King Saud University
College of Computer & Information Sciences
Computer Science Department

Design and Performance Study of a Cooperative Energy Efficient Routing Algorithm for Wireless Sensor Networks

Prepared By:

Najla A. Al-Nabhan

426221070

Supervised By:

Dr. Sami S. Al-Wakeel

Dr. Jalal F. Al-Muhtadi

Submitted in Partial Fulfillment of the Requirements for the Master's Degree in the Department of Computer Science at the College of Computer and Information Sciences

King Saud University

Riyadh

1429 H / 2008 G

Project Title:

Design and performance study of a cooperative energy efficient routing algorithm for wireless sensor networks.

Field: Wireless Sensor Network

Prepared By: Najla Abdulrahman Al-Nabhan

Supervisor: Dr. Sami Saleh Al-Wakeel

Co-Supervisor: Dr. Jalal Farid Al-Muhtadi

Examination Committee members:

Dr. Sami Saleh Al-Wakeel :________________________

Dr. Jalal Farid Al-Muhtadi :________________________

Dr. Yousef Ahmed Al-Ohali:_______________________

Abstract

Advances in the miniaturization of micro-electromechanical systems have led to battery-powered sensor nodes that have sensing, communication and processing capabilities. These sensor nodes can be networked in an ad hoc manner to perform distributed sensing and information processing. Such ad hoc sensor networks provide greater fault tolerance and sensing accuracy and are typically less expensive compared to the alternative of using only a few large isolated sensors. These networks can also be deployed in inhospitable terrains or in hostile environments to provide continuous monitoring and processing capabilities [1].

In wireless sensor network (WSN), sensor collects data from the monitored area (such as temperature, sound, vibration, pressure, motion, or pollutants). Then it routes data back to the base station (BS) [2,3]. Data transmission is usually a multi-hop from node to node toward the base station. The scarcest WSN's resource is energy, and one of the most energy-expensive operations is route discovery and data transmission. For this reason, algorithmic research in WSN mostly focuses on the study and design of energy aware algorithms for routing and data transmission.

In this project, we propose a design for a cooperative energy-efficient routing algorithm (CEERA) for wireless sensor network. In CEERA, nodes cooperate in delivering data to the base station, and control packet retransmission through an address-based timer. Besides, CEERA achieves overhead reduction through controlling various parameters that affect energy dissipation. The Algorithm is expected to minimize the overall energy consumption of the WSN and improves the lifetime of network's nodes.

Keywords: *WSN routing algorithm, WSN performance, WSN simulation, Energy-Efficient routing, System Design, Simulations, Network measurements, Performance study.*

Table of Content:

Table of Figures:

Table of Tables:

CHAPTER 1: Introduction

1.0. Introduction to WSN

The wireless sensor networks of the near future are envisioned to consist of hundreds to thousands of inexpensive wireless nodes [4]. Basically, each sensor node comprises sensing, processing, transmission, mobilizing, position finding system, and power units (some of these components are optional, like the mobilizer). These sensors have the ability to communicate either among each other or directly to an external base station (BS) [3]. Sensor nodes are usually scattered in a sensor field, which is an area where the sensor nodes are deployed. They coordinate among themselves to produce high-quality information about the physical environment. Each sensor node bases its decisions on its mission, the information it currently has, and its knowledge of its computing, communication, and energy resources [5].

Sensors are intended for a broad range of environmental sensing applications from vehicle tracking to habitat monitoring [5]. They can be deployed for habitat modeling, temperature monitoring and industrial sensing. They also find applications in battlefield awareness and emergency (first) response situations [6]. In most of the applications sensors are required to detect events and then communicate the collected information to a distant BS (base station) where parameters characterizing these events are estimated.

In most of the applications sensors are required to detect events and then communicate the collected information to a distant BS (base station) where parameters characterizing these events are estimated. When a sensor node needs to initiate communication with a certain destination terminal, a routing operation is necessary to find a route from the source to the destination [4]. Since sensor nodes are severely constrained by the amount of battery power available, it is necessary for communication protocols to conserve nodes' energies so as to maximize their lifetime [4]. The untended nature of sensor nodes and the hazardous sensing environment preclude manual battery replacement. For these reasons, energy awareness becomes the key research challenge for sensor network protocol design. Several researchers have addressed energy conservation recently. The conventional on-demand routing protocols ask for network-wide flooding to discover routes [4]. Flooding-based route discovery generally causes large routing overhead and high energy consumption.

While the cost of transmitting a bit is higher than a computation [4], Limiting number of transient nodes that are involved in data transmission toward the base station would limit number of transmitted replicated data packets which directly affects energy dissipation and network lifetime.

1.1. WSN Applications:

A typical sensor network application is inventory tracking in factory warehouses. A single sensor node can be attached to each item in the warehouse. These sensor nodes can then be used for tracking the location of the items as they are moved within the warehouse. They can also provide information on the location of nearby items as well as the history of movement of various items. Another typical application of sensor networks lies in military situations. Sensor nodes can be air-dropped behind enemy lines or in inhospitable terrain. These nodes can self-organize themselves and provide unattended monitoring of the deployed area by gathering information about enemy defenses and equipment, movement of troops, and areas of troop concentration. They can then relay this information back to a friendly base station for further processing and decision making[1].

1.2. Typical Sensor Nodes Specification:

Sensor nodes are typically characterized by small form-factor, limited battery power, and a small amount of memory. Sensor nodes rely on wireless channels for transmitting data to and receiving data from other nodes. The *maximum distance* that a node can communicate with another node is characterized by the communication unit on the sensor node. The maximum operation range is approximately up to 100ft (30.48 meters). The *sensing area* of a sensor node depends on the type of physical sensors used on that node. A range sensor is able to detect a target from 6 inches away up to a distance of 35 feet.

Figure 1.1 illustrates the basic structure of a sensor node. The attributes of some commercially available nodes are listed in Table 1.1.

Table 1.1: Specifications of inexpensive wireless sensors [1]

Model	MICA2DOT	MICA2	MICAz
Battery	3V coin cell	AA $\times$ 2	AA $\times$ 2
Size (mm)	25 $\times$ 6	58 $\times$ 32 $\times$ 7	58 $\times$ 32 $\times$ 7
Weight (g)	3	18	18
Range (m)	150	1503̃00	751̃00
Data rate	38.4 KBaud	38.4 KBaud	250 Kbps
Program flash memory	128 KB	128 KB	128 KB
Serial flash memory	512 KB	512 KB	512 KB
EEPROM	4 KB	4 KB	4 KB
RF (MHz)	315/433/868/916	315/433/868/916	2400

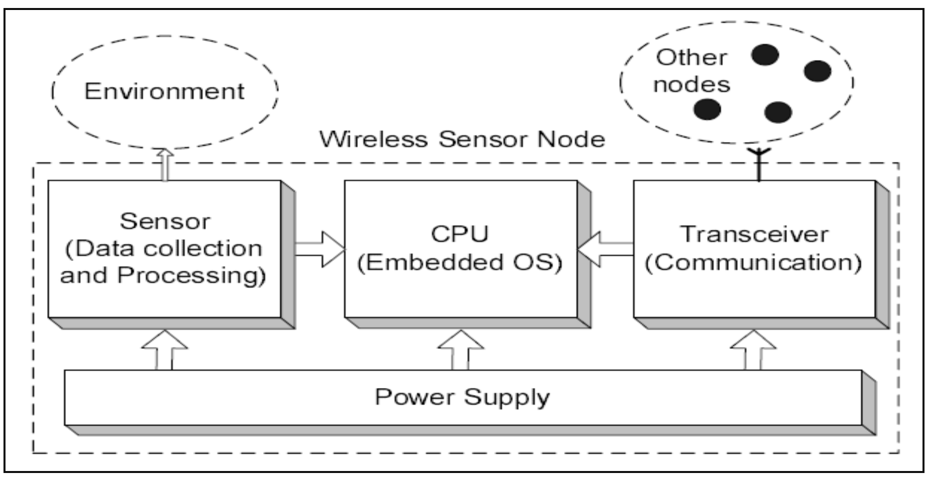


Fig.1.1: Sensor Node Architecture

1.3. Challenges:

In this section, we motivate the challenges involved in the design of self organization and data dissemination protocols in wireless sensor networks. Due to their resource constraints and unique application requirements, sensor networks pose a number of challenges. These are summarized below:

1.3.1. Deployment:

Wireless sensor networks differ from other ad hoc networks by requiring a deployment phase. Once deployed, the sensor network needs very little human intervention and can function autonomously. Thus, deployment phase is extremely important to any WSN. It affects the overall network coverage, performance and connectivity. However, it becomes impossible to deploy a uniform distribution. Deployment is often made in inhospitable locations and sometimes in environments where the ambient conditions can significantly vary. In some places, it is not feasible to deploy sensor nodes in a grid arrangement. Sensors are normally placed in special environments without guarantee of position [1]. On the other hand, applications such as monitoring a cyclone require a fast deployment phase, since this kind of phenomenon is not predictable, and it is necessary to distribute sensor nodes as quick as possible to maximize the coverage area, connectivity, etc [1,2].

Sensor node deployment problems have been studied in a variety of contexts and many algorithms were proposed for the deployment of WSNs. In general, deployment is either **random** or **deterministic**. In random deployment**,** sensor nodes are deployed by air-dropping them or throwing them randomly in a target area. Using a deterministic scheme, sensors placed at pre-determined locations. Protocols for self-configuration of a randomly-deployed network may not be well suited for a deterministically-deployed sensor network. Similarly, data dissemination algorithms designed for deterministic sensor networks may not perform well when used in randomly-deployed sensor networks.

1.3.2. Resources of Sensor Nodes:

An important consideration in sensor networks is the amount of energy and storage required for sensing, computation, and communication. Due to their limited resources, many of the methods developed for the Internet and mobile ad hoc networks cannot be directly applied to sensor networks. Furthermore, radio communication typically costs more in terms of energy compared to computation costs in a sensor node. Therefore, protocols designed for sensor networks should utilize only a few *control messages [1]*.

1.3.3. Fault Tolerance:

Sensor nodes are prone to failure. This may be due to a variety of reasons. Loss of battery power may lead to failure of the sensor nodes. Similarly, when sensor nodes are deployed in hostile or harsh environments as in the case of military or industrial applications, sensor nodes might be easily damaged. Thus, protocols designers should build fault tolerance into their algorithms for improving the utility of sensor networks [1].

1.3.4. Scalability:

The number of sensor nodes in a sensor network can be in the order of hundreds or even thousands. Hence, protocols designed for sensor networks should be highly scalable [1].

1.4. Objectives:

The main goal of this project is to design and study the performance of an energy-efficient routing algorithm for a wireless sensor network (WSN). In this algorithm, we attempt to minimize routing overhead and energy consumption by:

1. Eliminating rout discovery and reconfiguration operation (sitting and selecting cluster heads and rout table update)

2. Using few control messages (only BS Acknowledgment)

3. Avoiding redundant message transmission and duplicated arrival by delaying transmission to a variable amount of time relevant to its address and some scalar factor.

4. Managing congestion and extra memory occupation resulted by in appropriate message storage and/or transmission using duplication factor and BS acknowledgment.

5. Considering reality of environment situation by using randomization at different stages (deployment, message origin and arrival time, and adjustment to base station location)

In CEERA, the tendency of a sensor node to cooperate in message retransmission is varies from node to another. Furthermore, its trend to store a message is also decreases with every duplicate receiving, which indicates that other nodes carry out the transmission. We performed a detailed study for the effect of changing control parameters over network performance and how to obtain optimal values of these factors.

1.5. Problem Statement:

The idea of this project is to design a routing protocol that keeps network connectivity while achieving a minimum delay and energy dissipations.

1.6. Problem Solving Methodology:

In this project, we used simulation extensively to design the project routing scheme, which is a powerful technique for studying the behavior of computer network. We developed network model in phase 1of this project. Based on that model, we implemented program that simulates our routing algorithm. We extract results from simulation to investigate and analyze the performance of our algorithm.

1.7. Literature Review:

The wireless sensor networks are an active research area with numerous workshops and conferences arranged each year. One of the earliest and important research topics in the literature of WSN are those related to providing the theoretical framework and required information to design an efficient WSN routing protocol. For this, our project references on sensor network can be categorized to the following:

A. General survey of sensor networks and its security issues.
B. General routing algorithms of sensor networks and its design.
C. Energy-efficient routing algorithms.

Table 1.2 shows a list of various references and these main categories.

Category	Sub-categories	Approaches
Survey	Generic	[3] [6] [7] [8] [9]
Routing Algorithm Design & Performance	General Routing Algorithms	[2][10]
	Energy-efficient Routing Algorithms	[4][5][11][12] [13][14][15][16]

Table 1.2: List of various WSN references and their categories.

In what follow, we describe selected references of each category used in this proposal:

A) References Survey of sensor network and its security

1. Reference [3], discuss the role of the Sensor networks in the upcoming period of pervasive computing, and presents the issues and consider mechanisms to achieve secure communication in these networks.

2. Reference [6], discuss application of sensor network in sensitive fields like a battlefield awareness to sense chemicals and other gases used in chemical and biological warfare. It focuses on the reliability of data and event transfer in such applications. It proposes a reliable event transfer mechanism making use of an overlay network of relay nodes. The overlay network removes the burden of data relaying from the sensor nodes and results in increasing the lifetime of the network. Simulation results prove the benefits of such architecture. Reliability is increased 10-30% with reduction in event traffic of 60-80%

3. Reference [7], outline a survey on sensor networks and sensor nodes specifications, motivation, Design Criteria, Application, the ways of classification of sensor network, and last a detail description of routing protocol of sensor networks.

4. Reference [8], addresses some of the key design considerations for future micro-sensor systems including the network protocols required for collaborative sensing and information distribution, system partitioning considering computation and communication costs, low energy electronics, power system design and energy harvesting techniques.

5. Reference [9], discuss the architecture, possible application, challenges, and requirement of sensor networks.

B) References for General Routing algorithm of sensor networks and its design

1. Reference [10], presents a comprehensive taxonomy of the various routing techniques in wireless sensor networks, first it outline the design challenges for routing protocols in WSNs followed by a comprehensive survey of routing techniques, and demonstrate the categories based on the underlying network structure: flit, hierarchical, and location-based routing. Also the paper study the design trade-offs between energy and communication overhead savings in every routing paradigm, also highlights the advantages and performance issues of each routing technique. The article concludes with possible future research areas.

2. Reference [11], present Greedy Perimeter Stateless Routing (GPSR), a novel routing protocol for wireless datagram networks that uses the *positions* of routers and a packet's destination to make packet forwarding decisions. It describe the GPSR protocol, and use extensive simulation of mobile wireless networks to compare its performance with that of Dynamic Source Routing. The simulations demonstrate GPSR's scalability on densely deployed wireless networks.

C) References for Energy-Efficient Routing algorithm of sensor networks

1. Reference [4], presents an optimal energy-adaptive clustering algorithm which is motivated from the LEACH protocol presented in [11]. They optimize LEACH's random cluster-head selection algorithm to ensure the balanced energy depletion over the whole network thus prolongs the network lifetime. Simulation results

show that this modification over LEACH outperforms LEACH by about 20% to 35% when 1%, 50%, 100% of nodes die for different network sizes and topologies.

2. Reference [5], presents a family of adaptive protocols, called SPIN (Sensor Protocols for Information via Negotiation), that efficiently disseminates information among sensors in an energy-constrained wireless sensor network. Nodes running a SPIN communication protocol name their data using high-level data descriptors, called metadata. They use meta-data negotiations to eliminate the transmission of redundant data throughout the network. In addition, SPIN nodes can base their communication decisions both upon application-specific knowledge of the data and upon knowledge of the resources that are available to them. Simulation shows that SPIN protocols can deliver 60% more data for a given amount of energy than conventional approaches.

3. Reference [11], proposed a protocols named LEACH (Low-Energy Adaptive Clustering Hierarchy), which is a clustering-based protocol that utilizes randomized rotation of local cluster base stations (cluster-heads) to evenly distribute the energy load among the sensors in the network. LEACH uses localized coordination to enable scalability and robustness for dynamic networks, and incorporates data fusion into the routing protocol to reduce the amount of information that must be transmitted to the base station.

4. Reference [12], introduces a region based routing that considers an efficient routing operations between any two nodes in an ad hoc network that is linked to wired networks by an access point. To build routes with low routing overhead efficiently, the RBR utilizes hop counts between mobile nodes and the access point to localize a route discovery within a limited topological region. Limiting the region of route discovery results in fewer routing messages and therefore reduces routing overhead.

5. Reference [13], considers issues in the design of a simple, Scalable, Energy-efficient Location Aided Routing (SELAR) protocol for WSN. In SELAR, location

and energy information of neighboring nodes together with the location information of the sink node are used to perform the routing function.

6. Reference [14], describes an energy efficient self-organization protocol designed for a wireless sensor network. It considers keeping sensor nodes turned off as long as possible as the most efficient way to save energy is. These sleeping-or-awaking nodes need the capabilities of self-organization and re-organization to adapt to dynamic environment and network settings.

7. Reference [15], discuss the address-free sensor network. It propose an Efficient Data Delivery Scheme (EDDS) for address-free wireless sensor networks, in which data source nodes randomly select a probabilistically unique identifier for each data flow instead of statically assigned node addresses. The packet forwarding mechanism in EDDS combines the tasks of routing and MAC layer via cross-layer design, which simplifies the protocol stack while maintaining the necessary functionality.

8. Reference [16], discuss the utilization of clustering to achieve energy efficiency for the on-off wireless sensor network, whose member nodes alternate between active and inactive states. In the proposed Distributed and Energy Efficient Self-Organization (DEESO) scheme, the head election is adjusted adaptively to the energy reserve of local active nodes. Furthermore, it applies the Adaptive Channel Assignment to address the on-off topology changes. Simulation results show that DEESO delivers 184% amount of data to the base station as LEACH for the same amount of dissipation and the effective network lifetime is extended around 56%.

1.8. Project Outlines

In this chapter, we introduced project's main objectives. Also, we provide general introduction to the subject of WSN routing. The rest of this document is organized as follow: Chapter 2 presents the design strategy of CEERA. It discusses in details network model, deployment model, traffic model, performance measure, and power model. In Chapter 3, we present our simulation model. Chapter 4 includes results and performance study. Finally, in chapter 5 we conclude our work in this project.

CHAPTER 2: Design and Modeling of CEERA

2.0. Introduction

In order to study the performance of CEERA, it is required to monitor the behavior of each node, the interaction between nodes and the overall network state. We build the following network model that specifies network, traffic, deployment, sensor nodes, and type of massages exchanged in the network.

2.1. Network Model:

In any WSN, data being sensed by the nodes in the network must be transmitted to a control center or base station, where the end-user can access the data. There are many possible models for these sensor networks, we consider sensor networks where:

- The BS is fixed at a far distance from the sensor nodes.
- The sensor nodes are homogeneous and energy constrained with uniform energy.
- Nodes have no location information.
- Not all nodes are able to reach BS.
- Symmetric propagation channel: that means the energy required to transmit a message from node A to node B is the same as the energy required for transmitting a message from node B to node A [4,11].

We assumed that nodes are located in Cartesian coordinates and the base station is located at a fixed location [e.g. initially at the origin at (0,0) but it might be placed at any (x, y) location]. Nodes are randomly distributed and positioned across a Cartesian coordinate grid. A given number of nodes **S** are distributed randomly around the base station (origin) in a square region of side length equals **50 meters.**

For every node of the 2^m nodes, we assign a fixed binary address of **M** bits. Limiting number of nodes to S, resulted in limiting the address space in data packet to be M=**Log (S)** bits for both the massage source node and transient node. We limits the distance that node traveling signal could be reached by its neighbor to a number $\mathbf{D_{max}}$, maximum receiving distance between nodes, except the base station signals that can be received by all nodes in the network.

2.1.1. Node Specification:

We adapt the general component of a sensor node, where a node is decomposed of the four basic components, as shown in Fig. 1: a sensing unit, a processing unit, a transceiver unit, and a power unit. The additional application-dependent components such as a location finding system, power generator, and mobilizer are not considered in our network model since they are not utilized in CEERA. Sensing units are usually composed of two subunits: sensors and analog-to-digital converters (ADCs). The analog signals produced by the sensors based on the observed phenomenon are converted to digital signals by the ADC, and then fed into the processing unit. The processing unit, which is generally associated with a small storage unit, manages the procedures that make the sensor node collaborate with the other nodes to carry out the assigned sensing tasks. A transceiver unit connects the node to the network. One of the most important components of a sensor node is the power unit. Power units may be supported by power scavenging units such as solar cells. All of these subunits may need to fit into a matchbox-sized module [3].

2.1.2. Specification of Data Messages

In our model, every data message (packet) has its unique identification number (ID) represented in X bits. Thus, up to 2^x messages can be transmitted simultaneously by a node before it refreshes its counter. The size of message content (data) is also limited to K bits. Every data message in our network contains: M-bits source node address, M-bits address for every transient node, K-bits data, and X-bits message ID. In total, a data message maximally contains: *[M+K+X]* bits. Also, an acknowledgment packet (ACK) is decomposed of source node address (M-bits), and message ID (X-bits).

2.2.Deployment Model:

As the selected topology influences the outcome of the simulation, realistic topologies are required to produce realistic simulation results. In section 1.3.1 we discussed the deployment schemas of WSNs. The commonly used and realistic deployment strategy is random deployment. This project is based on random deployment where sensors are distributed randomly in the target area of size 50 X 50 meters. Ns2 (Network simulator2) has a built in utility for generating random network topology for wireless sensor network. We used this module to assign a random location for each node of 100 nodes in our network. Resulted topology is shown in next figure. Base station location is extremely important to show how protocol is effectively works with variable network layout. For this we experiment different locations for base station: at origin, center and two random locations.

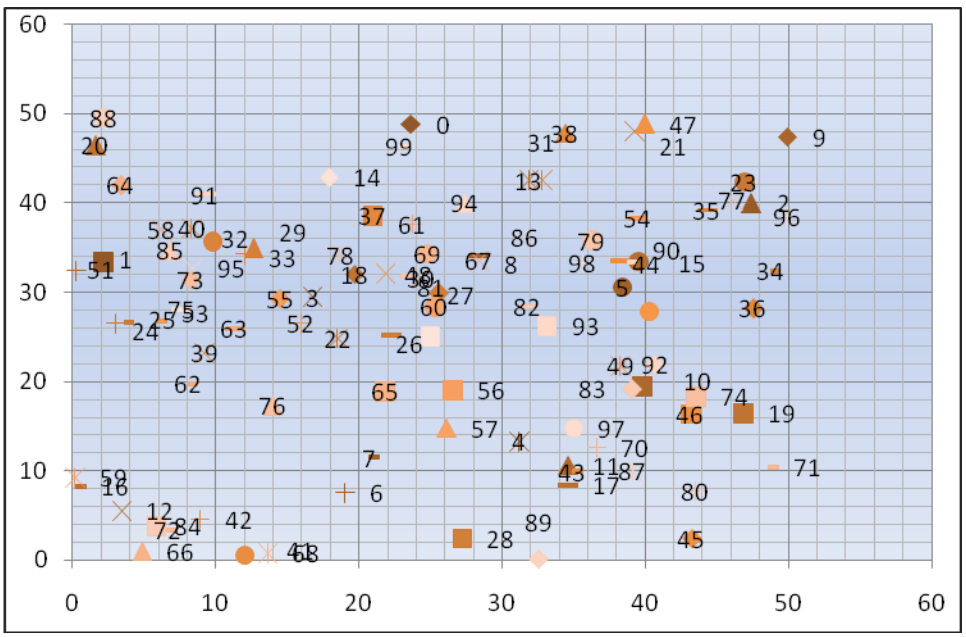


Fig.2.1: Random network topology

2.3.Traffic Model:

In out model, the arrival process of a new massage is characterized by Poisson process, which is a stochastic process time that's used for modeling random events in time that occur to a large extent independently of one another. The inter-arrival time between message arrivals is environment dependent. However, we studied its effect on network performance.

2.4.Energy Model

An important consideration in sensor networks is the amount of energy required for sensing, computation, and communication. Different assumptions about the radio characteristics, including energy dissipation in transmit and receive modes, will change the advantages of different protocols. In our work, we use a radio model similar to the radio model discussed in [4]. In this model, a radio dissipates E_{elec} = 50 nJ/bit in the transmitter or receiver circuitry. And ε_{amp}=100 pJ/bit/m2 for the transmitter amplifier to achieve an acceptable E_b/N_o. to transmit a *k*-bit message a distance *d*, the radio expends:

$$E_{Tx}(k, d) = E_{elec} * k + \varepsilon_{amp} * k * Dmax \qquad (1)$$

To receive this message, the radio expends:

$$E_{Rx}(k) = E_{elec} * k \qquad (2)$$

2.5.Performance Measures:

In this section, we define set of performance measures used to describe network performance. These measures represent special challenge for the design of routing algorithm in different network conditions. Main WSN performance measures are:

- Throughput: Number of successful transmissions that have arrived to the base station.

- Delay: The time difference between message transmissions at the source node until the time it received by the base station.

- Delay Time Jitter (DTJ): The variation in inter-packet arrival time as introduced by variable transmission delay over the network is called "jitter". Its value shows the variation of a metric (delay) with respect to some reference metric (average delay). The highest DJT value indicates that large variation between inter-packet arrivals.

- Total Energy dissipations: The total amount of energy dissipated from all nodes in the network.
- No. of died nodes: No of died nodes
- First Node to die lifetime (FND): Indicates the duration for which the sensor network is fully functional. For some sensitive applications such as intrusion and fire detection, it is important to know when the first node dies.
- Beta of nodes to die lifetime: Total number of messages received until a β fraction of the nodes die. This indicates the amount of information collected until that time [16].
- Half of the Nodes Alive (HNA) lifetime: It indicates the duration of time for which only the half of nodes in the sensor network are alive.
- Last Node to die lifetime (LND): This measure denotes the duration for which all nodes in network dies.
- Redundant (duplicated) Arrival (DA): This measure shows number of duplicated arrival of messages to the base station. Duplicated arrival is not preferred as it causes extra dissipation of energy. We include this measure to test CEERA's ability to limit and prevent redundant arrival through controlling transient transmission.
- Congestion (Overhead): In computer network, Congestion occurs when there is too much traffic on the network and all the server requests cannot be processed quickly. When offered load exceeds the capacity of a data communication path.

We assumed that congestion occurs when a node receives the same data message several times (n). It is undesired to waste more energy for receiving the same message and also to block memory for storing the message while other neighbors carry out the transmission of this message. We have monitored CEERA's ability to minimize and control congestion as its affects both energy and memory. We threshold duplication tolerance to an introduced factor called “duplication factor”. Through simulation, we studied the effect of this value on network performance.

2.6. Routing Design:

2.6.1. CEERA Design

In CEERA, data transmission is cooperative from one node to another until it received by the base station. The base station sends an acknowledgement for every received message. Its transmissions can be received by all sensor nodes. Acknowledgments include important information for controlling both message transmissions and storage. When a source node decides to transmit its sensed data, it generates a packet with a unique ID and transmits the packet after appending its ID. Also data message is flagged to be transmitted in either source-rout mode or cooperative mode. Based on its energy level, a node may not participate in data transmission if its current energy is less than a predefined energy threshold (**E_{min}).**

In CEERA, data transmission from a source node can be direct to BS (if range is acceptable). If not received, transmission is cooperative as each transient sensor node in N_s who received the transmission will carry the following:

1) Calculates the ID difference as =$ID_d = ID_t - ID_s$
2) Starts a timer counter with value ID_d * scale factor
3) Listens to BS's ACK, and periodically decrements its timer.

If the acknowledgment doesn't received within the timer value, the transient node retransmit message with appending its address to the address list of transient nodes. This process is repeated by every node until ACK is heard. Upon ACK reception all nodes clear the call and reset their counters.

2.6.2. CEERA Design Flowchart:

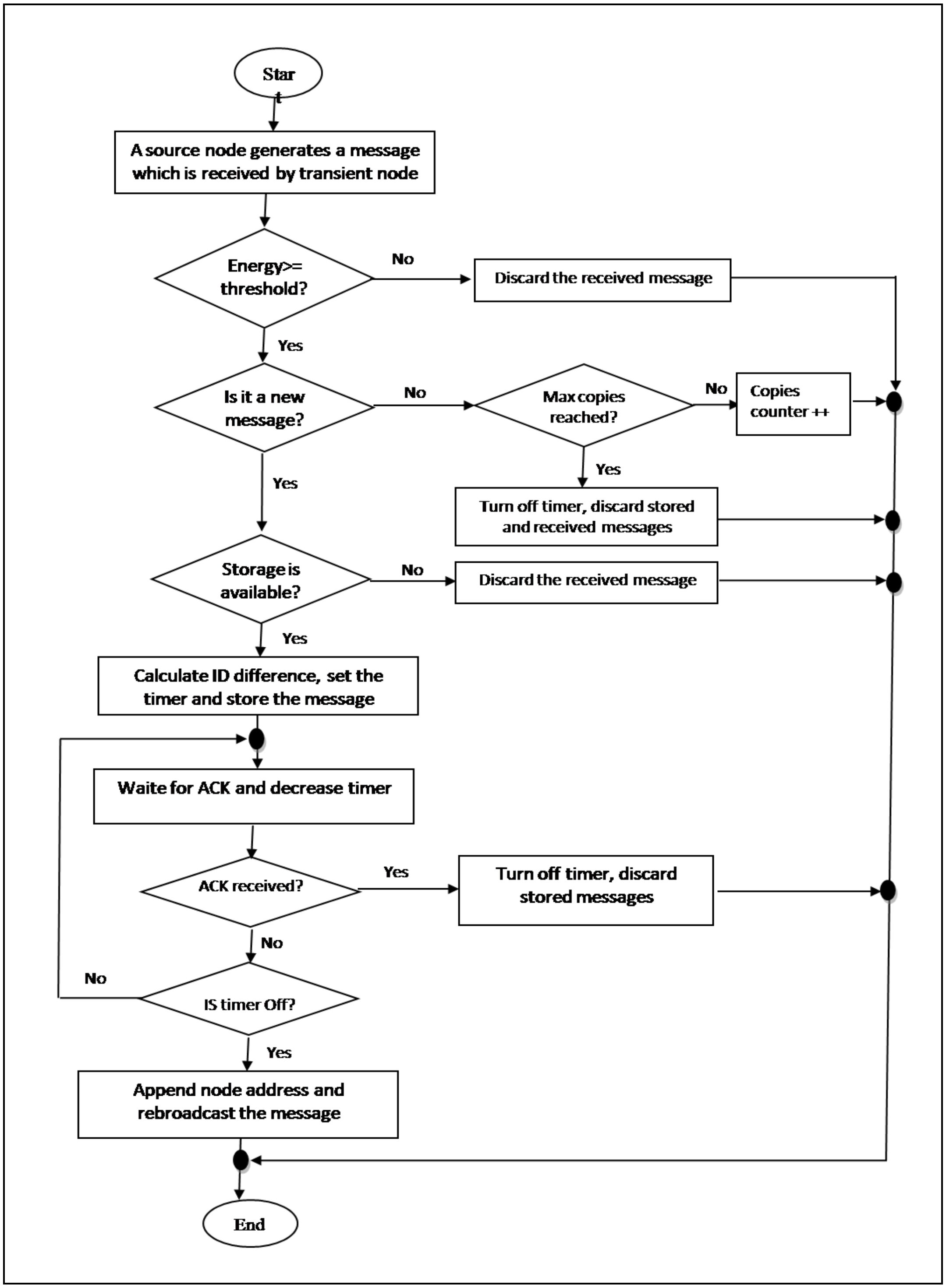


Fig.2.2: CEERA flowchart

CHAPTER 3: CEERA Performance & Developed Software

3. Introduction

The objective of this chapter is to present the simulation model used to evaluate the performance of CEERA. Model operations, assumptions, input and output are presented in details:

3.0. Simulation Model:

Simulation is a technique in which one system is substitute for another system that it resembles in certain important aspects. When building a simulation model of a real-life system under investigation, one does not simulate the whole system but simulates those related sub-systems. Simulation model of a system is a representation of it. It involves analyzing the entire system from a performance standpoint. For most analysis it's not necessary to account for all different aspects of the system. A model simplifies the system to sufficiently detailed level to permit valid conclusions to be drawn on that system. Dependent on the objectives being pursued by the analyst, a given system can be represented by several models [17].

The simulation model can be classified into static and dynamic models. A static simulation model is a representation of a model at a particular time, or any one that may be used to represent a system in which time simply plays no role. A dynamic model represents a system as it evolves over time.

Another way to classify a simulation model is by deterministic or stochastic simulation model. If a simulation model does not contain any probabilistic (i.e. random) components, it's called deterministic. It is a stochastic if system the system response may assumes a range of values for a given initial state and input.

A simulation model maybe continues or discrete model. A continues model is one in which the state of the state variable change continuously with time. The model is characterized by smooth changes in the system state. A discrete model is one which state variables assume a discrete set of values. Such model is characterized by discrete changes in system state.

Since simulation is the dynamic portrayal of the state of a system over time, an automatic internal clock must drive a simulation model. Two approaches have been suggested for advancing the simulation clock: next event time advance and fixed-increment time advance. With the next event time-advance approach, the simulation clock is initialized to zero the times of occurrence of future of future events are determined. The simulation is then advanced to the time of occurrence of future events, at which point the state of the system updated to account the for the fact that an event has occurred, and our knowledge of times of occurrence of future events is also updated. Then the simulation clock is advanced to the time of new most imminent event, the state of the system is updated and the future events times is determined, etc. In the second approach, the simulation clock is advanced one “tick” of Δt. this approach is simple but inefficient since some actions most take place at each clock click “tick” [17,19]. The simulation we consider in this thesis is will be discrete, dynamic and stochastic model. The timing mechanism used is next-event time advanced mechanism.

3.1. Program Flowchart

The block diagram for the simulation program is shown in Fig. 4. Many events are included and scheduled in event list and multiple many routines and steps. First, initialization routine initializes network topology, network and simulation parameters. It also initializes event list by scheduling the first message arrival event. The execution of message arrival event schedules the message generation event and next message arrival at exponential *t* time for a randomly selected node *i*. Then the program apply CEERA algorithm by scheduling and executing suitable events. The state of system is updated within events execution. At the end of simulation, the simulation program reports the performance results.

