AN ADAPTIVE CONGESTION CONTROL ALGORITHM FOR FAIR AND EFFICIENT DATA TRANSMISSION IN WIRELESS SENSOR NETWORKS

A Thesis

Submitted in partial fulfillment of the requirements for the

award of the degree of

DOCTOR OF PHILOSOPHY

in

Computer Science and Engineering

By

Makul Mahajan

41500119

Supervised By

Dr. Sukhkirandeep Kaur

LOVELY PROFESSIONAL UNIVERSITY

PUNJAB

2022

DECLARATION

I declare that the thesis entitled “An Adaptive Congestion Control Algorithm for Fair and Efficient Data Transmission in Wireless Sensor Networks” has been prepared by me under the guidance of Dr. Sukhkirandeep Kaur, Assistant Professor, Department of Computer Science and Engineering, Lovely Professional University, India. No part of this thesis has formed the basis for the award of any degree or fellowship previously.

Makul Mahajan

School of Computer Science and Engineering

Lovely Professional University

Jalandhar Delhi G.T.Road (NH-1)

Phagwara, Punjab 144411

India

Date: 01-08-2022

CERTIFICATE

This is to certify that the thesis entitled “An Adaptive Congestion Control Algorithm for Fair and Efficient Data Transmission in Wireless Sensor Networks”, Which is being submitted by Mr. Makul Mahajan for the award of Doctor of Philosophy in Computer Science and Engineering, Lovely Professional University, India, in entirely based on work carried out by him under my supervision and guidance. The work embodies the original work of the candidate and has not been submitted to any other university or institution for the award of any degree or diploma, according to best of my knowledge.

Dr. Sukhkirandeep Kaur

Date: 03-08-2022

Sukhkirandeep Kaur Sidhu

Signature of Supervisor

ACKNOWLEDGEMENT

First of all, I would like to express my thankfulness to my supervisor, Dr. Sukhkirandeep Kaur, for her supervision, leadership and advice from the very early stage of this research work as well as giving me astonishing experiences throughput the work. I am truly very lucky to have the opportunity to with her. I found this guidance to extremely valuable.

I would like to show my gratitude to Lovely Professional University for providing me a right research atmosphere to carry out my research work in suitable time. I would like to thanks the division of research and development and School of Computer Science and Engineering for all provision and inspiration throughout the Ph.d duration.

I am also very thankful to my parents and all family members for their continuous support and encouragement. I am also very thankful to the entire faculty member for their continuous encouragement and support.

Finally, I am thankful to god for showing me the right path from time to time during my research work.

Makul Mahajan

Date: 01-08-2022

Abbreviations

Abbreviations	Description
WSNs	Wireless Sensor Networks
OSI	Open System Interconnection
BS	Base Station
ADC	Analog to digital Convertor
RF	Radio Frequency
ECG	Electrocardiogram
EMG	Electromyogram
IA	Industrial Automation
DSS	Decision Support System
QoS	Quality of service
MAC	Medium access layer
EWMAs	Exponentially weighted moving averages
ACO	Ant Colony Optimization
RT	Real Time
CBR	Constant bit rate
FEACC	Fair, efficient, and adaptable congestion control
AODV	Ad-hoc on demand distance vector
DoS	Denial of service
RREQs	Route request
RERR	Random error
FDMM	Flooding detection and mitigation mechanism
RREP	Route reply packet
DRP	Dummy reply packet

ABSTRACT

Wireless Sensor Networks (WSNs) is a collection of tiny sensor nodes that communicate with each other to route the information sent by the course node to sink node which is also called base station. A typical sensor network contains source node which generates an event or sense some data and a sink node which is used for process the data collected source node. The intermediate sensor nodes are used or act as a router to route the date from one node to another node until it reaches the destination node. There are lot of challenges are there while routing the data in sensor network like energy consumption, resource constraints, security, mobility, node deployment, localization, congestion etc. This thesis focuses on congestion control algorithm primarily and problem of congestion control is divided into three objectives. Basically, congestion occurs when the intermediate is not able to accommodate the data in its queue and when it is not able to accommodate the data, packets start dropping which thereby decreases throughput, packet delivery fraction, network's energy and increases end to end delay. So in order to make an efficient network, it is mandatory to apply a suitable congestion control algorithm for better processing of network operation. This thesis has been divided into 3 objectives.

In the first objective, we have designed a novel algorithm for controlling the congestion by routing the traffic through multiple paths using modified Ant Colony Optimization algorithm. The basic goal of first objective is to find an optimal path from list of viable paths for routing the sensitive traffic in the event of congestion occurrence. Initially, the algorithm check the status of congestion in the network and if at any node congestion occurs, algorithm, will try to find he optimal path from set of available paths in the network using path preference probability. The node with maximum path preference probability among the set of neighbor nodes will be selected for routing the sensitive traffic thereby controlling the congestion and improving the network performance.

Nature of WSNs is such that, when an event occurs, the load on the network increases unexpectedly which results in congestion on the sensor nodes which are unable to process the data generated quickly. In the second objective, we have

proposed a novel algorithm which will adaptively provide a fair and efficient rate to each flow in the network by allocating approximately fair bandwidth. In the proposed work, we have used available bandwidth resources, and active flows belonging to a particular node to estimate the fair rate that can be efficiently assigned to each node in the network. We suggest a new congestion regulator scheme that attains a roughly equal bandwidth distribution to all the nodes in the network. Intuitively, we expect each flow to obtain a proportionate segment of the existing bandwidth based on its generating rate.

In the third objective, we have designed an algorithm for controlling the congestion in WSNs under the network attack. In WSNs, there are number attacks which can degrade the performance of network by launching a particular attack on the network. In the proposed algorithm we have controlled the congestion in the presence of flooding attack. In this attack, the malicious node tries to flood the intermediate nodes with fake packet requests there by exhausting the network resources and make the overall network congested. In this objective, the algorithm will initially identify the node causing the flooding attack, and then we will isolate that node for further communication. Once algorithm will be able to identify the malicious node causing the flooding attack, the algorithm will send a control message to all the nodes in the network to reject the route request from detected node.

So overall in this thesis, we have designed three algorithm for controlling the congestion in different scenarios and overall results of our simulation is positive and tend to improve the performance of overall network in terms of various network parameters.

Table of Contents

DECLARATION ii

CERTIFICATE iii

ACKNOWLEDGEMENT iv

ABBREVATIONS v

ABSTRACT vi

TABLE OF CONTENTS viii

LIST OF FIGURES xii

LIST OF TABLES xiv

1. Introduction 1

1.1 Wireless Sensor Networks 1

1.1.1 Wireless Sensor Networks Architecture 2

1.1.2 Components of sensor nodes 4

1.2 Constraints and Challenges in WSNs 6

1.3 Applications of WSNs 15

1.3.1 Environmental/Earth monitoring 16

1.3.2 Healthcare applications 17

1.3.3 Industrial applications 17

1.3.4 Military Applications 18

1.3.5 Flora and fauna Application 19

1.3.5.1 Greenhouse surveillance 19

1.3.5.2 Crop Monitoring 19

1.3.5.3 Livestock Monitoring 19

1.4 Congestion 20

1.4.1 Congestion in Wired-Networks 20

1.4.2 Congestion in WSNs 20

1.4.2.1 Congestion detection 24

1.4.2.2 Congestion Notification 25

1.4.2.3 Congestion Control 26

1.5 Aim and objective of the Thesis 28

1.6 Main contribution and results in brief 30

2. Review Of Literature 31

2.1 Introduction 31

2.2 Techniques and Procedures proposed for controlling congestion and for evaluating the optimal path 31

2.3 Techniques and Procedures proposed for fair rate assignment for congestion control 53

2.4 Techniques and Procedures proposed for detecting and mitigating the effect of adversaries in WSNs 67

2.5 Limitation of existing techniques identified from review of literature 83

3. An intelligent path evaluation algorithm for congestion control in WSNs 85

3.1 Introduction and Problem statement 88

3.2 Working of Traditional Ant Colony Optimization Algorithm 90

3.3 Proposed Algorithm: Congestion Detection and Path Evaluation Algorithm - CDPEA 90 92

3.3.1 Congestion prediction model 95

3.3.2 Path evaluation model 95

3.4 Complexity Analysis of Proposed algorithm 95

3.4.1 Time complexity of congestion detection module 97

3.4.2 Time complexity of Path Evaluation Module 98

3.5 Simulation and results 99

3.5.1 Simulation results by varying the number of nodes 99

3.5.1.1 Average Energy Consumption Analysis 100

3.5.1.2 Average Throughput Analysis 102

3.5.1.3 Average Packet Loss Rate Analysis 103

3.5.1.4 Average End to End Delay 105

3.5.2 Simulation results by varying the simulation time Average End to End Delay 106

3.5.2.1 Average Energy consumption versus simulation time analysis 107
3.5.2.2 Average Throughput versus simulation time analysis 108
3.5.2.3 Average Packet loss rate versus simulation time analysis 110
3.5.2.4 Average End to End delay versus simulation time analysis 111
3.6 Conclusion of the chapter
4. A fair and efficient rate allocation based congestion control in WSNs 112
4.1 113
4.2 Network Model and problem statement 114
4.2.1 Problems in existing systems and motivation behind proposed solution 115
4.3 Proposed Work: Fair, Efficient, and Adaptable congestion control– FEACC 120
4.4 Complexity Analysis of proposed algorithm– FEACC 121
4.5 Simulation and Result analysis 122
4.5.1 Simulation results by varying number of nodes 123
4.5.1.1 Average Throughput Analysis 124
4.5.1.2 Average Packet delivery ratio 126
4.5.1.3 Average Energy Consumption Analysis 127
4.5.1.4 Average End to End delay 129
4.5.2 Simulation results by variation of offered traffic load 130
4.5.2.1 Average Throughput Analysis 131
4.5.2.2 Average Packet Delivery Ratio Analysis 132
4.5.2.3 Average Energy Consumption Analysis 133
4.5.2.4 Average End to End Delay Analysis 135
4.6 Conclusion of the chapter

5 Detection and Mitigation of network adversaries in WSNs in the event of congestion

5.1 Introduction 136

5.2 Flooding attacks in WSNs 136

5.2.1 Types of Flooding attacks 137

5.3 Problem Statement 139

5.4 Proposed Algorithm – FDMM 140

5.5 Complexity analysis of Flooding detection and Mitigation Mechanism (FDMM) 144

5.6 Simulation setup and Result analysis 146

5.6.1 Performance Metrics 147

5.6.1.1 Average Packet Delivery Ratio Analysis 147

5.6.1.2 Average Throughput Analysis 149

5.6.1.3 Average Residual Energy Analysis 150

5.6.1.4 Average End to End Delay 152

5.7 Conclusion of the Chapter 154

6 Conclusion and Future Scope 155

BIBLIOGRAPHY 158

Publications 169

LIST OF FIGURES

Figure Number	Figure Name	Page Number
1.1	Basic architecture of WSNs.	2
1.2	Layered communication architecture	3
1.3	Clustered Network Architecture	4
1.4	Components of sensor nodes	5
1.5	Single hop and multi-hop communication in WSNs	7
1.6	Node level congestion in WSNs	21
1.7	Link-level congestion in WSNs	22
1.8	Congestion process in WSNs	24
3.1	Flowchart of the proposed algorithm	97
3.2	Average energy consumption of proposed algorithm, AntSensse, FCC and CODA algorithm versus number of nodes	100
3.3	Average Throughput analysis versus number of nodes for proposed algorithm, AntSense FCC and CODA algorithm	101
3.4	Average Packet loss rate analysis versus number of nodes for proposed, AntSense, FCC and CODA algorithm	103
3.5	Average End to End delay analysis versus number of nodes for proposed, Basis AntSense, FCC and CODA algorithm	104
3.6	Average Energy Consumption versus simulation time for proposed, Basis AntSense, FCC and CODA algorithm	106
3.7	Average Throughput versus simulation time for proposed, AntSense, FCC and CODA algorithm	107
3.8	Average packet loss ratio versus simulation time for proposed, AntSense, FCC and CODA algorithm	109

3.9	Average End to End delay for proposed, AntSense, FCC and CODA algorithm versus simulation time	110
4.1	Working of proposed protocol FEACC	116
4.2	Flowchart depicting the proposed algorithm FEACC	121
4.3	Average Throughput analysis of proposed algorithm - FEACC, CCF and CAP algorithm	124
4.4	Average Packet Delivery ratio analysis of proposed algorithm – FEACC, CCF and CAP algorithm	125
4.5	Average Energy Consumption analyses of proposed algorithm – FEACC, CCF and CAP	127
4.6	Average End to End delay analyses of proposed algorithm – FEACC, CCF and CAP	128
4.7	Average Throughput analysis of proposed algorithm – FEACC, CCF and CAP	130
4.8	Average Packet delivery Ratio analysis of proposed algorithm – FEACC, CCF and CAP	131
4.9	Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP	133
4.10	Average End to End delay analysis of proposed algorithm – FEACC, CCF and CAP	134
5.1	Basic scenario of flooding attack in WSNs	138
5.2	Types of Flooding Attack	139
5.3	Flowchart depicting the proposed algorithm (FDMM)	145
5.4	Average Packet delivery fractions versus number of nodes	148
5.5	Average Throughput Analysis versus number of nodes	150
5.6	Average Residual Energy versus number of nodes	151
5.7	Average End to End delay versus number of nodes	153

LIST OF TABLES

Table Number	Table Name	Page Number
2.1	Literature survey of Congestion control and path Evaluation algorithms	45
2.2	Literature survey of fair rate assignment for congestion control	61
2.3	Literature survey of network adversaries – detection and mitigation algorithms	76
3.1	Time complexity of congestion detection module	96
3.2	Time complexity of Path Evaluation module	97
3.3	Simulation parameters and rules	98
3.4	Average Energy consumption of proposed, Antsense, FCC and CODA algorithm	100
3.5	Average Throughput analysis of proposed, basic AntSense, FCC and CODA algorithm	102
3.6	Average packet loss ratio of proposed, AntSense, FCC and CODA Algorithm	103
3.7	Average End to End delay of proposed, AntSense, FCC and CODA algorithm	105
3.8	Average Energy consumption of proposed, FCC, AntSense and CODA algorithm versus summation time	106
3.9	Average Throughput consumption of proposed, AntSense, FCC and CODA algorithm versus summation time	108
3.10	Average packet loss ratio of proposed, AntSense, FCC and CODA algorithm versus summation time	109
3.11	Average End to End delay for proposed, AntSense, FCC	110

	and CODA algorithm versus simulation time	
4.1	Time complexity of Fair, Efficient, and Adaptable congestion control - FEACC	120
4.2	Simulation parameters and rules	122
4.3	Average Throughput analysis of proposed algorithm - FEACC, CCF and CAP algorithm	124
4.4	Average Packet Delivery ratio analysis of proposed – FEACC, CCF and CAP.	126
4.5	Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP.	127
4.6	Average End to End delay analysis of proposed algorithm – FEACC, CCF and CAP	129
4.7	Average Throughput analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load	130
4.8	Average Packet Delivery Ratio analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load	132
4.9	Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load	133
4.10	Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load	134
5.1	Time complexity of Flooding Detection and Mitigation Mechanism	144
5.2	Simulation Parameters and Rules	147
5.3	Simulation result of packet delivery fraction	148
5.4	Simulation result of throughput analysis	149
5.5	Simulation result of Average Residual Energy	151
5.6	Simulation results of Average End to End Delay	153

CHAPTER 1

Introduction

1.1 Wireless Sensor Networks

Wireless Sensor Networks (WSNs) have been regarded as a primary technology of an evolving field that is aimed to resolve various problems, while also facilitating the accommodation of new advanced services and providing an effective process to the ever-increasing user demands. WSNs are self-organized and infrastructure-free networks in which a number of small and frivolous sensor nodes are installed in the environment to monitor dissimilar physical phenomenon of interest and events. Most of the WSNs used one sink node for communication, and these networks seemed to have a delinquent with scalability as the number of sensor nodes in the network and the size of data gathered by sink nodes increased. Furthermore, network performance cannot be estimated without taking network size into consideration [1] [2] [3]. WSNs in general possess multiple sinks for communicating information between nodes. Multiple sink nodes lessen the chance of isolated clusters of sensor nodes that are unable to distribute their information due to adverse environmental conditions. Because a large number of sensor nodes improve performance, these networks are more adaptable and scalable. However, multiple sink network scenarios are not a natural generalization of single sink network scenarios. Occasionally, the sensor nodes send the obtained data to a single sink node chosen from a pool of several, and then the data is sent to the gateway for transmission to the final users. For data transmission, a suitable sink node can be identified from a multitude of alternatives. Sensor nodes transfer information to a chosen sink node among different sink nodes in different scenarios, and the sink node then broadcasts the information to the final user [4][5].

The basic structure of WSNs is shown in figure 1.1.

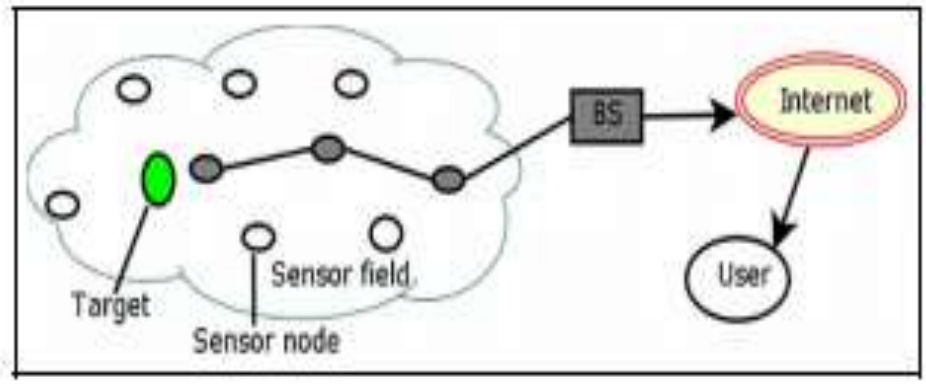

Figure 1.1: Basic architecture of WSNs.

1.1.1 Wireless Sensor Networks Architecture

WSN architecture is divided into two types: layered architecture and clustered architecture. The following is a detailed description of architectures:

- Layered Network Architecture
- Clustered Network Architecture [5] [6].

Layered Network Architecture

The Open Systems Interconnection (OSI) model is another name for the layered communication architecture. The OSI model is used in the majority of WSN architectures. Both sink and source nodes transmit data in layered communication architecture. The communication architecture is divided into five layers and three cross-planes (power management plane, mobility management plane, and task management plane). The power management plane is in charge of the node's power consumption, the mobility management plane is in charge of identifying the node's mobility and maintaining information about neighboring nodes, and the task management plane is in charge of scheduling the sensing task in the given area [5]. The layered network architecture is shown in the figure 1.2

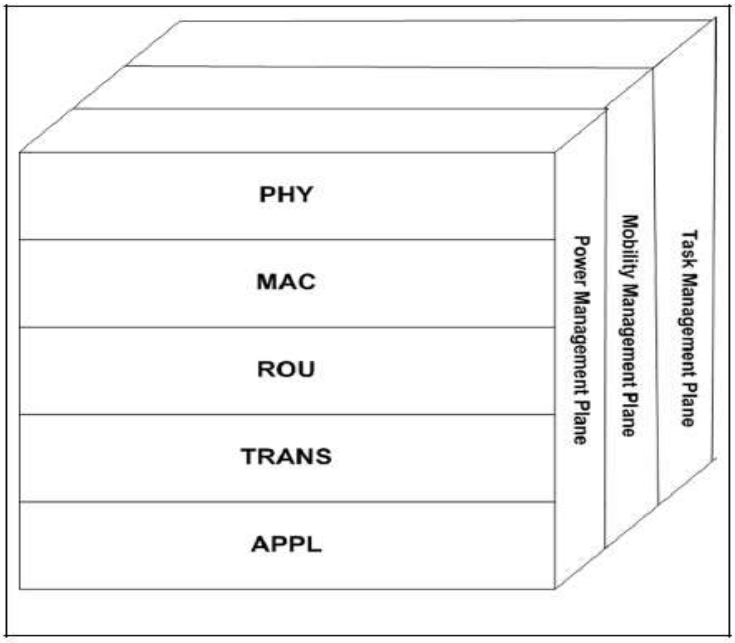

Figure 1.2: Layered communication architecture [4].

Clustered Network Architecture

The clustered architecture was created with the idea that sensor nodes with higher energy are used for data transmission from one point to another, while sensor nodes with lower energy perform sensing tasks in specific fields. In this architecture, sensor nodes are organized into clusters. Each cluster has a cluster head who oversees the activities of all cluster members. Some nodes are elected as cluster heads based on energy, and the remaining nodes in the cluster become members of that cluster. Each member transmits the data gathered after sensing the target of interest to the cluster head. Furthermore, cluster heads broadcast the data to base station (BS). Creating clusters and delegating particular duties to cluster heads can greatly improve WSN lifespan, scalability, and energy efficiency. It demonstrates that the sensed information arrives at BS after two hops of communication. The clustering can be extended hierarchically for more effective communication. Because of its essential suitability for data fusion, the clustering architecture is generally advantageous to sensor networks. Cluster heads receive information from all cluster group members, and only the most important information is

extracted. Following that, the data is transmitted to BS. Base stations can be either mobile or stationary, and they are connected to the network in a sequential manner. The actuator, energy unit, storage, transceiver, and CPU are all part of this tiny sensor. Due to resource constraints, sensor nodes have limited energy, storage, and computer capacity [6] [7]. The clustered architecture is shown in the figure 1.3.

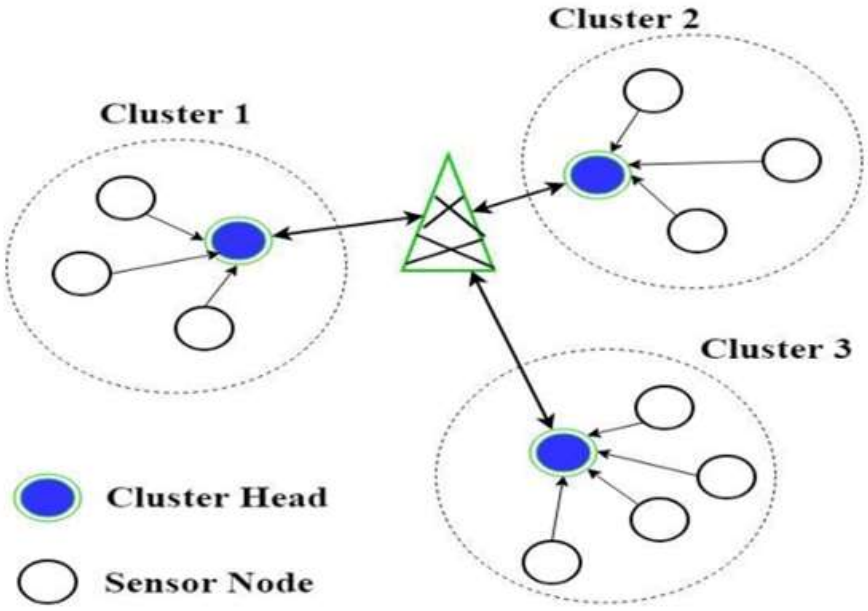

Figure 1.3: Clustered Network Architecture

1.1.2 Components of sensor nodes

WSNs are a network of geographically distributed sensor nodes that interact over a wireless channel. Recent advancements in micro electromechanical devices, small energy devices, and extremely embedded digital electronics have contributed to the growth of micro sensors [7]. The various components of sensor nodes are explained below in figure 1.4:

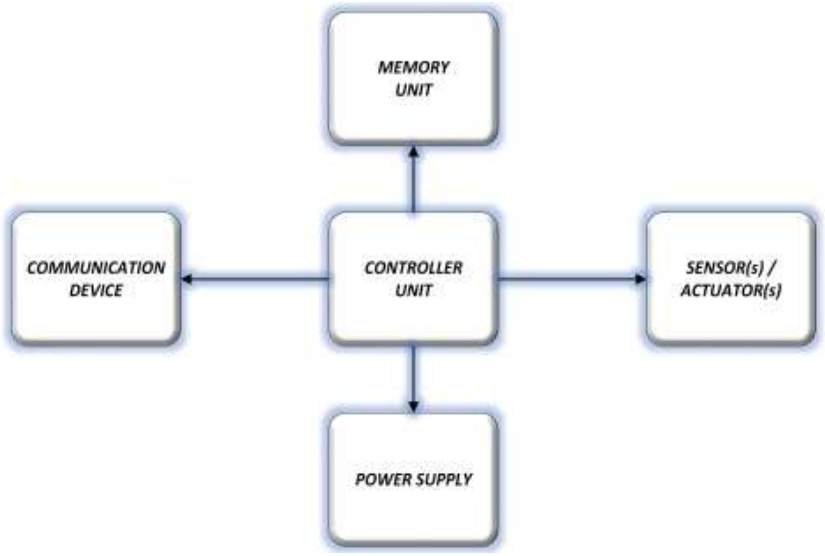

Figure 1.4: Components of sensor nodes

Memory Unit

The sensor nodes' sensed data is stored in the memory unit. Because sensors are so small, WSNs networks have limited memory.

Power Supply

The critical component of energy storage is the power system. The batteries in the sensor nodes can be rechargeable or non-rechargeable. Solar energy, for example, is used in the form of photovoltaic panels and cells to supplement battery storage, wind turbine power, and so on.

Controller

The microcontroller unit is in charge of a variety of tasks, including data processing and the control of other node components. This is the primary component of the device that controls and manages all of the node's components. The controller device can be a small storage device built on an embedded panel or a built-in memory.

Sensor Unit

This unit is also the primary component of the sensor node, which differentiate it from all other embedded systems in terms of communication capability. It is mainly made up of several sensors that collect information from numerous environmental and physical conditions. A sensor unit detects physical activity such as temperature, pressure, heat, sound, and light. The input detected is converted into digital signals by using ADC (Analog to digital converter) installed on the device.

Communication device

In general, a Communication Device is used to facilitate communication from a sensor node to a sink node for information exchange. With this technological advancement, the reduced size and cost of devices ignited people's interest in the potential use of a large number of unmonitored disposable devices.

1.2 Constraints and Challenges in WSNs

WSNs are designed and deployed to perform various tasks associated with object detection and recognition. Sensor nodes can be deployed in various remote regions that are unavailable to wires and heavy devices. Sensor nodes in the network may be located many hops apart from one another, and direct communication precipitated more power. When all sensor nodes are large enough that data can be transmitted directly to the base station using a single hop, this is referred to as one-hop communication. However, because sensor networks frequently span large geographical areas, radio transmitting capacity should be kept to a bare minimum in order to reduce energy consumption. Multi-hop communication is preferred in WSNs, which have a mesh topology. Sensor nodes of this type not only collect information, but also act as a relay for other sensor nodes, allowing them to communicate sensor data to a base station. The figure represents the single hop and multi hop communication in the network [8] [9].

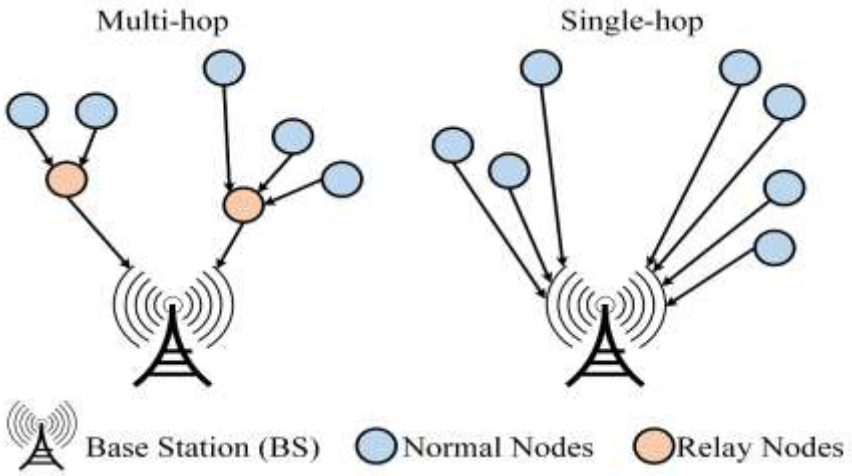

Figure 1.5 Single hop and multi-hop communication in WSNs

Various constraints and challenges in WSNs are considered below in detail:

Energy Consumption

The unique characteristics of each sensor node present a variety of challenges to the communication protocol, most particularly in terms of energy consumption. Each sensor node has a limited capacity, and these nodes are powered by batteries, which must be recharged or replaced when depleted. Both options are inappropriate for some nodes and are simply discarded when their energy runs out [9].

Wireless Medium

WSNs communicate using wireless media, and a sensor network developer faces several challenges when relying on wireless networks and communications. A Radio Frequency (RF) signal travels through a wireless medium that is influenced by large and small scales. As the distance between sensor nodes and base stations grows, so does the required power for transmission. As a result, it is more power efficient to divide a large distance into several smaller distances, promoting multi-hop transmission and tracking [10].

Network Lifetime

Sensor node network lifetime is also a critical issue. It is difficult to increase the lifetime of each sensor node without increasing power. One of the most common network life expressions is "the period when the first network server does not have the power to transmit a signal," because losing a node may result in the network losing some features [10] [11].

Security

Because of the wireless medium used for transmission, WSNs security is a major challenge. Communication in WSNs takes place over an open wireless channel, which means that anyone with a wireless communication device operating in the same frequency band can receive data packets. An attacker can destroy a wireless network by listening in, spooking, and attacking data packets. An attacker can alter received data packets and re-inject them into the network [11].

Mobility

The position of nodes may change after the initial deployment due to environment requirements. Node mobility can be caused by environmental factors such as water, wind, or other natural factors; sensor nodes may be attached by moving objects and may be controlled by automation. In other words, sensor node mobility may be an unintended consequence or an essential requirement of the system. Mobility can be implemented for all sensor nodes in the network or for specific nodes. As a result, node mobility has an impact on network performance [11].

Node Deployment

Another design issue in the WSNs is the deployment of sensor nodes, and the position of sensor nodes must be predetermined. Sensor nodes are distributed at random in remote areas, terrain, or disaster relief operations. This random deployment necessitates the use of self-organizing protocols for the communication protocol stack. Each node must handle itself in such a way that it can configure itself, cooperate with other nodes, adjust to loss, and handle itself in changing environments without human

interference. All of these characteristics must be designed to avoid the use of unnecessary overhead energy [11].

Hardware and Software Issues

WSNs are made up of a large number of sensor nodes for sensing purposes. These sensor nodes should be inexpensive and lightweight. Sensor node storage memory should be kept as low-cost as possible. The limited memory of each sensor node is also a challenge, and for some nodes, a large memory for storage is required. The CPU is in charge of regulating the computational capability and energy consumption of sensor nodes. Radio range ensures network connectivity and data collection in the physical environment that sensor nodes monitor. The software should be light, low-cost, energy-efficient, and hardware-independent [11].

Self-Management

WSNs are made up of a large number of sensor nodes that are typically deployed in a static configuration. However, due to node failure, the topology of the WSN changes often. Additionally, additional sensor nodes can be redeployed at any time, resulting in network reorganization. As a result, a sensor network system must be adaptive to changes in connectivity. The network topology is continually changing throughout network operations. Once deployed, WSNs should be self-contained and capable of network configuration, adaptation, repair, and maintenance [12].

Heterogeneity

Sensor nodes are nodes that have diverse sensing, calculation, and processing capabilities. When two completely independent WSNs communicate with each other, heterogeneity exists. Heterogeneity can invent different challenges to communications and network configuration. New routing protocols should be designed and implemented with the goal of judiciously utilizing sensor node heterogeneity to extend the lifetime of WSNs [12].

Redundant Data

One of the primary goals of WSNs is to collect data from sensor nodes. Sensor nodes collect data from their surroundings, process it, and send it to a base station or sink. When the same data is sensed by more than one node, data redundancy occurs. In general, sensor nodes are densely deployed, which increases the likelihood of receiving duplicate data packets at the sink or base station. It occurs when nodes in close proximity sense and transmit the same packets. This data should not be transmitted to the base or sink node since it will waste energy or infrastructure. Aside from that, it is possible that the more essential information is left behind [13].

Quality of Service (Qos)

QoS is a measure of a sensor network's capacity to achieve application-specific requirements. QoS can be defined from two viewpoints: implementation-specific and network-specific. QoS application-specific parameters include network coverage, the optimal number of available sensors, sensor data measurement quality and accuracy, latency and delay. From a network perspective, QoS refers to the problem of effectively managing energy and bandwidth while satisfying application requirements. WSNs QoS should enable for node addition and deletion. Managing QoS parameters for sensor networks is challenging even though network topology is continually evolving and routing information is ambiguous [13].

Fault Tolerance

The fault tolerance property implies that WSNs should remain operational in the event of faulty sensors and sensor node death. When the power of sensor nodes falls below a certain threshold value in WSNs, they begin to malfunction or are unable to transfer accurate information to the next node or sink. Sensor networks are vulnerable to power outages, failures, and attacks. Faulty sensors are those that provide incorrect readings as a result of an attack or a change in environmental conditions. These sensors are functional or alive, but they provide inaccurate measurements. In such critical

situations, the network should keep operating and to provide the bare minimum of services. Sensor nodes consistently deplete power during WSNs operations. As a result, a few nodes die early and rest later in the network's lifespan. Aside from that, it is impossible to repair and maintain a sensor node after it has been implemented. Even if a few nodes die or to become faulty, the sensor network should stay attached and operational. As a result, for reliable operation, WSNs require fault-tolerant mechanisms [12].

Localization

Nodes are deployed at random in many real-world contexts, making it challenging to detect and manage them in the absence of supporting infrastructure. Learning the physical position of the deployed nodes is the problem of localization. Location discovery methods are used to help with localization. Distributed, accurate, resilient, and scalable algorithms should be used. They should also support node mobility in some instances [13].

Operating System

The sensor's operating system must be capable of providing basic memory and resource management features, but it should be less complex than a general operating system. Sensor node OS should be able to work in constrained environments, be hardware agnostic, and be application specific. It should be energy efficient and able to make decisions based on priorities. Tiny OS, Mantis, and Nano-Q plus are some of the operating systems designed for sensor nodes [12].

Architecture

Architecture includes a set of interfaces, functional components, protocols, and physical hardware, as well as a set of rules and regulations for implementing some functionality. Sensor Network architecture is a limiting feature that stymies advancement in this field. The architecture of a sensor network should be long-lasting and scalable. Because wireless sensor networks do not have a fixed set of communication protocols to

which they must adhere, it must be flexible to meet a wide range of target application scenarios even if the number of nodes is increased. Because direct coupling between processing speed and communication bit rates can result in suboptimal energy performance, the architecture must decouple the data path speed and the radio transmission rate [13].

MAC Layer Issues

Because some of the primary causes of energy waste are found at the MAC layer, such as collisions, control packet overhead, and idle listening, MAC solutions have a direct impact on energy consumption. Due to its high computing power requirements and the fact that long packets are normally not practical, the power-saving forward error control technique is difficult to implement [13].

Robustness

Each node must be built to be as robust as possible in order to support the lifetime requirements. Hundreds of nodes will have to work together for years in a typical deployment. To accomplish this, the system must be designed to withstand and adapt to individual node failure. Each node must also be designed to be as durable as possible. System modular design is a powerful tool for developing a reliable system. Each function can be fully tested in isolation before being combined into a complete application by dividing system functionality into isolated sub-pieces. To make these easier, system components should be as self-contained as possible, with narrow interfaces to avoid unexpected interactions. A wireless sensor network must be resilient to both node failure and external interference in addition to node failure. Because these networks will frequently coexist with other wireless systems, they must be able to adjust their behaviour. The use of multi-channel and spread spectrum transmitter can greatly improve the robustness of wireless links to external interference [13].

Multimedia Communication

The sensor network collects and transmits multimedia information. Multimedia data includes snapshot and streaming multimedia content, in addition to the data delivery modes typical of scalar sensor networks. Multimedia content processing and delivery are not independent, and their interaction has a significant impact on the quality of service that can be achieved. They necessitate a large amount of bandwidth for transmission [12].

Real Time Operation

Many real-time wireless sensor networks must perform in real time over extremely long periods of time. While energy harvesting has shown promise as a long-term enabler for wireless sensor networks, it also introduces new challenges for real-time processor scheduling due to fluctuating energy sources and limited energy storage capacity [12].

Homogenous versus Heterogeneous Wireless sensor networks

A heterogeneous sensor network is one with two or more types of nodes as opposed to a homogeneous sensor network, which comprises of identical nodes. In homogenous networks, the hardware complexity and battery life of every sensor node is the same. On the other hand, a heterogeneous sensor network employs two or more distinct node types, each of which has a unique battery energy and functionality. When network consist of mixer of these node, it increases the complexity of routing. Hence algorithm must be efficient enough to handle these situations.

Congestion

Sensor networks transmit a wide range of data, from simple periodic reports to unpredictably large bursts of messages triggered by external events. When the offered traffic load exceeds the available capacity of a node (in terms of buffer or link) at any point in the network, congestion occurs. Provisioning a wireless sensor network that isn't congested is a rare and difficult feat. There are two occasions at which congestion occurs:

1. In WSNs, sensor nodes can be added or removed depending on the applications, mobility, and node failure. The addition of a sensor node increases network traffic, which may result in an increase in the available load. The offered traffic load of the removed sensor node is diverted to other sensor nodes in the network when the sensor nodes are removed from the network. Due to the convergence of traffic (convergent nature) at one node, this scenario also increases the offered traffic load of sensor nodes. When sensed events cause bursts of messages, the offered traffic load increases. Congestion occurs when the offered load exceeds the available buffer or link capacities, resulting in the loss of some of the received data [27].
2. Because radio links are not shielded from each other like wires or cellular wireless links, another key symptom of congestion in wireless sensor networks is a decrease in the quality of the radio channel. The link capacity is harmed by poor and time-varying channel quality. Congestion occurs when the link capacity is less than the offered traffic load, resulting in the loss of some of the received data [27].

