryptography is not feasible in sensor networks because of their complexity

and consumes more energy as compared to the symmetric key cryptography techniques.

Misra *et al.* have proposed a black hole attack mitigation technique with the help of multiple base stations [93]. Each sensor transmits its data to all the deployed base stations so that data may reach to at least one base station. It causes extra computation and communication overhead on resource conservative sensor nodes. Sheela *et al.* have proposed a black hole mitigation technique using mobile agent [94]. In the normal conditions, the sensor forwards data to its nearest base station. In a black hole attack scenario, data is transferred to the multiple base stations. To identify the black hole node, the mobile agent moves across the network and checks every sensor. If the mobile agent observes the malicious activity, then it tries to remove the sensor from the routing activity.

Samir *et al.* have proposed an intrusion detection based solution (IDS) to handle the black hole attacks [95]. In each cluster, two cluster heads are selected. Each sensor uses primary and secondary cluster head to transfer data and control packets, respectively. The control packet contains node identifier and number of data packets transferred to the base station. The control packet information is useful to identify the black hole node in the network. However, it assumes that each sensor is in *1-hop* distance to the cluster head. It is not an energy efficient technique as sensors have to transfer their information for long distance.

Marti *et al.* have used the watchdog and path rater techniques to detect malicious nodes [96]. Watchdog technique observes the next node in a path to identify malicious activities. Path rater keeps ratings for the nodes and the rating varies from 0 to 0.8, where 0.5 signifies node as neutral. However, the watchdog technique needs to maintain the state information of the monitored nodes and the transmitted packets, which increases the memory overhead. The existing techniques for mitigating black hole attacks in the literature either use secret sharing and path diversity [97, 98, 99] or neighborhood interactions and message overhearing [100]. Implementation of neighborhood message interaction and overhearing techniques assume that the neighbors of black hole attacker node are not compromised, and can observe and report about the black hole nodes to the source. The neighborhood overhearing based techniques are ineffective when few sensors that are close to each other are compromised and collude among themselves.

5.1.2 Mitigation Techniques for Sink Hole and Gray Hole Attacks

In a sink hole attack, the attacker advertises itself as a powerful node and attracts all the traffic. Then, it either drops all the packets or selectively drop few packets. A secure path routing mechanism has been proposed to mitigate the impact of sink hole attacks in sensor networks [101]. Path risk has considered in routing to reduce traffic flow to high vulnerability nodes. Selecting low-risk nodes may lead to expensive energy paths. A sink hole detection mechanism has been proposed using message digest algorithm [102]. If a sensor advertises to provide a shorter route to the base station, then the message travels in both original and advertised routes. The sink identifies the attack when the message digests obtained from both the routes are different. Transferring same data through two paths creates message overhead and reduces the network lifetime. Ngai *et al.* have proposed an intrusion detection system for data forwarding attacks in sensor networks [103]. It identifies a collection of suspected nodes through validating data consistency. It effectively finds the intruder from the suspected node list by analyzing the network flow information. IEEE 802.11 MAC protocol has been used to analyze the performance of the detection mechanism. IEEE 802.11 MAC standard protocol does not perform well in energy conservative sensors.

In a gray hole (selective forwarding) attack, the attacker selectively drops the packets either from a selective source or some specific type of data. Brown *et al.* have proposed a security mechanism to detect the gray hole attack in a heterogeneous sensor network [104]. It consists of few powerful sensors (HS) and an enormous number of battery constrained sensors (LS). The powerful sensor acts as a cluster head. If any packet drop occurs in a cluster, the LS-sensor reports to the corresponding cluster head. Based on the report, the cluster head identifies whether a node is compromised or not. The cluster head uses sequential probability ratio test to identify the malicious sensor in a cluster. Yu *et al.* have proposed a multi-hop acknowledgment scheme [105]. All the intermediate nodes in the communication path act as in-charge for detecting malicious nodes. If any in-charge node detects the malicious information, it will forward an alert packet to the downstream/upstream nodes in a multi-hop fashion. It degrades the network lifetime as all the intermediate nodes participate in the detection process.

To overcome this drawback, Xiao *et al.* have proposed a lightweight security scheme for identifying gray hole attacks [106]. It randomly selects a set of intermediate nodes along the path as checkpoints, which are responsible for sending an acknowledgment to the each received packet. If the intermediate node does not receive enough acknowledgments

from the downstream checkpoint node, it marks the checkpoint node as a suspect node. It imposes a lot of burden on randomly selected checkpoint nodes and causes message overhead in the network. Disha *et al.* have proposed an efficient algorithm to detect black hole and gray hole attacks [107]. It uses course based detection technique. Each node does not observe all the nodes in the networks. It only observes its neighbor in a routing path. It improves the network lifetime and reduces the control packet overhead.

5.1.3 Trust based Mechanisms

A trust model has been designed to defend against the black hole and gray hole attacks in sensor network [108]. Each sensor maintains a trust value of the neighboring nodes. If the trust value is less than the threshold value, then the node will be avoided during data transmission. It does not consider node residual energy while forwarding the data. Hence, it may decrease the lifetime of low-risk nodes. Buchegger has proposed a cooperation of nodes fairness protocol in the dynamic ad-hoc networks [109]. It adds trust manager and reputation system to the watchdog and path rater scheme. The trust manager evaluates the events reported by the watchdog and sends an alarm packet to the neighboring node. The alarm packet contains the information about a malicious node. The malicious nodes are isolated from the network to provide secure communication. Niki *et al.* have proposed a secured routing protocol based on the trust level of a node [110]. When a sensor wants to deliver the data to the base station, then it selects a sensor which has the highest trust value among its *1-hop* sensors. Xin Li *et al.* have proposed a trust model based on the packet forwarding ratio [111]. A node packet forwarding ratio is defined as the ratio of number of packets forwarded to the number of packets received. The node trust value depends on the packet forwarding ratio. The above trust based mechanisms consider only whether the sensor forwards the packet or drops it while computing the trust value of the particular sensors. In wireless networks, due to network congestion the packet drops also occur. Hence, we should not depend on the packet forwarding ratio parameter only while computing the sensor trust value.

Jianqiao *et al.* have proposed a trust based security mechanism to overcome the above drawbacks [112]. It is a distributed mechanism, where each sensor trust value is computed using three parameters: direct trust value, indirect trust value, and mixed trust value. The direct trust value is generated from the monitoring nodes. The indirect trust value is computed from the recommendation of the indirect neighbors, and aging is performed to compute the mixed trust value. Later, Theodore *et al.* have proposed a detection technique to identify gray hole and black hole attacks [113]. It assigns unique trust weight

for forwarding packets, acknowledgments, integrity, and energy. Based on these weights node trust value is computed. It checks the intermediate node residual energy and trust value while forwarding the data. Hence, the high trust value nodes do not die early. Bin has introduced cloud theory in sensor networks to estimate the sensors' trust value, and it is a cross layered mechanism [114]. Each sensor uses trust expectation, entropy, and ultra entropy metrics to compute its *1-hop* neighbor trust value. Tian *et al.* have proposed a node trust prediction mechanism for sensor networks. It predicts sensor future trust value based on the past behavior evidence. It uses Bayesian network modeling and prediction grading techniques to estimate trust value of the node.

All the existing security mechanisms concentrate only on any one of the attacks from the black hole, gray hole, and sink hole attacks, but not as a whole. In this chapter, a secure coordination mechanism (SCM) is proposed to handle all the three attacks for the delay and energy aware coordination protocol (DEACP) proposed in Chapter 2.

5.2 A Secure Coordination Mechanism (SCM)

In the proposed secure coordination mechanism (SCM), the actors are assumed to be trustworthy. Each actor maintains a master key and the shared key between a sensor and actor is generated using the corresponding sensor id and the master key. The actor securely transfers the shared key to each of its cluster members using Diffie Hellman key exchange method [115]. It allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure channel. The actor does not maintain shared keys for its cluster members and stores only the master key. In SCM, each sensor analyzes the trust level of its *1-hop* sensors based on the experience, recommendation, and knowledge. The sensor transfers its *1-hop* neighbors trust value to the cluster head (actor). The actor analyzes these values and judges the final trust value for each of its cluster members. The analyzed trust values are transferred to the cluster members. Each sensor computes the message authentication code for the data using a secure hash algorithm-3 (SHA-3) and shared key. The obtained message authentication code is appended to the message. The appended data is transferred to the actor through a sensor which has the highest trust value among its *1-hop* neighbors. Once, the actor receives the data then it computes the message authentication code for the received data and compares it with the sender message authentication code. If both are equal then the actor accepts the data. It provides node authentication and data integrity.

5.2.1 Dynamic Trust Model

In SCM, each sensor calculates the trust of its neighbors using three parameters such as experience, recommendation, and knowledge. Initially, every sensor is assigned a value of 0.5. The trust value of a sensor is updated periodically based on the three parameters under consideration [116].