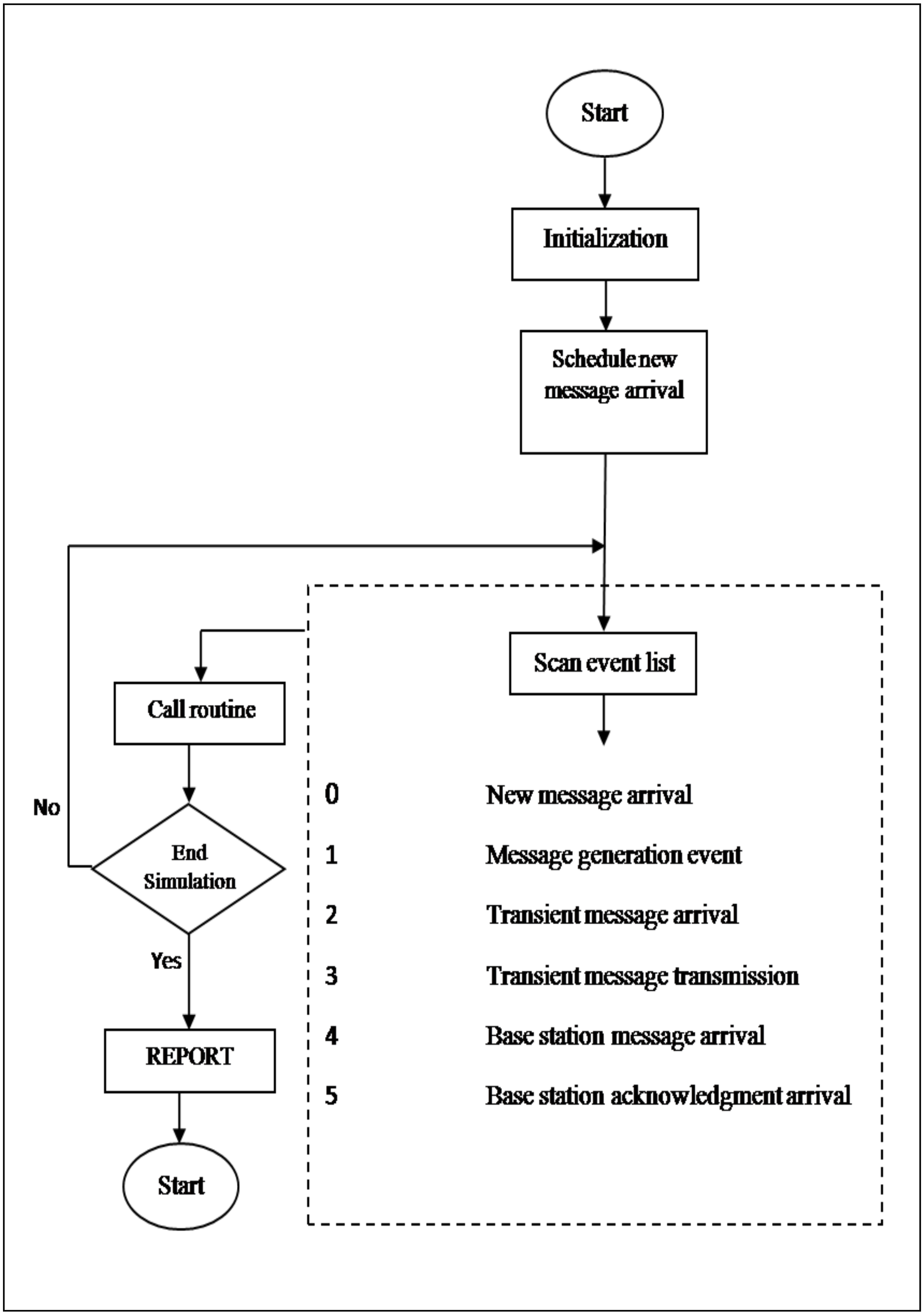


Fig.3.1: Main simulation program flowchart

3.2. Model Operations:

In simulation model, several major events and modules are considered. Events are shown in Fig. 4 and listed below. The abstracted description of each event is provided in section 3.5.

1. Initialization module
2. New message arrival event
3. Message generation event
4. Transient message arrival event
5. Transient message transmission event
6. Base station message arrival event
7. Base station acknowledgment arrival event

3.3. Simulation Variables:

All the relevant variables of a system under study are organized into three groups: Input, controllable, and output variables. The distinction between these categories mainly depends on the scope of the study.

3.3.1. Controllable Variables:

Controllable variables are manipulated to come up with a solution. We consider the following controllable simulation variables: Simulation clock, Events type and time, Simulation End, binding, receiving and acknowledged lists.

3.3.2. Input Variables:

Input variables are variables that are considered as given and are not to be manipulated during the simulation. These parameters are: Network diminutions, No. of nodes in the network, Base station location, Nodes location, Initial energy per Node, Energy threshold E_{min}, Scalar factor, D_{max}, Mean inter-arrival time, Maximum buffer size per node, Duplication factor, Processing Delay, Propagation delay, Beta, E_{elec}, E_{amp}, E_{calc}, Maximum no. of simulation messages.

3.3.3. Output Variables:

Set of variables calculated by the simulation program and represents the simulation results. These variables are used to measure the performance of the routing algorithm. Basically, we consider No. of successful transmissions (Throughput), Total Network Delay, Delay time jitter, Total Energy dissipations/initial energy, congestion, No. of duplicated arrival, frequency of memory blocking, hop count, No. of blocked transmissions for memory shortage, No. of blocked transmission for energy shortage, No. of died nodes, Beta of nodes to die lifetime, First Node to die lifetime (FND), Half of the Nodes Alive (HNA) lifetime, and Last Node to Die lifetime (LND) as the basic simulation output.

3.4. Development Tools:

Simulation model presented in this project was written in C++ programming language. Several development tools were used. We used Ns-2 to generate random network topology. We also used *true* random numbers generators software [19] offered by RANDOM.ORG. The generated randomness comes from atmospheric noise, which for many purposes is better than the pseudo-random number algorithms typically used in computer programs. This can be used by scientists for random sampling and as input to modeling and simulation applications.

3.5. Events Specification

This section provides a high level description for each of simulation events and the interaction between these events.

Event Name: Message Arrival
ID: 0
Triggers: Initialization routine
Events triggered by this Event: 1) New message arrival 2) Message Generation
Actions: 1) Select a live node n randomly to generate a MSG 2) Specify Message fields : MSG_ID, Source, Time 3) Update counters (Message/Network++ Message/Node++) 4) Schedule message generation event at time t= current time+ processing delay 5) If not end of simulation, generate a uniform random number *u* schedule a new MSG arrival at time t= current+ (inter-arrival time* - len (u))

Event Name: Message Generation
ID: 1
Triggers: Message Arrival
Events triggered by this Event:

1) Arrival of a MSG at BS 2) Transient MSG arrival
Actions: Generate and Broadcasting the message by: 1) Find Energy dissipated/transmission: $E_{Tx}(k, d) = E_{elec} * k + \varepsilon_{amp} * k * Dmax$ 2) Update energy and node state 3) Find nodes who should receives the MSG (RL) 4) If the BS is in RL, Schedule an arrival of a MSG at BS event 5) For every i node in RL, schedule transient MSG arrival event

Event Name: Transient MSG arrival
ID: 2
Triggers: Message Generation, Transient MSG Transmission
Events triggered by this Event: 1) Transient MSG Transmission
Actions: 1) Check energy, Find Energy dissipated/receiving $E_{Rx}(k) = E_{elec} * k$ 2) Update energy 3) Checking MSG for earlier acknowledgment and MSG

source

4) Checking if MSG has been received earlier for number of times equals duplicate factor.

   1. Cancel the scheduled transmission event in event's list for this MSG by this node.

   2. Freeing memory that was locked by this MSG

   3. Calculate dissipated energy for computation

   4. Update energy value

5) If received earlier but less duplicate factor

   Duplicatedcopies++

6) If first time received

   1. Check energy

   2. Checking Buffer size, (if not, discard MSG)else BufferSize++

   3. Store MSG-ID in received list

   4. Store MSG in current binding list

   5. Calculate timer value

   6. Turn timer on by scheduling a transmission event in event list at time t equals= current time+ timer value

Event Name: Transient MSG Transmission

ID: 3
Triggers: Transient MSG arrival,
Events triggered by this Event: 1) Transient MSG arrival, arrival of a MSG at BS
Actions: 1) Calculate dissipated energy for timer activity 2) Update energy value 3) Check energy threshold (if less, discard the packet) Else (Broadcasting the message by: 1. Find Energy dissipated/transmission: $E_{Tx}(k, d) = E_{elec} * k + \varepsilon_{amp} * k * Dmax$ 2. Update energy 3. Find nodes who should receives the MSG (RL) Schedule a transient MSG arrival event for every i node in RL 4. If the BS is in accessible(Schedule an arrival of a MSG at BS event) 5. Free memory 6. Delete MSG from current binding list

Event Name: BS MSG Arrival
ID: 4
Triggers: Transient MSG Transmission, MSG generation event

Events triggered by this Event: 1) BS Acknowledgment arrival
Actions: 1) If MSG first time received : 1. Add MSG to received list 2. Update counters (NodeThroughput ++, NetworkThroughput++,TotalNetworkDelay, DelayPerNode) 3. Acknowledge MSG receiving by scheduling a BS Acknowledgment Arrival event to all nodes 4. Add MSG to Acknowledged list 2) If MSG is received earlier, Duplicated-arrival++

Event Name: BS Acknowledgement Arrival
ID: 5
Triggers: BS Acknowledgement Arrival
Events triggered by this Event: none
Actions: 1) If MSG is in binding list: 1. Cancel the Transmission event 2. Delete MSG from current binding list 3. Freeing memory that was locked by this MSG (BufferSize--) 4. Calculate dissipated energy for timer activities. 5. Update energy

```
2) If MSG origin (Calculate Delay and Update
   statistical counters: Nodedelay, NodeThroughput)
```

CHAPTER 4: Simulation Results

4.0. Introduction:

This chapter shows performance results of simulating CEERA. It also shows the impact of changing variety of input factors over performance measures. The performance of CEERA is measured and compared with flooding, minimum transmission energy (MTE) and direct communication protocol [11,20].

4.1. Experiment design:

Exercising the model is an important phase in any simulation experiments. Input factors have presumably some effect on the output performance measures. It is often of interest to estimate how a change in an input factor affects an output performance measure and how sensitive an output is to a change in an input. To do this effectively, it is needed to plan and design the experiment before doing the runs [21]. We aimed to study the impact of changes in scalar factor, D_{max}, buffer size, duplication factor over performance measures. The effect of each factor will be evaluated under different locations for the base station. These measurements are: throughput, delay, DTJ, memory occupation per node, energy dissipation per initial energy, no. of died nodes, FND, BND, HNA, LND, Hop count, congestion/overhead, and duplicated arrivals.

4.2. Varying Base station location:

The efficiency of CEERA considered four different locations for base station. These locations are listed below and shown in Fig. 5:

1) Origin O, at (0,0): in this case, base station is placed at away location. It's not accessible by most of the sensor nodes. Also, node density around BS is very low. The throughput of system is basically depends on the live time of these key nodes that connect BS with the rest of network.

2) Center, C, at (25,25): BS is at focal point of deployment area. In this case, BS is accessible by most no. of nodes and no. of nodes around BS is good.

3) Two random locations:

- A at (22.97, 31.79): BS is randomly placed approximately at middle of deployment area. Nodes density around BS is highest.
- B at (8.92, 4.57): BS is placed at random of deployment area. Nodes density around BS is low.

Location			No. of BS's reachable nodes for a given Dmax				
BS	X	Y	Dmax=5	Dmax=10	Dmax=15	Dmax=20	Dmax=25
O	0	0	2	7	10	10	15
C	25	25	4	18	32	60	76
A	22.97	31.8	8	20	35	54	75
B	8.92	4.57	3	9	12	17	28

Table.4.1: density of nodes around base station.

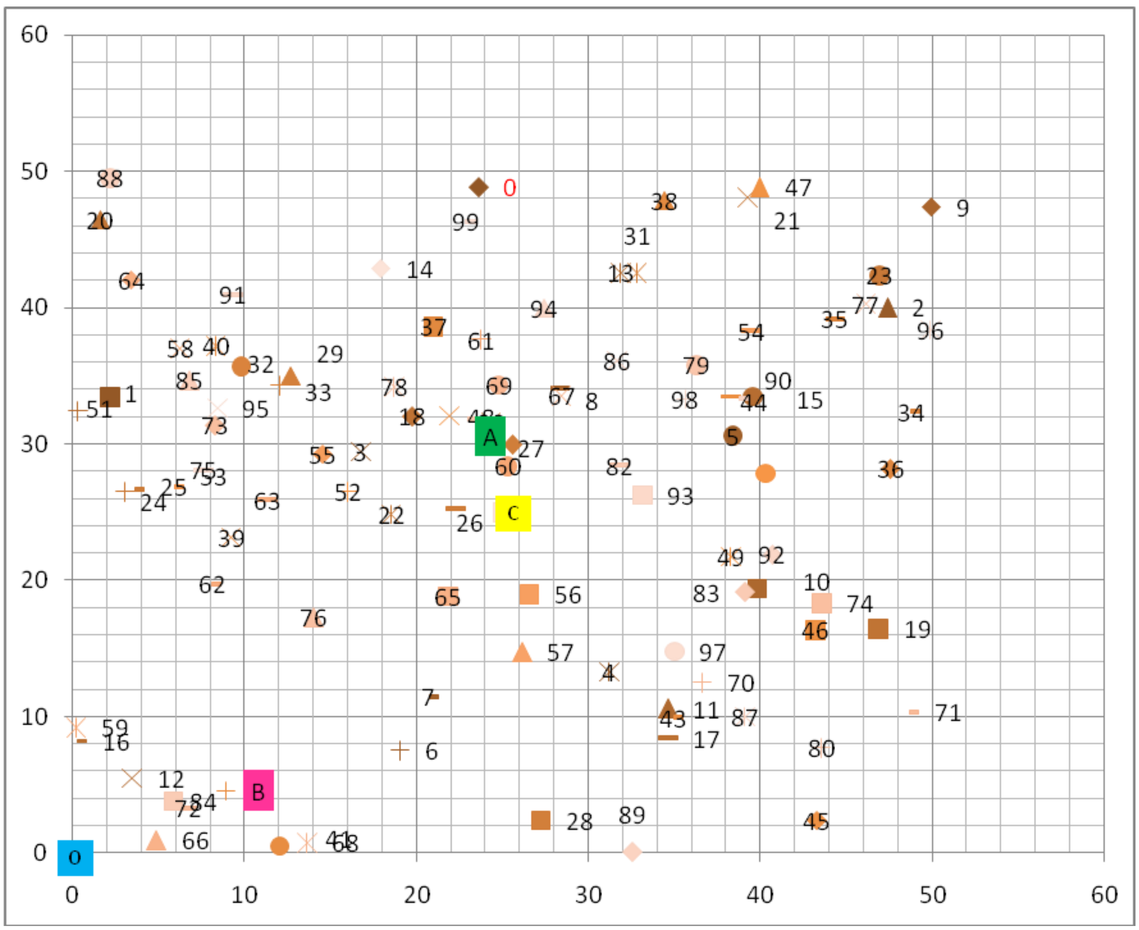


Fig. 4.1: Multiple locations for BS.

4.3. Simulation Run :

In the design of the simulation experiment, we try to make inputs realistic and suitable for both the specification sensor nodes and network functionality. Basically, we run simulation for inputs of 40.000 Simulation messages and initial energy per node equals 2.26 e+7nJ/bit. We assume node ability to send and receive 1000 message. However, this initial energy is very low compared to the expected amount of energy required to transmit 40.000 also the dissipation of energy resulted by receiving these message. For this reason, it's expected that most of nodes die, if not all, during the simulation run. We will compare throughput to energy dissipation/initial energy and nodes lifetime. We started with reasonable buffer size per node to store 20 messages. The transmission distance, D_{max}, that does ensure network connectivity is 10 m. The inter-arrival time is environment-dependent. It indicates traffic load carried by nodes in network. We assumed mean inter-arrival time is 0.5 s. Duplicate factor is also problem and environment dependent. Basically its value equals 5. We examine a variety of Scalar factor values. In most runs its value equals 0.5. At the beginning of next sections from 4.3.1 to 4.3.6 we specify simulation input exactly.

4.3.1. Varying Scalar Factor:

The value of scalar factor has a direct influence on end-to-end-delay since it controls the timer value (waiting period). Also, has a positive effect on energy dissipation. It delays the transmission of the received message for certain amount of time, within that time a message might be received and acknowledged by the BS which save energy and minimizes the overhead caused by this re-transmission. It also saves energy that would be dissipated if the message transmitted and received by neighbor nodes. In the other hand it has a negative effect on memory occupation as the message stored for long time upon ACK receiving or timer is turned off.

Given simulation's input of 40.000 Simulation messages, 20 buffer size, mean inter-arrival time = 0.5, initial energy per node = 101e+7, Duplicate factor=5, and Dmax= 10

m. We extract results scalar values: 0, 0.1, 0.5, 1, 2, 5, and 10. Results are shown in next four tables and figures from a.1 to a.13.

4.3.1.1. Scalar vs. throughput, Delay, and DTJ:

Figure a.1 shows highest throughput for BS at (A) which has the highest nodes density around BS. Non-zero scalars have a positive impact over throughput for all BS locations compared to zero scalar. The ongoing growth of scalar value does not deduce higher throughput. It’s important when determining scalar value to balance between the tendencies to save retransmission energy, preventing overhead, required storage and delay caused by scalar factor. The best scalar value is between 0.1, 0.5 and 1 as it has the best throughput, increase storage sharing and also less delay than other values. Upper scalar values (2, 5, and 10) reduces throughput, minimizes memory sharing, and has a very high delay. It causes large message loss due to the memory shortage as a result of the long duration of blocking but not frequency of blocking. As expected, scalar factor is directly proportional to delay and delay time jitter as shown in figure a.2 and a.3. Values of delay and DTJ for scalar 0.1 and 0.5 is reasonable compared to the improvement in network throughput.

4.3.1.2. Scalar vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

As explained in section 4.3, it’s expected that most nodes dissipate all of its energy and die. But what we focused on is that: 1) time nodes die and 2) the amount of message successfully routed by the mean of this energy. Figure a.4 shows that around 100% of energy is dissipated in all four networks. The late time a node die reflects better energy efficiency and higher network functionality. It’s important to clarify that zero value for FND, BND, HND, and LND indicates that this condition never satisfied; i.e. if LND=0 indicated that there is at least one node remains alive until the end of simulation.

Figures a.5, a.6, a.7, a.8, and a.9 show that direct relationship between network live time and scalar values, as expected. It’s also, shown that network “C” and “A” has higher measures than other networks in all FND, BND, HND, and LND. Network “C” shows better energy utilization than “A” die time in figures a.7 and a.9 for some scalar values due to changes in nodes density around BS at each case. However, this difference still small and range from zero and few seconds.

4.3.1.3. Scalar vs. storage occupation, duplicated arrival and congestion:

Figures a.10, a.11 and a.12 show poor values in congestion, duplicated arrival and memory occupation for zero scalar value. Escalating scalar value improves all these measures. Figure a.12 shows how scalar prevents duplicated message arrival to the BS. There is a similarity in the performance between network “A” and “C”; and between “O” and “B” since they have similar conditions.

4.3.1.4. Scalar vs. Hop count:

Figure a.13 shows that hop count is optimized with non-zero scalar value. The reason is that the amount of energy dissipated by direct retransmission affects most of the nodes that connects network segment in early time. As a result for this, later messages follow a long path to reach the base station.

4.3.1.5. Discussion on Scalar Factor:

The previous section shows how scalar factor affects the different performance measures. However, this effect is not linearly related in all cases. There is an optimal value for scalar factor that preserve energy and memory, with minimum delay. Scalar should be within that threshold, threshold value depends on the state of network. To obtain this, we concluded with the following proportional between scalar factor and other measures:

- **Scalar factor (SF) and Throughput (Thro.):**

$SF \propto Throughp$…………………….. for SF> 0 and SF< $SF_{Threshold}$

$SF \propto \frac{1}{Throug}$…………………….. for SF> $SF_{Threshold}$ and SF<

- **Scalar factor (SF), Delay (D) and DTJ:**

$SF \propto$ …………………... for SF> 0 and SF<

$SF \propto$ …………………... for SF> 0 and SF<

- **Scalar factor (SF) and node life time (L.):**

  $SF \propto$ …………………... for SF> 0 and SF< $SF_{Threshold}$

- **Scalar factor (SF) and Storage Occupation (SO):**

  $SF \propto$ …………………... for SF> 0 and SF< $SF_{Threshold}$

- **Scalar factor (SF) and Congestion (Con):**

  $SF \propto$ …………………... for SF> 0 and SF<

- **Scalar factor (SF) and Duplicated arrival (Dup):**

  $SF \propto$ …………………... for SF> 0 and SF<

Base station Location	Center						
Scalar Factor Value	0	0.1	0.5	1	2	5	10
Throughput	4703	15151	15147	15149	15115	14911	14795
Delay	0.00023	2.81936	14.0662	28.2405	60.2757	179.097	338.65
Delay Time Jitter	0.000117	3.82722	19.1394	38.3072	82.3954	275.467	529.082
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	98	97	99	99	99	99	99
FND	1359.84	2086.26	2094.39	2109.57	2168.79	2824.1	4198.88
BND	1739.01	4066.32	4098.71	4116.72	4284.27	4938.21	7410.86
HND	2373.44	5837.21	5708.44	5743.14	5601.44	7090.23	10597.2
LND	0	0	10208.6	9992.52	9893.33	11002.3	14042.4
Congestion	68179	9448	9468	9771	11003	10768	7295
Duplicated Arrival	11007	21	24	22	34	88	80
Hop count /message	10	2	2	2	2	2	2
Frequency Memory Occupation	100907	235434	235809	236096	224432	164275	120261

Table a.1: Performance results of varying SF for network C.

Base station Location	Origin						
Scalar Factor Value	0	0.1	0.5	1	2	5	10
Throughput	2849	5677	5677	5691	5700	5709	5709
Delay	0.000271	2.86646	16.8269	30.4126	55.9837	89.3468	123.502
Delay Time Jitter	0.000263	6.10709	37.7896	73.5379	142.987	247.34	360.592
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	98	99	99	99	99	99	99
FND	500.902	539.444	580.26	641.373	753.441	1279.06	1939.52
BND	527.825	673.546	779.148	1039.54	1096.15	1657.5	2814.23
HND	572.273	1196.28	1234.54	1495.2	2212.02	3044.87	6546.16
LND	0	6680.52	6300.32	6187.31	6885.03	9217.4	12440.3
Congestion	333092	82959	84943	82712	79559	65579	33306
Duplicated Arrival	2958	29	28	17	8	4	2
Hop count /message	6	2	2	2	2	1	1
Occupied Memory	105137	152899	140212	135639	128915	115978	102606

Table a.2: Performance results of varying SF for network O.

Base station Location	Random location A						
Scalar Factor Value	0	0.1	0.5	1	2	5	10
Throughput	4608	16829	16828	16824	16740	16600	16551
Delay	0.00	2.53	12.60	25.34	53.87	160.13	313.35
Delay Time Jitter	0.00	3.87	19.30	39.07	82.48	257.99	514.29
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	99	99	99	98	98	99	99
FND	968.92	2078.20	2094.26	2103.93	2070.06	2900.06	4264.82
BND	1468.19	3514.53	3565.95	3669.73	3654.33	4813.07	7980.79
HND	2231.52	5670.22	5642.78	5618.57	5568.54	7353.25	10671.40
LND	4045.43	10885.6	10722.4	0	0	12196	15107.7
Congestion	59851	6719	6699	7206	10333	9345	3974
Duplicated Arrival	12939	0	0	2	63	89	45
Hop count /message	13	2	2	2	2	2	2
Occupied Memory	102097	211501	212247	212165	195828	141750	107578

Table a.3: Performance results of varying SF for network A.

Base station Location	Random location B						
Scalar Factor Value	0	0.1	0.5	1	2	5	10
Throughput	4144	7567	7567	7568	7569	7564	7560
Delay	0.000192	1.84439	11.3982	19.1129	35.2904	54.8282	75.9604
Delay Time Jitter	0.000203	5.17728	32.673	61.2812	118.92	211.417	337.255
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	98	99	99	98	99	99	99
FND	510.505	531.274	575.872	630.981	749.733	1264	1938.52
BND	547.271	659.971	769.494	1068.08	1077.59	1621.04	2748.27
HND	595.085	1155.77	1213.54	1520.13	2116.63	3292.39	6758.26
LND	0	7768.55	7161.32	0	7750.04	10205	13478.8
Congestion	319063	82011	83002	84188	82118	67754	34298
Duplicated Arrival	3633	0	0	0	0	0	0
Hop count /message	5	1	1	1	1	1	1
Occupied Memory	101991	144951	131473	128186	120855	109025	97116

Table a.4: Performance results of varying SF for network B.