Congestion must be managed in mission-critical applications such as military, disaster management, and mining, as well as other applications such as habitat monitoring and the Environment Monitoring System (EMS), to avoid packet retransmission and increase the lifetime of network nodes.

Packet loss due to buffer overflow and increased delays are symptoms of congestion in both traditional wired and cellular wireless networks. Researchers have developed a combination of end-to-end rate (window) adaptation and network-layer dropping or signalling techniques over the last several years to ensure that such networks can function without collapsing due to congestion. Traditional Internet protocols (e.g., UDP and TCP) cannot be directly implemented for WSNs. UDP, for example, does not provide the level of reliability that many sensor applications require. It lacks flow and congestion control, which can result in packet loss and excessive energy use. Various drawbacks of TCP are:

1. In most event-driven applications, the overhead associated with TCP connection establishment may not be justified for data collection.
2. TCP's flow and congestion control mechanisms can discriminate against sensor nodes located far from the sink, resulting in inequitable bandwidth allocation and data collection.
3. Because TCP assumes packet loss is due to congestion and triggers rate reduction whenever packet loss is detected, TCP has a degraded throughput in wireless systems, especially in situations with a high packet loss rate.
4. End-to-end congestion control in TCP, in contrast to hop-by-hop control, has a delayed response, which means it takes longer to mitigate congestion and, as a result, causes more packet loss when congestion occurs.
5. To provide reliable data transport, TCP uses end-to-end retransmission, which uses more energy and bandwidth than hop-by-hop retransmission.

WSN congestion has a direct impact on energy efficiency and application quality of service (QoS). Congestion, for example, can cause buffer overflow, resulting in longer queuing times and more packet loss. Packet loss not only degrades application QoS and reliability, but it can also waste node energy. As a result, WSN congestion must be efficiently managed, either to avoid or mitigate it. Congestion detection, congestion notification, and congestion control are the three most common mechanisms for dealing with this issue [27].

From the above discussion, it is clear that we need an efficient algorithm for handling the congestion in WSNs. The congestion has also been explained in section 1.4. The discussion motivates us to work on this challenge of WSNs hence three objectives has been identified which are detailed in section 1.4.

1.3 Applications of WSNs

The advancement of communication and computation technology has increased the demand for distributed WSNs with a large number of sensor nodes. Each sensor node should be capable of continuously monitoring and reporting on physical activities

and communicate the information gathered with other sensor nodes. The random deployment of nodes refers to the distribution of sensor nodes throughout the desired region. There are no protocols that have been specified for such types of random deployment. WSNs are also used for detection and alerting purposes. When a malfunction occurs in any industry, it can send a signal to the control room [14] [15]. WSNs can be used in a wide range of applications, as shown below, depending on the specific requirements, and these applications are as follows:

1.3.1 Environmental/Earth monitoring

WSNs play an important role in various economic surveillance applications, which are typically divided into two categories: outdoor surveillance and indoor surveillance. Indoor applications primarily include building monitoring applications. Outdoor monitoring encompasses a wide range of applications, including habitat monitoring, traffic control, hazardous detection, and weather forecasting, among others. Wireless networks are preferred over wired networks for more reliable and systematic monitoring. WSNs can also be used to detect forest fires. Sensing nodes are scattered throughout the forest, and when a re begins, the sensor detects it. Sensor nodes transmit sensed data to the control room for further action. Sensing devices can assist in detecting the affected fire area and allowing fire fighters to control the re of forest. Early detection of activities is extremely advantageous for a successful action. In addition, sensor nodes are deployed to detect land sliding in hilly areas. Sensor nodes allow the system to detect minor changes in soil movement and small changes in various parameters that may occur during a landslide. It may be possible to predict the occurrence of landslides in the future using data collected from landslides. WSN nodes are used to monitor the quality of water in the ocean, lakes, rivers, and dams. The sensor nodes allow for a more accurate map of water status. Water monitoring in remote areas can be accomplished by deploying sensors using helicopters with no human intervention [16].

1.3.2 Healthcare applications

WSNs are very beneficial for healthcare-oriented applications and have proven their effectiveness in monitoring patients with epilepsy, heart problems, heart attack or stroke patients, elderly patients, autism patients, and Parkinson's diseases. Healthcare applications are not stand-alone systems; instead, they are integrated with complex health and rescue systems. While preventive health care has encouraged lower health-care spending and mortality rates, studies show that patients are uncomfortable, inconvenient, complicated, and interfere with their personal daily lives. The sensor network alerts doctors or nurses about the health of their patients and the need for their involvement. There are various sensors available that can detect various functional activities of the human body, such as hemoglobin, heart rate, blood pressure, electrocardiogram (ECG), electromyogram (EMG), blood flow, respiration, and oxygen level in the blood, among others. Another application of WSNs is to monitor gas, water, and oil pipelines. Managing pipelines is a difficult task. The pipelines' long length, high cost, high risk, and a variety of challenging conditions necessitate continuous and unobtrusive monitoring. Pipeline leakage can occur as a result of extreme deformations caused by collisions, land sliding, earthquakes, corrosion, wear and tear, material flaws, and intentional pipeline damage. Sensor nodes are used to detect such events in order to overcome these issues [17].

1.3.3 Industrial applications

Using WSNs, connected computers create an ecosystem that connects various devices into smarter, broader applications. It is best to provide the data of these machines to the end of the machine to answer any questions that may arise during the course of a project. These days, connected machines are built on two strategies: one inside the factory, where information about a machine's efficiency collected by smart sensors should be supplied to the key participants in the connected machine ecosystem. The other approach is much more macro in nature. Sensors that assess a number of factors useful in manufacturing applications, including visual strength or noise strength, voltage, current, strain, heat, speed, positioning, liquid stream frequency, and so on, are easily and cheaply fabricated. The sensed information is then transferred to a control room installed with a

microcontroller, where it can be stored and analyzed by humans before decisions are made, or to some automation software that collects data and takes some action/decision. As a result, WSNs are critical for Industrial Automation (IA). Sensor networks serve as a vital link between the control system and the physical world. Various cutting-edge software and hardware are being introduced for controlling systems, which creates opportunities for automation in factories, sugar plants, rice plants, and a variety of other processing plants [18].

1.3.4 Military Applications

New emerging and advanced technologies, such as WSNs, serve as a backbone for the military by rapidly transmitting vital information from the area of interest, and the networks are incorporated with defense applications due to their sensing, reliability, fault-tolerance, and scalability capabilities. The adversaries can destroy the deployed sensor networks, but these nodes are low-cost devices that suffer little damage. WSNs are extremely useful for military applications. In the military, there is always the possibility of being attacked by enemies. As a result, using low-cost and small sensor nodes helps to reduce loss. By deploying sensor nodes in available scenarios, enemies' suspicious activities, such as men on foot, weapon amounts, and weapon types, can be detected, identified, and classified in advance. The networks provide accurate real-time images of battle and improved situational awareness. Sensor nodes in the military perform sensing, monitoring, commanding, targeting, and other tasks. To report the current status of suspicious activities, small nodes are distributed with each troop, vehicle, critical missiles, and equipment. After collecting the reports, the information is forwarded to the base station, which is stationed in a safe area, and to the troop commander, who may then report to higher commanding authorities. Sensor nodes with a close inspection on opposing forces activity can identify critical locations and potential adversary entry routes. Sensor nodes are also mounted on intelligent missiles or shells [17] [18].

1.3.5 Flora and fauna Application

Every country requires both flora and fauna domains. Greenhouse surveillance, crop management, and livestock production are the three primary subcategories of flora and fauna uses of WSNs.

1.3.5.1 Greenhouse surveillance

Greenhouses play an important role in the agricultural sector. Many crops may be grown in them to produce sustainable food, and if particular conditions are met within the greenhouse, climatic crops can be harvested all year. As a consequence, WSNs may be utilized to improve greenhouse performance through monitoring and control. It provides a relevant greenhouse-specific system featuring energy management and interior climate control capabilities. The system's sensor nodes monitor critical greenhouse factors such as interior brightness, temperature, and relative humidity. [19] [20].

1.3.5.2 Crop Monitoring

[21] [22] constructs a computerized fertilizer applicator system comprised of three modules: input, decision support, and output. The input module may communicate real-time sensor data to the Decision Support System (DSS) module through Bluetooth technology. This technology seeks to automate water supply management in cultivated fields. The technology uses previous data as well as changes in climatic variables when calculating the quantity of water needed for irrigation.

1.3.5.3 Livestock Monitoring

[23] Describes a system based on RFID and cutting-edge technology that may be utilized in current agricultural large-scale management systems in WSNs. The suggested system is capable of identifying and tracking all farm animals as well as evaluating their environment and the health. [24] A low-cost, low-power collar is proposed to help cattle husbandry. The proposed device employs a solar power relay router and two cleverly positioned antennas to optimize collar radio coverage.

1.4 Congestion

1.4.1 Congestion in Wired-Networks

Network congestion arises when the proposed traffic burden surpasses the existing aptitude at any point in a network. Essentially, the main issue in wired networks is the effective and fair allocation of resources among competing nodes. The bandwidth of the links and the buffers on the routers where the packets are forwarded are the resources. The packets are queued in the routers, waiting for their turn to be transmitted. When there are too many packets waiting to use the same link, the router's queue overflows and packets must be dropped. When such drops occur frequently, the network becomes congested. Congestion in wired networks can be detected in the following ways:

- **Monitoring Queue length**

  If the routers' buffers begin to fill up and the broadcast rate is lesser than the packet onset rate, this indicates that congestion is imminent.

- **Monitoring packet losses**

  If routers start dropping packets due to overflow, this is an indication of congestion.

Congestion control in wired networks is typically accomplished through the use of end-to-end and network-layer mechanisms working in tandem. The computation capability, memory space, and energy supply of WSNs are all limited. Congestion is an event that increases energy waste in these networks and, in extreme cases, causes network collapse.

1.4.2 Congestion in WSNs

Congestion is the most troublesome issue in WSNs, as it has a harmful influence on the linkage's Quality of service (QoS) parameters. In the case of an incident, the volume of data generated by the sensor node exceeds the capacity of the sensor network, allowing the network to become congested. Congestion can be caused by a variety of

factors such as buffer overflow, concurrent transmissions, packet collisions, and the many-to-one nature of a sensor network. Congestion occurs primarily at two levels of the sensor network: node-level congestion and link-level congestion [27] [28].

Node-level Congestion

Congestion at the node level is common in both traditional and WSNs. Buffer overflow at a sink or intermediate node in the network is the most common cause of node-level congestion. Congestion at the node level causes packet loss and increased queuing delay. Congestion at the node level causes packet loss and energy waste in networks. As a result, this sort of congestion directly impacts link utilization and lifespan [27] [28].

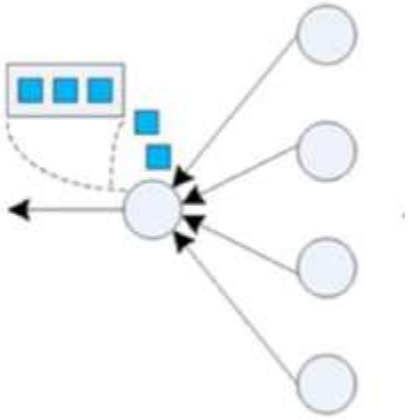

Figure 1.6 Node level congestion in WSNs

Link-level Congestion

Link-level congestion, on the other hand, occurs when multiple sensor nodes inside variety of one additional effort to communicate at the same time. This type of congestion reduces network link utilization and throughput while also increasing network delay and energy waste [27] [28].

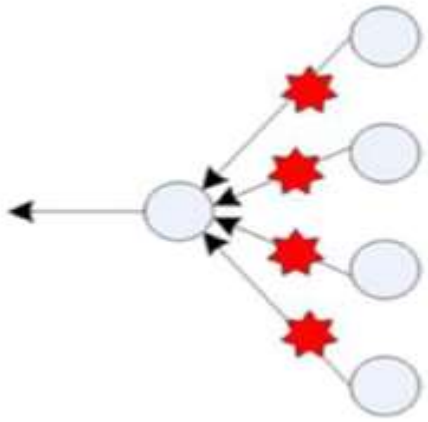

Figure 1.7 Link-level congestion in WSNs

Congestion should first be detected in sensor nodes, which is acknowledged as the congestion uncovering stage. Second, as congestion transpires, upstream sensor nodes should be reported, a process known as the notification phase. Ultimately, during the rate modification or congestion moderation phase, congestion should be minimized and an acceptable data rate selected. As a consequence, the congestion control process is separated into three stages: monitoring, reporting, and treatment [29] [30].

Congestion is WSNs can be classified into two major categories regarding how packets are misplaced and where the network packets are lost.

How the packets are lost

- **Packets lost in the medium**

  In a certain location, a number of nodes inside variety of each other attempt to broadcast concurrently, resulting in data loss due to intervention and, as a result, a reduction in network presentation for the nodes in that region [29].

- **Packets lost due to buffer overflow**

  When the queue or buffer of a particular node holding the packets to be transferred surpluses, the packets start drooping. It means node is receiving the

packets at higher rate than it can transmit. This is the traditional definition of congestion in networks [29].

Where the packets are lost in the networks

- **Near source**

  The loss of packets close to the source causes source congestion. Thickly installed sensors producing data during peak hours will generate a hotspot extremely near to the source. In this situation, a quick time scale and localized system capable of sending back pressure messages from the site of congestion to the sources would be beneficial for instant traffic control [29].

- **Near Sink**

  The packets lost near the sink leads to sink congestion. Congestion near the sink is usually caused by sensor nodes deployed sparsely and generating data at lower rate. In this situation due to transient network, sensor nodes may not be using backpressure messages. Also using the multiple sinks that can be deployed at different places at the network may equilibrium the traffic between different sinks to alleviate congestion [29].

- **Forwarder Congestion**

  A sensor node may be transmitting data from more than one flow and it is likely that that flows intersect with each other causing hotspots to be created at the point of intersection. For example, in tree network topology almost every node may be suffering from forwarder congestion In WSNs, where the network is extremely dynamic, predicting the network crossing point is challenging, making forwarder congestion difficult to control. In this situation, a combination of fast time scale localization and closed loop rate regulation can be used to alleviate congestion [29].

The entire scenario is depicted in the figure 1.8

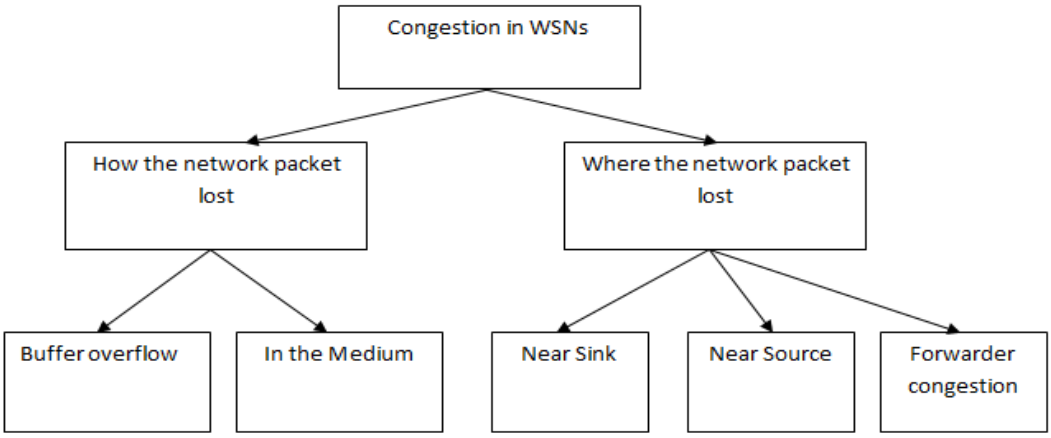

Figure 1.8 Congestion process in WSNs

1.4.2.1 Congestion detection

Congestion identification in WSNs mentions to the technique of recognizing and discovery congestion. In WSNs, the following metrics are used to detect congestion

- **Packet service time**

  It refers to difference of time taken by the packet to reach at medium admittance layer and time at which it is transmit. This parameter is the same as the one-hop node delay [31] [32].

- **Buffer occupancy**

  This congestion detection parameter is also known as queue length. If an incoming packet cannot be transmitted immediately, it is stored in a buffer by each device in the WSN. The algorithms that use this parameter for congestion detection typically use a constant buffer threshold. Congestion is detected and some notification

is raised when the queue length or buffer occupancy exceeds that constant threshold value. [31] [32].

- **Channel load**

  Another parameter for detecting congestion is channel load or busyness. In a wireless transmission medium, packet load is computed, and congestion is detected when packet transmission time exceeds a predefined threshold time. When the channel is congested, the likelihood of packet collision increases. When there is an increase in packet collision, some packets are removed from the buffer, resulting in a decrease in buffer occupancy.

Because a decrease in buffer occupancy can sometimes be misinterpreted as the absence of congestion, a hybrid approach that includes both buffer occupancy and channel load parameters must be used to detect congestion in this case [31] [32].

- **Delay**

It tells us how long it takes from the time a packet is engendered on the sender side until it is successfully received at the receiver or the next intermediate node. Delay can be divided into two types: one hop delay and end-to-end delay. One hop delay is the time it takes at the Medium access layer (MAC) layer to extract the header information from the segment. When segment information is extracted for processing at layer 2 and then passed on to layer 3 for routing information extraction, the total time taken for the same process is known as one hop delay. At the MAC layer, it addresses collision resolution, packet waiting time, and packet transmission time [31] [32].

1.4.2.2 Congestion Notification

When congestion is sensed, the congestion information must be propagated to the relevant sensor node so that appropriate action can be taken. This information can be as simple as a congestion bit or as complex as a control message. The information

transmitted as a congestion notification bit in the packet header is referred to as explicit congestion, while the information transmitted as a separate control message is referred to as implicit congestion.

- **Explicitly congestion notification**

  When congestion is detected in this technique, the congested node sends control packets to the source nodes informing them of the congestion state. Sending the congestion state in the form of control packets in an already congested network adds to the network's load. As a result, this method is not widely used and is only used by a few congestion control protocols. This is the primary disadvantage of explicitly sending the congestion notification [31] [32].

- **Implicitly congestion notification**

  On the other hand, the implicit method is a technique that does not use control packets to inform source nodes of congestion and places additional load on already congested nodes and networks. This congestion notification mechanism alerts other sensor nodes of congestion. A wide variety of congestion control approaches employ an implicit congestion notification mechanism to convey congestion information [31] [32].

1.4.2.3 Congestion Control

Congestion control is a critical component of WSNs when traffic exceeds the agreed-upon capacity of the underlying network. As a result, special thought is required to develop a technique for controlling congestion. Congestion control is regarded as one of the most important constraints in WSNs that are deployed over a large area with spatially distributed nodes. Congestion can occur in WSNs for a variety of reasons, including buffer overflow, packet collision, transmission channel contention, rate of transmission, type of data transmission scheme, and where the transmission channel is dynamic [30].

Various congestion control protocols are basically categorized into following parts:

- **Traffic control based protocol**

  When network traffic demand exceeds the capacity of network resources, traffic control mechanisms control the data packets that are injected into the network. This mechanism can also be thought of as lowering the quantity of traffic vaccinated into the network in order to control congestion. There are two methods for managing traffic control: window-based management and rate-based management. When utilizing a window-based method, a sender tests for available network bandwidth by progressively raising the congestion window; if congestion is discovered, the protocol significantly decreases the congestion window. In order to avoid network collapse, window size must be reduced in response to congestion. The available network bandwidth is explicitly calculated by rate-based algorithms.

  The responsibility of congestion mitigation lies exclusively on a single node, the source node, which is one advantage of traffic management protocols. Traffic control methods, on the other hand, are inefficient for event-based networks, in which sensor nodes are activated by an event. This is because reductions in network traffic jeopardize the network's mission because all packets contain priceless evidence about the occasion. Furthermore, due to their slowness, traffic control protocols cannot be used effectively during transient congestion situations.

- **Resource Control techniques**

  The resource control protocol was developed to address the shortcomings of the traffic control mechanism. When there is congestion in a resource control mechanism, data packets follow the least congested path available in order to reach the base station. The chief benefit of this practice is that packets have a

higher chance of reaching the sink node. The best protocol for congestion control is determined solely by the application. The resource control protocol is better suited for applications in which all data must be transferred to the base station, provided the network is dense enough to provide alternate paths [30].

- **Priority-based congestion control mechanism**

  Some protocols use prioritized MAC methods to deliver prioritized channel access to crowded nodes. In other statements, congestion is managed in this strategy by taking into account different expectations in congested areas [30].

- **Queue assisted congestion control mechanism**

  The queue length of the nodes is used to combat congestion. The queue-assisted protocols are primarily concerned with node queue length and employ a simple rate adjustment technique. [30].

Congestion control protocols can also be divided according to the following parameters like

- Direction of the traffic like upstream or downstream.
- Transfer of the packet i.e. either hop by hop or end to end.
- Malicious activities of the nodes.
- Energy efficiency etc.

1.5 Aim and objective of the Thesis

The overall motive of the presented thesis an adaptive congestion control algorithm for fair and efficient data transmission in WSNs is to accomplish three objectives which are listed below:

1. *To design a path evaluation algorithm for congestion control based on traffic classification using an intelligent technique.*

As the network traffic injected into the network exceeds the network capacity, congestion occurs. In our first objective we will try to control congestion by routing the traffic to different network path after classifying it into sensitive and non-sensitive traffic. We will use the best route for transmitting the packets belonging to sensitive traffic because their QoS requirements are quite stringent. Modified ant colony optimization is used for finding the optimized path among different paths available from source to destination.

2. *To design an algorithm that adaptively assign fair and efficient transmission rate to each node during network congestion.*

In addition to controlling the congestion we pursue to adaptively allocate a fair and efficient transmission rate to the nodes of the network in the event of congestion. It is highly desirable to apportion the available bandwidth to all the available paths to provide fairness among the different nodes of the network. It should not be like that a single node is holding the maximum part of bandwidth and other nodes are just having a lesser part of it. So we will design an algorithm that will adaptively allocate reasonable and well-organized transmission rate to all the nodes in the network.

3. *To design an algorithm that prevents congestion under the presence of adversaries in the network*

Malicious nodes in the network may also aggravate congestion by diffusing packets into the network or retransmitting them several times. In this objective, we will device an algorithm for mitigating the effect of flooding attack in the network. The main objective of flooding attack is flood the intermediate node with false route request packet thereby increasing the congestion in the network.

1.6 Main contribution and results in brief

In this thesis, we have proposed 3 research objectives. In the first objective, we have designed a novel algorithm for controlling the congestion by routing the traffic through multiple paths using modified Ant Colony Optimization algorithm. The basic goal of first objective is to find an optimal path from list of viable paths for routing the sensitive traffic in the event of congestion occurrence. Initially, the algorithm check the status of congestion in the network and if at any node congestion occurs, algorithm, will find he optimal path from set of available paths in the network using path preference probability. Simulation results show that the proposed algorithm outperform the existing state of art algorithms. We have conducted the extensive simulations by varying the number of nodes and simulation time.

In the second objective, we have proposed a novel algorithm which will adaptively provide a fair and efficient rate to each flow in the network by allocating approximately fair bandwidth. In the proposed work, we have used available bandwidth resources, and active flows belonging to a particular node to estimate the fair rate that can be efficiently assigned to each node in the network. Simulation results show that the proposed algorithm outperform the existing state of art algorithms. We have conducted the extensive simulations by varying the number of nodes and offered traffic load.

In the third objective, we have designed an algorithm for controlling the congestion in WSNs under the network attack. We worked on the flooding attack. In this attack, the malicious node tries to flood the intermediate nodes with fake packet requests there by exhausting the network resources and make the overall network congested. The proposed algorithm will detect and isolate the malicious node. Simulation results show that the proposed algorithm outperform the existing state of art algorithms. We have conducted the extensive simulations by varying the number of node.

CHAPTER 2

Review of Literature

2.1 Introduction

This chapter will provide detailed literature review of various congestion control protocols based on various techniques for detecting and controlling congestion. Later in the survey, the various techniques for controlling congestion by conveying reasonable and well-organized rate control to the nodes in WSNs are investigated. Finally, it describes different approaches and methodologies for noticing and stopping network attacks in WSNs.

Network congestion is one of the most essential issues to consider in QoS. In particular applications, congestion must also be regulated or avoided. Congestion must be noticed first in both cases. Congestion can be spotted using constraints such as transmission rate, source rate, channel load, packet collision, interference caused by simultaneous transmission, and so on. The incoming and outgoing data rates are used to calculate buffer occupancy. Exponentially weighted moving averages are also used to calculate buffer occupancy (EWMAs). Congestion is detected because when average buffer occupancy exceeds the threshold value. In addition to buffer busyness, network strain is exploited for congestion detection. The threshold factor is considered grounded on the demanding time of the channel.

2.2 Techniques and Procedures proposed for controlling congestion and for evaluating the optimal path

Many studies have been conducted to develop various methodologies for congestion control in WSNs. All of the techniques and procedures for controlling traffic have advantages and disadvantages. The various methodologies found in the literature are examined further below.

PSFQ was proposed by Wan et al. [33] to detect and governor congestion by modifying its pump/fetch ratio. Packets from a source node are distributed at a slow rate which is called pump slowly process. In the event of data loss at a specific node, the node tends to recover the same data segments by quick pumping. When a packet is lost, a negative ACK is issued to the sink/source node. Directed Diffusion and PSFQ can both prevent congestion, but procedures need some special features to be incorporated like load of the channel and channel state sampling as a technical features. The controller at the sink must be knowledgeable about the specialized settings. The efficiency of these strategies is determined by individual events rather than by a general rule.

The TADR proposed by Ren et al. [34] a hybrid scalar potential field made up of depth and queue length. The former acts as the backbone, sending packets to the sink through the shortest available path, while the latter monitors TADR traffic. In the case of congestion, requests are forwarded to other pathways composed of unoccupied or under-utilized nodes. To avoid hotspots, a bypassing hotspot rule is introduced. The dilemma is that it is difficult to comprehend the temporal variation potential fields that lead to measured result rather than ideal values.

Hull et al. [35] proposed a method for governing congestion that used three methods: (a) control of flow with hop-by-hop method, (b) limiting the packet rate, and (c) MAC prioritization. These techniques are known as fusion which halts the packet transmission that is designed to drop. The packet drop will be decided by the hop-by-hop flow control whenever there is a challenging space left in buffer of downstream node. Fairness in the network will be ensured by the limiting the packet transmission rate especially for the nodes which are away from sink node. Prioritized MAC is in charge of ensuring prioritized channel access for congested nodes. The suggested method does not trust on topological evidence, instead concentrating on single-sink and network made in spanning tree process. The difficult issues, such as interference between seemingly disjoint sets of nodes and the intrinsically lossy nature of wireless channels, are spoken suitably.

Wan et al. [36] proposed a novel algorithm for controlling congestion in WSNs. CODA combines the use of load of a channel and buffer possession to detect the congestion at in-between nodes. Open-loop as well as hop-by-hop backpressure notification is used to detect the congestion. Once congestion is detected, it notifies upstream nodes via backpressure. When ACK packets are lost, the source slows its sending pace. The fundamental downside of the CODA technique is that it needs control signals like ACK and back pressure, which demand extra energy from sensor nodes. The proposed procedure does not take fairness into consideration and uses AIMD approach which normally leads to loss of packets.

Tao and Yu [37] introduced Enhanced congestion detection and avoidance (ECODA). In multimedia application, large volume of data needs to be transmitted from several sensor nodes. These applications have special QOA requirements and dealing the congestion in these applications is more critical. To address these issues authors have proposed a novel protocol called ECODA. The method differs from the typical single buffer threshold method in that it can differentiate congestion levels and deal with them accordingly. ECODA's flexible queue scheduler can dynamically select the next packet to launch. Furthermore, when congestion occurs, it employs a novel way of filtering packets based on channel loading and packet priority. Transient congestion and chronic congestion are distinguished and treated differently. The hop-by-hop unspoken backpressure method is utilized for transitory congestion. The biggest downside of this strategy is that it is difficult to retrieve lost packets.

In IACC Kafi et al. [38] had proposed interference aware congestion control protocol for dense networks, as in a dense network using CSMA, nodes are fighting for shared medium and link suffers from capacity variation due to interference of data transmission by nodes. This study addresses these two difficulties. The suggested protocol maximizes the utilization of connection capacity for each node. IACC operates in two stages: It first determines the link capacity between each node and its parent. The predicted link capacity is transmitted to the base station. The second step is to construct

scheduling to ensure fairness while preventing congestion and controlling node interference. The predicted capacity assists the IACC protocol in establishing a schedule that allocates suitable time and rate for transmitting to each node. The schedules are calculated by the base station. After estimating, the arrangement procedure is divided into two steps: First, for each node u, the goal is to allocate its available bandwidth in such a way that all source nodes falling under node u's responsibility can send data equally and congestion is avoided by determining the sending rate for node u.

Jaiswal and yadav [39] had proposed FBACC - Fuzzy based adaptive congestion control protocol which controls the congestion for random traffic pattern using fuzzy logics. The fundamental idea behind FBACC to decrease the packet loose rate at in between neighbor nodes while increasing the amount of packets sent to sink using a Fuzzy Logic System. As a result, FBACC boosts average packet delivery, which is an important element for QOS over busy traffic. For congestion identification and mitigation, FBACC employs three parameters. These three factors are the traffic rate, the number of participants, and the buffer occupancy. FBACC employs the three factors listed above as inputs to calculate a single output reflecting congestion level, which is then compared to set thresholds for congestion detection. When congestion is observed, the congested node explicitly instructs neighbors to reduce transmission rate, therefore controlling packet drop. FBACC employs FIS rules to generate a single output variable E for congestion level. When congestion arises, it uses a feedback system to adjust the traffic rate and contributors in order to reduce congestion.

Sergiou et al. [40] had proposed Hierarchical Tree Alternative Path (HTAP) Algorithm for Congestion Control in WSNs. The algorithm faces the congestion by using alternative paths for avoidance of congested areas and nodes in WSNs. The authors had also added two features in HTAP algorithm in the paper. These features overcome the problem of delay in reaching the packets from source to sink when following the alternative path for packet sending in case of congestion. The first feature which is added is topology control scheme based on local minimum spanning tree algorithm where each

node builds its local minimum spanning tree and contain only those nodes in the neighbor list which are one hop away or near to the sink. The second feature which is added to HTAP algorithm is ability to recognize the deadlocks. If a node understands it will be unable to transfer a packet to a sink, it alerts nodes who are accessing it through negative response. In addition, the HTAP algorithm creates a new adaptive methodology for analyzing network congestion. This adaptive technique uses buffer occupancy as a first indicator of congestion occurring, and then it uses the out/in data rate ratio to initiate the building of an alternative way. As a result, the HTAP addressed in this study comprises of four schemes: topology control, hierarchical route generation, alternate path creation, and dead lock node handling.

Aghdam et al. [41] had proposed a protocol for controlling congestion in wireless multimedia sensor networks by considering the traffic pattern and video packets priority. WCCP is a two phase protocol. In the first phase, it uses source congestion avoidance protocol (SCAP), to alter the sending rate and dispersal of packets leaving the source. Initially SCAP uses the group of pictures calculation method to calculate the congestion and adjust the transfer rate and packet distribution leaving the source by seeing the WMSN traffic prototypical. In the second phase, it uses receiver congestion control protocol (RCCP), to notice congestion and notify the source nodes regarding congestion. The RCCP protocol detect the congestion by using the wireless multimedia queuing model and computes the permissible input rate of a node and send a response to SCAP protocol regarding congestion. If SCAP protocol receives a congestion notification it adjusts the output rate of the source node by ignoring the less valuable packets and distribution of packets leaving the source node is done according to the feedback received from the RCCP protocol. Simulations have been proposed in the paper and it has confirmed that it shows better performance in terms of video quality and energy consumption of sensor nodes.

Antoniou et al. [42] had proposed flock-cc protocol for congestion regulator in WSNs. The resource limitation and unpredictable behavior of WSNs demands a

decentralized, robust, self-adaptive and scalable congestion control protocol. Solutions inspired from nature are more effective, powerful and simple as self adaptation is one of the major strength of bio system. Flock-CC protocol used the flocking behaviors of birds and swarm intelligence to make a simple, robust, scalable, self-adaptive congestion protocol in WSNs. The fundamental concept is to direct the packets so that they form a flock and migrate towards a global attractor, which is a sink in WSNs. While moving the congestion is avoided by avoiding the dead nodes and congested nodes. If a packet is one hop away from another packet then it is said to have repulsive force between them and area covered is called zone of repulsion and if a packet is two hops away from a packet then attractive force is said to have between them and area covered is called zone of attraction. While interacting with each other in a flock, a packet will send the packet to only that node which comes under the zone of attraction to avoid collision. The proposed method is modest to device at distinct node level and requires nominal information altercation. It also includes self adaptive behavior to allow the packets to pick the random routes and choose the node with less congestion level.

Razaee et al. [43] had proposed a novel algorithm for controlling congestion in WSNs. The paper focused on congestion management in WSNs for health care applications. The proposed protocol considers two type of traffic. Sensitive traffic is used to transfer high prominence data and non-sensitive traffic is used to transfer the low importance data. The protocol uses multipath routing for sending sensitive and non-sensitive data separately. While sending the sensed information about the patient state, congestion might occur in the node to buffer overflow. AQM has been used in the proposed protocol for congestion mitigation and preventing the loss of packets due to buffer overflow. Basically the proposed protocol works in 4 phases: Request broadcasting, event broadcasting, route founding and data accelerating plus rate adjustment. In request dissemination phase, information about the patient's vital sign is broadcasted to all the network nodes by the sink node. Sink distribute the data among network nodes with different priorities using MLAF (multimedia location aided flooding) algorithm. This is required to distinguish the sensitive and non-sensitive traffic. In route

establishment phase, a confirmation packet is sent to the source node. After this, source node creates a final routing path based on the sensed information. Two routing path/tables are created for sensitive and non-sensitive information, which makes the protocol multipath. In data forwarding phase, sensed data is forwarded to sink from sensitive and non-sensitive routing paths based on priority of data. In this phase congestion is mitigated with the help of rate adjustment technique.

An efficient and QOS based multicast routing protocol is projected by banimelhem and khasawneh [44] for congestion avoidance in grid sensor network. The entire sensor network is divided into grids and each grid has a master node. The responsibility of master node is to collect the information for other sensors in the grid and from additional master nodes in the neighboring networks. For each master node, numerous slanting pathways are also established to connect the master node to the sink through other master nodes and entries for the path is made in the routing table of that node. Each master node available in the grid evaluates its buffer occupancy. The protocol distributes the traffic to other available paths in the event of congestion detection. The traffic is diverted to the new available path. However the author has also proposed the use of secondary master node within the grid in addition to primary master node in the event of congestion detection. The node with higher energy elected as a secondary master node. During congestion secondary node is responsible for collecting the data from the nodes as well as neighboring master nodes. The chief benefit of the protocol is energy efficiency and main disadvantage is the overhead incurred during the selection of master node and grid.

In this paper, Heikalabad et al. [45] provided a dynamically predictive congestion control protocol for highly dense network. Due to convergent nature of upcoming traffic in WSNs, application produces high flow rate near the sink of a network. This situation leads to network congestion and which further leads to packet loss and delay in the network. The proposed protocol in the paper was intended at forecasting the congestion and dissemination the traffic on the network equitably and animatedly. The protocol

consist of three components: backward and forward node selection, prognostic congestion discovery and vigorous priority based rate alteration.

In this paper sergiou et al. [46] had proposed self-motivated substitute path algorithm to avoid and control congestion in WSNs. DALPas is considered as a vibrant, dispersed and light weight which is intended to weaken the congestion occurrence probability. The algorithm works in 4 parts: setup stage, topology control protocol, soft-stage scheme and hard-stage scheme. In setup phase network is established and nodes discover each other's location by building the neighboring table. In soft-stage scheme, algorithm attempts to reduce the possibilities of congestion occurrence in the network. In this scheme, each node serves just one flow at a time. In this way algorithm is able to reduce the incidence of buffer founded congestion in the network. If soft-stage scheme does not work, then hard-stage scheme is employed, in which, data flows are enforced to alter their route in instruction to get the receiving node uncongested.

Misra et al [47] had proposed learning automata based congestion avoidance scheme (LACAS) for healthcare WSNs. Congestion in WSNs leads to packet loss, delay in arrival of data to destination, high energy consumption of the nodes and it is highly desirable to avoid congestion from occurrence. In this paper authors has proposed a novel algorithm to resolve the problem of congestion in the nodes of healthcare WSNs. The major aim of the proposed approach is to adapt the packet arrival rate of the child nodes to the paternal node's packet service rate, therefore preventing congestion. Another important feature of the approach is that it learns intelligently from the past and improves the performance of the network as the time progress by choosing better data rate in the future. The primary advantage of the approach is that automata stationed in the sensor node of the network interact with the atmosphere to select a nearby optimum action to control congestion.

Narawade and kolekar [48] had proposed a nature inspired metaheuristic and novel approach for congestion control management in WSNs. Several sensor nodes in WSNs collect the data rigorously and send it to sink. High data load on the network and

limited network resources lead to congestion. It is grounded on parasitism conduct of the cuckoo species. The main objective behind the proposal is to make equality between the packet arrival rate and packet service rate at child node as well as parent node respectively. Congestion is dodged based on packet fall likelihood, and in the occasion of congestion, the optimization algorithm returns a new share rate for the child nodes that is lower than the average rate of the previous period. The simulation shows that suggested procedure is more efficient than existing rate adjustment scheme.