The trust value of sensor S_i is calculated by its neighboring sensor S_j on the basis of its experience as,

$$Tr_{ex} = \begin{cases} 1 - \frac{1}{\{(\frac{A-B}{A+B}) * W_s\}+2} & if \ A \geq B \\ \frac{1}{\{(\frac{B-A}{A+B}) * W_u\}+2} & otherwise \end{cases} \tag{5.1}$$

where, A and B represent the number of successful and unsuccessful transmissions, W_s and W_u denote the weight of successful and unsuccessful transmissions, respectively. They are chosen based on the number of transmissions taken place. The trust experienced for a sensor in the successful transaction is in the range from (0.5 - 1). On the other hand, for an unsuccessful transmission Tr_{ex} is less than 0.5. The trust value of sensor S_j is computed based on the recommendation as,

$$Tr_r = \frac{\sum\limits_{S_k \neq S_i, S_k \neq S_j} Tr_{S_i}^{S_k} * Tr_{S_k}^{S_j}}{\sum\limits_{S_k \neq S_i, S_k \neq S_j} Tr_{S_i}^{S_k}} \tag{5.2}$$

where, $Tr_{S_i}^{S_k}$ is the trust of the sensor S_k given by sensor S_i. $Tr_{S_k}^{S_j}$ is the trust value of sensor S_j as transmitted from the sensor S_k (recommender). The trust value of the recommender as computed by the node S_k has a significant importance in the overall trust value. Trust value computed by sensor S_i for S_j is represented as,

$$Tr_k = W_e Tr_{ex} + W_r Tr_r \tag{5.3}$$

$$\text{with } W_e + W_r = 1$$

where, W_e and W_r are the weights of the trust of experience and recommendation, respectively. In WSAN, the loss of packets is not only due to the malicious nodes, but it also depends on the link quality and network congestion. A sensor should not be judged as a malicious node with one event, so past interactions should also be considered for estimating its trust value. It is also called as knowledge. The total trust value for sensor S_j by sensor S_i is computed as,

$$Tr_n = \alpha Tr_p + (1 - \alpha) Tr_k \tag{5.4}$$

where, Tr_p is the previous trust value of sensor S_j, α is a constant and its range is from 0 to 1. Based on the total trust value (Tr_n) a sensor selects its neighbor to forward its data to the actor.

In our trust model, every sensor computes the trust values for its neighboring sensors and forwards them to its cluster head (actor). The actor analyzes these values and judges the final trust value for each of its cluster member. The analyzed trust values are transferred to the cluster members. The secure backup cluster head (SBCH) selection phase will be enabled, whenever an actor wants to perform action in the event area or leaves the cluster to help its neighboring actors. In a cluster, the sensors which are *1-hop* away from an actor are called as *relay* nodes $R = \{RS_1, RS_2, ..., RS_{rn}\}$. In SCM, before selecting any *relay* node as a backup cluster head, the average trust value (T_{min}) of all the *relay* nodes in a cluster is computed as,

$$T_{\min} = \frac{1}{rn} \sum_{i=1}^{rn} Tr_n \tag{5.5}$$

The *relay* nodes which have more trust value than T_{min} are eligible to act as a backup cluster head. The secure backup cluster head suitability score ($SBCH\_Score$) for a relay node is computed as,

$$SBCH\_Score_{RS_i} = RE_{RS_i} * ND_{RS_i} * Tr_{RS_i} \tag{5.6}$$

Among the eligible *relay* nodes, the node which has the highest secure backup cluster head suitability score is selected as the backup cluster head. Newly elected backup cluster head takes over the role of cluster head and analyzes the trust value of its cluster members.

5.2.2 Secure Hash Algorithm-3 (SHA-3)

Message authentication mechanism allows the destination to check whether the data is sent by the valid source or not and also provides data integrity. In the proposed mechanism, SHA-3 algorithm has been used to provide data authentication while forwarding data to the actor. The functionality of the SHA-3 has proposed by Keccak, and it is accepted by the national institute of standards and technology (NIST) [117]. It consists of four cryptographic hash functions such as SHA3-224, SHA3-256, SHA3-384, and SHA3-512 and two extendable-output functions namely, SHAKE128 and SHAKE256. The hash function works on the binary data and generates fixed length output. The output of the hash function is called as digest or hash value. In SHA3-224, the numerical suffix 224 indicates the length of the digest. The extendable-output function (XOF) is a function that

generates the output digest of any desired size.

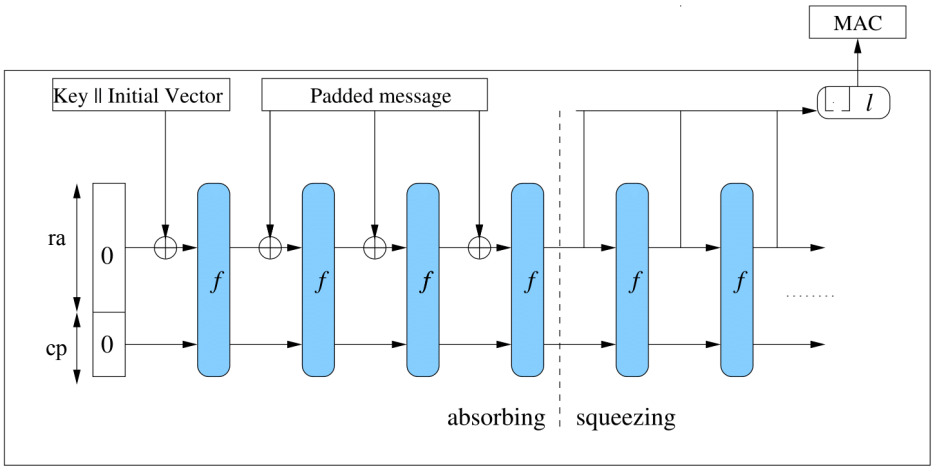


Figure 5.1: Sponge construction to generate message authentication code

Algorithm 6: Sponge construction

Input: ra,bw where ra ≤ bw

Output: string Z with len(Z)= ℓ

SPONGE(f,pad,ra)

Require: $ra < bw$

Interface:$Z = sponge(M, \ell)$ *with* $M \in Z_2^*$, *integer* $\ell > 0$ *and* $Z \in Z_2^\ell$

$P = M||pad[ra](|M|)$; /* Padding */

$s = 0^b$;

$P = P_0||P_1||\ldots||P_w$ *with* $|P_i| = ra$;

for $i = 0 \rightarrow w$ **do**

  $s = s \oplus (P_i||0^{(bw-ra)})$; /* Absorbing phase */

  $s = f(s)$;

end

$Z = \lfloor s_{ra} \rfloor$;

while $|Z| < \ell$ **do**

  $s = f(s)$; /* Squeezing phase */

  $Z = Z||\lfloor s_{ra} \rfloor$;

end

return $\lfloor Z \rfloor_\ell$;

In SHA-3, the six functions are used in the sponge construction. The sponge function is described as sponge(f, pad, ra) where f, pad, and ra denotes an underlying function on fixed-length strings, padding rule, and rate, respectively. The function f maps the single string of fixed length denoted by bw to the strings of same length (bw is called the width of f). The rate ra should be less than the width bw. The padding rule appends a string with appropriate length to another string so that it can be partitioned into a sequence of ra-bit strings. The width (bw) is the summation of capacity (cp) and rate (ra). The capacity (cp) is twice of desired output size (ℓ) i,e., $cp = 2 \times \ell$. The sponge construction framework consists of two stages: absorbing and squeezing as shown in Figure 5.1. In the absorbing phase, the ra-bit input message blocks are XORed into the outer part of the state, interleaved with applications of the function f. In the squeezing phase, the outer part of the state is iteratively returned as output blocks, interleaved with applications of the function f. The number of output blocks is chosen by the user. The number of iterations in the sponge construction is based on the number of bits ℓ requested by the user. The working process of the sponge construction algorithm is described in the **Algorithm** 6.

5.2.3 Countering Sink Hole Attack

In a sink hole attack, the intruder tries to attract all the traffic towards itself using false routing information. Then, the intruder may drop all the traffic or selectively drop few packets. The sink hole attack prevents the actor from obtaining complete sensing information from the sensors, and it causes a lot of problems in the network. In DEACP, a malicious sensor can attract the traffic from its neighboring sensors by announcing false residual energy information. The malicious sensor can drop the entire data received from its neighbors or drop a few packets from a specific neighbor. Figure 5.2 shows the sink hole attack in DEACP. In a sink hole attack, the intruder is not visible. However, his effects are noticeable. Thus, the sink hole attack can be handled by detecting malicious node.

Each sensor forwards the trust values of its neighbors to an actor for handling the sink hole attack. The actor decides the final trust value of its cluster members. Whenever a malicious sensor advertises itself as a better neighbor, then the recommendation to the malicious sensor also increases. The neighboring sensors of the malicious sensor provide a high recommendation. So, the total trust value increases for the malicious sensor. The experience parameter Tr_{ex} assigned by the neighboring sensors for a malicious node will be low because the malicious sensor drops the packets of its neighboring sensors. However, the malicious sensor total trust value Tr_n is definitely above 0.5 due to the high recommendation assigned by other nodes. To overcome this problem, the actor checks for anomalies in total

trust value (Tr_k) and experience trust value (Tr_{ex}) received for the same node in a cluster. On detecting an anomaly, the actor records the set of Tr_{ex} values that do not match with others. The actor computes mean and variance of the set. The mean value allows to find the location of the neighboring node, which is affected by a malicious sensor. The variance is used to identify the degree of an attack. The actor decreases the trust value for all those nodes for secure data transmission in the network.

