

# An Efficient and Reliable Core-Assisted Multicast Routing Protocol in Mobile Ad-Hoc Network

Faheem khan

Department of Computer Science  
University of Malakand  
KPK, Pakistan

Sohail Abbas

Department of Computer Science  
University of Malakand  
KPK, Pakistan

Samiullah Khan

Department of Computer Science  
Agriculture University KPK,  
Pakistan

**Abstract**—Mobile ad-hoc network is a collection of mobile nodes that are connected wirelessly forming random topology through decentralized administration. In Mobile ad-hoc networks, multicasting is one of the important mechanisms which can increase network efficiency and reliability by sending multiple copies in a single transmission without using several unicast transmissions. Receiver initiated mesh based multicasting approach provides reliability to Mobile ad-hoc network by reducing overhead.

Receiver initiated mesh based multicast routing strongly relies on proper selection of a core node. The existing schemes suffer from two main problems. First, the core selection process is not efficient, that usually selects core in a manner that may decrease core lifetime and deteriorate network performance in the form of frequent core failures. Second, the existing schemes cause too much delay for core re-selection(s) process. The performance becomes worse in situations where frequent core failures occur due to high mobility which causes excessive flooding for reconfigurations of another core and hence delays the on-going communication and compromising the network reliability.

The objectives of the paper are as follows. First, we propose an efficient method in which the core is selected within the receiver group on the basis of multiple parameters like battery capacity and location, as a result, a more stable core is selected with minimum core failure. Second, to increase the reliability and decrease the delay, we introduce the idea of the mirror core. The mirror core takes the responsibility as a main core after the failure of the primary core and has certain advantages such as maximum reliability, minimum delay and minimizing the data collection process. We implement and evaluate the proposed solution in Network Simulator 2. The result shows that this scheme performs better than the existing benchmark schemes in terms of the packet delivery ratio, overhead and throughput.

**Keywords**—MANET; Core; Mirror core; Multicast routing; Receiver initiated; Mesh based routing; NS2

## I. INTRODUCTION

Mobile ad hoc network (MANET) is an infrastructure-less network of mobile nodes with decentralized administration and dynamic topology. Due to its infrastructure-less nature, these networks are tempted to be deployed in places where there is no pre-deployed infrastructure or where there is costly to deploy one. Hence, MANETs can be used in various situations such as jungles, mountains, deserts, nuclear disaster, in a mining field clearance, military operation and communication, battlefield, earthquake scenario, etc. where no infrastructure exist [1]. In MANETs, unlike wired networks, there are no

dedicated routers for packet routing and forwarding. The MANET has a limited transmission range and all the nodes in the network strongly depend on intermediate nodes during data forwarding in multi-hop scenarios and hence, all the nodes in a network act as a host as well as router [2, 3]. Routing is an important function of the network. Routing can be of three types: unicast, broadcast and multicast routing. In unicast routing, data communication occurs in a one-to-one manner and only two nodes exchange their information with each other and in broadcasting the data communication occurs in a one-to-all fashion [4]. However, multicast routing works in one-to-many fashion and efficiently maintains the group communication by sending the similar copies of the same message to multiple nodes with a single transmission. In case of transmitting the similar data through several unicasts, multicasting minimizes channel capacity consumption, routing processes, energy utilization and end-to-end delay [5]. There are many applications of multicast routing [6, 7] such as armed forces operations and communications from one commander to a group/platoon, boss to subordinate communication, distance learning, information dissemination from air drones to a group of soldiers and presentation at the same time in different meeting rooms [8].

In MANETs, multicast routing protocols can be divided into tree based and mesh based routing protocols. In tree based multicast routing protocols, there is only one route between a sender and a receiver and is not robust against regular topology changes. However, it is well suited for environments where mobility is low [9]. Example of tree based multicasting are ad hoc multicast routing protocol utilizing increasing id numbers (AMRISs) [10], multicast ad hoc on-demand distance vector (MAODV) [11]. On the other hand, several paths are maintained from a source to the receivers in mesh based multicast routing. These multiple routes from source to all receivers give robustness, reliability and reduced latency at the cost of extra overhead as compared to the tree based multicast routing.

Mesh based multicast routing is further divided into sender initiated and receiver initiated routing protocols. In sender initiated approach every sender behaves as a core and it is the sender that initiates the mesh formation, maintains and updates the multicast paths to the receivers. Therefore, when the number of sources increases within a multicast group, the maintenance of the group becomes costly in terms of communication overhead. Example of sender initiated routing protocols are dynamic core based multicast routing protocol

(DCMP) [12] and on demand multicast routing protocol (ODMRP) [13]. Whereas, in a receiver initiated approach, one core is selected for the receiver group and it is the responsibility of the core node to maintain and update the receiver group. In situations where the number of receivers or sources increase, the receiver initiated protocols does not deteriorate performance in term of overhead as compared to the sender initiated protocol. Therefore, receiver initiated mesh based multicasting is more efficient than sender initiated mesh based multicasting. Example of receiver initiated routing protocols are preferred link based multicast (PLBM) [14], forward group multicast protocol (FGMP) [15], weight based multicast routing protocol (WBM) [16], data distribution management (DDM) [17], core assisted mesh protocol (CAMP) [18], protocol for unified multicasting through announcement (PUMA) [19], multicast for ad hoc network with swarm intelligence MANSI [20] and ODMRP.

The receiver initiated protocols suffer from two main problems. First, most of the protocols select the core node based on first come first serve basis, i.e. a node that first joins the receiver group. Therefore, the selected core may be in bad position with low battery capacity and hence not an efficient core is selected. In this protocol, we propose an efficient core selection method that elects core based on some parameters, such as battery capacity and location. As a result, the elected core would have prolonged lifetime. Second, core failures occur in the network due to various reasons, such as flat battery, out of range, or hardware fault that causes reconfiguration for re-selection of core node. As a result, this reconfiguration process will increase the overhead in the form of regular flooding of control messages and delay an ongoing communication; hence, the system will be considered as unreliable. In order to reduce the delay caused by reconfiguration and to enhance network reliability, we propose the notion of mirror core. When the core node is elected (based on some ratings), then the core will select the second topmost node as a mirror core. In case of the primary core failures, the mirror core will take the charge as a main core without causing any delay and extra overhead. Hence, the system will become more robust and the group communication will be continued without any delay.

The novelty of this research work is twofold. First, we propose a stable core that will ultimately reduce core failures. Second, we propose a solution to reduce the data collection process, delay and overhead when core failures occur. The rest of the paper is organized as follows. In Section 2, we briefly describe the literature survey of sender and receiver initiated protocols and explains their drawbacks related to the core selection and core failure. Section 3 introduces the design of ERASCA (Efficient and Reliable Core Assisted protocol in Mobile Ad-hoc Network) which builds and maintains receiver initiated mesh based multicasting with the method of core election and mirror core selection. Section 4 and 5 evaluates the proposed scheme using Network Simulator 2 (NS-2) and compared with other benchmark schemes using various metrics. The paper concludes in Section 6.

## II. RELATED WORK

The primary goal of ad-hoc multicast routing protocols is to construct and maintain a robust and efficient topology even during high network dynamics and limited bandwidth. Among these protocols, mesh based multicasting is considered more robust and reliable than tree based multicasting. The mesh based multicast routing is divided into sender initiated and receiver initiated multicast routing protocols [21].

