IDENTIFICATION AND ENHANCEMENT OF DEVICE CONNECTIVITY ISSUES IN SCALABLE IOT

Dissertation submitted in fulfillment of the requirements for the degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

SYKAM V NARENDRA KUMAR

11501320

Supervisor

Mr. RAVISHANKER

School of Computer Science and Engineering

Lovely Professional University

Phagwara, Punjab (India)

June, 2017

PAC FORM

TOPIC APPROVAL PERFORMA

School of Computer Science and Engineering

Program : P172::M.Tech. (Computer Science and Engineering) [Full Time]

COURSE CODE : CSE546 **REGULAR/BACKLOG :** Regular **GROUP NUMBER :** CSERGD0020

Supervisor Name : Ravishanker **UID :** 12412 **Designation :** Assistant Professor

Qualification : ______________________ **Research Experience :** ______________________

SR.NO.	NAME OF STUDENT	REGISTRATION NO	BATCH	SECTION	CONTACT NUMBER
1	Sykam V Narendra Kumar	11501320	2015	K1518	8099996298

SPECIALIZATION AREA : Networking and Security **Supervisor Signature:** __________________

PROPOSED TOPIC : Identification and Enhancement of device connectivity issues in scalable IoT

Qualitative Assessment of Proposed Topic by PAC		
Sr.No.	**Parameter**	**Rating (out of 10)**
1	Project Novelty: Potential of the project to create new knowledge	6.80
2	Project Feasibility: Project can be timely carried out in-house with low-cost and available resources in the University by the students.	6.80
3	Project Academic Inputs: Project topic is relevant and makes extensive use of academic inputs in UG program and serves as a culminating effort for core study area of the degree program.	6.80
4	Project Supervision: Project supervisor's is technically competent to guide students, resolve any issues, and impart necessary skills.	7.00
5	Social Applicability: Project work intends to solve a practical problem.	6.80
6	Future Scope: Project has potential to become basis of future research work, publication or patent.	7.00

PAC Committee Members		
PAC Member 1 Name: Prateek Agrawal	UID: 13714	Recommended (Y/N): Yes
PAC Member 2 Name: Pushpendra Kumar Pateriya	UID: 14623	Recommended (Y/N): Yes
PAC Member 3 Name: Deepak Prashar	UID: 13897	Recommended (Y/N): Yes
PAC Member 4 Name: Kewal Krishan	UID: 11179	Recommended (Y/N): Yes
PAC Member 5 Name: Anupinder Singh	UID: 19385	Recommended (Y/N): NA
DAA Nominee Name: Kanwar Preet Singh	UID: 15367	Recommended (Y/N): Yes

Final Topic Approved by PAC: **Identification and Enhancement of device connectivity issues in scalable IoT**

Overall Remarks: Approved

PAC CHAIRPERSON Name: 11011::Dr. Rajeev Sobti **Approval Date:** 26 Oct 2016

ABSTRACT

As there is a rapid expansion of connected devices in the world, IoT has a huge potential to change the topologies and it should be flexible towards the scalability and there is a need to analyze the node connectivity issues which needs to be resolved. Hence in the view of scalability, various factors influencing the network failure and node connectivity should be detected earlier and thus enhance the communication among the nodes. This report discussed the different communication modes in IoT and how the 'things' are being transformed to meet the changing requirements in dynamic environment. WSN in association with IoT has more scope for expansion of existing network. Hence the study is focused on device connectivity issues, in the sense, maintaining the efficient link stability and formation of clusters in the network. This report has discussed the importance of various components of IoT, approaches for intelligent D2D communication, node coverage issues: identification and repairing of lost connectivity, role of cloud in IoT, importance of middleware for providing QoS, and various clustering techniques. This report has presented an enhancement of the existing HMPC algorithm by introducing a metric called Strength of Device Synergy (SDS) for link stability and Firefly algorithm based clustering technique for choosing the best suitable cluster head (CH) aiming at increasing the network lifetime and to reduce the packet loss ratio.

DECLARATION STATEMENT

I hereby declare that the research work reported in the dissertation entitled "**IDENTIFICATION AND ENHANCEMENT OF DEVICE CONNECTIVITY ISSUES IN SCALABLE IOT**" in partial fulfillment of the requirement for the award of Degree for Master of Technology in Computer Science and Engineering at Lovely Professional University, Phagwara, Punjab is an authentic work carried out under supervision of my research supervisor **Mr. Ravishanker**. I have not submitted this work elsewhere for any degree or diploma.

I understand that the work presented herewith is in direct compliance with Lovely Professional University's Policy on plagiarism, intellectual property rights, and highest standards of moral and ethical conduct. Therefore, to the best of my knowledge, the content of this dissertation represents authentic and honest research effort conducted, in its entirety, by me. I am fully responsible for the contents of my dissertation work.

Signature of Candidate

Sykam V Narendra Kumar

11501320

ACKNOWLEDGEMENT

No research can be done in an isolated environment and this, certainly was not an exception. It was a concerted effort of all my friends, family and above all me. I would like to thank everyone from core of my heart.

I would like to express the deepest appreciation to my mentor, **Mr. Ravishanker**, who has the attitude and the substance of a genius. He always helped to clear all doubts generated during different parts of this literature review and formulation of statement for my research work. His guidance is also a motivation for me to do work on time. His guidance was crucial for formulation of problem statement and carry forward approach.

SUPERVISOR'S CERTIFICATE

This is to certify that the work reported in the M.Tech Dissertation entitled "**IDENTIFICATION AND ENHANCEMENT OF DEVICE CONNECTIVITY ISSUES IN SCALABLE IOT**", submitted by **Sykam V Narendra Kumar** at **Lovely Professional University, Phagwara, India** is a bonafide record of his original work carried out under my supervision. This work has not been submitted elsewhere for any other degree.

Signature of Supervisor

Ravishanker

Counter Signed by:

1) **Concerned HOD:**

HOD Signature: ________________

HOD Name: ____________________

Date: __________________________

2) **Neutral Examiners:**

External Examiner

Signature: ______________________

Name: _________________________

Affiliation: _____________________

Date: __________________________

Internal Examiner

Signature: ______________________

Name: _________________________

Date: __________________________

TABLE OF CONTENTS

CONTENTS	PAGE NUMBER

LIST OF TABLES

LIST OF FIGURES

SUPERVISOR'S CHECKLIST

Name: Ravishanker **UID:** 12412 **Domain:** Networking and Security

Registration No.: 11501320 **Name of student:** Sykam V Narendra Kumar

Title of Dissertation: Identification and Enhancement of Device Connectivity Issues in Scalable IoT

- ☐ Front pages are as per the format.
- ☐ Topic on the PAC form and title page are same.
- ☐ Front page numbers are in roman and for report; it is like 1, 2, 3…….
- ☐ TOC, List of Figures, etc. are matching with the actual page numbers in the report.
- ☐ Font, Font Size, Margins, line Spacing, Alignment, etc. are as per the guidelines.
- ☐ Color prints are used for images and implementation snapshots.
- ☐ Captions and citations are provided for all the figures, tables etc. and are numbered and center aligned.
- ☐ All the equations used in the report are numbered.
- ☐ Citations are provided for all the references.
- ☐ Objectives are clearly defined.
- ☐ Minimum total number of pages of report is 50.
- ☐ Minimum references in report are 30.

Here by, I declare that I have verified the above mentioned points in the final dissertation report.

Signature of Supervisor with UID

CHAPTER 1

INTRODUCTION

The varieties of data of which billions of current IoT based devices exchange are not quite the same as the conventional Internet – in any event the Internet we have known about it since the 1980s. Consider the internet with two unique elements; one is places in the cyberspace, and the other is individuals in the same [1]. The places could be space dispensed to a human as well as a thing i.e., a place for people and place for things. "Things" can be the collection of web pages in the web site. A web place requires not containing the web server(s) alone, but rather it is a virtual place that contains a few servers or a solitary shareable server that connects with different spots (people or things). The feeling of exploring or moving from one site page to the next is created solely by the substance, yet not by the intercession of people. At present, the notion of a human is non-existent on the web. This could be the early phase of modern transition towards IoT due to the non-human mediation of communication among things.

IoT is named as System of Systems [2] which can be alluded to as an highly enhanced and finish last phase of transition that is framed out of legitimately incorporating and uniting individual and autonomous frameworks with the goal of offering any time and any place on-request qualitative services. Machine-to-machine (M2M) communication [3] is made conceivable in IoT. For this to happen, minimal packaging and presentation overhead is required. The individual transmission from a large portion of the IoT devices could be considered as uncritical, for instance, for basic sensor devices, the varieties in conditions may be negligible. All these individual transmissions are gathered and prepared at some place in the system. For quick understanding, consider a moisture sensor that may need to transmit a solitary incentive about volumetric substance of water. For this, a couple of characters of information is required that may incorporate extra information of area or ID tag, consequently expecting a moderate change in the value, however the recurrence of important updates will be less. On account of complex devices, for example, remote monitoring framework, minimal high traffic will be made, yet in short arrangements obscure to people, is again gathered and afterward translated by other IoT devices. Overall, the measure of significant information produced from each individual devices is too small that utilizes low data transmission, cost saving, ideal use of battery etc, which is the pattern practically inverse to that of conventional Internet.

The technology transition in IoT can be alluded as fiction-to-reality transition because of fast increment of intelligent devices that are prepared to intermingle with each other for intelligent and non-human mediated communication. For a system to be enhanced or to be newly presented, it demands change on effectively existing procedures, or need to acquaint new methods to meet the objectives of IoT. The "things" that necessities such change are different tools and applications, gateways, and services that are required to the system and given by the system. The following figure demonstrates the overview of such 'things'.

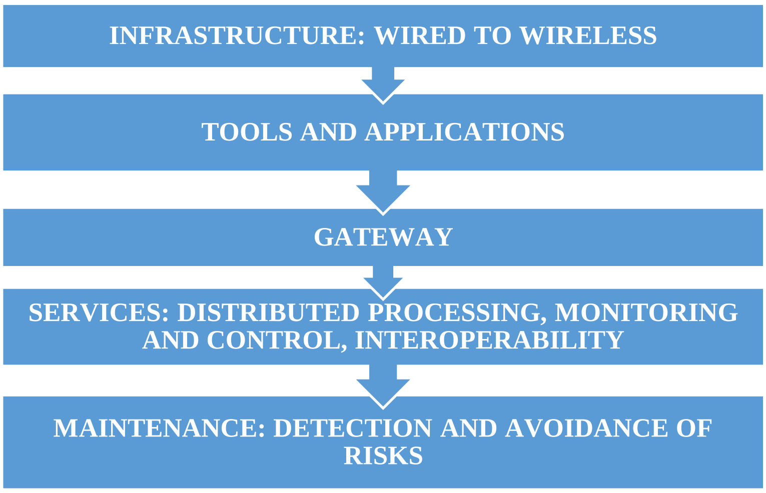

Figure 1: Various 'things' participating in IoT

The transition from wired to wireless infrastructure which can be alluded as Cable-to-radio technology transition, could be seen around nineteenth century in which the term "radio" got its own recognition, despite the fact that fiber-optic cables has its own importance. With the introduction of wireless telegraph receiver [4] used to get the electric signals from the transmitter, it turns into a definitive methods for communication till date. Research is as yet being done in the field of radio communication. Tools and applications give simplicity to work and simple to utilize GUIs to clients with the introduction of automation of 'things'. The trend has definitely been changing from manual setting/embeddings conventions/rules into a device to a total intelligent adapting of conventions on-request, for instance, fuzzy rule-based intelligent system in this manner IoT applications bolster the detecting of devices in a network intelligently to maintain a strategic distance from more power utilization, more transmission capacity, and more delay of information delivery.

The concept of intelligently configured gateways make the network secure from the interruptions. For instance, DOG Gateway [5] which offers building blocks to bolster the changing needs of home automation making a beeline for another Intelligent Domotic Environment (IDE). It hence bolsters different services like distributed data processing, monitoring and control of data stream in secure paths, interoperability, and maintenance. Support is the significant service that occasionally needs human intercession to ensure that all the security systems are working legitimately. It incorporates discovery and avoidance of security risks into the system.

The applications of IoT are not restricted to any particular domain. It spreads over many simple and large complex domains such as smart energy [6], smart water [7], smart education [8], [9], smart agriculture [10], smart vehicles [11], [12], so on. This paper is limited with the discussion of various protocols being used in such IoT domains.

IoT offers a practical resource scheduling by guaranteeing their maximal use and service to the customers. IoT gets features from various computing environments, for example, mobile computing, WSNs, and pervasive computing. From this, obviously IoT depends on upon various essential capabilities and developments like machine learning, signal handling/processing, real time computing, data analysis in big data analytics, data curation, information sharing, information storage, information exchanging, and representation. Security and assurance concerns are everywhere, which are authoritative scopes of data correspondence where threats and attacks are to be perceived and resolved.

A vital undertaking to be done in IoT communication is to change over the distinguished or accumulated data into keen information. Alongside this, the gadgets in IoT have limit of perceiving and detailing the region, history over the remote associations. For an aggregate operation of an IoT framework, rest of the functionalities are data gathering, association and communication, computational limit, and data storage and handling, where most by far of the devices are of vitality obliged and compelled memory which are real trying issues that ought to be settled.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Different Types Of Device/Human Communication In IoT

In [13], the authors described a brief overview on how an intelligent D2D Communication will be accomplished in IoT environment. Likewise they concentrated on accomplishing D2D Communication in IoT utilizing state-of-the-art routing algorithms. As people are being interconnected as well as devices prefer sensors, brilliant home devices, RFID [14] labels, and so forth., for accumulation, and sharing of data utilizing arrangement of communication between them which may include people. In IoT environment, devices communicate with each other utilizing Internet protocols and standards, where devices are power and memory compelled which brings about different systems administration issues that couldn't be settled utilizing traditional routing protocols. For an IoT framework to work viably there is a requirement for the devices to bolster intelligent routing protocols.