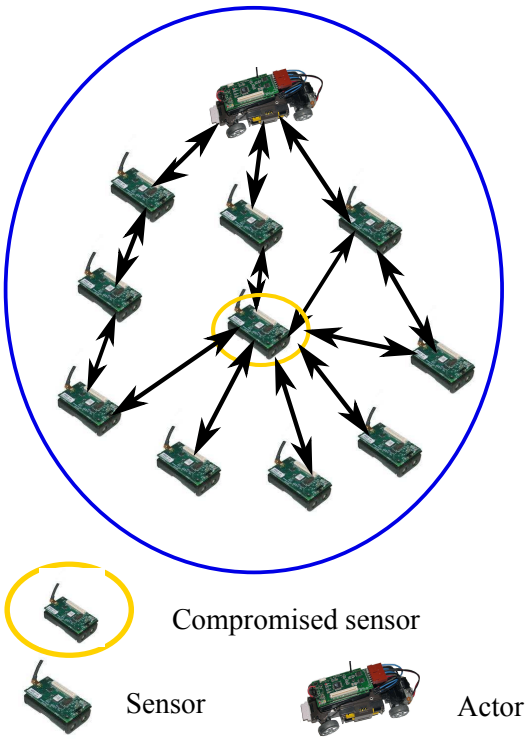


Figure 5.2: Sink hole attack scenario in DEACP

5.2.4 Countering Black Hole and Gray Hole Attacks

The black hole node refuses to forward all the packets and simply drop them instead of forwarding them to the destination. In DEACP, a malicious sensor drops the packets to save its energy instead of forwarding them to the destination as shown in Figure 5.3. In a gray hole attack, the attacker may refuse to forward few packets so that they do not reach to the destination as shown in Figure 5.4. In the proposed trust model, each sensor computes the message authentication code for the data using SHA-3 and appends it to the data. The appended data is transferred to the sensor which has the highest trust value among its *1-hop* sensors.

The cluster head (actor) computes the unique shared key for a sensor by using a master key and the corresponding sensor id. It securely sends the shared key to the corresponding

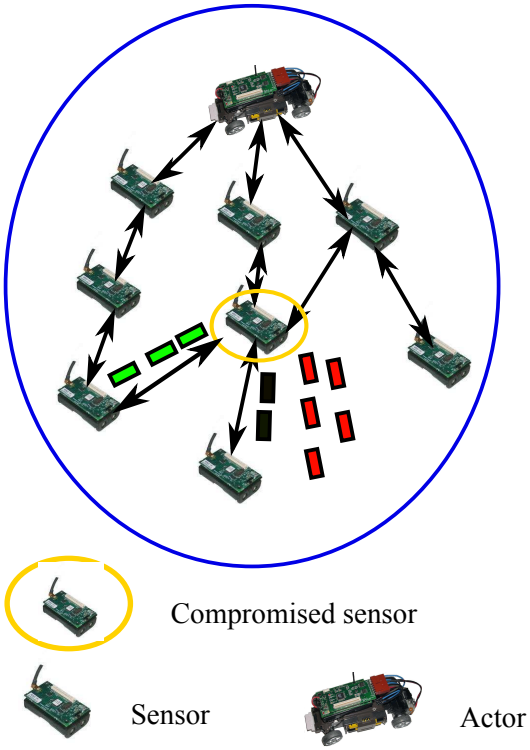


Figure 5.3: Black hole attack scenario in DEACP

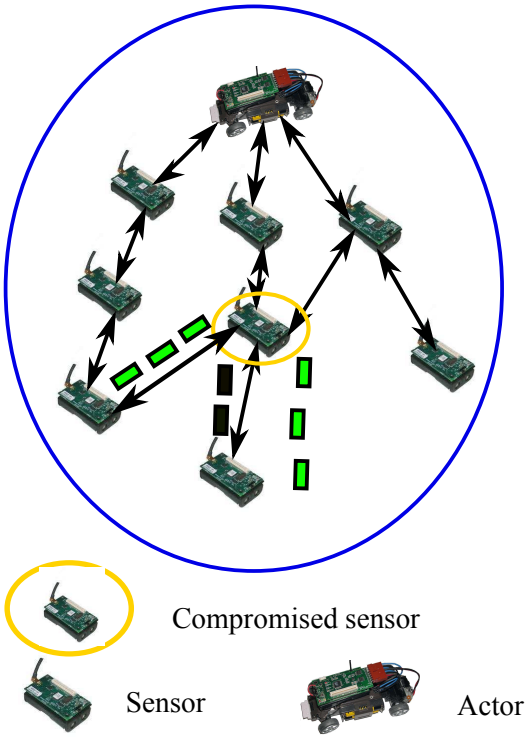


Figure 5.4: Gray hole attack in a selected node scenario for DEACP

sensor. So, whenever a sensor wants to transfer the data to an actor then it uses the shared key to compute the message authentication code. If any sensor joins in the cluster, then the actor transfers the shared key to the corresponding sensor. In sensor-sensor coordination, the sensor S_i computes the hash using SHA-3 algorithm and shared key (between sensor S_i and actor). The hash uses the format as,

$$S_i : h_{S_i} = MAC_{s_{ke}^{S_i}}(S_i, A_k, Rand_i, RE_{S_i}, Data) \tag{5.7}$$

The sensor S_i sends the data and hash to the sensor S_j in the following format.

$$S_i \rightarrow S_j : (S_i, A_k, Data, RE_{S_i}, Rand_i, [], h_{S_i}) \tag{5.8}$$

where, $Rand_i$ represents the freshness of the data. h_{S_i} denotes the hash or digest of the data and RE_{S_i} represents the residual energy of sensor S_i. In sensor-actor coordination, the intermediate sensor S_j performs data aggregation to improve the network lifetime. It computes the hash for the aggregated data using SHA-3 algorithm and shared key (between sensor S_j and actor). The hash uses the following format

$$S_j : h_{S_j} = MAC_{s_{ke}^{S_j}}(S_j, A_k, Rand_j, RE_{S_j}, aggr(Data)) \tag{5.9}$$

The sensor S_i sends the aggregated data and hash to the actor A_k in the following format.

$$\begin{aligned} S_j \rightarrow A_k : (S_j, A_k, Rand_i, Rand_j, RE_{S_i}, RE_{S_j}, \\ [S_i], h_{S_i}, h_{S_j}, aggr(Data)) \end{aligned} \tag{5.10}$$

When the actor receives a packet, then it computes the message authentication code for the received data and verifies it with the sender message authentication code. If both are equal, then the actor accepts the data; otherwise it rejects the data and sends an alarm packet to the source. The message authentication code provides the data integrity and message authentication. According to the expected maximum idle time (EMI) parameter in the DEACP, each sensor should send the data to an actor after waiting for threshold amount of time. If the black hole or gray hole node drops the sensor data, then the actor does not receive data from the sensor after the expected maximum idle time. In this scenario, the actor sends an alarm message to the corresponding sensor. If the alarm message is lost, then the actor waits for a random amount of time and retransmits the alarm message to the corresponding sensor. The sender reduces the experience parameter value Tr_{ex} of the sensor to which it has forwarded the data. Subsequently, it selects next best sensor to forward the data. A sensor is not selected for the routing process if its trust level is less

than the minimum threshold. The average trust value of all the sensors are considered as minimum threshold value. If the sensor wants to participate in the routing process, then it has to forward the packets honestly in the future.

5.3 Simulation Results and Analysis

The performance of secure coordination mechanism (SCM) is evaluated using NS2 simulator. Each sensor is enabled with a single radio, and directional antenna whereas an actor is embedded with two radios for sensor-actor and actor-actor coordination. Multiple channels and omni directional antenna are enabled on each radio of an actor. In our simulation, we have used three different desired output lengths i,e., $\ell \in \{32\}$.

Table 5.1: Simulation parameters for SCM

Parameters	Values
Network Area	$1000 \times 1000\ m^2$
Simulation Duration	200 s
Traffic Flow	CBR
CBR packet interval	0.05 s
Routing protocol	DEACP
MAC protocol	DM-MAC
Seed value	0
Number of Sensors	100 - 1000
Number of Actors	3 - 12
Number of channels	3
width of the Keccak-f	100 bits
Number of rounds n_r	24
Packet Size	64 B
Sensor's Initial Energy	2J

The width of the Keccak-f function (bw), number of rounds (n_r) are fixed to 100 bits and 24, respectively. The width (bw) is the summation of capacity (cp) and rate (ra). The capacity (cp) is twice of desired output size (ℓ) i,e., $cp = 2 \times \ell$. The rate $ra \in \{36\}$ for 100 bits width and 24 rounds. 100 - 1000 static sensors are deployed uniformly in 1000 $\times$ 1000 network area. The optimal number of actors is computed based on the number of sensors and network area. 10 % of sensors are considered as malicious nodes to evaluate the performance of the proposed mechanism. The other network parameters like duration of simulation, traffic flow, and routing protocol are listed in Table 5.1. A radio model is used to compute the energy consumption while transmitting and receiving the data is described in Section 2.1 (Chapter 2).

5.3.1 Simulation Scenario 1

The simulation has been carried out by varying the number of sensors from 100 - 1000 in a step of 100. Three channels are used in the network to provide multi-channel communication in the network. Based on the number of sensors, optimal number of actors varied from 3 to 12. Each active sensor transfers 20 pkts/s. Along with the proposed SCM protocol, recent protocols like Mobile Agent [94] and IDS [95] are also simulated using same parameters for performance comparison. The network metrics like packet delivery ratio, average end-to-end delay, and average energy dissipation in the network are used to analyze the performance of the proposed SCM with existing schemes.

Figure 5.5 depicts the packet delivery ratio of all the three protocols under consideration for number of sensors varied from 100 - 1000 in a step of 100. Due to the malicious sensor nodes the packet delivery ratio in the network decreases as compared to the normal conditions. Further, it decreases in the data forwarding attacks scenario. In SCM, a trust value is assigned to each sensor to mitigate the packet drops in the network. If a malicious sensor drops the packet, then its trust value will be reduced. A sensor is not selected for the routing process if its trust level is less than the minimum threshold value. If the sensor wants to participate in the routing process, then it has to forward the packets honestly in the future. The proposed SCM achieves 10% more packet delivery ratio as compared to the existing Mobile Agent and IDS mechanisms.