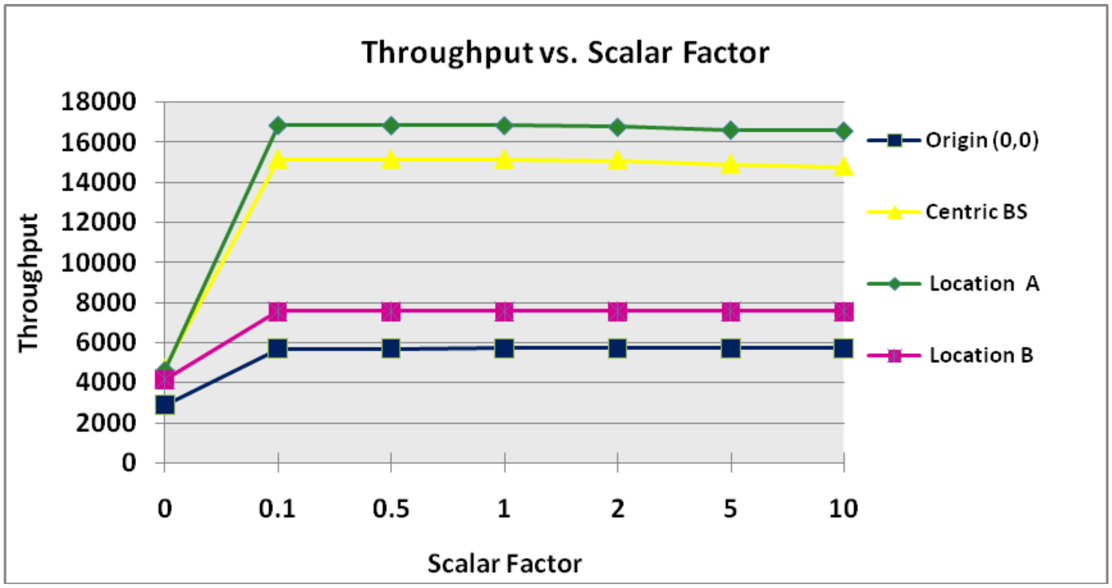


Fig. a.1: Throughput vs. Scalar Factor

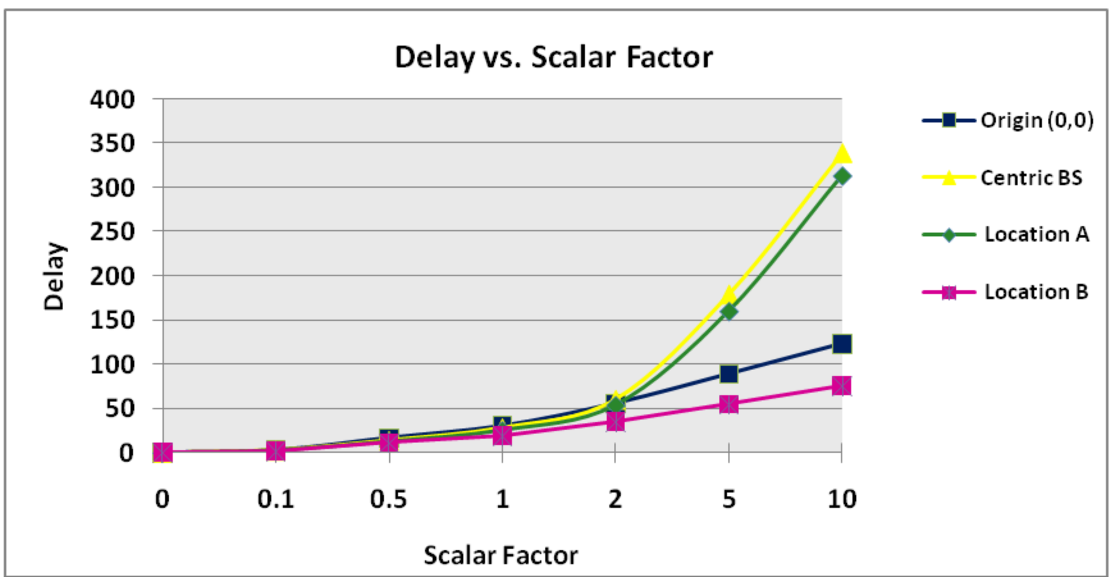


Fig. a.2: Delay vs. Scalar Factor

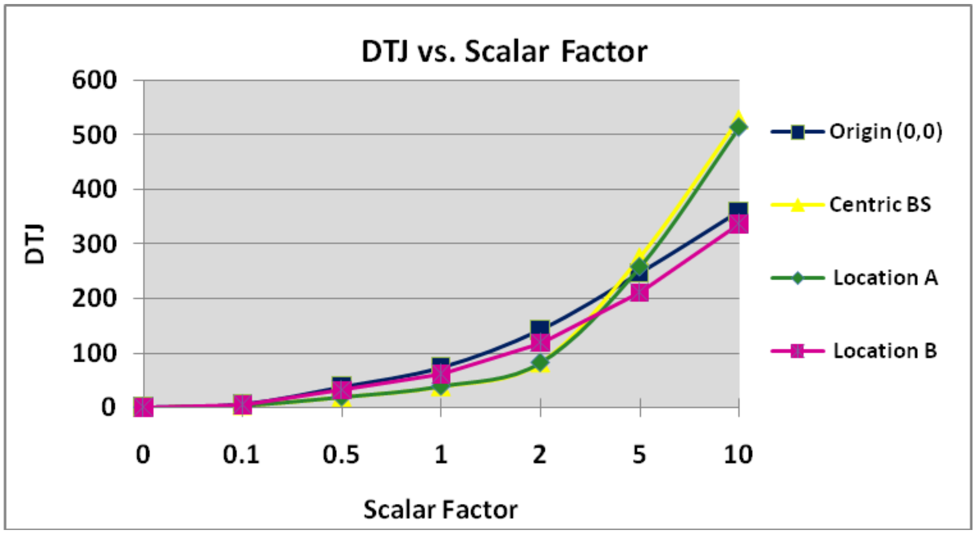


Fig. a.3: DTJ vs. Scalar Factor

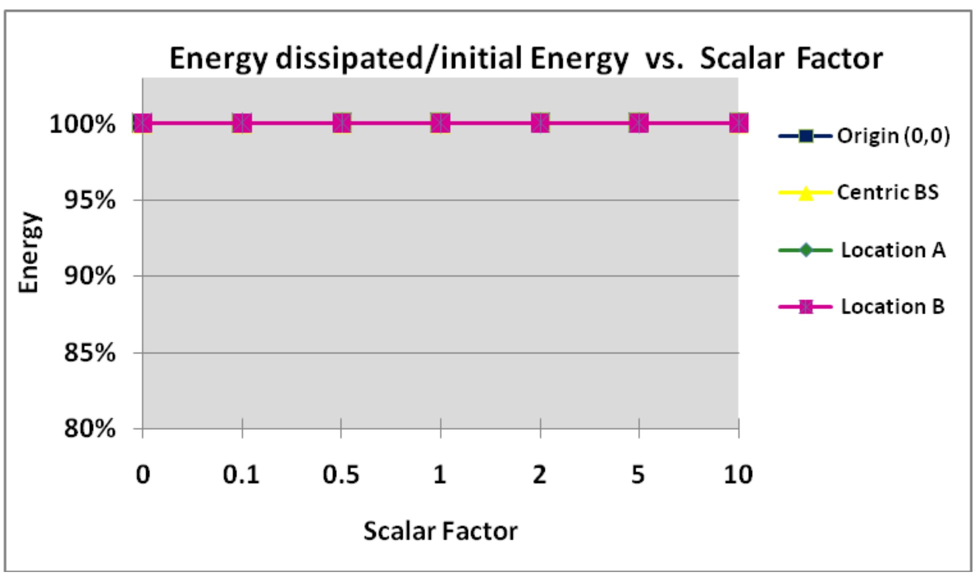


Fig. a.4: Energy dissipated/initial Energy vs. Scalar Factor

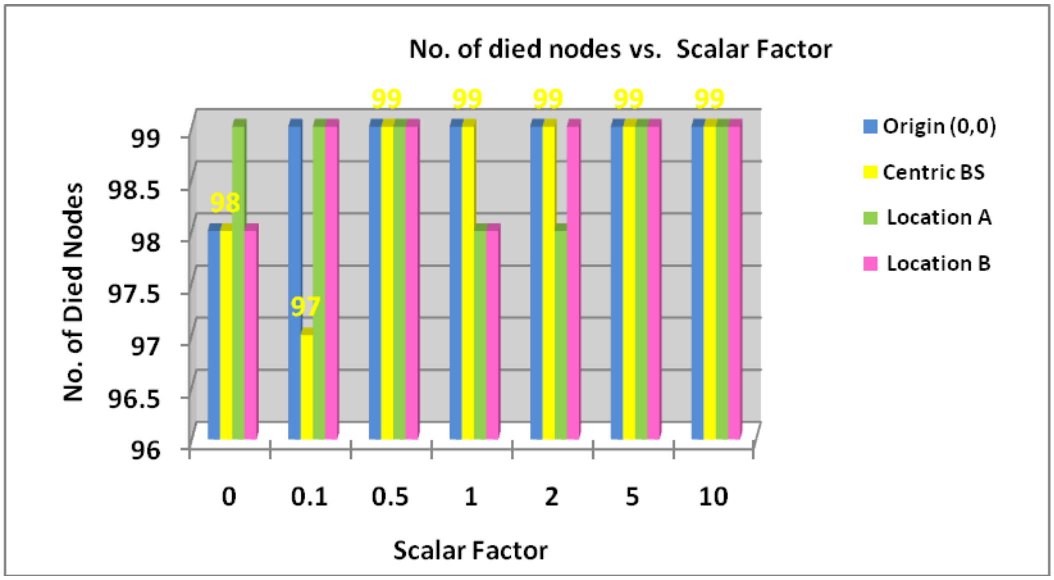


Fig. a.5: No. of died nodes vs. scalar factor

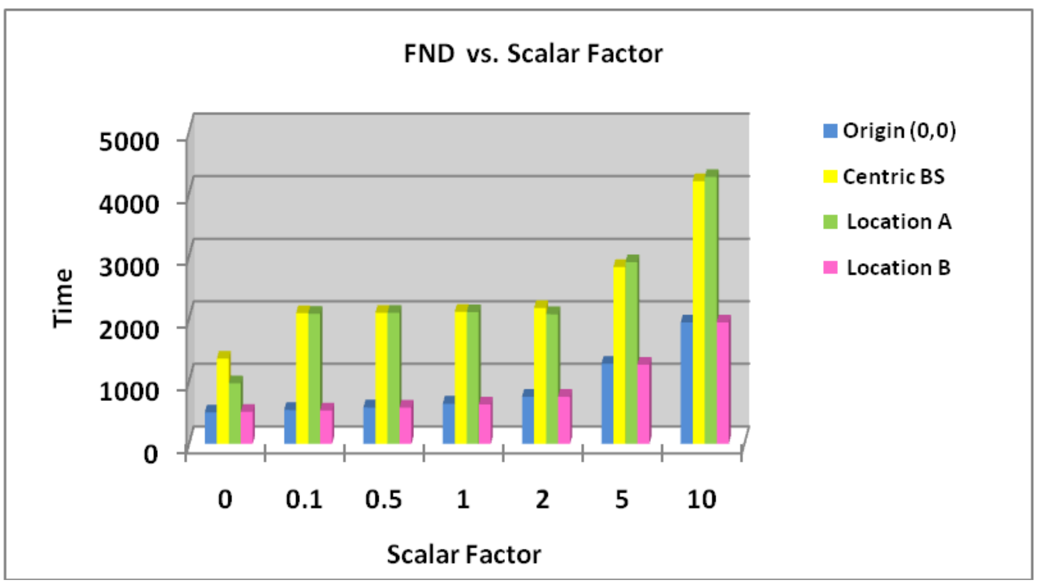


Fig. a.6: FND vs. Scalar Factor

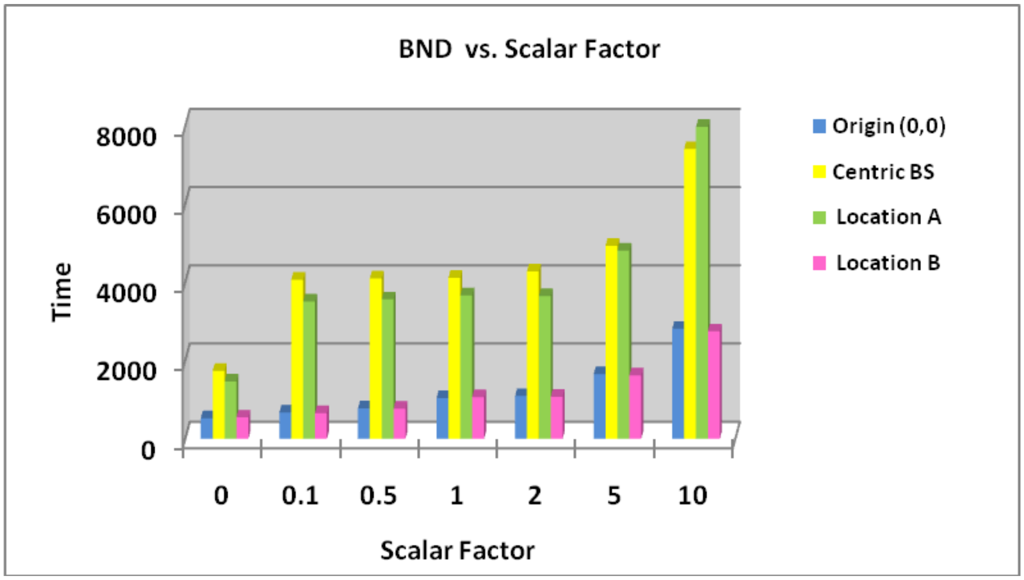


Fig. a.7: BND vs. Scalar Factor

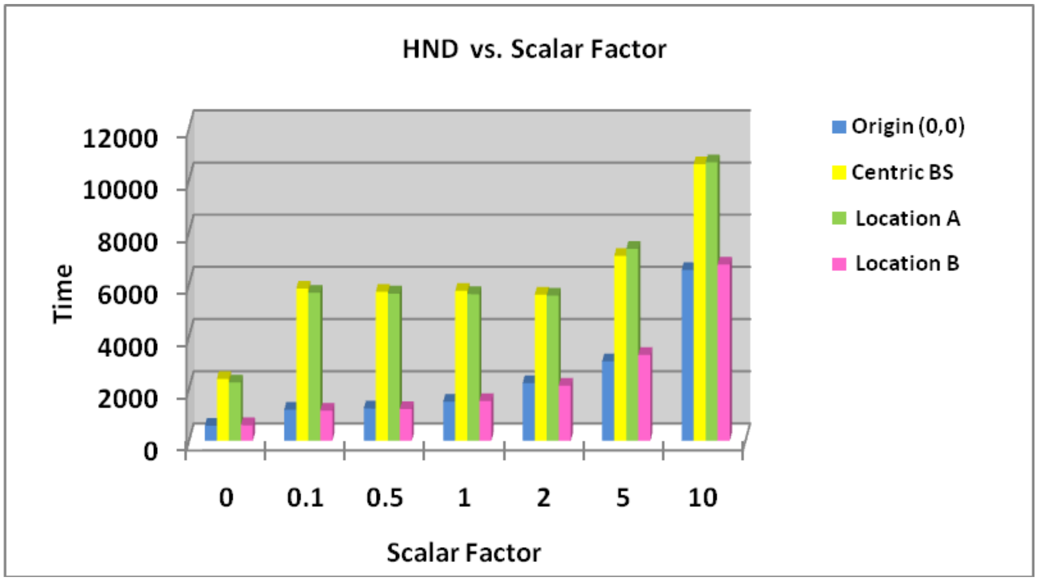


Fig. a.8: HND vs. Scalar Factor

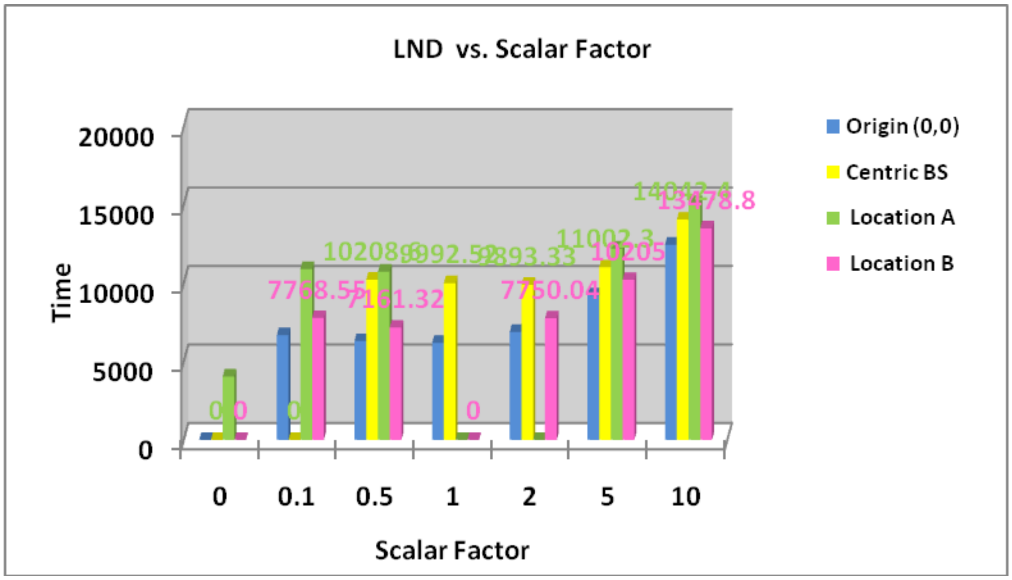


Fig. a.9: LND vs. Scalar Factor

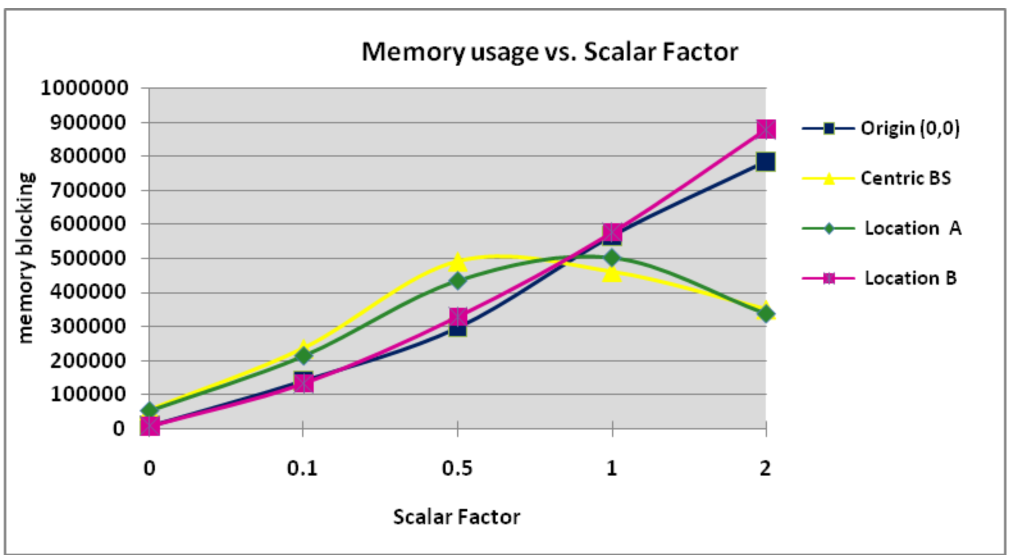


Fig. a.10: Memory usage vs. scalar factor

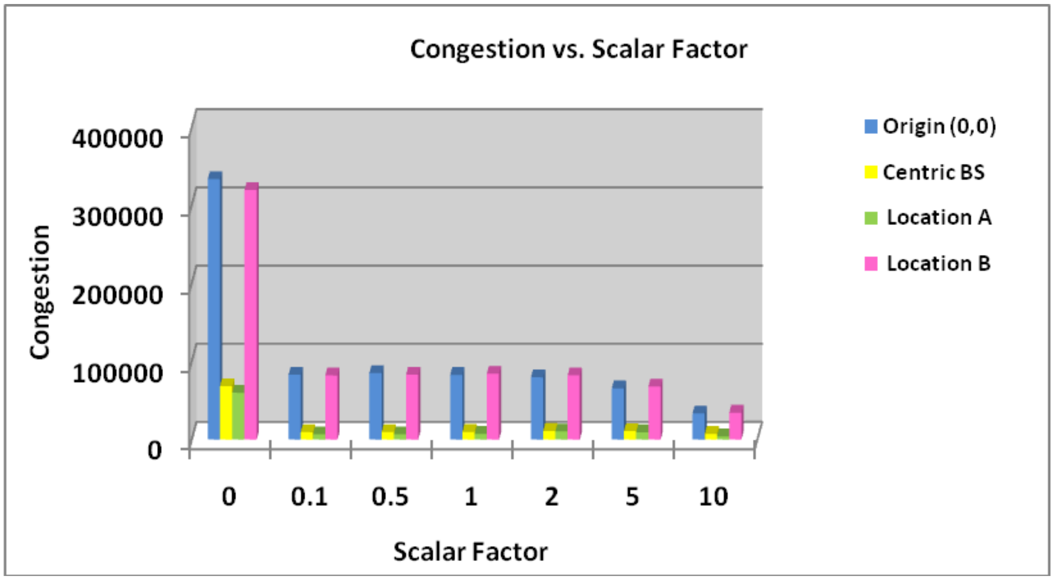


Fig. a.11: Congestion vs. scalar factor

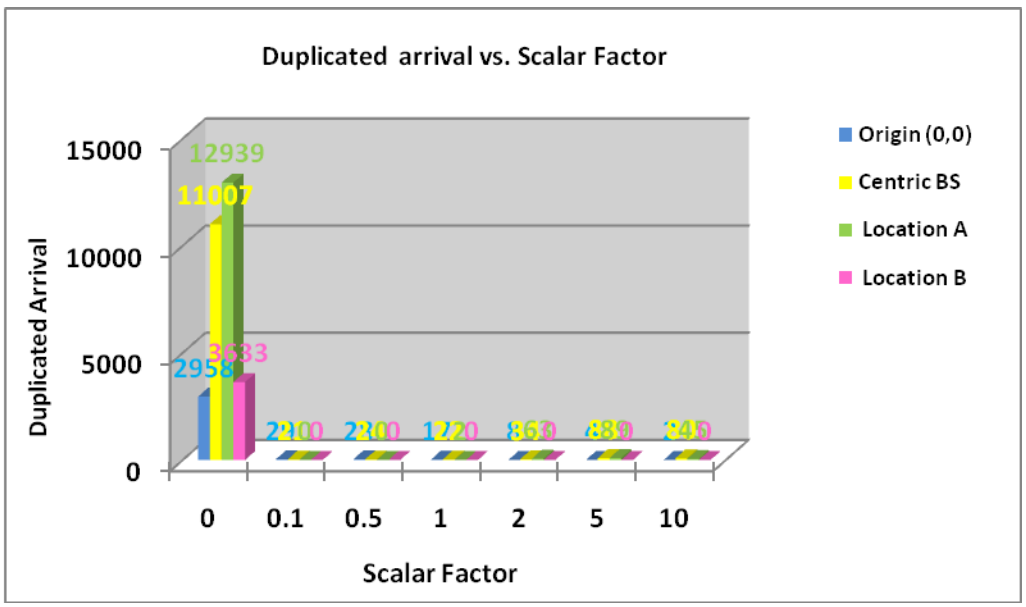


Fig. a.12: Duplicated arrival vs. scalar factor

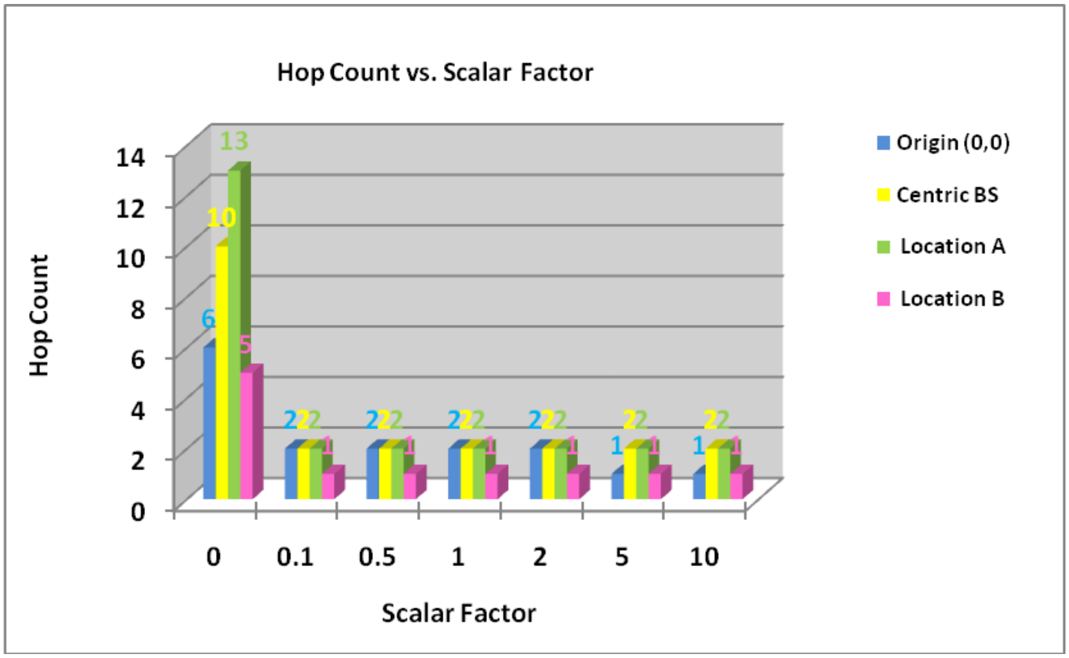


Fig. a.13: Hop count vs. scalar factor

4.3.2. Varying Dmax:

Given simulation input of 40.000 Simulation messages, 20 buffer size, mean inter-arrival time = 0.5, initial energy per node = 101e+7, Duplicate factor=5, and **Scalar factor=0.5**. We extract results for Dmax: 5, 10, 15, 20, and 25. Results are shown in next two tables and figures from b.1 to b.13.

4.3.2.1. Dmax vs. throughput, Delay, and DTJ:

Dmax is inversely proportional to delay and DTJ as shown in figure b.2 and b.3. However, it is not always proportional to throughput, when a message spans the network it cause a large amount of energy dissipation for receiving it message by large no. of nodes. Figure b.1 shows that throughput is reduces for Dmax =25 in network “A” and”C” for the explained reasons. It also show continues increasing in network “O” and “B” since the distant base station became accessible for other parts of network. For the same reason, figure b.3 shows a high DTJ value for Dmax=10 compared to dmax=5. The selection of Dmax is predetermined to network deployment and it’s restricted to the limitation of nodes resources and bandwidth.

4.3.2.2. Dmax vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

Figure b.3 show that initial energy is consumption for different Dmax values. The lowest dissipation indicated for Dmax=5 since 49% of nodes are inaccessible and isolated from other nodes and base station. The relationship between Dmax, FND, BND, HND, and LND depends on the layout network. Figure b.7 and b.8 show 20% and 50% of nodes of network “A” and “C” remains alive for Dmax=10.

4.3.2.3. Dmax vs. storage occupation, duplicated arrival and congestion:

Figures a.10, a.11 and a.12 show increasing in duplicated message arrival and high memory occupation compared to average no. of hops to reach BS. Duplication occurs since more than one node will have similar timer value and will dissipate the message in the same time and before receiving BS ACK.

4.3.2.4. Dmax vs. Hop count:

Figure a.13 shows escalating Dmax minimizes hop count in network “C” and “A”. In network “O” and “B” it increased since the base station can receive messages from the distant message in network through in multi-hop fashion.

4.3.2.5. Discussion on Dmax:

The previous section shows that the effect of Dmax is different from one network to another. Essentially, Dmax value is hardware dependant. Large Dmax dissipate large energy for receiving. In most cases, a very large Dmax is not supported routing is multi hop. For this, its value should be determined according to the location of base station, deployment area and deployment schema. We concluded with the following proportional between Dmax and other measures:

- **Dmax and Throughput (Thro.):**

  $\mathbf{Dmax} \propto \mathrm{Throughp}$………… for Dmax>0and SF< $\text{Dmax}_{\text{Threshold}}$

  $\text{Dmax} \propto \frac{1}{\text{Throug}}$………… for $\text{Dmax}_{\text{Threshold}}$ > 0 and Dmax <

- **Dmax and Delay:**

  $\mathbf{Dmax}$ …………………... for Dmax > 0 and Dmax <

- **Dmax and Congestion (Con):**

  $\text{Dmax} \propto$ …………………... for Dmax > 0 and Dmax <

- **Dmax and Duplicated arrival (Dup):**

  $\text{Dmax} \propto$ …………………... for Dmax > 0 and Dmax <

Base station Location	Random Location A					Centric (25,25)				
Dmax Value	5	10	15	20	25	5	10	15	20	25
Throughput	5786	16828	30010	39693	35000	2715	15147	26806	40000	40000
Delay	27.48	12.60	4.31	1.79	0.39	51.8135	14.0662	4.73355	0.986106	0.36365
Delay Time Jitter	38.18	19.30	8.97	4.12	0.77	41.9125	19.1394	7.90716	2.08838	0.796836
Total energy dissipated/initial Energy	55%	100%	100%	89%	55%	53%	100%	100%	78%	61%
No. of Died Nodes	47	99	99	63	6	50	99	99	27	5
FND	1988.61	2094.26	4014.72	7923.31	11934.50	2012.49	2094.39	4845.97	11084.6	14720.4
BND	2586.68	3565.95	7109.00	14080.60	0.00	2518.6	4098.71	8391.61	16375.1	0
HND	4794.16	5642.78	10375.70	17877.70	0.00	3628.22	5732.47	11328.8	0	0
LND	0	10722.4	15830.9	0	0	17800.7	10208.6	13872.7	0	0
Congestion	1337	6699	1970	110	0	1453	9468	1068	169	0
Duplicated Arrival	0	0	577	1580	1810	0	24	1217	1817	1860
hop count /message	3	2	1	1	1	5	2	2	1	1
Occupied Memory (Sum)	51307	212247	432700	500191	335598	53544	235809	492488	461485	350521

Table b.1: Performance results of varying Dmax for network C and network A.

Base station Location	Origin (0,0)					Random location B				
Dmax Value	5	10	15	20	25	5	10	15	20	25
Throughput	979	5677	8723	8579	13045	1743	7567	10625	15253	25016
Delay	16.9859	16.8269	8.13427	7.07226	4.75431	17.5681	11.3982	6.13383	4.25857	2.56386
Delay Time Jitter	11.6656	37.7896	14.007	8.99029	5.79872	15.7286	32.673	12.2799	7.12527	5.69703
Total energy dissipated/initial Energy	9%	100%	100%	100%	100%	9%	100%	100%	100%	100%
No. of Died Nodes	6	99	98	99	99	5	99	99	99	99
FND	7060.05	580.26	981.803	2019.74	1923.56	9661.72	575.872	1199.78	2028.52	2873.03
BND	7060.05	779.148	1378.63	2970.12	4173.6	9661.72	769.494	1805.57	3627.81	6069.6
HND	12185.7	1254.4	1882.66	3516.76	5107.06	15260.4	1233.72	2293.89	4464.72	7748.41
LND	0	6300.32	0	4921.76	6795.97	0	7161.32	6074.36	7874.53	12532
Congestion	0	84943	250387	268554	274631	0	83002	210080	187736	103172
Duplicated Arrival	0	28	17	230	568	0	0	28	182	884
hop count /message	2	2	2	3	2	2	1	2	2	1
Occupied Memory (Sum)	6823	140212	296463	565839	784630	5431	131473	327554	575144	878827

Table b.2: Performance results of varying Dmax for network O and network B.