In sender initiated approach, a sender starts the formation of the mesh. In this approach it will be the responsibility of the sender to maintain and update the multicast paths to receiver. The first problem appears when the number of receivers increases, as the number of reply packets sent back by the receivers to a sender also increases because after every successful reception of packets a receiver must reply back to the receiver, which creates a bottleneck at the sender end. Second, a sender initiated approach depends on a consistent network flooding, as every sender behaves as a core when it joins the network that leads to the problem of creating large overhead and energy consumption. Finally, the sources must be part of the multicast mesh group, even when they are not interested in a transmission. Therefore, when the source node increases, then the flooding from every source increases and will produce large overhead. Examples of sender initiated mesh based multicast protocols are DCMP, Neighbor supporting multicast protocol (NSMP) [22], ODMRP etc. On the other hand, a receiver initiated approach transfers maximum responsibility on the receivers for reliable data delivery and will solve the issues related to sender initiated routing protocols. First, in receiver initiated approach, only the receiver which didn't receive the packet will reply to the source. Hence, receiver initiated will not be affected when the number of receivers grows as compared to its counterpart. Second, in the receiver initiated approach, the core is responsible for the maintenance and update of the receiver group as compared to sender initiated protocol where each source needs to maintain the path from each source to its corresponding receivers. Finally, this approach does not make an extra overhead as compared to the sender initiated approach because in a receiver initiated approach the sources are not forced to be a part of the group.

ODMRP is a sender initiated mesh based protocol. It uses forwarding group concept in order to transmit multicast packets through flooding. In ODMRP, the source node administers the membership of the group, maintains and updates the multicast path and the multicast group. In ODMRP, when data packets sent by the source are received by the receiver, an acknowledgment is sent by the receiver to the sender that the data is received otherwise the sender will retransmit the packet after a period of waiting. This retransmission will continue until the reception of acknowledgement from the receiver, which will create congestion and overhead as the numbers of the receiver increases. Second, in order to maintain and update the group and the paths to the receivers, ODMRP depends on the consistent network flooding from the source nodes that leads to

the problem of scalability in situations where the source node increases [23].

MAODV is a receiver initiated multicast routing protocol. In MAODV, a receiver group will be established with the help of Hello messages and will make the connectivity list within the group. The first node which joins the group will be selected as a Leader (core). The Leader updates and maintains the receiver of a group with the help of Hello messages. In MAODV, when the nodes in one group find another group, they would like to merge the groups with each other. The main drawback of MAODV is the frequent link failure in high mobility because of its tree infrastructure and a single point of failure, which is the core node. Also the merging concept of one group within the other group in MAODV make it more complicated because the node will have to find the superior core within each other, which can create unnecessary delay.

CAMP is a receiver initiated on demand multicast routing protocol. It uses mesh based topology and a unicasting technique in order to establish and maintain a multicast group member to known destinations. CAMP establishes a mesh composed of shortest paths from senders to receivers and one or multiple core can be defined for each mesh. In small networks the CAMP work well, but creates a considerable amount of overhead and unreliability in large networks and high mobility [24]. Moreover, if any branch of a multicast tree fails, then all the components of the tree and its related branches must be reconnected for packet forwarding to continue the communication between the source and the destination.

PUMA is a receiver initiated mesh based protocol. It uses core node to transmit its multicast packets to the desired destination group. All the receivers are attached along the optimal path to the core, the core is selected among the receivers and therefore each and every node on the shortest path between a core and the receiver establishes the mesh. The first problem appears in PUMA, is that the first receiver in a group will be selected as the core. This first-come-first-serve based selection may cause illegitimate or inappropriate nodes to be selected as core which may have a minimum lifetime; hence, increase core failure chances in the network decreases the efficiency. Second, core failures can further cause reconfigurations and as a result reliability and network lifetime will be compromised. As, the main core fails, there is no alternative technique to prevent reconfiguration and save the existing information of the every node in the network because with core failure every node will delete all information related to the group which was achieved through communication.

All the above mentioned protocols select the core on a first-come-first-serve basis (i.e. the first node that joins the group). However, the CAMP uses the Extended Ring Search (ERS) method for another core and PUMA selected the core by election but with limited parameters. Hence the process of a core selection in all these protocols is not efficient as the selected core may be in bad position with low battery capacity and may cause frequent core failures thereby increasing overhead. Furthermore, the time and network resources required for the new core selection may cause the protocols to become inappropriate for a Quality of Service (QoS) based

applications, especially the delay caused in the process. However, in ERASCA, the first receiver who joins the group is selected as a core like the above protocols, but after the failure of the first core, it does not continue the same procedure, but elects the core on battery capacity and location. This will select a resourceful core within the group and decreases the core failure. As a result, decreases the reconfiguration and increases efficiency in term of minimizing the overhead. In order to increase the reliability and reduces the delay occurs in the new core selection process, we introduce the mirror core. The mirror core acts as a primary core after the first core failure occurs and prevents the network to go into orphanage phase.

### III. PROTOCOL DESCRIPTION

#### A. Overview

ERASCA uses the IP multicast service model of permitting any source node in a network to transmit its packets to the multicast group without knowing the constituency of the group. Furthermore, the ERASCA is based on a receiver initiated approach in which the sources are not required to be part of the receiver group for the transmission of data to receiver group.

In ERASCA, if the receiver does not receive any invitation from the group then it will announce himself as a core of the group. This core node will start the formation of the receiver group through Status Declaration message (Explain in Subsection B). In ERASCA, the receivers join a multicast group using the address of a core node. As a result, a group will form and every receiver in a group will be informed from each other status (i.e. battery capacity and location).

The receiver connects to the core node through intermediate nodes with the help of the SD message, which will be flooded by the core node and form the connectivity list. All the intermediate nodes connecting receivers to a core node acting as relay nodes, collectively form the mesh. With the help of connectivity list, the sender sends a data packet towards the mesh through the best possible route. On reception of data packets through any mesh member, it is flooded within the mesh members of the group and ultimately reaches to all the receivers of the group.

In ERASCA, the receivers elect the core to become the point of contact between the mesh members and non-mesh members (these terms will be explained in Subsection E in detail) and it is the responsibility of the core to periodically broadcast the updates about entry and exit of the mesh members, group members, mirror core and about its own existence to the rest of the network by using SD messages. Hence, it is the core node that updates and maintains the mesh group.

#### B. Status Declaration Message

In ERASCA, the core node uses SD Message Packet Format as shown in Fig.1 to maintain and update the mesh of a group by periodically flooding the SD messages to form a connectivity list. Connectivity lists are formed throughout the network with each node, which allows the sources to send the data to the mesh of a group. Each SD message specifies a core ID, group ID, parent node, sequence number, distance to the core mesh member flag and battery capacity. With the

information contained in the SD messages, define the path for sources outside a multicast group to transfer the data packets towards the group. The SD message maintains and updates the mesh of a group by informing others about the leaving of a mesh members or joining of the new receiver in the group.

0 15 31

Mesh Membership flag	Distance to Core
Group ID	
Core ID	
Mirror Core ID	
Sequence Number	
Parent ID	

Fig. 1. SD Message Packet Format

**Core ID:** Core node identifier

**Mirror Core ID:** Mirror Core node identifier

**Group ID:** Group ID of the concerned group

**Sequence number:** The sequence number in the best Status Declaration in which fresher sequence number is given

**Mesh member flag:** If the node is a part of the mesh, then the flag will be set otherwise it will not be set

**Distance to core:** The distance to the core in the best Status Declaration

**Parent:** The nearest neighbor from which it received the best Status Declaration

### C. Connectivity List

A core node periodically transmits the SD messages for the concerned group due to which each node forms a connectivity list in the network. With the help of connectivity lists, nodes in the network can calculate the best path from a source to a group through parent nodes. Parent node shows the preferred neighbor (which shows the shortest path to the core) to reach the core. The source node may or may not be the group member. All nodes in the network store the information they collect from their neighbors via SD messages along with the received time into the connectivity list. Fresher SD message (one with a higher sequence number) from the neighboring nodes is preferred over the lower sequence number for the same group. Therefore, for the same group a node contain only one entry in the connectivity list for the specific neighbor with a fresher sequence number for the given core. Hence, for the same core ID, the SD message with a fresher sequence number is preferred as shown in Table 1. For the same core ID and fresher sequence number, SD message with less distance to the core is preferred. For the same core ID, when all those fields are same then SD message with higher battery capacity is valid. For the same core ID, when all those fields are same than the SD message that arrived earlier is considered valid.