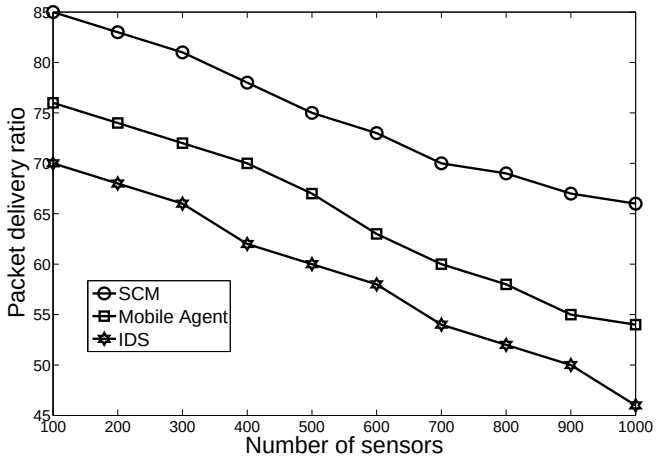


Figure 5.5: Comparative analysis of packet delivery ratio with number of sensors

Due to the sink hole attack the average end-to-end delay increases in the network. In SCM, to handle the sink hole attack, each sensor forwards the trust values of its neighbors to an cluster head (actor). The actor decides the final trust value of its cluster members. The

actor verifies the anomalies in Tr_k, Tr_{ex} values received for the same node in the cluster to identify the malicious node. Figure 5.6 shows that the average end-to-end delay increases with the increase in the number of sensors. It can be observed that the proposed SCM transfers the data with 17% less average end-to-end delay as compared to the existing Mobile Agent and IDS mechanisms.

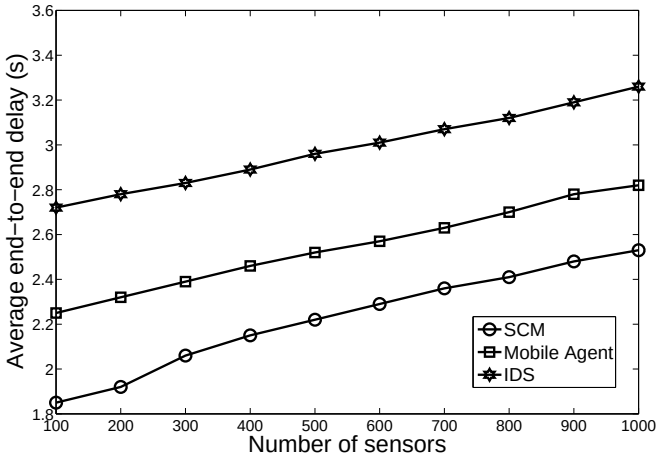


Figure 5.6: Comparative analysis of average end-to-end delay with number of sensors

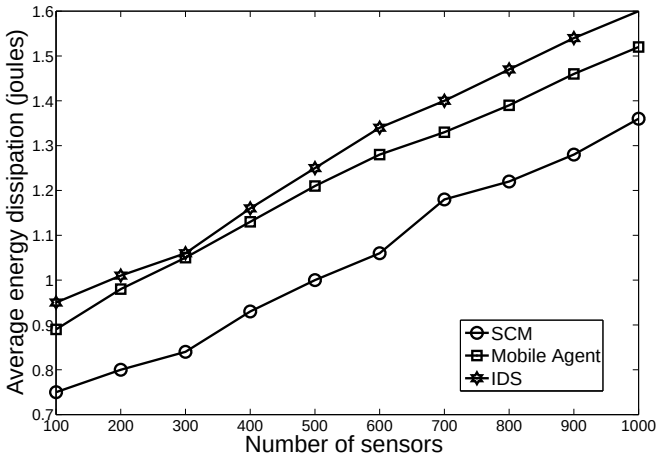


Figure 5.7: Comparative analysis of average energy dissipation with number of sensors

When a malicious sensor drops the packet then the sensors have to retransmit their data to the actor. On the other hand, because of sink hole attack, the sensors may transmit their data to the wrong destination. Hence, it consumes a lot of energy in the network. In the proposed SCM model, the sensor transmits its data to the intermediate node based on its trust value. The sensor whose trust value is less than the minimum threshold value can not

participate in the routing mechanism. Figure 5.7 shows that the SCM protocol consumes 14% less average energy as compared to its competitive security protocols.

5.3.2 Simulation Scenario 2

In this scenario, the data transfer rate varied from 20 pkts/s to 60 pkts/s in a step of 10 pkts/s. The number of sensors and actors are fixed to 500 and 7, respectively. Three number of channels are used to transfer the information in the network. Figure 5.8 illustrates the packet delivery ratio for data transfer rate from 20 - 60 pkts/s. It indicates that the packet delivery ratio decreases with the increase in data transfer rate. It can be observed that the proposed SCM protocol achieves 8% more packet delivery ratio as compared to its competitive security mechanisms.

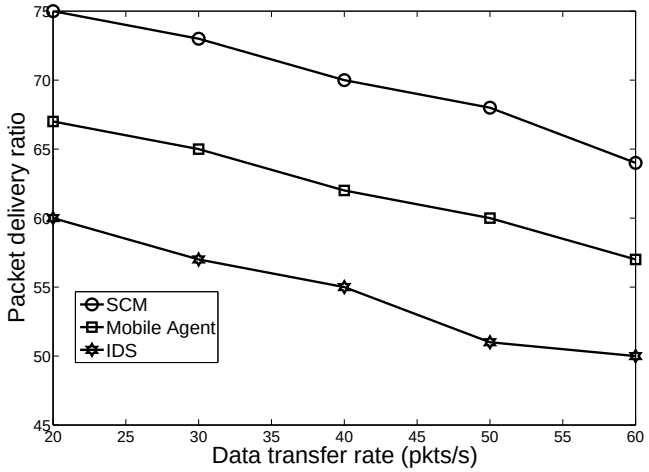


Figure 5.8: Comparative analysis of packet delivery ratio with data transfer rates

The average end-to-end delay of all the three protocols under consideration for 20 - 60 pkts/s data transfer rate is shown in Figure 5.9. In the proposed SCM, each sensor considers its *1-hop* sensor trust value before transmitting data to it. If any malicious node whose trust value is less than the threshold value wants to participate in the communication, then it has to transfer the data honestly to the destination. Hence, the proposed SCM handles the data forwarding attacks properly and delivers data to the destination with 18% less delay as compared to the existing Mobile Agent and IDS mechanisms.

The performance of the three protocols under consideration with respect to average energy dissipation for variable data transfer rate is shown in Figure 5.10. In SCM, the actor performs resource conservative tasks and reduces the burden on sensors to improve

the network lifetime. It can be observed that the proposed SCM consumes 16% less energy as compared to its competitive security mechanisms.

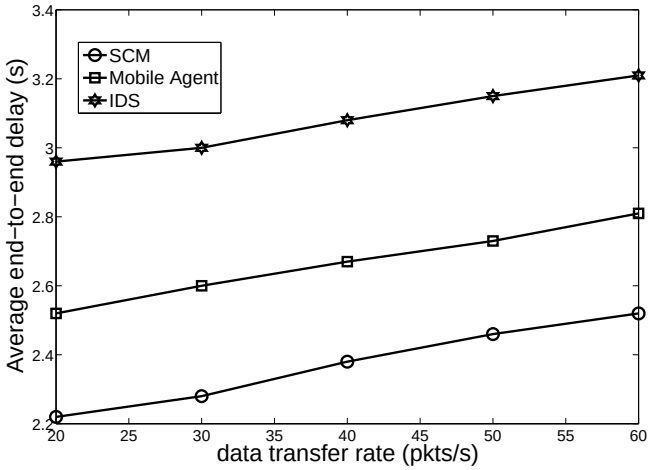


Figure 5.9: Comparative analysis of average end-to-end delay with data transfer rates

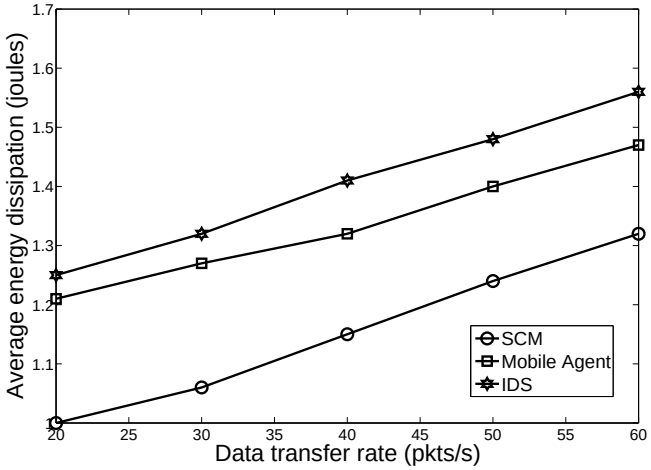


Figure 5.10: Comparative analysis of average energy dissipation with data transfer rates

5.4 Summary

In this chapter, a secure coordination mechanism (SCM) has been proposed to counter the data forwarding attacks for our proposed DEACP (Chapter 2). In the SCM, each sensor analyzes the trust level of its neighboring sensors based on the experience, recommendation, and knowledge. The analyzed trust value is transferred to the actor, and it analyzes these values to identify the malicious nodes in its cluster region. Each sensor computes message

authentication code using the SHA-3 algorithm and appends it to the data. The sensor selects a *1-hop* sensor which has the highest trust value among its neighbors. When the actor receives the data, it computes the message authentication code and verifies it with the received message authentication code. If both are equal, then the actor accepts the data. To evaluate the performance of the proposed mechanism, it is simulated in NS2 and analyzed using QoS metrics such as a packet delivery ratio, average end-to-end delay, and average energy dissipation in the network. The simulation results indicate that the proposed security mechanism performs well as compared to its competitive mechanisms.