Yadav et al. [49] proposed the ECA-HA congestion control algorithm for wireless sensor networks, which can be used in a variety of applications such as health care, agriculture, and transportation monitoring. An ant colony is used in the proposed algorithm, backward ant and forward ant algorithms are used in this approach. To find more than one path for data, ant algorithms are used. Moreover, Huffman coding was applied on top of the sample space provided by the ant colony approach to select one of the less congested paths as the optimal data routing path. The proposed algorithm's extensive performance results outperform state-of-the-art algorithms in terms of various parametric situations. Thus, in the current round, the proposed algorithm forwards packets using the best data routing path, which reduces hop-by-hop delay, improves packet delivery ratio, or reduces node death percentage, all of which improve network throughput and lifetime, respectively. The proposed algorithm's only flaw is that the network grows at a rapid rate, but this approach is suitable for small search spaces. As a result, authors will be looking for a large search space in the future. In the future, in the context of the Internet of Things, authors will incorporate additional parameters such as dynamic traffic load or link breakage to find the most optimal data routing path using a reinforcement learning approach.

Sangeetha et al. [50] Worked on the main Quality of service factors in wireless sensor networks, such as energy consumption reduction and congestion control, in this paper. Initially, the power used was reduced by repeatedly forming periodic clusters in order to find new CH that could withstand the stress. Congestion is a major factor that

must be taken into account in order to reduce the amount of power consumed in WSN. When there is congestion, the best path is chosen using the heuristic LRTA search algorithm, which is mission critical for locating the goal. To reach the sink node, fuzzy logic rules are used to make a priority decision that will be treated with a search algorithm. This reduces energy consumption by preventing congestion in the WSN. In terms of energy loss rate and packet loss rate, simulation results show that the proposed work outperforms CCFL. As a result, the proposed framework HPCCF reduces data loss and the time required to reach the sink node, reducing end-to-end delay. The futuristic step in this proposal is to compare other goal-based search algorithms to determine the best path to the sink node and integrate them into congestion control.

In this paper, Ding et al. [51] proposed an optimising routing algorithm based on congestion control. Both energy efficiency and congestion control are considered in the proposed algorithm. The application of node location and the reasonable allocation of traffic flow in the network are used to optimise network transmission performance. The link gradient function, in most cases, provides the basic routing backbone for directing packets to the sink. Furthermore, the traffic radius function can distribute excessive packets over multiple paths that include idle and underloaded nodes. These two functions are combined to form flow rate, which is used to make dynamic routing decisions. Under the same criteria, the proposed algorithm is compared to MintRoute and TADR. Simulation results show that the proposed algorithm can effectively relieve traffic congestion while maintaining high energy efficiency, albeit with a relatively high delay under heavy traffic loads.

In this paper, Raman et al. [52] propose a fast congestion control (FCC) scheme for WSNs in a closed experimental setup using a hybrid optimization algorithm. First, authors use a multi-input time on task optimization algorithm to find the best intermediate node. This algorithm uses the waiting delay, received signal strength, and mobility as multi-inputs to improve network lifetime and reduce congestion. Then, using a modified gravitational search algorithm, a path between sources and sink nodes is

computed, providing a better routing path. In comparison to the existing CCOR scheme, experimental results show that the FCC scheme can effectively control congestion while minimising data loss, energy consumption, and maximising hop counts and network lifetime. Because the NS2 has limited scalability, the work can be moved to a distributed environment by using other real-time distributed network protocols instead.

Rezaee et al. [53] propose a congestion control protocol for WSNs with medical applications in this paper. The best case scenario for the proposed protocol is that sensor nodes display patients who are unable to move and are bedridden in a special ward of a hospital or private clinic, as well as those who have suffered a heart attack or have died from brain death. This protocol proposes a new active queue management method that combines the RED method and the FuzzyPID approach to calculate packet loss probability. In addition, proposed algorithm estimates each node's output transmission rate and then assigns a suitable transmission rate to each node. Overall, the simulation results suggest that the proposed protocol could outperform CCF, PCCP, and OCMP protocols in terms of parameters like packet loss ratio and end-to-end delays. In the future, the current study will need to be improved. This protocol could be used to monitor mobile patients in a hospital or in cities in the future.

WSNs can use an Optimized QoS based Multipath Routing Protocol (OQoS-MRP) proposed by Onthachi et al. [54] to establish near-optimal routes for data transmission under multi-constrained QoS. To find the next hop neighbour, the SingleSink-AllDestination algorithm is used. The simulation results are assessed and compared to the existing EE-LEACH and MRBCH protocols. The performance of these protocols is assessed in terms of QoS. The simulation results showed that the proposed protocol improves communication reliability with minimal delay while consuming less energy and extending the network's lifetime. Furthermore, OQoS-MRP improves load balancing by dynamically selecting an alternate path for data transmission from a subset of best-case paths. As a result, the optimality principle is valid.

Yogarajan et al. [55] proposed an improved Clustering-based data collection using Ant Lion Optimization in this paper. The clustering algorithm's performance is assessed and compared to that of existing clustering protocols. In comparison to other clustering algorithms, the proposed clustering algorithm increases the sensor network's lifetime by at least 13%.The sensor node's residual energy is high due to the proposed heuristic Antlion optimization algorithm's optimal CH selection and clustering. Because of the proposed algorithm's optimal use of a node's available energy in the battery, the network's lifetime is extended, a greater number of packets are received at the base station, improving the network's throughput, and the number of individual nodes is reduced after clustering. Large-scale homogeneous wireless sensor networks are well-suited to the proposed algorithm. When the Discrete Ant Lion Optimization algorithm is applied to the selected cluster heads, an optimal tour for collecting all of their data by a mobile sink is generated. In comparison to other algorithms, the proposed algorithm reduces the overall distance travelled by the mobile sink by at least 8%, extending the network's lifetime.

Kumar et al. [56] proposed AntQHSeN, which is an ant-based QoS routing protocol for heterogeneous WSNs. The routing algorithm divides all traffic into two categories: routing traffic and data traffic. Multimedia traffic and scalar traffic are two types of data traffic. The routing decision is made based on the traffic type as well as the traffic's QoS constraints. The rapid advancement of technology has resulted in the proliferation of multimedia sensor nodes, resulting in a multiplicative increase in WSN application areas. Multimedia and scalar sensors can be used to monitor environmental data as well as detect intrusion in a given area. As a result, application layer data can be classified as scalar or multimedia, depending on the QoS requirements. This paper proposes three methods for dealing with three different types of data, namely routing, multimedia, and scalar data, in order to improve network performance. In terms of packet delivery fraction, end-to-end delay, and routing overhead, simulation results show that AntQHSeN outperforms the standard AODV and EEABR.

One of the most effective population-based search techniques for tackling the well-known NP-hard combinatorial optimization problem known as the TSP is ACO. How to obtain a Pareto front with good approximation and even distribution is a persistent problem in the field of multiple objective optimization [57]. This article by cheng et al. suggests the MoACO/D framework, which uses Tchebycheff decomposition to solve bTSPs. An ant colony is divided into a specific number of overlapping subcolonies in MoACO/D in order to accommodate the decomposition of a bTSP into a number of scalar optimization subproblems. Authors described three MoACO algorithms that were created by, respectively, fusing MoACO/D with AS, MMAS, and ACS. Extensive testing on simulator of varying complexity reveals that the MoACO/D framework is successful and efficient at solving bTSPs.

Hackel et al. [58] explained how combining the ACO Meta-heuristic with Dynamic Programming and using the information values from the Look-Ahead Heuristic can significantly improve the outcomes. The single-objective version of the optimization problem can be solved in an appropriate amount of time, which underlies the usefulness. The proposed idea can be used to solve any problem with a similar structure, not just shortest path problems. The only crucial prerequisite is that the solution may employ Ant Colony Optimization and that the objective functions of the criteria fully satisfiy all the terms of Dynamic Programming.

Prasad et al. [59] provides the multiobjective ant colony based Steiner tree algorithm for balancing the hop count and total transmitted power at the terminal nodes. For each goal, the strategy employs a pheromone trail matrix. It is simple to expand on the work in the paper by taking into account more than two objective functions. Simulations are used to verify the effectiveness of the suggested technique and show that it is capable of locating the Pareto ideal solution. Future study will hybridise the approach with some heuristics in order to expand it.

For mobile opportunistic networks, we Liu et al. [60] have presented a QoS-aware data delivery strategy in this paper. In order to support effective prioritising and

redundancy control, it uses QDP to represent a node's ability to deliver data to a destination within a specified delay budget and maintains a prioritised queue, where the priority is decided by a function of traffic class and dynamic redundancy. To show and assess our suggested QoS-aware data delivery strategy, we conducted two experiments. In the first experiment, air and ground mobile nodes with controlled mobility are used to connect several clusters of static Crossbow sensors. The second experiment is conducted over a two-week period in a mobile social network environment, with the prototype running on Dell Streak Android tablets carried by 23 volunteers with arbitrary and varied mobility patterns.

By introducing the multi-route and examining simple genetic algorithms, simulated ant colony algorithms, and genetic algorithms, Sun et al. [61] suggested a new method. The stimulation testing demonstrated the fusion algorithm's viability and effectiveness. The outcomes demonstrate that, under the condition of concurrent search in opposite directions, the revised algorithm is practicable. All ants are split into two groups, which are then instructed to move away from the source location and toward the goal point in different directions to find better pathways. The search algorithm has been applied to more complicated terrain, and its effectiveness has been tested.

An energy-efficient congestion control technique for sensor networks named CODA is given by wan et al. [62] in this paper. Three crucial mechanisms make up the framework, which is aimed towards CSMA-based sensors: I receiver-based congestion monitoring; (ii) open-loop hop-by-hop backpressure; and (iii) closed-loop multi-source control. Average energy consumption and average fidelity penalty are two performance metrics we developed to measure how CODA affects the performance of sensing applications. This initial deployment produced a number of significant results. It was possible to assess CODA using a general data dissemination strategy and measure, channel loading at the receiver.

Table 2.1: Literature survey of Congestion control and path Evaluation algorithms

Reference Number	Author/Authors	Year	Main Contribution	Evaluation Parameter Taken
[33]	Wan et al.	2013	Prevents the congestion by distributing the packets from the source node at slow rate in the event of congestion occurrence. In the event of data loss at a specific node, the node tends to recover the same data segments by quick pumping. When a packet is lost, a negative ACK is issued to the sink/source node.	• Good put • Delay • Frame rate Error
[34]	Ren et al.	2012	Protocol chooses to send the traffic from alternative paths in the event of congestion. The path selected for transfer of packets should be shortest. In the case of congestion, requests are forwarded to other pathways composed of unoccupied or under-utilized nodes consisting shortest path.	• Number of packets • CDF of delay • Normalized queue length • Number of routing loops
[35]	Hull et al.	2014	Proposed a unique approach for controlling the congestion by limiting the packet transmission rate. Hop by Hop congestion control technique is applied in this protocol. The suggested method does not trust on topological evidence, instead concentrating on single-sink and network made in spanning tree process.	• Average received throughput • Fairness • Median received throughput • Latency
[36]	Wan et al.	2016	A novel algorithm for	• Normalized

			preventing the congestion by using the combination of buffer occupancy and channel load. Congestion notification is done with the help of backpressures. The proposed procedure does not take fairness into consideration and uses AIMD approach which normally leads to loss of packets.	network throughput • Normalized node throughput • Packet loss rate • Achieved fairness
[37]	Ta and Yu	2015	A novel algorithm for detecting and preventing congestion for wireless multimedia sensor networks. The main contribution is that it handles different congestion levels with different priority depending on traffic type. The biggest downside of this strategy is that it is difficult to retrieve lost packets.	• Throughput • Number of retransmission • Delay • Total packet received
[38]	Kafi et al.	2018	Controls the congestion by sensing the link's channel capacity for each node and it parent node. According to statistics scheduling of packets and transmission rate is adjusted. The predicted capacity assists the IACC protocol in establishing a schedule that allocates suitable time and rate for transmitting to each node.	• Packet reception ratio • Emission attempted • Throughput
[39]	Jaiswal and Yadav	2017	Proposed a congestion control algorithm based on fuzzy logics. Algorithm boost the packet delivery fraction by limiting the packet transmission rate and increasing the number of packets sent to sink node.	• Number of packets • Buffer occupancy • Threshold

			When congestion is observed, the congested node explicitly instructs neighbors to reduce transmission rate, therefore controlling packet drop.	
[40]	Sergiou et al.	2018	Proposed algorithm controls the congestion by routing the packets from alternative paths in the event of congestion thereby decreasing the delay of transferring the packets from source node to sink node. This adaptive technique uses buffer occupancy as a first indicator of congestion occurring, and then it uses the out/in data rate ratio to initiate the building of an alternative way.	• Sink throughput • Average received packet ratio • Average hop by hop delay • Percentage of network remaining energy
[41]	Aghdam et al.	2018	Proposed a novel algorithm for congestion control in WMSNs by adjusting the sending rate and dispersal rate of packets. The unique feature of the protocol is that it ignores less valuable packets and sending the valuable packets with high priority.	• Loss probability • Number of packets lost • Number of frames lost • Number of packets in GOP • PSNR • Average delay • Network throughput • Average energy consumed at source node
[42]	Antoniou et al.	2013	A nature inspired congestion control protocol has been proposed based upon self-adaptive and scalable nature of flocking behaviors of birds. Protocol controls the congestion by routing the packets from lively nodes	• Packet delivery ratio • End to End delay • Packet loss • Energy Tax

			thereby ignoring the dead and congested node.	
[43]	Razaee et al.	2013	Proposed a novel algorithm for controlling the congestion in WSNs health care application. In health care application we can have sensitive data regarding patient. The protocol sends the sensitive data through non congested path for better delivery of information.	• Life time • Network remaining energy • Packet loss • Average end to end delay • Average queue size
[44]	Banimelhem and Khasawneh	2012	Proposed a novel algorithm for congestion control by dividing the WSNs into grid of sensors. In each grid there is a master node that collects the information from all the nodes in the grid. The master node is connected to number of available paths and in the event of congestion, the master node sends the data through best available path.	• Average residual Energy • Average end to end packet delay • Delivery ratio • Average buffer occupancy
[45]	Heikalabad et al.	2017	Proposed a novel algorithm for dynamically predicting the congestion in the wireless sensor networks. The proposed protocol in the paper was intended at forecasting the congestion and dissemination the traffic on the network equitably and vigorously.	• Packet loss • Throughput • End to End delay
[46]	Sergiou et al.	2014	Proposed a self-motivated substitute path algorithm to avoid and control congestion in WSNs. Proposed Algorithm attempts to reduce the chances of congestion by choosing the alternative paths available in the	• Average received packet ratio • Number of packets received by the sink. • Average Throughput • Average hop by

			networks.	hop delay
[47]	Misra et al	2009	The major aim of the proposed approach is to adapt the packet arrival rate of the child nodes to the paternal node's packet service rate, therefore preventing congestion. Another important feature of the approach is that it learns intelligently from the past and improves the performance of the network as the time progress by choosing better data rate in the future.	• Throughput • Number of packets received to the sink node. • Energy Consumed
[48]	Narawade and kolekar	2019	The main objective behind the proposal is to make equality between the packet arrival rate and packet service rate at child node as well as parent node respectively. Congestion is dodged based on packet fall likelihood, and in the occasion of congestion, the optimization algorithm returns a new share rate for the child nodes that is lower than the average rate of the previous period.	• Throughput • Delay • Normalized packet loss • Sending rate • Normalized queue length • Congestion level
[49]	Yadav et al.	2021	An ant colony is used in the proposed algorithm, backward ant and forward ant algorithms are used in this approach. Moreover, Huffman coding was applied on top of the sample space provided by the ant colony approach to select one of the less congested paths as the optimal data routing path. The proposed algorithm's	• Average Energy consumption • Average residual energy of node • Average throughput • Average hop by hop delay • Packet delivery ratio • Node death percentage

			extensive performance results outperform state-of-the-art algorithms in terms of various parametric situations.	
[50]	Sangeetha et al.	2016	In proposed algorithm, When there is congestion, the best path is chosen using the heuristic LRTA search algorithm, which is mission critical for locating the goal. To reach the sink node, fuzzy logic rules are used to make a priority decision that will be treated with a search algorithm. This reduces energy consumption by preventing congestion in the WSN.	• Energy Consumption • Packet loss
[51]	Ding et al.	2016	In proposed algorithm, the application of node location and the reasonable allocation of traffic flow in the network are used to optimize network transmission performance. The link gradient function provides the basic routing backbone for directing packets to the sink. Furthermore, the traffic radius function can distribute excessive packets over multiple paths that include idle and under loaded nodes. These two functions are combined to form flow rate, which is used to make dynamic routing decisions.	• Packet loss rate • Average routing hop • Energy consumption per received packet
[52]	Raman et al.	2019	In proposed algorithm, First, authors use a multi-input time on task optimization algorithm to	• Network life time • Average hop count • Energy

			find the best intermediate node. Then, using a modified gravitational search algorithm, a path between sources and sink nodes is computed, providing a better routing path. In comparison to the existing CCOR scheme, experimental results show that the FCC scheme can effectively control congestion.	consumption • Data drop ratio
[53]	Rezaee et al.	2018	This protocol proposes a new active queue management method that combines the RED method and the FuzzyPID approach to calculate packet loss probability. In addition, proposed algorithm estimates each node's output transmission rate and then assigns a suitable transmission rate to each node. Overall, the simulation results suggest that the proposed protocol could outperform CCF, PCCP, and OCMP protocols	• Energy • Mean queue length • Mean end to end delay • Packet loss probability • Packet loss rate
[54]	Onthachi et al.	2019	A novel algorithm to establish near-optimal routes for data transmission under multi-constrained QoS. Furthermore, proposed algorithm improves load balancing by dynamically selecting an alternate path for data transmission from a subset of best-case paths	• Network lifetime • Throughput • End to end delay • Total energy consumption • Packet delivery ratio
[55]	Yogarajan et al.	2020	Proposed an improved Clustering-based data collection using Ant Lion Optimization in this paper. Because of the proposed algorithm's	• Number of dead nodes • Number of alive nodes • Cumulative

			optimal use of a node's available energy in the battery, the network's lifetime is extended, a greater number of packets are received at the base station, improving the network's throughput, and the number of individual nodes is reduced after clustering.	residual energy • Throughput • Lifetime of network • Number of individual nodes • Number of cluster heads
[56]	Kumar et al.	2014	Proposed AntQHSeN, which is an ant-based QoS routing protocol for heterogeneous WSNs. The routing algorithm divides all traffic into two categories: routing traffic and data traffic. The routing decision is made based on the traffic type as well as the traffic's QoS constraints. In terms of packet delivery fraction, end-to-end delay, and routing overhead, simulation results show that AntQHSeN outperforms the standard AODV and EEABR.	• Packet delivery fraction • End to End delay • Normalized routing overhead
[57]	cheng et al.	2012	A novel bi-objective algorithm based on ant colony optimization is proposed in this paper. Main objective is to provide the optimal path for travelling salesman problem.	• Statistical parameters- Mean and Standard deviation.
[58]	Hackel et al.	2008	Ant colony optimization technique is used for calculating the shortest path in order to approximate praetor-fronts.	• Diversity structure
[59]	Prasad et al.	2013	A novel technique has been proposed in this paper to find and construct an optimal	• Delay • Error rate

			energy efficient path for routing the real time traffic.	
[60]	Liu et al.	2014	A quality of service aware algorithm has been proposed to enhance the reliable delivery of data without delay. In addition to this priority queue has been defined to route the packets according to priority.	• Average delivery rate • Average message redundancy • Delay
[61]	Sun et al.	2010	Route optimization method is proposed for routing the traffic in wireless sensor networks. Optimization is obtained by combination of ant colony optimization technique and genetic algorithm.	• Fitness value
[62]	Wan et al.	2003	An energy efficient congestion detection and control mechanism has been proposed in this paper. The receiver node is being used for d signaling the event of congestion and hop by hop backpressure technique is used for controlling the congestion.	• Average energy tax • Average fidelity penalty

2.3 Techniques and Procedures proposed for fair rate assignment for congestion control

Rate-based congestion management approaches control congestion by restricting the traffic rate at which packets from the source node are sent to the sink node. There are two types of rate-based congestion control methods: end-to-end and hop-by-hop. When using hop-by-hop rate control, all relay node lengthways the path keeps packet evidence in its buffer. This strategy is energy-efficient because the shorter retransmission distance

conserves a node's energy. Furthermore, under high feedback delays, this technique significantly reduces the required buffer size. Each intermediate node keeps an eye on the overflow of its buffer. As a result, when linked to the end-to-end mechanism, congestion is detected quickly and easily. When congestion develops, all intermediary nodes from source to destination are alerted and ordered to take necessary action. When a timeout occurs or the end node gets duplicate acknowledgement messages, congestion is indicated. The prime of a specific method, such as hop-by-hop or end-to-end, is influenced by the dependability and time-sensitivity of the fundamental request. Congestion management techniques based on traffic offer their own set of benefits, such as the ability to target crowded nodes with a congestion control approach. The summary of such methodologies is given below:

In [63], a novel algorithm for rate based priority for Multimedia WSNs was proposed. For congestion control, PRRP employs a hop-by-hop method. The PRRP protocol has three phases: congestion discovery, congestion announcement, and rate revision. The PRRP protocol employs buffer occupancy during the congestion detection phase. It calculates the level of congestion at each node using minimum and maximum threshold values. If the queue length hits the supreme verge, the child nodes must decrease their data speeds to avoid congestion and packet loss. The real congestion level is assessed and the data rate is adjusted accordingly if the queue length falls between the maximum and minimum values. The PRRP protocol has the drawback of not choosing the optimal path from the source node to the sink.

Adaptive Rate Control (ARC) [64] observes packet inoculation as well as route-through traffic in the traffic stream. Each node calculates the number of upstream nodes, and bandwidth is allocated equally between route-through and locally produced traffic, with the former receiving priority. As a result, each node's bandwidth allocation is nearly equal. Furthermore, a drop in route-through traffic transmission rates has a backpressure impact on upstream nodes, causing them to reduce their transmission rates.

The event-to-sink reliable transport protocol is proposed by the authors in [65]. ESRT assigns transmission rates to sensors in order to receive an application-defined quantity of sensor interpretations at the sink while keeping the network free of congestion. The rate distribution is done centrally at the base station. If a sensor node's local buffer level exceeds a certain threshold, ESRT adds a congestion notification bit to the packets it sends to the sink. This technique, however, has a few disadvantages. For example, because the sink must show this control indication at a extraordinary energy level so that all sources can hear it, an ongoing event transmission can be interrupted by this high powered congestion signal.

Rangwala et al. [66] have projected Interference-Aware Fair Rate Control Protocol for WSNs. In WSNs, when numerous nodes are communicating the data to a single base station, their transmissions may interfere with each other. IFRC provides a novel congestion control mechanism by dynamically allocating fair and efficient transmission rate to the nodes whose transmission may interfere with each other. The protocol works in four interrelated parts: first one that extent the level of congestion at each node, second that share the congestion information with the potential interferer, third that acclimates the rate using AIMD technique and fourth that provide efficient and fair rate transmission. IFRC uses buffer occupancy for measuring the congestion level of each node. When the queue size of a node exceeds the pre-defined threshold value, node is said to be congested. Congestion sharing part of the protocol deals with sharing the queue length (congestion level) explicitly with the potential interferer. Finally AIMD approach is used for adapting the rate in case of congestion. If a node is congested, then its rate halves by some pre-defined value and if the node is not congested, its rate is increased by some pre-defined value. The protocol provides the efficient and fair rate, by allocating the rate to potential interferer not more than the congestion node.

The Rate Controlled Reliable Transport protocol is proposed by the authors in [67]. This principle is designed for programs that demand reliable data transmission from source to sink nodes and are loss-intolerant. End-to-end explicit loss recovery is used in

RCRT using a NACK-based approach. All congestion detection and rate adaption capability is also placed in the sinks by RCRT, resulting in a centralized congestion control method.

Authors have outlined a plan for WSNs congestion control in this paper [68]. The bandwidth allocation in resource-constrained WSNs has been fairly distributed thanks to the new mechanism for congestion control presented in this paper. In terms of packet loss, energy efficiency, channel utilization, and fairness, simulations have demonstrated that FACC performs better than backpressure schemes. In particular, compared to no congestion control, FACC can improve throughput by up to 20%. When it comes to packet loss, FACC becomes stable over time regardless of the increase in offered traffic load, but when there is no congestion control, packet loss increases linearly over time or as offered traffic load increases.

A novel congestion controller protocol for reliable transmission was proposed in [69] known as CCRT. For reliable delivery of essential information, it employs a priority-based congestion control technique. Each node has a particular queue priority, such as high, mid, and low. The queue length and queue fluctuation rate are used by CCRT to detect congestion. A positive queue variation rate implies that congestion is likely to develop in the following time interval, whereas a negative variation rate suggests that congestion has been alleviated. When the queue fluctuation rate crosses a set percentage, it signifies the possibility of congestion. However, if the wait length is expanding at a significantly higher rate, it indicates that there is a strong likelihood of congestion in the near future.

The authors of [70] investigated a congestion identification protocol for specific application based on data priority for clustered WSNs. When the assessment of a seized packet in PASCCC hits a preset threshold value, all basis node turns on its radio, detects the environment, gathers the data, and sends it upstream to the sink station. Sensitive packets are prioritized during congestion to ensure that they get at the base station on time. As a result, data transmitted across the network is assured to be dependable without

sacrificing quality. To maintain coverage integrity, each cluster head utilizes a novel queue scheduling mechanism that ensures the extra resources used by faraway nodes are properly utilized. The mobility of the nodes during the setup phase of PASCCC results in unacceptably long delay. This is primarily due to the frequent changes in a node's position.

A novel algorithm for controlling the congestion in WSNs was proposed in [71] which is effectively based on weighted fairness technique. WFCC implements node weighting to indicate the relevance of each node. It is founded on the concept that the data produced at each node varies considerably depending to the node's significance. The WFCC analyzes congestion by calculating the predicted packet service time to the average packet inter-arrival time. The feedback at each interval causes WFCC to suffer unnecessary overhead.

Grover et al. [72] proposed a novel congestion control scheme (RACC technique) to control congestion and improve delay performance. RACC improves the existing congestion control process by using three levels of buffer occupancy for a single node. The queue length on each node is checked on a regular basis to ensure that the buffer occupancy percentage is maintained. The level is determined based on the buffer occupancy, and the data rate is reduced accordingly. In terms of throughput, packet delivery ratio, normalised routing overhead, MAC overhead, and average end-to-end delay, the RACC mechanism outperforms the existing DACC approach, according to the findings. When comparing J-ERLB and RACC techniques, however, energy remains a tradeoff factor. Furthermore, the most appropriate modulation scheme for the proposed RACC mechanism is evaluated in order to achieve overall efficiency for sensor nodes using RACC in WSNs in order to transmit data to distant sensor nodes.

Tambe et al. [73] proposed a study to compare the lifetime and energy consumption of WSNs with and without mobile nodes, as well as the results obtained by varying the number of mobile nodes, their location, and their speed. The RR is also adjusted to control the buffer occupancy of each node and to alleviate sensor network

congestion. Based on the simulation results, the proposed work extends the life of nodes with high power consumption rates and improves the network's overall longevity. The use of a dual threshold for buffer and mobile node can help to reduce congestion and data delay by reducing the time it takes for data to arrive.

Pan et al. [74] examined the cause of BBR's unfairness to different RTT flows in this paper, which was based on a study of BBR congestion control. BBR-ACW was proposed to address the original BBR's fairness issue when dealing with different RTT flows. The scaling factor a and b were defined to adjust the corresponding cwnd gain based on the delivery rate and queue status of different RTT flows. This scheme ensures that the various RTT flows compete fairly for the bottleneck bandwidth. On the NS3 platform, many simulation experiments for various RTT flows with various bottleneck bandwidths and buffer sizes have been carried out. The results show that BBR-ACW outperforms BBR, BBQ, and DA-BBR in terms of improving RTT fairness, with the highest fairness index of the four algorithms under various experimental conditions. Furthermore, in data transmission experiments, BBR-ACW significantly reduced the number of retransmissions when compared to original BBR. Even in small bottleneck buffers, lower packet retransmission can be maintained, indicating that the buffer size has a smaller restriction on BBR-ACW. Furthermore, latency was greatly improved, with BBR-ACW having a lower average queuing delay than BBR, BBQ, and DA-BBR.

By applying and experimenting in real testbed experiments, kim et al. [75] proposed an improved bottleneck queue buildup suppression method in this paper. In comparison to various RTT flows, the proposed BBR-E had a higher throughput than BBR. In addition, we revealed that the BBR-E occupied sufficient bandwidth when compared to the BBR through a concurrent competition experiment with multiple flows.

Esmaeelzadeh et al. [76] modeled MAC-layer delay over head of a CR node in this paper. The probability density functions of CR attainable sending rate and the input rates of different CR nodes have been modeled in the CRSN with respect to the sending rate of the collecting CR sensors in the transport layer, based on the MAC-layer delay

model. The steady-state queue length distribution of a CR node in the CRSN has been modeled using the probability density function of input rate and the attainable send ing rate of a CR node in the CRSN (SMC). The proposed queue length distribution was used to calculate the congestion probability of each CR node in the network. Semi-Markov chains have been used to modify the steady-state sending rates distribution of CR collecting sensors for generic AIMD and generic AIAD congestion control schemes. Various simulation experiments have been used to validate the sending rate distribution model.

Srivastva et al. [77] proposed a project with the goal of lowering energy consumption while improving network reliability and QoS. To solve the WSNs network's congestion problem, the proposed method employs a novel rate-based congestion control method. With the use of AWF optimization technology, energy could be consumed. Over a hierarchical wireless sensor network, the Hybrid K-means and Greedy Best First Search algorithms provide QoS clustering. This technique allows for energy-efficient transmission. The proposed method has been validated, and it has been concluded that it is best suited for wireless applications that require congestion control, minimal delay, and a high level of QoS. Over a hierarchical wireless sensor network, the Hybrid K-means and Greedy Best First Search algorithms provide QoS clustering. This technique allows for energy-efficient transmission. The proposed method has been validated, and it has been concluded that it is best suited for wireless applications that require congestion control, minimal delay, and a high level of QoS.

Wei et al. [78] proposed a BBR-based Congestion Control and Packet Scheduling (BCCPS) for Multipath TCP in order to improve the performance of MPTCP in heterogeneous wireless networks by taking advantage of BBR and considering bottleneck fairness. This paper focuses on reducing the completion time of mouse flows while providing excellent goodput to elephant flows. BCCPS uses two new mechanisms: a well-designed BBR-based congestion control and a fine-grained packet scheduling algorithm to maximise goodput over all available paths. BCCPS uses these mechanisms

to continuously monitor and analyse the dynamic network environment, as well as to estimate the quality of each transmission path. BCCPS intelligently detects the shared bottleneck based on the path quality evaluation output and balances the congestion among the subflows while maintaining fairness with the single flow that shares the bottleneck. Furthermore, BCCPS uses a two-phase packet scheduling scheme to maximise elephant throughput while reducing end-to-end delay for mouse flows. Data delivery efficiency is improved by using a two-phase packet scheduling scheme. In terms of goodput and end-to-end delay, simulation results and real-world network tests show that the proposed BCCPS outperforms the default MPTCP.

One of the contributing factors in the performance of Wireless Sensor Networks is transmission rate (WSNs). Reduced network response time, queuing delay, and more packet loss are all symptoms of a congested network. Kazmi et al. [79] proposed a transmission rate control method to address this problem. The current node in WSNs adjusts its transmission rate based on information from the downstream node about traffic load. Support Vector Machine is used to control congestion using multi classification (SVM). Differential Evolution (DE) and Grey Wolf Optimization (GWO) algorithms are used to tune the SVM parameters to reduce miss classification error. In terms of classification error, the proposed approaches DE–SVM and GWOSVM outperform the other classification techniques, according to the comparative analysis.

A low power and constrained broadcast range ad hoc network's performance is typically worsened by the congestion issue in terms of several performance metrics. Chughtai et al. [80] proposed a method for reducing traffic congestion is employed to proactively reduce network congestion. Contrary to conventional methods, CAP doesn't change the source traffic rate to reduce congestion. CAP makes use of both explicit and implicit strategies to reduce network congestion. It attempts to locally alleviate congestion as part of its implicit mechanism for reducing congestion in order to lessen information loss. If and only if the implicit congestion alleviation mechanism was unable to choose the next hop nodes with low congestion and high energy level, it could actively

alleviate congestion. As per the congestion or low energy, it uses its congestion alleviation process to bypass the damaged node or link.

In this paper, brahma et al. [81] presented a distributed congestion control technique that aims to adaptively allocate each node a fair and effective transmission rate for tree-based communications in wireless sensor networks. Each node in proposed algorithm keeps track of the overall output and input traffic rates. A node then chooses whether to increase or decrease the bandwidth allotted to a flow coming from itself and to those being routed through it based on the difference between the two. Proposed congestion control is built to respond to changes in the underlying routing topology and operates independently of the underlying routing algorithm. The total increase or reduction in traffic rate is calculated by the usage controlling module. The fairness module determines the precise distribution of the required overall change in traffic rate among the flows.

Table 2.2: Literature survey of fair rate assignment for congestion control

Ref No.	***Author(s)***	***Year***	***Main Contribution***	***Evaluation parameter taken***
[63]	Tshiningayamwe et al.	2016	A novel algorithm for rate based priority for Multimedia WSNs was proposed. The PRRP protocol employs buffer occupancy during the congestion detection phase. It calculates the level of congestion at each node using minimum and maximum threshold values. If the queue length hits the supreme verge, the child nodes must decrease their data speeds to avoid congestion and packet loss	• Throughput • Delay • Packet delivery ratio • Average energy consumption
[64]	Woo et al.	2011	Proposed algorithm used packet injection as well	• Packets received by base station

			as route-through traffic in the traffic stream. Each node calculates the number of upstream nodes, and bandwidth is allocated equally between route-through and locally produced traffic, with the former receiving priority.	• Mean throughput • Transmission probability
[65]	Akan et al.	2005	Proposed algorithm assigns transmission rates to sensors in order to receive an application-defined quantity of sensor interpretations at the sink while keeping the network free of congestion. The rate distribution is done centrally at the base station. If a sensor node's local buffer level exceeds a certain threshold, algorithm adds a congestion notification bit to the packets it sends to the sink	• Normalized event reliability • Energy consumption
[66]	Wan et al.	2018	Proposed algorithm uses buffer occupancy for measuring the congestion level of each node. When the queue size of a node exceeds the pre-defined threshold value, node is said to be congested. IFRC provides a novel congestion control mechanism by dynamically allocating fair and efficient transmission rate to the nodes whose transmission may interfere with each other. If a node is congested, then its rate halves by some pre-defined value and if the node is not congested, its	• Throughput • Fairness • Packet delivery fraction

			rate is increased by some pre-defined value.	
[67]	Paek et al.	2010	A novel algorithm designed for the networks that demand reliable data transmission from source to sink nodes and are loss-intolerant. End-to-end explicit loss recovery is used in proposed algorithm using a NACK-based approach. All congestion detection and rate adaption capability is also placed in the sinks by algorithm, resulting in a centralized congestion control method.	• Throughput • End to end delay • Packet delivery fraction • Energy consumption
[68]	Yin et al.	2009	A novel algorithm for bandwidth allocation in resource-constrained WSNs is proposed. The protocol will be able to assign a fair bandwidth to all the nodes in the networks in the event of congestion. In terms of packet loss, energy efficiency, channel utilization, and fairness, simulations have demonstrated that FACC performs better than backpressure schemes.	• Number of dropped packets • Total source rate • Energy expenditure • Throughput
[69]	Hua et al.	2014	For reliable delivery of essential information, proposed algorithm employed a priority-based congestion control technique. Each node has a particular queue priority, such as high, mid, and low. When the queue fluctuation rate crosses a set percentage, it signifies the possibility of congestion.	• Total normalized throughput • Delay • Lost probability
[70]	Jan et al.	2014	Proposed a congestion	• Packets count

			identification protocol for specific application based on data priority for clustered WSNs. Sensitive packets are prioritized during congestion to ensure that they get at the base station on time. As a result, data transmitted across the network is assured to be dependable without sacrificing quality.	• Number of nodes associated with cluster head • Life time of network • Number of packets
[71]	G et al.	2012	A novel algorithm for controlling the congestion in WSNs which is effectively based on weighted fairness technique. It is founded on the concept that the data produced at each node varies considerably depending to the node's significance. The algorithm analyzes congestion by calculating the predicted packet service time to the average packet inter-arrival time.	• Throughput • End to end delay • Packet delivery ratio • Energy consumption
[72]	Grover et al.	2021	Proposed a novel congestion control scheme to control congestion and improve delay performance. The queue length on each node is checked on a regular basis to ensure that the buffer occupancy percentage is maintained. The level is determined based on the buffer occupancy, and the data rate is reduced accordingly.	• Throughput • Packet delivery ratio • Normalized routing overhead • Mac overhead • Average end to end delay • Average remaining energy •
[73]	Tambe et al.	2018	Proposed a study to compare the lifetime and	• Average energy consumption

			energy consumption of WSNs with and without mobile nodes, as well as the results obtained by varying the number of mobile nodes, their location, and their speed. The rate is also adjusted to control the buffer occupancy of each node and to alleviate sensor network congestion.	• Total energy consumption • Throughput • Fairness index • Life time
[74]	Pan et al.	2021	BBR-ACW was proposed to address the original BBR's fairness issue when dealing with different RTT flows. This scheme ensures that the various RTT flows compete fairly for the bottleneck bandwidth. On the NS3 platform, many simulation experiments for various RTT flows with various bottleneck bandwidths and buffer sizes have been carried out.	• Throughput • Jain's fairness index • Retransmission • Round trip time
[75]	Kim et al.	2019	Proposed a novel algorithm which is an improvement of bottleneck queue buildup suppression method in this paper. In comparison to various RTT flows, the proposed BBR-E had a higher throughput than BBR	• Throughput
[76]	Esmaeelzadeh et al.	2015	The proposed queue length distribution was used to calculate the congestion probability of each CR node in the network. Semi-Markov chains have been used to modify the steady-state sending rates distribution of CR collecting sensors	• Probability mass function • Cumulative mass function

			for generic AIMD and generic AIAD congestion control schemes.	
[77]	Srivastva et al.	2020	Proposed a project with the goal of lowering energy consumption while improving network reliability and QoS. To solve the WSNs network's congestion problem, the proposed method employs a novel rate-based congestion control method. The proposed method has been validated, and it has been concluded that it is best suited for wireless applications that require congestion control, minimal delay, and a high level of QoS.	• Energy efficiency • Average end to end delay • Average throughput • Packet delivery proportion • Reliability analysis
[78]	Wei et al.	2021	Proposed a BBR-based Congestion Control and Packet Scheduling (BCCPS) for Multipath TCP in order to improve the performance of MPTCP in heterogeneous wireless networks by taking advantage of BBR and considering bottleneck fairness. BCCPS intelligently detects the shared bottleneck based on the path quality evaluation output and balances the congestion among the subflows while maintaining fairness with the single flow that shares the bottleneck.	• Detection accuracy • Normalized download time • Goodput ratio to single path • End to end RTT
[79]	Kazmi et al.	2019	proposed a transmission rate control method to address this problem. The current node in a WSNs adjusts its transmission	• Throughput • Avergae energy consumption • End to end delay

			rate based on information from the downstream node about traffic load. Support Vector Machine is used to control congestion using multi classification.	• Packet delivery fraction
[80]	Chughtai et al.	2016	Proposed a congestion alleviation procedure to route the traffic through less congested path in the event of congestion. The chosen path may be energy efficient and optimized in terms of congestion.	• Throughput • Energy consumption per packet • End to end delay • Packet delivery fraction
[81]	Brahma et al.	2010	Proposed a controlling module to alter the traffic rate of source node or intermediate nodes otherwise for controlling the congestion. Ultimate goal is to achieve fairness in the network by controlling the traffic rate in the overall wireless sensor network	• Mean packet generation rate • Throughput • Data generation rate • Mean packet re transmission rate

2.3 Techniques and Procedures proposed for detecting and mitigating the effect of adversaries in WSNs

The authors of [82] suggested a novel algorithm which is secure as well as energy-efficient which provides security and prolonged lifetime of WSNs. Each node in proposed technique is responsible for keeping track of the routes in the routing tables, which relate to potential neighbors who can relay packets to the sink node. The suggested method is distinguished by its ability to determine a routing path utilizing a base station rather than relying on the source or sink node. As a result, it expressly combats routing protocol assaults that lure traffic by giving a high-quality path to the base station. The suggested design is similar to client/server paradigm, with the base station acting as the

server and the sensor nodes operating as clients. The base station is in charge of discovering, maintaining, and selecting routes.