Fig.2 shows the dissemination of the SD message all

through the network and Table 1 shows the building of connectivity lists at node 8. The solid arrow shows the node from which it receives its best SD message. Node 8 has four neighbors in its connectivity list, i.e. 1, 7, 9 and 10. Neighbor 10 is not selected as a best entry because it has two hops distance with minimum battery capacity and a maximum delay. Neighbor 1 is not selected as a best entry because it has the minimum battery capacity and a maximum delay than node 7 and 9. However, it selects the entry it receives from neighbor 9 as the best entry, because it receives earlier than node 7. Now node 8 uses this best entry to produce its own SD message which contains all the fields as shown in Table 1.

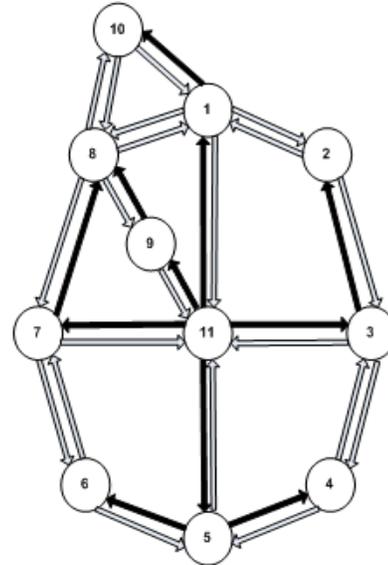


Fig. 2. Dissemination of SD Message

When a node receives a multicast data packet from a source node, it forwards it to the node from which it receives a best SD message. If the concern path is broken, then it tries next best path available, because in mesh multiple routes are available from source to group. As soon as the data packet reaches to any mesh member of the group, the mesh member floods the packet within the mesh group until the desired receivers get the data packet. Mesh members use a packet ID cache to detect and remove duplicate data packets during flooding. The routing of data packets within the network from sources to receivers are also used for the update of the connectivity lists. Because when the sender sends a data packet to the receiver through non-member then the non-member expects its parent node to forward the data packet to the mesh. As the MANET is broadcasted by nature, therefore the node also receives the packet when it is forwarded by its parent node and receives an implicit acknowledgment from the parent node that forwards its packet. But if the neighbors do not receive an implicit acknowledgment within a specific time interval from the parent node, it eliminates the parent node from its connectivity list. Therefore, connectivity lists are updated immediately as soon as it detects its parent lost.

TABLE I. CONNECTIVITY LIST AT NODE 8

Neighbor	Core ID	Group ID	Seq. no	Parent	Distance to core	BC	Time
9	11	224.0.1.2	64	11	1	90%	11132
7	11	224.0.1.2	64	11	1	90%	11138
1	11	224.0.1.2	64	11	1	87%	11144
10	11	224.0.1.2	64	1	2	80%	11159

D. Receiver Group Formation

In ERASCA two situations arise for the receiver group formation. First, a node n is interested to become a part of any receiver group, if any group exists. Hence, if the receiver group is existed, then it will have a core node. A core node is periodically transmits the SD messages for the receiver group. If node n receives SD message from any group member say m, then n will send a Join Request to the m for the receiver group membership. In reply a Join Acknowledgment is transmitted to n by m. Now it adopts the group specified in the SD message it has received and starts transmitting messages that specifies for the group. Second, if there is no receiver group then it will announce itself as a core node and start SD messages periodically to inform other receivers through SD messages to join the group.

After joining the receiver group, if receiver does not receive the SD message within 3 x SD interval after the first time, then it assumes that core has been failed. To confirm whether the core has failed or not, a receiver floods a Core Failure Announcement (CFA) Request. In CFA Request, the sequence number field is set to the highest sequence number from the old core. After sending a CFA Request, the receiver sets a CFA wait flag, as well as starting a timer with time period CFA ack timeout interval. Intermediate nodes will receive a CFA Request Reply with their fresher SD message, if they receive SD message with higher sequence number than the sequence number in CFA Request. Otherwise they will also set the CFA wait flag to TRUE, and start a timer CFA ack timeout interval and forward the CFA Request. On the other hand if the core failure is not occurred then the CFA Request will finally reach a receiver which receives a latest SD message than the sequence number in the CFA Request. The receiver then broadcast SD message with fresher sequence number in the receiver group which is forwarded back on the same route to initiator which originated the CFA Request. If a core failure has indeed occurred then the CFA Request will never reach the initiator because of the loss of the connectivity list and as a result the CFA ack timeout interval expires. Therefore, when the receiver initiating the CFA Request recognizes that it is not receiving the SD message with a higher sequence number within a specific time interval, then it will confirm and announce a CFA and will conduct an election as shown in Subsection F. CFA ack timeout interval should be set to

specific time interval which is sufficient for the CFA Request to come back to a receiver which originated the CFA Request.

E. Mesh Formation

The network nodes are categorized as group member and non-group member. Non-group members (NM) do not belong to the mesh and are shown in black color nodes. On the other hand, group members are further divided into End Receiver (ER), Intermediate Receiver (IR) and Group Relay (GR). The nodes in white represent the End Receivers (ERs). The ERs are terminal receivers, i.e. mesh is terminated on them, and they do not participate in the packet relay process. Whereas, the GR nodes can only act as intermediate nodes between the receiver and the core and we denote them by blue dots. Likewise, IR is the receiver node as well as the intermediate node simultaneously, denoted by red dots. As shown in the Fig.3, the intermediate node between R47 and core is R42. R42 is a receiver node but in this situation it also acts as an intermediate node and hence will be termed as IR.

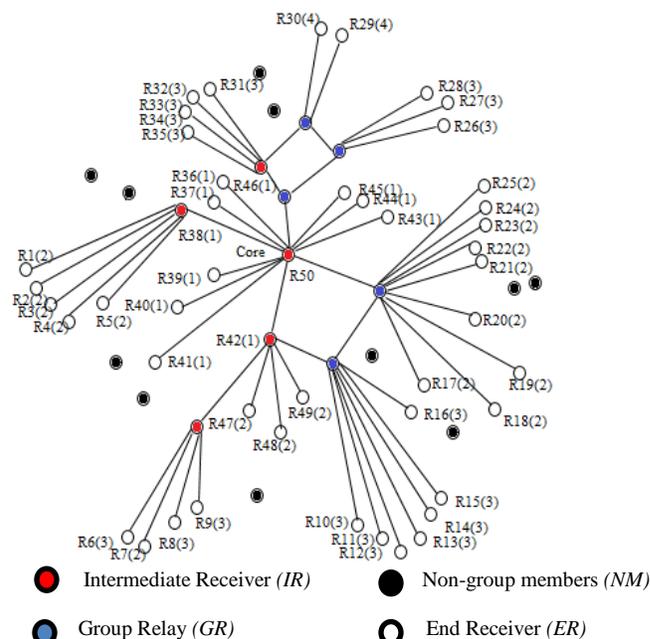


Fig. 3. Mesh Formation

Initially only receivers consider themselves as a mesh members but now GR will also consider themselves as a mesh member because they exist between core and ER and forward the packets between them and hence will be considered as part of a mesh group. A mesh group will be composed of ER, IR and GR nodes. As shown, blue nodes (GR) are the intermediates nodes that exist between the core and the receiver and having at least one or more ER node connected to it. It should be noted that flooding of SD from the core will only be carried on by the IR and GR nodes, instead by all the group members. To limit the flooding only to the IR and GR nodes, considerably reduced the overhead. In ERASCA, only the ER and IR node can be selected as a core, whereas GR cannot be selected as core node because GR acts only as an intermediate node and not a receiver.