Chapter 6

Conclusions

Wireless sensor-actor network (WSAN) is a variant of wireless sensor network (WSN) where there are resource-rich actors work in association with sensors in the area of deployment. Unlike sensor networks, it needs sensor-actor and actor-actor coordination and the protocols that work for WSN need substantial modifications at all layers of network. In this thesis, we have proposed four different protocols for WSAN that work in different layers. In Chapter 2, a delay and energy aware coordination protocol (DEACP) has been developed to deliver the maximum number of packets with in the bounded delay. It is a two-level hierarchical *K-hop* clustering algorithm. In the first level, sensors form a *K-hop* cluster by placing actor nodes as cluster heads and in the second level, sink acts as the cluster head and forms a cluster among actors. The sensors which are *1-hop* away from an actor are represented as *relay* nodes. The actor elects a *relay* node as a backup cluster head (BCH) based on the residual energy and the node degree. BCH resumes the data gathering process when an actor leaves the cluster to help its neighboring actor. Further, a priority based event forwarding mechanism has been proposed to forward an event data based on its bounded delay. The proposed DEACP has been simulated using NS2 simulator. The simulation results indicate that the proposed coordination mechanism outperforms its competitive protocols with respect to event reliability, average event waiting time, and average energy consumption in the network.

In Chapter 3, the suggested interference aware multi-channel MAC (IAMMAC) protocol discusses how channels are assigned for the communication among nodes in DEACP. An actor acts as a cluster head for a *K-hop* sensors and computes the shortest path for all the sensors. An actor partitions the cluster into multiple subtrees and assigns a non-interference channel to each subtree. An actor broadcasts the BCH information to the remaining *relay* nodes using a common control channel. To communicate with BCH, the *relay* sensors utilize the same channel as used by BCH. However, the other cluster members do not

change their data channel. Subsequently, a throughput aware multi-channel MAC protocol has been proposed for actor-actor coordination. The comparative analysis shows that the proposed IAMMAC protocol performs better than the existing MAC protocols in terms different performance parameters.

Even though IAMMAC protocol performance is superior, it is susceptible to be attacked because it uses a single static channel between two sensors in the entire communication. To overcome this problem, in Chapter 4, a lightweight dynamic multi-channel MAC protocol (DM-MAC) has been developed for sensor-sensor coordination. Each sensor dynamically selects a channel which has the highest packet reception ratio among the available channels with the destination. The proposed DM-MAC protocol outperforms its competitive MAC protocols with respect to packet delivery ratio and average goodput parameters.

Finally in Chapter 5, a secure coordination mechanism (SCM) has been proposed to counter the data forwarding attacks which include black hole, gray hole, and sink hole attacks in DEACP. In SCM, each sensor analyzes the trust level of its neighboring sensors based on the experience, recommendation, and knowledge. The analyzed trust value are transferred to the actor, and it analyzes these values to identify the malicious nodes in its cluster region. Each sensor computes message authentication code using a secure hash algorithm-3 (SHA-3) and appends to the data. The sensor selects a neighbor which has the highest trust value among its *1-hop* sensors to transfer its data to the actor. It is inferred from the simulation results that the SCM outperforms its competitive protocols.

Scope for future work

The work described in this thesis unwraps some interesting research directions in WSAN. In coordination mechanism the sensors are deployed uniformly to compute the optimal number of actors. The optimal number of actors computation with random deployment of sensors is not explored in this thesis and can be considered for future study. The accuracy of sensor location and duplicate Hello packets elimination should be analyzed in future. In secure coordination mechanism data forwarding attacks on sensors are discussed in this thesis. Various active attacks such as worm hole attack, node replication attack, sybil attack on sensors can be explored further. In this thesis, we have considered the actors as trustworthy and they are free from attacks. However, further investigations can be made by considering various active and passive attacks on actor nodes in addition to sensors.

Bibliography

[1] I. Akyildiz, S. Weilian, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pages 102–114, August 2002.

[2] D. Trossen and D. Pavel, "Sensor networks, wearable computing, and healthcare applications," *IEEE Pervasive Computing*, vol. 6, no. 2, pages 58–61, April 2007.

[3] A. Ameer and Y. Mohamed, "A survey on clustering algorithms for wireless sensor networks," *Computer Communications (Elsevier)*, vol. 30, no. 14, pages 2826–2841, October 2007.

[4] M. Akbas, M. Erol, and D. Turgut, "Localization for wireless sensor and actor networks with meandering mobility," *IEEE Transactions on Computers*, vol. 64, no. 4, pages 1015–1028, April 2015.

[5] H. Shibo, J. Chen, P. Cheng, Y. Gu, H. Tian, and Y. Sun, "Maintaining quality of sensing with actors in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pages 1657–1667, September 2012.

[6] Z. Cai, X. Ren, G. Hao, B. Chen, and Z. Xue, "Survey on wireless sensor and actor network," in *Proceedings of the 9th IEEE world congress on intelligent control and automation*, pages 788–793, 2011.

[7] X. Li, X. Liang, R. Lu, S. He, J. Chen, and X. Shen, "Toward reliable actor services in wireless sensor and actor networks," in *Proceedings of the IEEE 8th international conference on mobile adhoc and sensor systems*, pages 351–360, 2011.

[8] M. Imran, N. Haider, and M. Alnuem, "Efficient movement control actor relocation for honing connected coverage in wireless sensor and actor networks," in *Proceedings of the IEEE 37th conference on local computer networks*, pages 710–717, 2012.

[9] M. Dong, K. Ota, S. Du, H. Zhu, and S. Guo, "ANTS: Pushing the rapid event notification in wireless sensor and actor networks," in *Proceedings of the IEEE international joint conference on awareness science and technology and ubi-Media computing*, pages 753–758, 2013.

[10] I. Akyildiz and H. Ismail, "Wireless sensor and actor networks: research challenges," *Ad Hoc Networks*, vol. 2, no. 4, pages 351–367, October 2004.

[11] L. Barolli, T. Yang, M. Ikeda, A. Durresi, and F. Xhafa, "A simulation system for routing efficiency in wireless sensor-actor networks: a case study for semi-automated architecture," in *Proceedings of the 14th IEEE international conference on parallel and distributed systems*, pages 567–574, 2008.

[12] V. Ranga, M. Dave, and V. Anil, "A distributed approach for selection of optimal actor nodes in wireless sensor and actor networks," in *Proceedings of the IEEE international conference on contemporary computing and informatics*, pages 312–319, 2014.

[13] S. Sedighian, M. Sharifi, S. Azhari, and H. Momeni, "Service requirements for actor-actor coordination through sensor nodes in wireless sensor actor networks," in *Proceedings of the IEEE international conference on innovations in information technology*, pages 475–479, 2008.

[14] N. Sabri, S. Aljunid, R. Ahmad, M. Malik, A. Yahya, R. Kamaruddin, and M. Salim, "Towards smart wireless sensor actor networks: design factors and applications," in *Proceedings of the IEEE symposium on industrial electronics and applications*, pages 704–708, 2011.

[15] S. Kashi and M. Sharifi, "Connectivity weakness impacts on coordination in wireless sensor and actor networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pages 145–166, February 2013.

[16] A. Manjeshwar and D. Agrawal, "APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Proceedings of the IEEE international Symposium on Parallel and Distributed Processing*, pages 195–203, 2002.

[17] M. Akba, M. Brust, and D. Turgut, "SOFROP: Self-organizing and fair routing protocol for wireless networks with mobile sensors and stationary actors," *Computer Communications (Elsevier)*, vol. 34, no. 18, pages 2135–2146, December 2011.

[18] H. Kim and J. Cobb, "Optimization trade-offs in the design of wireless sensor and actor networks," in *Proceedings of the 37th IEEE conference on local computer networks*, pages 559–567, 2012.

[19] A. Alamuti, "Three protocols for actor selection in wireless sensor and actor networks," in *Proceedings of the IEEE International conference on education and e-Learning innovations*, pages 1–3, 2012.

[20] A. Boukerche, R. Araujo, and L. Villas, "A wireless actor and sensor networks QoS-aware routing protocol for the emergency preparedness class of applications," in *Proceedings of the 31st IEEE conference on local computer networks*, pages 832–839, 2006.

[21] N. Tas, C. Sastry, and Z. Song, "IEEE 802.15. 4 throughput analysis under IEEE 802.11 interference," in *Proceedings of the IEEE international symposium on innovations and real-time applications of distributed sensor networks*, pages 686–689, 2007.

[22] Z. Ngai, E Yangfan, M. Lyu, , and L. Jiangchuan, "Reliable reporting of delay-sensitive events in wireless sensor-actuator networks," in *Proceedings of the IEEE international conference on mobile adhoc and sensor systems*, pages 101–108, 2006.

[23] S. Zhang, X. Wu, and H. Wang, "Length-aware topology reconfiguration in wireless sensor-actor networks to recover from an actor failure," in *Proceedings of the IEEE 33rd Chinese control conference*, pages 304–309, 2014.

[24] S. Kashi and M. Sharifi, "Connectivity weakness impacts on coordination in wireless sensor and actor networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pages 145–166, February 2013.

[25] A. Abbasi, M. Younis, and U. Baroudi, "Restoring connectivity in wireless sensor-actor networks with minimal node movement," in *Proceedings of the IEEE 7th international conference on wireless communications and mobile computing*, pages 2046–2051, 2011.

[26] P. Bose, P. Morin, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless Networks*, vol. 7, no. 6, pages 609–616, November 2001.

[27] W. Chen, J. Hou, and L. Sha, "Dynamic clustering for acoustic target tracking in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pages 258–271, August 2004.