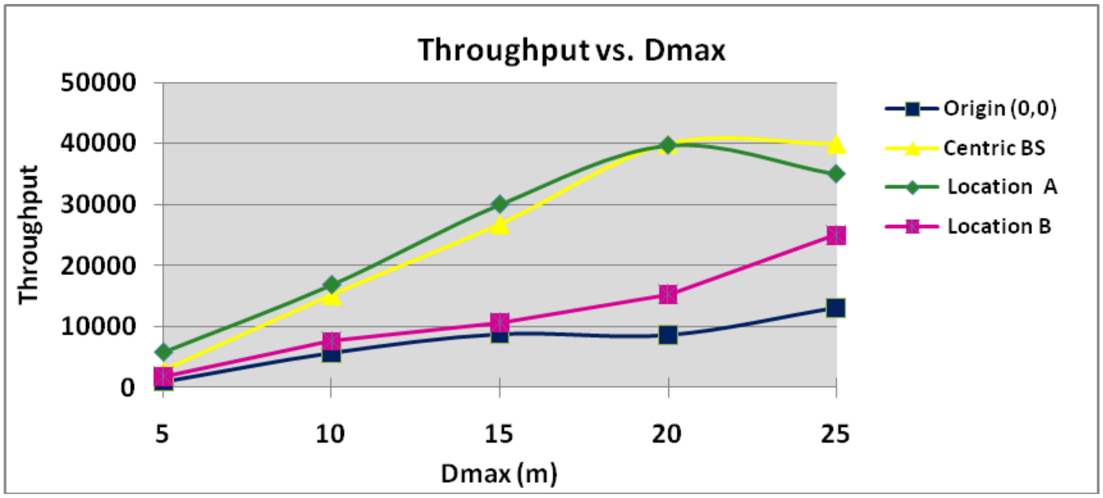


Fig. b.1: Throughput vs. Dmax

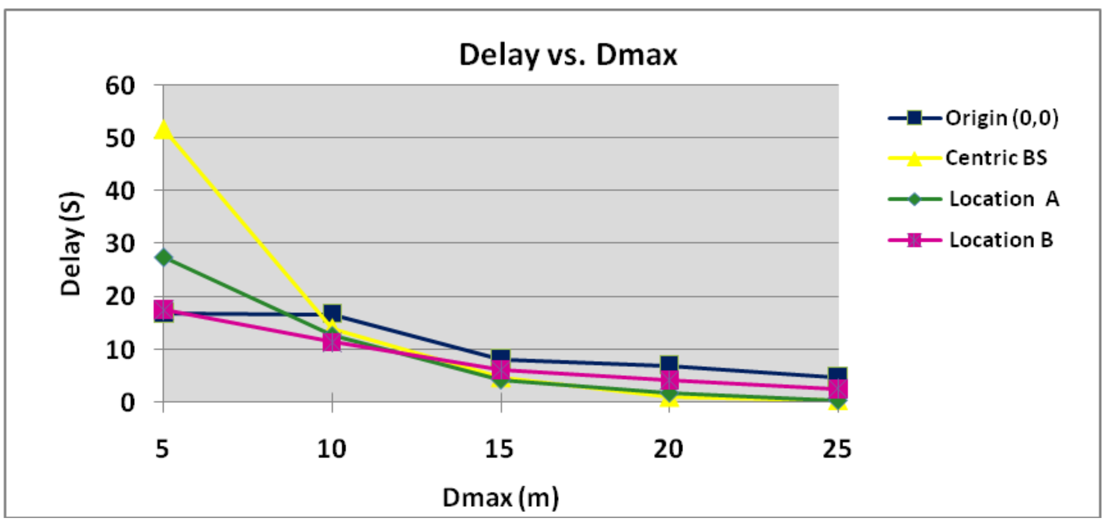


Fig. b.2: Delay vs. Dmax

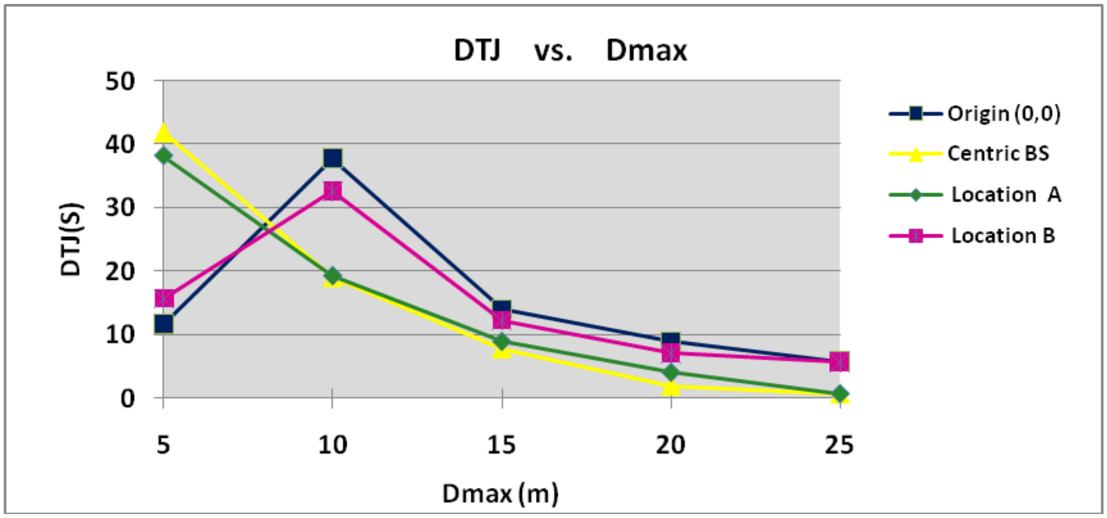


Fig. b.3: DTJ vs. Dmax

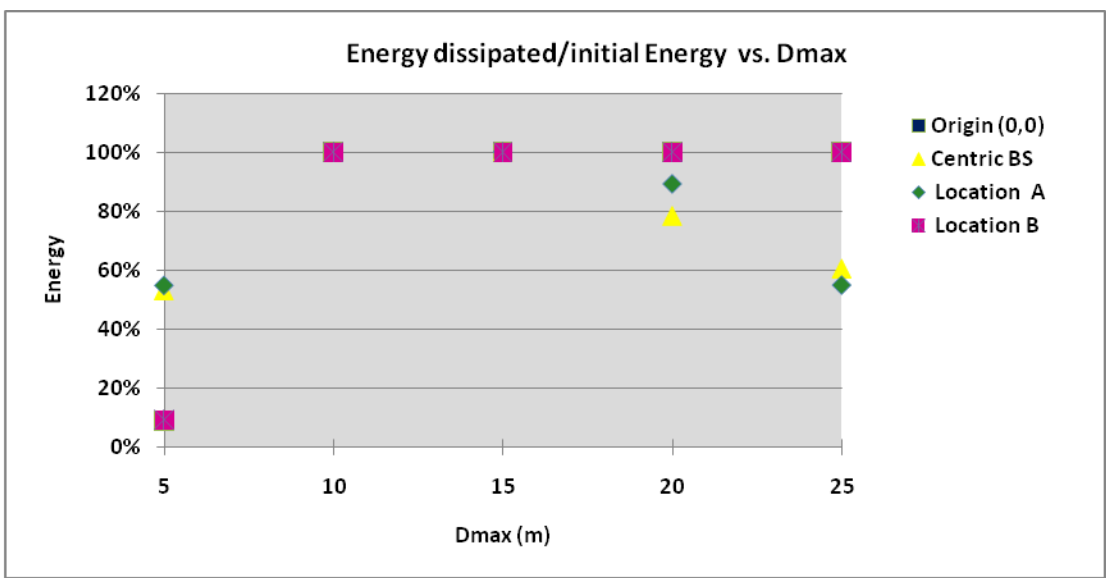


Fig. b.4: Energy dissipated/initial Energy vs. Dmax

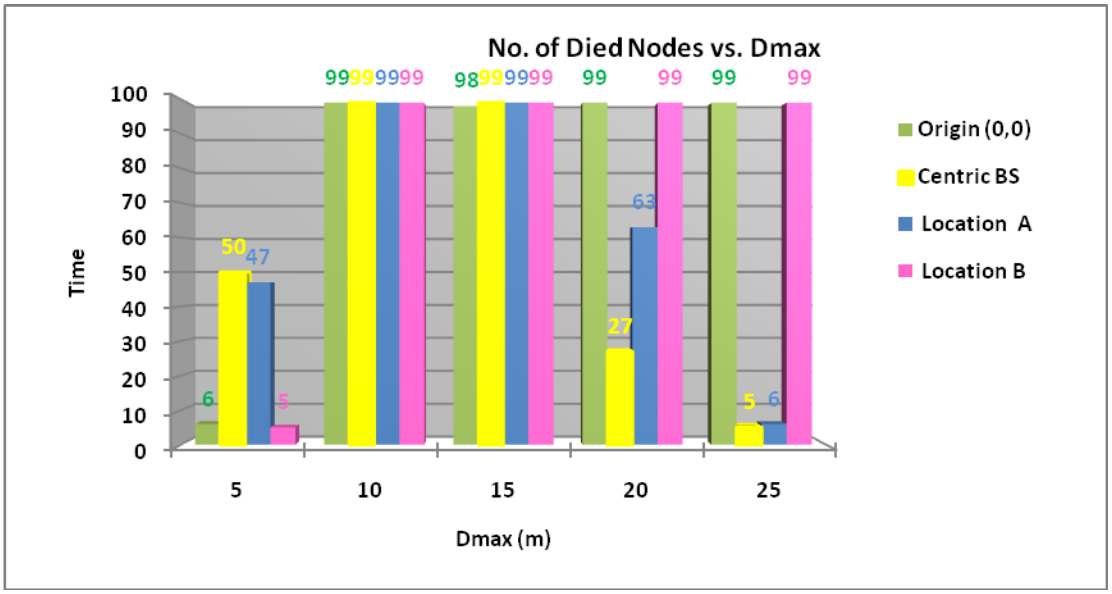


Fig. b.5: No. of died nodes vs. Dmax

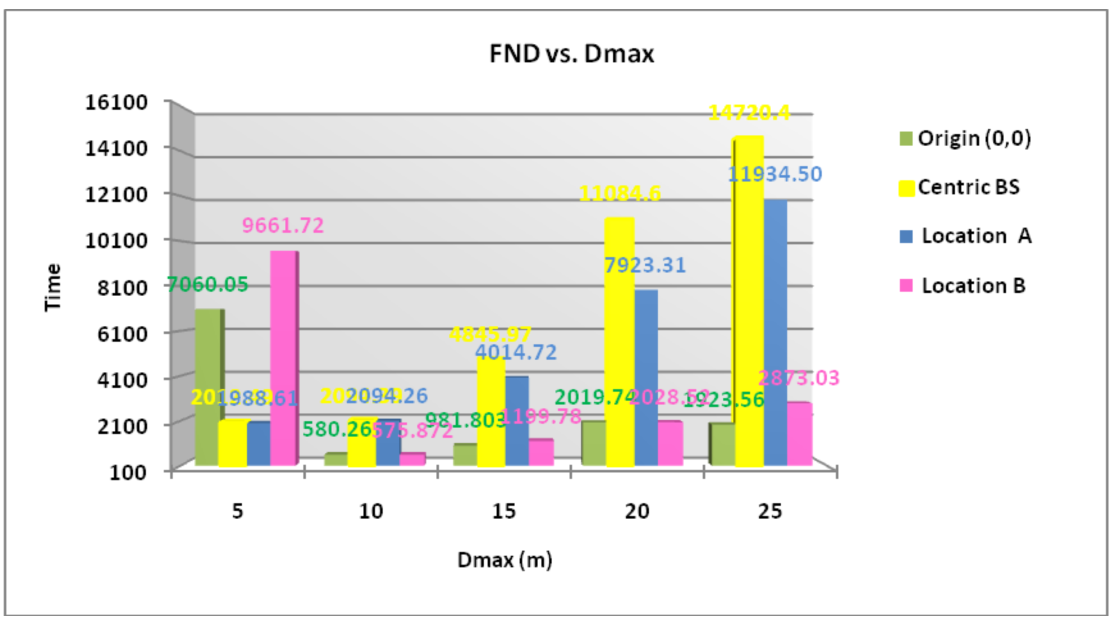


Fig. b.6: FND vs. Dmax

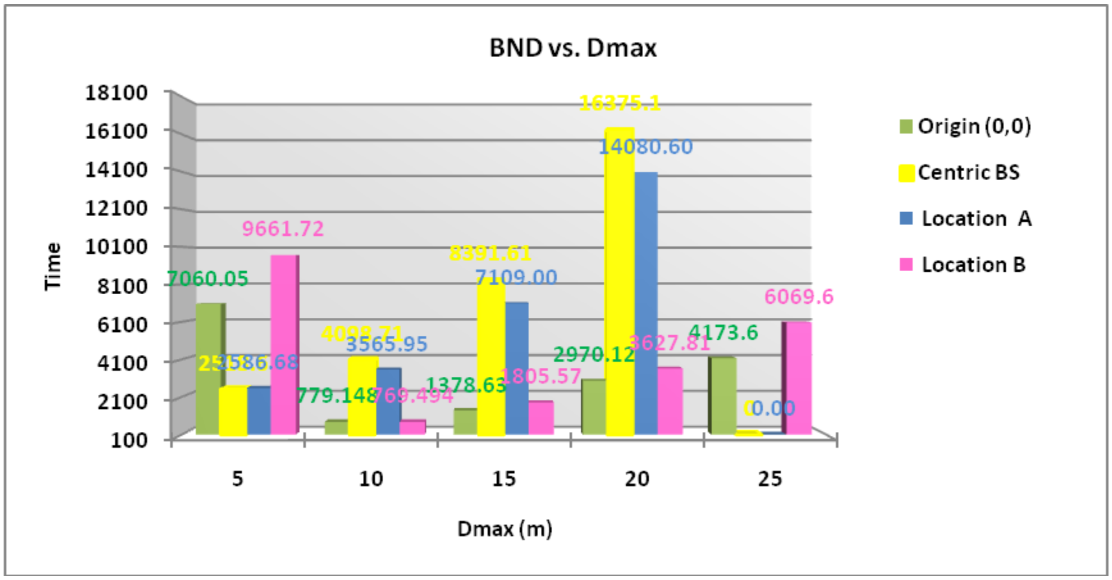


Fig. b.7: BND vs. Dmax

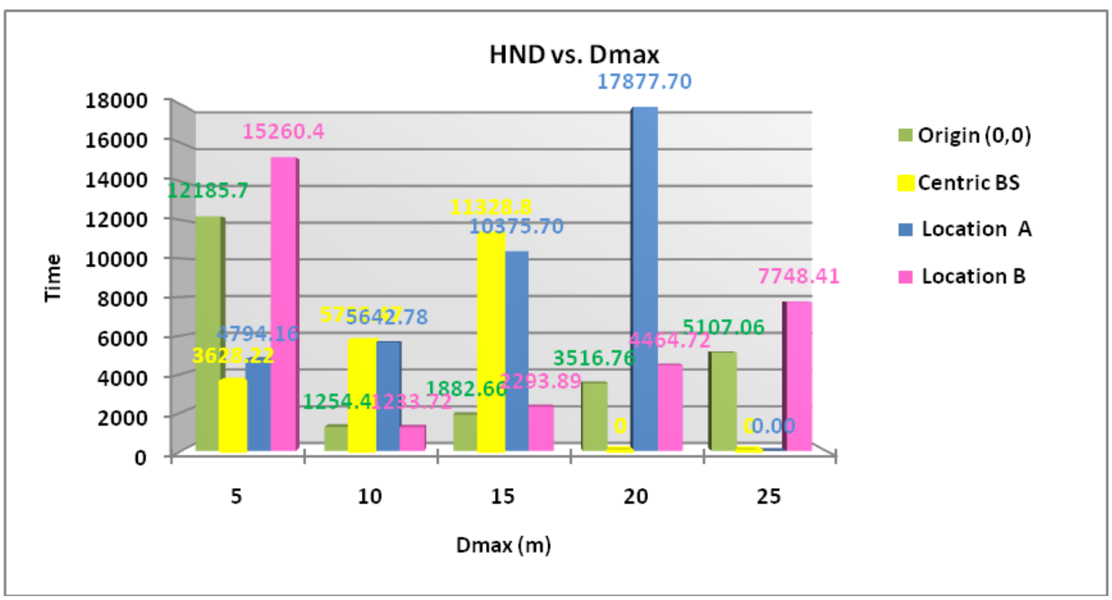


Fig. b.8: HND vs. Dmax

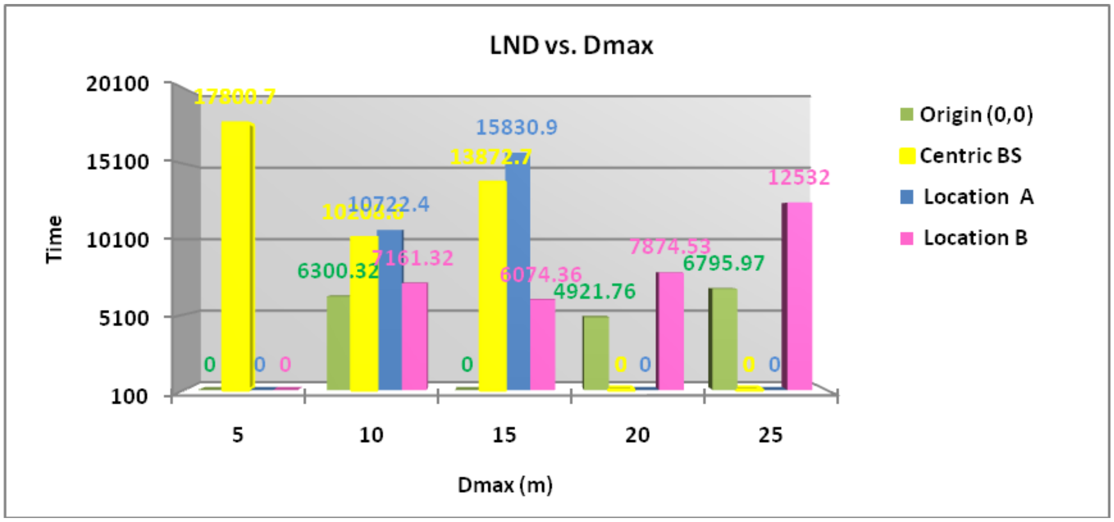


Fig. b.9: LND vs. Dmax

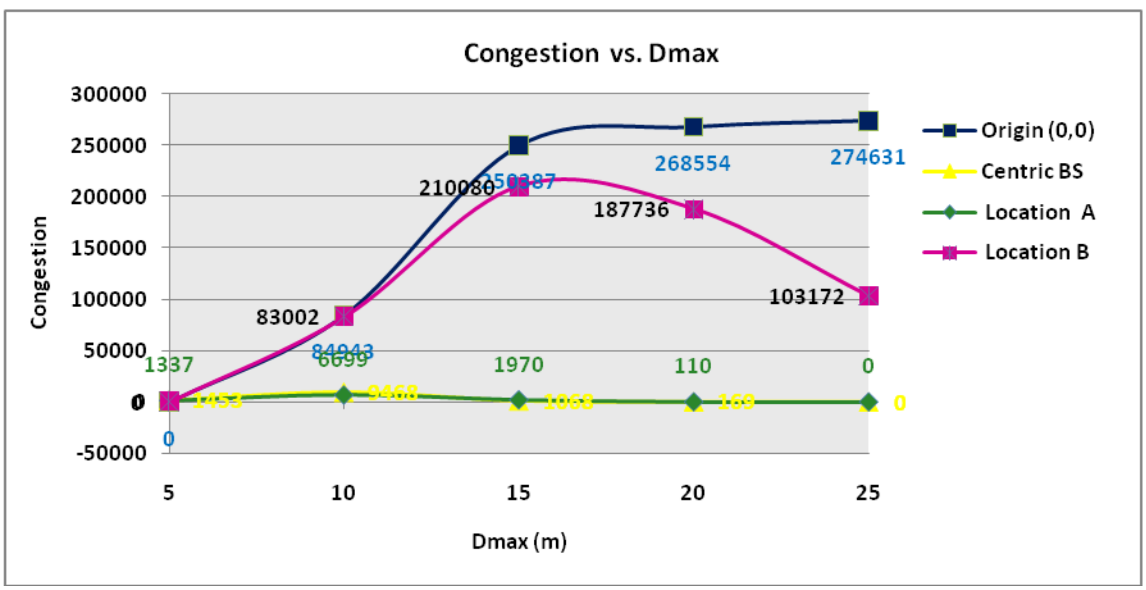


Fig. b.10: Congestion vs. Dmax

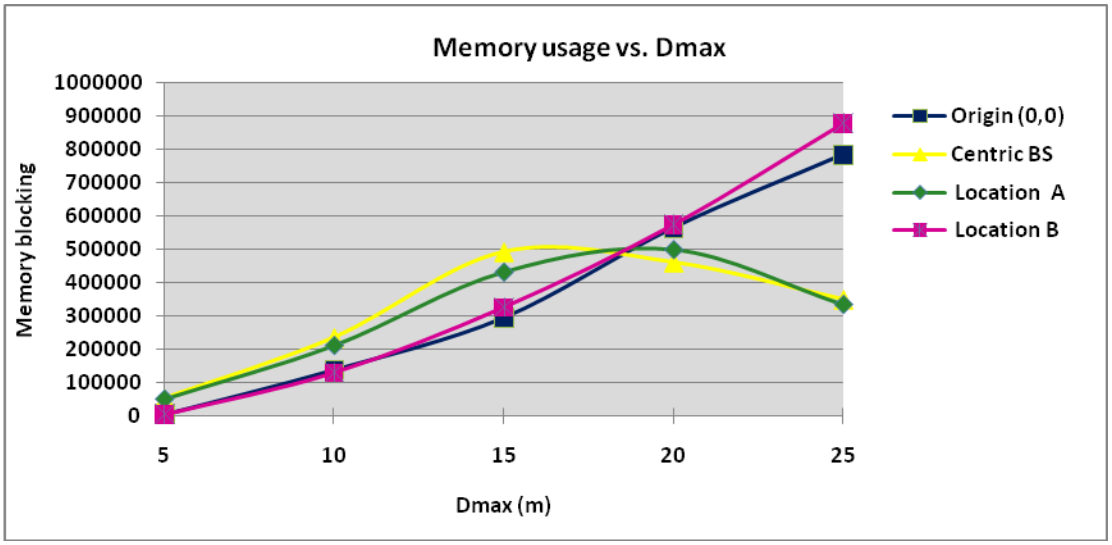


Fig. b.11: Memory usage vs. Dmax

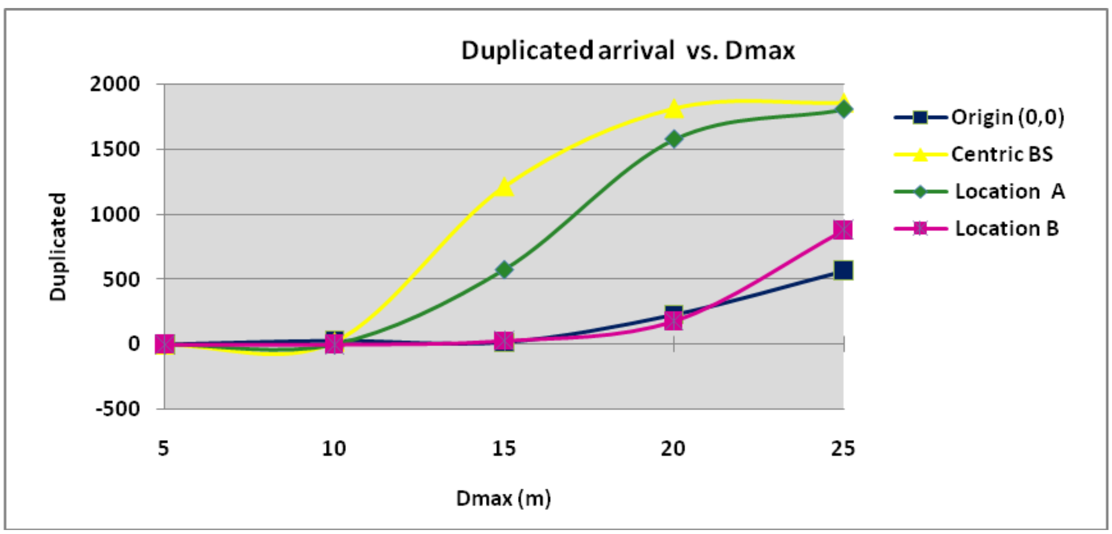


Fig. b.12: Duplicated arrival vs. Dmax

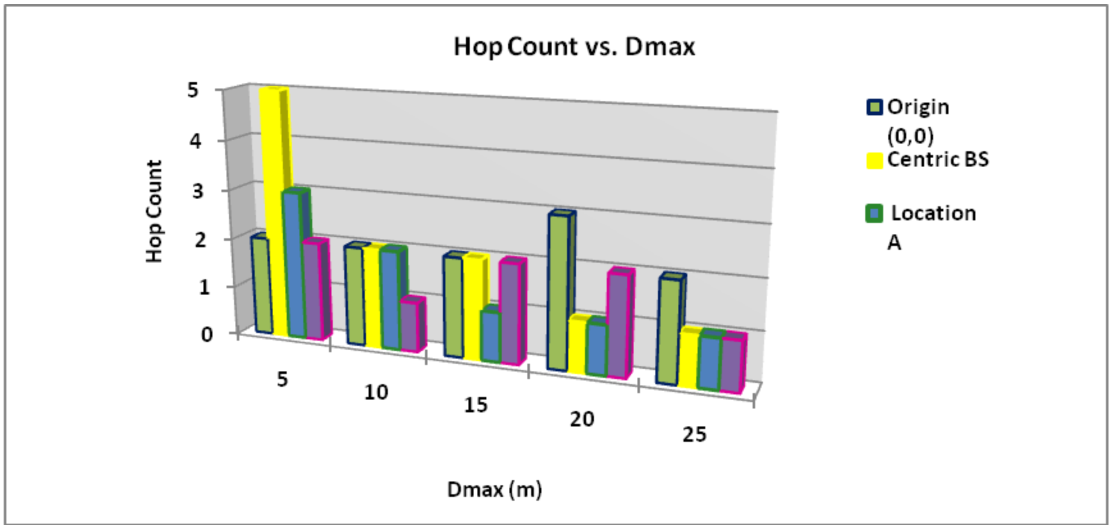


Fig. b.13: Hop count vs. Dmax

4.3.3. Varying Buffer size:

It's expected that buffer size is proportional to throughput, and inversely proportional to delay, conjunction, and duplicated arrival. Maximizing buffer size allows more messages to be stored and routed later.

Given simulation input of 40.000 Simulation messages, Dmax= 10, mean inter-arrival time = 0.5, initial energy per node = 101e+7, Duplicate factor=5, and Dmax= 10 m, we extracted results for buffer size values: 5, 10, 15, and 20. Results are shown in next two tables and figures from c.1 to c.12.

4.3.3.1. Buffer size vs. throughput, Delay, DTJ:

From figures c.1.1, c.1.2, and c.1.3, we noticed that increasing buffer size improves throughput for all networks. Figures c.2 and c.3 show delay and DTJ increased in network "B" and "O", this is because more messages are able access BS after while we increased storage of intermediate nodes. The same figures show a very small change in throughput for increasing buffer size from 10 to 20. Which implies the CEERA's efficiency is not restricted that the amount of memory. For all monitored networks, it's sufficient to include a realistic storage area and no need to extra storage.

4.3.3.2. Buffer size vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

The effect of buffer size over these measures is shown in figures c.4 to c.9. We obtained from these figures, and also from observing simulation run, that there small effect for buffer size over energy dissipation. Number of blocked messages for memory shortage arises and retransmission operations minimized, for this reason its effect over energy dissipation is unpredictable.

4.3.3.3. Buffer size vs. storage sharing, duplicated arrival, Hop count and congestion:

Figures c.10, c.11 and c.12 show that congestion, duplicated arrival and memory sharing is optimized with increasing buffer size.

4.3.3.4. Discussion on Buffer Size variation:

Previous sections show the improvement in performance measures while buffer size increased. However, this improvement is not proportional to buffer size. It’s sufficient to include a realistic storage size and also to take the advantage of duplication factor to manage the available storage.

- **Buffer Size (Buf) and Throughput (Thro.):**

  Buf $\propto$ Throughp……………….. for Buf> 0 and Buf < Buf $_{Threshold}$

- **Buffer Size (Buf) and Congestion (Con):**

  Buf $\propto$ ………………….. for Buf> 0 and Buf < Buf $_{Threshold}$

- **Buffer Size (Buf) and Duplicated arrival (Dup):**

  SF $\propto$ ………………….. for Buf> 0 and Buf < Buf $_{Threshold}$

- **Buffer Size (Buf) and Storage occupation (SS):**

  SF $\propto$ …………………… for Buf> 0 and Buf < Buf $_{Threshold}$

Base station Location	Location A				Centric			
Buffer Size Value	5	10	15	20	5	10	15	20
Throughput	16693	16819	16829	16828	15041	15142	15147	15147
Delay	13.27	12.81	12.63	12.60	15.8171	14.4234	14.0763	14.0662
Delay Time Jitter	20.45	19.63	19.35	19.30	21.5974	19.5582	19.1264	19.1394
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	99	98	99	99	99	98	99	99
FND	1728.57	2092.36	2094.26	2094.26	2057.12	2094.26	2094.39	2094.39
BND	3753.46	3774.94	3565.95	3565.95	3974.86	4068.84	4098.71	4098.71
HND	5595.06	5665.93	5682.95	5642.78	5450	5735.04	5708.29	5708.44
LND	11233.1	0	10816.3	10722.4	10281.9	0	10240.3	10208.6
Congestion	11763	7397	6772	6699	13791	9848	9493	9468
Duplicated Arrival	66	3	0	0	70	30	24	24
hop count /message	2	2	2	2	2	2	2	2
Occupied Memory (Sum)	185062	210859	211829	212247	210067	234281	235529	235809

Table c.1: Performance results of varying the buffer size for network A and C.

Base station Location	Origin (0,0)				Location B			
Buffer Size Value	5	10	15	20	5	10	15	20
Throughput	5697	5686	5680	5677	7565	7567	7567	7567
Delay	12.3398	14.0311	15.1507	16.8269	7.73483	8.92752	10.6559	11.3982
Delay Time Jitter	31.2802	33.261	35.8335	37.7896	26.0107	28.5202	32.649	32.673
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	99	98	99	99	99	99	99	99
FND	630.084	594.85	592.087	580.26	613.779	592.772	584.542	575.872
BND	1050.47	1035.36	859.158	779.148	1017.39	981.397	918.847	769.494
HND	2842.43	1699.67	1351.9	1234.54	3170.77	1577.83	1317.38	1213.54
LND	8895.99	0	7510.84	6300.32	10476.3	9460.8	8010.39	7161.32
Congestion	63422	62985	78731	84943	60590	69276	78755	83002
Duplicated Arrival	11	21	25	28	0	0	0	0
hop count /message	2	2	2	2	1	1	1	1
Occupied Memory (Sum)	123306	126690	136371	140212	113019	121933	127128	131473

Table c.2: Performance results of varying the buffer size for network O and B.