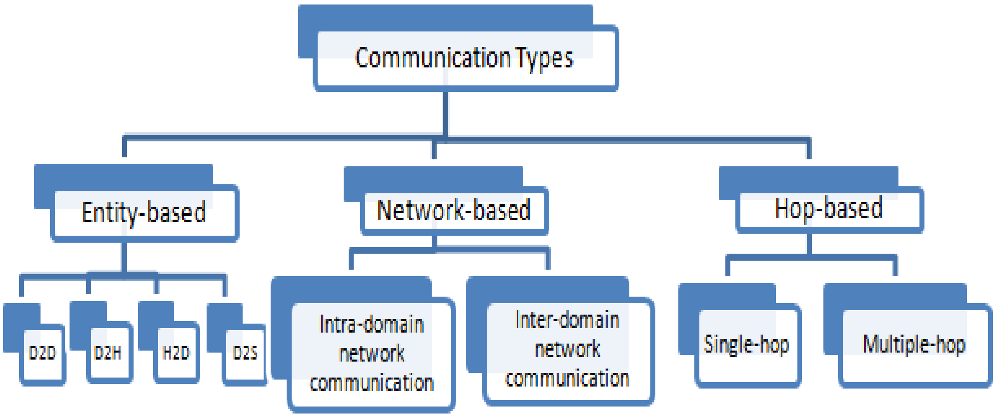

Figure 2: Possible types of communication in IoT environment

According to the author, as mentioned in the above figure, there are three possible ways of communications in the IoT environment.

2.1.1 Entity-based communication

Entity could be any device or human that includes in the system. D2D communication without human mediation/intervention is essentially the majority of the circumstances to continue for additionally activities, for example, distinguishing and finding another device in this way devices could intelligently decide. Simply saying, the device ought to have the property of link discovery i.e., the device ought to have the capacity to discover its neighboring devices inside its communication range and can know about the environment in which it is being worked and additionally collaborates with other device which would be participating in the information management and different exercises like self-automation, controlling and observing its own procedures to synchronize one device's activities to that of other device's in this manner performing information exchange and sending. The author in this paper focused on such communication.

Some vital qualities of D2D IoT communication are heterogeneity of devices, concurrence and joint effort among devices, device limitations, for example, battery life-time, memory and handling limit, self-outline and self-affiliation and self-association, non-unsurprising device portability designs, and multihop communication . D2H communication can likewise be alluded to as "D2D communication with human intercession" which is important to pass on the obliged data to the people and in this manner setting off a caution for his/her basic leadership, so as the H2D communication. D2S communication includes communication of any device with the information stockpiling specialist to send the gathered information for additionally processing and storing.

2.1.2 Network-based communication

Intra-domain network communication includes different devices inside the system itself which impart each other to accumulate the data and to share their state among them. In inter-domain network communication, assortments of devices communicate with each other utilizing gateway routing protocols. Both of the situations may include heterogeneous devices which vary in transmission and handling capacity, memory and battery life also. Consequently the communication procedure in IoT should bolster heterogeneous device and system communication. The operational order of IoT systems:

Constrained network contains devices that have low battery life, less memory and poor information rate. Cases are short-run remote systems like Wireless Personal Area Networks (WPANs) and Wireless Local Area Networks (WLANs), low-control lossy system (LLN, for example, low-bit rate WPAN, delay-torrent network which can be deployed in rockets, submerged activities and so on., and remote sensor systems (WSNs).

Unconstrained network devices: Unlike the previous kind of devices, here the devices have no such confinements on power utilization, and memory too.

2.1.3 Hop-based communication

In single hop communications, every device interact with each other by means of system infrastructure i.e., via access points or base stations. In multi-hop communications, devices will transfer information to each other for accomplishing end-to-end communication among source and target devices.

2.2 Approaches for D2D Communication

Devices can hand-off the information to alternate devices with or without including a systems infrastructure. Likewise it can utilize licensed and unlicensed spectrum. The procedure of link discovery and authentication is simple if devices belong to the same network providers. Or communication ought to be set up utilizing unlicensed range. Henceforth D2D communication will be accomplished through two methodologies.

2.2.1 Network-Infrastructure-Dependent Communication

This was proposed for a cellular network with a specific end goal to expanding the reusability of the licensed frequency spectrum in this way to build the system capacity. Typically the link discovery process requires broadcasting of signals which devours additional time and vitality. In any case, these issues can be settled by the framework which will set aside a few minutes expending and high vitality proficient connection finding process by hinting close-by devices and distributing time and recurrence for exchanging the signals.

This kind of communication has no automated security components for device verification at the season of association foundation for which human intercession is required. Four operations modes were proposed for the devices to diminish reusability of range and impedance issues. Silent mode is best when the system couldn't ensure bolster for conveying of activity in light of inadequacy of assets and additionally when there is high obstruction because of range reuse. Reuse mode offers ideal using of otherworldly assets which in this manner causing obstruction in the system. In Dedicated mode, few range assets are devoted to D2D correspondences. Cellular mode offers a typical method for movement hand-off through the base station.

Restrictions of this sort of D2D communication are the devices should subscribe to cellular network and to a similar administrator, and the usage is constrained because of heterogeneity of devices.

2.2.2 Network-Infrastructure-Independent Communication

A Qualcomm project known as FlashLinQ taken a shot at D2D communication over authorized range without utilizing a systems administration framework in which connect building up was finished utilizing an idea called paging (it is a procedure of starting the communication) in this manner giving security of communication after link establishment among approved devices.

This approach utilizes intermittent detecting of recurrence range whether it is sit still or not. Along these lines authorized range can be utilized powerfully. Unlicensed range comes into picture if there is a need of self-sorting out, and remains solitary D2D communication is to be set-up that needs no framework on whenever, anyplace the length of the devices prepared for communication in which interface revelation and validation procedures ought to be finished by the devices. To make D2D communication in a clever way, supportive intelligent algorithms are to be required. For instance: Stochastic/Probabilistic Algorithms, Bio-inspired algorithms like swarm intelligence based algorithms which can be the ant colony optimization (ACO), Hierarchical Algorithms like tree-based and cluster-based algorithms, and Context-Aware Algorithms.

2.3 RFID System Delay Optimization

Coverage of RFID nodes/devices to accurate reading of RFID tags for effective communication is one of the major challenges in wireless networks. Reasons for uncoverage could be missing of tags from its communication range. In [15], this challenge is considered to be solved using set-cover depending on approximation algorithm. The problem with a large numbers of RFID tag readers in a network is that the coverage area of the RFID readers may probably contains overlapping and interference among the readers due to range of communication area leads to increase of load on the network. Thus it is of great importance to choose and deploy the optimal number of RFID readers for maximum and accurate coverage of the network. For this problem to be solved, the author proposed a set-cover based approximation algorithm. The three important objectives of the algorithm for effective RFID coverage are:

- To choose minimal number of readers
- To reduce overlapping and interference among the reader's coverage area, and
- To balance the load among various readers

2.4 Identification and Repairing of Node Connectivity Loss in WSN

A latest algorithm called Heterogeneous Mobility Pattern Clustering (HMPC) algorithm [16] aims at repairing and reestablishing the connectivity of detached sub-clusters which are formed after any disturbance or damage caused to the connectivity in a large cluster of mobile sensor nodes thereby improving the performance in terms of packet loss ratio and throughput.

It considered DBSCAN clustering algorithm and Velocity-Based clustering algorithm to track and to determine the connectivity depletion among nodes. DBSCAN makes the HMPC algorithm to determine the clusters having weak connectivity followed by repairing them which involves three core functionalities namely making of connectivity period, cluster determination, repairing of connectivity.

The nodes in a dynamic network move randomly towards and/or away from each other to a distance with some velocity which are supposed to having same kind of communication and sensing range, and each nodes are aware of their location information used to track and repair connectivity issues. The spatial and temporal features are calculated using Cluster Head (CH) selection and Minimum Period of Connectivity (MPC) between neighbor nodes respectively.

MPC is the best candidate as a predefined parameter as it represents the ratio of remaining distance to stay connected to the velocity in case of an opposite direction movement among the neighbor nodes, based on which each node calculates their weights. Each node is associated with a minimum number of neighbor nodes N_{MPC}. The CH node should have its weight equal or approximately equal to zero which reveals that the MPC of CH node to each of its neighbor nodes is equal to the desired global minimum period of connection (MPC_g) of the network. The spatial feature can be termed as the direction in which the node moves in the cluster.

If the nodes move away from each other in the direction opposite to the movement of its neighbor node, then a cluster of nodes is said to have depletion in its connectivity with another departed cluster of nodes. Such identified connectivity depletions are then repaired and restored in which the border nodes uses connectivity repair algorithm and the remaining nodes are stayed as core nodes. Cluster head selection techniques are presented in [17]–[19].

Working of HMPC Algorithm is as follows:

Step 1: Identification of neighbor nodes of any desired node n

Step 2: Calculation of weight of the node n which is then broadcasted to its neighbor nodes

Step 3: Node n is elected as a cluster head (CH) node, if its weight is less than any of its neighbors'. Otherwise it becomes a border node which runs the connectivity repair algorithm.

Step 4: If any of the neighbor nodes is neither a CH nor a cluster member, then based on the MPC and $MPC_{g,}$ it is decided whether to include the neighbor node in the cluster or not. In this way, a cluster head (CH) node which guarantees the best connectivity among its neighbors in the course of cluster scope.

2.5 IoT Cloud Sensing Service Based on Device Location

A model scheme that can achieve notable improvement when comparing with the traditional wireless sensor networks (WSNs). In this proposed model, the IoT-cloud offers a sensing service which is on demand depending upon both interest as well as locations of particular mobile user, and also cloud plans a schedule of the sensor networks on-demand that relies on location of the mobile user. The above mentioned service can be named as sensing-as-a-service, which allows cloud to providing sensing data for many applications. The sensor nodes in this model are restricted so that those devices could transmit only the processing jobs to reduce the energy consumption. The on-demand sensing service put forward the following major advantages:

- Sensors work only when needed.
- Redundancy can be reduced by collecting the sensing data on demand.
- The quality of sensing service could be made depending upon interests of application users.

The proposed map functions:

Set of three major functions are given below which are used to merge the IoT enabled devices and the cloud depending upon locations of mobile users.

Function 1: Any applications needs registration with IoT-cloud for which function $f_1(\alpha_{SI})$ is used to mapping sensors of interest of the applications to set of definite sensor types $\tau^*_\alpha \subset \tau$ which is formulated as shown below. Thus IoT-cloud gives sensing services on-demand relied upon the locations of mobile users.

$$f_1(\alpha_{SI})= \tau^*_\alpha =(\tau_j : \tau_j \in \tau) \quad (1)$$

Function 2: If $\mu_L^{current} \in \alpha_{RI}$, then the cloud assigns a set of physical sensors $\xi^{\mu}{}_L^{current}$ whose types matches with τ^*_α surrounds the current locations of the user μ.

$$f_2(\mu_L^{current})= \xi\mu_L^{current} =(s_j : s_{j-\tau} \in \tau_\alpha^*) \text{ if } \mu_L^{current} \in \alpha_{RI} \quad (2)$$

Function 3: After identifying the resources of physical node in the locations, the cloud mapping those devices to a set υ of prevailed virtual sensors (i.e., if existed) or generates a fresh set of virtual sensors (i.e., if needed).

$$f_3(\xi\mu_L^{current})= \upsilon \quad (3)$$

The proposed on-demand IoT cloud-based scheduling algorithm for WSNs is described as follows:

1. Identify the current locations of mobile users μ.
2. The cloud maps that user to set of physically sensor nodes $\xi\mu_L^{current}$.
3. Cloud maps or creates a group of virtual sensors υ.
4. Cloud generates a schedule of sensor devices that matching with application of interest and locates at current location of the mobile users depending up on the applications' requirement (i.e., sensing interval, delay).
5. Cloud then sends scheduling request to respective base station BS.
6. BS then makes a broadcasting of scheduling request to sensor nodes.
7. Nodes after receiving the scheduling requests, all of them will set their own scheduling based upon the requests.

If user moves outside of the locations range, cloud then sends a request message to cancel the scheduled requests.

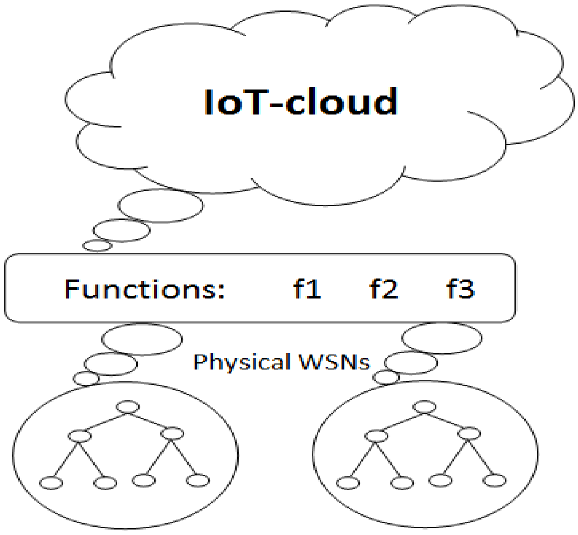

Figure 3: Location-based IoT-cloud integration using functions f1, f2, f3

When there occurs a longer sleep interval and where there is no need of data transmission, i.e., when there are no such serving of requests, this design helps in saving energy for restricted nodes. On the other hand, with low sleep interval, the design could provide better quality of sensing data to the application's as well as users. Hence major advantage of the design is that depending upon the cloud's knowledge about when and where the necessary sensing of services is needed, WSNs are planned in such a way that they save energy during non-processing time period and made available to serving the users on-demand with good quality of data based on application's requirement.

2.6 Importance of Middleware in IoT for Qualitative Services

Middleware in IoT [20] provides ease of making the development process of diverse applications and corresponding services where the task of doing so is a challenging task. It is because of large-scale network of devices, heterogeneity in network where multiple events are generated by the devices. In this scenario, middleware offers services that are common for the applications and thereby supports interoperability among variety of applications and the available services for the devices. A complete functional IoT middleware comprises the integration of four significant technologies, RFID, WSNs, M2M communication, and Supervisory Control and Data Acquisition (SCADA) from which few properties are inherited. For example, 'resource-constrained' property is inherited from RFID and Sensor Networks (SNs) technologies where as 'intelligence' property is from WSNs and M2M technologies. A middleware summarizes hardware complexities related to communication and computation thereby providing a layer among applications, operating system and network layer offers ease of interaction in the view of communication. It also provides a layer between system software and application software in the view of computation. The requirements for middleware are classified into two categories as follow:

Services that should be provided by middleware; for example resource discovering, resource managing, data managing, event managing, and code managing which comes under functional requirements. Non-functional requirements include factors of scaling, timeliness, reliable available, security, and ease-of-deployment.