[28] C. Eduardo, D. Manuel, L. Luis, and R. Bartolom, "HERO: A hierarchical, efficient and reliable routing protocol for wireless sensor and actor networks," *Computer Communications*, vol. 35, no. 11, pages 1392–1409, June 2012.

[29] Y. Haidong, M. Huadong, and L. Hongyu, "Coordination mechanism in wireless sensor and actor networks," in *Proceedings of the first international multi-symposiums on computer and computational sciences*, vol. 2, pages 627–634, 2006.

[30] M. Tommaso, P. Dario, C. Vehbi, and I. Akyildiz, "A distributed coordination framework for wireless sensor and actor networks," in *Proceedings of the 6th ACM international symposium on mobile ad hoc networking and computing*, pages 99–110, 2005.

[31] H. Fei, C. Xiaojun, S. Kumar, and K. Sankar, "Trustworthiness in wireless sensor and actuator networks: towards low-complexity reliability and security," in *Proceedings of the IEEE global telecommunications conference*, vol. 3, pages 1696–1700, 2005.

[32] L. Yen-Ting and S. Megerian, "Low cost distributed actuation in large-scale ad hoc sensor-actuator networks," in *Proceedings of the international conference on wireless networks, communications and mobile computing*, vol. 2, pages 975–980, 2005.

[33] K. Shahzad, K. Fazlullah, and S. Afzal, "Delay and throughput performance improvement in wireless sensor and actor networks," in *5th IEEE symposium on information technology: towards new smart world*, pages 1–5, 2015.

[34] D. ZhiCheng, W. Bingwen, L. Zhi, and A. Yin, "VDSPT: A sensor-actor coordination protocol for wireless sensor and actor network based on voronoi diagram and shortest path tree," in *Proceedings of the international symposium on computer network and multimedia technology*, pages 1–4, 2009.

[35] B. McLaughlan and K. Akkaya, "Coverage-based clustering of wireless sensor and actor networks," in *Proceedings of the IEEE international conference on pervasive services*, pages 45–54, 2007.

[36] A. Durresi, V. Paruchuri, and L. Barolli, "Delay-energy aware routing protocol for sensor and actor networks," in *Proceedings of the 11th international conference on parallel and distributed systems*, vol. 1, pages 292–298, 2005.

[37] H. Wen, N. Bulusu, and J. Sanjay, "A communication paradigm for hybrid sensor/actuator networks," in *Proceedings of the 15th IEEE international symposium on personal, indoor and mobile radio communications*, vol. 1, pages 201–205, 2004.

[38] A. Durresi and V. Paruchuri, "Geometric broadcast protocol for sensor and actor networks," in *Proceedings of the 19th international conference on advanced information networking and applications*, vol. 1, pages 343–348, 2005.

[39] E. Cayirci, T. Coplu, and O. Emiroglu, "Power aware many to many routing in wireless sensor and actuator networks," in *Proceeedings of the Second European Workshop on wireless sensor networks*, pages 236–245, 2005.

[40] Z. Yangfan, E. Ngai, M. Lyu, and L. Jiangchuan, "POWER-SPEED: A power-controlled real-time data transport protocol for wireless sensor-actuator networks," in *Proceedings of the IEEE conference on wireless communications and networking*, pages 3736–3740, 2007.

[41] T. Fuhrmann, "Scalable routing in sensor actuator networks with churn," in *Proceedings of the 3rd Annual IEEE communications society on sensor and ad-hoc communications and networks*, pages 30–39, 2006.

[42] L. Fei, "Wireless sensor actuator network for light monitoring and control application," in *Proceedings of the 3rd IEEE conference on consumer communications and networking*, pages 974–978, 2006.

[43] L. Qing, Z. Qingxin, and M. Wang, "Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks," *Computer Communications (Elsevier)*, vol. 29, no. 12, pages 2230–2237, August 2006.

[44] W. Zeng, J. Zhang, and P. Qu, "The study of MAC protocol for industrial wireless sensor network based on ultra-wide band," in *Proceedings of the IEEE international conference on identification, information and knowledge in the Internet of things*, pages 194–197, 2014.

[45] R. Diab, G. Chalhoub, and M. Misson, "Hybrid multi-channel MAC protocol for wireless sensor networks: Interference rate evaluation," in *Proceedings of the 78th IEEE conference on vehicular technology*, pages 1–6, 2013.

[46] M. Kumaraswamy, K. Shaila, V. Tejaswi, K. Venugopal, S. Iyengar, and L. Patnaik, "QoS driven distributed multi-channel scheduling MAC protocol for multihop wsns," in *Proceedings of the IEEE international conference on computer and communication technology*, pages 175–180, 2014.

[47] K. Alexander and B. Lothar, "Performance study of a preamble based MAC protocol in multi-hop wireless networks," in *Proceedings of the IEEE wireless advanced conference*, pages 132–137, 2012.

[48] G. Zhou, C. Huang, T. Yan, H. Tian, J. Stankovic, and T. Abdelzaher, "MMSN: Multi-frequency media access control for wireless sensor networks," in *Proceedings of the 25th IEEE international conference on computer communications (Infocom)*, pages 1–13, 2006.

[49] W. Shih, Y. Tseng, C. Lin, and J. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *The Computer Journal*, vol. 45, no. 1, pages 101–110, August 2002.

[50] K. Umesh, H. Gupta, and R. Samir, "A topology control approach to using directional antennas in wireless mesh networks," in *Proceedings of the IEEE international conference on communications*, vol. 9, pages 4083–4088, 2006.

[51] W. Chen, J. Liu, T. Huang, and Y. Chang, "TAMMAC: an adaptive multi-channel MAC protocol for manets," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pages 4541–4545, December 2008.

[52] Y. Wu, A. Stankovic, H. Tian, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *Proceedings of the 27th IEEE conference on computer communications*, pages 1867–1875, 2008.

[53] T. Carley, A. Moussa, B. Rajeev, and D. Stewart, "Contention-free periodic message scheduler medium access control in wireless sensor/actuator networks." in *24th IEEE real-time systems symposium*, pages 298–307, 2003.

[54] S. Jungmin and H. Nitin, "Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *Proceedings of the 5th ACM international symposium on mobile ad-hoc networking and computing*, pages 222–233, 2004.

[55] J. Chen, S. Sheu, and C. Yang, "A new multichannel access protocol for ieee 802.11 ad hoc wireless lans," in *Proceedings of the 14th IEEE proceedings on personal, indoor and mobile radio communications*, pages 2291–2296, 2003.

[56] L. Jinbao and D. Zhang, "A multi-channel MAC protocol with multiple channel reservation for wireless sensor networks," in *Proceedings of the IEEE international conference on cyber-enabled distributed computing and knowledge discovery*, pages 113–120, 2010.

[57] D. Gong, M. Zhao, and Y. Yang, "A multi-channel cooperative MIMO MAC protocol for wireless sensor networks," in *Proceedings of the 7th IEEE international conference on mobile ad-hoc and sensor systems*, pages 11–20, 2010.

[58] S. Cui, A. Goldsmith, and A. Bahai, "Energy-efficiency of MIMO and cooperative MIMO techniques in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pages 1089–1098, 2004.

[59] P. Bahl, A. Adya, J. Padhye, and A. Walman, "Reconsidering wireless systems with multiple radios," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 5, pages 39–46, 2004.

[60] J. Wang, H. Zhai, and Y. Fang, "Opportunistic packet scheduling and media access control for wireless lans and multi-hop ad hoc networks," in *Proceedings of the IEEE wireless communications and networking conference*, pages 1234–1239, 2004.

[61] K. Ramachandran, E. Belding, K. Almeroth, and M. Buddhikot, "Interference-aware channel assignment in multi-radio wireless mesh networks," in *Proceedings of the 25th IEEE international conference on computer communications (INFOCOM)*, pages 1–12, 2006.

[62] R. Diab, G. Chalhoub, and M. Misson, "Enhanced multi-channel MAC protocol for multi-hop wireless sensor networks," in *Proceedings of the IEEE IFIP wireless days*, pages 1–6, 2014.

[63] ——, "Channel allocation evaluation for a multi-channel MAC protocol," in *Proceedings of the 24th IEEE international symposium on personal indoor and mobile radio communications*, pages 1857–1862, 2013.

[64] Z. Liu and W. Wu, "A dynamic multi-radio multi-channel MAC protocol for wireless sensor networks," in *Proceedings of the IEEE second international conference on communication software and networks*, pages 105–109, 2010.

[65] K. Jagadeesh, B. Majhi, B. Ramesh, and T. Meenakshi, "A multi-channel MAC protocol for actor-actor coordination in WSAN," in *Proceedings of the 35th IEEE TENCON conference*, pages 1–6, 2014.

[66] S. Waharte, B. Ishibashi, R. Boutaba, and D. Meddour, "Performance study of wireless mesh networks routing metrics," in *Proceedings of the international conference on computer systems and applications*, pages 1100–1106, 2008.

[67] K. Phung, H. Tran, J. Tiete, L. Tran, and K. Steenhaut, "Low-overhead time synchronization for schedule-based multi-channel wireless sensor networks," in *Proceedings of the 19th IEEE workshop on local and metropolitan area networks*, pages 1–6, 2013.

[68] W. Ye, J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," *IEEE Transactions on Networks*, vol. 12, no. 3, pages 493–506, June 2004.

[69] L. Peng, Q. Chunming, and W. Xin, "Medium access control with a dynamic duty cycle for sensor networks," in *Proceedings of the IEEE conference on wireless communications and networking*, pages 1534–1539, 2004.

[70] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd international conference on embedded networked sensor systems*, pages 95–107, 2004.

[71] P. Huan and J. Sanjay, "An adaptive mobility-aware MAC protocol for sensor networks," in *Proceedings of the IEEE international conference on mobile ad-hoc and sensor systems*, pages 558–560, 2004.