### F. Core Election

In traditional approaches, when the core node fails because of mobility or battery capacity, the group will again select the core irrespective of its position and battery capacity. With inappropriate core location, i.e. in a less populated area, the core will face a large delay with maximum link failure which decreases the efficiency of the network. Likewise, having low remaining battery capacity, the core failure occurs soon and hence a core with high battery capacity should be preferred which will possibly increase the lifetime of the network. In this approach an election is conducted, to elect a core. Thus after the failure of the first core, it does not continue the same traditional approaches, but elects the core with proper election on battery capacity and location (i.e. dense part of the network or maximum connectivity). This will select a resourceful core within the group. In order to select a resourceful core, the following steps will be performed.

**Step 1:** In situations when the core node fails, the group members will be aware of the core failure situation through *CFA*. The *CFA* will be flooded by the *IR* and *GR* within the group, if *IR* and *GR* do not receive 3 consecutive *SD* message within a specific time interval. Each *SD* message is announced after 3 seconds.

**Step 2:** After the *CFA*, an election is conducted in a receiver group. For this purpose, a *receiver n* floods the *Election Request* message to all receivers within the group as shown in Fig.4. The purpose of this message is to inform all receivers that the core has been failed. If the core is really failed, the receiver *n* will receive an *Election Reply* message from all receivers in a group; otherwise, it will receive nothing after some time interval. It would mean that the initiator may be gone out of range of the network.

**Step 3:** In reply all receivers in a group will flood the *Election Reply* message to receiver *n*, if the core is also recorded to be failed with all receivers in a group. The purpose of *Election Reply* is twofold. First, shows its willingness to participate in core election process. Second, each receiver will establish paths to every other receiver in a group.

**Step 4:** For knowing a Remaining Battery (*RB*) and number of connected neighbors of all receivers, a *Core Election Message* is flooded in a group by a *receiver* to elect the best receiver in a group.

**Step 5:** All receivers will also flood a *Core Election Message (CEM)* within the group in which each receiver must include its *BC* and number of connected neighbors. As a result, every receiver will have a list of receivers. Each receiver floods its topmost receiver in a group. This will enable every receiver to have all the votes regarding topmost receiver node.

**Step 6:** As a result, all the receivers will know the estimated battery capacity and number of connected neighbors of each other. After exchanging information through *CEM*, receiver *n* elects its topmost receiver in a group.

**Step 7:** All receivers will also elect its topmost receiver in a group.

**Step 8:** Top most receiver is flooded by the receiver *n* as well as by the all group members.

**Step 9:** As a result, a topmost receiver within the receiver group with high battery capacity and maximum numbers of neighbor is elected as a core node.

**Step 10:** The core node will flood this news within the group about its own existence through *SD* message.

There are two aspects of core election process. Firstly, an efficient core is selected on the basis of battery capacity and best position, which will perform its duty as a core for a long period of time in the network. It is obvious that a good location of the receiver might be the one that is less dynamic or that the neighborhood environment is less stagnantly changing i.e. fewer changes occur in a given time interval. Also, it can be assumed that a node with maximum number of neighboring nodes will probably be in the center of the group despite at the corner of the receiver group. Secondly, the group will get rid of frequent core failures hence, overhead will be decreased. After core election, the core node sends *SD* message with its node ID to the whole network.

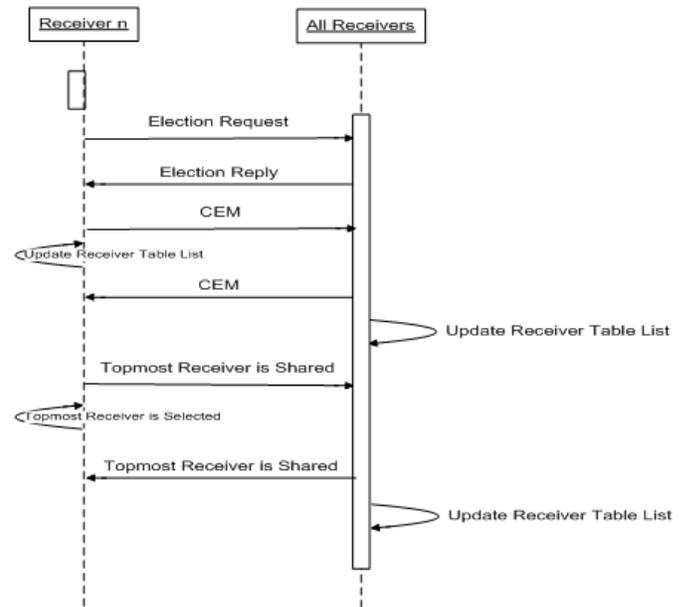


Fig. 4. Sequence Diagram

### G. Mirror Core Selection

In order to resolve the issues related to core failures, we introduce the mirror core. The primary core selects the mirror core from the rating list (which should be the second topmost node). Therefore, when the main core fails the mirror core takes the responsibility as a main core and the mesh group will be maintained and updated continuously without any delay. It has certain advantages such as maximum reliability, minimum delay and minimization of the data collection process.

After the selection/election of the core, it is compulsory for the core to select the mirror core of the group. For this reason, the core selects the most suitable receiver within its broadcast range (preferably with one hop distance) as shown in a black dotted circle in Fig.5. Likewise, the mirror core can also be found within two hop distance with the help of *GR* in blue dots (N51, N52 and N53) and *IR* in red dots (R38, R46 and R42)

and are explained Subsection H. Here only the ER and IR node can be selected as a mirror core and the GR nodes cannot be selected as a mirror core because GR nodes are not the member of the receiver group but only serving as an intermediate node. As soon as the mirror core becomes a primary core, it starts to transmit SD messages in the network about its status as being the core node.

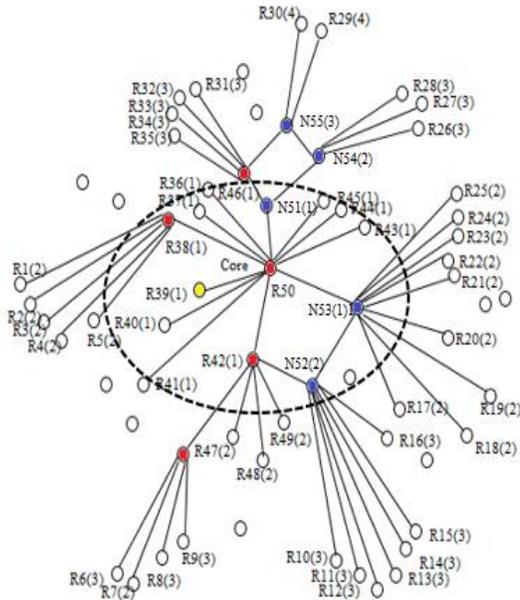


Fig. 5. Mirror Selection within neighborhood

After the core election/selection, it is the responsibility of the core node to select the mirror core. The mirror core is selected by the main core on factors like battery capacity and distance to core within a receiver group. After the core election/selection the primary core will first prefer the suitable receiver within one hop distance, if not found then prefer 2-hop distance and so on. The suitable receiver must have the highest aggregate after the core node. The aggregate depends on battery capacity and numbers of hop between the receiver and the core. For the mirror core selection the core floods the Mirror Core Selection Request (MCSReq) within the receiver group. The purpose of this message is to inform other receivers that the mirror core has not yet been selected. As a result, the core will receive a Mirror Core Selection Reply (MCSRep) from the other receivers in a group through unicasting. The purposes of MCSRep are twofold. First, it shows willingness to participate in the mirror core selection process. Second, each receiver will establish paths to the core in a group.