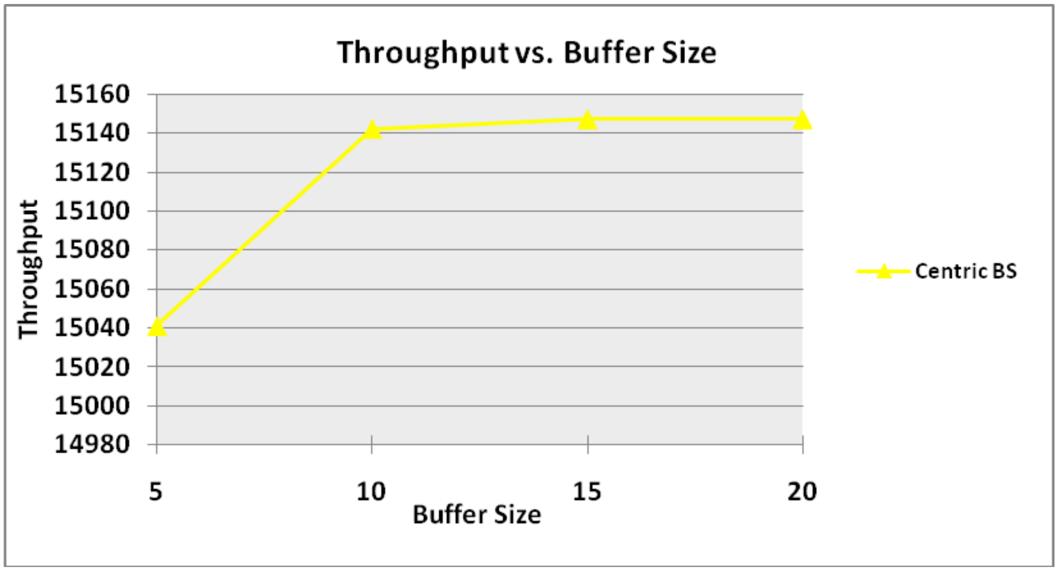


Fig. c.1.1: Throughput vs. Buffer size in centralized BS

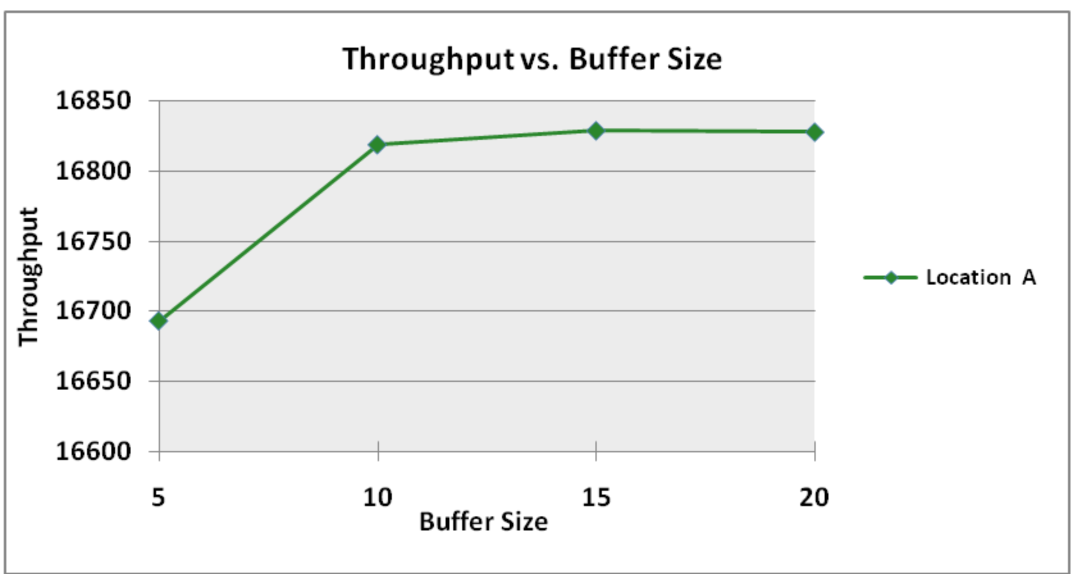


Fig. c.1.2: Throughput vs. Buffer size while BS at location A

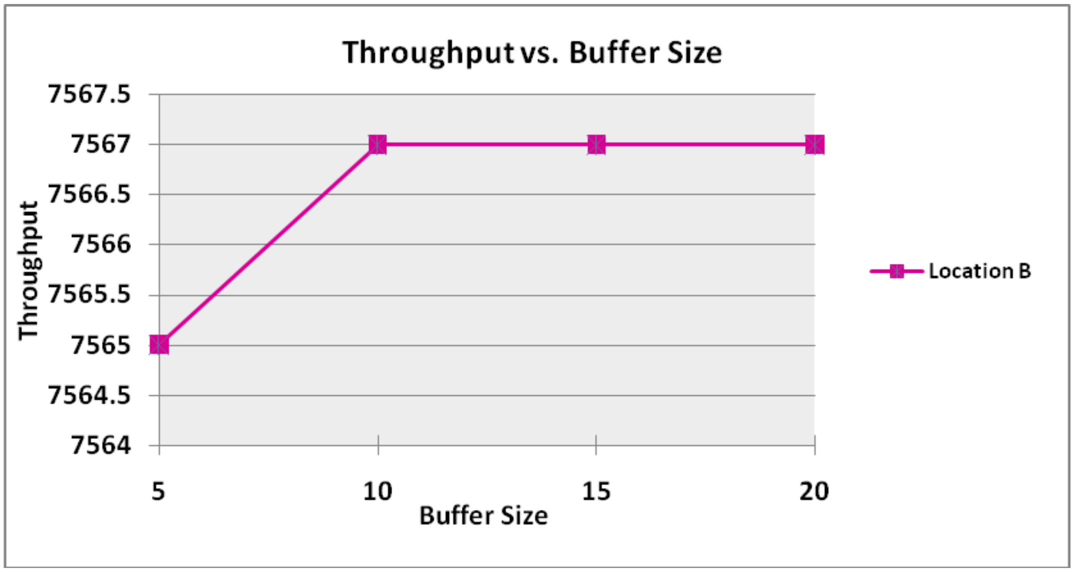


Fig. c.1.3: Throughput vs. Buffer size while BS at location B

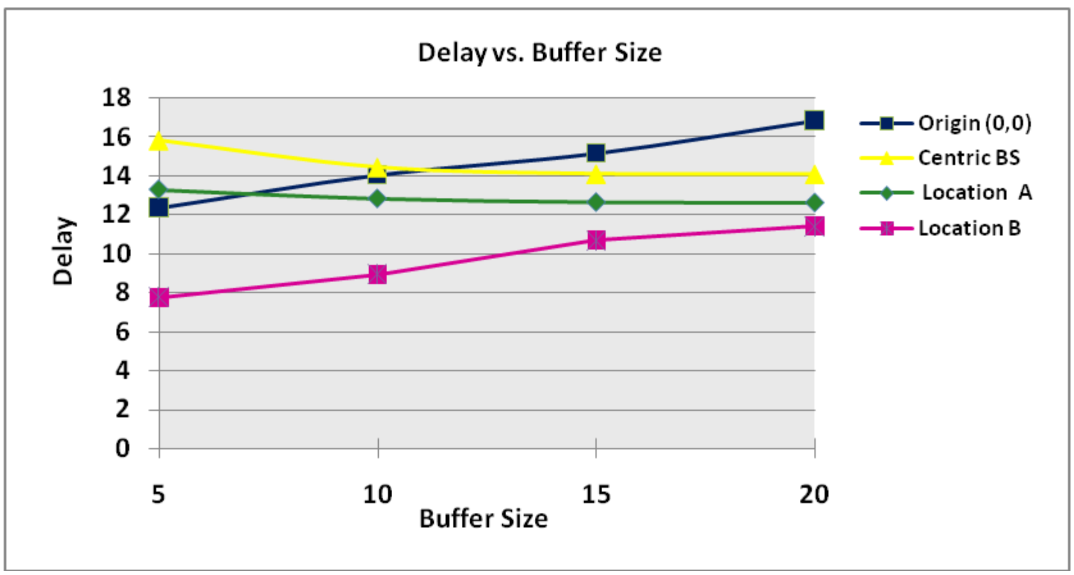


Fig. C.2: Delay vs. Buffer size

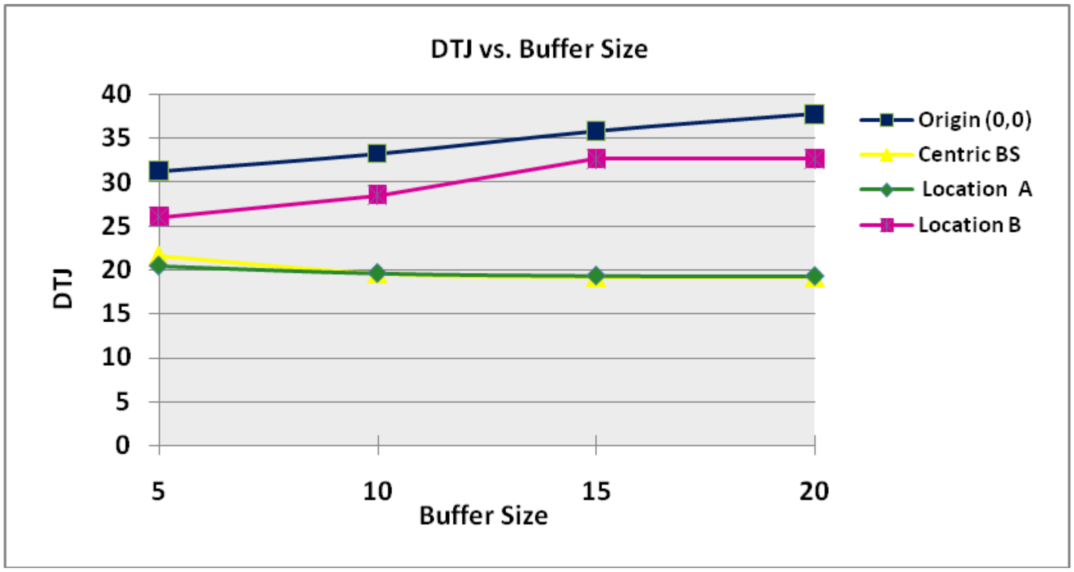


Fig. C.3: DTJ vs. Buffer size

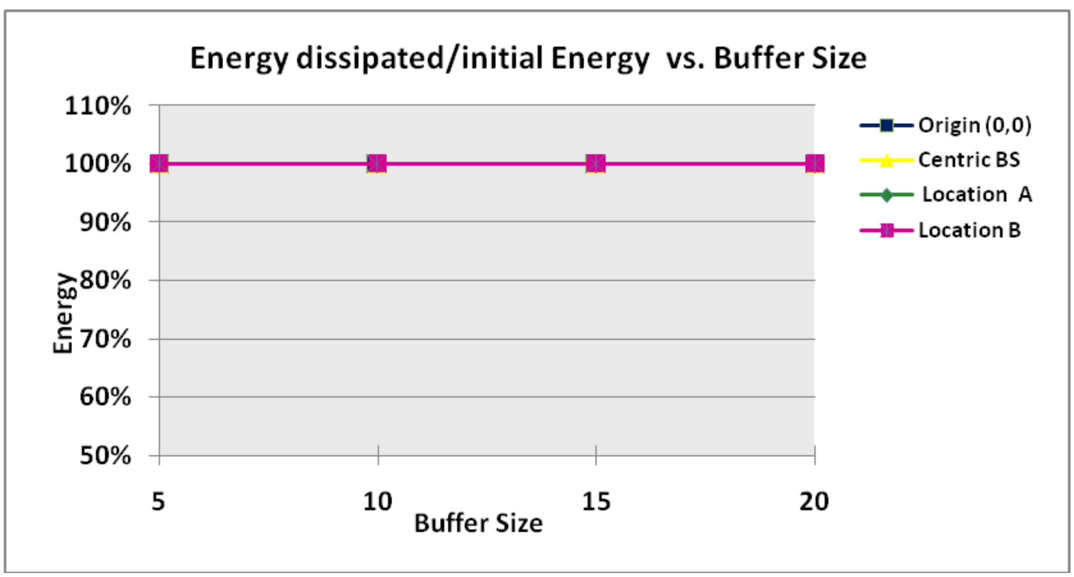


Fig. C.4: Energy dissipated/initial Energy vs. Buffer size

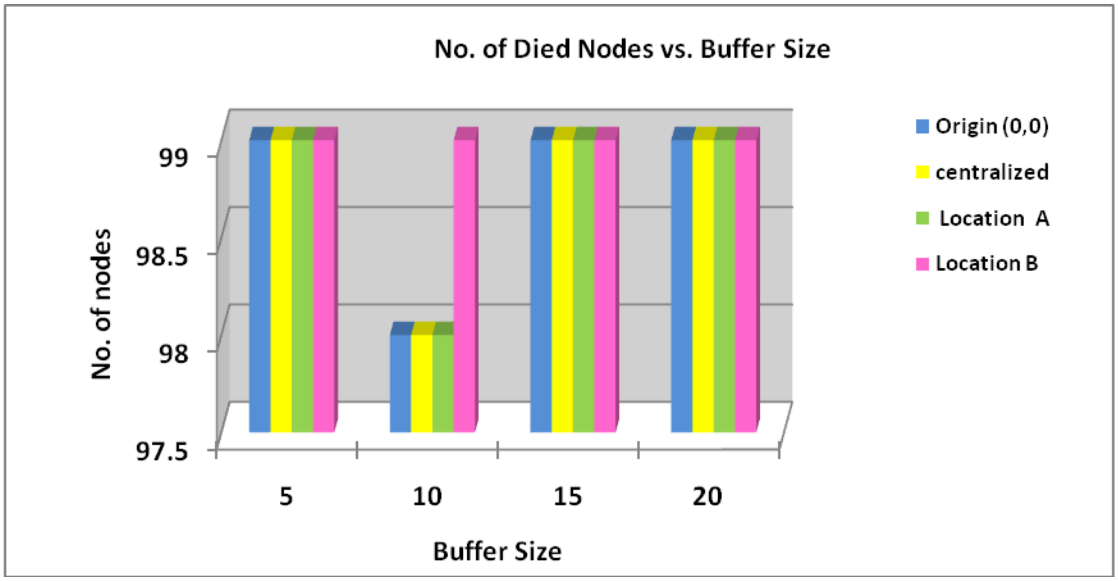


Fig. C.5 : No. of died nodes vs. Buffer size

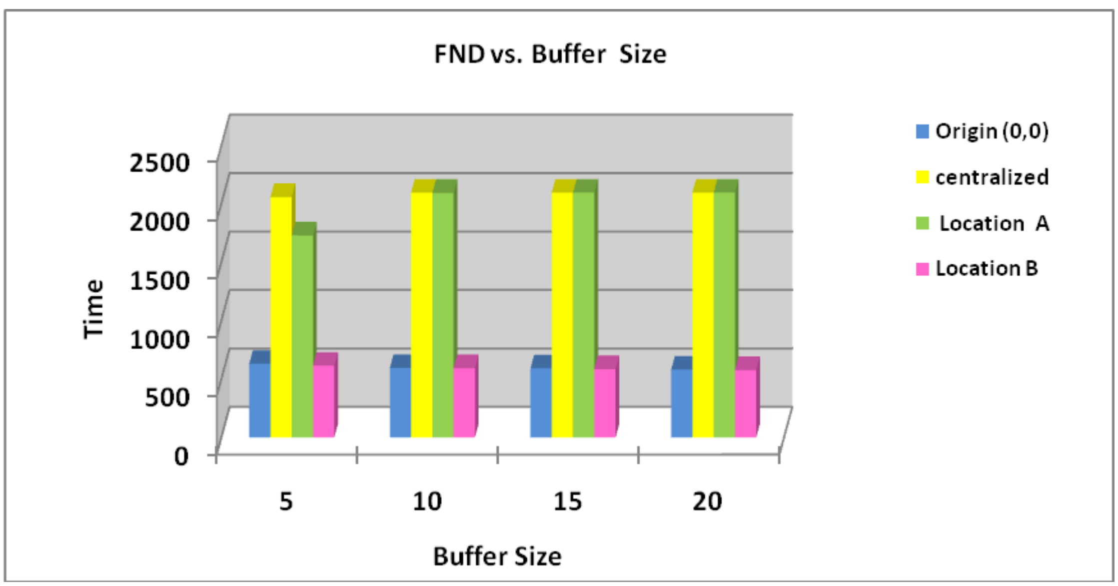


Fig. C.6: FND vs. Buffer size

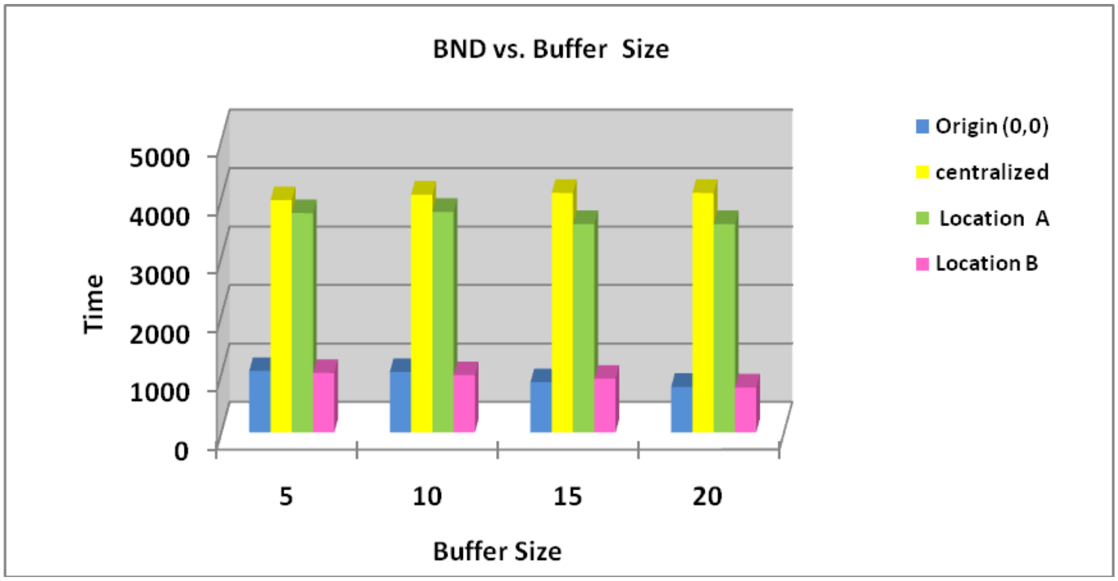


Fig. C.7 : BND vs. Buffer size

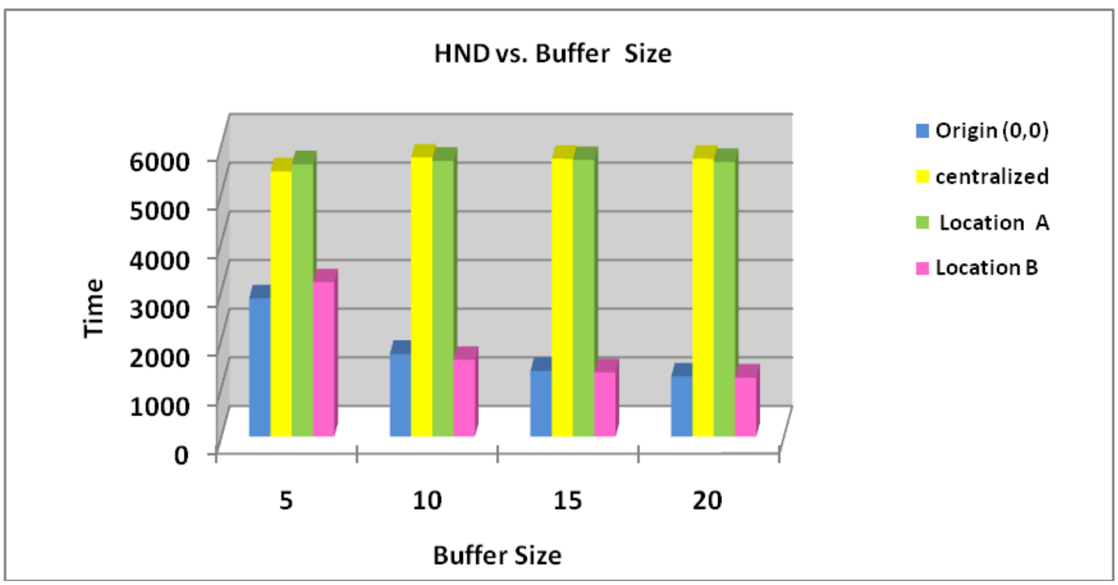


Fig. C.8 : HND vs. Buffer size

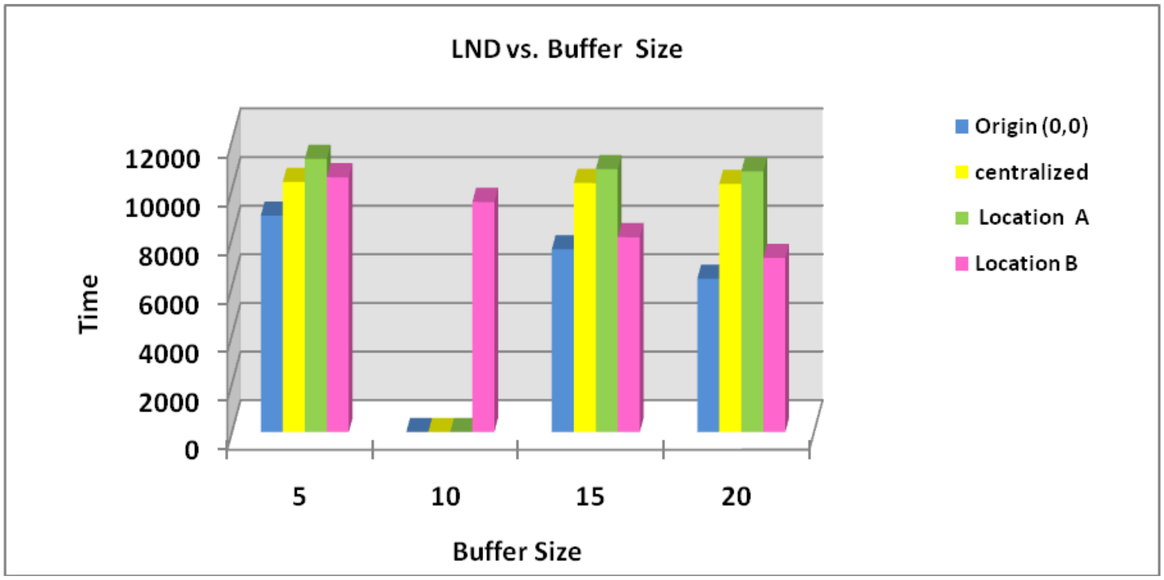


Fig. C.9 : LND vs. Buffer size

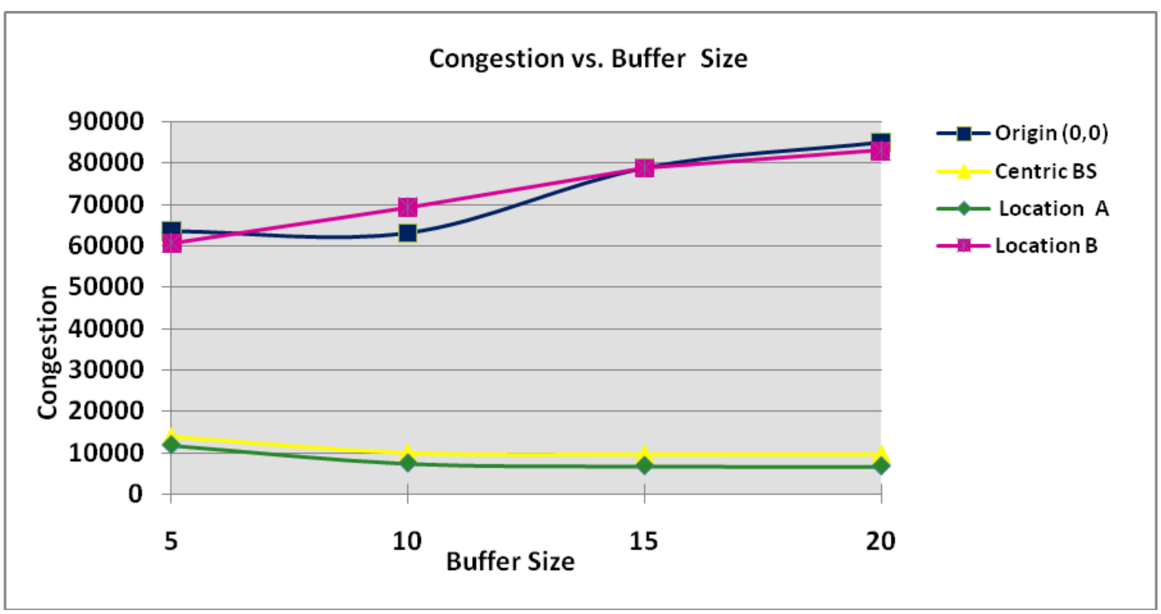


Fig. c.10: Congestion vs. Buffer size

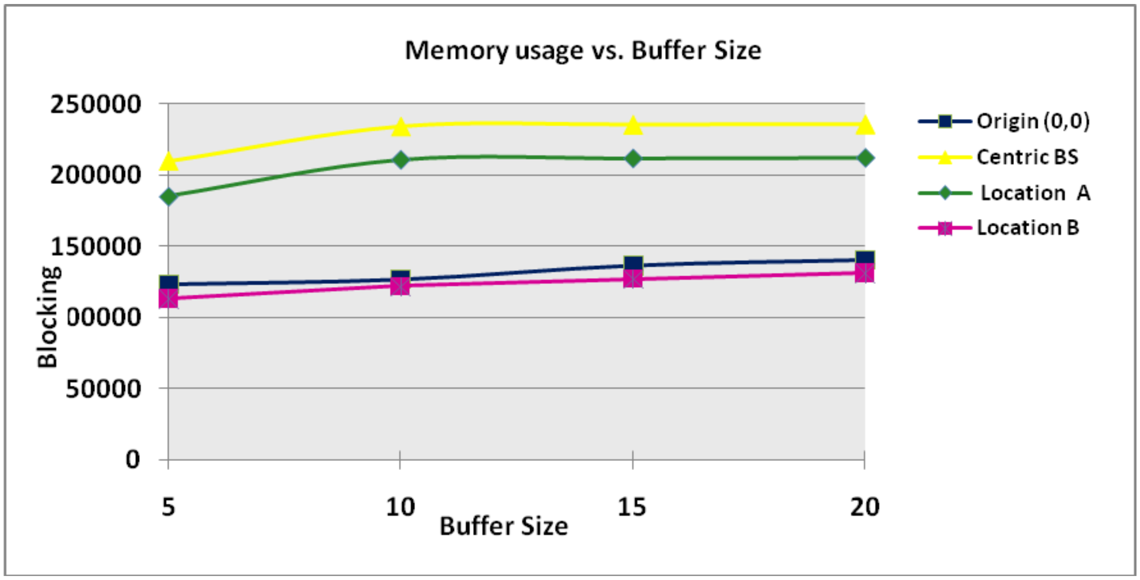


Fig. C.11: Memory usage vs. Buffer size

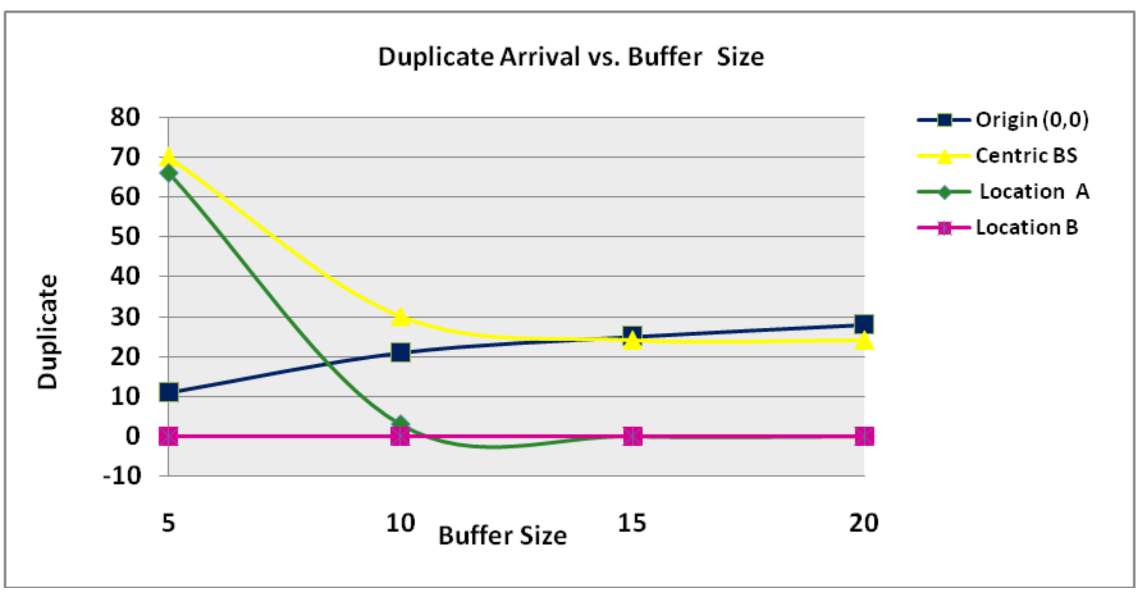


Fig. C.12: Duplicated Arrival vs. Buffer size

4.3.4. Varying Duplication Factor:

The goal of introducing duplication factor is to save memory by limiting long message storage while it's stored in other neighbors. Duplication factor should be sited carefully so that it avoid message loss and to save the available storage.

Given simulation input of 40.000 Simulation messages, Buffer size= 20, mean inter-arrival time = 0.5, initial energy per node = 101e+7, and Dmax= 10 m. We extracted results for duplication factor values: 1, 2, 4, and 8. Results are shown in next two tables and figures from d.1 to d.13.

4.3.4.1. Duplication factor vs. throughput, Delay, and DTJ:

Figures d.1, d.2, and d.3, show that increasing duplication factor value from 1 to 4 improves throughput for all networks. This improvement does not highly change with increasing duplication factor to 8. We obtained that a small value for duplication factor causes message loss, since message deleted when it's received for next times.

4.3.4.2. Duplication factor vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

The effect of over these measures is shown in figures d.4 to d.9. The impact of duplication factor over these measures is similar to the buffer size.

4.3.4.3. Duplication factor vs. storage occupation, duplicated arrival, Hop count and congestion:

Figures d.10, d.11, d.12 and d.13 show that congestion, duplicated arrival and memory sharing is affected by increasing duplication factor. A higher value for this factor would increase duplicated arrival and congestion. On the other hand, it increases memory sharing.

4.3.4.4. Discussion on the mean inter-arrival:

Previous sections show that MIT has a very small role of duplication factor in saving both memory and energy. The optimal value for duplication depends on available storage, no. of nodes, and deployment.

- **Duplication factor (DF) and Throughput (Thro.):**

  $\mathbf{DF} \propto \text{Throughp}$……………….. for DF> 0 and DF < DF $_{\text{Threshold}}$

- **Duplication factor (DF) and Storage occupation (SO):**

  $\mathbf{DF} \propto$ ……………….. for DF> 0 and DF < DF $_{\text{Threshold}}$

Base station Location	Random A				Centric			
Duplication Factor Value	1	2	4	8	1	2	4	8
Throughput	7679	15474	16823	16825	6829	14853	15166	15157
Delay	0.00	14.89	12.99	11.67	8.63684E-05	15.7574	14.7055	13.1348
Delay Time Jitter	0.00	22.03	19.45	18.09	0.000010457	22.3649	19.4083	17.9672
Total energy dissipated/initial Energy	46%	90%	100%	100%	46%	92%	100%	100%
No. of Died Nodes	0	51	99	99	0	56	98	99
FND	0.00	3403.07	2143.93	2086.60	0	4053.48	2603.33	2094.26
BND	0.00	8071.60	3900.62	3489.31	0	7626.29	4486.14	3842.34
HND	0.00	19128.90	6027.44	5230.37	0	15683.6	6168.45	5170.93
LND	0	0	11584.8	10219.4	0	0	0	9445.87
Congestion	0	7686	11849	1236	0	7569	13714	1679
Duplicated Arrival	0	0	4	5	0	0	22	17
hop count /message	1	2	2	2	1	2	2	2
Occupied Memory (Sum)	4524	131865	216977	205257	2174	159085	243450	223808

Table d.1: Performance results of varying the duplication factor for network A and C.

Base station Location	Origin				Random B			
Duplication Factor Value	1	2	4	8	1	2	4	8
Throughput	2379	4614	5676	5683	3198	4244	7568	7567
Delay	0.00009	9.9849	15.8658	16.1377	8.25E-05	7.68793	9.8359	10.6018
Delay Time Jitter	1.31798E-11	20.3929	36.9605	37.6834	1.39E-05	19.1283	29.9006	32.3445
Total energy dissipated/initial Energy	44%	67%	100%	100%	44%	62%	100%	100%
No. of Died Nodes	0	16	99	99	0	14	99	99
FND	0	1778.42	593.887	579.042	0	1900.16	584.282	568.292
BND	0	0	916.599	690.98	0	0	907.325	695.672
HND	0	0	1954.72	907.112	0	0	1907.44	860.213
LND	0	0	10052.3	4433.22	0	0	11343.3	5541.42
Congestion	0	8813	74922	66544	0	6728	74186	66574
Duplicated Arrival	0	0	31	23	0	0	0	0
hop count /message	1	2	2	2	1	1	1	1
Occupied Memory (Sum)	2047	49562	139594	121189	5453	32022	129203	112627

Table d.2: Performance results of varying the duplication factor for network O and B.