A novel Energy Efficient Node Split Routing Protocol [83] is presented for finding numerous pathways depending on the node's energy consumption rate and occupied queue length. The digital signature crypto system is being used to improve safety by constructing the digital signature with the MD5 hash function and creating a pair of private and public keys with the RSA technique. To eliminate loops, the network is intended to be divided into stages based on the numeral of hops between the source and destination. The routing tables are generated by sink node to preform multipath routing.

The authors of [84] proposed a novel strategy that takes broadcast range modification into account to improve energy effectiveness. Prior to modifying the transmission range based on the position of their neighbor node, node disjoint many channels for data transmission are chosen to divide the burden across various nodes. The primary objective of transmission range modification is to minimize the amount of energy used by WSNs. As a result, the transmission range is reduced if the neighbor node is closer. The proposed protocol process is divided into two phases: routing table generation and data transfer. If a node gets the route building packet, its route construction sequence number is increased by one before forwarding. Data is transferred on the principal channel while retaining multiple pathways during the data transmission phase.

The authors of [85] propose a novel approach for multipath and clustered sensor networks to increase network performance, resistance, and endurance. SCMRP employs the right usage of cryptographic methods to give enough security to the sensor network. Proposed algorithm is also intended to address security, orphan node, and multi-hop difficulties in clustering protocols. During route discovery, the certificate is used to validate any node with the base station's public key, while the unique shared key is used to interact with the base station. To identify neighbor events, the node begins advertising and receiving the node's ID and CERT in the neighbor detection packet. If the sender

node is validated, the receiver node adds its ID to the list of neighboring; then, the packet is dropped. As a result, the unlawful node is no longer permitted to interfere with the neighbor discovery phase. The base station computes the secret key for each pair of neighbor nodes and builds the network topology based on the network's neighbor information. In addition, BS generates a neighbor matrix in order to locate the multiple pathways.

The authors of [86] suggested a secure on-demand routing method that is resistant to attacks. The idea of k-connectivity is extended in the proposed approach to k-t-connectivity, where t is the disjointness threshold that indicates the maximum number of nodes shared by any two paths in the set of t constructed paths. Markers in datagrams are used to attach the threshold t during the route finding phase. To withstand routing attacks, the suggested approach employs an elliptic threshold monogram, a watchdog instrument, and multipath. Because most multipath routing methods produce maximally discontinuous paths, node performance suffers and resources are depleted, especially when the routes comprise several hops and the node density is large. To address this problem, the proposed technique enriches multipath routing algorithms by enforcing a service-orientated threshold during route identification, allowing for a trade-off amid fault tolerance and data transmission. A sensor node should check that *t* paths between itself and the base station are accessible before transmitting data. Then, to reduce the likelihood of communication breakdown, it transmits repeated copies of data across the alternate pathways.

Because the majority of WSNs are deployed in hostile environments, some form of security is highly desirable. The secure routing protocol is a practical method of ensuring data security as it travels from source to destination. Alrajeh et al. [87] detailed a secure routing protocol for WSNs based on the ant colonisation technique in this paper. Hello packets are used by authors to discover their neighbours. The mechanism used by Authors is based on forward ants, which collect and increment reputation values along the path. Similarly, the destination node employs backward ants, which carry information

and instructions about route security from the destination node. Not only does the proposed mechanism use two paths for data forwarding to overcome the problem of node failure, but it also improves the overall network's efficiency. When compared to other routing protocols like iACO and LEACH, the proposed routing scheme has a better end-to-end delay, routing overheads, and data forwarding efficiency. Furthermore, in the presence of malicious nodes, the proposed mechanisms show a high data delivery rate.

Bok et al. [88] proposed an energy-efficient securing scheme for wireless sensor networks in this paper to improve energy efficiency and security while minimising energy consumption by restricting the decryption of exposed data even if transmitted data in a wireless sensor network is intercepted. To suit the characteristics of wireless sensor networks, the proposed scheme used a TinyMD5 algorithm, a variant of MD5 that is a one-way hash function. Furthermore, by transmitting divided data with GPSR, the number of communication messages between nodes was reduced, allowing for more efficient energy use while also enhancing security. Due to the use of the hash function, the size of transmitted data remains constant regardless of the data sizes used, giving the proposed scheme the advantage of having no communication influence in the sensors despite data size increases. In comparison to existing schemes, the proposed scheme showed lower communication costs and longer network lifetimes during the performance evaluation, while maintaining security.

Salim et al. [89] proposed a dataset for detecting wireless sensor networks generated by a computer. The wireless sensor network environment was simulated using the network simulator NS-2, which uses the LEACH routing protocol to gather data from the network and preprocess it into 23 features that classify the state of each sensor and simulate five types of DoS attacks. The Intrusion Detection System (IDS) is a critical cyber-security tool for detecting and tracking intrusion attacks. With the recent advancement of information technology, the use of computer networks for a variety of applications such as finance, business, industry, health, and other aspects of human life has increased even more. As a result, information technology administrators must

develop and deploy secure and reliable networks. The rapid advancement of information technology has resulted in a number of threats to the construction of a stable and reliable network. Computer networks are vulnerable to a variety of attacks that jeopardise their confidentiality, integrity, and availability. To detect and classify the attacks, the CNN-LSTM learning model was trained over 10 and 25 epochs with a 0.001 learning rate. The overall learning algorithm had a training accuracy of 96.57 percent, and the detection model successfully identified the five types of attacks available. On ten training epochs, the CNN-LSTM intrusion detection model achieves accuracy, precision, and recall scores of 0.89, 0.894, and 0.894, respectively, on a scale of 0-1.

With the help of the proposed DLDM frame structure, the network is protected against DoS attacks in this paper [90] . This lightweight detection mechanism, combined with deep learning for efficient detection and isolation, is used for scalable and energy-efficient cluster-based countermeasures. Because the nodes' energy consumption is reduced, their lifetime is increased. The simulation results and performance analysis of the DLDM and MAS approaches show that the proposed DLDM frame structure has the best results in terms of detecting DoS attacks. This mechanism can be used on nodes that have low mobility or none at all.

In this paper, Sahoo et al. [91] propose a pragmatic energy model for WSNs using HBMA, as well as a trust-based secure and energy-efficient clustering method. We also argue that our energy model is the most appropriate for a real-world scenario because it includes all of a sensor node's basic functions. Clusters closer to the base station are smaller in size than clusters further away from the base station in order to balance energy expenditure among cluster heads. As a result, cluster heads that are closer to the base station conserve energy. We've also added the most desirable TRUST mechanism to choose an appropriate cluster head, which prevents malicious nodes from becoming cluster heads.

Pajila et al. [92] proposed a new FBDR method for detecting DDoS (Flooding) attacks and redirecting data packets to the sink via a different path. The FBDR method

examines each sensor's energy consumption, response time, and data packet count. To detect the occurrence of a DDoS attack, the FBDR method employs a type 1 fuzzy-based rule. As a result, it quickly pinpoints the sensor node that was hit by the DDoS attack. Furthermore, the packets are redirected to the sink via an alternate path using the recovery method to avoid packet loss. The recovery method employs a type 2 fuzzy-based rule that examines packet size, energy consumption, and node distance. When compared to other schemes, the proposed method saves up to 20% on energy consumption. The proposed research investigates the FBDR method's energy efficiency by looking at buffer usage, packet drop rate, response time, and network lifetime. The future focus would be on prevention measures using the neuro-fuzzy approach, based on the findings of this study. As a result of the findings of this study, we intend to work on making the FDBR method more up-to-date by combining the advantages of neural networks and fuzzy inference systems. So that the DDoS attack can be mitigated at an early stage.

The proposed DRBN frame structure by Prem Kumar et al. [93] is used to protect the network from DoS attacks in this paper. A lightweight detection mechanism is described, as well as deep learning, for efficiently detecting and isolating DoS attacks. The nodes' lifespan is extended as their energy consumption is reduced. Simulation findings and performance analysis of the DRBN and MAS approaches show that the suggested DRBN system can achieve the best results in terms of detecting DoS attacks. This mechanism can be used in nodes with very little or no movement. This model can be used to improve deep learning by incorporating various optimization methods, as well as for any public dataset that may be subject to DoS attacks in the future. The outcome of this model guides future spatial knowledge research, assisting in the discovery of new ways to eliminate unknown adversaries from wireless networks. Wireless sensor networks have been shown to be more resilient to DoS attacks than wired sensor networks.

Raghav et al. [94] proposed a Beeware routing scheme that is simple and effective in dealing with routing attacks like flooding, spoofing, and Sybil. Other methods can also counter these attacks, but not to the same extent as WSN. The proposed scheme includes two types of scout bees: primary and secondary scout bees. The primary scout bee aids in the identification of the nearest hop node by locating a suitable selection of nearby nodes using whooping signal (Hello messages). Calculating the node's trust value is an effective way to select a path. The secondary scout bee identifies the malicious node in the network using handshake schemes and encryption techniques. When there are a large number of secondary scout bees, it also contains a rank-based selection mechanism. This method is used to show which path is the most efficient for transporting data to the target node. Beeware routing scheme includes a security mechanism that is compared to other mechanisms. The proposed scheme achieves a better result by reducing the end-to-end delay and packet loss. With a high packet delivery ratio, it extends the network's lifespan. The proposed scheme contains two or optimal paths by rank selection method, and the primary and secondary scout bees help to forward the data without the interference of the attackers, resulting in a high path efficiency.

Opinder Singh et al. [95] created the novel model to detect and isolate the route request Flooding attack. The proposed model recognizes the attacker node based on two parameters: the mean numeral of route request packets forwarded by the intermediate nodes of the network and the mean abnormality from the mean of all route request packets. After these two factors have been computed, the threshold value is decided. Any node that transmits more RREQs than the threshold is flagged as an intruder node, and an alarm is issued to isolate that node. SAODV results revealed a high throughput that was comparable to the built-in AODV and a short delay that was also comparable to the built-in AODV.

T. Pandikumar et al. [96] suggested a model to protect against the route request Flooding assault. Each node in the proposed approach records the number of requests made and received in order to compute the average of route request packets, which is

utilized to determine the limit. The quantity of limit is then used to generate the threshold value, and any node sending more than the estimated threshold value is removed and tagged as an invader node. This model reduces the Packet Loss Ratio in two independent scenarios when compared to the built-in AODV under assault.

Sheetal Jatthap et al. [97] suggested an energy-based approach for detecting and isolating RREQ Flooding aggressor nodes. The proposed method evaluates an energy consumption of the node before and after the attack. The investigation is being conducted to determine the maximum and minimum energy thresholds. The node is considered dead if its energy is equal to or less than the minimal energy threshold. If the sender node has more energy than the maximum threshold, it is classified as an attacker node and put to the blacklist, thereby isolating it and restricting communication with it.

Surendra Kumar et al. [98] created a technique to protect MANETs from RREQ attacks. There are three lists for each node: whitelist, graylist, and blacklist. When a node receives a request, it looks through these three lists to find the sender. The request is dropped if the sender is on the blacklist; if the packet is on the graylist, it is checked to determine if the sender node has received a black alarm. If such an alarm occurs, the request is discarded; otherwise, the request is served. Finally, the request is fulfilled if the sender is on the whitelist. The request number obtained from the node is used to make a choice on nodes. If it surpasses the major threshold, it gets placed to the blacklist, and a black alarm is set off. If it exceeds the minor threshold, it is added to the graylist and a grey alarm is generated. If not, it is added to the whitelist.

Shashi Gurung et al. [99] suggested a novel method for mitigating RREQ Flooding attacks in MANETs. The proposed algorithm is alienated into three stages: dynamic threshold computation, authorization, and rearranging. Promiscuous mode is used by F-IDS nodes to watch the activity of other nodes in the network. After some time has elapsed, each network node generates the threshold level by using normal distribution of the number of received requests. If a node transmits false requests in exceed of the threshold, it will be considered a malevolent node in the second phase. A control packet

is forwarded to all the intermediate nodes to alert the existence of malicious node and algorithm will classify that node in the category of black list nodes. After a period of time, algorithm gives 3 chances to blacklisted node to behave as a normal node. If it's still misbehaving, the algorithm will put that node as a blocked node and further communication will be debar.

[100] Proposes the FT-AODV protocol, in which each node's traffic history is updated each time the node collects the request packet. They developed a statistic known as Threshold_ Reliable, which describes the smallest number of useable data packets that may be sent from any node in the network. The drawback of this method is that it believes a node that has been on the network for a long time is more trustworthy than a new node, yet in a selective flooding attack, the node can perform both lawfully and maliciously.

To limit the impact of a flooding attack, [101] proposes an intrusion detection program. This suggested work is founded on a finite state mechanism, wherein the states are characterized as either legitimate or malignant, and transfers from one state to another are activated by individual node activity. The engine continuously analyses the AODV protocol operations of the host node for detecting the malicious activity of individual node. The proposed technique, on the other hand, incurs both computational and memory costs.

The author of [102] presented a technique in algorithm calculates the number of route request received at an interval of 10 seconds and checks if its total value is less than the threshold value. If the number of route request packets generated by a particular node is less than the threshold value then we can decrease its blacklist index by 10 otherwise its blacklist index will be increased by 5 and node is regarded as a malicious node. The disadvantage of this strategy is a lot of overhead incurred while calculating the threshold by running the scenarios for large number of times on different traffic patterns.

The author suggested an RREQ Priority Assessment index system in [103] to limit the consequence of flooding assaults in the AODV routing protocol by employing dual threshold values, Min and Max Threshold. The proposed algorithm classifies the RREQ originator node into three levels: acceptable, moderate, and highest. The router node retains packets and sends them according to the request rate limit threshold policy for a node with acceptable priority. The relay service is carried out with priority upgrading for the node with a low priority. The node having the highest priority, all the route request will not forwarded and packets will be rejected. The author anticipated that all nodes could operate in eavesdropping mode, resulting in energy waste on the mobile nodes.

The authors of [104] suggested a new method which uses a balanced indexed value for each node to confirm whether to accept or reject a route request. There is no verification methodology in this plan that can clearly classify a node to be a malicious node and tends to drop the genuine route request of a node. A fake reply packet is forwarded to the node to test its behavior if the frequency of request packets exceeds the balanced indexed value and i.

Table 2.2: Literature survey of network adversaries – detection and mitigation algorithms

Ref No.	***Author (s)***	***Year***	***Main Contribution***	***Evaluation parameter taken***
[82]	Nasser et al.	2017	Suggested a novel algorithm which is secure as well as energy-efficient which provides security and prolonged lifetime of WSNs. The suggested method is distinguished by its ability to determine a routing path utilizing a base station rather than relying on the source or sink node.	• Lifetime • Overhead • Throughput
[83]	Murthy G et al.	2012	Proposed a novel algorithm for finding numerous pathways depending on the node's	• Packet delivery fraction • End to end delay

			energy consumption rate and occupied queue length. The digital signature crypto system is being used to improve safety by constructing the digital signature with the MD5 hash function and creating a pair of private and public keys with the RSA technique.	• Normalized routing load • Energy spent
[84]	Sangeetha et al.	2012	Proposed an energy efficient multipath routing protocol which provides better energy efficiency, security and reliability authentication. By using multipath routing and adjusting the transmission range of the path nodes, this protocol offers energy efficiency. Proposed protocol provides security through the use of Asymmetric key cryptographic algorithm RSA and MD5 hash function.	• Network lifetime • Throughput • End to end delay
[85]	Kumar et al.	2010	Proposed a novel approach for multipath and clustered sensor networks to increase network performance, resistance, and endurance. Proposed algorithm is also intended to address security, orphan node, and multi-hop difficulties in clustering protocols	• Throughput • Packet delivery ratio • Network lifetime
[86]	Tirki et al.	2019	Proposed a secure on-demand routing method that is resistant to attacks. The idea of k-connectivity is extended in the proposed approach to k-t-connectivity, where t is the disjointness threshold that indicates the maximum number of nodes shared by any two paths in the set of t constructed paths. The proposed technique also enriches multipath routing algorithms by enforcing a service-orientated threshold	• Throughput • Packet delivery fraction • End to end delay • Network life time

			during route identification, allowing for a trade-off amid fault tolerance and data transmission.	
[87]	Alrajeh et al.	2013	Proposed a secure routing protocol for WSNs based on the ant colonization technique. The mechanism used by Authors is based on forward ants, which collect and increment reputation values along the path. Similarly, the destination node employs backward ants, which carry information and instructions about route security from the destination node.	• End to end delay • Route discovery • Routing overhead • Data forwarding efficiency • Packet loss
[88]	Bok et al.	2016	Proposed an energy-efficient securing scheme for wireless sensor networks to improveenergy efficiency and security while minimizing energy consumption by restricting the decryption of exposed data even if transmitted data in a wireless sensor network is intercepted. In comparison to existing schemes, the proposed scheme showed lower communication costs and longer network lifetimes during the performance evaluation, while maintaining security.	• Energy consumption • Network lifetime
[89]	Salmi et al.	2022	Proposed a dataset for detecting wireless sensor networks generated by a computer. To detect and classify the attacks, the CNN-LSTM learning model was trained over 10 and 25 epochs with a 0.001 learning rate. The overall learning algorithm had a training accuracy of 96.57 percent, and the detection model successfully identified the five types of attacks	• Loss • Accuracy

			available.	
[90]	Premkumar et al.	2020	With the help of the proposed DLDM frame structure, the network is protected against DoS attacks. This lightweight detection mechanism, combined with deep learning for efficient detection and isolation, is used for scalable and energy-efficient cluster-based countermeasures. The simulation results and performance analysis of the DLDM and MAS approaches show that the proposed DLDM frame structure has the best results in terms of detecting DoS attacks	• Energy consumption • Throughput • Packet delivery ratio
[91]	Sahoo et al.	2013	Proposed a pragmatic energy model for WSNs which is trust-based secure and energy-efficient clustering method. . Authors also added the most desirable TRUST mechanism to choose an appropriate cluster head, which prevents malicious nodes from becoming cluster heads.	• Number of alive nodes • Average energy consumption • Average residual energy
[92]	Pajila et al.	2022	Proposed a novel method for detecting Flooding attacks and redirecting data packets to the sink via a different path. The Proposed method examines each sensor's energy consumption, response time, and data packet count. To detect the occurrence of a DDoS attack, the proposed method employs a type 1 fuzzy-based rule. The research investigates the proposed method's energy efficiency by looking at buffer usage, packet drop rate, response time, and network lifetime	• Buffer usage • Packet drop rate • Response time • Network lifetime
[93]	Prem Kumar et al.	2021	Proposed algorithm is used to protect the network from DoS attacks. A lightweight	• Throughput • Energy consumption

			detection mechanism is described, as well as deep learning, for efficiently detecting and isolating DoS attacks in this paper. Simulation findings and performance analysis show that the proposed system can achieve the best results in terms of detecting DoS attacks.	• Control overhead • Delay •
[94]	Raghav et al.	2020	Proposed a Beeware routing scheme that is simple and effective in dealing with routing attacks like flooding, spoofing, and Sybil. This method is used to show which path is the most efficient for transporting data to the target node. Beeware routing scheme includes a security mechanism that is compared to other mechanisms.	• Routing overhead • Data forwarding efficiency • Path discovery efficiency • End to end delay • Packet loss • Packet delivery ratio
[95]	Opinder Singh et al.	2017	Proposed a novel model to detect and isolate the route request Flooding attack. Any node that transmits more RREQs than the threshold is flagged as an intruder node, and an alarm is issued to isolate that node. Proposed algorithm's results revealed a high throughput that was comparable to the built-in AODV and a short delay that was also comparable to the built-in AODV.	• Packet loss • Throughput • Number of packets • Delay
[96]	T. Pandikumar et al	2017	Suggested a model to protect against the route request Flooding assault. Each node in the proposed approach records the number of requests made and received in order to compute the average of route request packets, which is utilized to determine the limit. This model reduces the Packet Loss Ratio in two independent scenarios when compared to	• Average throughput • End to end delay

			the built-in AODV under assault.	
[97]	Jatthap et al.	2016	Suggested an energy-based approach for detecting and isolating RREQ Flooding aggressor nodes. The proposed method evaluates an energy consumption of the node before and after the attack. The node is considered dead if its energy is equal to or less than the minimal energy threshold and then isolated from the network.	• Throughput • Packet delivery ratio • End to end delay
[98]	Kumar et al.	2015	Created a technique to protect MANETs from RREQ attacks. There are three lists for each node: whitelist, graylist, and blacklist. When a node receives a request, it looks through these three lists to find the sender. The request is dropped if the sender is on the blacklist.	• Normalized routing load • Throughput • Packet delivery ratio • Accuracy detection
[99]	Gurung et al.	2018	Suggested a novel method for mitigating RREQ Flooding attacks in MANETs. The proposed algorithm is alienated into three stages: dynamic threshold computation, authorization, and rearranging. The algorithm will put malicious node as a blocked node and further communication will be debarred.	• Packet delivery rate • Packet loss rate • Average throughput • Routing overhead • Normalized routing load
[100]	Singh et al.	2012	Proposes a protocol, in which each node's traffic history is updated each time the node collects the request packet. The algorithm	• Throughput • Packet delivery ratio

			calculates the vale Threshold Reliable to categorize a node as a malicious node.	
[101]	Panos et al.	2014	This suggested work is founded on a finite state mechanism, wherein the states are characterized as either legitimate or malignant, and transfers from one state to another are activated by individual node activity.	• Packet delivey ratio • Detection accuracy • Control packet overhead
[102]	Abdelshafy et al.	2014	presented a technique in algorithm calculates the number of route request received at an interval of 10 seconds and checks if its total value is less than the threshold value. If the number of route request packets generated by a particular node is less than the threshold value then we can decrease its blacklist index by 10 otherwise its blacklist index will be increased by 5 and node is regarded as a malicious node.	• Send data packets • Packet delivery ratio • Routing overhead • End to end delay • Routing discovery latency • Normalized routing load
[103]	Jiang et al.	2013	Suggested an RREQ Priority Assessment index system to limit the consequence of flooding assaults in the AODV routing protocol by employing dual threshold values, Min and Max Threshold. The router node retains packets and sends them according to the request rate limit threshold policy for a node with acceptable priority.	• Total number of route request packets • Power consumption

[104]	Faghihniy et al.	2017	Suggested a new method which uses a balanced indexed value for each node to confirm whether to accept or reject a route request. There is no verification methodology in this plan that can clearly classify a node to be a malicious node and tends to drop the genuine route request of a node.	• Number of RREQ • Packet delivery ratio • Overhead • End to end delay • Throughput

2.5 Limitation of existing techniques identified from review of literature

A number of authors have proposed a number of techniques for controlling congestion, fairness control and mitigating network attacks. Following are the various limitation of existing proposals identified from literature survey.

- Most of the algorithms/works presented in the literature consider only single type of traffic i.e. either scalar or multimedia traffic. But in real world scenario's we have nodes which can generate various types of traffic.
- Most the algorithms/works present in the literature takes one of parameters among buffer occupancy, bandwidth, residual energy, distance etc. for calculating the best path from source to destination path. A collection of all parameters have not considered.
- Most of the algorithms/woks in the literature considered only basic Ant colony optimization algorithm and simulated the network. The simulated results are compared with the existing well-known techniques. But modifying the Ant colony optimization by tuning the parameters was missing in WSNs.

- Most of the work presented in the paper uses the single path for routing the same type of traffic. Present work in the literature do not use multi path routing while considering the congestion and optimal path routing.
- Most of the techniques from the literature focus on controlling the congestion by only reducing the data rate from the source or intermediate nodes. They do not focused on the matter that data rate can also be increased if congestion is alleviated.
- Generally, most of the algorithms considers only input and output traffic rate decide the flow rate of data. And by considering the difference of them is used to decide the reduced rate of traffic. They have not considered the other parameters like bandwidth and active flow rates associated with the nodes.
- Most of the proposal has focused on flooding attacks but they have not considered the aspect of congestion in it.
- Most of the proposal has not used the statistical analysis for identifying the malicious node in the network. They used the non-statistical methods in their proposals.

CHAPTER 3

An intelligent path evaluation algorithm for congestion control in WSNs

3.1 Introduction and Problem statement

WSNs are simple computers with a variety of devices mounted on them that are used to configure environmental as well as physical data such as pressure, sound, orientation, vibration, and temperature, amongst many other factors. The input obtained from the sensors is redirected from one sensor to another via a multi-hop routing protocol to access the sink node. The base station is in possession of the results aggregation and analysis. Sensor nodes in WSNs have limited resources in terms of power consumption, resources, storage, bandwidth, processing capabilities, and so on. Initially, the researchers managed to come up with a routing scheme that would enable them to make efficient use of WSNs resources. Nodes are mostly self-organized and connect with each other via wireless networks, which are implemented throughout the WSNs on an ad hoc basis [105]. The main obstacles amongst the other possible problems is that the routing mechanism of the WSNs can be compromised, culminating in inevitable losses due to network congestion.

Congestion is the most prominent issue in WSNs that adversely affect the network QoS parameters, must be handled properly. When the multiple sensor nodes start sending the data simultaneously then the data generated at intermediate nodes increases. As a result, the number of data packets and generated data from the source nodes exceeds the underlying capacity of the network queue and at the same time makes the networks underperforming due to congestion. In WSNs, we can have a variety of causes for congested networks, such as overloading of the queue, collision of packet with each other, parallel transfer of data packets sharing the bandwidth etc.

In real-world scenarios, there is a number of optimization problems exists which are difficult to solve and categorized as NP-hard category problems. The two basic categories of techniques that can be used to solve such kind of problems are: Providing the actual algorithm and the approximate results algorithms which optimally generate the approximate results through a sequence of iterations. The main difference between the two techniques is that the former one will provide us with the optimal solutions with more computation time, but the later one will use less computation time for generating the optimal results for NP-hard problems.

The routing problem based on Ant Colony Optimization (ACO) is a category of NP-hard approach that is implemented in WSNs in particular. ACO is an algorithm based on swarm intelligence, motivated by the discovery behavior of ant colonies present in the natural environment, and is a powerful technique to fathom combinational optimization algorithms.

So from the above discussion and review of literature presented in chapter 2, It can be concluded that there must be some routing algorithm in WSNs for sending the data from source to destination without congestion that can consider the type of traffic along with the routing in the event of congestion. The Ant Colony Optimization used by number of authors in the literature suggests that a number of techniques can achieve optimal route from source to destination using the said algorithm in the event of congestion. However there are various constraints and challenges exits in the current literature review which are listed below:

Various constraints and challenges in the existing literature are

- Most of the algorithms/works presented in the literature consider only single type of traffic i.e. either scalar or multimedia traffic. But in real world scenario's we have nodes which can generate various types of traffic.
- Most the algorithms/works present in the literature takes one of parameters among buffer occupancy, bandwidth, residual energy, distance etc. for

calculating the best path from source to destination path. A collection of all parameters have not considered.

- Most of the algorithms/works in the literature considered only basic Ant colony optimization algorithm and simulated the network. The simulated results are compared with the existing well-known techniques. But modifying the Ant colony optimization by tuning the parameters was missing in WSNs.
- Most of the work presented in the paper uses the single path for routing the same type of traffic using Ant Colony Technique. Present work in the literature do not use multi path routing while considering the congestion and optimal path routing.

So in order to address above said constraints and challenges and for handing the congestion in the network a novel algorithm have been proposed that will consider the importance of the traffic like real time and non-real time traffic. In the event of congestion, proposed algorithm will route the packets belonging to real time traffic (audio/video traffic) from best path evaluated by the algorithm. There are lot of work has been done in evaluating the best path among various available paths in the network from source to destination. There are number of intelligent techniques has been used for evaluating the best path including well know Ant colony optimization. So we decided to work upon it and modified the existing algorithm of Ant colony optimization according to our network parameters.

However, most of the techniques are either using residual energy or the distance between the node and its destination is used to measure the likelihood of packet transmission using a neighbor node. So, in order to design an efficient congestion control algorithm, residual energy of the neighbor node, buffer occupancy and bandwidth allocated for calculating the probability of the next neighbor node for selection have been taken. The proposed algorithm has been divided into 2 two modules

- Congestion detection
- Path evaluation

The proposed algorithm has been explained in more detail in section 3.4.

In order to consider the factor of congestion in the routing algorithm, a novel algorithm for routing the information from optimal path has been designed. The main contributions of the proposed work are as follows.

- To evaluate the various paths available for routing the information from neighbor node to the next node. Path evaluation will be done by calculating the path preference probability of the neighbor nodes using updated Ant colony optimization probability function and selecting the neighbor node with maximum path preference probability.
- To consider the residual energy of neighbor node, bandwidth allocated and buffer occupancy for calculating the path preference probability.
- In the event of congestion, route the traffic belonging to real time data (audio/video traffic) from best calculated path.
- To simulate the proposed algorithm by using NS-2.33 simulator and analyzing the results on network parameters average throughput, average energy consumption, average packet loss rate and average end to end delay. Finally the results of proposed algorithm with existing state of art algorithms i.e. Antsense [115], FCC [52] and CODA [62] algorithm have been compared.

In the next sections the conventional Ant Colony Optimization algorithm and proposed algorithm have been explained. Later in this chapter, the proposed algorithm has been simulated and compared with existing state of art algorithms which are Antsense, FCC and CODA algorithm

3.2 Working of Traditional Ant Colony Optimization Algorithm

Met heuristic is a collection of dedicated rules that can be utilized to establish heuristic procedures that apply to a large variety of different optimization problems. [106] has formalized ACO as a met heuristic for mixture optimization problems. ACO takes inventiveness from the provender practices of ant species who inject pheromones

on the ground to signify any suitable routes that other members of the colony should follow. To solve problems of optimization, ACO uses a similar mechanism. An individual sensor is insect type species with minimal memory requirement and involved in executing simple ventures. The colony communicates a diverse collective activity that provides an intelligent process to the problems such as moving heavy items, building bridges and discovering the quickest pathways from the nest to the source of food, since a unique insect has no broad understanding of the function it plays and its activities are based on local settlements and are highly irregular. This intelligent behavior of course occurs as a result of self-sustaining and incidental communication between ants, which can be referred to as [106] [107] swarm intelligence.

In the basic part of ACO algorithm the process iterates through three steps after initialization: ConstructAntSolutions, ApplyLocalSearch algorithm, and UpdatePheromones process. A series of surrogates are established by ants that are optimally improved by a local search algorithm and the update pheromone process. In view of this formulation, by moving simultaneously and asynchronously to a sufficiently defined graphical representation that represents any given problem domain, ACO ants produce solutions to the problem. Ants select the next neighbor hop node j from a node i according to the probability formula defined given below [106]:

$$P_{i,j} = \frac{[T_{i,j}]^{\alpha} * [D_{i,j}]^{\beta}}{\sum_{n \in N}[T_{i,j}]^{\alpha} * [D_{i,j}]^{\beta}} \quad 3.1$$

Where $P_{i,j}$ is the probability of ant selecting the next node j from node i. $T_{i,j}$ is the concentration of pheromone in the path i to j. N indicated the number of next neighbor nodes that an ant can select. α indicates the weight of the concentration of pheromone. $D_{i,j}$ is the distance of next neighbor node from i node and β indicates the weight of the distance.

WSNs routing is a difficult problem since the features of the network, such as traffic loading and network topology, can differ stochastically and in a time of varying nature. Traffic- adaptive and multipath routing is given by the collection of unique properties that characterizes ACO objects for routing problems, based on both passive and active data tracking and collection, using hypothetical units, not allowing local processes to have a global effect, setting up routes in a less greedy way than in strictly shortest path techniques preferring balancing of network load [106].

3.3 Proposed Algorithm: Congestion Detection and Path Evaluation Algorithm - CDPEA

In the suggested algorithm, path evaluation will be done using the intelligent behavior of ANT colony optimization algorithm and congestion will be controlled, if occurs, by routing the different type of traffic from different paths based upon the goodness of paths available. Congestion in the network will be detected at the node level by checking the buffer occupancy of the node. The proposed algorithm is based on devising the algorithm for evaluating the best path for real-time traffic like audio/video using an updated Ant colony optimization algorithm. For this purpose algorithm will identify if there is congestion in the network by checking the buffer occupancy of the neighbor node for transferring the packets. If buffer occupancy falls under the situation that can lead to congestion, we need to identify the packets belonging to different traffic types like audio, video or scalar traffic. The packets belonging to real-time traffic will be routed through the best node among the neighbors. In sensor networks, there can be variety of sensor nodes generating the data packets which may belong to scalar as well as real-time traffic. The chief impartial of the proposed algorithm is to route the real time packets through the nodes having maximum path preference probability.

The proposed algorithm will work in two different models:

3.3.1 Congestion prediction model

The main idea behind congestion prediction model is to detect the congestion level well in advance so that effective action can be taken. If congestion is predicted in the network we will re calculate the optimized path from source to destination and will send the packets belonging to real time traffic from calculated optimized path. In order to predict the congestion we will use two parameters i.e. node's queue occupancy and threshold value of the queue for that node. The queue occupancy of the node x at time t is given by the following mathematical equation:

$$Queue_x(t+1) = Queue_x(t) + I_x(t) - O_x(t) \quad 3.2$$

Where $I_x(t)$ is the incoming traffic rate of the node x at time t, $O_x(t)$is the outgoing traffic rate of node x.

We will calculate the threshold queue value of the node x by using the following mathematical equation:

$$T_x\,(t+1) = \frac{O_x\,(t)}{I_x\,(t)} * \left[Max_{queue_{size\_x}} - Queue_x\,(t)\right] \quad 3.3$$

Where $Max_{queue_{size\_x}}$ is the maximum occupancy of the node that it can accommodate, $T_x\,(t+1)$ is the threshold value of the node. Basically threshold value is inversely proportional to the incoming rate of traffic in the node and directly proportional to the outgoing traffic of the node. So the parameter $T_x\,(t+1)$ is the desired level of queue occupancy and any node having queue occupancy greater than this will lead to congestion.

Now algorithm will predict the congestion according to the following conditions:

- Algorithm will conclude that there is no congestion in the node is $Queue_x(t+1) < T_x(t+1)$. It means queue of the node is empty and there are no chances of congestion.
- Algorithm will conclude there is a congestion in the node if $Queue_x(t+1) >= T_x(t+1)$. It means the queue occupancy of the node reaches the $Max_{queue_{size\_x}}$. In this case we can say the congestion is detected and there is a need to apply the proposed algorithm.

From the above procedure we have concluded that when the network detects the congestion and once the congestion is detected it need to optimized path to route the traffic which belongs to real time data who's QoS is quite stringent. It has been discovered that real time (RT) traffic demands low latency and good dependability, and so must be prioritized.

3.3.2 Path evaluation model

As in congestion prediction model we have formulated the process of detecting the congestion. When there is congestion, the proposed technique computes the path preference probability of all adjacent nodes from which data might be routed. The path preference probability will be calculated by considering the residual energy, bandwidth and buffer occupancy of the neighbor nodes and node with the highest path preference probability will be chosen to route the information. The model will primarily follow following steps:

- Identify all the neighbor nodes which are likely to participate in the routing.
- Calculate the path preference probability of the neighbor nodes. The process will be repeated for all the nodes in the network.