For knowing a Remaining Battery (RB) and distance of core (in term of hops) from each receiver, a Mirror Core Selection Message (MCSM) is flooded in a group by the core to select the best receiver in a group. In reply all the receivers in a group will send their Mirror Core Selection Message (MCSM) through unicasting to the core node. Now the core node has a table of receiver list through which the core will select the topmost receiver as a mirror core. Now the core will

transmit the SD message with mirror core in its packet format and after the failure of the main core the mirror core takes the responsibility as a main core without data collection process and starts SD message without any delay. It is important to mention here, when the mirror core takes on the role of the primary core in the mesh. There are two occasions. First, when the core node depletes its resources quickly, i.e. battery capacity and explicitly announces “resource exhaustion” message. Second, when the core node has abnormally disappeared due to mobility or hardware faults, etc. In the above two situations, the mirror core takes the charge as a primary core.

H. Connectivity List of Mirror Core

If the mirror core is not found by the main core in one hop distance, then the core will select the mirror core within the group through IR and GR nodes. To select the mirror core in the group, a core should be aware from the status (battery capacity and distance to the core) of every receiver in a group. For this purpose MCSM is flooded in the group by the core through IR and GR nodes. In reply all the receivers will send their status to the core node, which gives an expanded choice to the core for the mirror core selection. Therefore, through IR and GR a suitable mirror core can be selected.

It should be noted that a mirror core should only be selected by the core node within one hop distance based on battery capacity and distance to the core, however, the receiver with low battery capacity within one hop distance is not preferred. On the other hand, a suitable receiver with high battery capacity within a two or three hop neighborhood and not more than 3-hop neighborhood will be preferred over a receiver with low battery capacity within one hop neighborhood. Because a mirror core failure will occur soon within one hop with less battery as compared to a mirror core with a high battery capacity within two or three hop neighborhood.

Table 2 shows the connectivity list of node R38. In this situation only R38 is consider, where five receivers are connected to R38 at two hop distance. R3, R4 and R5 will not be selected as a mirror core, as they have a minimum battery capacity than R1 and R2. Likewise, R1 and R2 can be selected as a mirror core with high battery capacity, but priority will be given to R1, because R1 receives the SD message earlier than R2 and hence R1 will be selected as a mirror core of the primary core. Similarly, the mirror core can be selected through (N51, N52 and N53) and (R46 and R42).

TABLE II. CONNECTIVITY LIST AT NODE R38

Neighbor	Core ID	Group ID	Seq. no	Parent	d_to core	BC	Time
R1	R50	224.0.0.2	5	38	2	88%	14132
R2	R50	224.0.0.2	5	38	2	88%	14135
R3	R50	224.0.0.2	5	38	2	75%	14138
R4	R50	224.0.0.2	5	38	2	70%	14150
R5	R50	224.0.0.2	5	38	2	30%	14155

#### IV. SIMULATION SCENARIO

This paper implement, evaluate and compare this proposed solution in a network simulator with the benchmark schemes like PUMA and MAODV and use NS-2.35 on Ubuntu platform using Tcl/Otcl and C++ as a front and back-end languages respectively for implementing the proposed ideas. Likewise, an AWK script is developed to collect data from NS-2 trace file.

##### A. Metrics

In this experiment, the following metrics are used, i.e. throughput, packet delivery fraction (PDF) and overhead with the following parameters as given in Table 3.

TABLE III. SIMULATION PARAMETERS

Simulator	Network simulator (NS2)
Simulator time	450 Sec
Number of nodes	50
Mobility	5
Simulation area	1000m x 1000m
Data packet size	512 bytes
IfqLen	60
MAC type	MAC802_11

**Throughput:** is the measurement of performance of MANET, which shows the amount of data transfer from one location to another location in the specified amount of time and depends on multiple factors like channel capacity and bandwidth etc.

**Packet Delivery Fraction:** can be defined as a data packet received divided by the data packet sent.

PDF = total number of packets received/ total number of packets sent

**Overhead:** is the total packet sent (control packet + data packet) divided by the data packets received.

Overhead= total packet sent (control packet + data packet) / data packets received

Several scenarios have been simulated in order to determine the effect of mobility, number of receivers, number of senders, ifqLen and simulation area on the performance metrics for each protocol. Five scenarios have been simulated in different environments and on the basis of these scenarios we evaluate these protocols and make the conclusion on the basis of results.

**Scenario 1:** Mobility changes across {0, 10, 20, 30, 40} m/s

Senders = 5, Receivers = 20, ifqLen = 60, Simulation area = 1000 x 1000

**Scenario 2:** Senders changes across {1, 2, 3, 4, 5}

Mobility = 5, Receivers = 20, ifqLen = 60, Simulation area = 1000 x 1000

**Scenario 3:** Receivers changes across {1, 2, 3, 4, 5}

Mobility = 5, Senders = 20, ifqLen = 60, Simulation area = 1000 x 1000

**Scenario 4:** ifqLen changes across {10, 20, 30, 40, 50, 60}

Senders = 5, Receivers = 20, Mobility = 5, Simulation area = 1000 x 1000

**Scenario 5:** Simulation area changes across {500 x 500, 1000 x 1000, 1500 x 1500, 2000 x 2000}

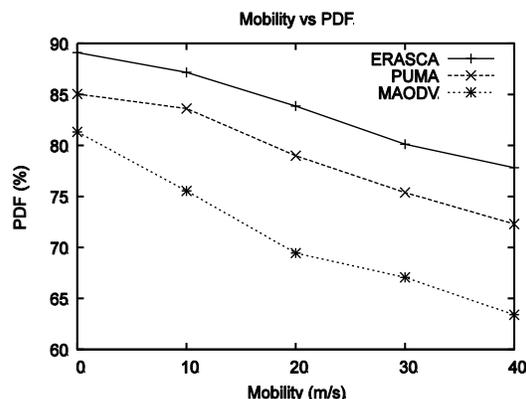
Senders = 5, Receivers = 20, Mobility = 5, ifqLen = 60

#### V. PROTOCOL COMPARISON

In this paper, the performance of ERASCA is compared with PUMA and MAODV which are the benchmark schemes for mobile ad-hoc network using the network simulator (NS2) parameters given in Table 3. ERASCA, PUMA and MAODV are receiver initiated routing protocols. However, ERASCA and PUMA are mesh based protocols and MAODV is a tree based protocol.