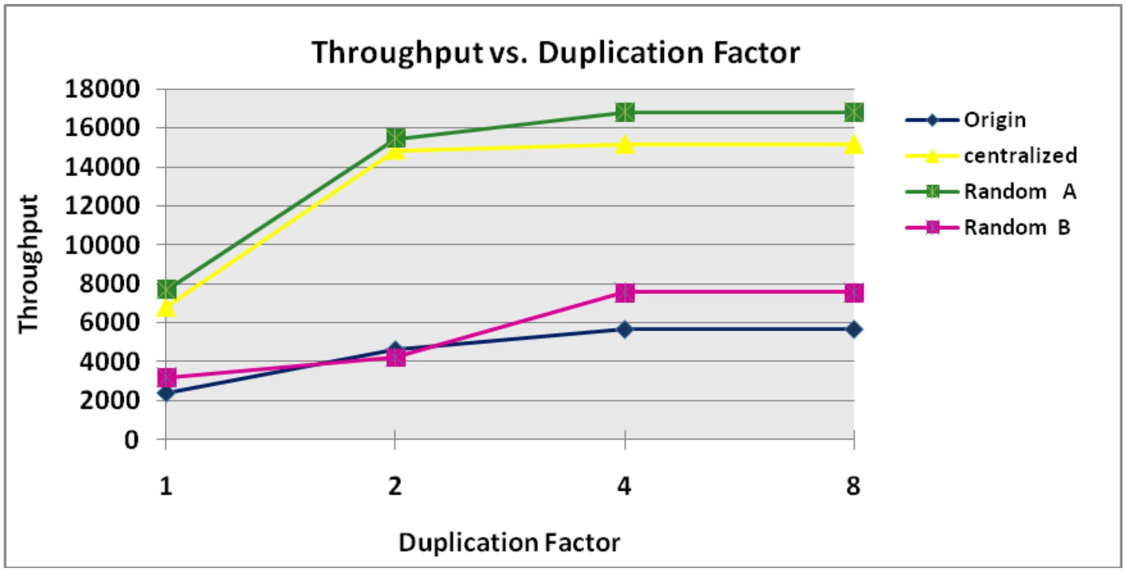


Fig. D.1: Throughput vs. duplication factor

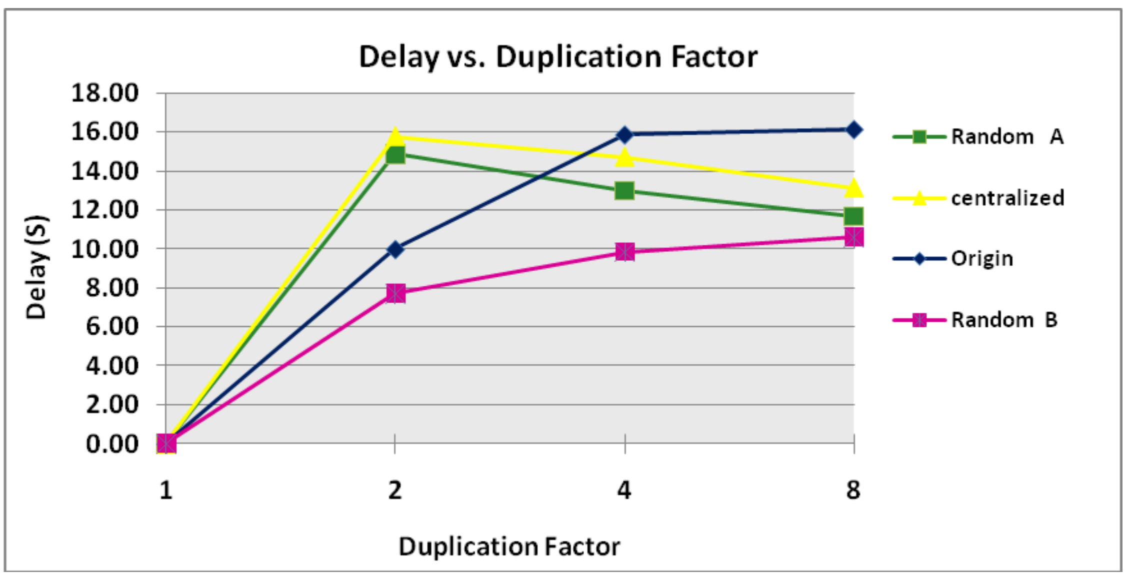


Fig. D.2: Delay vs. duplication factor

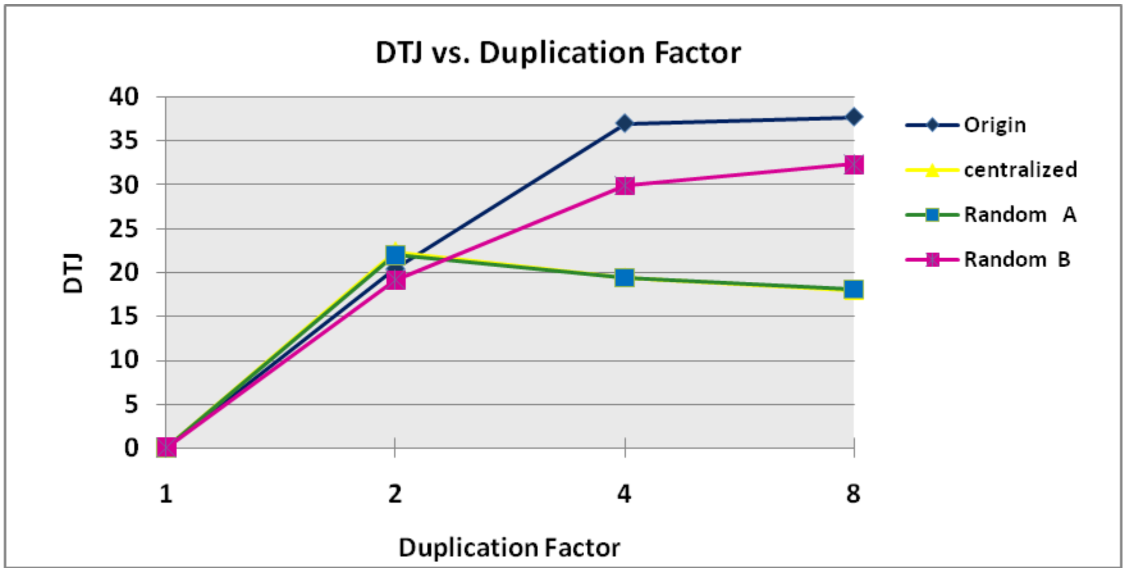


Fig. D.3: DTJ vs. duplication factor

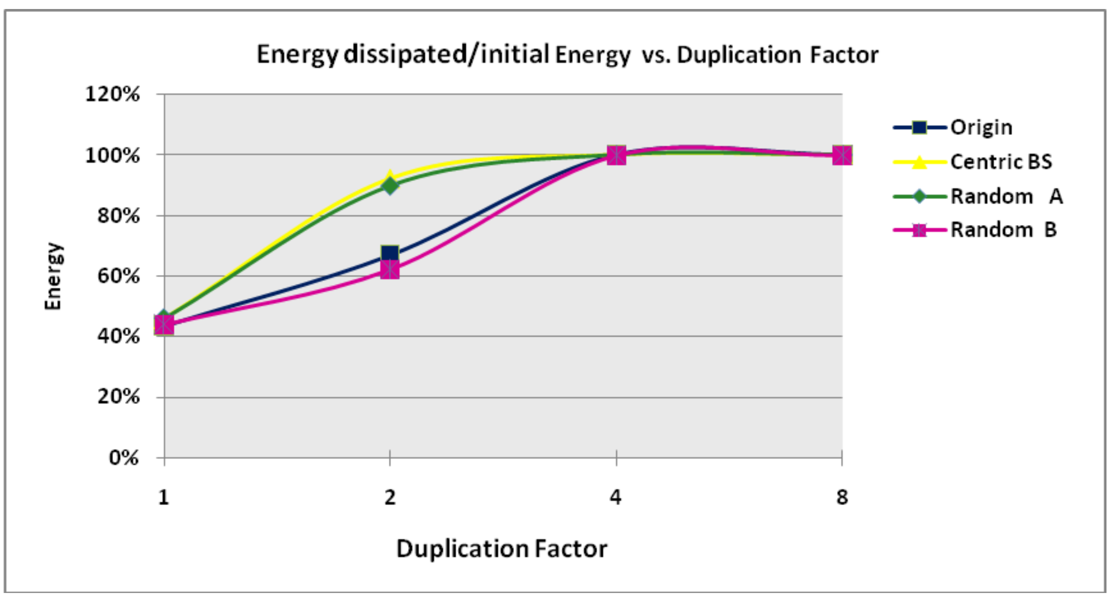


Fig. D.4: Energy dissipated/ per initial energy vs. duplication factor

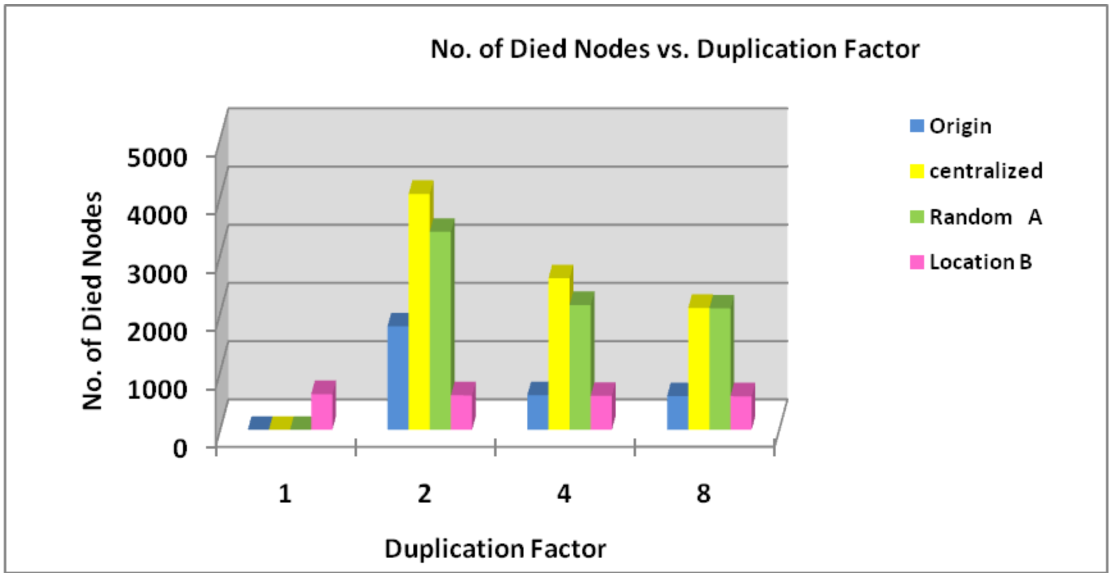


Fig. D.5: No. of died nodes vs. duplication factor

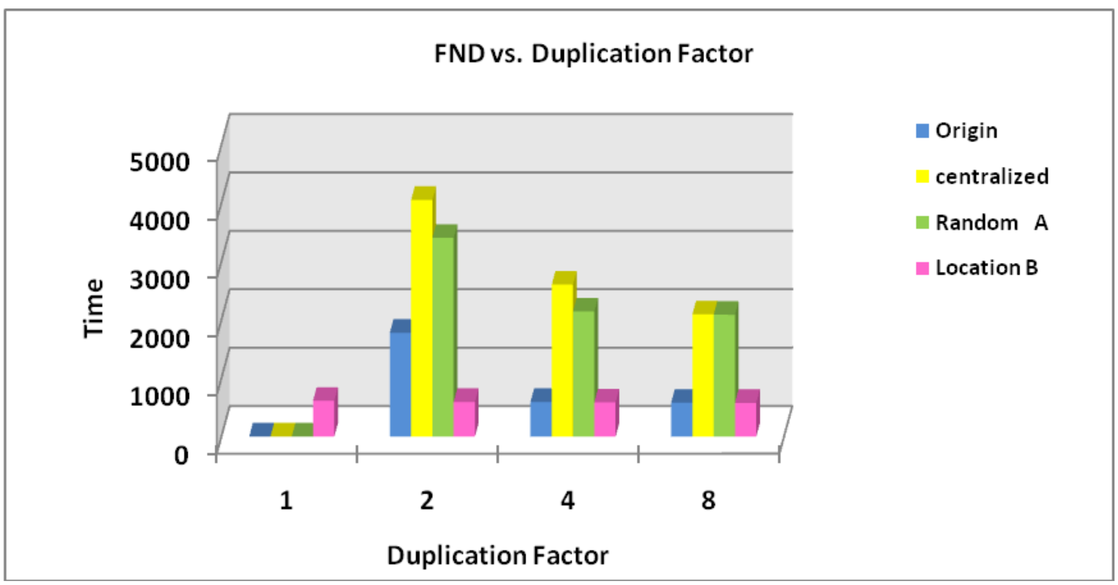


Fig. D.6: FND vs. duplication factor

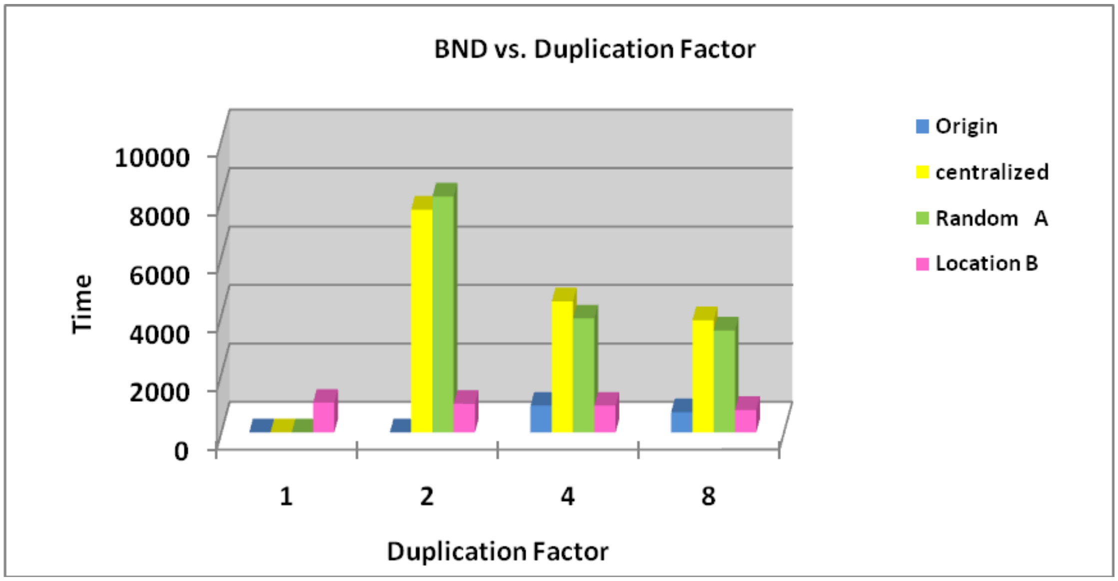


Fig. D.7: BND vs. duplication factor

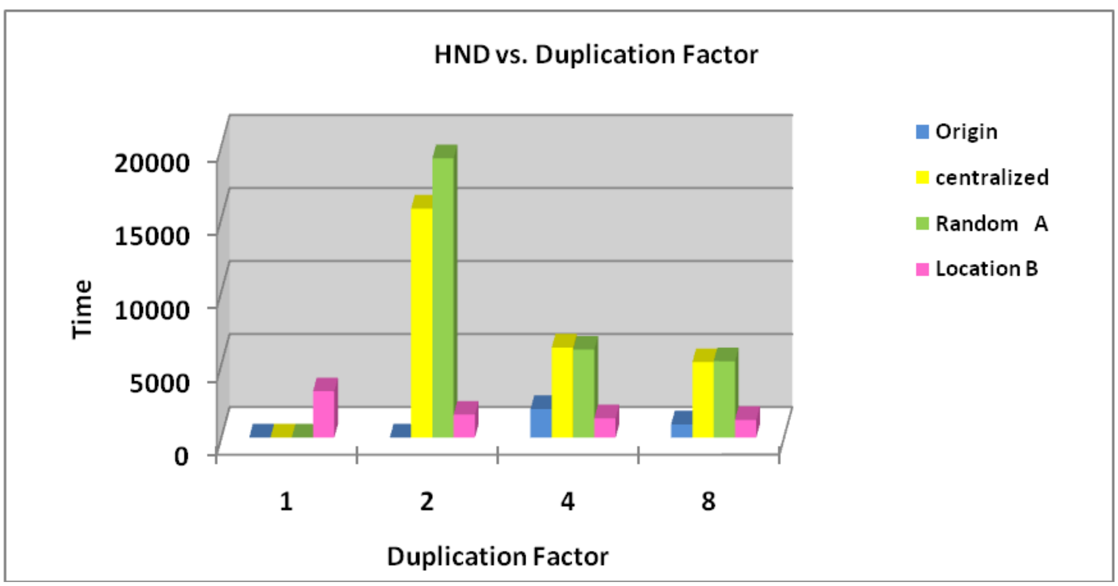


Fig. D.8: HND vs. duplication factor

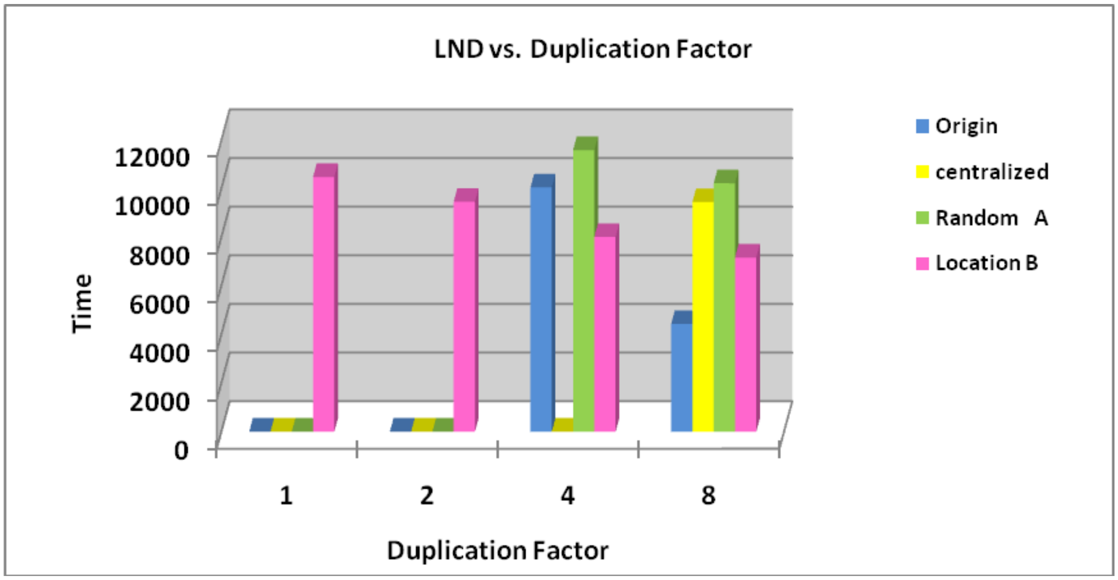


Fig. D.9: LND vs. duplication factor

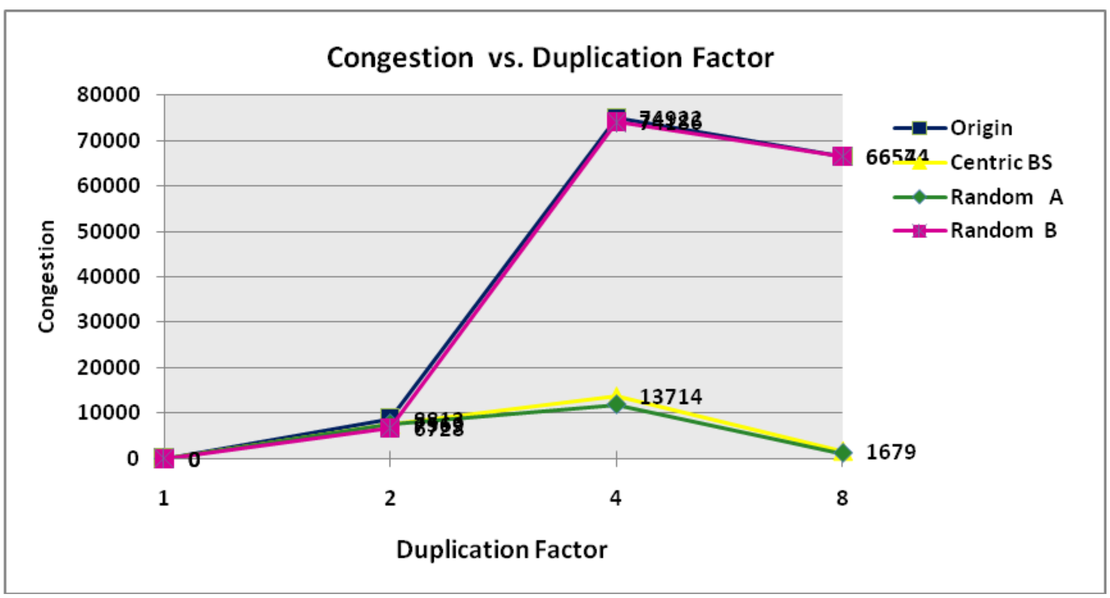


Fig. D.10: Congestion vs. duplication factor

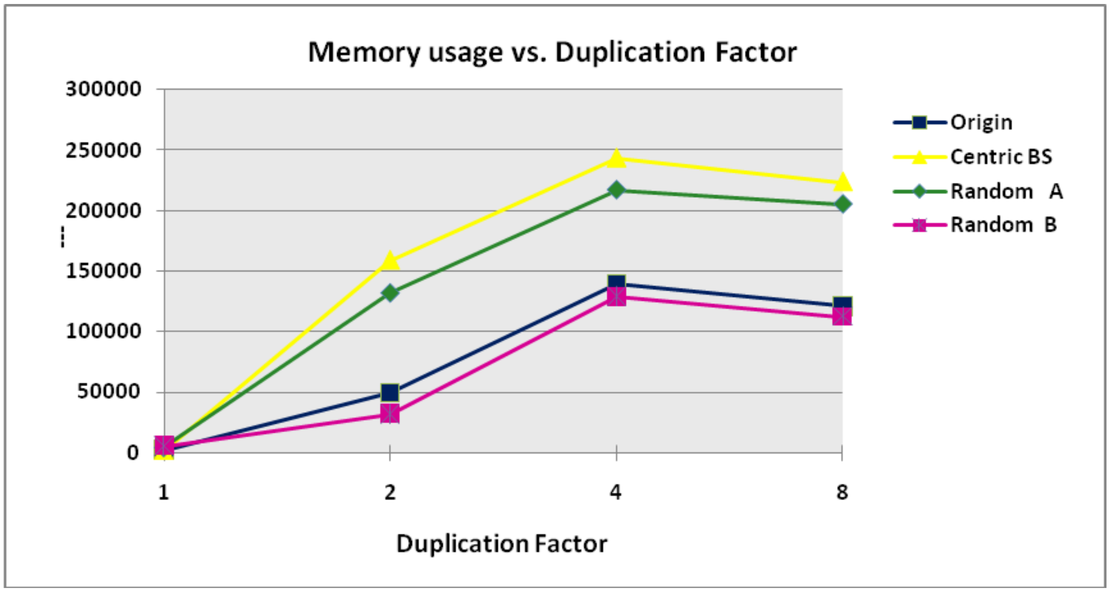


Fig. D.11: Memory usage vs. duplication factor

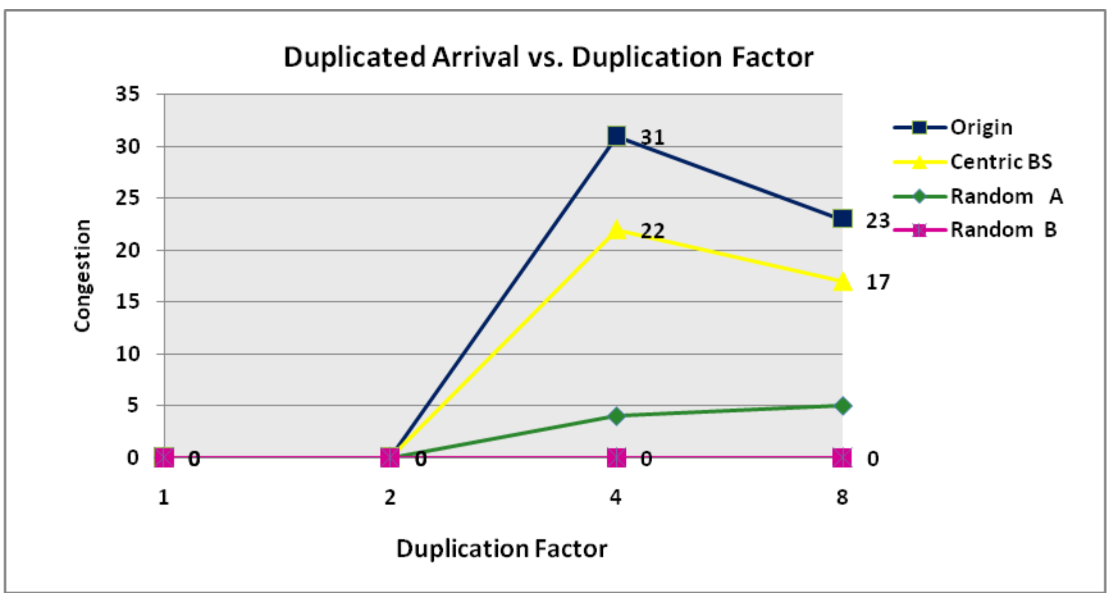


Fig. D.12: Duplicated Arrival vs. duplication factor

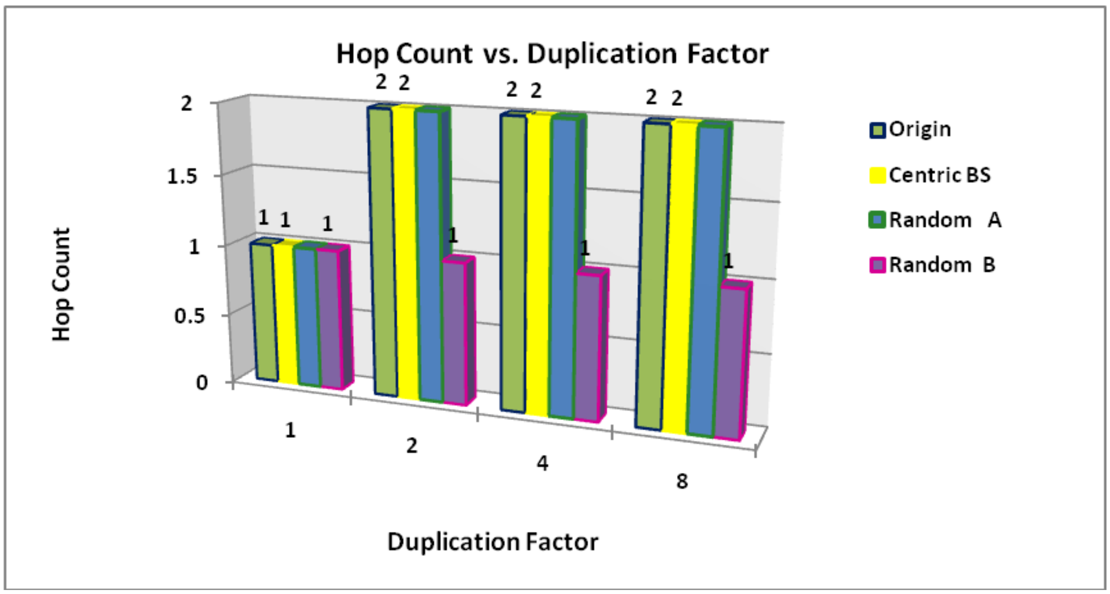


Fig. D.13: Hop count vs. duplication factor

4.3.5. Varying mean inter-arrival time:

The inter-arrival time is explained in section 4.3 as an environment-dependent variable. We want to examine CEERA under variable traffic load.

Given simulation input of 40.000 Simulation messages, Buffer size= 20, initial energy per node = 101e+7, Duplicate factor=5, and Dmax= 10 m. We extracted results for mean inter-arrival values: o.1, 0.5, 1, and 2. Results are shown in next two tables and figures from e.1 to e.13.

4.3.5.1. Mean inter-arrival time vs. throughput, Delay, and DTJ:

Figures e.1, e.2, and e.3, show that increasing mean inter-arrival time (MIT) value from 0.1 to 2 improves throughput for all networks especially for network “B” and “O” as expected. However, this effect still small compared to other factors.

4.3.5.2. Mean inter-arrival time vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

The effect of over these measures is shown in figures e.4 to e.9. Increasing MIT increases the values of these measures.

4.3.5.3. Mean inter-arrival time vs. storage occupation, duplicated arrival, Hop count and congestion:

Figures e.10, e.11, e.12 and e.13 show that congestion, duplicated arrival and memory sharing is affected by increasing mean inter-arrival time. A higher value for MIT allows more messages to be stored in key nodes (within BS distance) for this utilizing memory decreased, where duplicated arrival and congestion increase.

4.3.5.4. Discussion on the variation of mean inter arrival:

Previous sections show that MIT has a small role on performance measures compared to other factors. Its value is not be manipulated as it’s controlled by the sensed environment. We obtained the following proportional with MIT.

- **Mean inter-arrival time (MIT) and network lifetime (LT.):**

MIT $\propto$ ……………….. for MIT> 0 and MIT < ∞

Base station Location	Random A				Centric			
Mean inter-arrival time	0.1	0.5	1	2	0.1	0.5	1	2
Throughput	16699	16828	16749	16749	15059	15147	15100	15096
Delay	15.62	12.60	12.17	12.01	16.4276	14.0662	14.6814	14.4741
Delay Time Jitter	23.33	19.30	20.09	19.90	21.8489	19.1394	20.6659	20.6058
Total energy dissipated/initial Energy	100%	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	98	99	99	99	98	99	98	98
FND	421.45	2094.26	5590.70	11172.50	476.586	2094.39	5660.58	11293.9
BND	861.32	3565.95	8190.21	16331.50	939.155	4098.71	8826.3	17627.7
HND	1328.05	5642.78	13818.20	26749.30	1310.64	5708.44	13801.7	27261.4
LND	0	10722.4	28595.9	57699.6	0	10208.6	0	0
Congestion	24910	6699	17418	17428	25738	9468	19162	18972
Duplicated Arrival	60	0	75	74	52	24	105	105
hop count /message	2	2	2	2	2	2	2	2
Occupied Memory (Sum)	192689	212247	222853	222208	223155	235809	253356	252455

Table e.1: Performance results of varying mean inter-arrival time for network A and C.

Base station Location	Origin				Random B			
Mean inter-arrival time	0.1	0.5	1	2	0.1	0.5	1	2
Throughput	5704	5677	5020	4894	7568	7567	5157	4937
Delay	11.3586	16.8269	9.38026	9.58845	6.823	11.3982	6.94545	6.75798
Delay Time Jitter	28.4089	37.7896	20.2107	20.1062	23.9872	32.673	19.524	18.3907
Total energy dissipated/initial Energy	100%	100%	87%	81%	100%	100%	84%	80%
No. of Died Nodes	99	99	41	35	99	99	37	31
FND	215.049	580.26	1342.24	2522.64	213.898	575.872	1321.78	2592.99
BND	341.67	779.148	5259.94	19926	345.367	769.494	4205.4	22036.4
HND	853.407	1234.54	0	0	826.817	1213.54	0	0
LND	2406.44	6300.32	0	0	2529.46	7161.32	0	0
Congestion	77036	84943	24640	19272	87178	83002	26844	17419
Duplicated Arrival	6	28	0	3	0	0	0	0
hop count /message	2	2	1	2	1	1	1	1
Occupied Memory (Sum)	134246	140212	81581	72215	127462	131473	70678	59077

Table e.2: Performance results for network for when BS is at origin and at location B.

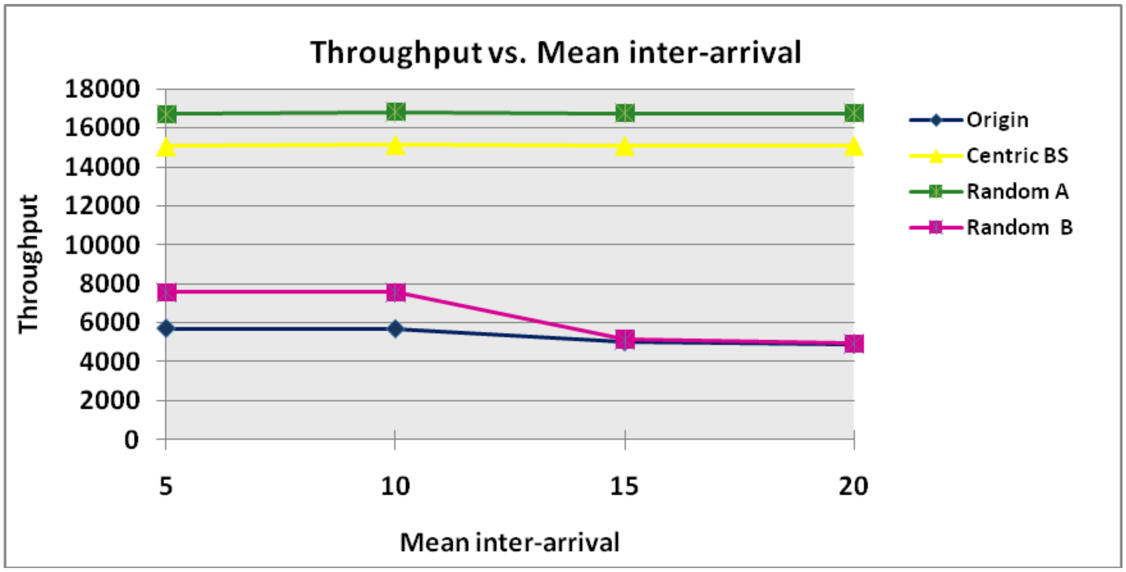


Fig. e.1: Throughput vs. Mean inter-arrival time

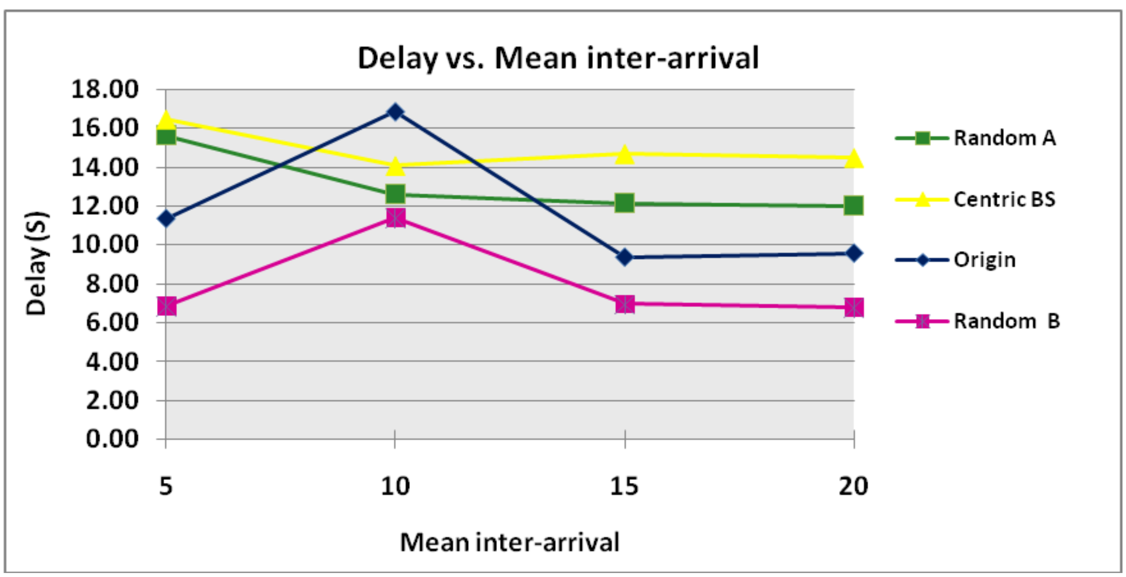


Fig. e.2: Delay vs. Mean inter-arrival time

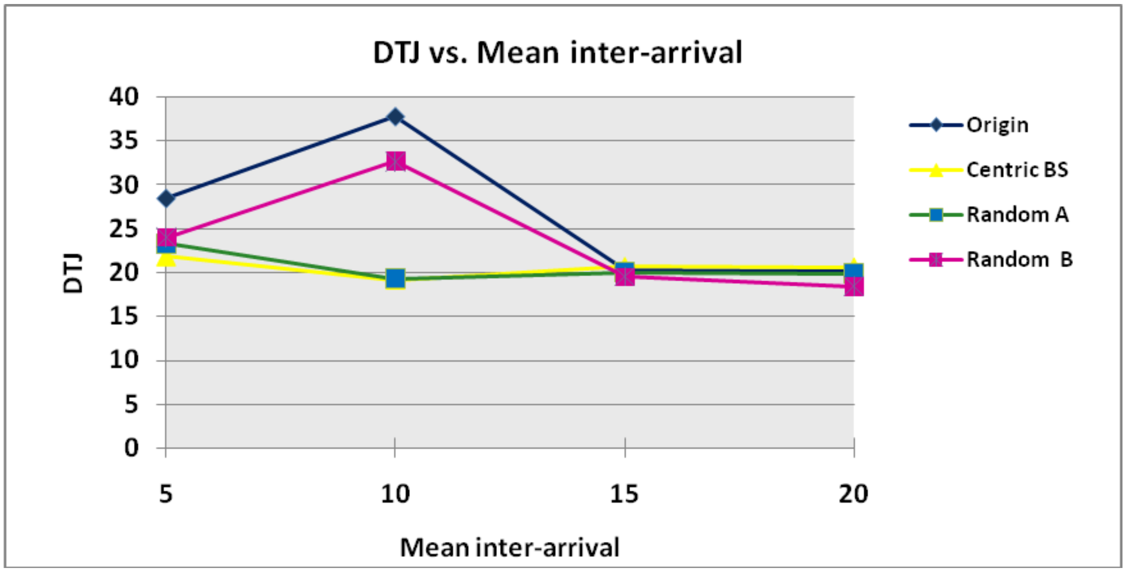


Fig. e.3: DTJ vs. Mean inter-arrival time

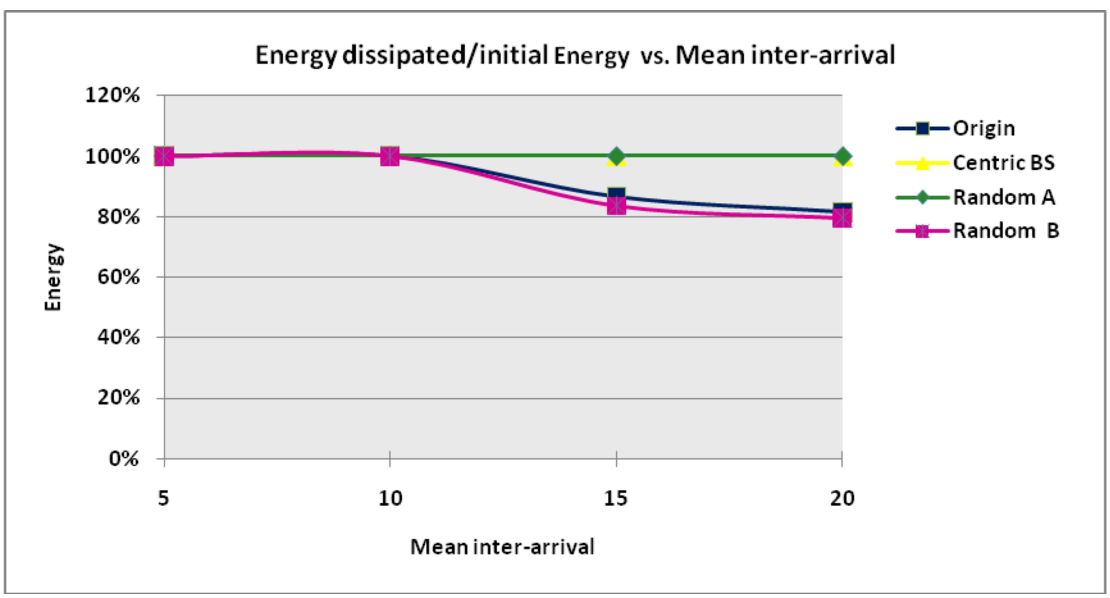


Fig. e.4: Energy dissipated/initial Energy vs. Mean inter-arrival time

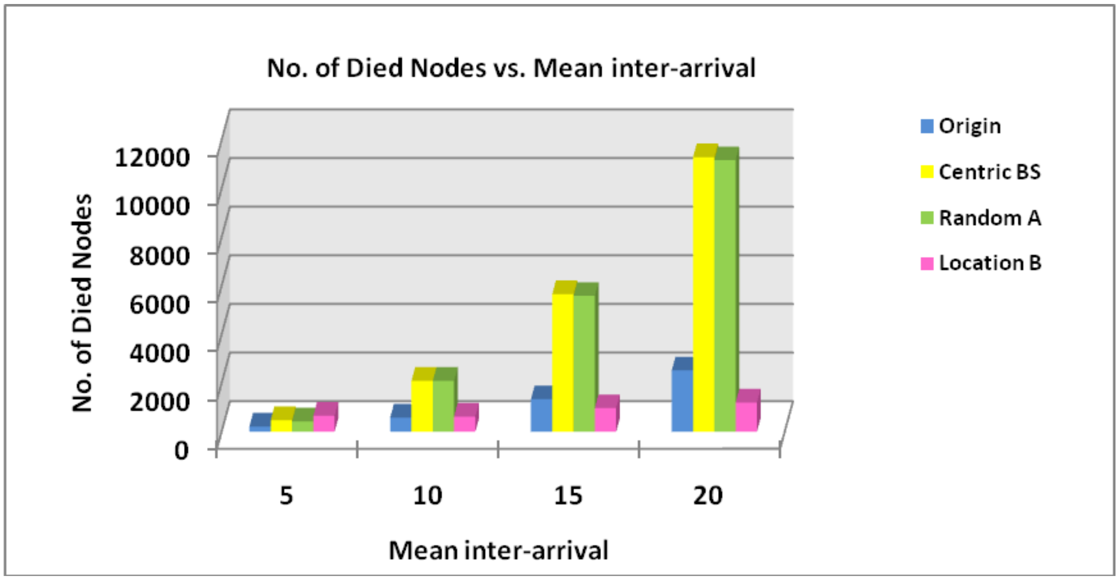


Fig. e.5: No. of die nodes vs. Mean inter-arrival time

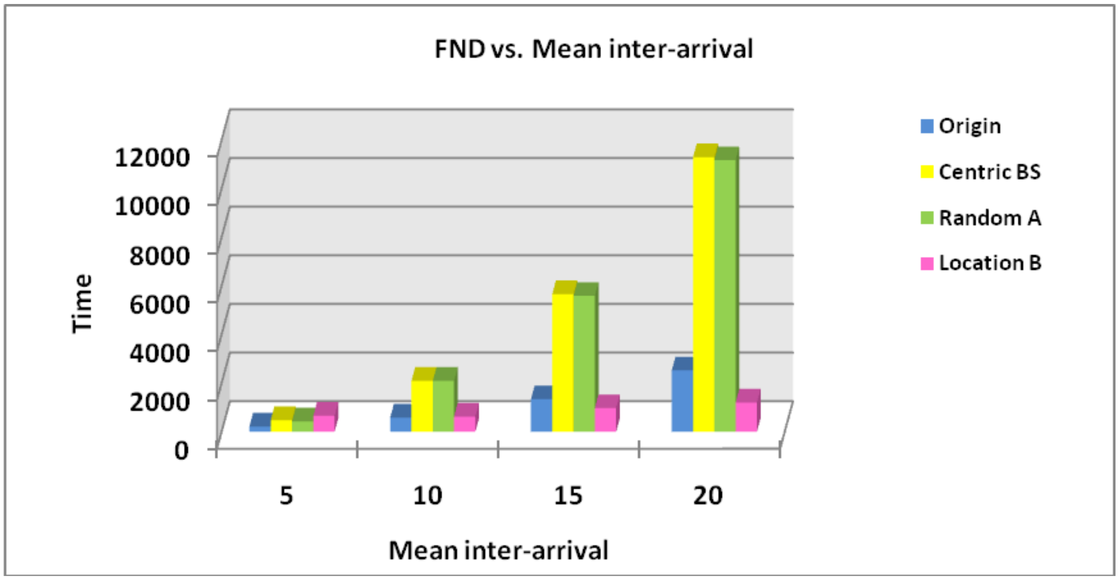


Fig. e.6: FND vs. Mean inter-arrival time

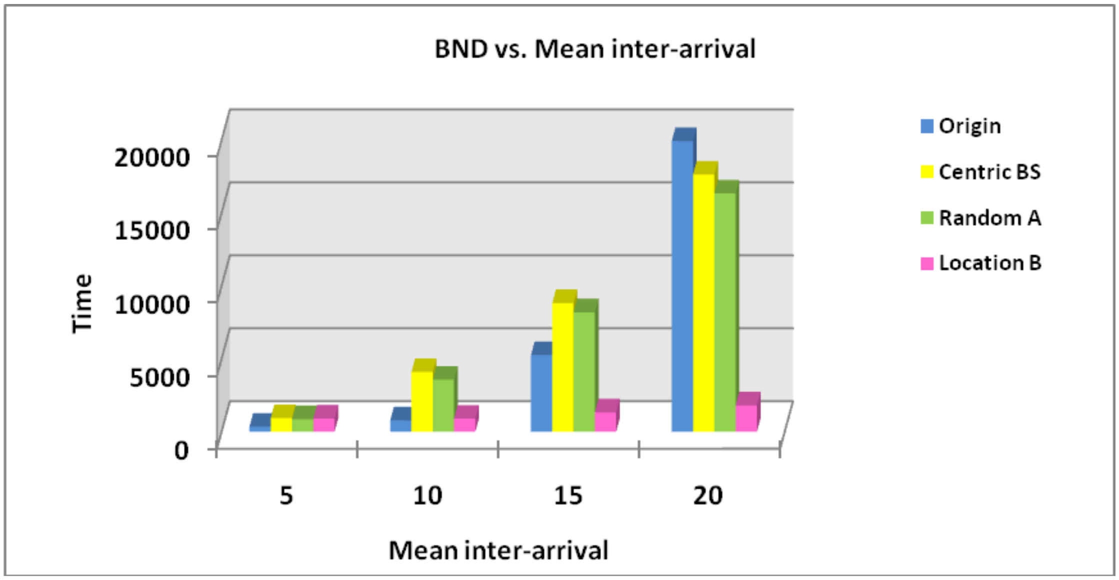


Fig. e.7: BNF vs. Mean inter-arrival time

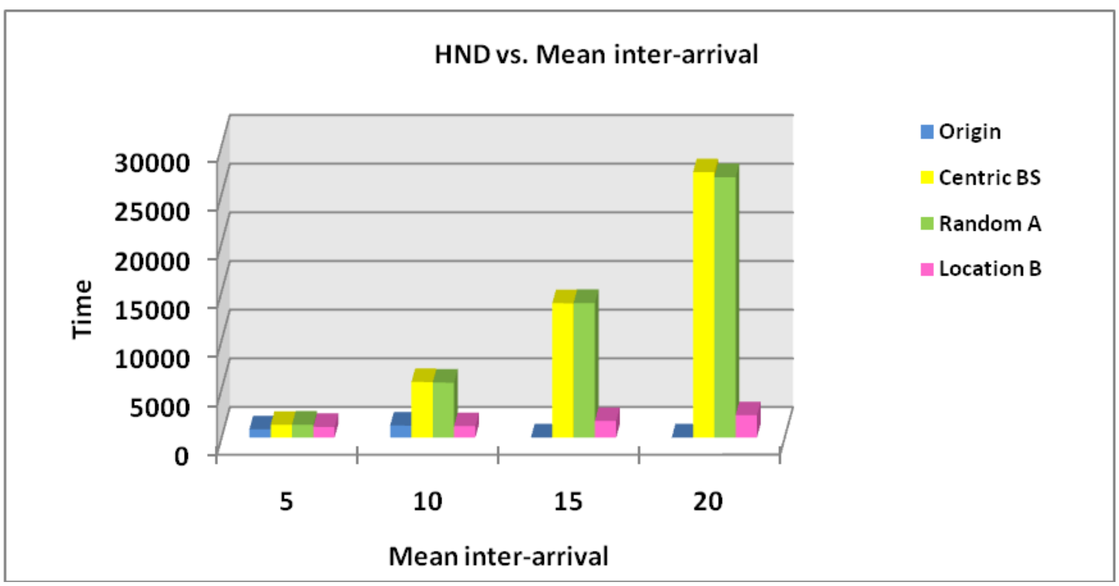


Fig. e.8: HND vs. Mean inter-arrival time

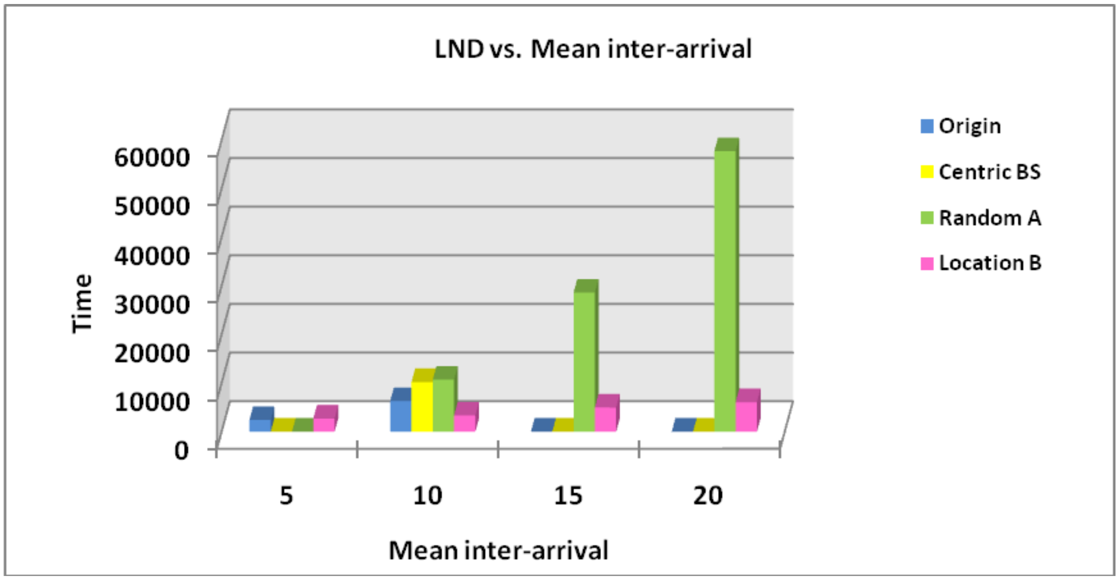


Fig. e.9: LND vs. Mean inter-arrival time

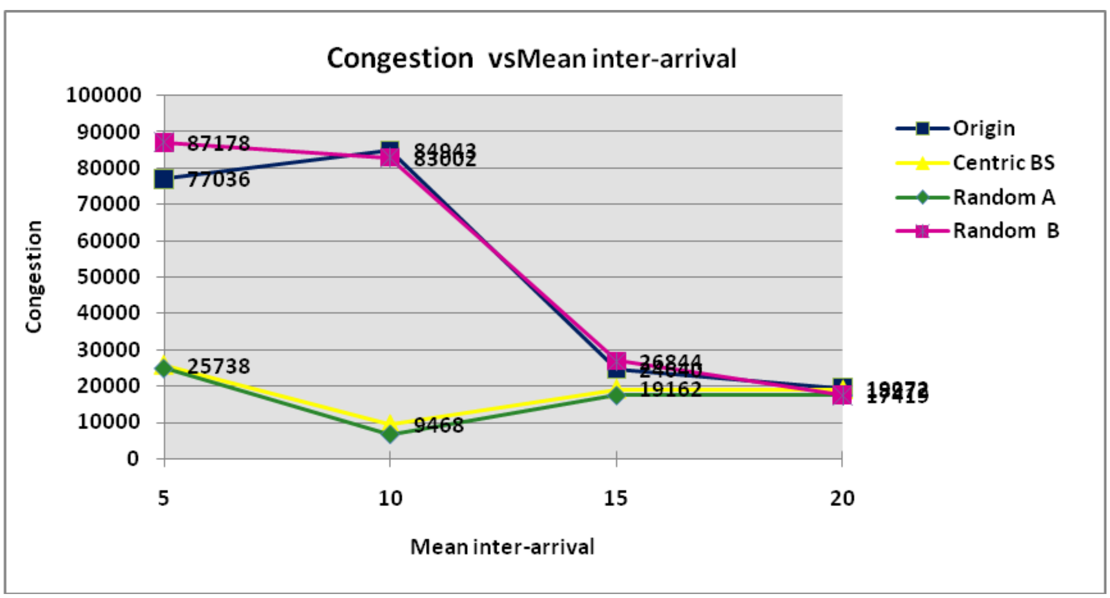


Fig. e.10: Congestion vs. Mean inter-arrival time

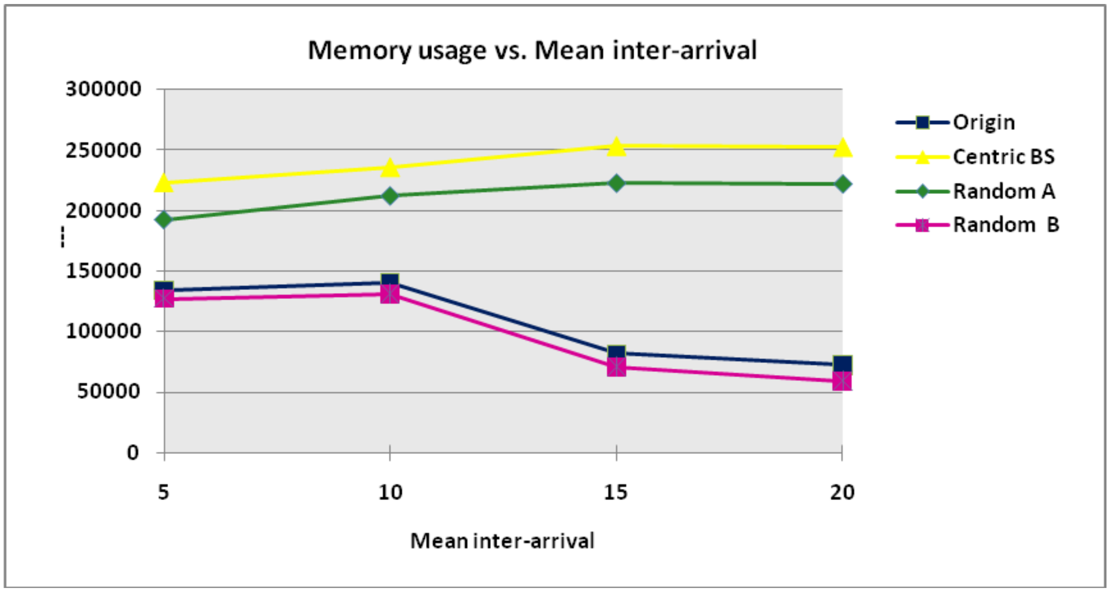


Fig. e.11: Memory usage vs. Mean inter-arrival time

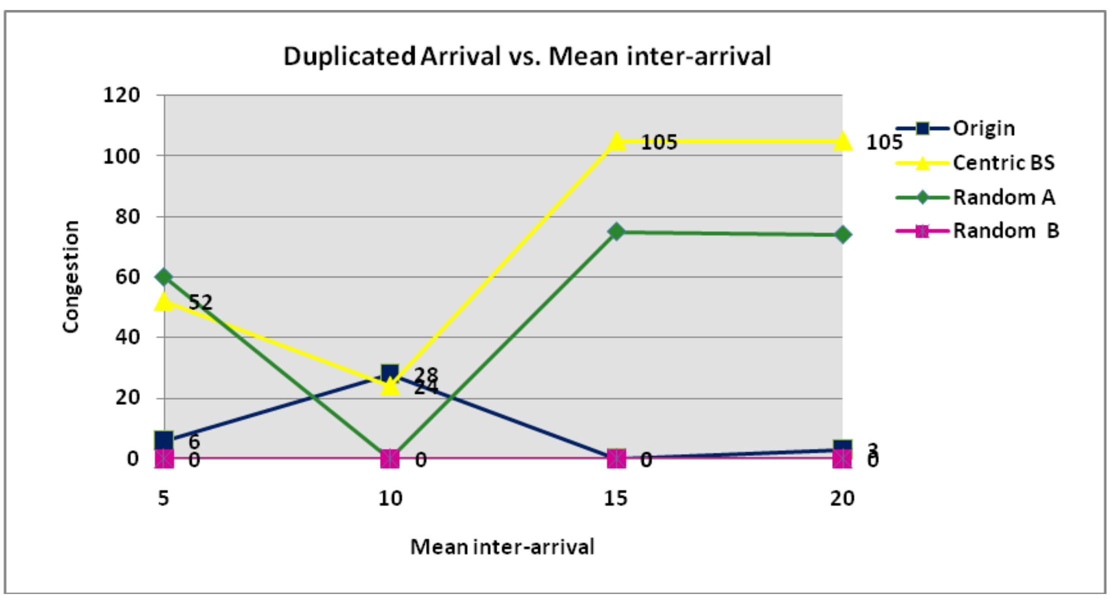


Fig. e.12: Duplicated Arrival vs. Mean inter-arrival time

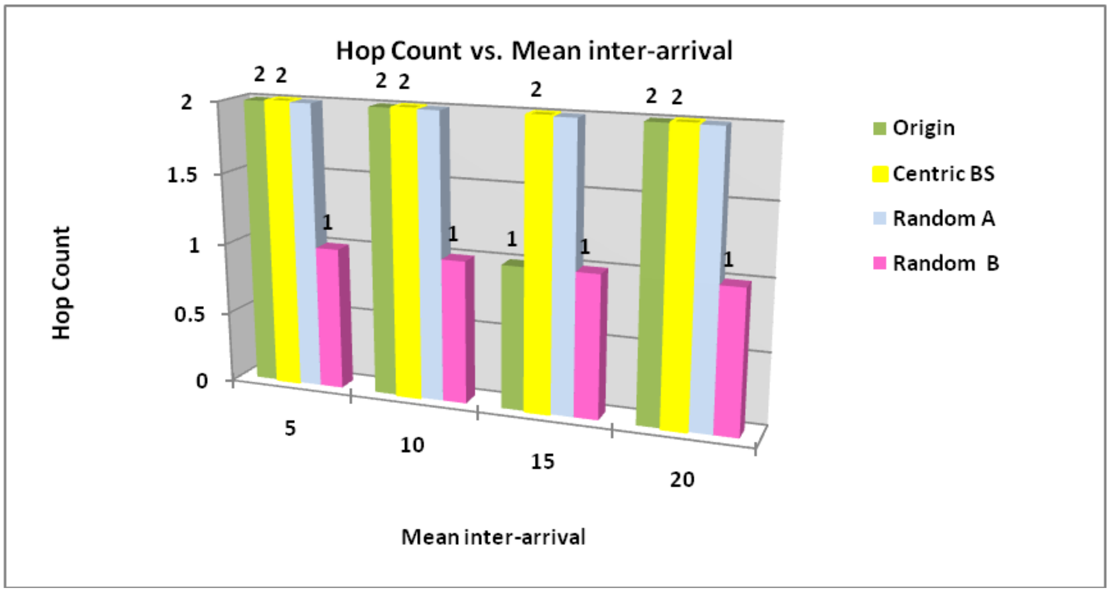


Fig. e.13: Hop count vs. Mean inter-arrival time

4.3.6. Varying number of nodes:

Network size is determined by number of nodes deployment area. We examine CEERA with different no. of nodes within network. We extracted results for network size of 30, 50, 75 and 100 nodes. Results are shown in next two tables and figures from f.1 to f.12.