Support that should be provided to system architecture; Examples include abstraction of programming concepts, interoperability, service-based support, adaptivity, context-awareness, self-governance and geographical distribution.

Following are various existing middleware solutions that could solve problems in IoT environment.

2.6.1 Event-Based Middleware

In this context, all the components and applications interact with each other through events where each and every event consists of type along with parameters whose value indicates change to the producer's state. The generated events at producer side (application component that sends or produces the event) are transferred to the consumer's side (application component that receives or consumes the event).

Message-oriented Middleware (MOM) is a kind of event-based middleware in which the communication is based on messages that contains additional metadata compared to events. Moreover, events use broadcasting to send the information but in MOM, message contains the address of both sender and receiver and intended receivers get the message. Event-Based Middleware [21] contains groups of subscribers and publishers which uses publish/subscribe messaging. Subscribers can access the publisher's data streams through a common database that are registered for specific events. Thus the subscribers receive subsequent and asynchronous event notifications. This design approach addresses various non-functional requirements like reliability, performance, scalability, availability, and security. Event-based middleware methodology supports strong decoupling of publishers and subscribers. Moreover, the properties mentioned above are not completely satisfactory. Hence there is a need of security and privacy to be provided using appropriate protocols and designs.

Examples of event-based middleware are: HERMES for scalability, interoperability and reliability, EMMA for availability, and autonomous, GREEN for reprogrammability and heterogeneity.

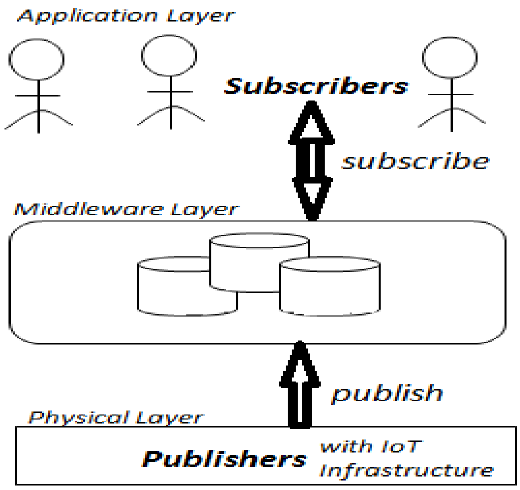

Figure 4: Event-Based Middleware with publish/subscribe messaging

2.6.2 Service-Oriented Middleware

The service-oriented design model develops applications in the form of services. Services-Oriented Computing (SOC) relies on services-oriented architecture (SOA) methodologies. The properties that are inherited from SOC are reusability, discoverability and composability of services, and technology neutrality. By implementing some functionalities/services such as data management service, service management service, service discovery, and QoS management, various challenges that were mentioned earlier like resource-constrained devices and mobility characteristics can be resolved. SOM model is beneficial because it provides adaptive service compositions in case of any unavailability of service.

Examples of service-oriented middleware are: HYDRA for dynamic self and reconfiguration, the MUSIC for QoS-aware and context-based dynamic adaptivity, the SOCRADES for accessibility, KASOM for reliability and efficiency.

2.6.3 VM-Based Middleware

The Virtual-Machine(VM) [22] based Middleware model offers programmability support to create a safely execution in environment to users applications by virtualizing the infrastructure where each node in the network contains a VM that can interpret small distributed modules that were formed from a large application will be injected and thus distributed to the whole network, thereby addresses various architectural requirements like programming abstraction, self-governance, and adaptivity by supporting transparency in distributed heterogeneous IoT system.

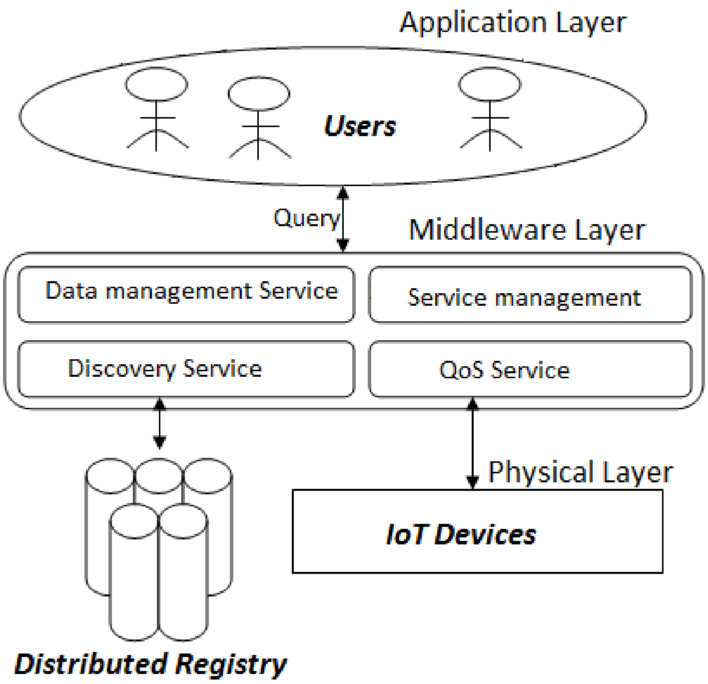

Figure 5: Service-Oriented Middleware design model

VMs are of two types: One is middleware-level VM where VMs are placed in between operating system (OS) and applications. Thus provides concurrency to the underlying OSs. And the other is system-level VM where each VM acts as a replacement or substitution of the complete operating system, and thus these VMs free up the consumed resources by other OSs.

Examples of VM-based middleware are: MATE for adaptability, ASVMs for overhead reduction.

2.6.4 Agent-Based Middleware

This methodology is similar to the VM-based methodology in which applications are divided into small programs in order to provide injection and distribution of the same into the network with the help of mobile agents. The decentralized systems will be able to tolerate partial failures in the network due to the maintenance of mobile agent's execution state. Mobile agents [23] gives more benefits by reducing the network load and network latency, asynchronous and autonomous execution , protocol encapsulation, robustness and fault tolerance, adaptivity and heterogeneity.

The use of agent-based systems reduces the designing complexities of various systems by providing some higher-level policies instead of direct administration. Examples of agent-based middleware are: IMPALA for resource management, openness, and scalability, Ubiware for flexibility, interoperability and scalability.

2.6.5 Tuple-Spaces Middleware

Each device has its own local tuples spacing structure. A tuple space is a data storage which could be accessing concurrently. All the tuple spaces form a large aggregated tuple space on a gateway.

This methodology is suitable for the mobile devices in IoT environment due to their short-term sharing of data. Hence the applications can write in and read from tuples in aggregated tuple space.

Examples of tuple-spaces middleware are: LIME for abstraction of programming, TS-Mid for data management.

2.6.6 Database-Oriented Middleware

In this methodology, the complete network can be viewed as a virtual relational database. Any interested application would access the database through queries.

Examples of database-oriented middleware are: SINA for resource management, IrisNet for heterogeneity, Sensation for scalability and energy-awareness.

2.6.7 Application-Specific Middleware

Application-specific or application-driven methodology concentrates on resources management (QoS) supporting for a particular applications or applications-domain by applying an architecture which polishes a network by changing, altering or adjusting the network configuration.

As studied earlier, there is a need to move a step ahead for the research in the area of requirements such as system-wide scalability, context-awareness, reliability, security and privacy, knowledge and intelligence integration, dynamic resource discovery and composition, and interoperability.

Examples of application-specific middleware are: MidFusion for information fusion, Limitations of this approach are, it does not create a general purpose solution but only for specialized purpose, and it does not address heterogeneity due to tight coupling between applications and middleware.

2.7 Node-to-Node Distance Calculation Techniques

There are plenty of algorithms that can be used to find out the distance between two nodes that are placed in a network. Optimized algorithms have their recognition in IoT. As optimization is the major goal of IoT. Minimum distance between two objects can be determined and computed by using combinatorial global optimization algorithms i.e., combination of algorithms like Simulated Annealing (SA) [24], Genetic Algorithms (GAs), etc. Speaking about scalability of IoT devices, natural phenomena helps us to choose the best suitable techniques and processes, for example, social insect colonies, pollen propagation, larval young dissemination, and others. There are various ways to calculate such distance which will be discussed further. Applications includes minimizing of the energy function, travelling salesman problem (minimizing the path length), safety engineering (analysis of Gibbs energy) etc.

2.8 Optimization of various performance parameters using FFA

Performance parameters can be optimized using Firefly algorithm (FFA) [25] by performing clustering in WSNs. WSNs has wide range of applications in which the entire network should be able to operate in the difficult-to-enter environment. In any hierarchical network structure, each cluster contains cluster head as well as normal sensor nodes. Cluster Heads (CH) always occupy the top level and the remaining cluster members at the lower level. This CH node gathers the data and transfers it to the base station either in a direct way or via other CH nodes. Finally through base station, the user can access the required data, which is located far away from the sensor nodes. The researcher focuses on energy efficient clustering in WSN.

Using the residual energy values and distance between the nodes, fitness function can be derived. Then depending on the function's optimal value, clustering will be performed.

$$\sum_{i=1}^{n}(w_1 \,.\, E_i + w_2 \,.\, d) \tag{4}$$

Where w_1, w_2 are weights satisfies the condition; $w_1+w_2=1$

Residual energy is denoted by E_i

And the distance between nodes by d

The phenomenon of bioluminescence is the fundamental concept of fireflies. Bioluminescence is nothing but the process of yielding flashes of less duration by fireflies to attract other fireflies. By this we can say that the flash intensity is the significant parameter which obeys the following three rules:

Rule 1: Fireflies attract other fireflies irrespective of their gender.

Rule 2: Depending upon the brightness of the flash emitted by the fireflies, the duration of attraction varies.

Rule 3: The brightness of fireflies is determined by using the objective function.

Below is the structure of Firefly Algorithm in which flash brightness is termed as attractive factor which depends on light intensity.

1. Initializing all the variables
2. f(x) is the objective function where $x=(x_1,........,x_d)$
3. Initializing the fireflies x_i (i=1,2 ,..., n)
4. Calculating the Light Intensity I_i at x_i
5. While (t <MaxGen); where MaxGen is the maximum generations
6. for i = 1 to n
7. for j = 1 to n
8. If ($I_j > I_i$),
9. Firefly i : j
10. end if
11. Attractiveness factor = Exp [$-\gamma r^2$], where γ represents the coefficient of light absorption, r is the distance between the fireflies
12. Update the value of intensity
13. End for j;
14. End for i;
15. End while;

Intensity of light is inversely proportional to distance 'd' from the light source. This relation proves that light intensity increase as the distance from the light source decrease and vice versa.

$$I(n) = I_0.\exp(-\alpha d_{ij}\string^(2)) \quad (5)$$

Where, I is the light intensity

I_0 is its initial value

α is the light absorption coefficient

d distance between firefly i and j

By rule number 3 as mentioned above, these values of distance and residual energy will be provided to the objective function. Fitness value is then calculated for performing the network clustering. Each node in the network broadcasts its energy value to the remaining nodes in the network, then light intensity value will be calculated by the receiver nodes using the following formula. Higher the value of light intensity, higher will be the attraction of fireflies so as to form the clusters. Finally the node with highest energy will be the cluster head (CH) and it will be changed after each cycle.

2.9 FFA in Grid Computing

The firefly algorithm has turned out to be a best suitable meta-heuristics searching technique on persistent optimization issues. Standard firefly algorithm can't be used to deal with discrete issues specifically as its positions are of real numbers. Numerous researchers tackled discrete optimization issues by applying adjusted nature motivated meta-heuristics optimization strategies. The exploration in applying the smallest position value rule (SPV) [26] for refreshing the positions of the fireflies in which everyone benefits from the standard firefly algorithm are reserved. Numerous researchers have implemented SPV in optimization issues to change over the persistent position values to discrete permutations. In this segment the proposed firefly algorithm for grid scheduling issue is discussed; the appeal of the firefly is portrayed, and the development towards the brighter fireflies is examined. The portrayal of firefly algorithm for network planning issue is a basic component for getting a sensible output.

In all optimization approaches, one of the key issues in outlining a fruitful firefly algorithm is the representation technique which tries to locate an appropriate mapping between problem, solution and the firefly algorithm. Every firefly denotes a candidate solution of the grid scheduling issue in vector formation, with n components; where n is the quantity of jobs to be planned. Firefly[i] indicates the resource to which the job number i is assigned. In this way, the vector values are natural numbers. Likewise we take note of that the vector values are the resource IDs and henceforth the resource ID may seem more than once in the firefly vector. This comes into picture because of the reason that more than one jobs may be assigned to a similar resource.

2.10 Fireflies in forming groups for clustering the data

Information clustering is a standout amongst the most vital methods in information analysis. Despite the fact that the k-means clustering technique has been generally utilized because of its effectiveness of execution, the execution of the strategy relies on initial solutions, having the downside of getting locally optimal solution. In [27], the researcher to explain this issue, have proposed an data clustering technique in light of the firefly algorithm joined with the k-means clustering strategy for data clustering. In our technique, the firefly algorithm initially endeavors to locate the semi-optimal solution. At that point, given the solutions gotten by the firefly algorithm as an underlying solution, k-means technique make data clustering converging to the initial solution, or last clustered data set. Results showed that the proposed technique can be powerful for information clustering utilizing some well known data sets.

2.11 Role of Firefly Algorithm in WSN Routing

Firefly algorithm is a nature propelled algorithm for advancement that should be investigated further to understand a number of issues to demonstrate its genuine potential. Routing is an outstanding strategy for drawing out the life of WSNs. The Routing algorithm is produced with new fitness function which is based upon residual energy and node degree as well as distance. The proposed algorithms are widely tested upon different contexts to demonstrate its performance and contrasted with various algorithms, for example, EADC, DHCR and also Hybrid Routing. Experimental outcomes [28] shows that proposed algorithms performs better as contrasted with the existing techniques.

Euclidean Distance: It can be defined as the Euclidean separation between the cluster head (CH) to the next-hop and from there to the base station (BS). In the case that the distance is least then it will cost less energy. Subsequently, second target is to reduce the distance among CH and BS. It will improve the lifetime of the system.

Distance and the corresponding movement: In every cycle, firefly with low brightness move towards the firefly with high brightness and the position of every firefly will be updated. In the algebraic computation, if newly updated position may not fulfill the range i.e. negative or higher than m + 1. In such scenario, position will be replaced by any random number produced in the vicinity of 1 and m + 1. All the procedures repeat iteratively until it reaches the greatest number of iterations.