In order to calculate the path preference probability, we will be using the mathematical model illustrated below. The path preference probability from node i to neighbor node j can be calculated using the following mathematical function:

$$PPF_{i,j} = \frac{[E_{i,j}]^{\alpha} * [B_{i,j}]^{\beta} * [BO_{i,j}]^{\gamma} * [T_{i,j}]^{\theta}}{\sum_{n \in N}[E_{i,j}]^{\alpha} * [B_{i,j}]^{\beta} * [BO_{i,j}]^{\gamma} * [T_{i,j}]^{\theta}} \quad 3.4$$

Where $PPF_{i,j}$ is the path preference probability of ant selecting the next node j from node i. N indicates the number of next neighbor nodes that an ant can select. $E_{i,j}$ is the energy of next neighbor node j from i node and α indicates the weight of energy. $B_{i,j}$ is the bandwidth of next neighbor node from i node to node j and β indicates the weight of bandwidth. $BO_{i,j}$ is the buffer occupancy of the node j and γ indicates the weight of buffer occupancy. $T_{i,j}$ is the accumulation of pheromone in the path i to j and θ indicates its weight.

The proposed algorithm has also been listed in the following algorithm:

Algorithm 3.1 : *Congestion detection and path finding*

1: Initialize the network of n nodes.
2: Initiate the route discovery process using underlying algorithm.
3: For all the nodes in the network:
Calculate the buffer occupancy of each node using the following equation:

$$Queue_x(t+1) = Queue_x(t) + I_x(t) - O_x(t)$$

4: Calculate the threshold value for all the nodes in the network using the following formula:

$$T_x\,(t+1) = \frac{O_x\,(t)}{I_x\,(t)} * \left[Max_{queue_{size\_x}} - Queue_x\,(t)\right]$$

5: Check the congestion status:

If ($Queue_x(t+1) < T_x\,(t+1)$) then:
No congestion.
Route the packets through decided path according to traditional algorithm.

Else if ($Queue_x(t+1) >= T_x\,(t+1)$) then:
Classify the packet into real time and non-real time traffic.
Calculate the path preference probability using the following equation:

$$PPF_{i,j} = \frac{[E_{i,j}]^{\alpha} * [B_{i,j}]^{\beta} * [BO_{i,j}]^{\gamma} * [T_{i,j}]^{\theta}}{\sum_{n \in N}[E_{i,j}]^{\alpha} * [B_{i,j}]^{\beta} * [BO_{i,j}]^{\gamma} * [T_{i,j}]^{\theta}}$$

If(packet_type==”RT”)
Send the packets belonging to real-time traffic by selecting the node with highest path preference probability among the neighbor nodes.

Else
Send the packet through the path decided by underlying algorithm.

6: Repeat the steps 2 to 5 until simulation ends.

3.4 Complexity Analysis of Proposed algorithm:

Complexity of an algorithm is defined in terms of time and space. The amount of time and space required by an algorithm is very crucial with respect to evaluation of algorithm. Out of time and space complexity we have focused on evaluating the time complexity of proposed algorithm.

The proposed algorithm has been divided into two modules:

- Congestion detection module
- Path evaluation module.

3.4.1 Time complexity of congestion detection module:

In congestion detection module, we have to calculate the congestion status of all the nodes in the network so we have to run the loop for all the nodes in the network. If there are 'n' number of nodes in the network then the time complexity of congestion detection module will be O(n) in both best as well as worst case. The same thing has been summarized in the following table.

Table 3.1: Time complexity of congestion detection module

Time Complexity of Congestion detection module	
Best Case	Worst Case
O(n)	O(n)

3.4.2 Time complexity of Path Evaluation Module.

The time complexity of path evaluation module depends upon the congestion status in the network. If there is no congestion in the network then algorithm will send the traffic whether real time or non-real time through a single dedicated path. But the algorithm has to consider all the nodes in the network for finding the path from source to destination, which is commonly known as route discovery. In this case, the time required to find the path will be a function of 'n' number of nodes in the network. So the time complexity will be O (n). When there is no congestion in the network, then we can consider it as a best case.

On the other hand, if there is congestion in the network, then algorithm has to work in nested loop. In the outer most loop, algorithm has to take care of all the

nodes in the network denoted by ‘n’ and in the inner most loop the algorithm has to consider the neighbors of all the nodes in the network denoted by ‘m’ so time complexity of the module will be O(n*m). When there is congestion in the network then algorithm has to rigorously find the optimal path and can be considered as worst case. The time complexity of the module is shown in the following table.

Table 3.2: Time complexity of Path Evaluation module

Time Complexity of Congestion detection module	
Best Case	Worst Case
O(n)	O(n*m)

The flowchart of the proposed algorithm is shown in figure 3.1

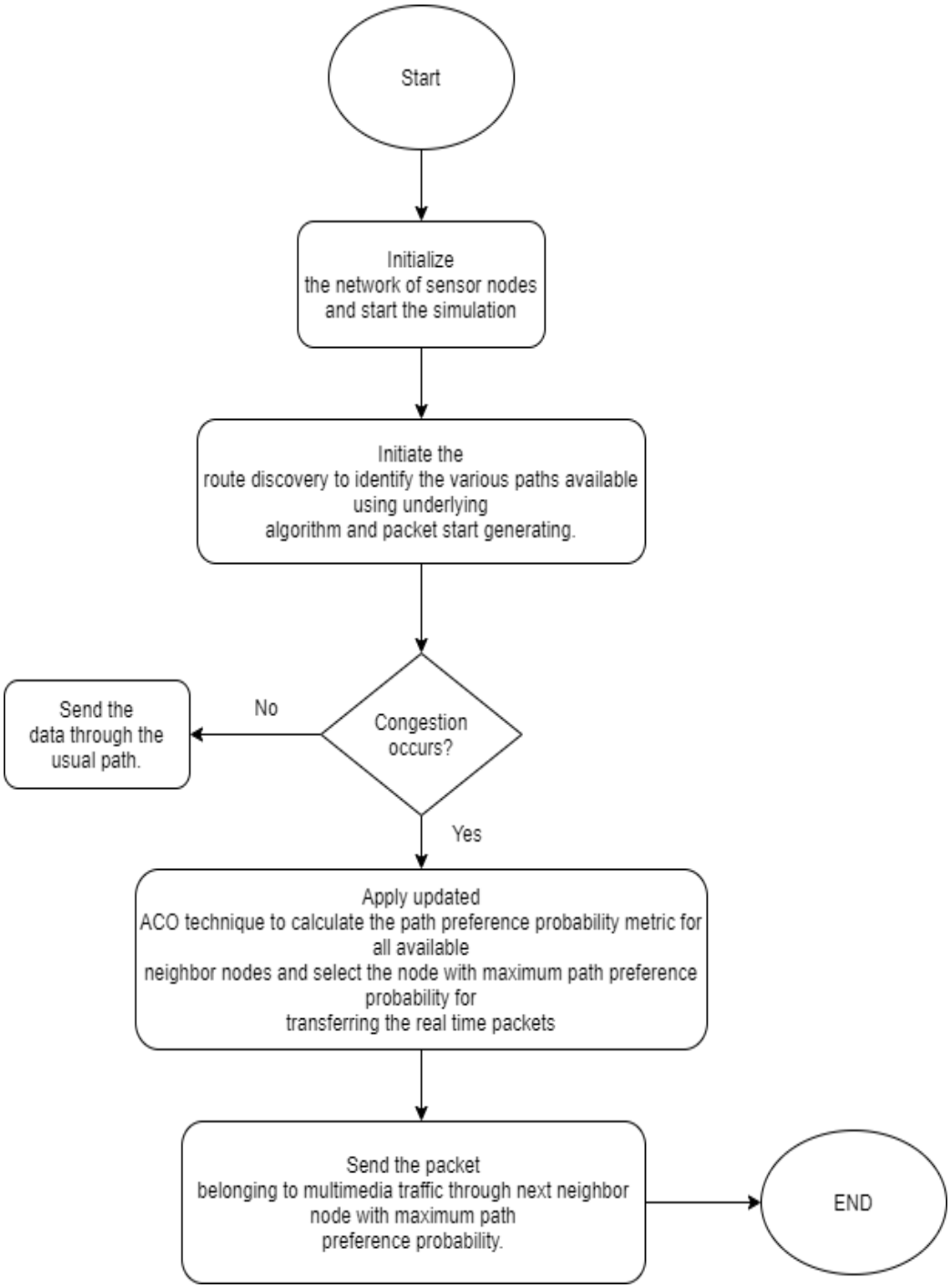

Figure 3.1 Flowchart of the proposed algorithm

3.5 Simulation and results

Proposed algorithm using updated ant colony optimization algorithm have been simulated by using a well-known simulator NS-2.33 by varying number of sensor nodes as well as network simulation time. The proposed algorithm has been compared with existing state of art algorithms i.e. Basic AntSense, FCC and CODA algorithms from review of literature. A number of extensive simulations have been performed to check the efficiency of the developed algorithm. Nodes are put randomly in a 100 m x 100 m zone in all scenarios. Each mobile node's maximum transmitting range is 100 m. Static nodes are called the sensor nodes that have been simulated on the network. The channel bandwidth is 2 Mbps and the IEEE 802.11 the MAC layer. Each simulation runs for 100 seconds at a time. A variety of constant bit rate (CBR) sources with multiple sources for the development of multiple traffic types and with a single destination have generated data traffic. Various parameters taken for the network simulation is listed

Simulations results have been divided into two parts

- Simulations by varying number of nodes
- Simulation by varying the simulation time.

Table 3.3: Simulation parameters and rules

Simulation Scenario and parameters	Corresponding value
Wireless propagation model	TwoWay around
Node's Transmission range	100m
Queue Type	PriQueue
Node's initial energy	90J
Packet's Size	1024 Bytes
Number of nodes (n)	10,20,30,40,50
Simulation time (s)	50,100,150,200,250
Antenna Pattern Used	OmniAntenna

3.5.1 Simulation results by varying the number of nodes:

In this section, the proposed algorithm, AntSense, FCC and CODA algorithm by varying the number of nodes have been simulated. The comparative study among them is discussed. The proposed algorithm has been evaluated on following parameters.

- Average Energy Consumption Analysis
- Average Throughput Analysis
- Average Packet loss rate Analysis
- Average End to End delay Analysis

3.5.1.1 Average Energy Consumption Analysis

The figure 3.2 represents the average energy consumption at the end of simulation and it has been observed that the proposed algorithm shows less energy consumption as compared to AntSense, FCC and CODA algorithm. This is due to the fact that the proposed algorithm uses residual energy, bandwidth and buffer occupancy combination for selecting the next best hop for packet transmission. On the other hand , FCC consider only delay and signal strength for selecting the next hop, AntSense uses only energy consumption for selecting the next hop and CODA do not use any optimization technique for selecting the next hop. As shown in the figure extensive simulations have been performed by varying the number of nodes in the network and in most of the scenarios the proposed algorithm is showing the positive result. On an average, the proposed algorithm consumes 1.24%, 6.54% and 12.63% less energy compared with FCC, AntSense and CODA algorithm, respectively. Table 3.4 summarizes results shown in figure 3.2.

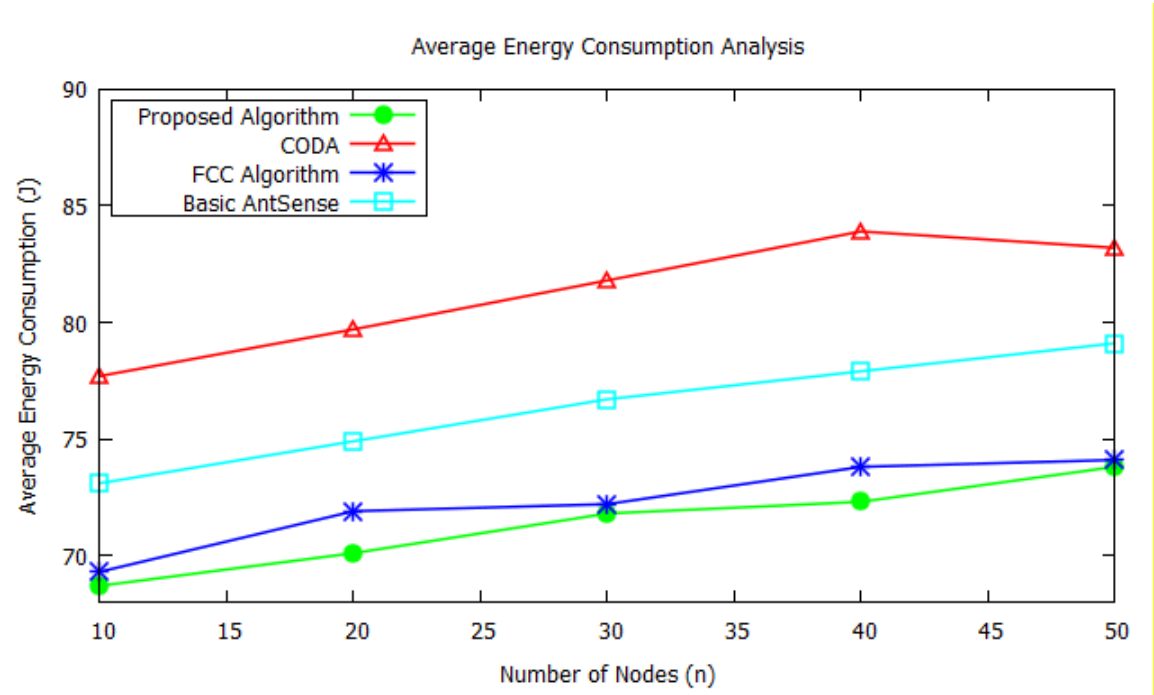

Figure 3.2: Average energy consumption of proposed algorithm, AntSensse, FCC and CODA algorithm versus number of nodes

Table 3.4 Average Energy consumption of proposed, Antsense, FCC and CODA algorithm

Average Energy Consumption (J)				
Number of nodes	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
10	68.7	69.3	73.1	77.7
20	70.1	71.8	74.9	79.7
30	71.8	72.2	76.7	81.8
40	72.3	73.8	77.9	83.9
50	73.8	74.1	79.1	85.2

3.5.1.2 Average Throughput Analysis

The figure 3.3 represents the average throughput obtained at the end of simulation and it has been observed that the proposed algorithm shows higher throughput as compared to AntSense, FCC and CODA algorithm. This is due to the fact that the FCC consider only delay and signal strength for selecting the next hop and do not consider energy, bandwidth and buffer occupancy for selecting the next hop. AntSense consider only energy consumption and CODA do not apply any optimization parameters. As shown in the figure 3.3 extensive simulations have been performed by varying the number of nodes in the network and in all the scenarios the proposed algorithm is showing the positive result. Simulation results show that the proposed algorithm shows 1.24%, 12.45% and 22.75% increase in average throughput compared with FCC, AntSense and CODA algorithm. Table 3.5 summarizes the results obtained in tabular form.

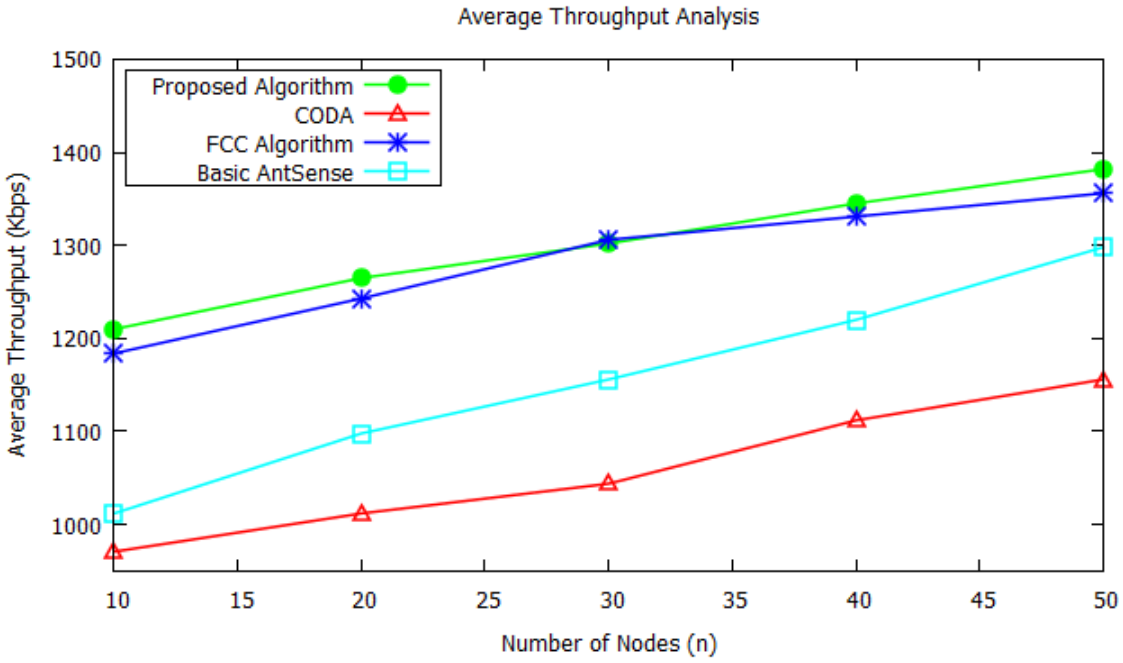

Figure 3.3: Average Throughput analysis versus number of nodes for proposed algorithm, AntSense FCC and CODA algorithm

Table 3.5 Average Throughput analysis of proposed, basic AntSense, FCC and CODA algorithm

Average Throughput Analysis (Kbps)				
Number of nodes	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
10	1210	1184	1012	971
20	1265	1243	1098	1012
30	1302	1306	1156	1044
40	1345	1331	1220	1112
50	1382	1356	1298	1156

3.5.1.3 Average Packet Loss Rate Analysis

The Packet loss ratio (PLR) is one of the important metric for measuring the performance of traffic related to real-time flows such as video and Voice over IP.However, as shown in figure 3.4, the proposed algorithm has lower average packet loss ratio than that of AntSense, FCC and CODA algorithm. This is due to the fact the proposed algorithm uses the blend of parameters required for QoS data transfer for selecting the next hop in contrast to FCC, AntSense and CODA which takes either few parameters or no parameters for selecting the next hop. Simulation results confirms that the proposed algorithm shows 11.68%, 13.9% and 17.07% decrease in packet loss ratio compared to FCC, AntSense and CODA algorithm figure 3.4. Table 3.6 summarizes the results obtained in tabular form.

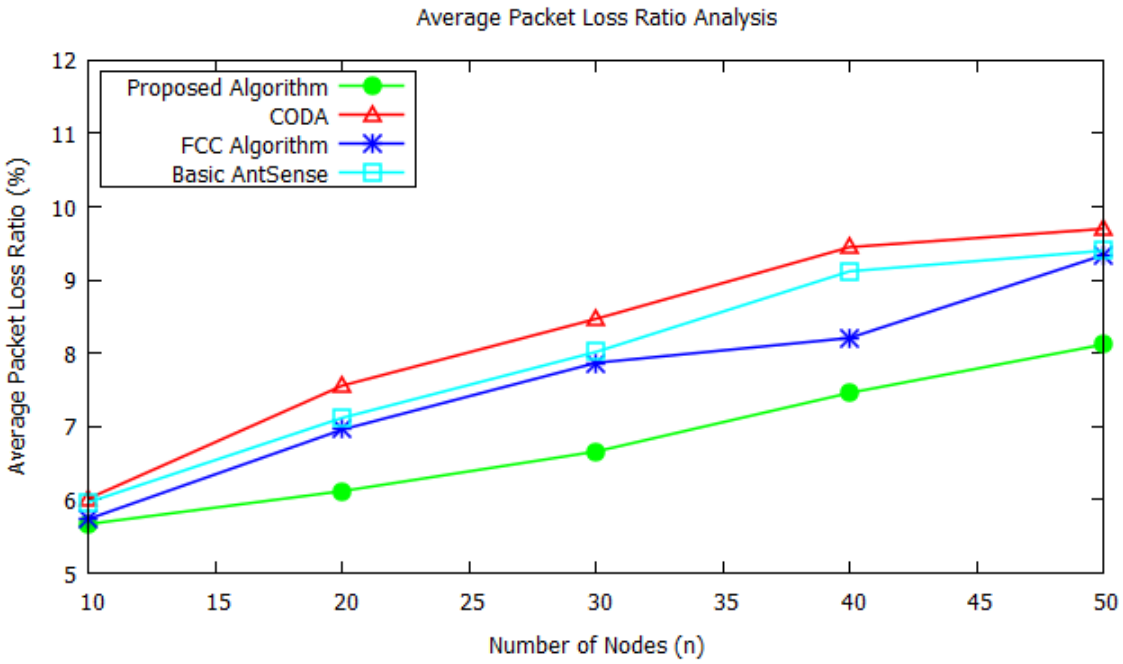

Figure 3.4: Average Packet loss rate analysis versus number of nodes for proposed, AntSense, FCC and CODA algorithm

Table 3.6: Average packet loss ratio of proposed, AntSense, FCC and CODA Algorithm

Average Packet Loss Rate Analysis (%)				
Number of nodes	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
10	5.67	5.74	5.97	6.02
20	6.12	6.96	7.12	7.56
30	6.66	7.87	8.02	8.47
40	7.46	8.21	9.12	9.45
50	8.12	9.78	9.40	9.70

3.5.1.4 Average End to End Delay:

The average end to end delay is one of the important metric for measuring the performance of a network. The average end to end delay must be low in order to prove the smooth and seamless data transmission. The proposed algorithm has lower average end to end delay than that of the AntSense, FCC and CODA algorithm. This is due to the fact that the proposed algorithm is capable of selecting the next best hop by considering the blend of various parameters required for good QoS requirements.Simulation results confirms that the proposed algorithm shows 11.36%, 57% and 70% decrease in average end to end delay compared with FCC, AntSense and CODA algorithm. The simulation results are shown in the figure 3.5. Table 3.7 summarizes the results obtained in tabular form.

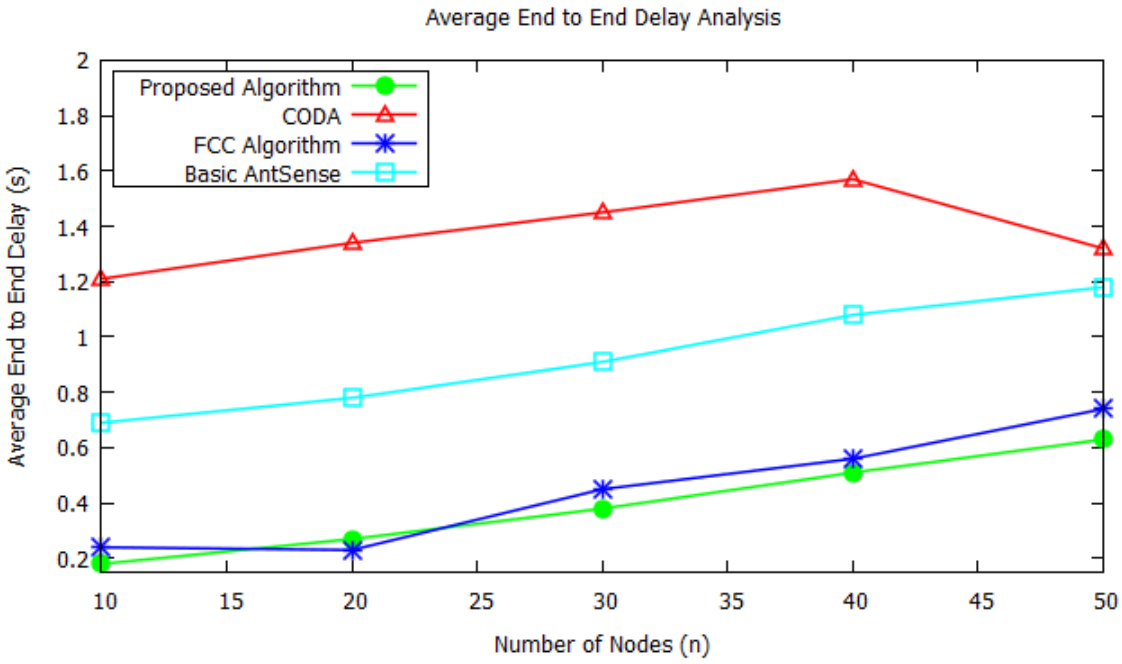

Figure 3.5: Average End to End delay analysis versus number of nodes for proposed, Basis AntSense, FCC and CODA algorithm

Table 3.7: Average End to End delay of proposed, AntSense, FCC and CODA algorithm

Average End to End Delay Analysis (ms)				
Number of nodes	**Proposed Algorithm**	**FCC Algorithm**	**Basic AntSense Algorithm**	**CODA Algorithm**
10	0.18	0.24	0.69	1.21
20	0.27	0.23	0.78	1.34
30	0.38	0.45	0.91	1.45
40	0.51	0.56	1.08	1.57
50	0.63	0.74	1.18	1.32

3.5.2 Simulation results by varying the simulation time.

In this section we have simulated the proposed algorithm, AntSense, FCC and CODA algorithm by varying the simulation time and proposed algorithm has been compared with existing state of art algorithms i.e. AntSense, FCCand CODA algorithm. The proposed algorithm has been evaluated on following parameters.

- Average Energy Consumption Analysis
- Average Throughput Analysis
- Average Packet loss rate Analysis
- Average End to End delay Analysis

3.5.2.1 Average Energy consumption versus simulation time analysis

As shown in figure 3.6, we have compared the result of proposed algorithm with basic AntSense, FCC and CODA algorithms by varying simulation time. It has been observed from the simulations results energy consumption in the network increases with increase in simulation time. However energy consumption of proposed algorithm is near about same as that of FCC when simulation time is less but with the increase of simulation time, the proposed algorithm shows less energy consumption when compared to other algorithms. Simulation results confirms that proposed algorithm shows 4.47%,

7.72% and 11.77% decrease in energy consumption compared with FCC, AntSense and CODA algorithm respectively. The result of simulations has also been tabulated in table number 3.8.

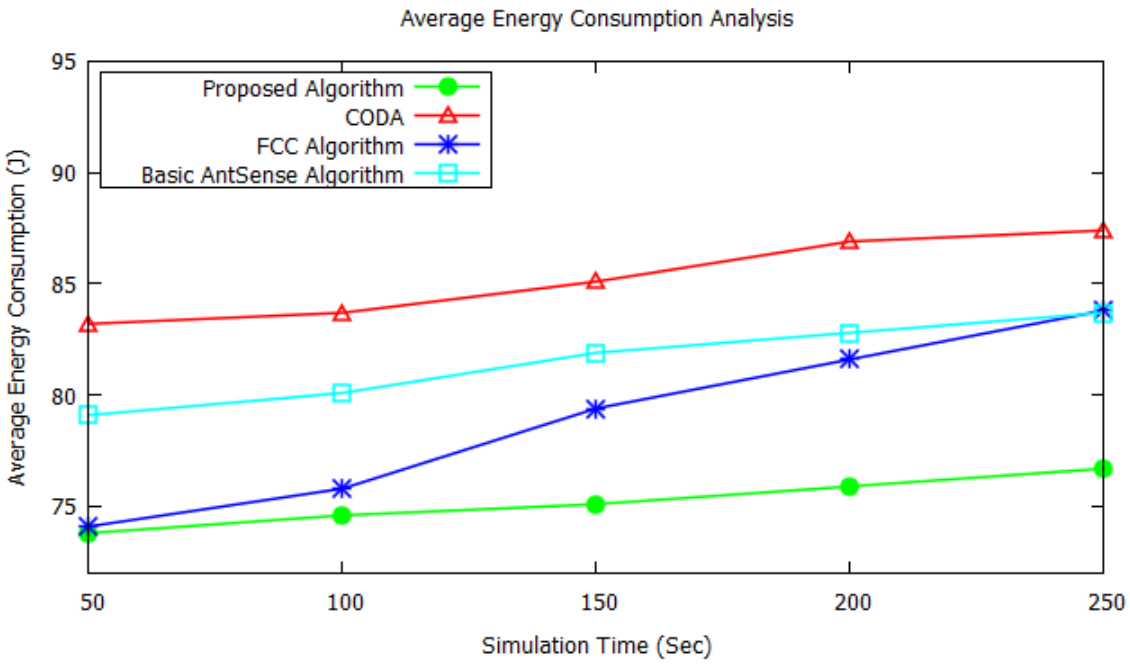

Figure 3.6: Average Energy Consumption versus simulation time for proposed, Basis AntSense, FCC and CODA algorithm

Table 3.8: Average Energy consumption of proposed, FCC, AntSense and CODA algorithm versus summation time

Average Energy consumption Analysis (J)				
Simulation Time	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
50	73.8	74.1	79.1	83.2
100	74.6	75.8	80.1	83.7
150	75.1	79.4	81.9	85.1
200	75.9	81.6	82.8	86.9

250	76.7	82.8	83.7	87.4

3.5.2.2 Average Throughput versus simulation time analysis

As shown in figure 3.7, the result of proposed algorithm with basic AntSense, FCC and CODA algorithms by varying simulation time have been simulated. It has been observed from the simulations results that the proposed algorithm outperforms in terms of average throughput. The proposed algorithm shows more throughput than that of the other existing state of art algorithms taken for comparison. But one more analysis came from this graph is that, with increase of simulation time the average throughput is nearby same as that of proposed algorithm and FCC during the less simulation time. Simulation results shows that the proposed algorithm shows 2.66%, 6.68% and 14.56% increases in throughput compared with FCC, AntSense and CODA algorithm on an average. The result of simulations has also been tabulated in table number 3.9.

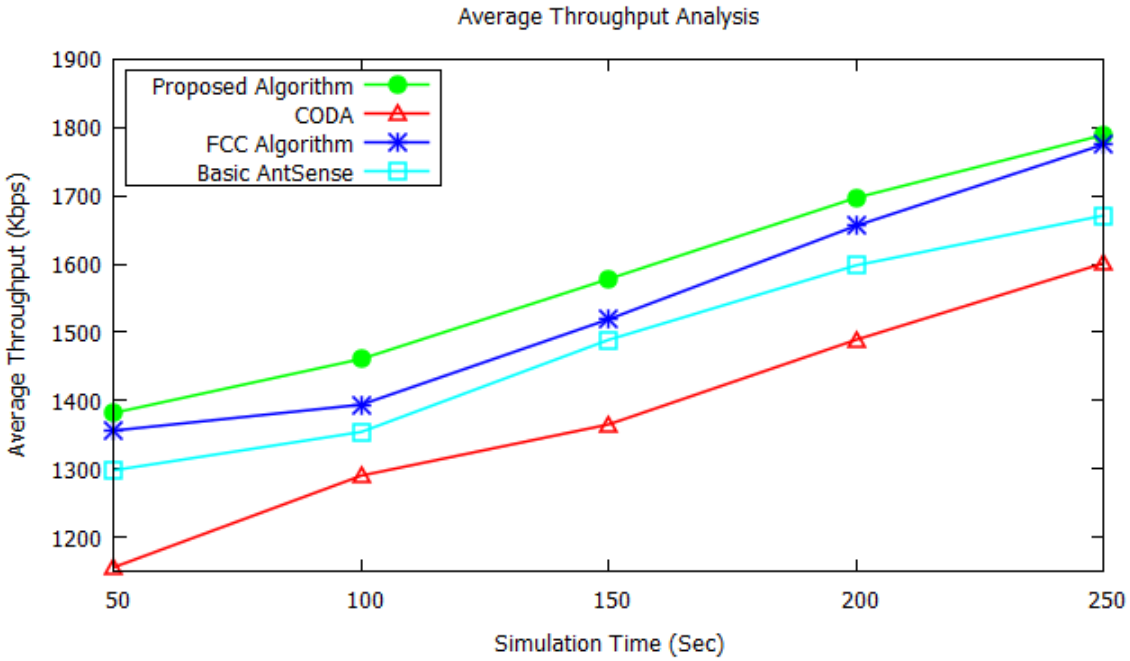

Figure 3.7: Average Throughput versus simulation time for proposed, AntSense, FCC and CODA algorithm

Table 3.9: Average Throughput consumption of proposed, AntSense, FCC and CODA algorithm versus summation time

Average Throughput Analysis (Kbps)				
Simulation Time	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
50	1382	1356	1298	1156
100	1461	1394	1354	1290
150	1578	1519	1489	1365
200	1697	1656	1598	1489
250	1789	1775	1671	1601

3.5.2.3 Average Packet loss rate versus simulation time analysis

As shown in figure 3.8, we have compared the result of proposed algorithm with basic AntSense, FCC and CODA algorithms by varying simulation time. It has been observed from the simulations results that the proposed algorithm outperforms in terms of average throughput. The proposed algorithm shows more throughput than that of the other existing state of art algorithms taken for comparison. With respect to increase of simulation time, packet loss rate also increases but FCC and proposed algorithms have near about same packet loss at 50s simulation time. Simulation result shows that the proposed algorithm depicts 6.25%, 14.28% and 17.43% of decrease in PLR compared with FCC, AntSense and CODA algorithm respectively. The result of simulations has also been tabulated in table number 3.10.

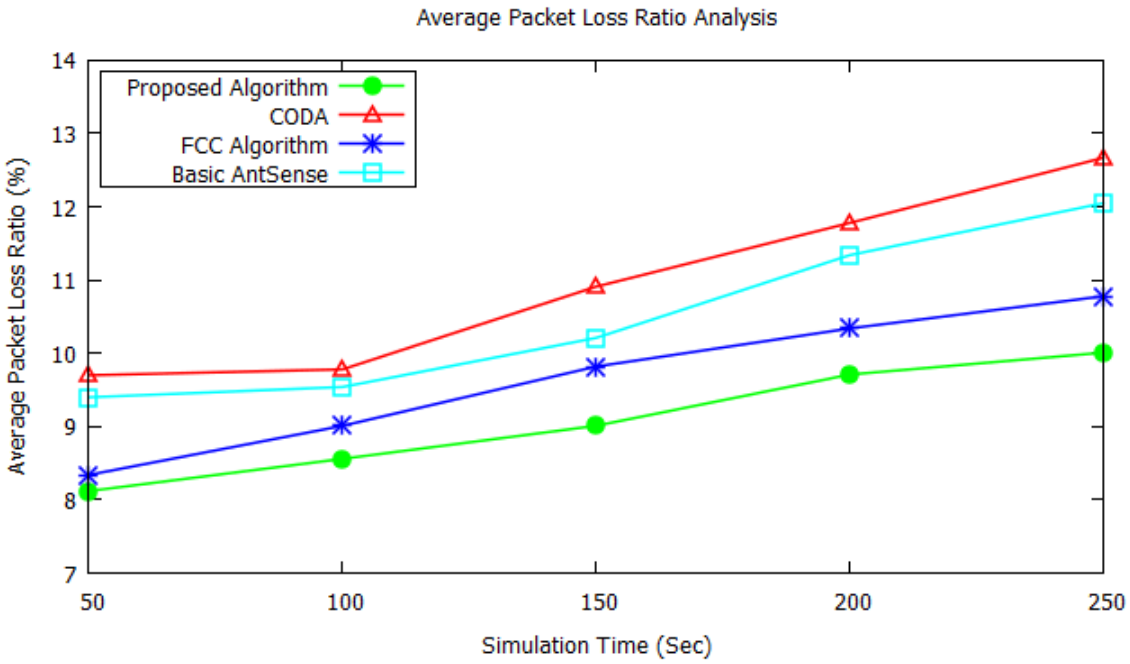

Figure 3.8: Average packet loss ratio versus simulation time for proposed, AntSense, FCC and CODA algorithm

Table 3.10: Average packet loss ratio of proposed, AntSense, FCC and CODA algorithm versus summation time

Average packet loss ratio Analysis (ms)				
Simulation Time	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**
50	8.12	8.34	9.40	9.70
100	8.56	9.01	9.54	9.78
150	9.01	9.82	10.21	10.91
200	9.71	10.34	11.34	11.78
250	10.01	10.78	12.05	12.67

3.5.2.4 Average End to End delay versus simulation time analysis

As shown in figure 3.9, we have compared the result of proposed algorithm with basic AntSense, FCC and CODA algorithms by varying simulation time. It has been observed from the simulations results that the proposed algorithm outperforms in terms of average End to End delay. The proposed algorithm shows less delay than that of the other existing state of art algorithms taken for comparison. Simulation result shows that the proposed algorithm depicts 17.17%, 37.40% and 50.30% of decrease in delay compared with FCC, AntSense and CODA algorithm respectively. The result of simulations has also been tabulated in table number 3.11.