##### A. Scenario 1

In scenario 1, the mobility is changed from 0-40 and makes all other parameters fixed as given in Table 3. On the basis of such parameters, multiple simulations are performed on protocols like PUMA and MAODV and compare their matrices like PDF, throughput and overhead with ERASCA. As shown in the Fig.6, throughput, PDF and overhead change with respect to mobility. In low mobility the packet drop decreases and PDF increases, but the opposite happens when the mobility increases. As with high mobility the link failure increases and therefore the delay is higher. As a result, throughput decreases because throughput is the packet transmission per second. In such a situation frequent flooding is used to minimize link failure and hence the overhead increases and an ongoing communication is delayed. Because of the delay, link failure and overhead the throughput and PDF decreases. Hence, the network performance decreases because of the frequent link failure and core failure. MAODV shows poor performance as compared to PUMA and ERASCA. MAODV is a tree based protocol, as tree based protocols are not resilient against mobility because of a single route between a sender and a receiver and therefore the packet delivery ratio is very less and overhead is high as compared to mesh based multicast routing protocols. It is important to mention that



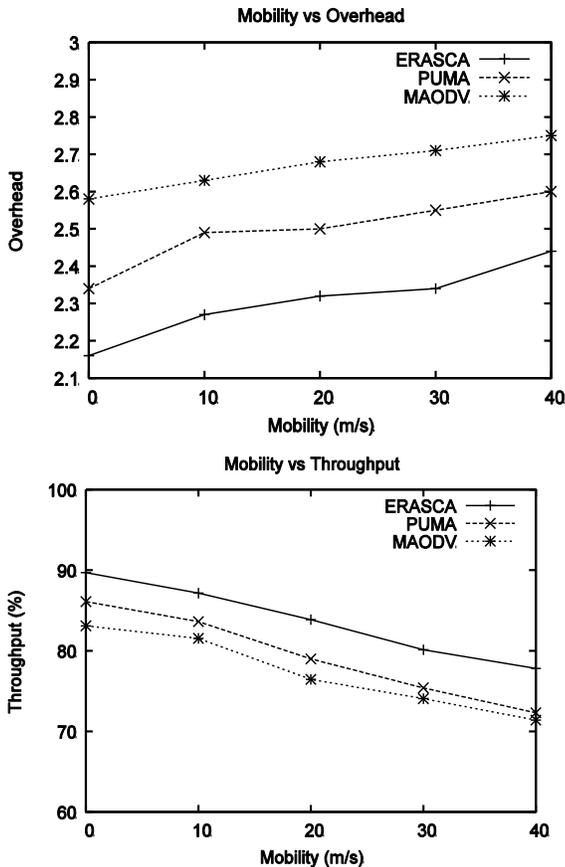


Fig. 6. Comparison of Mobility with PDF, Overhead and Throughput

When the link failure and packet drop increases then the PDF decreases. On the other hand, PUMA shows better performance than MAODV but less performance to ERASCA because of the frequent core failures. Therefore, ERASCA shows better PDF, throughput, overhead and delay as compared to MAODV and PUMA because of the stable core selection. As in ERASCA less core failure occurs and hence decreases the reconfiguration process.

B. Scenario 2

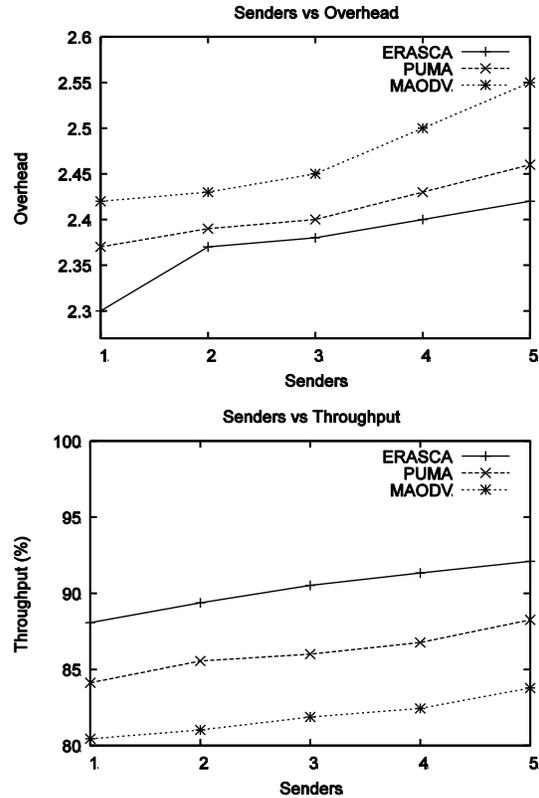
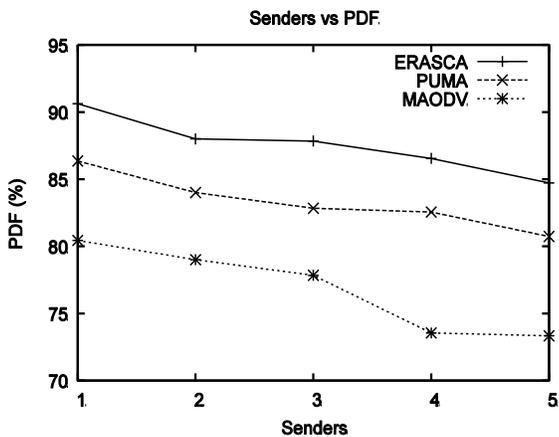


Fig. 7. Comparison of Senders with PDF, Overhead and Throughput

As shown in the Fig.7, throughput, PDF and overhead change with respect to senders. In routing when the numbers of sender increases, then the overhead and throughput also increase because of the inclusion of multiple packets from multiple senders. In Fig.7, MAODV performance is not satisfactory because of the single path between the sender and the receiver.

Since, in high mobility the possibility of link failures also increases between the source and the destination as compared to PUMA and ERASCA which are mesh based protocols. On the other hand, PUMA and ERASCA show a little difference in performance because of the redundant path availability between sender and receiver group. However, the little difference in performance is because of the frequent core failure situation in PUMA, as the inappropriate core is selected on bad location with low battery capacity.

C. Scenario 3

In Fig.8, when the number of receivers increases, then the overhead and throughput increases as it should be. The PDF increases because of the availability of multiple and short paths between the group of receivers, as well as it provide robustness to the network and decreases packet drop as compared to the long and fewer route. The maximum number of receivers also provides richer connectivity to the network; as a result, high throughput is achieved. As compared to PUMA and MAODV, ERASCA gives higher performance because of the appropriate core selection.