To select the path of routing from each CH to BS, both are major tasks in WSN. The purpose for that computation is that complex nature extended with the size of the problem. In the present study, firefly algorithm has been embraced to take care of WSN related issues. A firefly based routing solution was proposed by considering residual energy, distance as well as node degree as its fitness function deduction. The proposed algorithms were broadly tried with some existing calculations in various scenarios, for example, HF, EADC and also DHCR. During this experiment study, FARW is discovered aggressive/superior in the vast majority of the situations when contrasted with existing ones.

2.12 Firefly Algorithm for Image Segmentation

Image segmentation is an essential step in the space of image processing in which we divide the image into a few sections which convey certain sort of data for the client. Image segmentation is exceptionally troublesome step in image processing which goes for extricating the data from image. Clustering helps in fragmenting the image. Clustering algorithms are a piece of data mining algorithm that gathers the information into different number of given clusters. Every data points in each cluster have comparable properties in light of which they are grouped i.e. every cluster has least distinction between its focuses and most extreme contrast from other data points of the cluster. The proposed algorithm [29] utilizes k-mean algorithm and firefly to group image pixels into k cluster for segmentation. Since k-mean clustering algorithm gets caught in the local optima, it is improved by using firefly algorithm. Algorithms based on Swarm intelligence provide the basis of the firefly algorithm which has a few applications and used to tackle optimization issues. Firefly algorithm has been connected in many research and advancement areas. Firefly algorithm and its hybridized adaptation have been utilized to take care of different issues effectively. To apply firefly algorithm to wide range of issues the firefly algorithm must be changed or incorporated with different algorithms. Directly meta-heuristic nature of algorithm assumes a critical part and current improvement algorithm incorporates this nature and is extremely effective in taking care of NP-hard problems.

The process of image segmentation by using proposed algorithm is as follows:

1. Suppose I be the chance to be the input image having the dimensions M*N by the client.
2. Perform different operations to improve the image.
3. Image vectorization is done into (M*N,1) dimensional lattice.
4. This matrix is passed to the K-mean firefly algorithm for clustering.
5. The pixels having similar intensities are sorted into different clusters.
6. After the formation of clusters, the pixels will be replaced with their corresponding relegated centroids of clusters.
7. The matrix that is obtained will be then changed in accordance with gray scaling images.
8. Perform different accuracy estimations on the obtained segmented images.

2.13 FFA in reducing interference in Cognitive Radio Networks

Cognitive radio (CR) is the novel innovation in the field of wireless communication, which has a capacity to utilize the empty range of essential user insightfully. NCOFDM, a multi-carrier system is considered as the suitable possibility for physical layer of CR, yet because of its high sidelobes it experiences more radiation of out of band, that can influence its sharing capacity of the spectrum. In this paper, the researcher proposed two searching algorithms: Genetic Algorithm and Firefly algorithm for assessing the amplitudes of the Cancellation sub-carriers, which are embedded on either side of the utilized signal of NCOFDM to cross out the sidelobes. Results come about demonstrate that better reduction in sidelobe is accomplished by the proposed algorithm when contrasted with the current techniques.

In [30], researchers proposed two distinctive techniques for searching; GA and FFA to figure out the amplitudes of the CCs, which are utilized to counterbalance the sidelobes of the first signal of NCOFDM and think about the aftereffects of these proposed searching techniques with the current techniques. From the results, it is demonstrated that the proposed strategies, show signs of improvement cancellation of sidelobes as contrasted with the current techniques, and obtains the better performance of FFA.

2.14 FFA for scheduling workflow in cloud computing environment

Cloud computing is the new era of systems that utilizes remote servers facilitated on the Internet for different uses, for example, information storage, management of data, and so forth. There are number of resources available and hence users can use them in any means they need to. Now-a-days, researchers try to discover more up to date ways for Scheduling of Workflow [31] which could function admirably in the cloud environment. Workflow scheduling is the most essential job in cloud computing domain and users need to pay for the resources that were utilized based on pay-per-use scheme. Consequently Workflow scheduling assumes a fundamental part in getting most extreme advantage from the resources that are given. Another important component to be considered about cloud computing is Load balance. This guarantees that each machine does the extremely same amount of work at any given time. To ensure this, there is a need to suggest on utilizing fill controlling. Here in this record, we suggest heuristic technique; Firefly criteria for successful fill controlling. This measure depends on the travelling behavior of the fireflies which go searching for the nearest possible maximum options. Researchers implement Firefly algorithm to plan the tasks and in this way uniformly disseminate the load and thus minimize the full completion time (makespan).

Workflow scheduling assumes an essential part here, so that the workload will be managed and benefit can be maximized. The algorithm proposed here is FFA which works based upon swarm behavior of the fireflies. This FFA was first outlined by a man named Yang who was pulled in by a swarm of fireflies. It is known to be the best among the various algorithms based on swarm intelligence because of its fast properties of convergence, adaptability and error tolerance capabilities. This FFA is a meta-heuristic algorithm and hence the accuracy depends on the parameter positions in the scenario.

Workflow scheduling is a fundamental origination in cloud computing environment. There is huge number of algorithms molded particularly for scheduling the workflows.

1. Cat swarm optimization (CSO), a heuristic optimization technique: CSO is developed in view of the motivation towards the swarm behavior of cats. Cats which generally have swarm behavior are said to have two methods of behavior; seeking mode and tracking mode.

2. Particle Swarm Optimization: A compound in PSO is much similar to a fowl or even seafood going through a search region. The movement of each compound is related with a speed which is said to have both scale as well as route. Position of each compound at once is affected by its best position and furthermore inside the best compound in an issue range. There is a specific property called as fitness value for every particle which is utilized to quantify the performance capability of every particle in the space. By using the same, the globally best solution is detected by repeating to discover local best solution corresponding to the fitness function. Each compound place at once is influenced by its best place and inside the best compound in a problem space. The capacity of a particle is computed by the value of fitness.

The present development in the domain of cloud computing is the event of distributed systems generally. A distributed system is a collection of components situated in diversely located systems arranged together all of which have the need of communicating with each other. As in distributed system, the workflow scheduling gets increasingly more complex as the total of jobs should have been scheduled continues increasing. As workflow scheduling is more important to process the jobs, there should be some sort of algorithm for workflow scheduling. Additionally the workload must be equally distributed, so that the execution time of the procedures will be reduced as much as possible.

Here in we denote a heuristic algorithm called as the firefly algorithm for workflow scheduling. This algorithm envelops every possible method to plan the work flow in the most effective way possible. This FFA is designed in view of the motivation on the swarm behavior of fireflies. Fireflies are generally referred to exist as groups and they are said to have a swarm sort of behavior. The light blinking nature of fireflies is their attribute of attractiveness for attracting the mates and to protect themselves from the predators. The swarm of fireflies generally moves towards the brightest one. The various fireflies with lower light intensities move towards the flies with higher intensity of light. So as the distance among the fireflies continues increasing, the intensity of light will also increase.

FFA is implemented on the need of scheduling the workflow where the execution time should be decreased in the most possible way. In FFA, the fireflies are considered as Virtual machines. All fireflies are compared to virtual machines. The swarm of fireflies gets attracted moves toward the brightest fly. The fitness value of every firefly is computed and furthermore the brightness will be determined. The attractiveness of every firefly relies upon their brightness. The separation between every firefly is resolved and updated after every cycle/iteration. The fireflies are ranked in view of the fitness function and the general global best is resolved. With regards to job scheduling on the virtual machines, the best virtual machine at the execution time is chosen and jobs are assigned to that chosen virtual machine. During searching of the best one, the one with the best specifications will be chosen and the jobs in queue are assigned to those virtual machine.

2.15 Challenges

- Optimization of communication resources [32]: Resource can be either global or local which are to be optimally utilized to handle the traffic. Globally resources are those of network resources like frequencies spectrum and locally resources are processing power, life-time of battery, and memory capacity of the device. There is a need to design the future communication mechanisms which optimizes local resources.
- Discovery of optimal routes and route management [33]: There are various factors that affect the discovery and maintenance of multi-hop routes which includes mobility, lack of device cooperation, inadequate resources, and spectrum switching.
- Device collaboration: This can be a major challenge which affects the optimal routes. Due to device constraints, participation in communication gets reduced. To handle this, effective mechanisms should be implemented.
- Security [34]: Provision of security in IoT is still a major challenge. Authentication is still an issue in autonomous D2D network.
- Platform Compatibility [35]: As various manufacturers create diverse devices with various equipment and architectural platforms, all these ought to be made perfect with the single IoT basic platform. As the entire IoT space contains heterogeneous gathering of devices with various equipment, and information storage and processing segments, dependency and similarity issues emerges. This may be an

obstacle for those devices to share and reuse the information. The upgraded guidelines need to mind the present Internet standards.

- Accessibility [36]: In most of the circumstances, information accessing is finished using simplex communication mode as opposed to duplex mode. Information is received from one device in one way; forward it to the next utilizing diverse ways where handling of that information is done, lastly the sending device get response in a different way. This emerges the requirement for information accumulation and information elucidation.
- Power usage: In conventional IP-empowered devices, "business" includes real power utilization by different parts independently inside a solitary device; processor, memory, OS, transmitter, collector, and MAC/PHY. If there should arise an occurrence of IoT convention empowered devices, high power is consumed by very number of the previously mentioned segments.
- Maintainability: It is exceedingly hard to direct and deal with the addressing of sufficient of devices through conventional methods for communication utilizing IPv6. Thus self-learning idea will make the undertaking simple which incorporates self-tending to and self-arrangement. Aside from this, giving security and upgradation to huge pool of savvy devices from assortment of producers is hazardous.
- Connectivity [37]: The quantity of devices that partake in IoT will be developing, and might be more than the quantity of people on the Earth. Conventional Internet contains about 5 billion devices whereas IoT has least of 700 billion to trillion devices. Consequently the current conventions, devices and procedures need to experience facilitate improvements to meet the necessities.
- Scalability [38]: The extent of the smart devices to be blended is higher than the desired with a quick development. IoT includes machine-to-machine interaction which is unique in relation to traditional Internet.
- Robustness [39]: IoT does not require robust protocols, for example, TCP/IP because of its overhead prerequisite. Additionally, most IoT end devices are of minimal processing, adequate memory and communication capabilities. Such simple devices ought not be troubled with complete network protocol stack which prompts high cost and high intricacy.

CHAPTER 3

SCOPE OF THE STUDY

There are numerous devices which continue to grow from few thousands to billions, from soil moisture sensors to advance cloud-connected devices, from short range communication to remote control communication. Management is the major concept to be dealt with because of its role in IoT like data management, device management, quality of service (QoS) [40] management and others. The scope of this study is based on the flow of inheritance starting from traditional system to current IoT system which can be further extended to meet the scalability factor in IoT. Controlling these numerous unmanageable devices of IoT is to be accurately done so as to support the scalability. It is a clear cut view of various application domains of IoT that the traditional architectures of networking are being modified and transformed into emerging advanced architectures to meet various IoT requirements. Following are the 'things' that plays a major role in a scalable IoT.

- Intelligent Gateways, tools and applications, and middleware
- IoT Protocols
- Big data management
- Cloud computing

The scope of IoT is not limited to any specific domain. It has very wide range of scope such as smart cities, homes, vehicles, health, energy, education, and so on. IoT is the area of association with variety of wide areas like WSNs.

There is a need to move a step ahead for the research in the area of requirements such as system-wide scalability, context-awareness, reliability, security and privacy, knowledge and intelligence integration, dynamic resource discovery and composition, connectivity preserving and interoperability.

Of the above mentioned requirements, dynamic resource discovery and connectivity preserving are significant in IoT which uses two parameters: distance and velocity. Implementing an efficient technique for this scenario is focused.

CHAPTER 4
PRESENT WORK

4.1 Problem Formulation

4.1.1 Problem Motivation

IoT offers a practical resource scheduling by guaranteeing their maximal use and service to the customers. IoT gets features from various computing environments, for example, mobile computing, WSNs, and pervasive computing. From this, obviously IoT depends on upon various essential capabilities and developments like machine learning, signal handling/processing, real time computing, data analysis in big data analytics, data curation, information sharing, information storage, information exchanging, and representation. Security and assurance concerns are everywhere, which are authoritative scopes of data correspondence where threats and attacks are to be perceived and resolved.

A vital undertaking to be done in IoT communication is to change over the distinguished or accumulated data into keen information. Alongside this, the gadgets in IoT have limit of perceiving and detailing the region, history over the remote associations. For an aggregate operation of an IoT framework, rest of the functionalities are data gathering, association and communication, computational limit, and data storage and handling, where most by far of the devices are of vitality obliged and compelled memory which are real trying issues that ought to be settled.

Wireless sensor network (WSN) consists of a huge number of tiny sensor nodes. Each sensor device has the capability to compute, communicate, sense and operate. Primarily, WSNs have been used widely in a variety of applications to perform tasks like environment monitoring and real time decision making. These tasks are purely dependent on the quality of data and information of the WSN.

Sensor nodes in the WSN are basically responsible for carrying up all the quality data and information. These sensor devices not only sense the situation but also cooperate with each other. So, sensing and communicating the data is the basic principle on which wireless sensor network works.

Besides sensor nodes, WSNs also contains sinks and sources. These all devices are connected with each other via wireless medium. This medium can be radio signals.

These sensor devices are not expensive and are lower in energy. Moreover, these nodes have different memory capabilities. The sensor devices have functionality of sensing environmental and physical factors like pressure, temperature and sound in the field of network where they are actually deployed.

Working of HMPC Algorithm is as follows:

Step 1: Identification of neighbor nodes of any desired node n.

Step 2: Calculation of weight of the node n which is then broadcasted to its neighbor nodes.

Step 3: Node n is elected as a cluster head (CH) node, if its weight is less than any of its neighbors'. Otherwise it becomes a border node which runs the connectivity repair algorithm.

Step 4: If any of the neighbor nodes is neither a CH nor a cluster member, then based on the MPC and MPC_g, it is decided whether to include the neighbor node in the cluster or not. In this way, a cluster head (CH) node which guarantees the best connectivity among its neighbors in the course of cluster scope.

4.1.2 Problem Statement

"Identification and enhancement of device connectivity issues in scalable IoT"

4.2 Objectives of the Study

4.2.1 Maximizing the Throughput

Throughput is a measure of how much actual information can be sent per unit of time over a network, channel or interface. While throughput can be a hypothetical term like bandwidth, for instance, to quantify the measure of information really sent over a network. Throughput is restricted by bandwidth, or by evaluated speed.