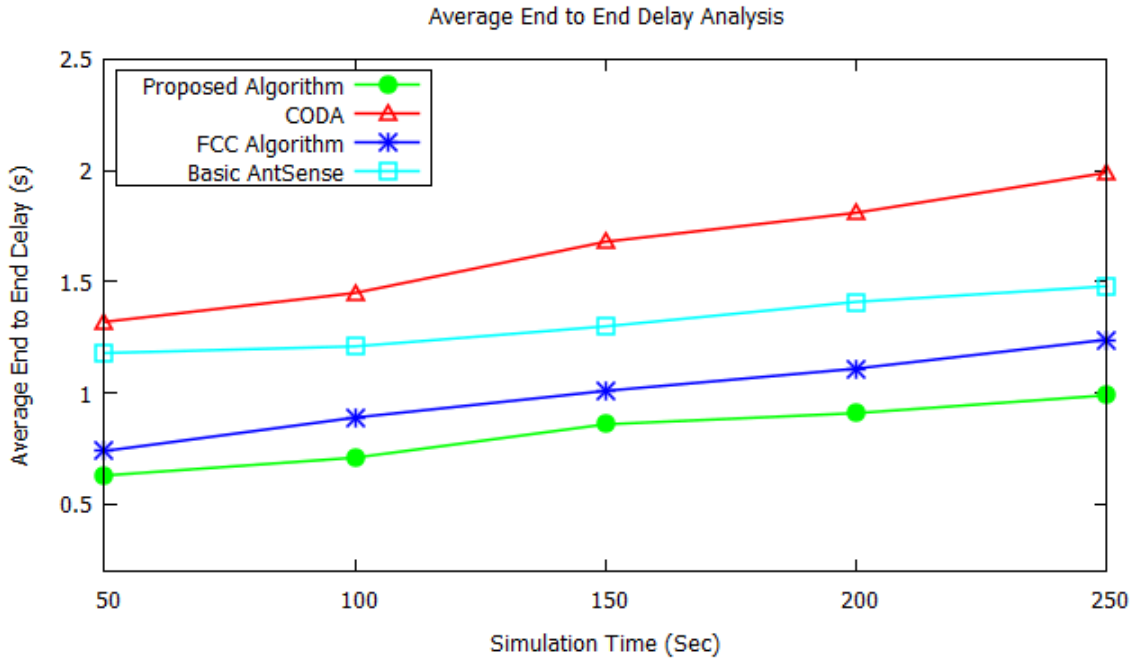

Figure 3.9: Average End to End delay for proposed, AntSense, FCC and CODA algorithm versus simulation time

Table 3.11 Average End to End delay for proposed, AntSense, FCC and CODA algorithm versus simulation time

Average End to End Delay Analysis (ms)				
Simulation Time	**Proposed Algorithm**	**FCC Algorithm**	**AntSense Algorithm**	**CODA Algorithm**

50	0.63	0.74	1.18	1.32
100	0.71	0.89	1.21	1.45
150	0.86	1.01	1.30	1.68
200	0.91	1.11	1.41	1.81
250	0.99	1.24	1.48	1.99

3.6 Conclusion of the chapter

In this chapter, a path evaluation algorithm based on modified ant colony optimization technique for controlling the congestion in WSNs have been propsoed. The proposed algorithm improves the heuristic function for selection the next neighbor node to transmitting the packet. The improved heuristic function considers residual energy, buffer occupancy and bandwidth allocated to the neighbor node and calculates path preference probability of all the neighbor nodes. The algorithm selects the neighbor node with maximum path preference probability. The proposed algorithm has been simulated by varying number of nodes in the network as well as simulation time in the network and results obtained have been compared with three existing state of art algorithms basic AntSense, FCC and CODA. The simulation result shows that the proposed algorithm shows the decrease in the average energy consumption, average packet loss ratio, average end to end delay and increase in the throughput of the network when compared with basic AntSense, FCC and CODA algorithm. The simulation results also reveal that the proposed algorithm shows overall performance gain in the range of 1.24%-12.63%, 1.24%-22.75%, 11.68%-17.07% and 11.36%-70% for energy consumption, throughput, packet loss rate and end to end delay respectively when compared with FCC, CODA and AntSense algorithm.

CHAPTER 4

A fair and efficient rate allocation based congestion control protocol in WSNs

4.1 Introduction

The event-driven nature of WSNs causes uncertainty in bandwidth utilization. WSNs traditionally operate at low or no load until sparked by a detected event. When the events are identified, the information in transit is critical. Congestion management strategies that are well-designed enable for the efficient transmission of huge amounts of data from a large number of nodes through one or more routes to data processing centers, which are commonly referred to as sink nodes in WSNs. Bulk data is regularly generated in addition to continually observed data in these high-data-rate applications. When the sensors discover a large anomaly, they produce and transmit massive amounts of data at a considerably faster rate. Under such traffic levels, network breakdown due to congestion is unavoidable if congestion is not controlled. In addition to these effects, one of the most significant consequences of WSNs is congestion. Congestion is deterioration in radio channel quality. Because data transfers across various radio connections interact with one another, channel quality is influenced not only by noise but also by traffic density. [68].

Congestion also has a negative impact on energy efficiency. The network loses energy by delivering packets from upstream nodes to the sink only to have them discarded due to congestion.

Furthermore, ensuring flow fairness is highly desirable. However, because application demands differ, it is challenging to establish the idea of fairness in sensor networks. However, how the available bandwidth is distributed among the flows is determined by the application's requirements. As a consequence, it would be desirable to abstract the idea of fairness while constructing a congestion management system for sensor networks. The constraints of limited bandwidth and node life necessitate smart handling of data via the network. As a result, considering the network's limited resources,

data transfer over the network without congestion is a technological problem. As a result, in such cases, it is critical to devise an effective rate management approach in order to avoid congestion while also extending the life of a node [68].

In this chapter, we proposed an innovative congestion control technique that accomplishes a roughly fair bandwidth distribution to all the nodes in the network. Intuitively, we foresee each flow to acquire a reasonable share of the accessible bandwidth depending on its rate of packets generation at the sensor nodes. As a matter of fact, technique to distribute a immobile rate to each flow of a sensor node seems to be hard. To achieve a roughly sustainable bandwidth share, we implement a novel mechanism identified as fair, efficient, and adaptable congestion control algorithm (FEACC). In FEACC, we have used available bandwidth resources and active flows belonging to a particular node to estimate the fair rate that can be efficiently assigned to each node in the network.

4.2 Network Model and problem statement

WSNs are comprised of a huge quantity of sensors and a sink node, also known as a base station. The sink is linked to a data-collection center via an external network. We assume that sensor transmitters have variable transmission rates but the rate at which sensor generates the data is same. All the sensors in the network occupy the single wireless medium, and each packet is routed in the nodes located in their neighbors. Two sensors nodes are said to be neighbor of each other if they can communicate directly and are in same transmission range.

Consider WSNs of X sensor nodes; each node has been assigned an integer value for identification purpose in the range of [1, X] that uniquely identifies it. We assume that all the sensor nodes in the network always have data to send in their neighborhood node. The S_{th} flow, i.e., f_s denotes traffic originated by source node S. We are attempting to assign fair and productive rate to f_s. Congestion management is achieved by ensuring that the overall rate at which each sensor node routes the data packets should be less than or equal to usable bandwidth.

In WSNs, congestion has disastrous implications for network throughput, power consumption of the node, and packet delivery fraction. When the provided load exceeds the critical point of congestion, a lot of packets start drooping thereby generating the need of retransmission which further increases the energy consumption of the nodes in the network. When after retransmission packets are not able to deliver at sink node results in decreased throughput and packet delivery fraction. We should avoid sending these packets because they are likely to be lost. To prevent congestion, we change the sending rate for each flow as soon as possible. The chief objective of this work is to show how to efficiently and adaptively assign the fair rate to each node in the network.

4.2.1 Problems in existing systems and motivation behind proposed solution

- Most of the techniques from the literature focus on controlling the congestion by only reducing the data rate from the source or intermediate nodes. They do not focused on the matter that data rate can also be increased if congestion is alleviated.
- Generally, most of the algorithms considers only input and output traffic rate decide the flow rate of data. And by considering the difference of them is used to decide the reduced rate of traffic. They have not considered the other parameters like bandwidth and active flow rates associated with the nodes.

The motivation behind proposing a novel model to make an algorithm that can approximately able to provide the fair rate to all the nodes of the sensor networks in addition to controlling the congestion. Primarily in our algorithm, fairness will be achieved in the event of congestion only in order to address the issues of congestion. Whenever congestion occurs in the network, we may assume that the sending rate of source node or intermediate node is greater than the processing nodes. So a fairness mechanism is needed to adjust the rate of source node in the event of congestion. The detailed algorithm along with flowchart in described in section 4.3.

4.3 Proposed Algorithm: Fair, Efficient, and Adaptable congestion control – FEACC

Both the accessible bandwidth and the traffic generated load in WSNs differ over time. If we consider contention and interference in wireless medium it is very difficult to implement equal resource allocation. We use the channel busyness ratio M_b as a parameter to represent the network usage and congestion status which is capable to allocating the resources to all the nodes in the network and their corresponding flows. We calculate the amount of accessible bandwidth and the number of flows which are active for each node. As a result, based on the aforementioned metrics, we create a transmission control mechanism considering the fairness parameter which assigns a fair rate to all the nodes in the event of congestion.

The process followed in the proposed algorithm is shown in the figure 4.1. When available load exceeds the critical point of congestion, there is an increase in packet loss and decrease in network throughput. Furthermore, when a packet is discarded, the energy consumed by upstream sensors on the packet is squandered. The greater the loss, the greater the distance travelled by the packet. We should avoid sending these packets, which are almost certainly going to be dropped. We alter the transmission rate for each flow as quickly as feasible to avoid congestion. Our primary issue is figuring out how to properly change the transmitting rate for each flow.

Whenever a packet drops at the intermediate sensor node it indicates that there is congestion at that path and sending rate of the source node need to be adjust on such a way that congestion can be controlled. So when a packet drops at the intermediate node, our algorithm tends to send a warning message to immediate a parent node regarding congestion status In this instance, our system will compute and assign the approximate fair rate to all flows going through that node. To attain this, algorithm will send the control message to the source node to change the sending rate by calculated fair rate. Our algorithm will work in the situation when there is congestion occurs. To control

congestion, each node in the network has been allocated a fair share rate for efficient transmission.

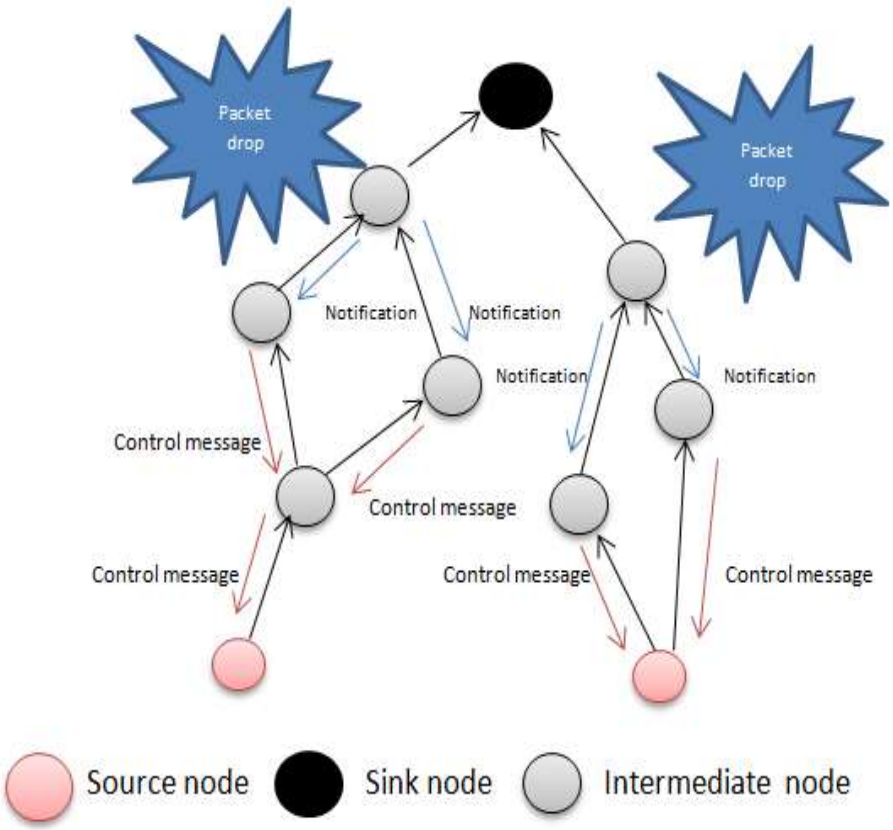

Figure 4.1: Working of proposed algorithm FEACC

The various steps followed in proposed work are as follow:

Step 1: Estimate the bandwidth available

Basically the available bandwidth for each node depends upon the busyness ratio of medium (M_b). Medium busyness ratio is the sum of time interval in which medium is busy in transmitting/receiving the packets or collision to the total time taken. We will take the medium utilization value as 95% [15] as a threshold value (M_{thb}). The available bandwidth (BW_n) for each node can be calculated by using the following formula

$$BW_n = \begin{cases} 0 & M_b \geq M_{thb} \\ \frac{B_w(M_{thb}-M_b)Size/T}{} & M_b < M_{thb} \end{cases} \quad (4.1)$$

Where B_w is rate of transmission of packets in the medium, Size is the packet size on the MAC layer and T is time of successful transmission. As a result, as long as the medium busyness ratio does not surpass the threshold decided, the node will not enter congested mode, and the bandwidth which is available will be able to accommodate all the traffic without any contention problem.

Step 2: Calculate the active flows for each node

All sensors nodes in WSNs either produce or transmit packets. All the packets will finally ends at the sink node for collection. As wireless medium is being used for both incoming as well as outbound traffic, the number of flows FL should differ from the definite number of flows. This is due to the fact that flows passing through a node will use the wireless medium twice: one for forwarding packet which comes from upper node and another its own packet generated. Thus FL can be computed as

$$\text{FL}= \begin{cases} 2N+1 & When\ a\ outgoing\ flow\ is\ generated \\ 2N & When\ no\ outgoing\ flow\ generatd \end{cases} \quad (4.2)$$

Step 3: Calculate the fair rate for each node in the network

Now we need to calculate and allocate the fair rate to each node which indicates the increased or decreased flow rate. Hence the fair rate shares (F_t) is calculated using the following equation:

$$F_t = \frac{M_{thb}}{M_b} * S/FL \quad (4.3)$$

Now we need to assign this flow rate to all the nodes in the network to achieve fairness among all the nodes in the network. This can be done using following process:

$$FR = \begin{cases} F_t & Current_{flow_{rate}} > F_t \\ Current_{flow_{rate}} & Current_{flow_{rate}} \leq F_t \end{cases} \quad (4.4)$$

Where $Current_{flow_{rate}}$ is the flow rate of a particular node and FR is the final rate to be allocated. Once the FR has been decided algorithm will send a control message to all the nodes in the network to update regarding new rate.

The flow of proposed work is shown in the following figure 4.2.

The algorithm of the proposed work is explained below:

Algorithm 4.1 : *Fair Rate Allocation for Congestion Control*

1: Initialize the network of n nodes.
2: Initiate the route discovery process using underlying algorithm.
3: Repeat the steps 3 to 5 for the node at which congestion occurs:
Calculate the bandwidth available (for congestion indication) using the following equation:

$$BW_n = \begin{cases} 0 & M_b \geq M_{thb} \\ \frac{B_w(M_{thb}-M_b)Size/T}{} & M_b < M_{thb} \end{cases}$$

4: Calculate the active flows for each node using the following formula:

$$\text{FL}= \begin{cases} 2N+1 & When\ a\ outgoing\ flow\ is\ generated \\ 2N & When\ no\ outgoing\ flow\ generatd \end{cases}$$

5: Calculate the fair rate for each node in the network using the following equation:

$$F_t = \frac{M_{thb}}{M_b} * S/FL$$

6: Assign the fair flow rate calculated in the step number 5 according to following rule:

$$\text{FR}= \begin{cases} F_t & Current_{flow_{rate}} > F_t \\ Current_{flow_{rate}} & Current_{flow_{rate}} \leq F_t \end{cases}$$

7: Update all the other source nodes regarding new flow rate FR.

8: Repeat the steps 2 to 6 until simulation ends.

4.4 Complexity Analysis of proposed algorithm– FEACC

The time complexity of this algorithm will depend primarily upon the number of nodes in the network and number of paths on which the congestion occurs. The underlying algorithm will calculate the fair and efficient transmission rate for the source node in the network in the event of congestion and control information will be passed to the source node through the path where congestion is reported. So the algorithm will work in two loops i.e. the first outermost loop will transverse all the nodes in the network which will run for 'n' times which is number of nodes in the network and another for calculating the fair and efficient transmission rate for source node in the network which will send the fair rate to the source node.

So if we consider a case, when there is no congestion in the network then the algorithm will keep traversing all the nodes in the network assuming the source rate as fair and efficient. This case can be considered as a best case and the time complexity of this case will be O (n). Next if there is congestion in the network so we have to run the loop for only that paths where the congestion occurs so we can denote that number of paths by 'm'. So we can conclude the time complexity of the algorithm in worst case will be O (n+m).

The complexity analysis of the proposed algorithm has also been summarized in the following table.

Table 4.1: Time complexity of Fair, Efficient, and Adaptable congestion control algorithm

Time Complexity of Fair, Efficient, and Adaptable congestion control algorithm	
Best Case	Worst Case
O(n)	O(n+m)

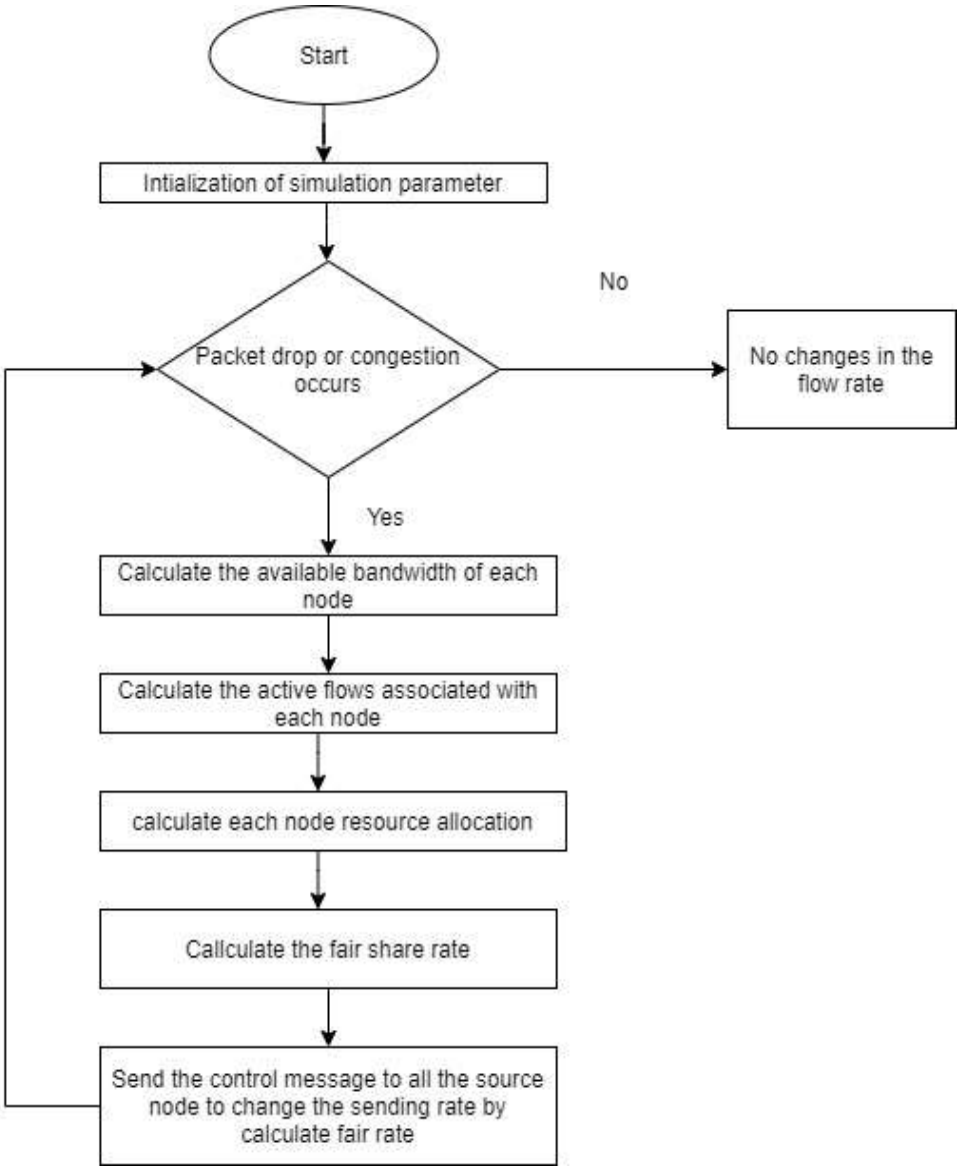

Figure 4.2: Flowchart depicting the proposed algorithm FEACC

4.5 Simulation and Result analysis

We have evaluated the proposed algorithm by taking the variable number of nodes. To run the simulations, we use the network simulator NS 2 version 2.33. Table 4.2 describes the default simulation parameters

Table 4.2: Simulation parameters and rules

Simulation Parameter	**Corresponding value**
Routing Protocol	AODV
Number of sensor nodes	10,20,30,40,50
Queue Length	100
Initial Energy	90 J
Simulation Time	50 Sec
Packet Size	50 bytes
Number of source and sink node	1,1
MAC protocol	802.11
Offered traffic load	100,200,300,400,500 Kbps

AODV protocol for simulating our algorithms will be used. Proposed algorithm-FEACC has been simulated using NS 2.33 and results have been compared with existing state of art algorithms i.e. CCF [81] and CAP [80]. It has been observed from the results that the proposed algorithm outperforms the existing state of art algorithms in terms of average throughput analysis, packet delivery fraction and energy consumption.

The simulation has been carried in two parts:

- By variation in number of nodes
- By variation in offered traffic load

4.5.1 Simulation results by varying number of nodes

In this section, different number of nodes (10, 20, 30, 40, and 50) has been taken for simulating the proposed FEACC and existing state of art algorithms i.e. CCF and CAP by keeping the offered traffic load as 100 Kbps. Our main aim of performing this simulation is that we want to check the effect of number of nodes on the network. The evaluation parameters taken for the simulation are:

- Average throughput analysis
- Average packet delivery fraction
- Average Energy consumption
- Average End to End delay

4.5.1.1 Average Throughput Analysis

In this section we have evaluated the performance of proposed algorithm – FEACC on average throughput analysis parameter. The simulation results show that proposed algorithm for controlling the congestion achieves higher throughput (in kbps) than the existing state of art algorithms CCF and CAP as shown in figure 4.3. This is due to the fact that proposed algorithm adaptively assigns the fair data rate to all the nodes in the network and no node is over loaded or under loaded there by increasing the throughput by less number of dropped packets. On the other hand CCF algorithms only consider the input and output data rates, based on difference of them they decide to increase or decrease the transmission rate during congestion. Similarly CAP does not implement any rate adjustment during the congestion and its simply routing the traffic to alternate paths available in the network. Proposed algorithm - FEACC shows 3.47% increase in throughput as compared to CCF algorithm and 12.30% increase in throughput compared with CAP algorithm. The simulation results have also tabularized in table number 4.3.

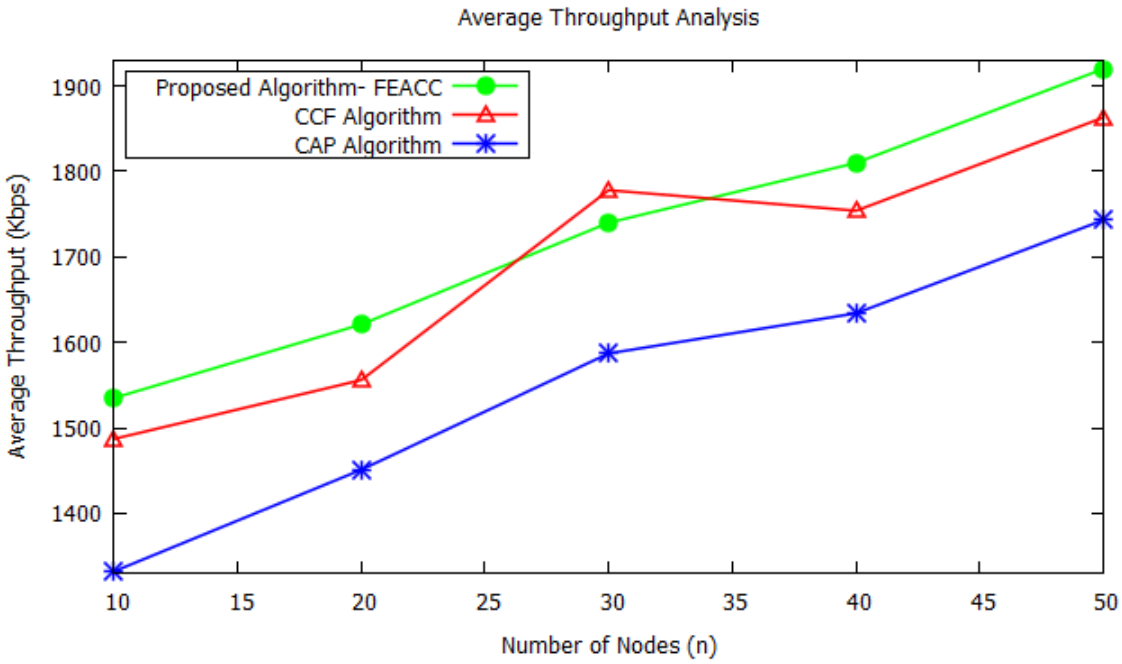

Figure 4.3 Average Throughput analysis of proposed algorithm - FEACC, CCF and CAP algorithm

The simulation results have also been summarized in the following table 4.3:

Table 4.3: Average Throughput analysis of proposed algorithm – FEACC, CCF and CAP algorithm

Average Throughput Analysis (Kbps)			
Number of Nodes	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
10	1535	1487	1332
20	1721	1556	1451
30	1740	1778	1521
40	1810	1754	1634
50	1920	1863	1743

4.5.1.2 Average Packet delivery ratio

The simulations show that proposed algorithm for controlling the congestion results in a high packet delivery ratio than existing state of art algorithms CCF and CAP.

Figure 4.4 demonstrates the packet delivery fraction with respect to number of nodes in the network. In our proposed algorithm the packet delivery fraction is more due to the adaptive nature of algorithm to assign the fair rate thereby reducing the congestion in the network and due to controlled congestion, less number of packets gets dropped. On the other hand CCF rely on input and output traffic rates for providing fairness resulting lower packet delivery fraction. CAP shows worst packet delivery fraction as compared to FEACC and CCF due to no control on rate adjustment during congestion. The FEACC algorithm shows 6.35% and 12.26% increase in packet delivery ratio as compared to CCF and CAP algorithm. The simulation results have also tabularized in table number 4.4.

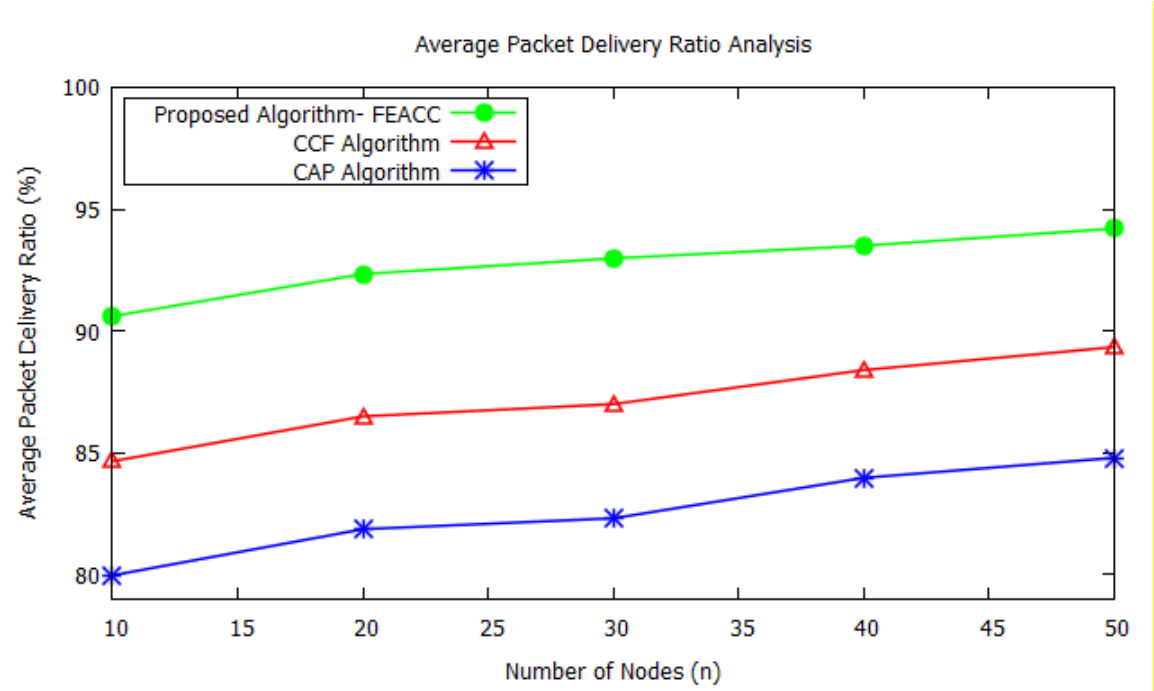

Figure 4.4 Average Packet Delivery ratio analysis of proposed algorithm – FEACC, CCF and CAP algorithm.

The simulation results have also been summarized in the following table 4.4:

Table 4.4 Average Packet Delivery ratio analysis of proposed – FEACC, CCF and CAP.

Average Packet Delivery ratio analysis (%)

Number of Nodes	FEACC Algorithm	CCF Algorithm	CAP Algorithm
10	90.61	84.66	79.98
20	92.34	86.50	81.87
30	92.98	87.00	82.32
40	93.50	88.40	83.98
50	94.20	89.34	84.80

4.5.1.3 Average Energy Consumption Analysis

Average Energy consumption is another parameter to evaluate the performance of WSNs. When there are chances of congestion, there is a need of retransmission of the packets which is not delivered to the destination there by increasing the energy consumption of the nodes to great extent. Simulation has been conducted and it has been concluded that our proposed algorithm reduces the packet drop and thereby decreasing the retransmission of the packets hence energy consumption of the network will decrease when compared to CCF and CAP as shown in figure 4.5. CCF does not consider the other important parameters for evaluation of rate hence there by chances of packet drop occurs and increase the total energy consumption. CAP do not use any fairness control and node's load is not justified which may cause drop of packets and increases the energy consumption. The FEACC shows 1.72% and 3.53% decrease in average energy consumption compared to CCF and CAP. Figure 4.5 represents the total energy consumption of proposed algorithm – FEACC, CCF and CAP. The simulation results have also tabularized in table number 4.5.

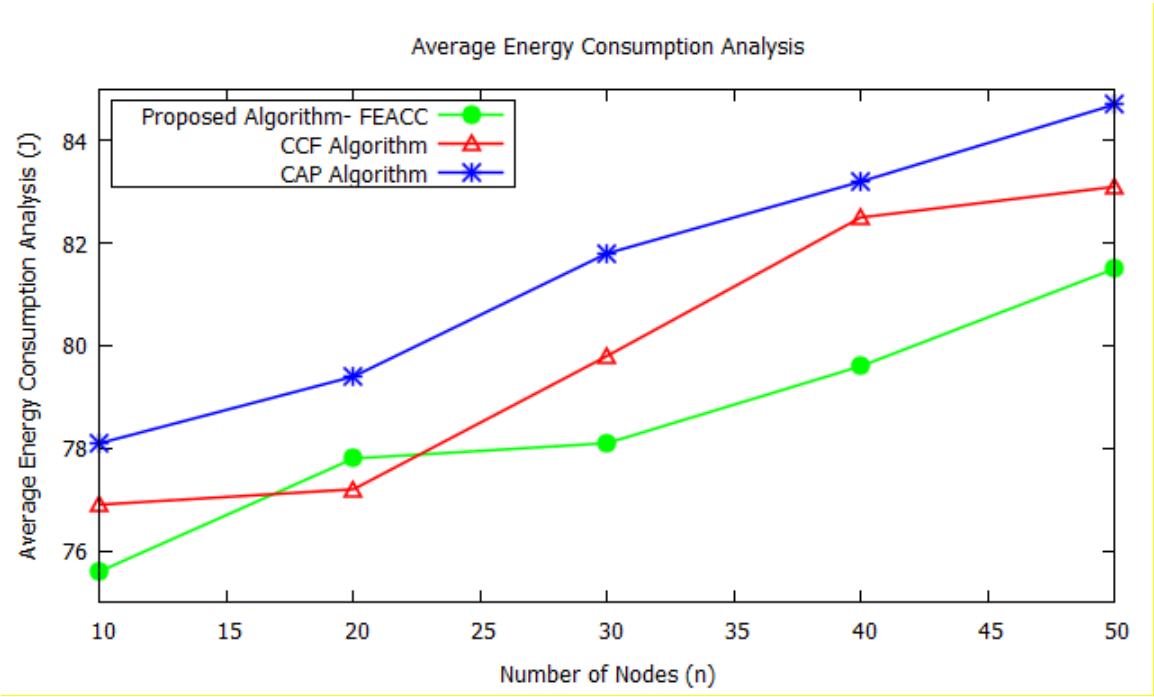

Figure 4.5 Average Energy Consumption analyses of proposed algorithm – FEACC, CCF and CAP

The simulation results have also been summarized in following table 4.5.

Table 4.5 Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP.

Average Energy Consumption (J)			
Number of Nodes	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
10	75.6	76.9	78.1
20	77.8	77.2	79.4
30	78.1	79.8	81.8
40	79.6	82.5	83.2
50	81.50	83.1	84.7

4.5.1.4 Average End to End delay

Average End to End delay is another parameter to evaluate the performance of WSNs. When there are chances of congestion, there is a need of retransmission of the packets which is not delivered to the destination there by increasing in delay in reaching the packets to great extent. Simulation has been conducted and it has been concluded that our proposed algorithm reduces the packet drop ratio and thereby decreasing the retransmission of the packets and hence decreases the delay as shown in figure 4.6. In the event of congestion our algorithm quickly able to calculate the new fair rate for the nodes and informs the source node to adjust its rate accordingly hence End to End delay will decrease as compared to CCF and CAP which do not provide the proper adjustment rate for nodes in the network. The FEACC shows 10.36% and 27.58% decrease in average energy consumption when compared with CCF and CAP algorithm. Figure 4.6 represents the total End to End delay of proposed algorithm – FEACC, CCF and CAP. The simulation results have also tabularized in table number 4.6.

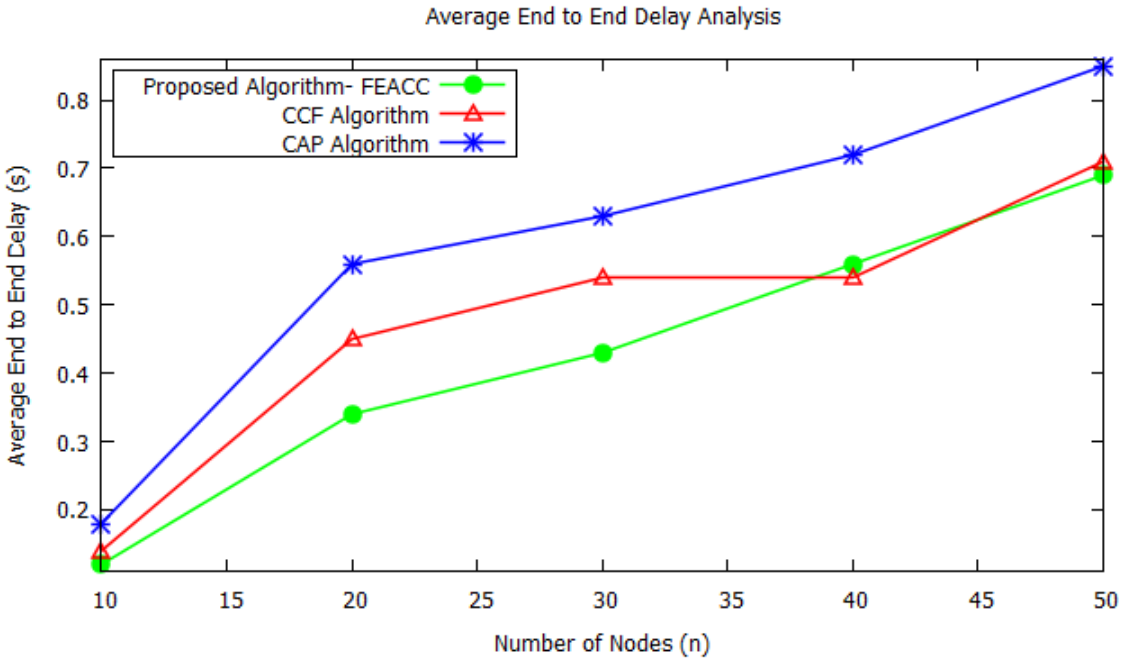

Figure 4.6 Average End to End delay analyses of proposed algorithm – FEACC, CCF and CAP

Table 4.6 Average End to End delay analysis of proposed algorithm – FEACC, CCF and CAP

Average End to End delay analysis (s)			
Number of Nodes	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
10	0.12	0.14	0.18
20	0.34	0.45	0.56
30	0.43	0.54	0.63
40	0.56	0.54	0.72
50	0.69	0.71	0.85

4.5.2 Simulation results by variation of offered traffic load

In this section, different traffic load has been taken (100, 200,300,400,500 Kbps) for simulating the proposed FEACC and existing state of art algorithms i.e. CCF and CAP by keeping the number of nodes as 50. Our main aim of performing this simulation is that we want to check the effect of offered load on the network. The evaluation parameters taken for the simulation are:

- Average throughput analysis
- Average packet delivery fraction
- Average Energy consumption
- Average End to End delay

4.5.2.1 Average Throughput Analysis

Figure 4.7 represents the throughput performance comparison FEACC with CCF and CAP algorithms. A good increase in throughput has been observed in all the schemes as traffic load increases. This is due to the successful reception of large amount of packets at the sink node. But after 300 Kbps traffic load there is a slight decrease in the

throughput due to congestion occurs because of heavy traffic load. Although throughput decreases with increase traffic load but overall FEACC outperform the CCF and Cap algorithm. On an average FEACC shows 2.12% and 7.30% increase in throughput when compared with CCF and CAF respectively. This is due to the reason that CCF consider only input and output rate to calculate the fair share rate for nodes and CAP only route the packets through alternative paths without fairness. The simulation results have also been tabularized in table number 4.7.