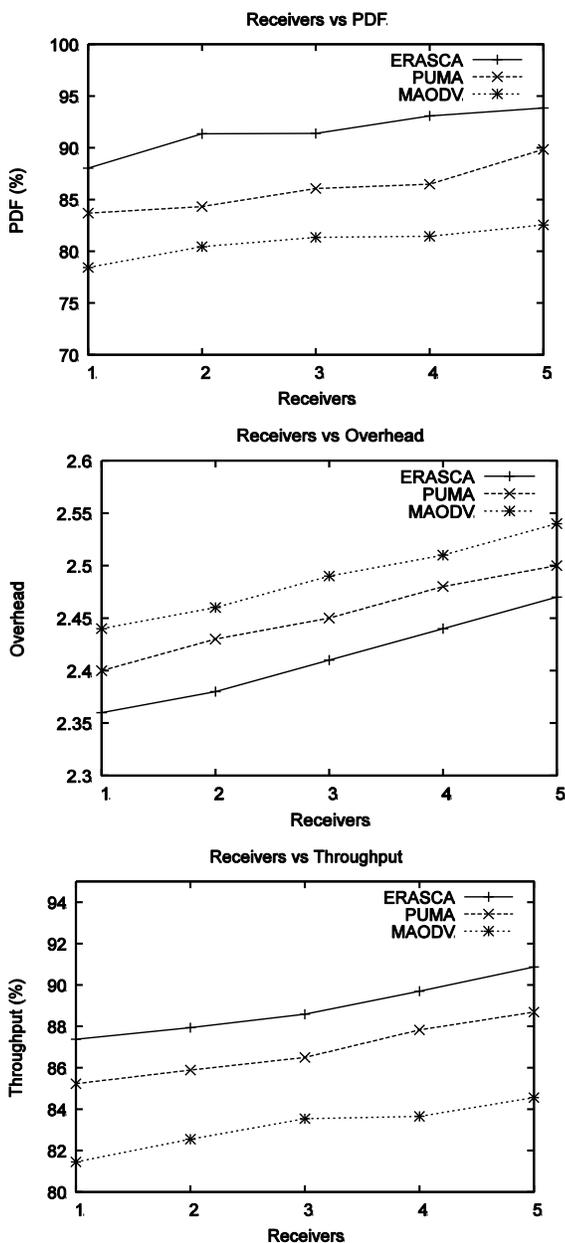


Fig. 8. Comparison of Receivers with PDF, Overhead and Throughput

The core selection is very important in MANET which affect network efficiency and lifetime of the network, but the core selection process are not efficient in PUMA and MAODV which deteriorate the performance of both protocols in the form of high overhead, as the selection of core in MAODV is not appropriate in location and energy wise. On the other hand, PUMA selects the core appropriately, but with minimum metrics and therefore it is believed that these approaches are not efficient because the selected core within a receiver may be in bad position in the network with minimum numbers of receivers and with low battery remaining. This selection increases the core failure and hence increases the packet drop and overhead. But in ERASCA, the core is selected within the best position in a receiver with high battery capacity; hence the

core failure situation won't occur frequently and will improve the performance of ERASCA than PUMA and MAODV.

D. Scenario 4

Here the ifqLen is referring to the buffer size. At the start of the simulation, it is noticed that maximum packet drop occur in ERASCA, PUMA and MAODV because the smaller ifqLen represents a small buffer. Therefore, a large number of packets with small buffer ultimately increase the packet drop and hence decrease the PDF and increases the delay. Because of the

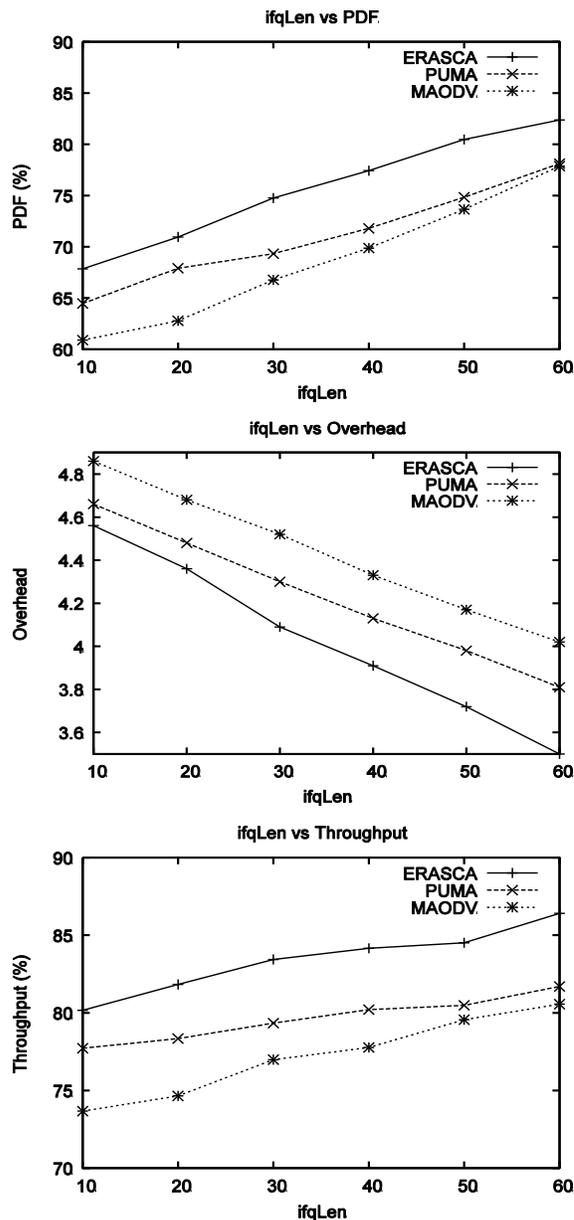


Fig. 9. Comparison of ifqLen with PDF, Overhead and Throughput

maximum packet drop, sender will frequently transmit the data packet to the destination until the data is received by the destination. Thereby, increases the overhead by the frequent transmission from the source.

Fig.9 shows that with an increase in the ifqLen decreases the overhead, because with large ifqLen packet drop decreases and the data may successfully and frequently reaches from the sources to the group. As a result, flooding will be decreased and hence the overhead decreases. Therefore, in large queue, a large number of packets from source to destination are entertained and hence, the throughput increases. In ERASCA, the packet drop is less because of its core selection methodology. The appropriate core selection decrease core failure and hence minimize the flooding, as flooding reduces the packet drop, link failure and overhead with the increases in PDF and throughput.

#### E. Scenario 5

Multicast routing protocols generally show good performance within a small simulation area with shortest paths between senders and receivers, as data delivery latency and possibilities of link failure is low. Therefore, the throughput and PDF are increases and overhead decreases but in a large simulation area the throughput and PDF decreases, but increases the overhead and energy consumption with frequent packet drop, link failure and core failure. In such a situation core failure increases, as the distance between the receiver and the core increases. Hence, the regular reconfiguration for the next core node results in continuous flooding of control messages across the network, which increases the congestion, packet drop, delay and link failure. As a result, PDF and throughput decreases and increases the overhead. In Fig.10, with increase in the simulation area increases the link failure, resending of data from source to destination as well as frequent core failure. However, ERASCA shows better performance as compared to PUMA and MAODV because a stable core selection works well in large simulation area. As in the large simulation area a more stable core is required to minimize core failure because the frequent core failure in a large simulation area affects the performance of the network badly in term of link failure and delay. Hence, efficiency increases with improved lifetime of the network.

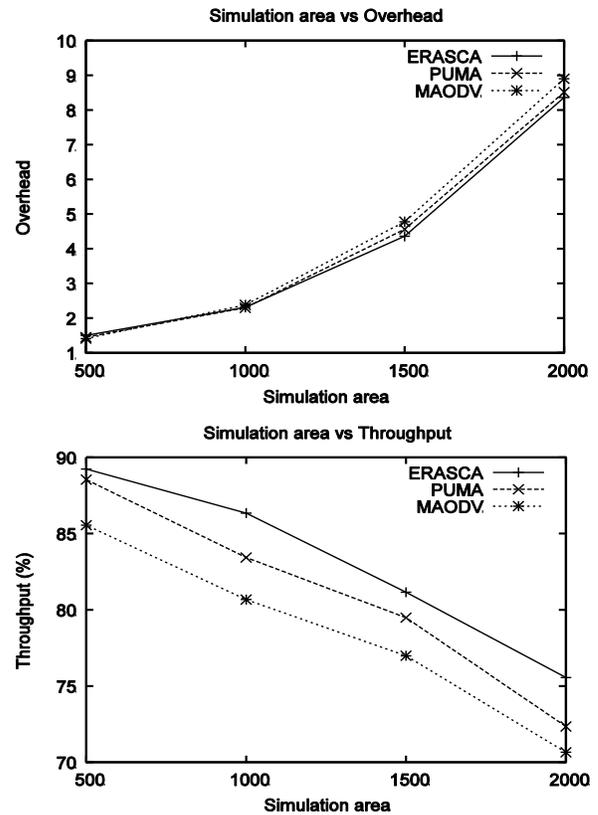
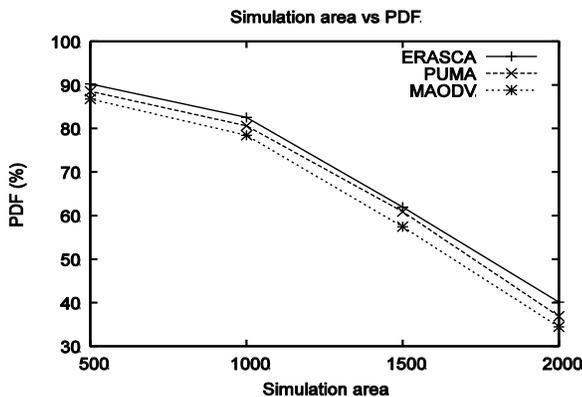