A wireless network is a multi-hop wireless network comprising of huge number of nodes, some of which are called gateway nodes and associated with a wired network. Utilizing various channels rather than a single direct channel in multi hop wireless network appeared to have the capacity to enhance the system throughput significantly. The throughput of a network [41] can be significantly expanded by using various channels rather than a solitary channel.

4.2.2 Minimizing the Packet Loss Ratio

Packet loss happens when at least one packet of information traversing a network neglect to achieve its destination. Packet loss [42] is normally brought about by network congestion or blockage. It is measured as a rate of packets lost as for packets sent. At the point when content reaches the base for a managed period at a given router or network segment at a rate more noteworthy than it is conceivable to send through, then there is no other alternative than to drop packets. However the cause occurs, the effect must be reduced over the network.

Cause: It can be occurred by various different variables that can degenerate or lose packets in transit, making it impossible to remove or multi-path fading [43] (in radio transmission), flawed network administration equipment, or broken network drivers. Packets are purposefully dropped by ordinary routing routines, (for example, Dynamic Source Routing).

Effect: Packet loss can reduce the throughput for a given sender, whether unintentionally because of network failure, breakdown, malfunction [44], or deliberately as a way to adjust accessible data transmission between various senders when a given router or network link reaches to its maximum limit.

4.2.3 Increasing the Network Lifetime

Remote sensor networks are for the most part sent in a remote workplace, since sensor devices are small in size, cost-effective, low-powered, and have restricted battery power supply. Due to constrained power source, energy utilization has been considered as the most basic factor while designing the protocols for sensor networks. The network lifetime mostly relies on the battery lifetime of the hub. The principle concern is to expand the lifetime as for energy constraints. One method for doing this is by switching down the excess nodes to rest mode to save energy while dynamic nodes can give fundamental coverage, which enhances adaptation to non-critical failure.

4.3 Research Methodology

Cluster formation is the major task in any field. And also identifying the connectivity failures in the network and restoring the same has been a significant issue in the field of wireless sensor networks. For these to happen, a better solution is to be considered. Till now there has been many researches going on for enhancing the performance of the wireless networks especially WSNs. Different heuristic approaches have been proposed for resolving such issues. Such techniques work effectively under a condition that the sensor nodes should have a specific mobility pattern. But this is not the case at all the places, for example, remote monitoring at unreachable locations, military battle fields, nuclear pollution environment monitoring and so on.

For the lost connection to get restored, centralized or distributed approaches should be followed. For this to happen, these approaches should further consider the mobility of the nodes. Hence it is practically difficult to detect the location of the node or device which has a connectivity failure.

To resolve the above issues, a better clustering algorithm should be used. As mentioned in [16], latest algorithm called Heterogeneous Mobility Pattern Clustering (HMPC) algorithm aims at repairing and reestablishing the connectivity of detached sub-clusters which are formed after any disturbance or damage caused to the connectivity in a large cluster of mobile sensor nodes thereby improving the performance in terms of packet loss ratio and throughput. It considered DBSCAN clustering algorithm and Velocity-Based clustering algorithm to track and to determine the connectivity depletion among nodes. DBSCAN makes the HMPC algorithm to determine the clusters having weak connectivity followed by repairing them which involves three core functionalities namely making of connectivity period, cluster determination, repairing of connectivity.

In our Grouping Approach using Strength of Device Synergy (GASDS), Firefly Algorithm (FFA), which is swarm intelligence based algorithm, is used because of its ability to form the clusters in an effective manner. Swarm Intelligence [45] methodology deals with different agents that communicate with each other in an artificially environment involving in communicating with the environment too. These agents can be either homogeneous agents or heterogeneous. The homogeneous agents share different properties of intelligence and behavior. The heterogeneous ones vary in these properties. The heterogeneous systems don't satisfy the property of control of decentralization. On account of homogeneous agents, the property of decentralization control additionally holds well with self-association.

- Signaling system in fireflies: Fireflies has the specialized ability of signaling the other fireflies by emitting light. This unique light blinking pattern is an optical signaling technique helps in finding out their potential mates.
- Cold light emission: The light of the fireflies is the most efficient light on the Earth because almost 100 percent of the energy is emitted as light with almost zero heat.
- Unique patterns: Fireflies can be able to recognize their corresponding species and also the sex of the other fireflies by using species-specified patterns i.e., variation in pulse-rate, duration as well as the number of pulse generations.

Hence, by considering the above capabilities of the Firefly algorithm, and the simulation results in better output by comparing with the existing HMPC Approach in terms of throughput and packet loss. Following is the proposed GASDS Approach which adapts Firefly Algorithm, considers a new metric called Strength of Device Synergy (SDS). The entire algorithm is categorized into four phases described as follows.

Phase 1: Calculation of SDS and defining objective functions

Phase 2: Cluster formation using FFA

Phase 3: Identifying connectivity issue

Phase 4: Connectivity restoration

During the first phase, SDS and objective function calculation phase, the objective of setting up the clusters can be reached through calculating the SDS metric. Then the nodes of the clusters are said to have a strong or weak connectivity based on SDS. The strength (SDS) of the nodes is associated with the distance among the nodes in the network. In order to get strong clusters, there is a need to introduce a parameter which reflects the strength of the node/device compared with its neighboring nodes/devices. The SDS seems to be a good candidate as a parameter because it is explicitly associated with the basic properties of a sensor device; movement, position, speed, and frequency at which the sensor node is operated. The common formula used to calculate the speed is shown as below.

$$Speed\ of\ a\ node\ (R) = \frac{Distance\ between\ any\ two\ nodes\ (d_{ij})}{Time\ elapsed\ (t)} \quad (6)$$

$$\text{Where } d_{ij} = \sqrt[2]{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2} \quad (7)$$

Following is the condition for minimum connectivity time period at which No SDS (NSDS) is identified:

$$t = \frac{T}{R_x + R_y} \quad (8)$$

Where $T = T_{rc} - d_{ij}$

Another factor that contributes to the system is minimum detectable signal strength (MinDSS) which determines the possible lowest strength signals within the communication range of the node.

$$SDS_{xy} = \sum_{i=1}^{n} (R_{avg} * MinDSS * F_i) \quad (9)$$

Following are the major cases to be considered for determining the SDS:

Case i: If two given nodes are moving towards each other, then the strength (SDS) of the nodes is said to be high. Because of the mutual effect in the movement of nodes result in increasing the strength between the nearest neighbor nodes. The combined effect will produce greater strength.

Case ii: If any given node is moving far away from other node, then the combined mutual effect results in decreasing the strength of the node.

Case iii: A node is said to be isolated if and only if the node moves randomly out of the communication range of any node in the network. Then that particular node is said to have No SDS (NSDS) with respect to any other node.

Objective function f(x) is calculated by combining the three different sub objectives as follows:

1) First sub-objective: $f_1 = \sum_{i=1}^{k} S_{node_i}$; where S is the strength of node i.
2) Second sub-objective: $f_2 = \frac{1}{\sum_{i=1}^{k} d_{xy}}$; where d_{xy} is the distance between the nodes x and y.
3) Third sub-objective with respect to node degree: $f_3 = \frac{1}{\sum_{i=1}^{k} N_i}$; where N_i is the node degree.

The combined result of the above three sub-objectives is our main objective function f(x).

$$f(x) = (\mu_1 * f_1) + (\mu_2 * f_2) + (\mu_3 * f_3) \qquad (10)$$

Where μ_1, μ_2, and μ_3 are the assigned weights to each sub-objective satisfying the condition $\mu_1 + \mu_2 + \mu_3 = 1$.

During the second phase, Cluster formation phase, it follows the below described steps.

Step 1: Generating the initial population of all the nodes.

Step 2: Searching for the strong connectivity neighbor nodes that has better SDS which is proportional to the objective function f(x). Below is the distance calculation method helpful in detecting the nearest neighbor node. As the distance between the nodes increases, SDS is said to be lower, and when the distance is less, SDS will be greater.

Step 3: Now the node that has higher SDS will be ready to become a part of the cluster.

$\text{Exp}(- \gamma * d^2)$ is the factor of attractiveness

Step 4: Above steps are repeated for the further processing to obtain better nodes and the updation of SDS metric continues.

Step 5: Nodes will be ranked based on the SDS for finding the best node.

Step 6: Clusters of the best ranked nodes will be created with best CH.

In this way, cluster of nodes will be generated with three classes of nodes namely; Member nodes (MNs), Bridge nodes (BNs) and Unreachable nodes (UNs). Member nodes are the nodes that become the member of any cluster. Bridge nodes are responsible for maintaining the connectivity among different neighboring clusters with their corresponding Bridge nodes. Unreachable nodes, as the name itself reveal that they can become neither a member node nor a bridge node. BNs are the ultimate responsible nodes for connectivity restoration if any failure occurs.

Finally the third and fourth phases, identification of connectivity issues and connectivity restoration phases respectively, involves in detecting and resolving the lost connectivity issues in the network. For this to happen, Bridge nodes (BNs) come into picture. Among all the nodes in a given cluster, member nodes have greater SDS value than the bridge nodes which have low SDS value due to the distance from the cluster head, which has the maximum SDS value with the associated or connected member nodes of the cluster, because SDS value is associated with the distance. The relationship between SDS and distance is inverse proportionality. As the distance increases, SDS of a given node with other node decreases and vice versa.

The bridge nodes help in identifying the best node to become its cluster member. Comparison of SDS values results in yielding the best compatible and suitable cluster member.

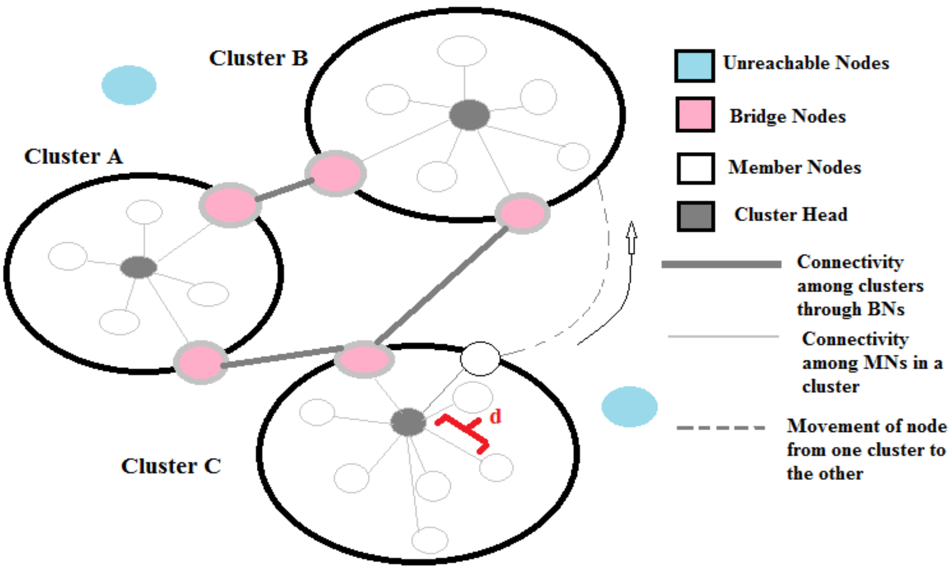

Figure 6: Cluster formation showing variety of nodes

4.3.1 Flow Chart

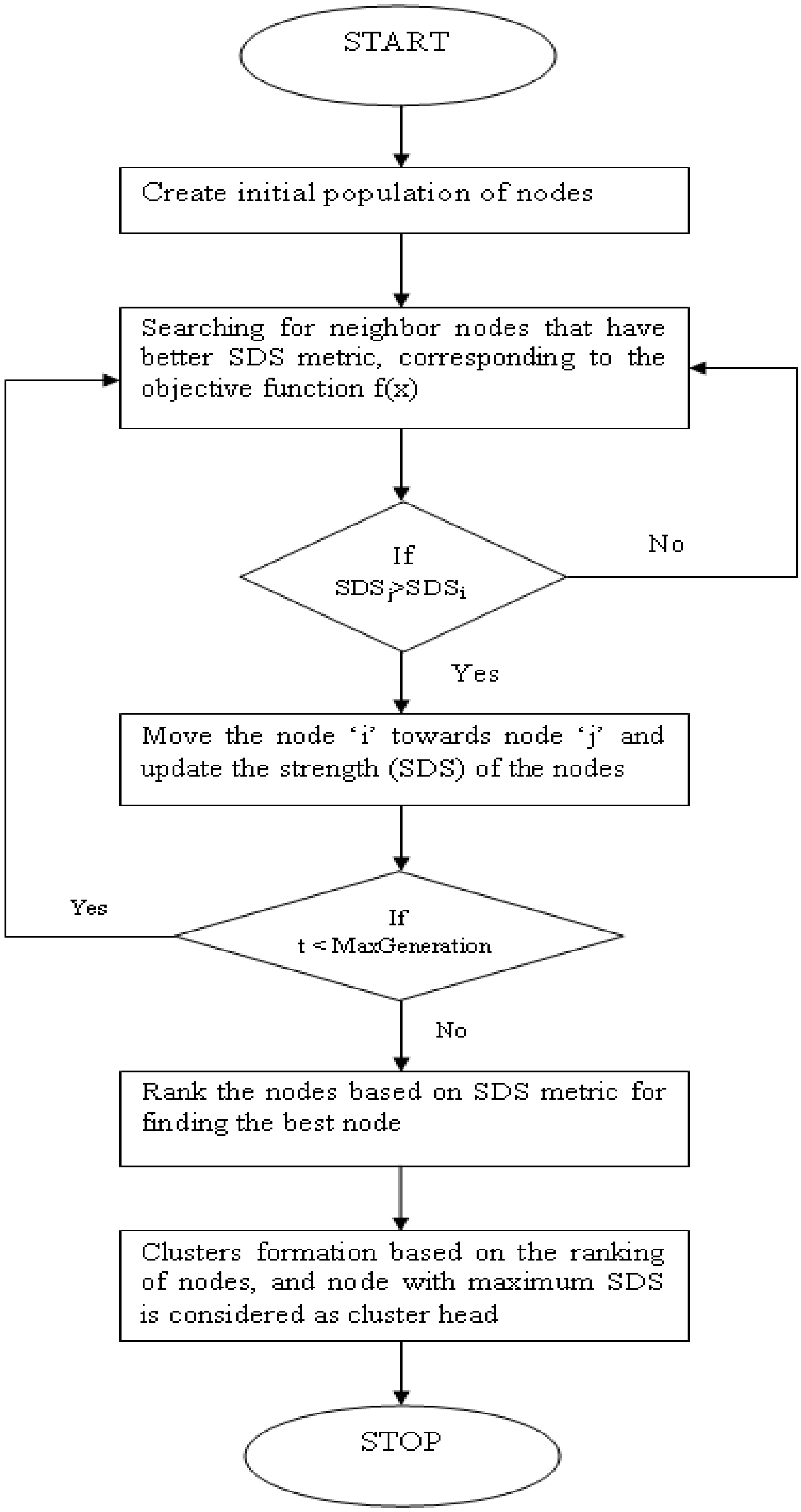

Figure 7: Flow chart representing proposed GASDS Approach

4.3.2 Proposed Algorithm:

Firefly Algorithm (FFA)