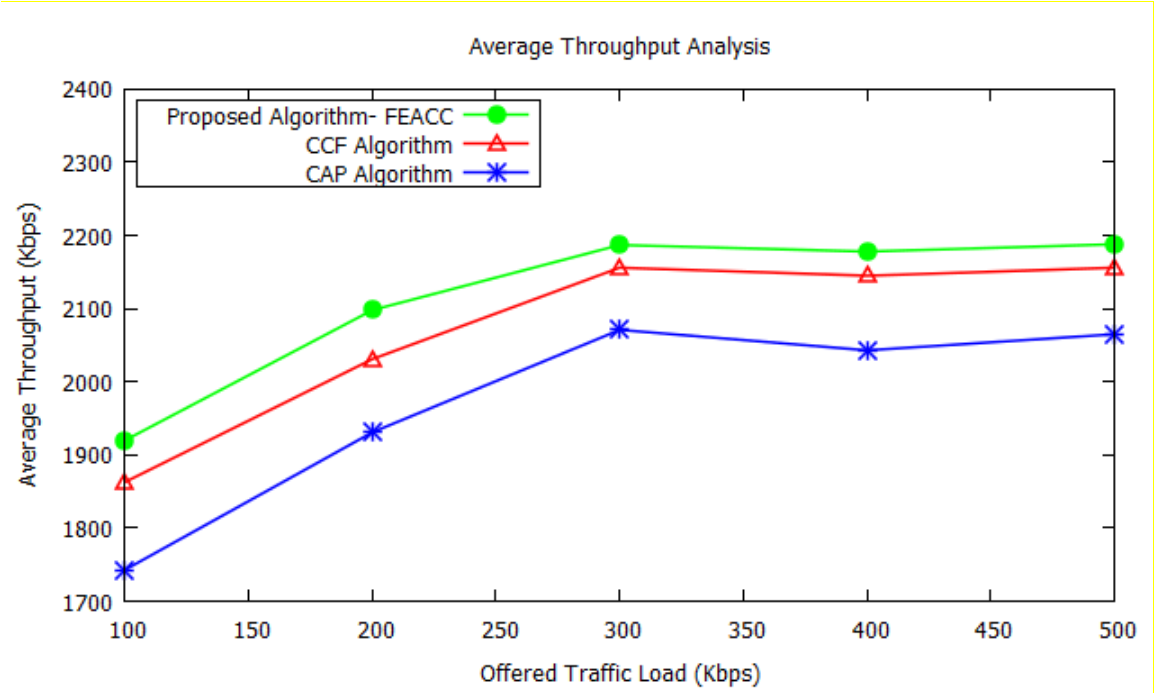

Figure 4.7 Average Throughput analysis of proposed algorithm – FEACC, CCF and CAP

Table 4.7 Average Throughput analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load

Average Throughput analysis (Kbps)			
Traffic Load (Kbps)	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
100	1920	1863	1743
200	2098	2031	1931
300	2187	2156	2071

400	2178	2145	2043
500	2188	2156	2065

4.5.2.2 Average Packet Delivery Ratio Analysis

Figure 4.8 represents the packet delivery performance comparison of FEACC with CCF and CAP algorithms. It can be observed from the graph that PDR deceases with the increase of offered traffic load. This is due to congestion occurs in the network as more number of packets will be generated with respect to increase in traffic load. Apart from this, the FEACC algorithm outperforms CCF and CAP algorithms by offering high PDR in most of the traffic load conditions. On an average FEACC shows 0.72% and 1.32% increase in PDR when compared with CCF and CAP respectively. FACC outperforms than CCF and CAP due to fair share rate allocation in the event of congestion. The simulation results have also been tabularized in table number 4.8.

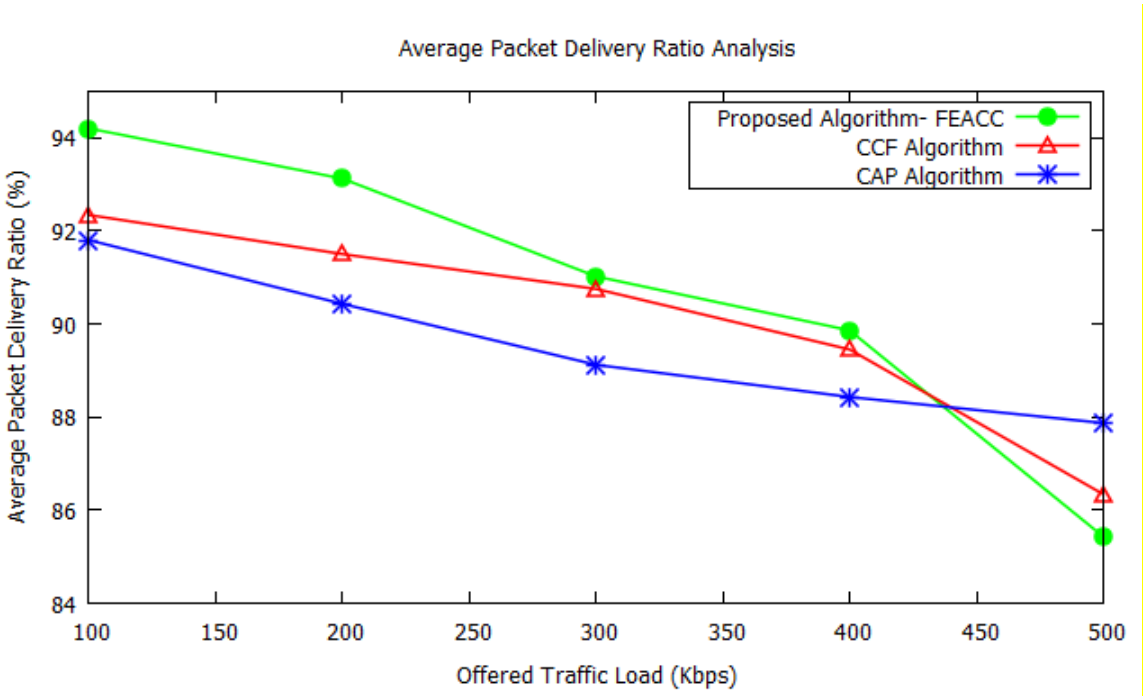

Figure 4.8 Average Packet delivery Ratio analysis of proposed algorithm – FEACC, CCF and CAP

Table 4.8 Average Packet Delivery Ratio analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load

Average Packet Delivery Ratio analysis (%)			
Traffic Load (Kbps)	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
100	94.20	92.34	91.80
200	93.12	91.50	90.43
300	91.02	90.75	89.12
400	89.86	89.45	88.43
500	85.43	86.34	87.87

4.5.2.3 Average Energy Consumption Analysis

Figure 4.9 represents the energy consumption comparison of FEACC with CCF and CAP algorithms. It can be observed from the graph that overall network's energy consumption increases with the increase in offered traffic load. This is due to congestion occurs in the network when we are increasing the traffic load. When traffic load increases, a number of packets start dropping which may need re-transmission. Due to re-transmission, network's energy gets wasted and decreases. Apart from this, the FEACC algorithm outperforms CCF and CAP algorithms by showing less energy consumption when compared with existing state of art algorithms in most of the traffic load conditions. On an average FEACC shows 1.43% and 2.80% decrease in energy consumption when compared with CCF and CAF respectively. The simulation results have also been tabularized in table number 4.9.

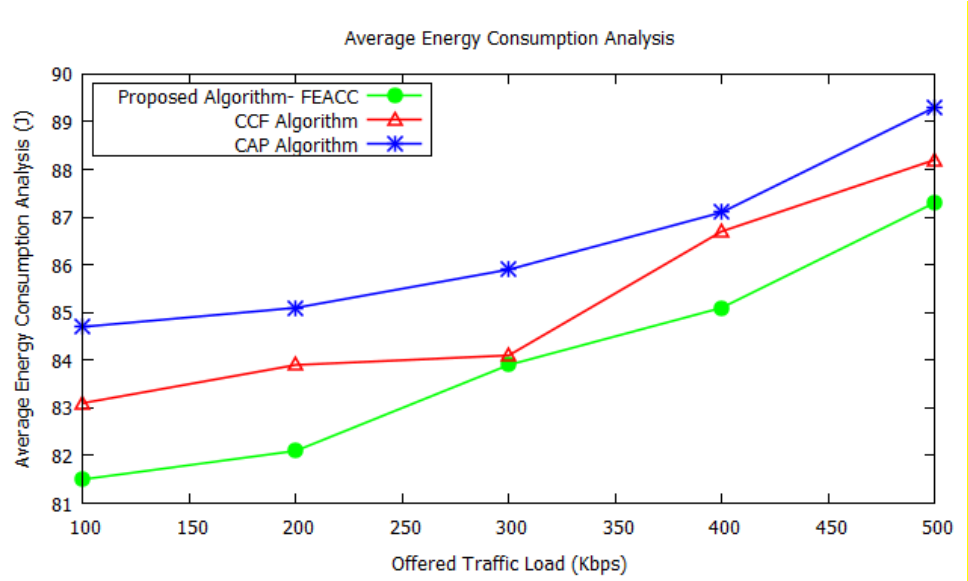

Figure 4.9 Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP

Table 4.9 Average Energy Consumption analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load

Average Energy Consumption analysis (%)			
Traffic Load (Kbps)	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
100	81.5	83.1	84.7
200	82.1	83.9	85.1
300	83.9	84.1	85.9
400	85.1	86.7	87.1
500	87.3	88.2	89.3

4.5.2.4 Average End to End Delay Analysis

Figure 4.10 represents the end to end delay comparison of FEACC with CCF and CAP algorithm. It can be observed that with respect to increase in traffic load, the overall delay in the network increases as more number of packets needs to be re-transmitted due to congestion. Hence it increases delay in the network. However results also reveals that overall delay encountered with the proposed algorithm FEACC is less as compared to

CCF and CAP. On an average FEACC shows 2.46% and 16.84% decrease in end to end delay when compared with CCF and CAF respectively. The simulation results have also been tabularized in table number 4.10.

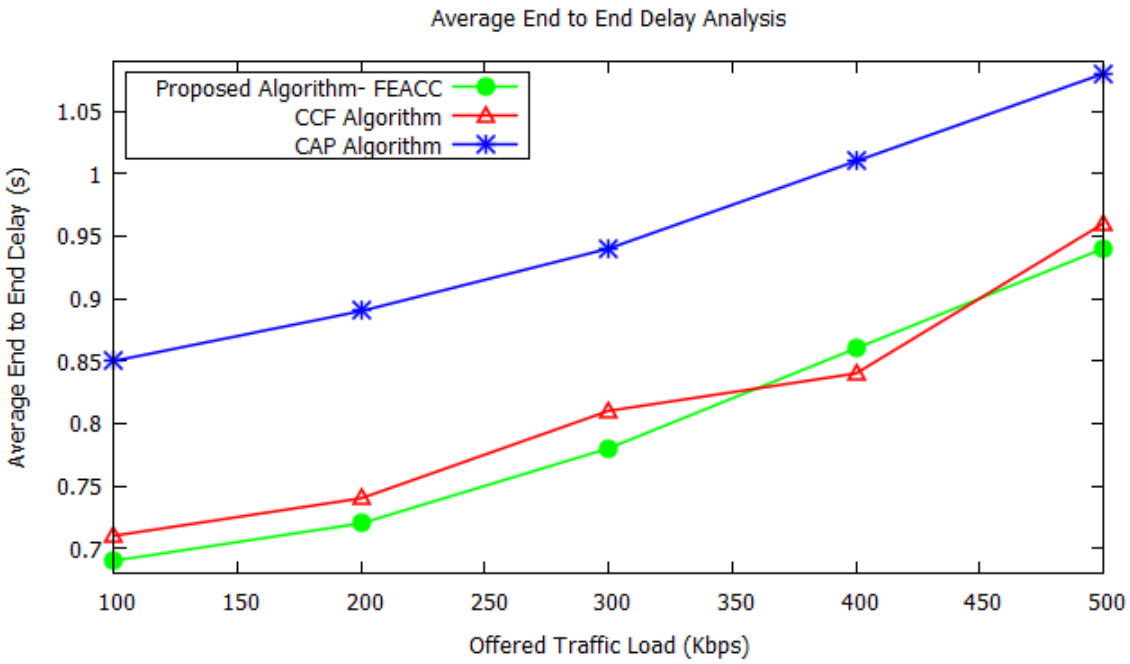

Figure 4.10 Average End to End delay analysis of proposed algorithm – FEACC, CCF and CAP

Table 4.10 Average End to End delay analysis of proposed algorithm – FEACC, CCF and CAP versus offered traffic load

Average End to End Delay analysis (s)			
Traffic Load (Kbps)	**FEACC Algorithm**	**CCF Algorithm**	**CAP Algorithm**
100	0.69	0.71	0.85
200	0.72	0.74	0.89
300	0.78	0.81	0.94
400	0.86	0.84	1.01
500	0.94	0.96	1.08

4.6 Conclusion of the chapter

Achieving a fair and efficient transmission rate to all the nodes in WSNs is very challenging due to its event driven nature. In this chapter, we propose a congestion control scheme based on fairness control for WSNs. This paper unveiled a novel mechanism for reducing congestion and achieving reasonably fair bandwidth allocation in WSNs by adjusting the rate at which source node can transmit the data. The main idea of the proposed work is to detect the congestion and in to alleviate the effect of congestion by adaptively allocating the fair rate to the nodes in the network. We indicated through simulations that FEACC outperforms the existing art of state algorithms CCF and CAP. This is due to the fact that the proposed algorithm uses bandwidth as well as active flows associated with the nodes to achieve a fair rate. On the other hand, CCF only consider input and output rate of a node which is not sufficient to accurately achieve a fair rate. Secondly CAP do not implement any fairness mechanism, it only route the packets to alternate paths in the presence of congestion. The proposed scheme has been evaluated on network parameters average throughput, average energy consumption and average end to end delay.

CHAPTER 5

Detection and Mitigation of network adversaries in WSNs in the event of congestion

5.1 Introduction

WSNs, which is a distributed and conscience network, is a group of sensor nodes with limited energy sources that work together to achieve a shared objective. WSNs are frequently deployed to perform various monitoring tasks in tough and hostile environments that are inaccessible and even dangerous. Adversaries can take advantage of the unattended nature of WSNs by launching a variety of physical attacks such as node replication attacks, signal or radio jamming, denial of service (DoS) attacks, flooding attacks, node outages, eavesdropping, and Sybil attacks [108] [109]. The attacker bombards the network with excessive fake route requests in order to prevent communication between source and destination in a route request flooding attack, A data flooding assault, on the other hand, bombards the network with excessive meaningless data packets with the objective of blocking communication between source and destination. [109][110]. In this chapter, we discussed the flooding attack, proposed a methodology for detecting and preventing flooding attacks, and finally evaluated the proposed methodology using the network simulator NS 2.33.

5.2 Flooding attacks in WSNs

Flooding is a sort of denial of service attack in which a malicious node sends a large number of duplicate fake packets into the network to waste network capacity and generate network congestion. It is a matter of fact when a malicious node floods the network's intermediate node with a large number of fake packet requests, the network gets congested. Due to congestion, network's performance degrades in terms of throughput, delay, and packet delivery fraction and energy consumption. In this category of attack, malicious node floods the network's intermediate nodes with such a large amount of fake requests that it will make impossible for intermediate node to send the

legitimate data packets to the sink node or base station. In a selective flooding attack, the node operates properly at times and maliciously at others. The selective route request flooding attack is a selective flooding attack that employs a fraudulent route request packet, whereas the selective data flooding attack employs a bogus data packet. [111][112].

5.2.1 Types of Flooding attacks

There are various types of flooding attacks which are discussed below and is shown in the figure 5.1 also:

Hello Flooding

In this type of attack, the malicious node will have high range transmitter which depicts the best node to forward the route request packets. This malicious node keeps spreading Hello messages, attempting to encourage other nodes that he is adjacent and a neighbor to send the data through malicious node. As a result, regular nodes continue to send packets to the compromised node, expecting it to transmit them to the sink node because it has more transmission range and energy than any other conventional node in the network. [113][114].

RREQ Flooding

This type of attack floods the network with route request packets (RREQs) for arbitrary node IDs that never appear in the network. Conventional nodes continue to forward these RREQs in order to gain a trail of phoney nodes. Figure 5.2 displays a WSNs RREQ Flooding assault The RREQ Flooding attack is a type of Denial of Service (DoS) attack that seeks to flood the network with bogus packets in order to interrupt network connection between nodes [113][114].

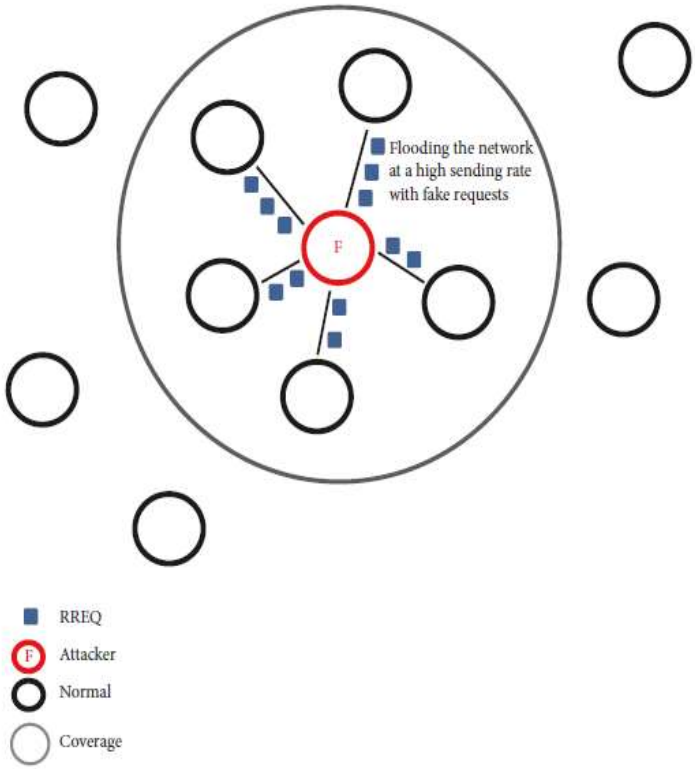

Figure 5.1: Basic scenario of flooding attack in WSNs

- **Data Flooding**

This is referred to as a Sleep Deprivation Attack. In this type of attack the malicious node start transmitting a vast volume of bogus data to each other at a high transmission rate, using the energy of every normal node in the path between the two malicious nodes[113][114].

- **Error Flooding**

The attacker node should be in the route of any two nodes transferring data to or near each other in this fashion. The attacker node then continues to send random error messages (RERRs) to nodes in its range. Because they assume one

of the nodes sending their packet is unreadable, this will cause disruptions in the transmission process between those nodes [113] [114].

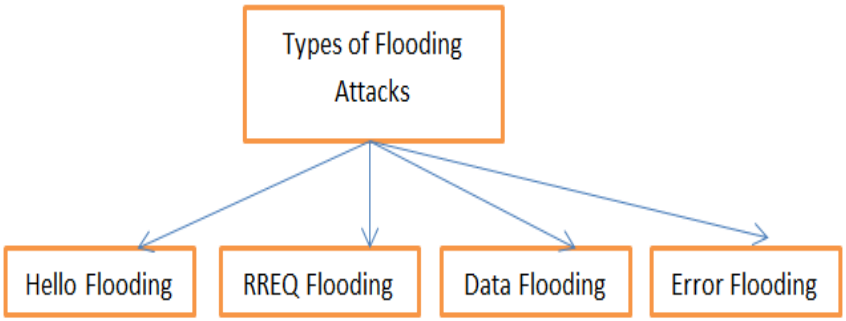

Figure 5.2: Types of Flooding Attack

5.3 Problem Statement

Due to de-centralized nature of WSNs, the nodes in the networks are prone to various types of threats. The malicious node tend to attack the networks in order to make the services unavailable thereby reducing the chances of delivering important information to the base station. So security of WSNs is very important to implement in order to protect the network from these adversaries. One of the attacks that are considered very harmful with respect to losing information is denial of service attack. Under denial of service attack, the network operation can be halted by flooding attack in which malicious node tend to send the multiple fake RREQ packets in order to create the congestion. The main task of malicious node in such type of attack is to congest the intermediate node so that the respective node will gets congested and no longer be able to hold the legitimate traffic thereby reducing the throughput, packet delivery fraction and end to end delay in the network. The traditional AODV protocol which is meant for routing does not include the process of detecting and preventing such type of adversaries [114].

With respect to above discussed problem, in this section we will propose an algorithm for detecting and preventing the effect of flooding attack in WSNs. We will consider the native AODV protocol and apply flooding attack on it with the intention of looking the effect of attack on the network. After this we will apply proposed algorithm for detection and prevention of flooding attack with the intention of looking at the effect of proposed technique on the performance of the network. The main idea of the proposed technique will be to control the congestion by detecting the malicious nodes which are injecting the fake RREQ request into the intermediate node. Once the malicious nodes are detected, we will delete it from the network for further communication so that congestion in the network can be controlled.

5.4 Proposed Algorithm - FDMM

In order to control the congestion by mitigating the effect of flooding attack, a novel mechanism has been proposed in this section. We will take the native AODV protocol and do the changes in that protocol so that it can detect and mitigate the effect of flooding attack. A flooding detection and mitigation mechanism (FDMM) has been proposed in this section. The FDMM work in three primarily phases: threshold setting phase, malicious node detection phase and isolation phase. The prime purpose of FDMM will be to detect and isolate the nodes in the network that are flooding the intermediate nodes with fake packet request. So in the first phase we will calculate the threshold that depends upon the number of route requests generated by a particular node, in the second phase we will categorize the node as a malicious node if its route request sending rate is greater than the threshold value and finally we will confirm the node as a malicious node by sending the dummy packet. If we do not get a reply with in the stipulated time frame, we will isolate that node from the further communication. A Control message will be broadcasted by the source node to the all the other nodes in the network regarding the malicious node.

The various steps followed in the proposed algorithm are shown below:

Threshold calculation

As AODV protocols start its process, source node broadcast the route request packet (RREQ) to the intermediate nodes in the network. The intermediate nodes will reply back with route reply packet (RREP) if it has a path to the destination. So the FDMM algorithm keeps track of activities of all the nodes in the network and records the number of RREQ packets broadcasted by all the nodes in the network for a period of time T. After the time T, algorithm will calculate the mean of the RREQ packets broadcasted by all the intermediate nodes as per the equation written below:

$$T = \sum_{i=1}^{n} \frac{RREQ_i}{n} \tag{5.1}$$

Where $RREQ_i$ is the route request of i_{th} node, n is the total number of nodes and T is the mean number of route request packets of n nodes.

Malicious node detection phase

In this phase we will check whether the node in the network is malicious or not. The computer values of mean in the previous step will act as a final threshold (T) and if number of RREQ packets broadcasted by any node is greater than the T then that node can be malicious but in order to get confirmation of the same we need to send a dummy reply packet (DRP). The FDMM will send a DRP to the suspected node and if that node start sending the reply with in the stipulated time (wait_timer) then we can trust that node and we will set the status of that node as 0. Now let's take case if we do not get a reply from suspected node we will consider that node as malicious node and status of that node will set as -1.

Isolation phase

Once the node has been identified as a malicious node, the FDMM need to alert all the nodes in the network so that communication can been stopped from that malicious node. For this purpose, the algorithm will send a control message containing the id of that

node and its status to inform the source node and other nodes in the network. Once the source node and other nodes will get that control message they will stop accepting the fake RREQ packets hence network will be saved from flooding and congestion will be controlled.

The main objective is to detect and isolate the nodes which are attempting the flooding attack in the network there by creating the congestion hence making the legitimate data packets to drop. The entire process of the proposed algorithm is explained in figure 5.3

The algorithm of proposed work (FDMM) is explained below:

Algorithm 5.1 : *Flooding Detection and Mitigation Mechanism (FDMM)*

1: Initialize the network of n nodes.
2: Initiate the route discovery process using underlying algorithm.
3: After a time frame 't' ,for all the nodes in the network:
Observe the number of route request forwarded by the nodes in the network and represent it
by $TRREQ_i$

4: Calculate the mean of the RREQ packets broadcasted by all the intermediate nodes as per the
and represent it by T

$$T = \sum_{i=1}^{n} \frac{RREQ_i}{n}$$

5: Check the malicious behavior of the node:

If($TRREQ_i \leq T$) then:

- Node is not a malicious node and no chances of congestion due to flooding

Else if ($TRREQ_i > T$) then:

- Node can be a malicious node.
- To confirm send a dummy_reply packet to the node.

If(got reply with in wait_timer) then:

- Node is not malicious node and we can continue with routing process

Else

- Node is identified as a malicious node and can lead to congestion by flooding the network with route request packets.
- Send a control message to all the intermediate nodes regarding existence of malicious nodes.
- Isolate the malicious node from further communication.

6: Repeat the steps 2 to 5 until simulation ends.

5.5 Complexity analysis of Flooding detection and Mitigation Mechanism (FDMM)

The time complexity of FDMM can be calculated by considering the fact that whether there is an attack on the network. The algorithm will come into action as soon as flooding attack happens on the network. If there is no flooding attack on the network, then it can be considered as a best case analysis and underlying AODV algorithm will perform the route discovery process for finding the route from source to sink node. With respect to this point, the time required for routing the traffic from source to sink node will be a function of number of nodes in the network so we can conclude that the time complexity will be O (n).

Let us consider a case when there is flooding attack in the network, in this case the network performance will start degrading and we have to apply FDMM to detect and mitigate the attacker node to isolate the attacker node from the network so that network performance can be enhanced. In this case we can consider the worst case analysis of our algorithm. For implementing our algorithm we have to use one more loop for calculating the route request broadcasted by all the nodes in the network in addition to finding the route from source to sink node hence the inner loop for calculating the same will depend upon outer loop which is running for number of 'n' nodes in the network. So theoretically we have two nested loop out of which one will take care of number of nodes in the network and another for calculating the route request broadcasted by all the nodes in the network. Hence the time complexity of the said algorithm will be O (n^2)

The time complexity analysis has been summarized in the following table also:

Table 5.1: Time complexity of Flooding Detection and Mitigation Mechanism

Time Complexity of Flooding Detection and Mitigation Mechanism FDMM	
Best Case	Worst Case
O(n)	O(n^2)

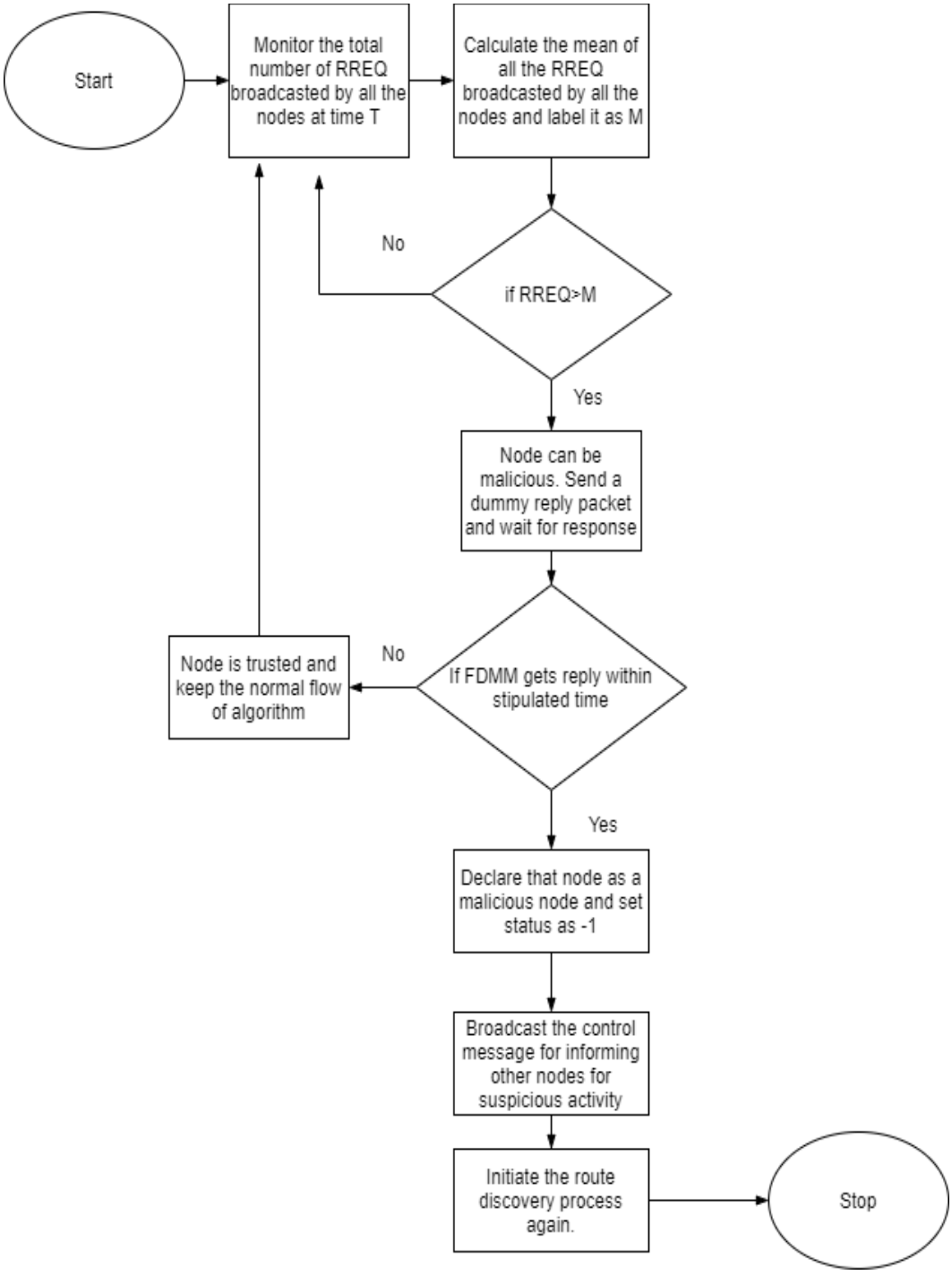

Figure 5.3: Flowchart depicting the proposed algorithm (FDMM)

5.6 Simulation setup and Result analysis

We have used NS 2.33 simulator for validating the effectiveness of proposed algorithm-FDMM. We initiated a route request flooding attack against the native algorithm. In the area of 100M X 100M we have randomly distributed the sensor node with a single static source node and single static sink node. Two nodes have been taken as a malicious node which starts flooding the network with route request packets. The network has been simulated for 100 Sec over the 10, 20,30,40,50 numbers of nodes. The traffic rate has been chosen as 50 Kbps as a CBR UDP packets. The various parameters assumed and taken for the simulation are listed in table 5.2

The performance of FDMM algorithm has been evaluated by using various performance metrics like average packet delivery fraction, average throughput, average residual energy and average end to end delay. The flooding attack will be launched by 2 malicious nodes in the network at 20 sec and will flood the network with multiple route request packets thereby creating the congestion in the network. We have considered the 3 malicious nodes for our proposed algorithm to check it effectiveness and more number of malicious nodes can be considered in future. The wait_timer is the time in sec for which algorithm should for the reply from node to confirm it being malicious. We have taken this value by performing a number of simulation and average time in which non malicious send the reply is less than 2 sec. The congestion in the network will start dropping the packets and legitimate packets will not be able to reach the destination node thereby decreasing throughput, packet delivery fraction and increasing energy consumption and end to end delay. So first of all we have implemented native AODV. Then we have implemented the proposed algorithm and results have been compared with existing state of art algorithm i.e. gill et al. [108] and AIF-AODV [111].

Table 5.2 Simulation Parameters and Rules

Parameter	Value
Area	100M X 100M
Protocol	AODV
Simulation	100 s
Number of nodes (Varying)	10,20,30,40,50
Malicious node	2
Wait_timer	2 s
Connection	UDP
Packet size	1024 bytes
Traffic type	CBR
Propagation Model	Two way around

5.6.1 Performance Metrics

Following performance metrics have been used for evaluating the performance of proposed algorithm FDMM. AWK scripts have been used for extracting the value of a particular metric by analyzing the trace file generated. These metrics are:

5.6.1.1 Average Packet Delivery Ratio Analysis

It is the ratio of total packets received by the sink node to total packets created by the source node. It is given by the following formula:

$$PDR = \frac{Recieved\_packets}{Sent\_packets} * 100 \quad (5.2)$$

Where PDR is packet delivery ratio, Received_packets are the total number of packets received by the sink node and Sent_packets are the total number of packets generated by the source node.

As shown in the figure 5.4, the average PDR of proposed algorithm has been compared with native AODV and two state of art algorithms i.e. gill et al. and

AIF_AODV. The PDR is highest when there is no attack on the network. PDR seems to be low in FDMM as network was attacked by malicious node but as shown in the figure the PDR of FDMM is above than gill et.al and AIF_AODV algorithm. This is due to the reason that FDMM uses the statistical concept of calculating mean of route requests generated by all the nodes in the network for setting the overall threshold value. On the contrary, gill et.al uses the coordinator node to transfer the status of malicious node by calculating number of route request generated by a node and AIF_AODV do not employ any mathematical technique for setting the threshold. It's just allowing the nodes to drop the packets when a particular node gets RREQ packets greater than predefined limit of a node and rest of the packets will be dropped. On an average, proposed algorithm shows 1.34% and 0.18% increase in PDF when compared with gill et al. and AIF-AODV respectively.

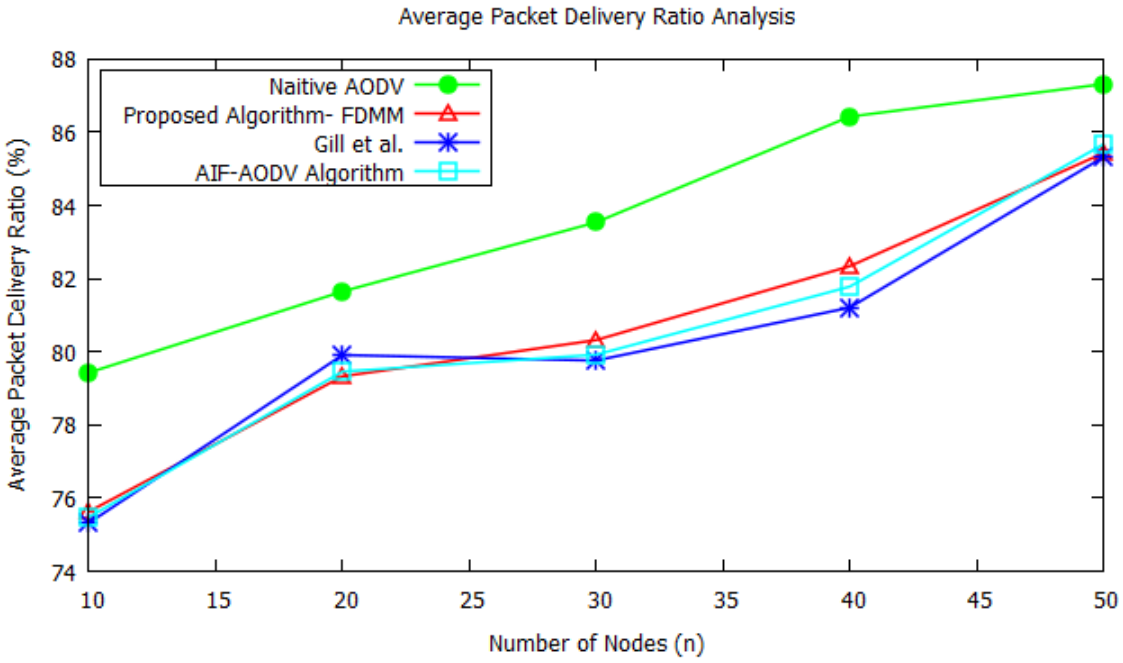

Figure 5.4 Average Packet delivery fractions versus number of nodes

Table 5.3: Simulation result of Average packet delivery fraction

Average Packet Delivery Fraction (%)				
Number of nodes	**Native AODV**	**Proposed Algorithm - FDMM**	**Gill et.al. [74]**	**AIF-AODV**
10	79.43	75.63	75.32	75.48
20	81.65	79.33	75.91	79.46
30	83.54	80.32	79.96	79.92
40	86.43	82.34	81.21	81.78
50	87.32	85.43	85.32	85.67

5.6.1.2 Average Throughput Analysis

Throughput represents the rate at which the data packets received by the sink node which was generated by source node over the period of simulation time. The formula for calculating throughput is given by:

$$Throughput = \frac{Recieved\_Packets}{End_{time} - Start_{time}} \quad (5.3)$$

Where Recived_Packets are the total number of packets received over the period of simulation time. End_{time} and $Start_{time}$ is the simulation time taken in general.

As shown in the figure 5.5, Native AODV shows the highest throughput as there is no attack on the network. Proposed algorithm – FDMM outperforms the gill et al. [74] and AIF-AODV in terms of throughput. The simulation results have also been shown in the table 5.3. It is due to the statistical approach used in the FDMM which helps us to identify the flooding attack and preventing the same as compared to state of art algorithms which do not use any statistical approaches for handling the said network adversary. On an average, the proposed algorithm-FDMM shows 6.03% and 5.47% increase in throughput when compared with gill et al. and AIF-AODV respectively. The results have also been tabulated in table number 5.4.