[72] G. Lu, B. Krishnamachari, and C. Raghavendra, "An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks," in *Proceedings of the 18th international conference on parallel and distributed processing symposium*, pages 224–230, 2004.

[73] T. Van and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st ACM international conference on embedded networked sensor systems*, pages 171–180, 2003.

[74] K. Ramaraju, R. Lydia, K. Rajgopal, and I. Sitharama, "Distributed energy aware MAC layer protocol for wireless sensor networks," in *Proceedings of the IEEE international conference on mobile ad-hoc and sensor systems*, vol. 3, pages 181–186, 2003.

[75] S. Chatterjea, L. Van, and P. Havinga, "AI-LMAC: an adaptive, information-centric and lightweight MAC protocol for wireless sensor networks," in *Proceedings of the sensor networks and information processing conference*, pages 381–388, 2004.

[76] D. Subhasis, R. Amulya, and A. Ajay, "Reliable energy aware multi-token based MAC protocol for wsn," in *Proceedings of the 26th IEEE international conference on advanced information networking and applications*, pages 144–151, 2012.

[77] M. Munir and F. Filali, "Low-energy, adaptive, and distributed MAC protocol for wireless sensor-actuator networks," in *Proceedings of the IEEE 18th international symposium on personal, indoor and mobile radio communications*, pages 1–5, 2007.

[78] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *Proceedings of the international symposium on parallel architectures, algorithms and networks*, pages 232–237, 2000.

[79] T. Luo, M. Motani, and V. Srinivasan, "CAM-MAC: A cooperative asynchronous multi-channel MAC protocol for ad hoc networks," in *Proceedings of the 3rd international conference on broadband communications, networks and systems*, pages 1–10, 2006.

[80] F. Wang, H. Zhao, A. Song, and C. Shi, "Ad hoc networks multi-channel MAC protocol design and channel width adaptation technology," in *Proceedings of the 7th international conference on wireless communications, networking and mobile computing*, pages 1–4, 2011.

[81] I. Wormsbecker and C. Williamson, "On channel selection strategies for multi-channel MAC protocols in wireless ad hoc networks," in *Proceedings of the IEEE international conference on wireless and mobile computing, networking and communications*, pages 212–220, 2006.

[82] O. D. Incel, L. van Hoesel, P. Jansen, and P. Havinga, "MC-LMAC: A multi-channel MAC protocol for wireless sensor networks," *Ad Hoc Networks (Elsevier)*, vol. 9, no. 1, pages 73–94, 2011.

[83] K. Akkaya, F. Senel, and B. McLaughlan, "Clustering of wireless sensor and actor networks based on sensor distribution and connectivity," *Journal of Parallel and Distributed Computing*, vol. 69, no. 6, pages 573–587, June 2009.

[84] M. T, D. Pompili, V. Gungor, and I. Akyildiz, "Communication and coordination in wireless sensor and actor networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 10, pages 1116–1129, October 2007.

[85] D. SrinivasaraRao, B. RameshBabu, V. Srikanth, K. Rajasekhararao, and T. Pavan, "Energy efficient routing protocol for wireless sensor and actor networks," in *Proceedings of the international conference on recent trends in networks and communications, Springer*, pages 114–123, 2010.

[86] M. Alaiwy, F. Alaiwy, and S. Habib, "Optimization of actors placement within wireless sensor-actor networks," in *Proceedings of the 12th IEEE symposium on computers and communications*, pages 179–184, 2007.

[87] F. Tseng, L. Chou, and H. Chao, "A survey of black hole attacks in wireless mobile ad hoc networks," *Human-centric Computing and Information Sciences*, vol. 1, no. 1, pages 1–16. Springer, 2011.

[88] L. Bysani and A. Turuk, "A survey on selective forwarding attack in wireless sensor networks," in *Proceedings of the IEEE international conference on devices and communications*, pages 1–5, 2011.

[89] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, pages 53–57, 2004.

[90] Z. Karakehayov, "Using reward to detect team black-hole attacks in wireless sensor networks," *Proceedings of the workshop on real-world wireless sensor networks*, pages 20–21, 2005.

[91] W. Lou and Y. Kwon, "H-SPREAD: a hybrid multipath scheme for secure and reliable data collection in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 4, pages 1320–1330, 2006.

[92] J. Yin and M. Sanjay, "A hierarchical secure routing protocol against black hole attacks in sensor networks," in *Proceedings of the IEEE international conference on sensor networks, ubiquitous, and trustworthy computing*, vol. 1, pages 376–383, 2006.

[93] M. Satyajayant, K. Bhattarai, and G. Xue, "BAMBi: blackhole attacks mitigation with multiple base stations in wireless sensor networks," in *Proceedings of the IEEE international conference on communications*, pages 1–5, 2011.

[94] D. Sheela, V. Srividhya, B. Asma, A. Anjali, and G. Chidanand, "Detecting black hole attacks in wireless sensor networks using mobile agent," *Proceedings of the international conference on artificial intelligence and embedded systems*, pages 20–21, 2012.

[95] A. Samir, D. Boubiche, and B. Azeddine, "Hierarchical energy efficient intrusion detection system for black hole attacks in WSN," in *World Congress on computer and information technology*, pages 1–5. IEEE, 2013.

[96] S. Marti, J. Thomas, K. Lai, and B. Mary, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proceedings of the 6th ACM annual international conference on mobile computing and networking*, pages 255–265, 2000.

[97] M. Takai, J. Martin, R. Bagrodia, and A. Ren, "Directional virtual carrier sensing for directional antennas in mobile ad hoc networks," in *Proceedings of the 3rd ACM international symposium on mobile ad-hoc networking and computing*, pages 183–193, 2002.

[98] R. Ramanathan, "On the performance of ad hoc networks with beamforming antennas," in *Proceedings of the 2nd ACM international symposium on mobile ad-hoc networking and computing*, pages 95–105, 2001.

[99] L. Eschenauer and V. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM conference on computer and communications security*, pages 41–47, 2002.

[100] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proceedings of the IEEE symposium on security and privacy*, pages 197–213, 2003.

[101] H. Nahas, D. Jitender, and E. Manley, "Secure and energy aware routing against wormholes and sinkholes in wireless sensor networks," in *Proceedings of the First IEEE international conference on communications and networking*, pages 1–8, 2006.

[102] C. Tumrongwittayapak and R. Varakulsiripunth, "Detecting sinkhole attacks in wireless sensor networks," in *Proceedings of the IEEE international joint conference ICROS-SICE*, pages 1966–1971, 2009.

[103] E. Ngai, J. Liu, and M. Lyu, "On the intruder detection for sinkhole attack in wireless sensor networks," in *Proceedings of the IEEE international conference on communications*, vol. 8, pages 3383–3389, 2006.

[104] J. Brown and D. Xiaojiang, "Detection of selective forwarding attacks in heterogeneous sensor networks," in *Proceedings of the IEEE international conference on communications*, pages 1583–1587, 2008.

[105] B. Yu and B. Xiao, "Detecting selective forwarding attacks in wireless sensor networks," in *Proceedings of the 20th IEEE international conference on parallel and distributed processing symposium*, 2006.

[106] B. Xiao, B. Yu, and G. Chuanshan, "CHEMAS: Identify suspect nodes in selective forwarding attacks," *Journal of Parallel and Distributed Computing (Elsevier)*, vol. 67, no. 11, pages 1218–1230, 2007.

[107] G. Disha, B. Kathole, and R. Sapna, "Detecting black and gray hole attacks in mobile ad hoc network using an adaptive method," *International Journal Emerging Technology Advance Engineering*, vol. 2, no. 1, pages 37–41, 2012.

[108] R. Suman, A. Sneha, S. Choudhury, and N. Debnath, "Countering sinkhole and black hole attacks on sensor networks using dynamic trust management," in *Proceedings of the IEEE symposium on computers and communications*, pages 537–542, 2008.

[109] S. Buchegger and J. Boudec, "Performance analysis of the CONFIDANT protocol," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and computing*, pages 226–236, 2002.

[110] P. Niki, T. Ghosh, and K. Makki, "Collaborative trust-based secure routing in multihop ad hoc networks," in *Lecture notes in computer science*, pages 1446–1451. Springer, 2004.

[111] X. Li, Z. Jia, P. Zhang, and H. Wang, "A trust-based multipath routing framework for mobile ad hoc networks," in *Proceedings of the 7th IEEE international conference on fuzzy systems and knowledge discovery*, vol. 2, pages 773–777, 2010.

[112] G. Jianqiao, P. Julong, X. Zhanyi, and L. Ziyin, "Towards trust-based security mechanism for wireless sensor networks," in *Proceedings of the 7th international conference on wireless communications, networking and mobile Computing (WiCOM)*, pages 1–5, 2011.

[113] T. Zahariadis, P. Trakadas, S. Maniatis, P. Karkazis, H. C. Leligou, and S. Voliotis, "Efficient detection of routing attacks in wireless sensor networks," in *Proceedings of the 16th IEEE international conference on systems, signals and image processing*, pages 1–4, 2009.

[114] B. Ma, "Cross-layer trust model and algorithm of node selection in wireless sensor networks," in *Proceedings of the IEEE international conference on communication software and networks*, pages 812–815, 2009.

[115] S. Mukesh, R. Bai, Y. Lin, Y. Wang, M. Yang, and Q. Zhang, "Key management protocols for wireless networks," *Lab for Advanced Networking, Dept of Computer Science, University of Kentucky, Technical Report*, 2004.

[116] S. Roy, S. Singh, S. Choudhury, and N. Debnath, "Countering sinkhole and black hole attacks on sensor networks using dynamic trust management," in *Proceedings of the IEEE symposium on computers and communications*, pages 537–542, 2008.