Begin

1) Objective function: ***f(x)***, where $x = (x_1, x_2, x_3, \ldots, x_k)$

2) Generating the initial population of nodes x_i, where $i = (1,2,3,\ldots,n)$

3) Calculate the intensity of light so that it should be associated with the above objective function *f(x)*

$$I \alpha f(x); \text{ simply } I = f(x)$$

4) Defining the coefficient of light absorptions γ;

5) While (t < MaxGeneration)

for i = 1 : n (all the existing n nodes)

for j = 1: n (n nodes)

if ($I_j > I_i$)

i) Then attractiveness will vary with distance between the nodes 'd' through ***exp(- γ*d²);*** $exp(-\gamma * d^2)$;

ii) Move the node i that has low intensity towards node j;

iii) Evaluation of new solutions by updating the I value;

end if

end for j

end for i

6) Rank the nodes based on I for finding the best node

7) Repeat step 5 for the condition of t < MaxGeneration to be satisfied

8) Process the results to generate strong and efficient clusters

End

4.3.3 Tool used

The simulation situation has been ready exploitation the MATLAB machine. The MATLAB simulation is known as Matrix Laboratory which was written in C/C++ for the windows/MAC/LINUX platforms. The MATLAB machine has been put in on the Windows version seven for the aim of experimentation during this thesis. The experimentation has been performed on the idea of assorted simulation aspects. The MATLAB machine uses the mat-lab language for the simulation programming. The simulation programming is that the technique to create the nodes, their property and behavior primarily based triggers.

For implementation of the above-mentioned scenario, “MATLAB” is used, which is a user-friendly simulator. It has the ability to auto-generate the code using MATLAB coder. It allows us to test the algorithms without the need of recompilation which is a great facilitating environment. It is flexible to call external libraries etc., and for debugging and compiling aspects. It also integrates programming, visualization as well as computation to be expressed in simple mathematical notations. Therefore, is easy to use for larger subsets.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Experimental Results

The computed experimental results are explained as below. The tool used for simulation purpose is MATLAB. The comparison of results is done with the existing HMPC Approach and the proposed GASDS Approach. Following are the experimental results demonstrating the upper hand in generating the better results in terms of throughput and packet loss.

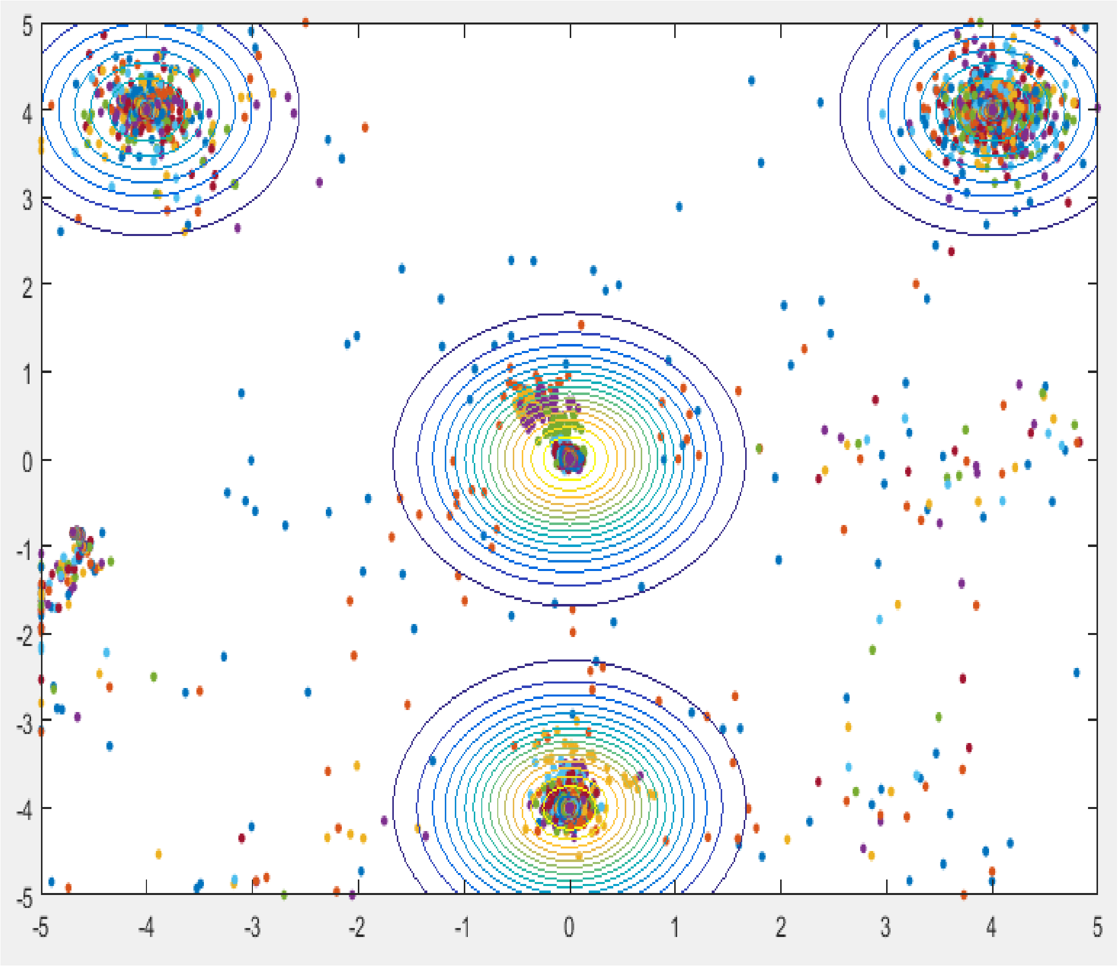

Figure 8: Topology of the proposed system

The above figure shows the distribution of various sensor nodes that are ready to transmit the data to each other after identifying the corresponding nearest neighbor nodes in the corresponding clusters.

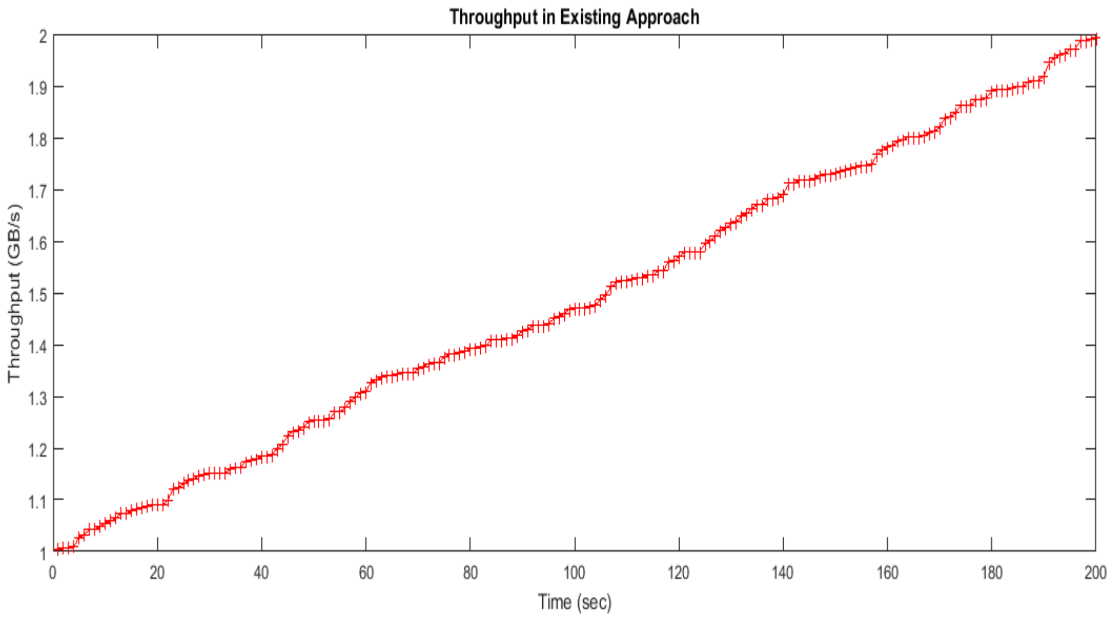

Figure 9: Throughput in Existing System

The above figure denotes the throughput which is obtained in the existing HMPC Approach, which uses DBSCAN Algorithm.

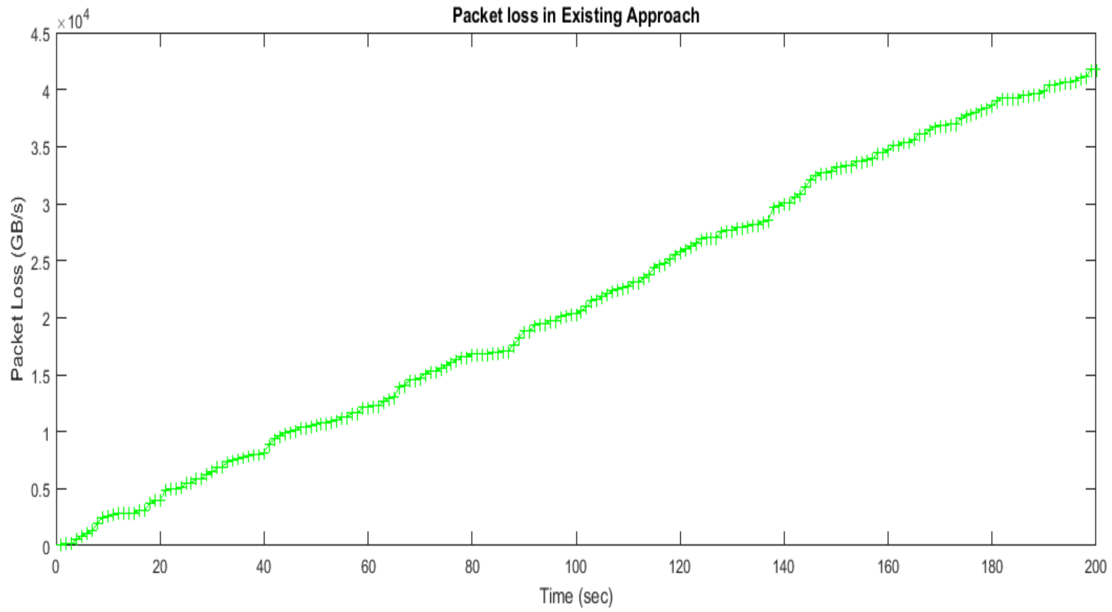

Figure 10: Packet Loss in Existing System

The above figure shows the packet loss in existing system, HMPC Approach, which uses DBSCAN Algorithm.

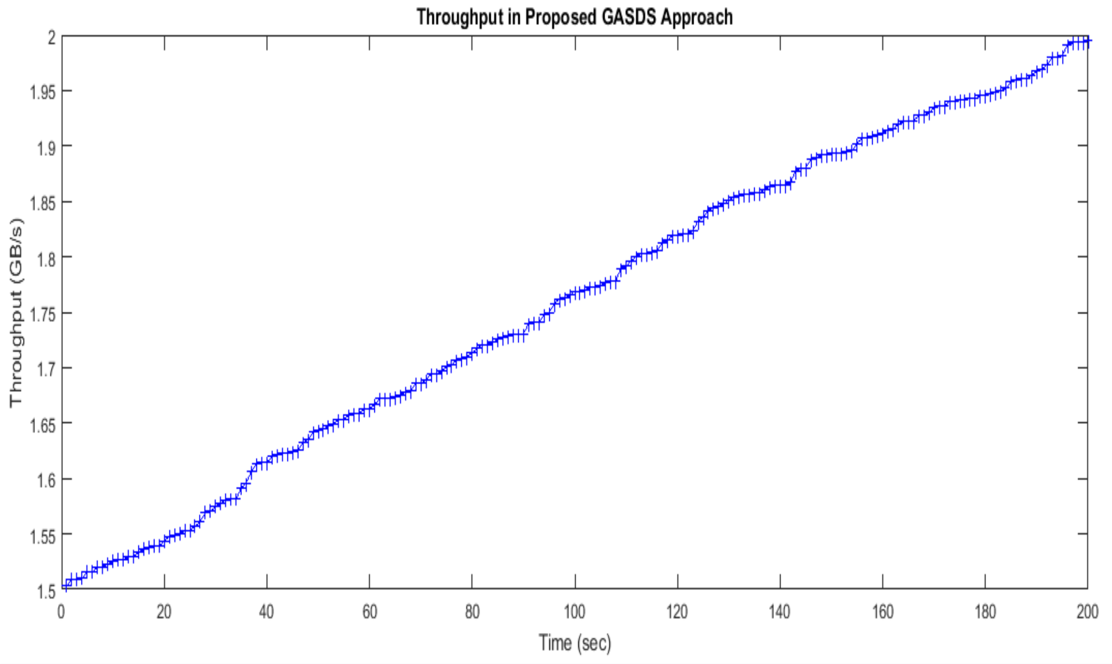

Figure 11: Throughput in Proposed System

The above figure demonstrates the throughput in the proposed approach, GASDS Approach, which is better than the throughput of HMPC Approach.