Table 5.4: Simulation result of Average Throughput Analysis

Average Throughput Analysis (%)				
Number of nodes	**Native AODV**	**Proposed Algorithm - FDMM**	**Gill et.al. [74]**	**AIF-AODV**
10	868	789	746	756
20	954	831	801	794
30	989	881	842	856
40	1076	943	897	900
50	1134	1087	987	990

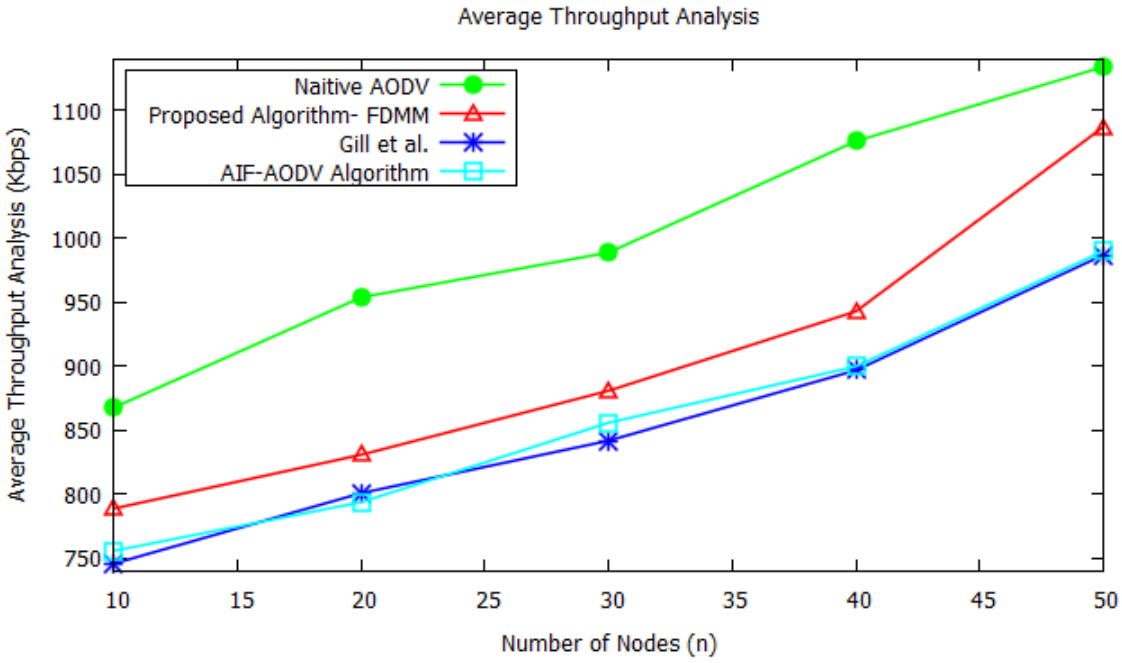

Figure 5.5 Average Throughput Analysis versus number of nodes

5.6.1.3 Average Residual Energy Analysis

It is the average of energy left for all the nodes in the network after simulation. It can be calculated by using the following equation:

$$ARL = \sum_{i=1}^{n} \frac{E_i^{initial} - E_i^{Consumed}}{n} \quad (5.4)$$

As shown in the figure 5.6, Average residual energy is very less at the end of simulation in case of gill et al. [74] algorithm and AIF-AODV shows an improvement over the gill et al. algorithm. However proposed algorithm – FDMM shows the more residual energy consumption when compared with state of art algorithm due to mathematical nature of algorithm which does not depend on number of route request sent by a particular node rather takes the mean of route requests forwarded by all the nodes and then threshold is decided. Obviously Native AODV will show the better residual energy among all the algorithms compared due to no attack in the network. On an average , the proposed algorithm-FDMM shows 1.93% and 0.54% increase in average residual energy when compared with gill at al. and AIF-AODV respectively. The simulation results have been listed in the table 5.5 also

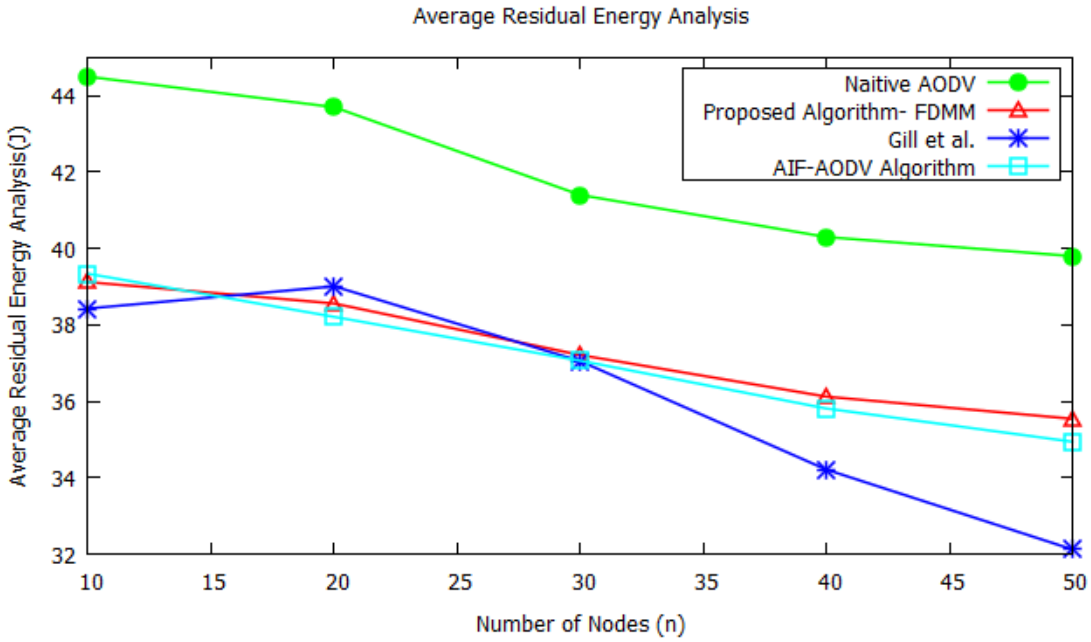

Figure 5.6 Average Residual Energy versus number of nodes

Table 5.5: Simulation result of Average Residual Energy

Average Residual Energy Analysis (%)				
Number of nodes	**Native AODV**	**Proposed Algorithm - FDMM**	**Gill et.al. [74]**	**AIF-AODV**
10	44.5	39.1	38.4	39.3
20	43.7	38.5	39.0	38.2
30	41.4	37.1	37.0	37.0
40	40.3	36.1	34.2	35.8
50	39.8	33.5	32.1	34.9

5.6.1.4 Average End to End Delay

The average end-to-end delay is the time it takes packets to travel from source node to sink node. It is also the indication of congestion level in the network. It can be calculated as:

$$Delay = \sum_{i=1}^{n} \frac{Recieved_{time} - Sent_{time}}{N} \quad (5.5)$$

As shown in the figure 5.7, proposed algorithm – FDMM outperforms the gill et al. and AIF-AODF algorithm in terms of average end to end delay. This is due to the reason that FDMM tries to identify and quickly resolves the flooding attack. Due to this there is less congestion in the network over the time and over all end to end delay of the network is decreased. Lack of mathematical calculations in the gill et al. and AIF-AODV will lead to more delays in the network as these algorithms are not capable of quickly resolving the attack thereby increasing the delay. And off course native AODV shows least delay in the network as there is no attack occurs in the network while implementing the same. On an average, the proposed algorithm shows 3.30% and 4.09% decrease in end to end delay when compared with gill et al. and AIF-AODV respectively. The simulation results have also been listed in the table 5.6

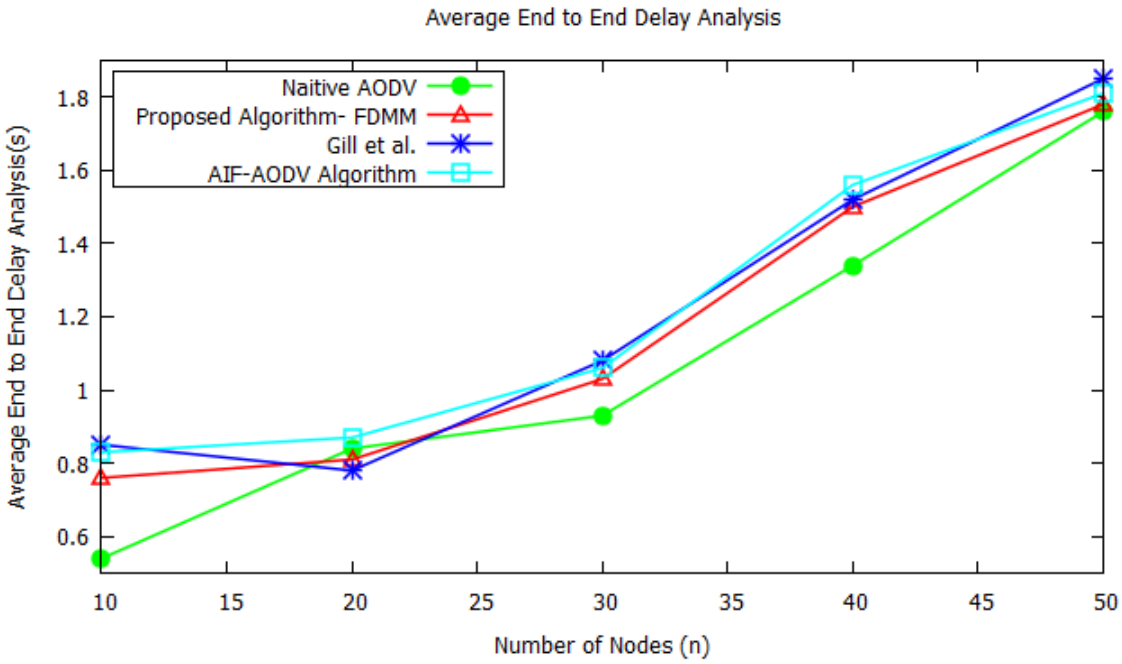

Figure 5.7 Average End to End delay versus number of nodes

Table 5.6: Simulation results of Average End to End Delay

Average End to End Delay Analysis (ms)				
Number of nodes	**Native AODV**	**Proposed Algorithm - FDMM**	**Gill et.al. [74]**	**AIF-AODV**
10	0.54	0.76	0.85	0.83
20	0.84	0.81	0.78	0.87
30	0.93	1.03	1.08	1.06
40	1.34	1.50	1.52	1.56
50	1.76	1.78	1.85	1.81

5.7 Conclusion of the Chapter

In this chapter, we have devised a novel algorithm to detect and isolate the malicious node which is responsible for flooding the network with fake RREQs packets using flooding attack. Because flooding is a denial of service attack which then tends to overburden the network with fictitious route request packets, causing congestion, it is extremely crucial to diagnose and segregate the nodes causing flooding. The impact of congestion is such that it can hamper the natural flow of packet by either dropping the legitimate packets or by delaying the packets from reaching the sink node. This chapter proposed a novel algorithm to detect and isolate the malicious node causing the flooding attack. The proposed algorithm works in 3 phase's threshold calculation, detection phase and isolation phase. The proposed algorithm FDMM is simulated and compared with native AODV and two state of art algorithms i.e. gill et al. and AIF-AODV. It has been observed from the simulation results that the proposed algorithm outperforms in terms of packet delivery fraction, throughput, average residual energy and average end to end delay.

CHAPTER 6

Conclusion and Future Scope

6.1 Conclusion and Future Work

One of the most prominent challenges in WSNs is congestion control. Congestion in the WSNs degrades the performance of network there by dropping the packets in the event of congestion occurrence. So it is highly desirable to detect and control the congestion as soon as possible to make the networks work properly. There are number of techniques available for detecting and controlling the congestion in WSNs as mentioned in literature survey. Some of the techniques focus either on traffic control or on resource control for controlling the congestion in WSNs. As seen in literature survey some authors have also purposed the congestion controlling techniques using some intelligent algorithm. Secondly fairness is also one of the techniques that can help us on controlling the congestion and network adversaries can also increase the chances of congestion in the network. By looking at the various perspectives, this thesis has proposed 3 objectives for controlling the congestion in different situations described below.

In the first objective we have used the resource control techniques for controlling the congestion. In the proposed algorithm - CDPEA, initially we have created a congestion detection mechanism and in the event of congestion, the proposed algorithm tries to mitigate the congestion. The proposed algorithm used updated Ant Colony Optimization for choosing the optimal path in the event of congestion for routing the sensitive traffic from best path. There is always number of nodes available for routing the traffic until it reaches the sink node. The algorithm will calculate the path preference probability for each neighbor node and node with maximum path preference probability will be selected for routing the packet until it reaches the sink node. So the proposed algorithm will try to choose the optimal or best path among number of available paths for routing the packets belonging to sensitive traffic. The proposed algorithm have been simulated by using a well know simulator NS 2.33 and is compared existing state of art

algorithms and it has been found from simulation that the proposed algorithm outperformed them in terms of throughput, energy consumption, end to end delay and packet loss rate.

Achieving a fair and efficient transmission rate to all the nodes in WSNs is very challenging due to its event driven nature. In the second objective, we propose a congestion control scheme named FEACC for WSNs. This objective unveiled a novel mechanism for reducing congestion and achieving reasonably fair bandwidth allocation in WSNs which is normally resource constrained. The main idea of the proposed work is to detect the congestion and in to alleviate the effect of congestion by adaptively allocating the fair rate to the nodes in the network. We indicated through simulations that FEACC outperforms the existing state of art protocols. We have simulated the proposed algorithm by varying the number of nodes and by varying the offered traffic load. The proposed scheme has been evaluated on network parameters throughput, packet delivery fraction, average energy consumption and end to end delay.

In the third objective, we have devised an algorithm named FDMM to detect and isolate the malicious node which is responsible for flooding the network with fake RREQs packets. Initially we have discussed the literature survey based on flooding attacks in networks then we have shown the impact of flooding attack on the network. Because flooding is a denial of service attack which then tends to overburden the network with fictitious route request packets, causing congestion, it is extremely crucial to diagnose and segregate the nodes causing flooding. The impact of congestion is such that it can hamper the natural flow of packet by either dropping the legitimate packets or by delaying the packets from reaching the sink node. This chapter proposed a novel algorithm to detect and isolate the malicious node causing the flooding attack. The proposed algorithm works in 3 phase's threshold calculation, detection phase and isolation phase. The proposed algorithm is simulated and compared existing state of art algorithms. It has been observed from the simulation results that the proposed algorithm outperforms in terms of packet delivery fraction, throughput, average residual energy and average end to end delay.

Main drawback of the proposed solutions in the thesis is: Firstly the proposed solution works only for static sensor nodes. However, in real life problems we can have sensor nodes that can be mobile. So, in future we will try to implement the proposed algorithms for mobile sensor nodes. Secondly, proposed solutions have not been implemented on Internet of Things and Fog computing environment. In future we will implement proposed algorithms on internet of things and fog computing environments. Lastly, In future we will try to increase the number of malicious nodes to see the effect on network performance and accordingly try to update the proposed algorithm.

BIBLIOGRAPHY

[1] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. Computer networks, 38(4), 393-422.

[2] Kim, B. S., Park, H., Kim, K. H., Godfrey, D., & Kim, K. I. (2017). A survey on real-time communications in wireless sensor networks. Wireless communications and mobile computing, 2017.

[3] Pau, G., Ferrero, R., Jennehag, U., & Zhang, H. (2019). Emerging applications through low-power wireless technologies for Internet of Things.

[4] Kumar, R., & Reichert, F. (2011, March). Towards a layer-less network architecture—A case from wireless sensor networks. In 2011 2nd International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE) (pp. 1-9). IEEE.

[5] Aponte-Luis, J., Gómez-Galán, J. A., Gómez-Bravo, F., Sánchez-Raya, M., Alcina-Espigado, J., & Teixido-Rovira, P. M. (2018). An efficient wireless sensor network for industrial monitoring and control. Sensors, 18(1), 182.

[6] Singh, H., & Singh, D. (2018, December). Concentric layered architecture for multi-level clustering in large-scale wireless sensor networks. In 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC) (pp. 467-471). IEEE.

[7] Zhang, Y., & Cheng, L. (2003, September). Cross-layer optimization for sensor networks. In New York Metro Area Networking Workshop, New York.

[8] Pešović, U. M., Mohorko, J. J., Benkič, K., & Čučej, Ž. F. (2010). Single-hop vs. Multi-hop–Energy efficiency analysis in wireless sensor networks. In 18th telecommunications forum, TELFOR.

[9] Anisi, M. H., Abdullah, A. H., & Abd Razak, S. (2011). Energy-efficient data collection in wireless sensor networks. Wireless Sensor Network, 3(10), 329.

[10] Bala, T., Bhatia, V., Kumawat, S., & Jaglan, V. (2018). A survey: issues and challenges in wireless sensor network. Int. J. Eng. Technol, 7(2), 53-55.

[11] Sharma, S., Bansal, R. K., & Bansal, S. (2013, December). Issues and challenges in wireless sensor networks. In 2013 International Conference on Machine Intelligence and Research Advancement (pp. 58-62). IEEE.

[12] Bala, T., Bhatia, V., Kumawat, S., & Jaglan, V. (2018). A survey: issues and challenges in wireless sensor network. Int. J. Eng. Technol, 7(2), 53-55.

[13] Sharma, S., Bansal, R. K., & Bansal, S. (2013, December). Issues and challenges in wireless sensor networks. In 2013 international conference on machine intelligence and research advancement (pp. 58-62). IEEE.

[14] Hasan, M. Z., Al-Turjman, F., & Al-Rizzo, H. (2018). Analysis of cross-layer design of quality-of-service forward geographic wireless sensor network routing strategies in green internet of things. IEEE Access, 6, 20371-20389.

[15] Kandris, D., Nakas, C., Vomvas, D., & Koulouras, G. (2020). Applications of wireless sensor networks: an up-to-date survey. Applied System Innovation, 3(1), 14.

[16] Arampatzis, T., Lygeros, J., & Manesis, S. (2005, June). A survey of applications of wireless sensors and wireless sensor networks. In Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control, 2005. (pp. 719-724). IEEE.

[17] Durišić, M. P., Tafa, Z., Dimić, G., & Milutinović, V. (2012, June). A survey of military applications of wireless sensor networks. In 2012 Mediterranean conference on embedded computing (MECO) (pp. 196-199). IEEE.

[18] Bokareva, T., Hu, W., Kanhere, S., Ristic, B., Gordon, N., Bessell, T., & Jha, S. (2006, October). Wireless sensor networks for battlefield surveillance. In Proceedings of the land warfare conference (pp. 1-8).

[19] Lim, H. B., Ma, D., Wang, B., Kalbarczyk, Z., Iyer, R. K., & Watkin, K. L. (2010, June). A soldier health monitoring system for military applications. In 2010 International Conference on Body Sensor Networks (pp. 246-249). IEEE.

[20] Naz, P., Hengy, S., & Hamery, P. (2012, May). Soldier detection using unattended acoustic and seismic sensors. In Ground/Air Multisensor Interoperability, Integration, and

Networking for Persistent ISR III (Vol. 8389, p. 83890T). International Society for Optics and Photonics.

[21] Khedo, K. K., Bissessur, Y., & Goolaub, D. S. (2020). An inland Wireless Sensor Network system for monitoring seismic activity. Future Generation Computer Systems, 105, 520-532.

[22] Lopes Pereira, R., Trindade, J., Gonçalves, F., Suresh, L., Barbosa, D., & Vazão, T. (2014). A wireless sensor network for monitoring volcano-seismic signals. Natural Hazards and Earth System Sciences, 14(12), 3123-3142.

[23] Cugati, S., Miller, W., & Schueller, J. (2003, June). Automation concepts for the variable rate fertilizer applicator for tree farming. In The proceedings of the 4th European conference in precision agriculture (pp. 14-19).

[24] Nikolidakis, S. A., Kandris, D., Vergados, D. D., & Douligeris, C. (2015). Energy efficient automated control of irrigation in agriculture by using wireless sensor networks. Computers and Electronics in Agriculture, 113, 154-163.

[25] Ma, C., Wang, Y., & Ying, G. (2011, April). The pig breeding management system based on RFID and WSN. In 2011 Fourth International Conference on Information and Computing (pp. 30-33). IEEE.

[26] Andonovic, I., Michie, C., Gilroy, M., Goh, H. G., Kwong, K. H., Sasloglou, K., & Wu, T. (2009, September). Wireless sensor networks for cattle health monitoring. In International Conference on ICT Innovations (pp. 21-31). Springer, Berlin, Heidelberg.

[27] Nikokheslat, H. D., & Ghaffari, A. (2017). Protocol for controlling congestion in wireless sensor networks. Wireless Personal Communications, 95(3), 3233-3251.

[28] Ghaffari, A. (2015). Congestion control mechanisms in wireless sensor networks: A survey. Journal of network and computer applications, 52, 101-115.

[29] Bohloulzadeh, A., & Rajaei, M. (2020). A survey on congestion control protocols in wireless sensor networks. International Journal of Wireless Information Networks, 27(3), 365-384.

[30] Kafi, M. A., Djenouri, D., Othman, J. B., Ouadjaout, A., & Badache, N. (2014). Congestion detection strategies in wireless sensor networks: A comparative study with testbed experiments. Procedia computer science, 37, 168-175.

[31] Pandey, D., & Kushwaha, V. (2020). An exploratory study of congestion control techniques in Wireless Sensor Networks. Computer Communications, 157, 257-283.

[32] Ghaffari, A. (2015). Congestion control mechanisms in wireless sensor networks: A survey. Journal of network and computer applications, 52, 101-115.

[33] Wan, C. Y., Campbell, A. T., & Krishnamurthy, L. (2002, September). PSFQ: a reliable transport protocol for wireless sensor networks. In Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications (pp. 1-11).

[34] Ren, F., He, T., Das, S. K., & Lin, C. (2011). Traffic-aware dynamic routing to alleviate congestion in wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems, 22(9), 1585-1599.

[35] Hull, B., Jamieson, K., & Balakrishnan, H. (2004, November). Mitigating congestion in wireless sensor networks. In Proceedings of the 2nd international conference on Embedded networked sensor systems (pp. 134-147).

[36] Wang, G., & Liu, K. (2009, September). Upstream hop-by-hop congestion control in wireless sensor networks. In 2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications (pp. 1406-1410). IEEE.

[37] Tao, L., & Yu, F. (2011, March). A novel congestion detection and avoidance algorithm for multiple class of traffic in sensor network. In 2011 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (pp. 72-77). IEEE.

[38] Kafi, M. A., Djenouri, D., Othman, J. B., Ouadjaout, A., Bagaa, M., Lasla, N., & Badache, N. (2014). Interference-aware congestion control protocol for wireless sensor networks. Procedia Computer Science, 37, 181-188.

[39] Jaiswal, S., & Yadav, A. (2013, August). Fuzzy based adaptive congestion control in wireless sensor networks. In 2013 Sixth International Conference on Contemporary Computing (IC3) (pp. 433-438). IEEE.

[40] Sergiou, C., Vassiliou, V., & Paphitis, A. (2013). Hierarchical Tree Alternative Path (HTAP) algorithm for congestion control in wireless sensor networks. Ad hoc networks, 11(1), 257-272.

[41] Aghdam, S. M., Khansari, M., Rabiee, H. R., & Salehi, M. (2014). WCCP: A congestion control protocol for wireless multimedia communication in sensor networks. Ad Hoc Networks, 13, 516-534.

[42] Antoniou, P., Pitsillides, A., Blackwell, T., Engelbrecht, A., & Michael, L. (2013). Congestion control in wireless sensor networks based on bird flocking behavior. Computer Networks, 57(5), 1167-1191.

[43] Rezaee, A. A., Yaghmaee, M. H., Rahmani, A. M., & Mohajerzadeh, A. H. (2014). HOCA: Healthcare aware optimized congestion avoidance and control protocol for wireless sensor networks. Journal of Network and Computer Applications, 37, 216-228.

[44] Banimelhem, O., & Khasawneh, S. (2012). GMCAR: Grid-based multipath with congestion avoidance routing protocol in wireless sensor networks. Ad Hoc Networks, 10(7), 1346-1361.

[45] Heikalabad, S. R., Ghaffari, A., Hadian, M. A., & Rasouli, H. (2011). DPCC: dynamic predictive congestion control in wireless sensor networks. International Journal of Computer Science Issues (IJCSI), 8(1), 472.

[46] Sergiou, C., Vassiliou, V., & Paphitis, A. (2014). Congestion control in wireless sensor networks through dynamic alternative path selection. Computer Networks, 75, 226-238.

[47] Misra, S., Tiwari, V., & Obaidat, M. S. (2009). LACAS: Learning automata-based congestion avoidance scheme for healthcare wireless sensor networks. IEEE Journal on Selected Areas in Communications, 27(4), 466-479.

[48] Narawade, V., & Kolekar, U. D. (2018). ACSRO: Adaptive cuckoo search based rate adjustment for optimized congestion avoidance and control in wireless sensor networks. Alexandria engineering journal, 57(1), 131-145.

[49] Yadav, S. L., Ujjwal, R. L., Kumar, S., Kaiwartya, O., Kumar, M., & Kashyap, P. K. (2021). Traffic and energy aware optimization for congestion control in next generation wireless sensor networks. Journal of Sensors, 2021.

[50] Sangeetha, G., Vijayalakshmi, M., Ganapathy, S., & Kannan, A. (2018). A heuristic path search for congestion control in WSN. In Industry Interactive Innovations in Science, Engineering and Technology (pp. 485-495). Springer, Singapore.

[51] Ding, W., Tang, L., & Ji, S. (2016). Optimizing routing based on congestion control for wireless sensor networks. Wireless Networks, 22(3), 915-925.

[52] Raman, C. J., & James, V. (2019). FCC: Fast congestion control scheme for wireless sensor networks using hybrid optimal routing algorithm. Cluster Computing, 22(5), 12701-12711.

[53] Rezaee, A. A., & Pasandideh, F. (2018). A fuzzy congestion control protocol based on active queue management in wireless sensor networks with medical applications. Wireless Personal Communications, 98(1), 815-842.

[54] Onthachi, D., & Jayabal, S. (2017). An optimized QoS-based multipath routing protocol for wireless sensor networks. IJIES, 11(2), 49-56.

[55] Yogarajan, G., & Revathi, T. (2018). Improved cluster based data gathering using ant lion optimization in wireless sensor networks. Wireless Personal Communications, 98(3), 2711-2731.

[56] Kumar, S., Dave, M., & Dahiya, S. (2014). ACO based QoS aware routing for wireless sensor networks with heterogeneous nodes. In Emerging trends in computing and communication (pp. 157-168). Springer, New Delhi.

[57] Cheng, J., Zhang, G., Li, Z., & Li, Y. (2012). Multi-objective ant colony optimization based on decomposition for bi-objective traveling salesman problems. Soft Computing, 16(4), 597-614.

[58] Hackel, S., Fischer, M., Zechel, D., & Teich, T. (2008, July). A multi-objective ant colony approach for pareto-optimization using dynamic programming. In Proceedings of the 10th annual conference on Genetic and evolutionary computation (pp. 33-40).

[59] Prasad, S., & Lobiyal, D. K. (2013, February). Multiobjective multicast routing in wireless ad hoc networks-An Ant Colony approach. In 2013 3rd IEEE International Advance Computing Conference (IACC) (pp. 511-514). IEEE.

[60] Liu, Y., Yang, Z., Ning, T., & Wu, H. (2014). Efficient quality-of-service (QoS) support in mobile opportunistic networks. IEEE Transactions on Vehicular Technology, 63(9), 4574-4584.

[61] Sun, Y., & Tian, J. (2010). WSN path optimization based on fusion of improved ant colony algorithm and genetic algorithm. Journal of Computational Information Systems, 6(5), 1591-1599.

[62]

[63] Tshiningayamwe, L., Lusilao-Zodi, G. A., & Dlodlo, M. E. (2016). A priority rate-based routing protocol for wireless multimedia sensor networks. In Advances in nature and biologically inspired computing (pp. 347-358). Springer, Cham.

[64] Woo, A., & Culler, D. E. (2001, July). A transmission control scheme for media access in sensor networks. In Proceedings of the 7th annual international conference on Mobile computing and networking (pp. 221-235).

[65] Akan, O. B., & Akyildiz, I. F. (2005). Event-to-sink reliable transport in wireless sensor networks. IEEE/ACM transactions on networking, 13(5), 1003-1016.

[66] Rangwala, S., Gummadi, R., Govindan, R., & Psounis, K. (2006). Interference-aware fair rate control in wireless sensor networks. ACM SIGCOMM Computer Communication Review, 36(4), 63-74.

[67] Paek, J., & Govindan, R. (2010). RCRT: Rate-controlled reliable transport protocol for wireless sensor networks. ACM Transactions on Sensor Networks (TOSN), 7(3), 1-45.

[68] Yin, X., Zhou, X., Huang, R., Fang, Y., & Li, S. (2009). A fairness-aware congestion control scheme in wireless sensor networks. IEEE transactions on vehicular technology, 58(9), 5225-5234.

[69] Hua, S. (2014). Congestion control based on reliable transmission in wireless sensor networks. Journal of Networks, 9(3), 762.

[70] Jan, M. A., Nanda, P., He, X., & Liu, R. P. (2014). PASCCC: Priority-based application-specific congestion control clustering protocol. Computer networks, 74, 92-102.

[71] Li, G., Li, J., & Yu, B. (2012, March). Lower bound of weighted fairness guaranteed congestion control protocol for WSNs. In 2012 Proceedings IEEE INFOCOM (pp. 3046-3050). IEEE.

[72] Grover, A., Kumar, R. M., Angurala, M., Singh, M., Sheetal, A., & Maheswar, R. (2022). Rate aware congestion control mechanism for wireless sensor networks. Alexandria Engineering Journal, 61(6), 4765-4777.

[73] Tambe, S. B., & Gajre, S. S. (2018). Novel Strategy for Fairness-Aware Congestion Control and Power Consumption Speed with Mobile Node in Wireless Sensor Networks. In Smart Trends in Systems, Security and Sustainability (pp. 85-111). Springer, Singapore.

[74] Pan, W., Tan, H., Li, X., & Li, X. (2021). Improved rtt fairness of bbr congestion control algorithm based on adaptive congestion window. Electronics, 10(5), 615.

[75] Kim, G. H., Song, Y. J., Mahmud, I., & Cho, Y. Z. (2019, July). Enhanced BBR congestion control algorithm for improving RTT fairness. In 2019 eleventh international conference on ubiquitous and future networks (ICUFN) (pp. 358-360). IEEE.

[76] Esmaeelzadeh, V., Hosseini, E. S., Berangi, R., & Akan, O. B. (2016). Modeling of rate-based congestion control schemes in cognitive radio sensor networks. Ad Hoc Networks, 36, 177-188.

[77] Srivastava, V., Tripathi, S., Singh, K., & Son, L. H. (2020). Energy efficient optimized rate based congestion control routing in wireless sensor network. Journal of Ambient Intelligence and Humanized Computing, 11(3), 1325-1338.

[78] Wei, W., Xue, K., Han, J., Xing, Y., Wei, D. S., & Hong, P. (2020). BBR-based congestion control and packet scheduling for bottleneck fairness considered multipath TCP in heterogeneous wireless networks. IEEE Transactions on Vehicular Technology, 70(1), 914-927.

[79] Kazmi, H. S. Z., Javaid, N., Imran, M., & Outay, F. (2019, April). Congestion control in wireless sensor networks based on support vector machine, Grey Wolf optimization and differential evolution. In 2019 Wireless Days (WD) (pp. 1-8). IEEE.

[80] Chughtai, O., Badruddin, N., & Awang, A. (2016, August). A novel congestion alleviation procedure in multi-hop wireless sensor networks. In 2016 6th International Conference on Intelligent and Advanced Systems (ICIAS) (pp. 1-6). IEEE.

[81] Brahma, S., Chatterjee, M., & Kwiat, K. (2010, March). Congestion control and fairness in wireless sensor networks. In 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops) (pp. 413-418). IEEE.

[82] Nasser, N., & Chen, Y. (2007). SEEM: Secure and energy-efficient multipath routing protocol for wireless sensor networks. Computer communications, 30(11-12), 2401-2412.

[83] Murthy, S., D'Souza, R. J., & Varaprasad, G. (2012). Digital signature-based secure node disjoint multipath routing protocol for wireless sensor networks. IEEE Sensors Journal, 12(10), 2941-2949.

[84] Sangeetha, R., & Yuvaraju, M. (2012, December). Secure energy-aware multipath routing protocol with transmission range adjustment for wireless sensor networks. In 2012 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-4). IEEE.

[85] Kumar, S., & Jena, S. (2010, December). SCMRP: Secure cluster based multipath routing protocol for wireless sensor networks. In 2010 Sixth International conference on Wireless Communication and Sensor Networks (pp. 1-6). IEEE.

[86] Triki, B., Rekhis, S., & Boudriga, N. (2010, July). A novel secure and multipath routing algorithm in wireless sensor networks. In 2010 International Conference on Data Communication Networking (DCNET) (pp. 1-10). IEEE.

[87] Alrajeh, N. A., Alabed, M. S., & Elwahiby, M. S. (2013). Secure ant-based routing protocol for wireless sensor network. International journal of distributed sensor networks, 9(6), 326295.

[88] Bok, K., Lee, Y., Park, J., & Yoo, J. (2016). An energy-efficient secure scheme in wireless sensor networks. Journal of Sensors, 2016.

[89] Salmi, S., & Oughdir, L. (2022). CNN-LSTM Based Approach for Dos Attacks Detection in Wireless Sensor Networks. International Journal of Advanced Computer Science and Applications, 13(4).

[90] Premkumar, M., & Sundararajan, T. V. P. (2020). DLDM: Deep learning-based defense mechanism for denial of service attacks in wireless sensor networks. Microprocessors and Microsystems, 79, 103278.

[91] Sahoo, R. R., Singh, M., Sahoo, B. M., Majumder, K., Ray, S., & Sarkar, S. K. (2013). A light weight trust based secure and energy efficient clustering in wireless sensor network: honey bee mating intelligence approach. Procedia Technology, 10, 515-523.

[92] Pajila, P. J., Julie, E. G., & Robinson, Y. H. (2022). FBDR-Fuzzy based DDoS attack Detection and Recovery mechanism for wireless sensor networks. Wireless Personal Communications, 122(4), 3053-3083.

[93] Premkumar, M., & Sundararajan, T. V. P. (2021). Defense countermeasures for DoS attacks in WSNs using deep radial basis networks. Wireless Personal Communications, 120(4), 2545-2560.

[94] Raghav, R. S., Thirugnansambandam, K., & Anguraj, D. K. (2020). Beeware routing scheme for detecting network layer attacks in wireless sensor networks. Wireless Personal Communications, 112(4), 2439-2459.

[95] Singh, O., Singh, J., & Singh, R. (2017). SAODV: statistical ad hoc on-demand distance vector routing protocol for preventing mobile adhoc network against flooding attack. Advances in Computational Sciences and Technology, 10(8), 2457-2470.

[96] Pandikumar, T., & Desta, H. (2017). RREQ flooding attack mitigation in MANET using dynamic profile based technique. International Journal of Engineering Science, 12700.

[97] Jatthap, S., & Dashore, P. (2016). Battery capacity based detection and prevention of flooding attack on MANET. International Journal of Advance Research in Computer Science and Management Studies (IJARCSMS), 4(9), 89-99.

[98] Kumar, S., Alaria, S., & Kumar, V. (2015). Prevention in sleep deprivation attack in MANET. International Journal of Latest Technology in Engineering (IJLTEMAS), 4(2), 139-144.

[99] Gurung, S., & Chauhan, S. (2018). A novel approach for mitigating route request flooding attack in MANET. Wireless Networks, 24(8), 2899-2914.

[100] Singh, P., Raj, A., & Chatterjee, D. (2012). Flood Tolerant AODV Protocol (FT-AODV). International Journal of Computer Applications, 9758887.

[101] Panos, C., Xenakis, C., Kotzias, P., & Stavrakakis, I. (2014). A specification-based intrusion detection engine for infrastructure-less networks. Computer Communications, 54, 67-83.

[102] Abdelshafy, M. A., & King, P. J. (2014). Resisting flooding attacks on AODV. SECURWARE, 25.

[103] Jiang, F. C., Lin, C. H., & Wu, H. W. (2014). Lifetime elongation of ad hoc networks under flooding attack using power-saving technique. Ad Hoc Networks, 21, 84-96.

[104] Faghihniya, M. J., Hosseini, S. M., & Tahmasebi, M. (2017). Security upgrade against RREQ flooding attack by using balance index on vehicular ad hoc network. Wireless Networks, 23(6), 1863-1874.

[105] Ketshabetswe, L. K., Zungeru, A. M., Mangwala, M., Chuma, J. M., & Sigweni, B. (2019). Communication protocols for wireless sensor networks: A survey and comparison. Heliyon, 5(5), e01591.

[106] Sharma, V., & Grover, A. (2016). A modified ant colony optimization algorithm (mACO) for energy efficient wireless sensor networks. Optik, 127(4), 2169-2172.

[107] Camilo, T., Carreto, C., Silva, J. S., & Boavida, F. (2006, September). An energy-efficient ant-based routing algorithm for wireless sensor networks. In International

workshop on ant colony optimization and swarm intelligence (pp. 49-59). Springer, Berlin, Heidelberg.

[108] Gill, R. K., & Sachdeva, M. (2018). Detection of hello flood attack on LEACH in wireless sensor networks. In Next-generation networks (pp. 377-387). Springer, Singapore.

[109] Wang, Y., Attebury, G., & Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks.

[110] Almomani, I., & Al-Kasasbeh, B. (2015, April). Performance analysis of LEACH protocol under Denial of Service attacks. In 2015 6th International Conference on Information and Communication Systems (ICICS) (pp. 292-297). IEEE.

[111] Abu Zant, M., & Yasin, A. (2019). Avoiding and isolating flooding attack by enhancing AODV MANET protocol (AIF_AODV). Security and Communication Networks, 2019.

[112] Nalayini, C. M., Katiravan, J., & Prasad, A. (2017). Flooding attack on MANET–a survey. International Journal of Trend in Research and Development (IJTRD), 25-27

[113] Pandikumar, T., & Desta, H. (2017). RREQ flooding attack mitigation in MANET using dynamic profile based technique. International Journal of Engineering Science, 12700.

[114] Rao, D. S., & Padmanabhuni, V. (2016). An efficient RREQ flooding attack avoidance technique for adaptive wireless network. International Journal of Applied Engineering Research (IJAER), 11(5), 3696-3702.

[115] https://eden.dei.uc.pt/~tandre/antsense/

Publications

1. Makul Mahajan and Dr. Sukhkirandeep Kaur (2021). Congestion Control Protocols in Wireless Sensor Networks: a comprehensive Survey. In 2020 International Conference on Intelligent Engineering and Management (ICIEM) (pp. 160-164). IEEE. **(Scopus Indexed)**

2. Makul Mahajan and Dr. Sukhkirandeep Kaur (2021). An Intelligent Path Evaluation Algorithm for Congestion Control in Wireless Sensor Networks. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(6), 3106-3114. **(Scopus indexed)**
3. Makul Mahajan and Dr. Sukhkirandeep Kaur (2021). A Fair and Efficient Rate Allocation Based Congestion Control Protocol in Wireless Sensor Networks. In 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 1-5). IEEE. **(Scopus Indexed)**
4. Makul Mahajan and Dr. Mritunjay Kumar Rai (2016). The Comparative Analysis of RED, GF-RED and MGF-RED for Congestion Avoidance. International Journal Of Control Theory And Applications, 41(9), 157-164. **(Scopus Indexed)**