4.3.6.1. No. of nodes vs. throughput, Delay, and DTJ:

Figures f.1, f.2, and f.3, show that increasing No. of nodes in network improves throughput for all networks as expected. However, it doesn’t have significant affect on delay and DTJ.

4.3.6.2. No. of nodes vs. energy dissipation, no. of died nodes, FND, BND, HND, and LND:

The effect of over these measures is shown in figures f.4 to f.9. Increasing no. of nodes in network maximizes network lifetime and also increases values of all these measures.

4.3.6.3. No. of nodes vs. storage occupation, duplicated arrival, Hop count and congestion:

Figures f.10, f.11, f.12 and f.13 show that congestion, duplicated arrival and memory sharing is affected by increasing no. of nodes in network. Introducing more no. of nodes allows more messages to be stored and thus more memory occupation. Also, the probability of conjunction and duplicated arrival are also increased.

4.3.6.4. Discussion on the variation of No. of nodes :

One of the most features required in any routing protocol is flexibility to accommodate changes in network size (scalability) and traffic load. Last section shows that CEERA is not affected negatively to the variation in network size. We plan to evaluate CEERA very large network size (thousands of nodes).We obtained the following proportional with no. of nodes in network:

- **No. of nodes (NN) and Throughput (Thr.):**

  $\mathbf{NN} \propto \mathbf{T}$……………….. for NN> 0 and NN < ∞

- **No. of nodes (NN) and network lifetime (LT.):**

  $\mathbf{NN} \propto$ ……………….. for NN> 0 and NN < ∞

- **No. of nodes (NN) and storage occupation (SO):**

  $\mathbf{NN} \propto \mathbf{S}$……………….. for NN> 0 and NN < ∞

Base station Location	Random A				Centric			
No. of nodes	30	50	75	100	30	50	75	100
Throughput	8603	14278	22901	30010	8240	11864	20549	26806
Delay	2.62	3.55	3.68	4.31	3.07972	4.92181	4.92967	4.73355
Delay Time Jitter	4.23	6.70	7.92	8.97	4.05343	6.96269	8.01306	7.90716
Total energy dissipated/initial Energy	93%	100%	100%	100%	93%	100%	100%	100%
No. of Died Nodes	26	48	74	99	27	49	74	99
FND	1693.53	1911.10	3044.03	4014.72	1990.23	2245.77	3399.11	4845.97
BND	1996.52	2762.72	4794.98	7109.00	3269.51	3824.13	6174.44	8391.61
HND	3293.81	4654.72	7237.51	10375.70	4123.97	5077.81	8379.96	11328.8
LND	0	0	12369.2	15830.9	0	6574.04	10640.8	13872.7
Congestion	0	83	756	1970	0	159	664	1068
Duplicated Arrival	135	19	174	577	518	605	829	1217
hop count /message	1	1	1	1	2	2	2	2
Occupied Memory (Sum)	33708	105934	236620	432700	38807	121814	269696	492488

Table f.1: Performance results of varying No. of nodes in network A and C.

Base station Location	Origin				Random B			
No. of nodes	30	50	75	100	30	50	75	100
Throughput	1988	3956	7778	8723	3970	4928	8740	10625
Delay	0.00009	9.01362	5.50506	8.13427	5.81345	8.09174	6.42461	6.13383
Delay Time Jitter	1.58E-11	13.3798	10.5681	14.007	8.37078	13.0956	13.1421	12.2799
Total energy dissipated/initial Energy	7%	100%	100%	100%	100%	100%	100%	100%
No. of Died Nodes	1	49	74	98	29	49	74	99
FND	14297.9	562.133	851.269	981.803	659.018	620.91	972.499	1199.78
BND	0	762.143	1143.97	1378.63	902.334	896.211	1325.52	1805.57
HND	0	970.418	1534.4	1882.66	1100.28	1170.37	1705.16	2293.89
LND	0	2945	4969.98	0	2708.32	3345.89	4969.21	6074.36
Congestion	0	37280	111420	250387	3037	36446	104448	210080
Duplicated Arrival	0	0	14	17	0	13	30	28
hop count /message	1	2	1	2	2	2	2	2
Occupied Memory (Sum)	0	73155	172880	296463	36503	77528	181838	327554

Table f.2: Performance results of varying No. of nodes in network B and O.

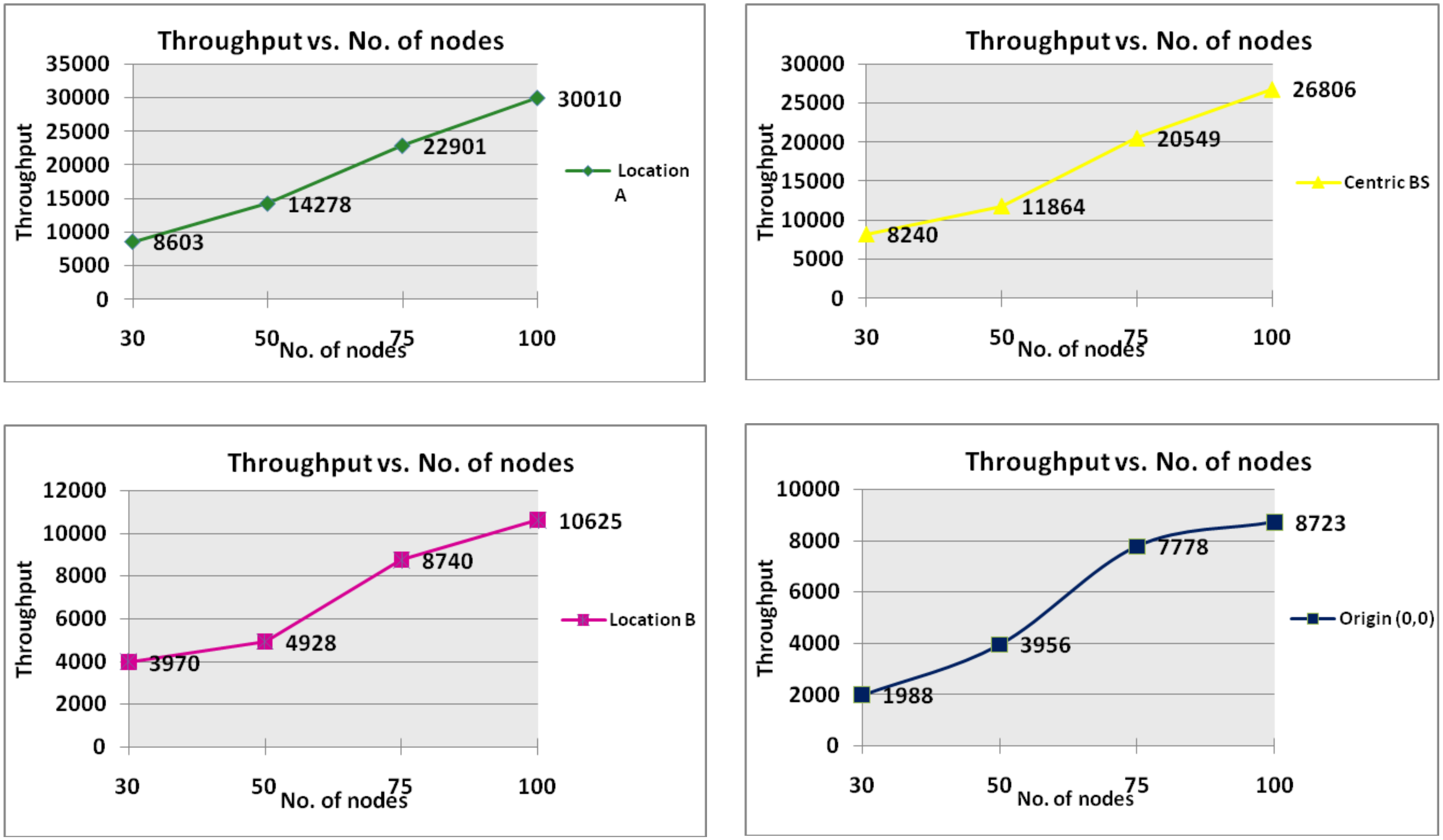


Fig. f.1: Throughput vs. No. of nodes for all networks

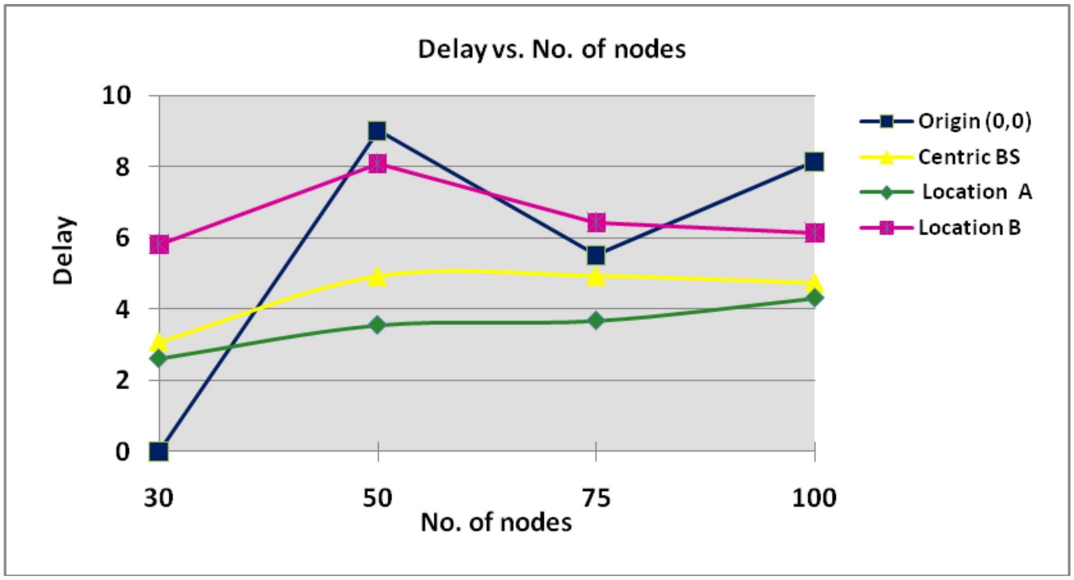


Fig. f.2: Delay vs. No. of nodes

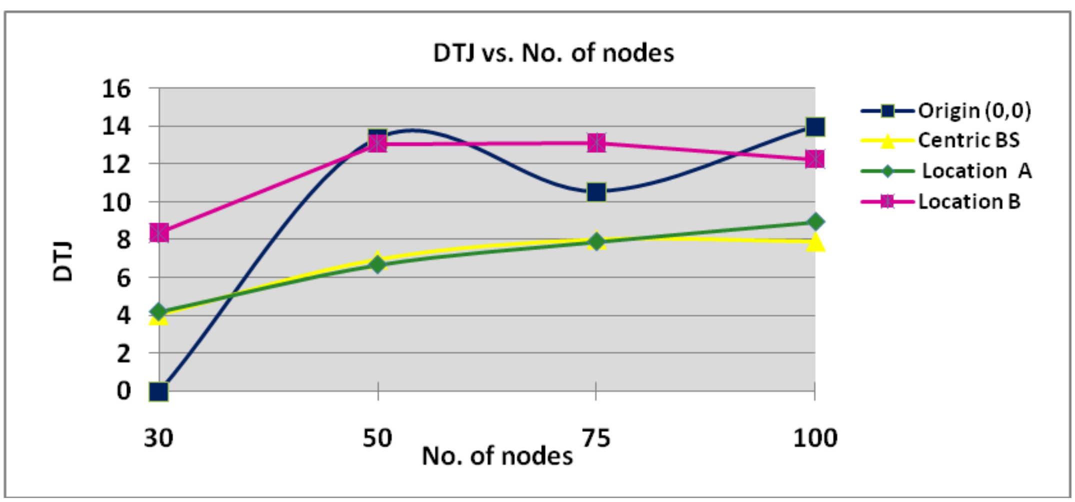


Fig. f.3: DTJ vs. No. of nodes

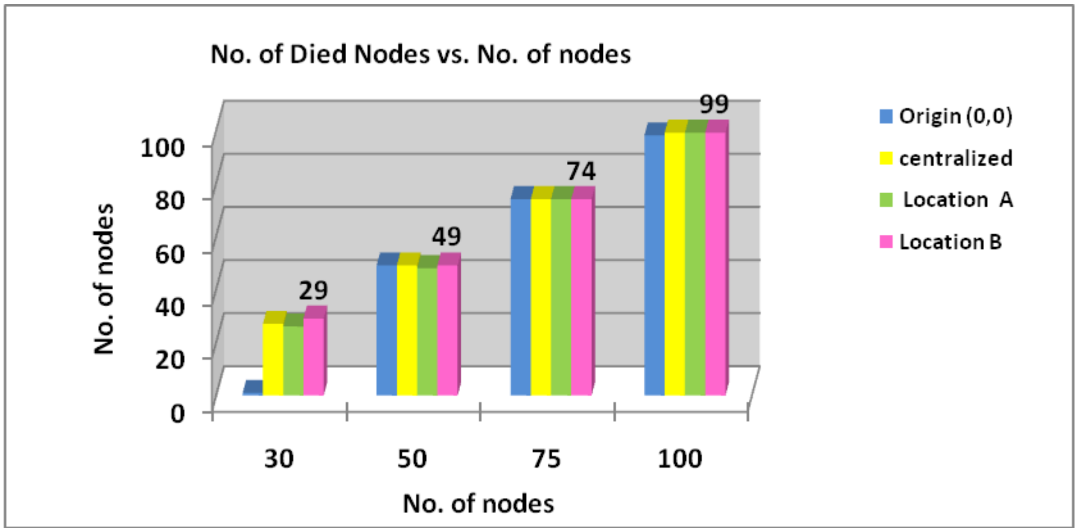


Fig. f.4: No. of died nodes vs. No. of nodes

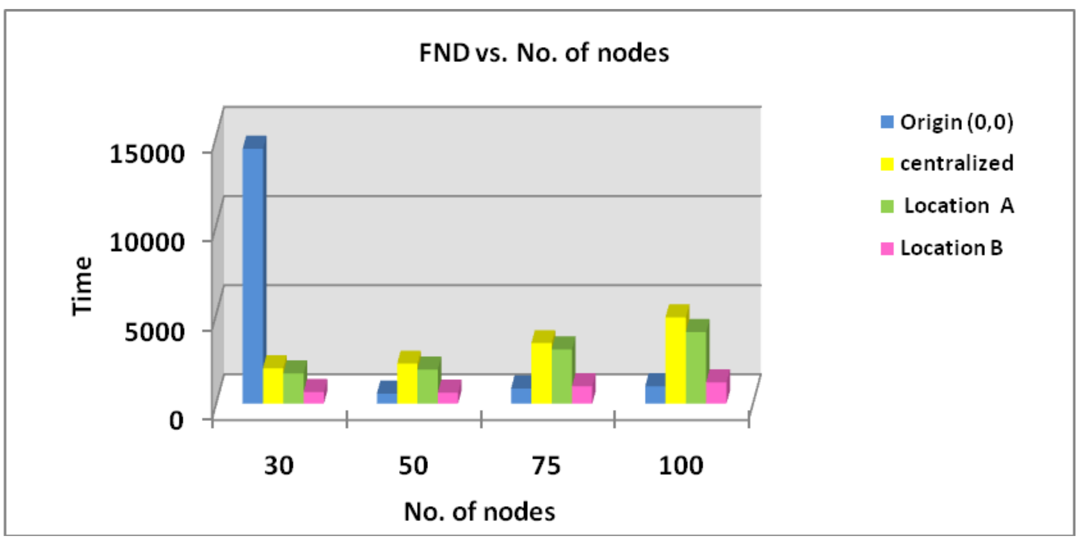


Fig. f.5: FND vs. No. of nodes

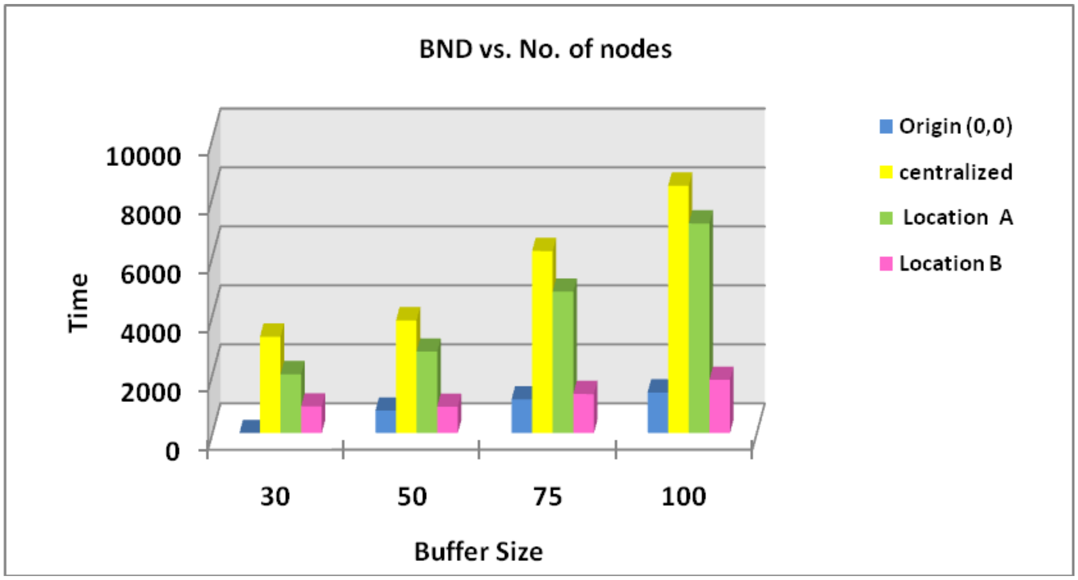


Fig. f.6: BND vs. No. of nodes

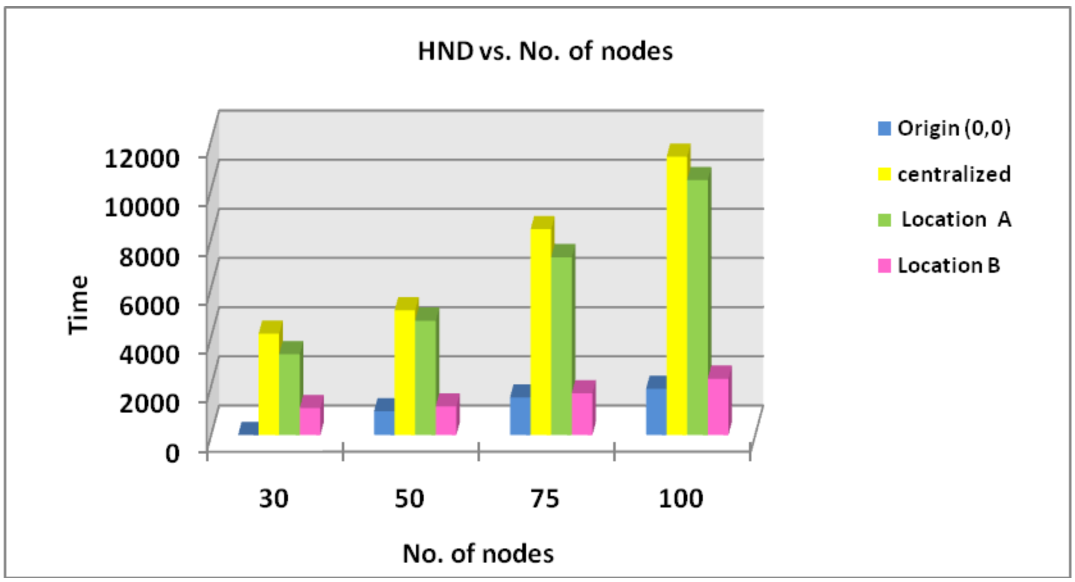


Fig. f.7: HND vs. No. of nodes

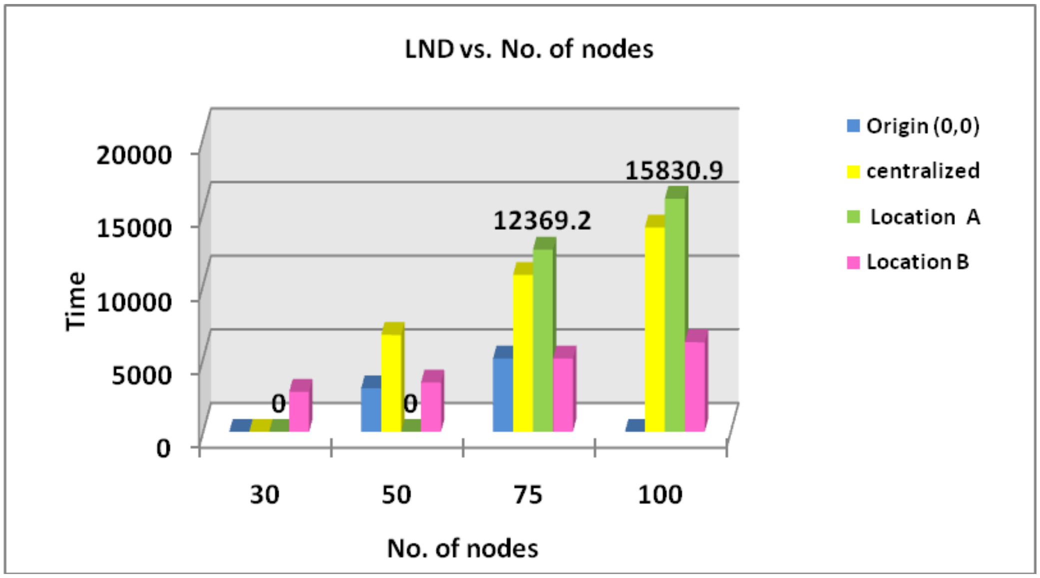


Fig. f.8: LND vs. No. of nodes

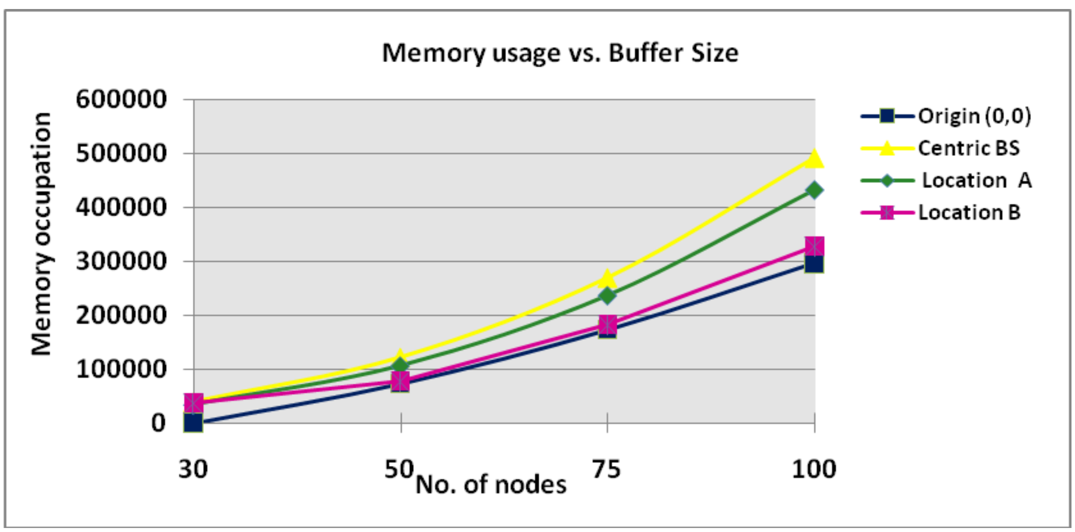


Fig. f.9: Memory occupation vs. No. of nodes

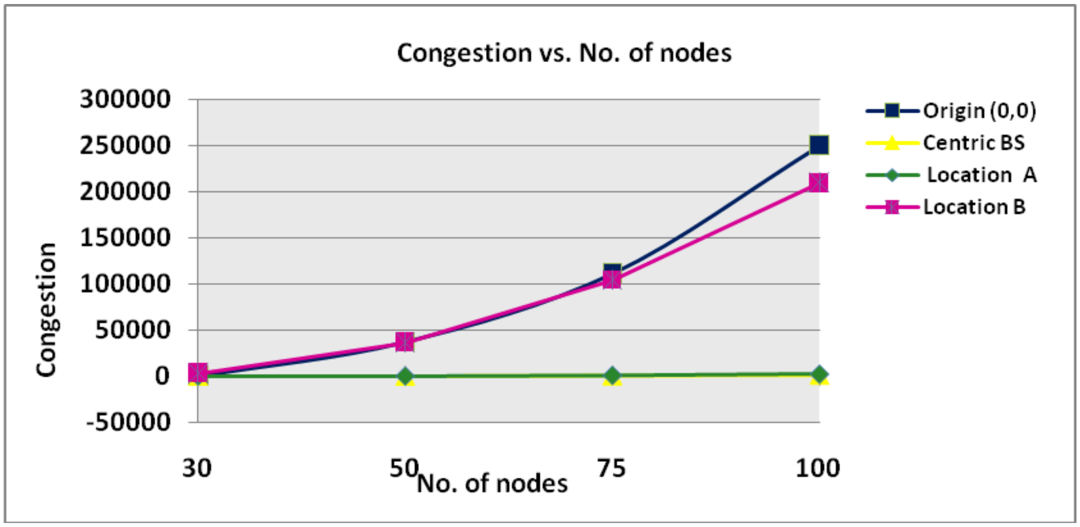


Fig. f.10: Congestion vs. No. of nodes

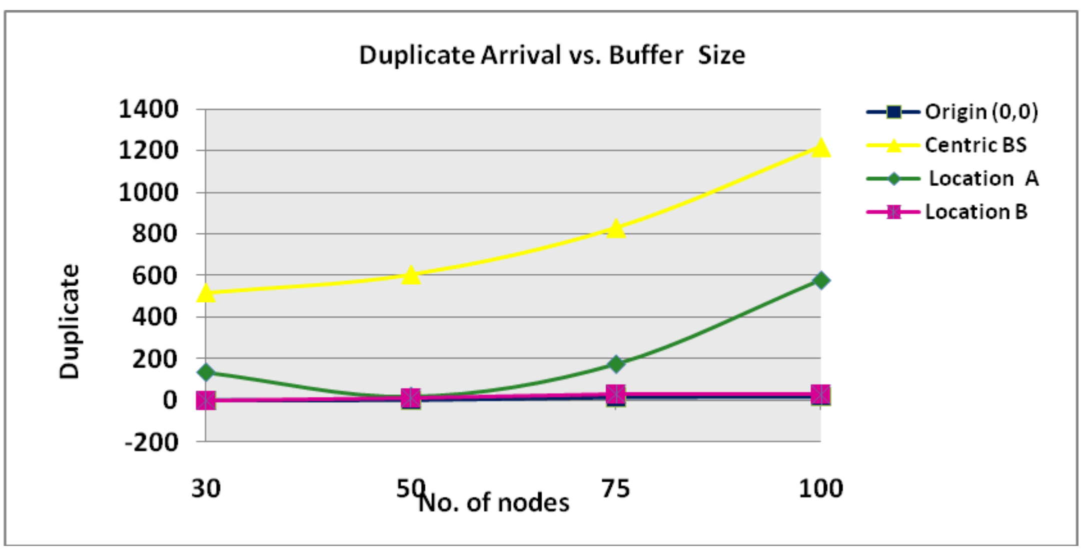


Fig. f.11: Duplicated arrival vs. No. of nodes

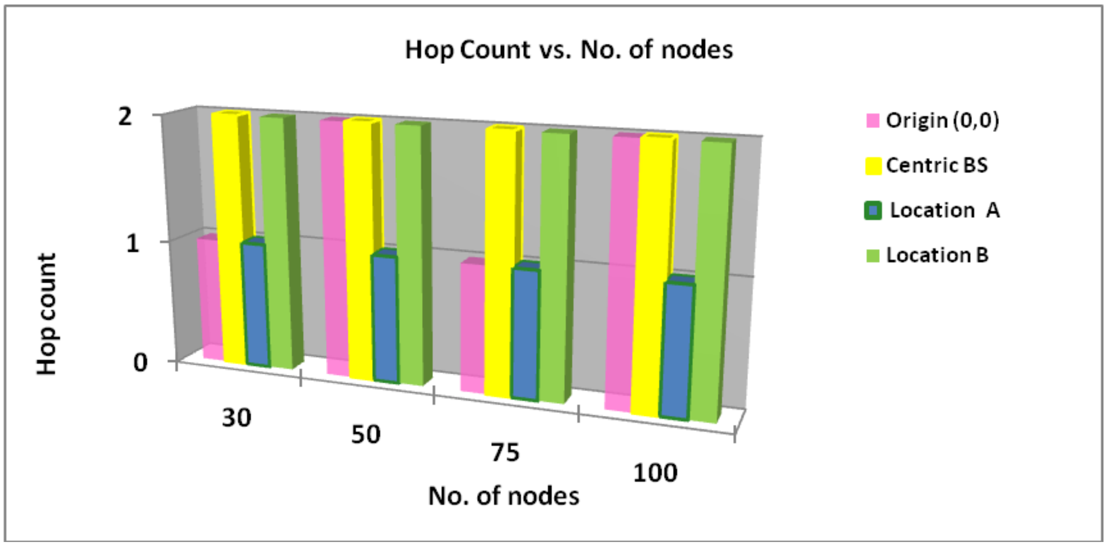


Fig. f.12: Hop count vs. No. of nodes

4.4. |Performance Comparison with other WSN Routing Protocols:

In this section, we present results of comparing CEERA to flooding and minimum transmission energy (MTE) routing protocol.

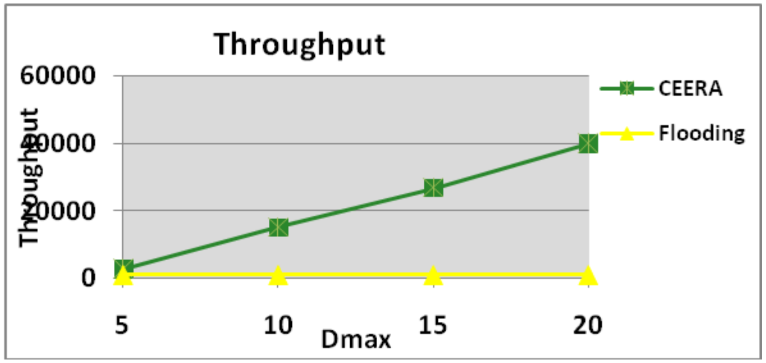
4.4.1. CEERA vs. Flooding

Flooding algorithm is one of the simplest algorithms for distributing data to every part of a connected network. In flooding, nodes act as both transmitter and receiver. The received message is forwarded by the node to every neighbor except the source. There are several disadvantages with this approach of routing. Flooding is wasteful in term of energy and network bandwidth [2].

We implemented flooding and compared it to CEERA. Simulation results show that CEERA achieves over a factor of 15, 27, and 41 reduction in energy dissipation compared to flooding for different BSs' locations. Results also show that CEERA uses a very small no. of hops compared to flooding. Details results are shown in next table and figures g.1 to g.10.

Routing Protocol	CEERA				Flooding			
Dmax	5	10	15	20	5	10	15	20
Throughput	2715	15147	26806	40000	984	973	969	973
Delay	51.8135	14.0662	4.73355	0.986106	0.000690864	0.000399	0.000232	0.000163
Delay Time Jitter	41.9125	19.1394	7.90716	2.08838	0.000393391	0.000242	0.000125	9.32E-05
Total energy dissipated/initial Energy	53%	100%	100%	78%	51%	100%	100%	100%
No. of Died Nodes	50	99	99	27	50	99	99	99
FND	2012.49	2094.39	4845.97	11084.6	837.657	454.67	459.236	463.061
BND	2518.6	4098.71	8391.61	16375.1	944.058	459.237	464.788	468.726
HND	3628.22	5732.47	11328.8	0	2566.32	472.467	471.75	474.416
LND	17800.7	10208.6	13872.7	0	0	583.204	519.496	505.313
Duplicated Arrival	0	24	1217	1817	1850	14844	28172	55135
hop count /message	5	2	2	1	15	61	87	150

Table g.1: Simulation results for both CEERA and flooding.