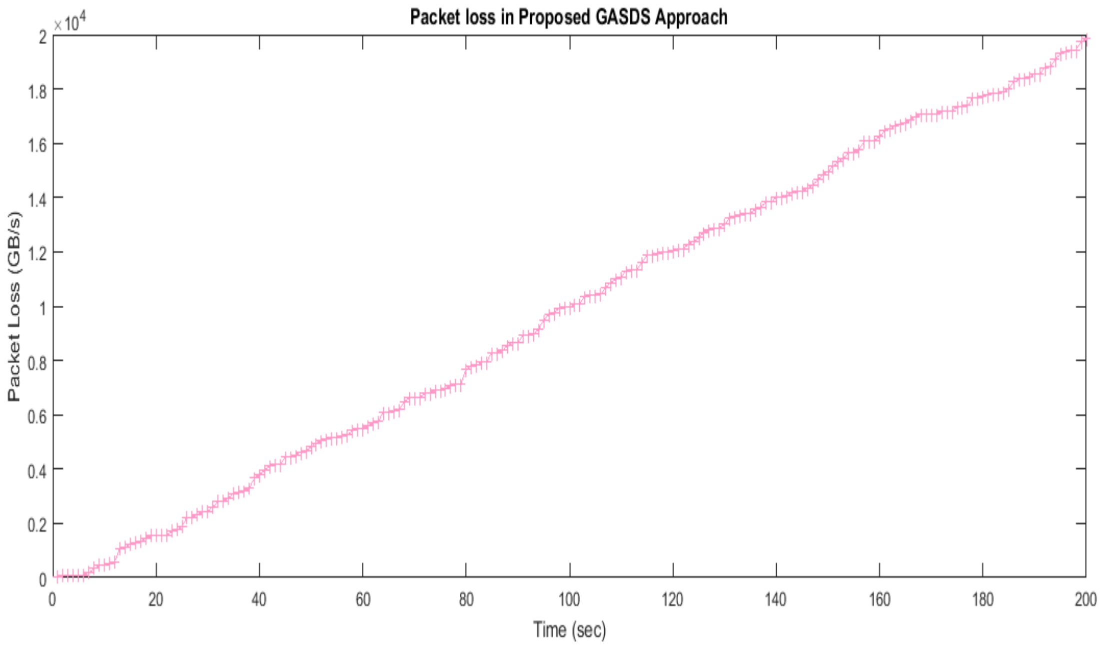

Figure 12: Packet Loss in Proposed System

The above figure shows the packet loss in proposed system, GASDS Approach, which is better than the existing HMPC Approach.

5.2 Comparison of Proposed technique with Existing technique

The proposed model has been well tested under various situations in the sensor network simulation tool, MATLAB. The proposed model has been well tested for the performance parameters of throughput as well as packet loss. The nodes in the proposed model simulation have performed well in terms of all of the above parameters. The throughput, packet loss rate has been recorded higher and lesser respectively in comparison with the existing HMPC Approach.

Table 1: Comparing values of Throughput and Packet Loss of HMPC & GASDS Approaches

Throughput (GASDS)	Throughput (HMPC)	Packet Loss (GASDS)	Packet Loss (HMPC)
1.5035	1.0028	0.0041	0.0061
1.5087	1.0028	0.0055	0.0149
1.5087	1.0052	0.0056	0.0259
1.5098	1.0099	0.0065	0.073
1.5154	1.0132	0.0071	0.0868
1.5159	1.018	0.0102	0.1075
1.5196	1.0276	0.0192	0.1311
1.5203	1.0362	0.0334	0.1446
1.5228	1.0364	0.0461	0.2487
1.5252	1.0428	0.0485	0.2631
1.5264	1.0515	0.0531	0.2689
1.5268	1.0524	0.0596	0.2793
1.5291	1.0591	0.106	0.2922
1.5298	1.0643	0.1103	0.311
1.5335	1.071	0.1241	0.3441
1.5359	1.0877	0.1304	0.3592
1.5374	1.091	0.1313	0.3594
1.5386	1.0939	0.1473	0.3697
1.5386	1.1003	0.153	0.3915
1.5437	1.1113	0.154	0.4083
1.5478	1.1165	0.1547	0.4255
1.5485	1.1184	0.1564	0.446
1.5504	1.1206	0.1696	0.4561
1.5528	1.1279	0.1766	0.49

The above table indicates comparative analysis of two significant parameters namely, throughput and packet loss in both existing and proposed GASDS (Grouping Approach using Strength of Device Synergy). Results show better performance with proposed GASDS technique.

Following are the corresponding simple bar char representations of throughput and packet loss of the two approaches, HMPC (existing approach) and GASDS (proposed approach), demonstrating the comparison of both existing as well as proposed approaches in terms of throughput and packet loss, where it reveals that the proposed GASDS Approach outperforms with maximum throughput and minimum packet loss.

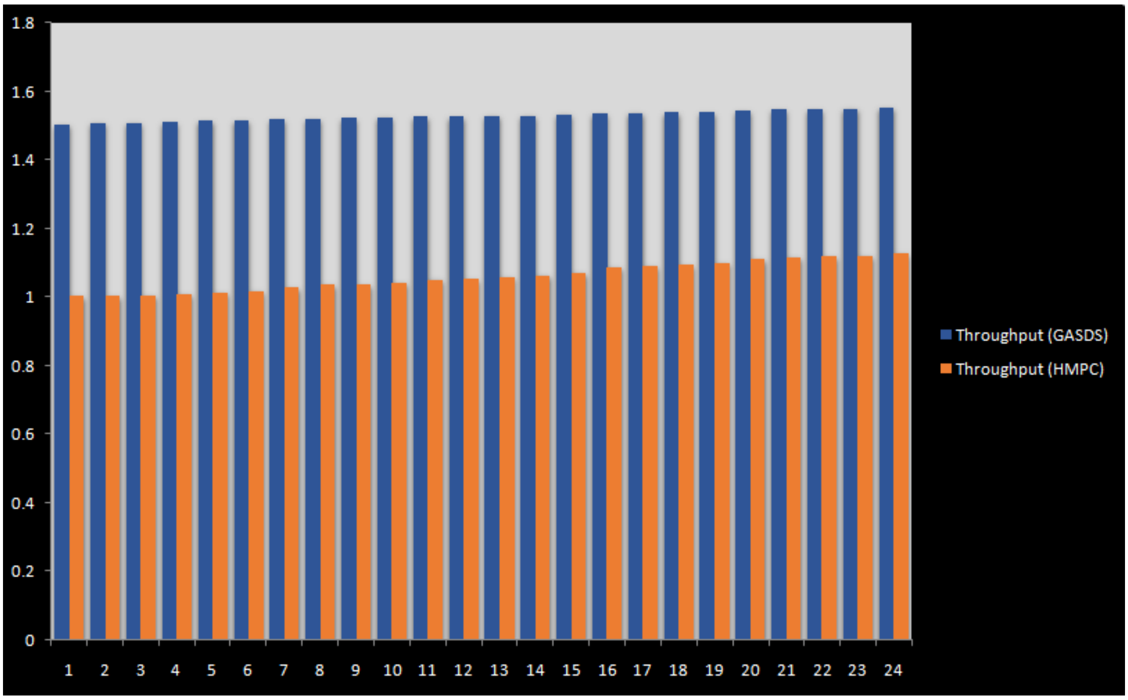

Figure 13: Comparing Throughputs of HMPC & GASDS

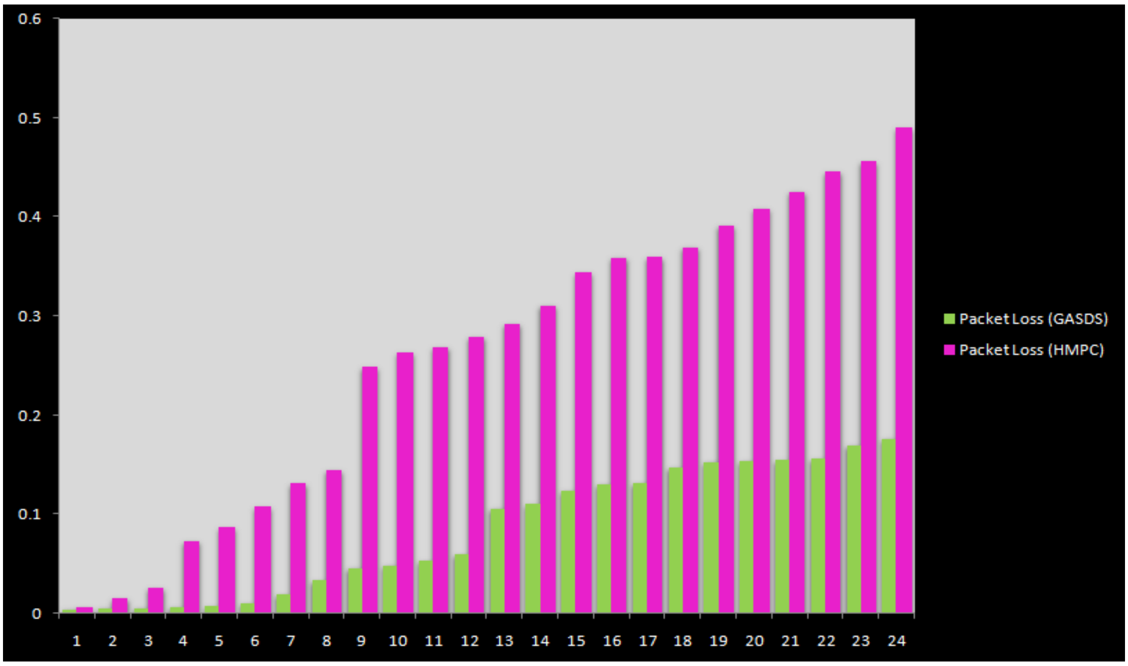

Figure 14: Comparing Packet Loss of HMPC & GASDS

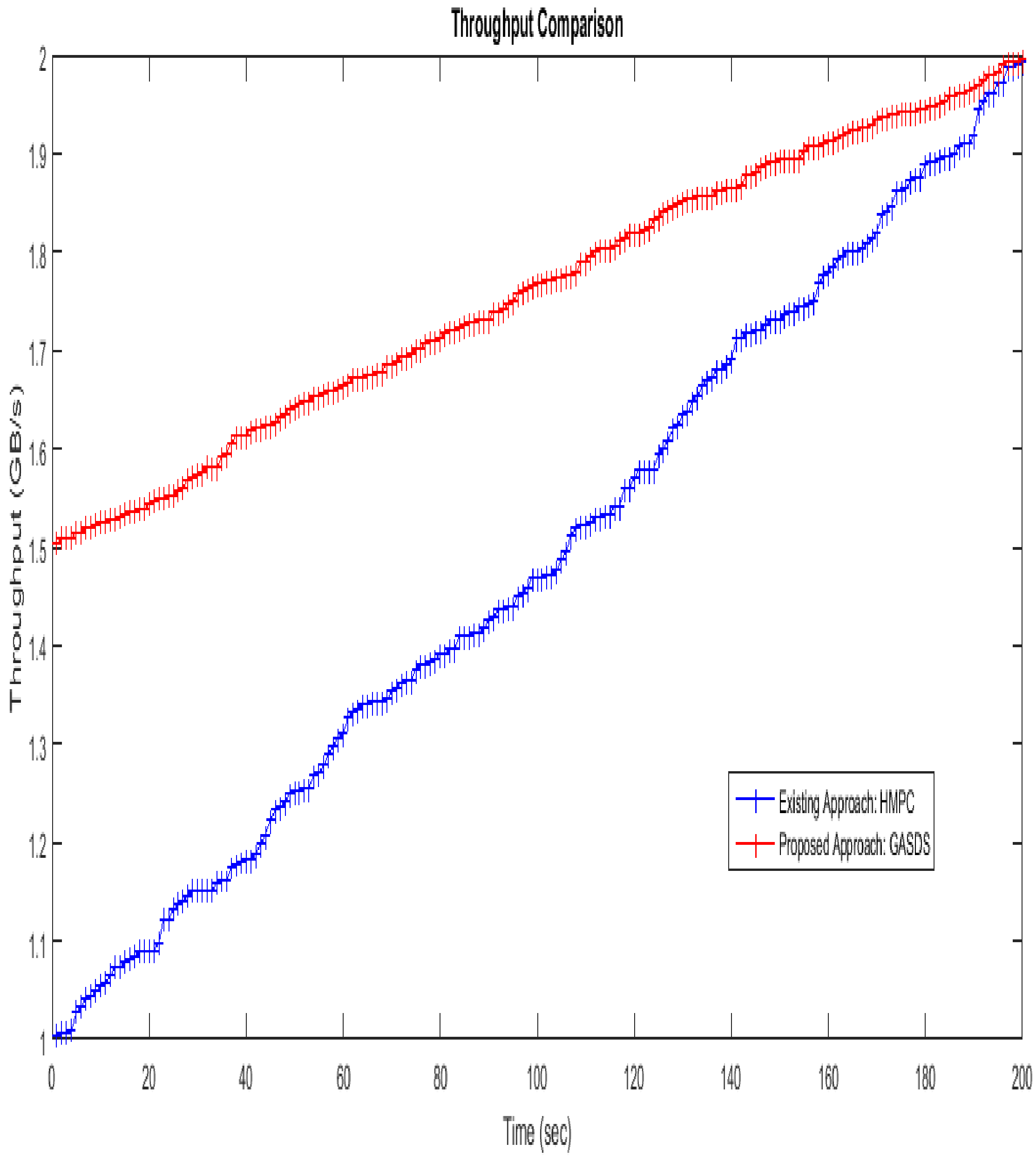

Figure 15: Comparison of Throughput

The above figure demonstrates the graphical representation of throughput with respect to time. The red colored marks indicate the throughput of the proposed technique, whereas blue color indicates the throughput of the existing technique. Throughput in proposed system is better than the existing technique.