Fig. 10. Comparison of Simulation area with PDF, Overhead and Throughput

## VI. CONCLUSION

Among the multicast routing protocols, ERASCA provides efficiency to MANET by reducing overhead. The ERASCA strongly relies on proper selection of core node in which the core is selected within the receiver group on the basis of multiple parameters like battery capacity and location in the network. As a result, a more stable core will be selected with high battery capacity and maximum numbers of neighbor. To increase the reliability of the network, the mirror core is introduced. Therefore, after the failure of the primary core the mirror core will take the responsibility as a primary core and will not affect the ongoing communication, hence minimizes the delay. ERASCA is compared with PUMA and MAODV; ERASCA demonstrated better performance with its core election process and in the presence of mirror core. Therefore ERASCA can be used efficiently and reliably in high mobility scenarios within large area irrespective of the number of receivers and senders with minimum packet drop and overhead with maximum reliability, throughput and PDF.

## VII. FUTURE WORK

In future the ultimate plan is to secure the core election process. As the core election is an important and sensitive process and an adversary or malicious entities will always try to take over the position of core and can disrupt the core formation/ core election process by fabricating the SD messages and disseminating false data in the network/group for malicious purposes, Hence, in future a solution is propose in order to secure the core election process and counteract the malicious attacks, such as dissemination of false or fabricated information. Likewise in ERASCA, as soon as the data packet is received by any mesh member, it is flooded within the receiver group and ultimately the destined receiver will get the data soon but at the cost of overhead. The flooding is best in situation, when the mesh member (that receive the data packet) and the destined receiver are far away from each other. Therefore, the destined receiver will receive the data with minimum delay but with increase overhead. But in situation when the mesh member (that receive the data packet) and the destined receiver are near to each other, then multicasting is the better approach which will decrease the overhead. In ERASCA the flooding is preferred because of the location unpredictability between the mesh member and the destined receiver. Therefore, in future such a protocol should be design, where multicasting should be used within the mesh group with minimum delay, overhead and node prediction technique.

### REFERENCES

- [1] A. Boukerche, "Algorithms and protocols for wireless, mobile Ad Hoc networks," John Wiley & Sons, vol. 77, 2008.
- [2] D. M. N. Hemangini, "Study of Routing Protocols in Mobile Ad-Hoc Network," SSRG International Journal of Mobile Computing & Application (SSRG-IJMCA). vol. 2, 2015.
- [3] M. A. Kodole and P. Agarkar, "A Survey of Routing Protocols in Mobile Ad-Hoc Networks," Multidisciplinary Journal of Research in Engineering and Technology, vol. 2, 2015.
- [4] L. Junhai, X. Liu, and Y. Danxia, "Research on multicast routing protocols for mobile ad-hoc networks," Computer Networks, vol. 52, pp. 988-997, 2008.
- [5] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu, "A survey of multicast routing protocols for mobile ad-hoc networks," Communications Surveys & Tutorials, IEEE, vol. 11, 2009.
- [6] C. E. Perkins, Ad hoc networking: Addison-Wesley Professional, 2008.
- [7] P. Mohapatra, AD HOC NETWORKS: technologies and protocols: Springer Science & Business Media, 2005.
- [8] M. Bouhorma, H. Bentaouit, and A. Boudhir, "Performance comparison of ad-hoc routing protocols AODV and DSR," International Conference on Multimedia Computing and Systems, pp. 511-514, 2009.
- [9] Survey on Tree Based, Mesh Based and Stateless Multicast Protocols in MANET," International Journal of Innovative Research in Computer and Communication Engineering, vol. 2, 2014
- [10] P. SAHU, "Disadvantage of AMRIS Protocol and its solution," International Journal of Engineering Research and Technology, 2012.
- [11] C.-H. Huang, C.-T. Wu, K.-W. Ke, and H.-T. Wu, "MAODV-based multisource multicast routing with fast route recovery scheme in MANETs," in Computer Symposium (ICS), International, pp. 79-84, 2010.
- [12] S. Mangai, A. Tamilarasi, and C. Venkatesh, "Dynamic core multicast routing protocol implementation using ant colony optimization in ad hoc wireless networks," International Conference on Computing, Communication and Networking, pp. 1-5, 2008.
- [13] M.-A. Kharraz, H. Sarbazi-Azad, and A. Y. Zomaya, "On-demand multicast routing protocol with efficient route discovery," Journal of Network and Computer Applications, vol. 35, pp. 942-950, 2012.
- [14] B. O. Reddy, C. V. Narayana, and S. S. Reddy, "Routing protocols classification for Ad Hoc Networks," International Journal of Advanced Research in Computer Science, vol. 2, 2011.
- [15] P. P. M. Krishna and M. S. D. K. S. Prasad, "Mesh based and Hybrid Multicast routing protocols for MANETs: Current State of the art," Global Journal of Computer Science and Technology, vol. 12, 2012.
- [16] S. K. Das, B. Manoj, and C. Murthy, "Weight based multicast routing protocol for ad hoc wireless networks," IEEE in Global Telecommunications Conference, GLOBECOM'02, pp. 117-121, 2002.
- [17] A. Boukerche and K. Lu, "Optimized dynamic grid-based DDM protocol for large-scale distributed simulation systems," 19th IEEE Proceedings in Parallel and Distributed Processing Symposium, pp.6, 2005.
- [18] O. S. Badameh and M. Kadoch, "Multicast routing protocols in mobile ad hoc networks: a comparative survey and taxonomy," EURASIP Journal on Wireless Communications and Networking, p. 26, 2009.
- [19] A. Antony, "Experimental Investigation Of Streaming Over Mobile Ad Hoc Networks Using PUMA," in International Journal of Engineering Research and Technology, 2013.
- [20] S. Vasundra and B. Sathyanarayana, "Fast Recovery From Topology Changes And Communication Link Failures," i-Manager's Journal on Software Engineering, vol. 5, p. 50, 2010.
- [21] M. Jahanshahi, M. Dehghan, and M. R. Meybodi, "LAMR: learning automata based multicast routing protocol for multi-channel multi-radio wireless mesh networks," Applied intelligence, vol. 38, pp. 58-77, 2013.
- [22] A. J. Selvarani and P. A. Selvam, "Study of Routing Protocols in Mobile Ad Hoc Networks."
- [23] A. M. A. Mo'men, H. S. Hamza, and I. Saroit, "A survey on security enhanced multicast routing protocols in Mobile Ad hoc Networks," in High-Capacity Optical Networks and Enabling Technologies (HONET), pp. 262-268, 2010.
- [24] J. Garcia-Luna-Aceves and E. L. Madruga, "The core-assisted mesh protocol," Selected Areas in Communications, IEEE Journal on, vol. 17, pp.1380-1394,1999.