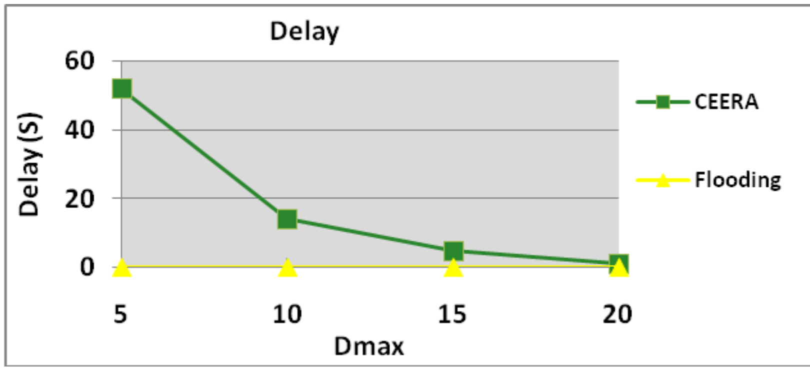


Fig g.1: throughput vs. Dmax in CEERA and flooding

Fig g2: Delay vs. Dmax in CEERA and flooding

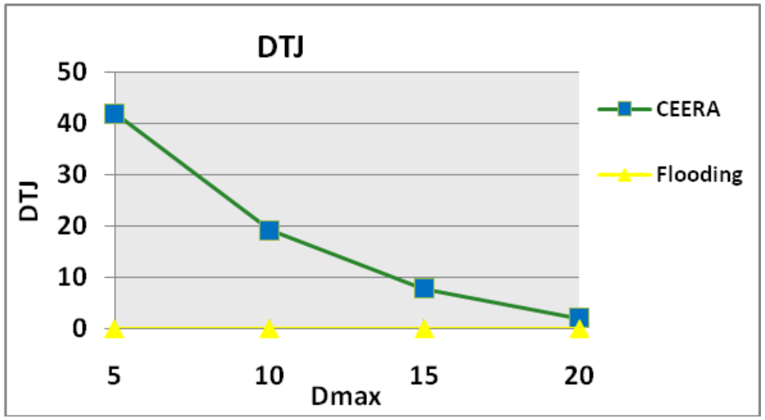


Fig g.3: DTJ vs. Dmax in CEERA and flooding

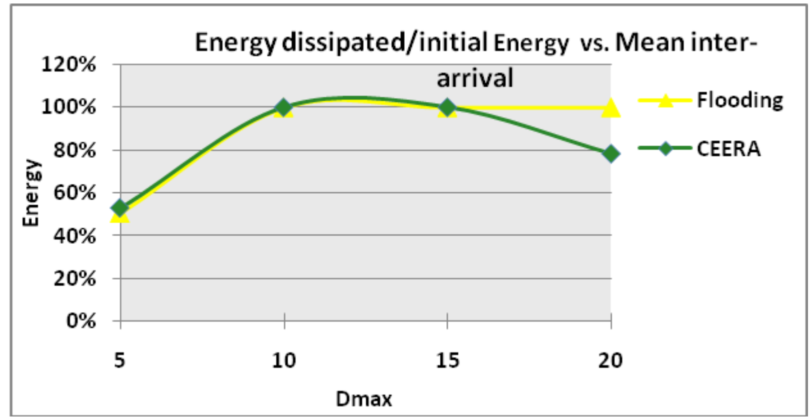


Fig g4: Energy dissipation vs. Dmax in CEERA and flooding

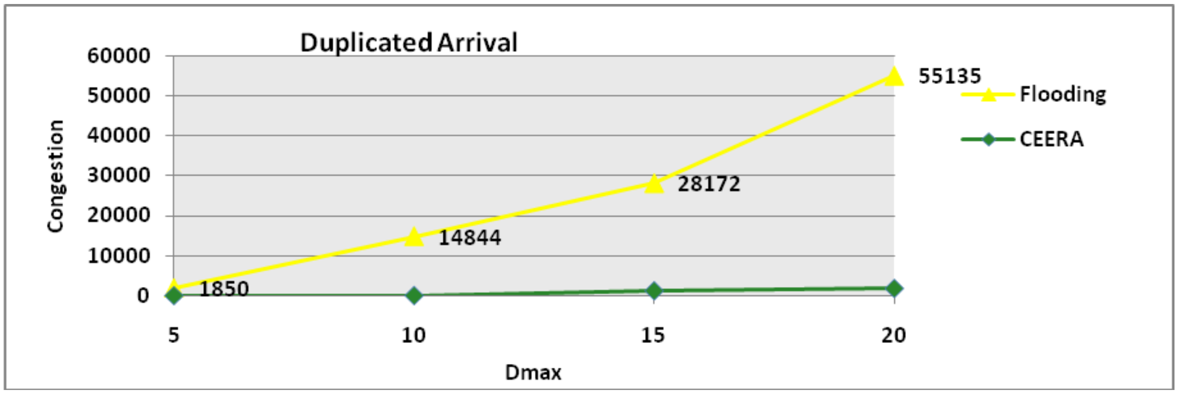


Fig g.5: duplicated arrival vs. Dmax both in CEERA and flooding

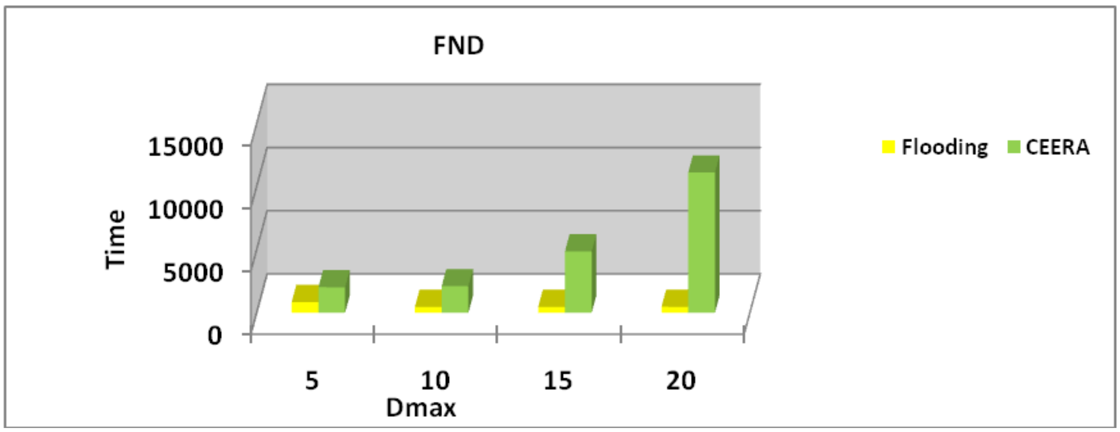


Fig g.6: FND vs. Dmax both in CEERA and flooding

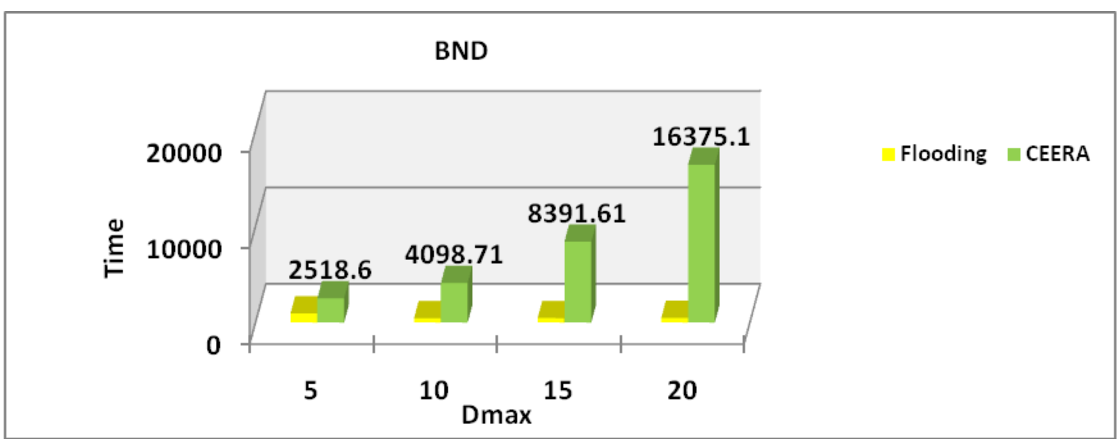


Fig g.7: BND vs. Dmax both in CEERA and flooding

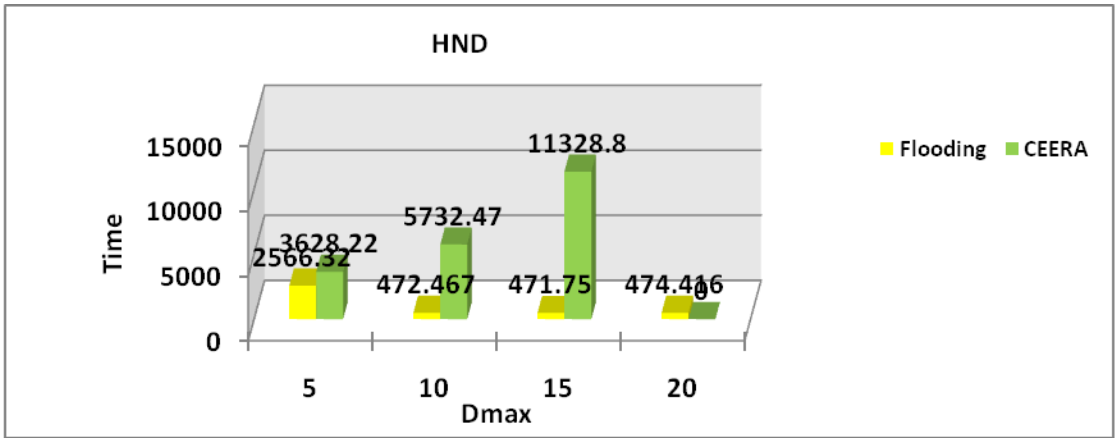


Fig g.8: HND vs. Dmax both in CEERA and flooding

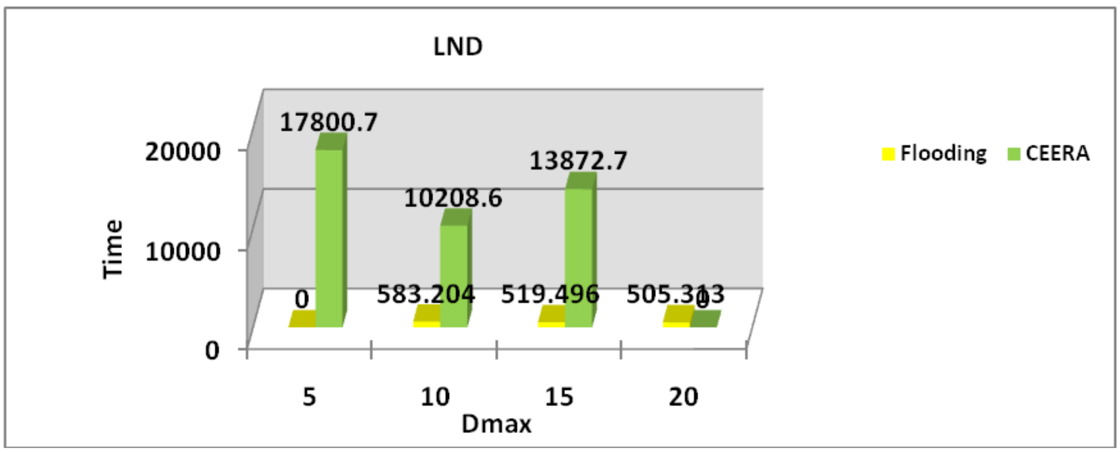


Fig g.9: LND vs. Dmax both in CEERA and flooding

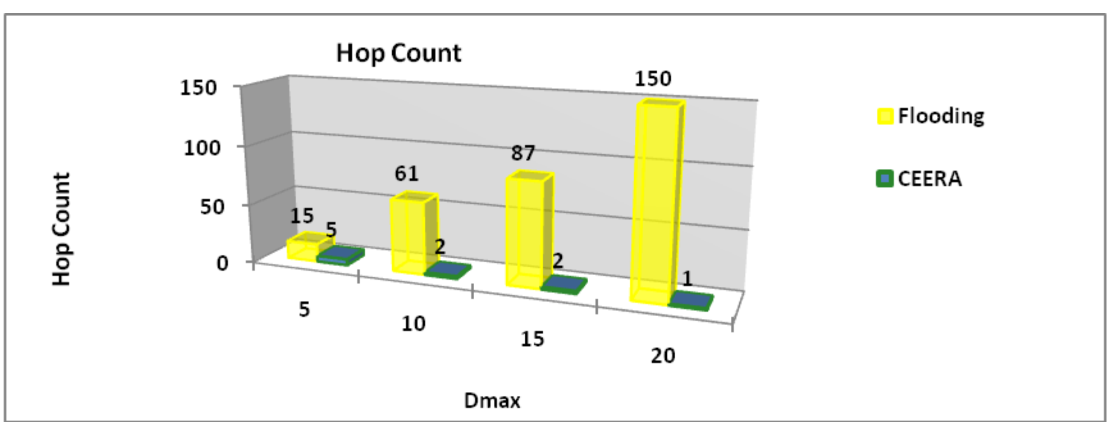


Fig g.10: Hop count vs. Dmax both in CEERA and flooding

4.4.2. CEERA vs. Minimum Transmission Energy Routing

For Minimum Transmission Energy (MTE) routing, routes from each node to the base station were chosen such that each node's next-hop neighbor is the closest node that is in the direction of the base station. Each node requires 100 nJ to determine their next-hop neighbor4. Nodes adjust their transmit power to the minimum required to reach their next-hop neighbor [20].

This reduces interference with other transmissions and reduces the nodes' energy dissipation. When a node receives data from one of its upstream neighbors, it forwards the data to its next-hop neighbor. This continues until the data reaches the base station[]20.

We implemented MTE routing and compared it to CEERA. Simulation results show that CEERA achieves over a factor of 1.34 and 1.26 reduction in energy dissipation compared to MTE for some BSs' locations. Results also show that CEERA uses a very small no. of hops compared to MTE. Details results are shown in next table and figures h.1 to h.9.

Routing Protocol	CEERA			MTE		
Base station Location	A	B	O	A	B	O
Throughput	4882	4978	3584	6747	3705	2850
Delay	7.97343	3.43494	3.30325	0.0136281	0.014296	0.014443
Delay Time Jitter	10.8377	8.70264	7.73493	0.00209826	0.002662	0.001736
Total energy dissipated/initial Energy	97%	100%	100%	100%	100%	100%
No. of Died Nodes	80	100	105	97	98	98
FND	1093.56	253.062	240.894	364.549	329.484	260.741
BND	1579.62	313.831	295.969	1488.48	898.616	748.085
HND	2061.19	384.608	358.191	2250.38	1184.81	1057.04
LND	0	4668.31	3772.8	0	0	0
hop count per message	3	1	2	1	2	1

Table h.1: Simulation results for both CEERA and MTE.

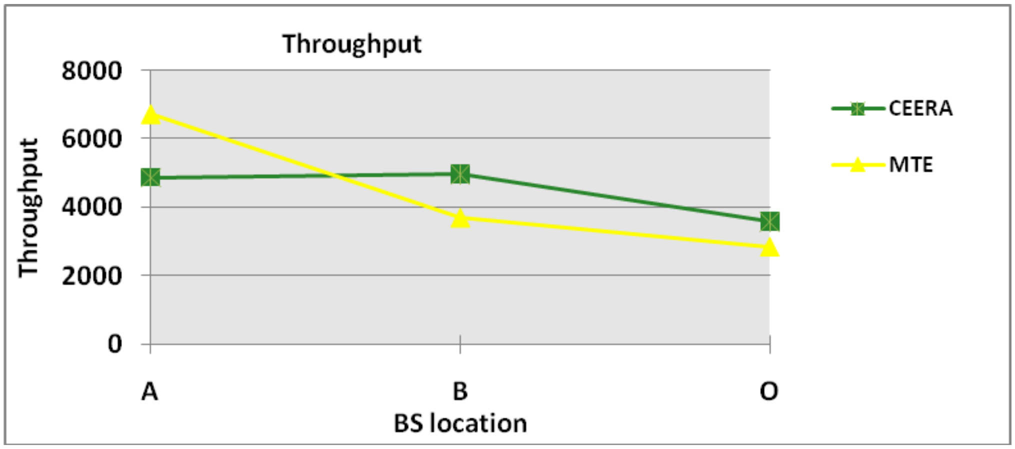


Fig. h.1: Throughput in both CEERA and MTE.

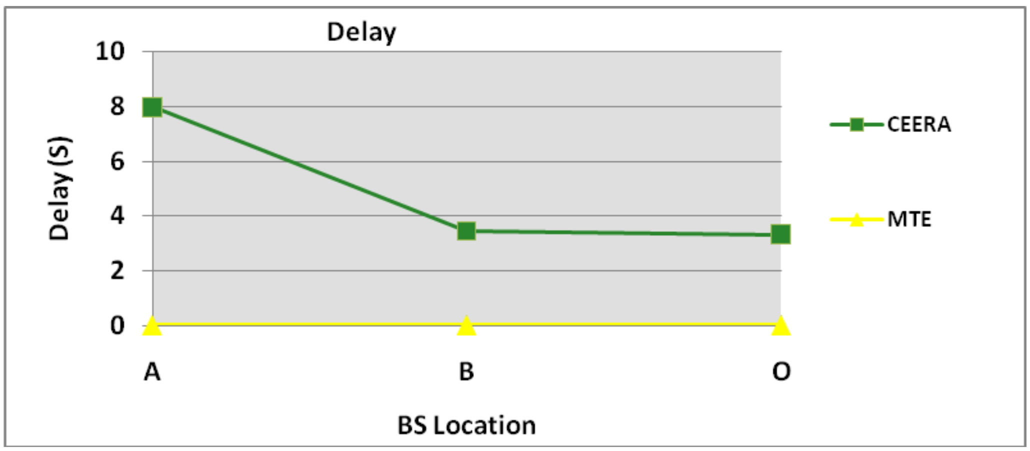


Fig. h.2: Delay in both CEERA and MTE.

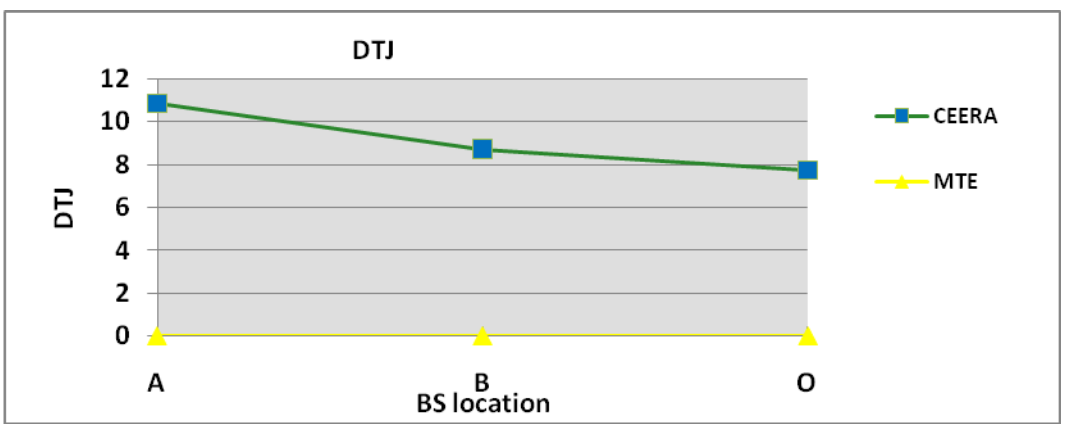


Fig. h.3: DTJ in both CEERA and MTE.

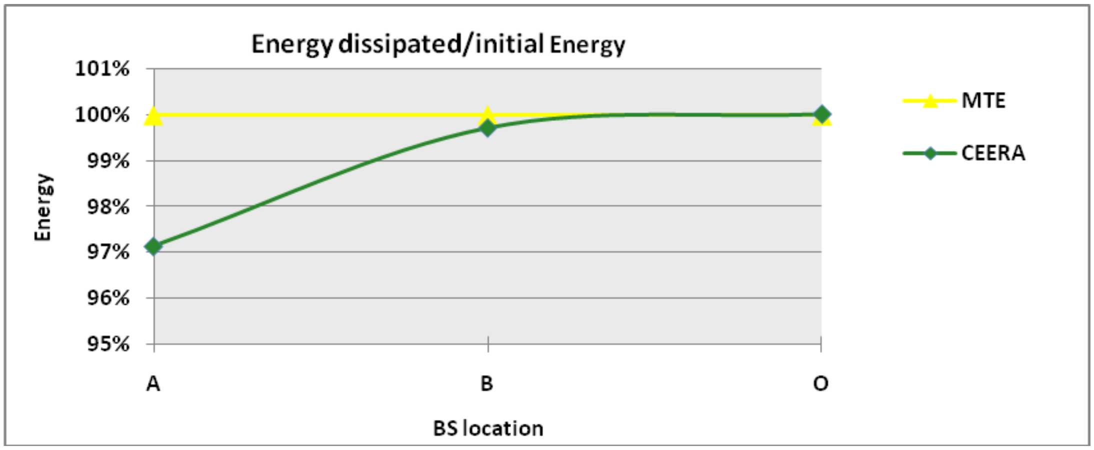


Fig. h.4: Dissipated energy in both CEERA and MTE.

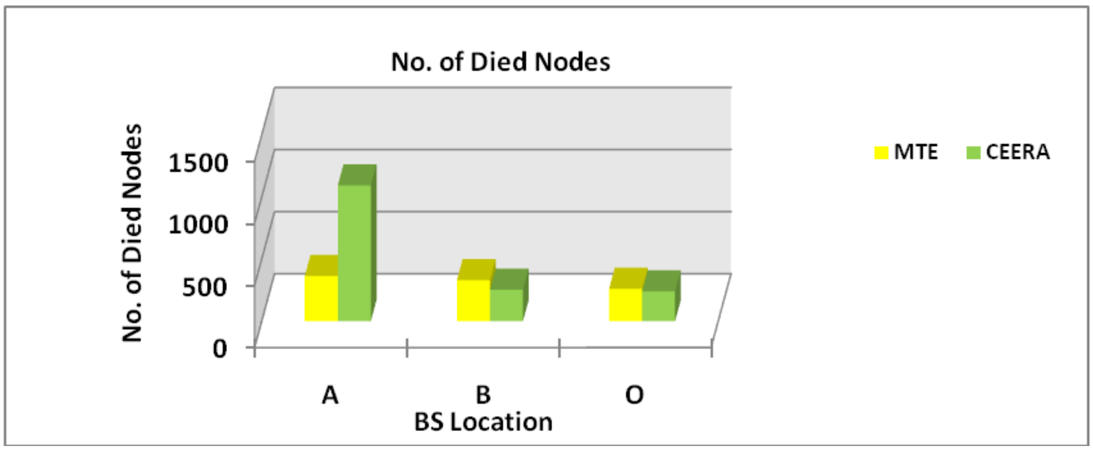


Fig. h.5: No of died nodes in both CEERA and MTE.

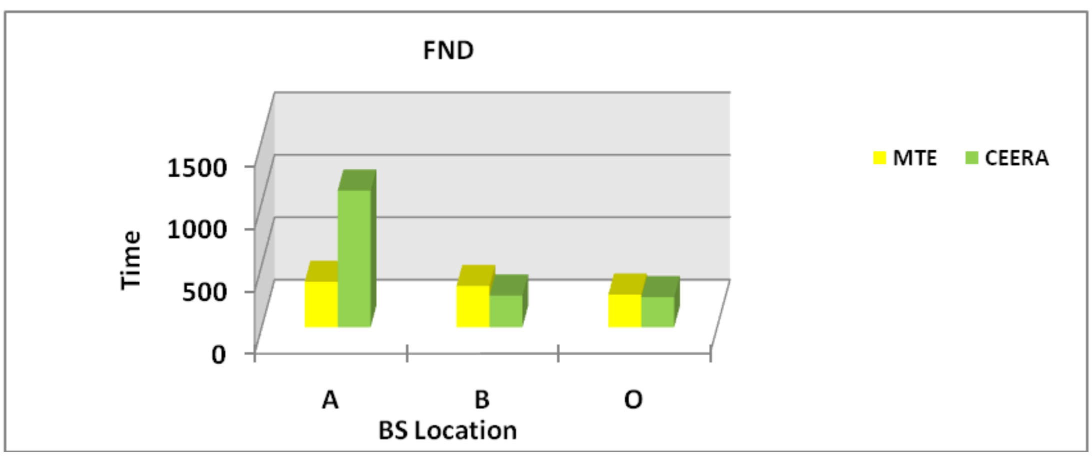


Fig. h.6: FND in both CEERA and MTE.

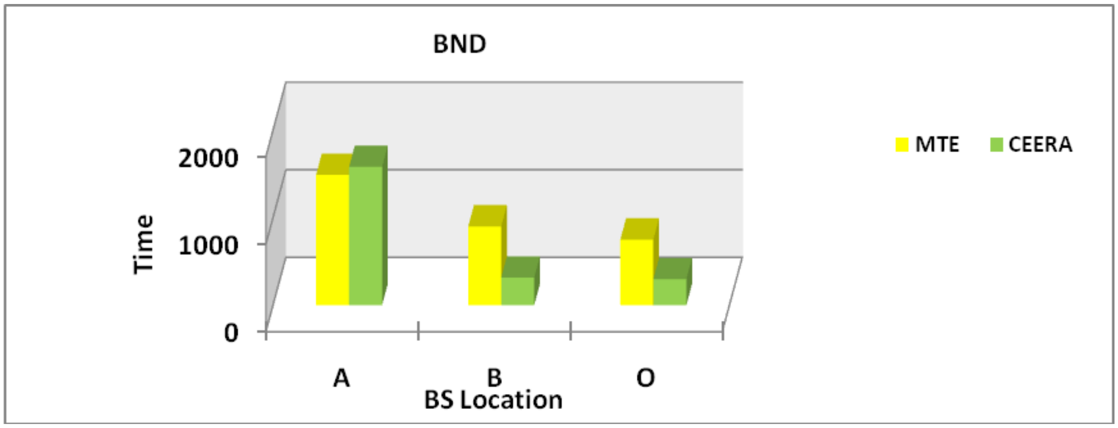


Fig. h.7: BND in both CEERA and MTE.

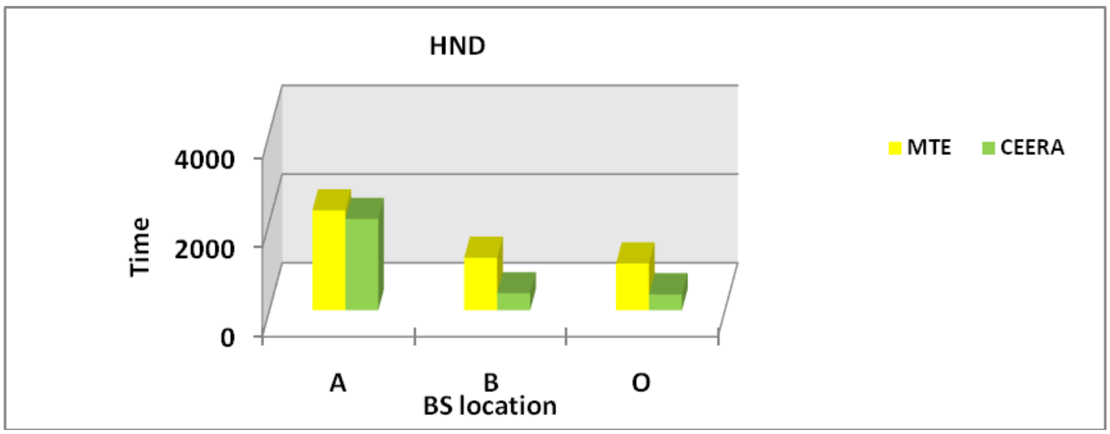


Fig. h.8: HND in both CEERA and MTE.

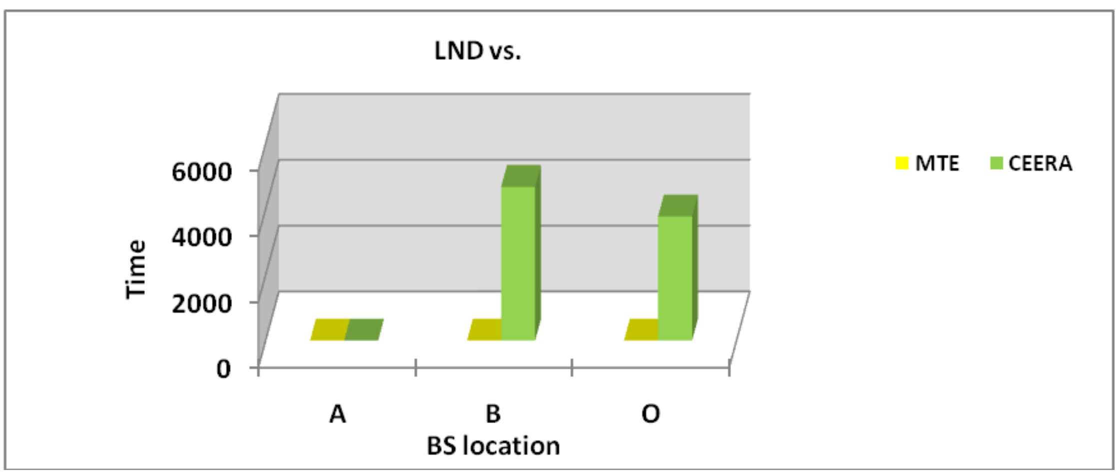


Fig. h.9: LND in both CEERA and MTE.

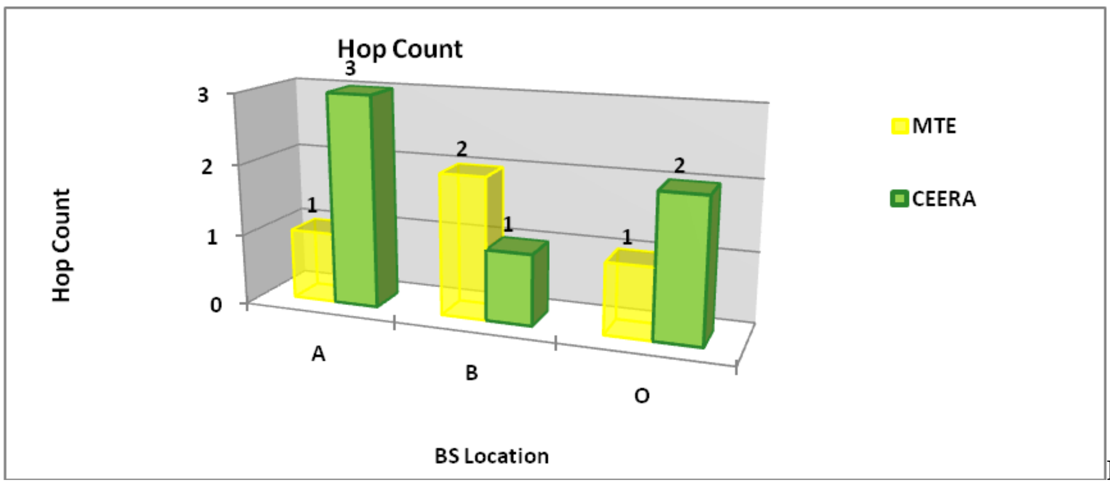


Fig. h.10: Hop count in CEERA and MTE.

CHAPTER 5: Conclusion

5.0. Conclusion

Routing in sensor networks is a promising research area. Applications of WSN show how it is important to design protocols and algorithms for wireless networks to be bandwidth and energy efficient. In this project, we proposed, designed and implemented an energy aware routing algorithm for wireless sensor networks. We designed CEERA with the following features to meet the design criterion we concentrated on:

- Ease of deployment: In order to ensure that the nodes can be easily deployed in remote, hostile, or difficult areas.
- Minimum-configuration: In CEERA, no prior knowledge or configuration is needed to make routing decision. A node determines wither to cooperate in retransmission the received message or to discard it locally, without exchanging more control messages to provide routing information. The only used control message is a small size BS ACK message.
- Minimum delay: Minimizing the latency caused by CEERA by selecting appropriate scalar factor value.
- Utilizing the available storage: we assumed sensors are equipped with average memory size. We introduced a factor, called duplication factor, to manage the use of this area.

Furthermore, we studied the performance of the designed algorithm extensively through simulation. For this, we have developed and implemented a suitable simulation model. Simulation shows a significant improvement in energy usage in routing with CEERA. CEERA outperforms Flooding 15, 27, and 41 times. Also, CEERA achieves over a factor of 1.34 and 1.26 reduction in energy dissipation compared to MTE for some BSs' locations. The research described in this project contributes to my understanding of the benefits of designing routing protocols. We evaluated protocol performance with variable factors.

5.1.Future Work:

There is still much work to be done in the area of protocols for wireless sensor networks. This project can be followed by some related studies. Some of them are listed below:

1. Optimizing CEERA’s by introducing zero scalar factor for key nodes
2. Finding a method for optimal factors selection and evaluation.
3. Incorporating data aggregation to minimize energy dissipation.

References:

[1] Krishnendu Chakrabarty and S. S. Iyengar, "Scalable Infrastructure for Distributed Sensor Networks", Springer, London, 2005

[2]Jamal N. Al-Karaki, Ahmed E. Kamal, "Routing Techniques In Wireless Sensor Networks: A Survey", *IEEE Wireless Communications* December 2004

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "*A Survey on Sensor Networks*," IEEE Communications Magazine, vol. 40, no. 8, pp. 102-116, August 2002.

[4]Ying Liang; Haibin Yu, "Energy Adaptive Cluster-Head Selection for Wireless Sensor Networks", Proceeding of the sixth International Conference on *Parallel and Distributed Computing, Applications and Technologies, (PDCAT'05), IEEE, 2005.*

[5] J. Kulik, R. Wendi, and H. Balakrishnan, "Adaptive Protocols for Information Dissemination in Wireless Sensor Networks", Washington, pp. 174 -185, 1999 ISBN:1-58113-142-9

[6] R.C. Doss and D. Chandra,"Reliable Event Transfer in Wireless Sensor Networks Deployed for Emergency Response", From Proceeding of Parallel and Distributed Computing and Systems, (Australia), ACTA press, 2005

[7] Shameem M. S. Sattar, "A Survey on Routing Protocols in Sensor Networks", Communication Networks Laboratory, November 2004

[8] Anantha Chandrakasan, R. Amirtharajah, S. Cho, J. Goodman, G. Konduri, J. Kulik, W. Rabiner, Alice Wang "Design Considerations for Distributed Microsensor Systems", Proceedings of the IEEE 1999, Custom Integrated Circuits, San Diego, 1999, pp. 279-286

[9] Malik Tubaishat and Sanjay Madria "Sensor networks: an overview", appeared in IEEE POTENTIALS, APRIL/MAY 2003

[10] Brad Karp and H. T. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM'00), ACM Press, N. Y., 2000.

[11] Wendi R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol forWireless Microsensor Networks". In Proceedings of the 33rd Hawaii International Conference on System Sciences – 2000, IEEE, 2000.

[12] Rui Teng, "RBR: A Region-Based Routing Protocol for Mobile Nodes in Hybrid Ad Hoc Networks", *IEEE Communications Magazine, November 2006.*

[13] G. Lukachan and M. Labrador," SELAR: Scalable Energy-Efficient Location Aided Routing Protocol for Wireless Sensor Networks", Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04), IEEE, 2004

[14] S. Olariu, Q. Xu and A. Y. Zomaya, "An Energy-Efficient Self-Organization Protocol for Wireless Sensor Networks", IEEE, 2004

[15] W. Fang, Y. Liu, and D. Qian, "EDDS: An Efficient Data Delivery Scheme for Address-free Wireless Sensor Networks", Proceedings of the Sixth International Conference on Networking (ICN'07), IEEE, 2007

[16] Liang Zhao' and Qilian Liang', "DISTRIBUTED AND ENERGY EFFICIENT SELF-ORGANIZATION FOR ON-OFF WIRELESS SENSOR NETWORKS", IEEE, 2004

[17] William Stallings, "A Practical Guide to Queuing Analysis", BYTE, February 1991

[17] Averill M. Law and W. David Kelton," Simulation Modeling and Analysis" McGraw-Hill, third edition, 2000.

[18] Gandham, S.R.; Dawande, M.; Prakash, R.;and Venkatesan, S, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations", Proceeding for

Global Telecommunications Conference, 2003. GLOBECOM apos;03. IEEE, Dec. 2003, pp. 377 - 381 Vol.1

[19] random.org, “ www.random.org”, Irelands, last accessed in February 1, 2008.

[20] Wendi Beth Heinzelman “Application-Specific Protocol Architectures for Wireless Networks; PhD. thesis”, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 2000.

[21] Ward Whitt,“ANALYSIS FOR THE DESIGN OF SIMULATION EXPERIMENTS”, Columbia University, New York, January, 2005