[117] I. Alshaikhli, M. Alahmad, and K. Munthir, "Comparison and analysis study of SHA-3 finalists," in *Proceedings of the international conference on advanced computer science applications and technologies (ACSAT)*, pages 366–371, 2012.

[118] R. Cai, C. Xue, J. Fu, and Y. Hu, "Integrated MEMS technology," in *Proceedings of the sixth IEEE conference on high density microsystem design and packaging and component failure analysis*, pages 177–180, 2004.

[119] R. Jafari, A. Encarnacao, A. Zahoory, F. Dabiri, H. Noshadi, and M. Sarrafzadeh, "Wireless sensor networks for health monitoring," in *Proceedings of the second annual international conference on mobile and ubiquitous systems: networking and services*, pages 479–481, 2005.

[120] C. Chee-Yee and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *IEEE Proceedings*, vol. 91, no. 8, pages 1247–1256, August 2003.

[121] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, vol. 11, no. 6, pages 6–28, December 2004.

[122] R. Rajagopalan and P. Varshney, "Data-aggregation techniques in sensor networks: a survey," *IEEE Communications Surveys Tutorials*, vol. 8, no. 4, pages 48–63, January 2006.

[123] G. Shah and M. Hassan, "A reliable event response framework for wireless sensor and actor networks," in *Proceedings of the IEEE international conference on advanced information networking and applications*, pages 396–401, 2011.

[124] V. Gungora, M. Vurana, and O. Akanb, "On the cross-layer interactions between congestion and contention in wireless sensor and actor networks," *Ad Hoc Networks*, vol. 5, no. 6, pages 897–909, August 2007.

[125] H. Momeni, M. Sharifi, and S. Sedighian, "A new approach to task allocation in wireless sensor actor networks," in *Proceedings of the first IEEE international conference on computational intelligence, communication systems and networks*, pages 73–78, 2009.

[126] A. Zamanifar, M. Sharifi, and S. Sedighian, "A distributed algorithm for restoring actor-actor connectivity in wireless sensor and actor networks," in *Proceedings of the IEEE international conference on electronic design*, pages 1–6, 2008.

[127] S. Yahiaoui, M. Omar, A. Bouabdallah, and Y. Challal, "Multi-actuators based anycast routing protocol for wireless sensor and actuator networks," in *Proceedings of the IEEE international conference on advanced networking distributed systems and applications*, pages 31–34, 2014.

[128] M. Alaiwy, F. Alaiwy, and S. Habib, "Optimization of actors placement within wireless sensor-actor networks," in *Proceedings of the 12th IEEE symposium on computers and communications*, pages 179–184, 2007.

[129] G. Shah, M. Bozyigit, and F. Hussain, "Cluster-based coordination and routing framework for wireless sensor and actor networks," *Wireless Communications and Mobile Computing*, vol. 11, no. 8, August 2011.

[130] A. Abbasi, F. Younis, and U. Baroudi, "Recovering from a node failure in wireless sensor-actor networks with minimal topology changes," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 1, pages 256–271, August 2013.

[131] N. Sabri, S. Aljunid, R. Ahmad, M. Malik, A. Yahya, R. Kamaruddin, and M. Salim, "Wireless sensor actor networks," in *Proceedings of the IEEE symposium on wireless technology and applications*, pages 90–95, 2011.

[132] G. Yuan, J. Wang, and X. Song, "Data collection scheme of mobile sink in wireless sensor and actor networks," in *Proceedings of the 11th IEEE World Congress on intelligent control and automation*, pages 2505–2508, 2014.

[133] P. Han, W. Huafeng, M. Dilin, and G. Chuanshan, "ELRS: an energy-efficient layered routing scheme for wireless sensor and actor networks," in *Proceedings of the 20th IEEE international conference on advanced information networking and applications*, pages 452–460, 2006.

[134] A. Waseem, H. Jaleel, and M. Egerstedt, "Energy-efficient data collection in heterogeneous wireless sensor and actor networks," in *Proceedings of the 52nd IEEE annual conference on decision and control*, pages 4164–4169, 2013.

[135] S. Muhammad, K. Muhammad, G. Shah, and M. Ahsan, "An efficient and reliable clustering algorithm for wireless sensor actor networks (WSANs)," in *Proceedings of the 53rd IEEE international Midwest symposium on circuits and systems*, pages 332–338, 2010.

[136] B. Javier, M. Diaz, I. Esteve, D. Garrido, L. Llopis, B. Rubio, and J. Troya, "Tc-wsans: A tuple channel based coordination model for wireless sensor and actor networks," in *Proceedings of the 12th IEEE symposium on computers and communications*, pages 173–178, 2007.

[137] V. Rafe, H. Momeni, and M. Sharifi, "Energy-aware task allocation in wireless sensor actor networks," in *Proceedings of the Second IEEE international conference on computer and electrical engineering*, pages 145–148, 2009.

[138] S. Chinnappen-Rimer and G. Hancke, "Actor coordination in wireless sensor-actor networks," in *Proceedings of the annual IEEE India conference (INDICON)*, pages 1–4, 2009.

[139] K. Jagadeesh, B. Majhi, , and B. Ramesh, "A voronoi diagram based efficient coordination mechanism for WSAN," in *Proceedings of the First IEEE international conference on networks soft computing*, pages 226–230, 2014.

[140] K. Jagadeesh and B. Majhi, "A new optimal delay and energy efficient coordination algorithm for WSAN," in *Proceedings of the IEEE international conference on advanced networks and telecommuncations systems*, pages 1–6, 2013.

[141] I. Wormsbecker and C. Williamson, "On channel selection strategies for multi-channel mac protocols in wireless ad hoc networks," in *Proceedings of the IEEE international conference on wireless and mobile computing, networking and communications*, pages 212–220, 2006.

[142] G. Xing, M. Sha, G. Zhou, X. Wang, and S. Liu, "Multi-channel interference measurement and modeling in low-power wireless networks," in *Proceedings of the 30th IEEE real-time systems symposium*, pages 248–257, 2009.

[143] J. Chen, S. Sheu, and C. Yang, "A new multichannel access protocol for ieee 802.11 ad hoc wireless lans," in *Proceedings of the 14th IEEE Proceedings on personal, indoor and mobile radio communications*, pages 2291–2296, 2003.

[144] N. Choi, Y. Seok, and Y. Choi, "Multi-channel MAC protocol for mobile ad hoc networks," in *Proceedings of the IEEE 58th conference on vehicular technology*, pages 1379–1382, 2003.

[145] V. Rajendran, K. Obraczka, and J. Garcia, "Energy-efficient, collision-free medium access control for wireless sensor networks," *Wireless Networks*, vol. 12, no. 1, pages 63–78. Springer, 2006.

Dissemination

Journals

1. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu Battula. Comparative Analysis of Routing Protocols in Wireless SensorActor Networks: A Review *International Journal of Wireless Information Networks*, ***Springer***, Volume 22, Issue 3, pages 220–239, 2015. **DOI** 10.1007/s10776-015-0271-2.

2. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu B. IAMMAC: An Interference aware Multi-channel MAC Protocol for Wireless SensorActor Networks *International Journal of Communication Systems*, ***Wiley***, Volume 29, Issue 4, pages 801–822, 2015. **DOI** 10.1002/dac.3034.

3. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu B. IDMMAC: Interference aware Distributed Multi-channel MAC Protocol for WSAN. *JIPS : Journal of Information Processing Systems*, Korea, 2015. **DOI** 10.3745/JIPS.03.0038 .

4. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu B. A Delay and Energy aware Reliable Coordination Mechanism for WSAN. *International Journal of Communication Systems*, ***Wiley***, 2016. **DOI** 10.1002/dac.3121.

Conferences

1. **Jagadeesh Kakarla** and Banshidhar Majhi. A New Optimal Delay and Energy Efficient Coordination Algorithm for WSAN. In *IEEE International Conference on Advanced Networks and Telecommunications Systems (**ANTS**)* , pages 1–6, Chennai, India, 2013. **DOI** 10.1109/ANTS.2013.6802871.

2. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu B. A Voronoi Diagram based Efficient Coordination Mechanism for WSAN. In *IEEE International Conference on Networks & Soft Computing*, pages 226–230, Guntur, India, 2014. **DOI** 10.1109/CNSC.2014.6906661.

3. **Jagadeesh Kakarla**, Banshidhar Majhi, Ramesh Babu Battula, and Meenakshi Tripathi. A Multi-channel MAC Protocol for Actor-Actor Coordination in WSAN. In *IEEE Region 10 Conference (**TENCON**)*, pages 1–6, Bangkok, 2014. **DOI** 10.1109/TENCON.2014.7022299.

4. **Jagadeesh Kakarla**, Banshidhar Majhi, and Ramesh Babu B. A Trust based Secured Coordination Mechanism for WSAN. In *IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems*, pages 1–5, Kerala, India, February 2015. **DOI** 10.1109/SPICES.2015.7091460.

Jagadeesh Kakarla

Department of Computer Science and Engineering,
National Institute of Technology Rourkela,
Rourkela – 769 008, Odisha, India.
+91 7205345884

jagadeesh0826@gmail.com

Qualification

- PhD (CSE) (*Continuing*)
  National Institute of Technology Rourkela
- M.Tech. (CSE)
  Pondicherry University, Puducherry
- B.Tech. (CSE)
  Jawaharlal Nehru Technological University Hyderabad

Publications

- Journals: 6
- Conferences: 6

Permanent Address

16-1-104/B, Gujarathipeta,
Srikakulam 532 001, Andhra Pradesh.

Date of Birth

26th November 1988