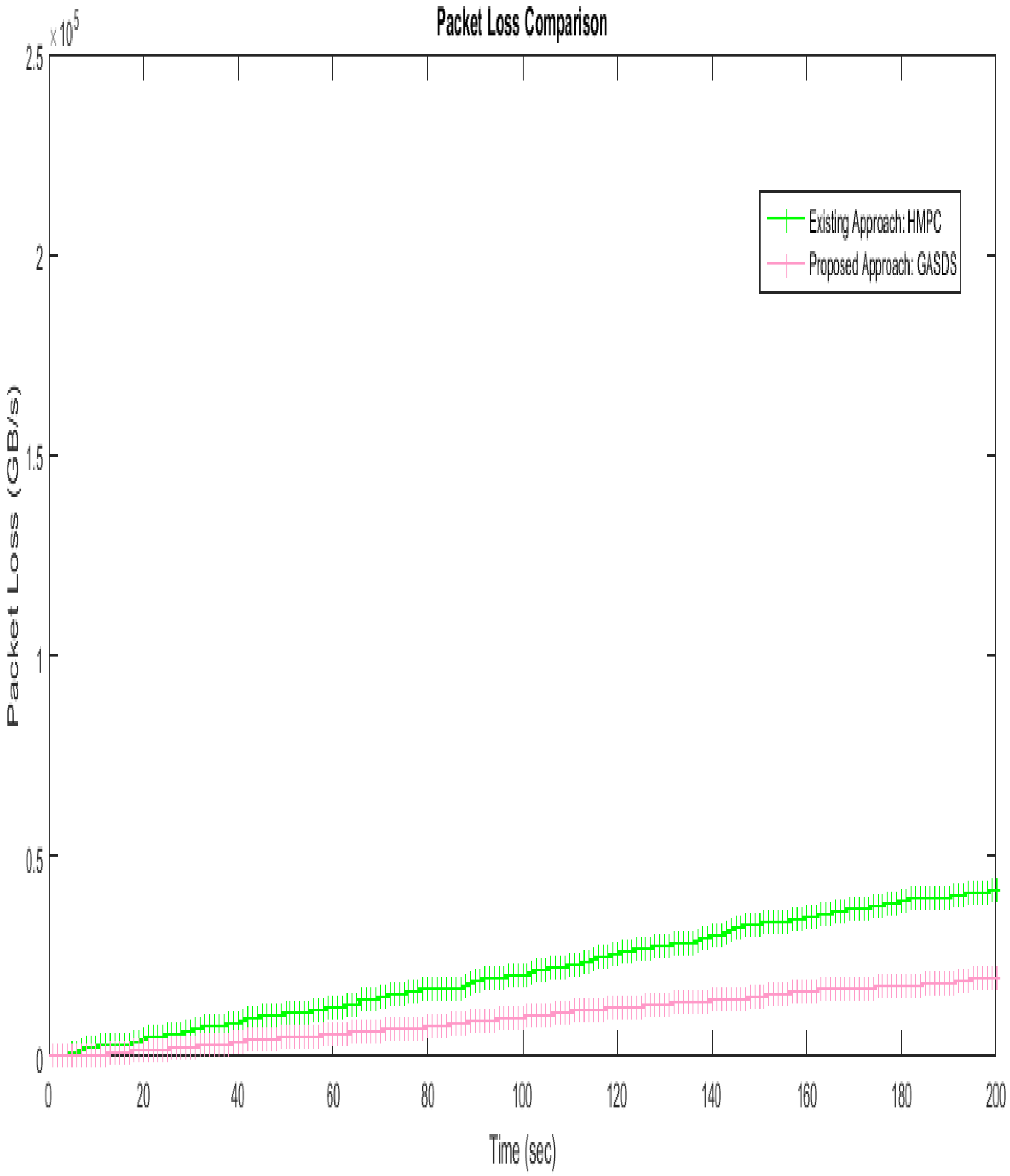

Figure 16: Comparison of Packet loss

The above figure demonstrates the graphical representation of packet loss with respect to time. The green colored marks indicate the packet loss of the existing technique, whereas pink color indicates the packet loss of the proposed technique, which is better than the existing technique.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

IoT offers a viable resource scheduling by ensuring their maximal utilization and services to the clients. IoT acquires features from different computing environments such as mobile computing, pervasive computing, and wireless sensor networks. From this, it is clear that IoT relies on upon different basic capabilities and innovations like machine learning, signal processing, real-time computing, big data analytics involves data analysis, data curation, data sharing, data storage, data transferring, and visualization. Security and protection concerns are all over the place, which are definitive ranges of information communication where threats and attacks are to be recognized and resolved. The study of this work suggests various ways to improve the efficiency of the remote sensor network, in the sense, reduction in latency during a sudden movement of nodes by using appropriate protocols, reduction in packet loss ratio, consumption of bandwidth, power, and other resources along with packet size, increasing the network lifetime and improving the QoS using the proposed SDS metric and replacement algorithm for device connectivity issues, which does not considers all these into account, but focusing on two major performance factors, by maximizing throughput and minimizing the packet loss.

The proposed GASDS approach uses a metric called “Strength of Device Synergy” (SDS) to measure the strength of the connectivity among the nodes in the network. SDS is capable of differentiating the clusters that have strong and weak connected nodes. Apart from this, GASDS is capable of repairing the connectivity failures among the nodes. Simulation results show that the proposed GASDS outperforms the existing HMPC Approach in terms of throughput and packet loss.

6.2 Future Scope

In the future scope, we can further improve the results, other than throughput and packet loss by modifying the proposed technique or by using any different technique instead of GASDS approach and make it hybrid. There could be a possibility to decrease the energy consumption and bandwidth reduction by using sophisticated techniques. Energy consumption is also one of the major concerns which is to be resolved in the field of WSN. The proposed approach can further be enhanced to resolve such issues as well.

REFERENCES

[1] D. A. D. Ayers, “From there to here.,” *IEEE Internet Comput.*, vol. 11, no. 1, p. 75, 78–80, 82 passim, 1986.

[2] E. Mathews and G. Muller, “Transition from closed system to Internet of Things: A study in standardizing building lighting systems,” *2016 11th Syst. Syst. …*, 2016.

[3] F. daCosta, “Rethinking the Internet of Things: A Scalable Approach to Connecting Everything,” *IEEE Internet Things J.*, p. 192, 2013.

[4] S. M. Kintner, “Wireless telegraph receivers,” *IEEE Internet Things J.*, pp. 65–71, 1906.

[5] D. Bonino, E. Castellina, and F. Corno, “The DOG Gateway : Enabling Ontology-based Intelligent Domotic Environments,” *IEEE Internet Things J.*, vol. 54, no. 4, pp. 1656–1664, 2008.

[6] J. Pan, R. Jain, S. Paul, T. Vu, A. Saifullah, and M. Sha, “An Internet of Things Framework for Smart Energy in Buildings: Designs, Prototype, and Experiments,” *IEEE Internet Things J.*, vol. 2, no. 6, pp. 527–537, 2015.

[7] Y. Fu and W. Wu, “Predicting household water use behaviour for improved hygiene practices in internet of things environment via dynamic behaviour intervention model,” *IEEE Internet Things J.*, vol. 5, pp. 143–151, 2016.

[8] Z.-T. Zhu, M.-H. Yu, and P. Riezebos, “A research framework of smart education,” *Smart Learn. Environ.*, vol. 3, no. 1, p. 4, 2016.

[9] I. Rom and J. Rivadeneira, “Towards a Smart Classroom : Development of an Augmented Reality Application for Education and Tele-education,” *IEEE Internet Things J.*, pp. 395–400, 2015.

[10] A. Bogdanski, “Integrated food–energy systems for climate-smart agriculture,” *Agric. Food Secur.*, vol. 1, no. 1, p. 9, 2012.

[11] S. Sicari, A. Rizzardi, and L. A. Grieco, “Security , Privacy & Trust in Internet of Things : the road ahead,” *Comput. NETWORKS*, no. November, 2014.

[12] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, “Connected vehicles: Solutions and challenges,” *IEEE Internet Things J.*, vol. 1, no. 4, pp. 289–299, 2014.

[13] O. Bello and S. Zeadally, “Intelligent Device-to-Device Communication in the Internet of Things,” *IEEE Syst. J.*, vol. 10, no. 3, pp. 1172–1182, 2016.

[14] B. Khoo, “RFID - From tracking to the internet of things: A review of developments,” *Proc. - 2010 IEEE/ACM Int. Conf. Green Comput. Commun. GreenCom 2010, 2010 IEEE/ACM Int. Conf. Cyber, Phys. Soc. Comput. CPSCom 2010*, pp. 533–538, 2010.

[15] K. Ali, W. Alsalih, and H. Hassanein, “Set-cover approximation algorithms for load-aware readers placement in RFID networks,” *IEEE Int. Conf. Commun.*, 2011.

[16] J. E. Said, J. Almhana, and L. Karim, “Efficient Connectivity Preserving Clustering Protocol for Mobile Sensor Network,” *IEEE Internet Things J.*, pp. 752–756, 2016.

[17] Y. Huang *et al.*, “Cluster-Head and Border-Node Based Cluster Routing Protocol for LR-WPAN,” *2016 IEEE 30th Int. Conf. Adv. Inf. Netw. Appl.*, pp. 132–139, 2016.

[18] D. P. Singh, R. H. Goudar, B. Pant, and S. Rao, “Cluster head selection by randomness with data recovery in WSN,” *CSI Trans. ICT*, vol. 2, no. 2, pp. 97–107, 2014.

[19] J. Vesanto, E. Alhoniemi, and S. Member, “Clustering of the Self-Organizing Map,” *IEEE Internet Things J.*, vol. 11, no. 3, pp. 586–600, 2000.

[20] M. A. Razzaque, M. Milojevic-jevric, and A. Palade, “Middleware for Internet of Things : a Survey,” *Ieee Internet Things J.*, vol. 3, no. 1, pp. 70–95, 2016.

[21] B. Mukherjee, “Villa : An Event Based Middleware for Real-time Collaboration on the Internet,” *IEEE Internet Things J.*, pp. 65–74, 1997.

[22] Y. Sun, R. Bie, P. Thomas, and X. Cheng, “Advances on data, information, and knowledge in the internet of things,” *Pers. Ubiquitous Comput.*, vol. 18, no. 8, pp.

1793–1795, 2014.

[23] A. Nijholt, J. Zwiers, and J. Peciva, “Mixed reality participants in smart meeting rooms and smart home environments,” *Pers. Ubiquitous Comput.*, vol. 13, no. 1, pp. 85–94, 2009.

[24] J. A. Carretero and M. A. Nahon, “Solving minimum distance problems with convex or concave bodies using combinatorial global optimization algorithms,” *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 35, no. 6, pp. 1144–1155, 2005.

[25] M. S. Manshahia and M. Dave, “Firefly Algorithm Based Clustering Technique for Wireless Sensor Networks,” *IEEE Internet Things J.*, pp. 1273–1276, 2016.

[26] A. Yousif, S. M. Nor, and A. H. Abdualla, “Job Scheduling Algorithms on Grid Computing : State-of- the Art,” *Int. J. Grid Distrib. Comput.*, vol. 8, no. 6, pp. 125–140, 2015.

[27] K. Mizuno, S. Takamatsu, T. Shimoyama, and S. Nishihara, “Fireflies can find groups for data clustering,” *Proc. IEEE Int. Conf. Ind. Technol.*, vol. 2016–May, pp. 746–751, 2016.

[28] P. Lalwani, I. Ganguli, and H. Banka, “FARW: Firefly algorithm for Routing in wireless sensor networks,” *Int. Conf. Recent Adv. Inf. Technol.*, pp. 248–252, 2016.

[29] A. Sharma, “Image Segmentation using Firefly Algorithm,” *IEEE Internet Things J.*, 2016.

[30] A. Elahi, I. M. Qureshi, M. Atif, and N. Gul, “Interference reduction in Cognitive radio networks using Genetic and Firefly Algorithms,” *2017 Int. Conf. Commun. Comput. Digit. Syst.*, no. 1, pp. 96–100, 2017.

[31] S. Bilgaiyan, S. Sagnika, and M. Das, “Workflow Scheduling in Cloud Computing Environment Using Cat Swarm Optimization,” *IEEE Internet Things J.*, no. 1, pp. 680–685, 2014.

[32] Z. Zheng, H. Liu, and L. Chen, “An Evolutionary Many-Objective Optimization Algorithm Based on Population Decomposition and Reference Distance.” *IEEE*

Internet Things J., 2016

[33] N. Heo and P. K. Varshney, "Energy-efficient deployment of Intelligent Mobile sensor networks," *IEEE Trans. Syst. Man Cybern. Part A Syst. Humans*, vol. 35, no. 1, pp. 78–92, 2005.

[34] S. Robinson, T. Hunniford, and S. Cook, "The application of optimal foraging theory in route finding algorithms for ad-hoc networks," *Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern.*, vol. 1, pp. 444–449, 2004.

[35] S. A. Kumar, T. Vealey, and H. Srivastava, "Security in Internet of Things: Challenges, Solutions and Future Directions," *Hawaii Int. Conf. Syst. Sci.*, pp. 5772–5781, 2016.

[36] J. P. Cofré, G. Moraga, C. Rusu, I. Mercado, R. Inostroza, and C. Jiménez, "Developing a Touchscreen-based Domotic Tool for Users With Motor Disabilities," *IEEE Internet Things J.*, 2012.

[37] N. S. V Rao, Q. Wu, S. S. Iyengar, and A. Manickam, "Connectivity-through-time protocols for dynamic wireless networks to support mobile robot teams," *ICRA 2004 Proc. 2004 IEEE Int. Conf. Robot. Autom.*, pp. 1653–1658, 2003.

[38] M. P. Andersen, G. Fierro, and D. E. Culler, "Enabling synergy in IoT: Platform to service and beyond," *Proc. - 2016 IEEE 1st Int. Conf. Internet-of-Things Des. Implementation, IoTDI 2016*, pp. 1–12, 2016.

[39] X. Li, R. Lu, X. Liang, X. Shen, J. Chen, and X. Lin, "Smart community: An internet of things application," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 68–75, 2011.

[40] S. Lee and M. Lee, "QRMSC: Efficient QoS-aware relay node placement in wireless sensor networks using Minimum Steiner tree on the Convex hull," *Int. Conf. Inf. Netw.*, pp. 36–41, 2013.

[41] M. D. S. Nandhini, "An Optimized Network Throughput and Delay Control in Hybrid Multi Channel Wireless Mesh," *IEEE Internet Things J.*, vol. 3, no. Ii, 2015.

[42] D. S. Kushwaha, V. K. Singh, S. Singh, and S. Sharma, “REDUCING PACKET LOSS USING MDRED TECHNIQUE IN ACTIVE,” *IEEE Internet Things J.*, no. 1, pp. 40–43, 2015.

[43] A. M. Ali and P. Sankar, “Network Routing Using Zone Based Ant Colony Optimization in VANET,” *Int. J. Adv. Res. Electr.*, vol. 4, no. 3, pp. 1706–1711, 2015.

[44] F. Xing and W. Wang, “On the Survivability of Wireless Ad Hoc Networks in the Presence of Routing Malfunction,” *IEEE Trans. Dependable Secur. Comput.*, vol. 7, no. 3, pp. 284–299, 2008.

[45] P. Rathee, “Swarm Intelligence: An Overview,” *IEEE Internet Things J.*, vol. 4, no. 1, pp. 96–98, 2015.