Scalable Location Aware Multicasting Protocol for Mobile Ad hoc Networks

Mohammad M. Qabajeh

Department of Computer Science, Palestine Technical University Kadoorie, Palestine

Summary
The wireless mobile networks and devices are becoming increasingly popular as they provide users access to information anytime and anywhere. Mobile Ad-Hoc NETworks (MANETs) can provide users with these features. Multicast communication has been widely considered in research comunity due to the necessity of the group-oriented applications over MANET. However, multicast routing in large-scale networks faces several difficulties and challenges that need to be addressed.

Key words:
MANETs, Multicast, Routing, Protocol, Position-based, GPS

1. Introduction

Mobile Ad hoc networks (MANETs) are composed of a collection of mobile nodes that communicate with each other over wireless links in the absence of any infrastructure or centralized administration. MANETs have gained significant interest and popularity since they have enormous potential in several fields of applications. Over the past few years, the necessity of applications that require close association of the member nodes over mobile Ad hoc networks gained high popularity [1]. Multicast communication is fundemntal communication mechanism in such type of applications to reduce the routing overhead. With the development of wireless communications and decreasing cost of wireless hardware, a mobile device is able to obtain its location information [5]. This awareness of location information has been utilized to improve scalability and efficiency through restricting the broadcast region and reducing the routing packets of MANETs routing protocols.
Clustering algorithms are proposed in Ad hoc networks as an attractive approach to improve routing protocol scalability [6]. A clustering algorithm is usually used to divide the network into smaller sub-groups. The idea of using clustering is not new. Previous efforts in this issue are discussed in [7-11]. In general, clustering can provide scalability and reduce signaling traffic [12]. This is essential in networks with large number of nodes (e.g. hundreds or thousands). For example, if a flat structure is used in a large network, routing tables and location updates would grow to a huge size. Therefore, partitioning the network into multiple clusters can limit the size of routing tables. Moreover, detailed topology information for a particular cluster is only exchanged among local cluster members whereas aggregated information is propagated between neighboring clusters in a higher hierarchical level [12]. Distributing the load among multiple nodes improves performance and scalability of the routing protocol. It also helps in achieving robustness and solving the single point of failure problem. Whenever multicast routing is applied in large-scale networks, the problem will become worse if all nodes maintain routing tables. Thus, clustering is utilized to address the scalability issue in multicast routing.
In this research, Scalable Position-based Multicast Routing Protocol (SPMRP) is proposed. The proposed virtual-tree-based structures significantly reduce the tree management overhead, support more efficient transmissions, and make the transmissions much more robust to dynamics. Geographic forwarding is used to achieve further scalability and robustness. To avoid periodic flooding of the source information throughout the network, an efficient source tracking mechanism is designed.
The rest of the paper is organized as follows: In the consequent section, an overview about the proposed protocol is introduced. Section 3 provides a detailed description of the resulted control overhead. Section 4 presents our simulation results. In section 5, a discussion of the generated results is provided. Finally, concluding remarks are summarized in section 6.

2. Related work

In recent years, several researchers have focused on multicast routing protocols over MANETs. This is because multicasting over MANETs still needs further research compared to unicast routing [3, 4]. ODMRP [5] and Multicast Ad hoc On-demand Distance Vector Routing (MAODV) [6] are well known examples of flat topology-based multicast routing protocols. This type of protocols produces large control overhead specially when the network size grows up.
ODMRP [5] is a mesh on-demand multicast routing protocol that uses the forwarding node concept. When the source has packets to send, it broadcasts a JOIN_QUERY control packet to the entire network. The source periodically floods this packet (e.g. at an interval of 3

Manuscript received January 5, 2018
Manuscript revised January 20, 2018

seconds) to refresh the membership and update the routes. When a non-member node receives this packet it stores the upstream node_ID and rebroadcasts the packet. While, when a multicast member receives this packet, it creates or updates the source entry in its MEMBER_TABLE and broadcasts a JOIN_REPLY packet to its neighbors. The nodes that receive this reply packet check if the next node_ID of one of the entries in JOIN_REPLY_TABLE matches its own ID. If true, the node realizes that it is on the path to the source and becomes part of the forwarding group by setting the Forwarding Group Flag (FG-FLAG). Then, it broadcasts its own JOIN_REPLY_TABLE built upon matched entries. The group members continue to propagate the JOIN_REPLY packet until it reaches the source through the shortest path. The set of forwarding nodes constructs or updates the multicast mesh from sources to receivers.

A zone-based protocol is proposed in [7]. This protocol is proposed to improve scalability of location-based multicast protocols. EGMP divides the network topology into geographical non-overlapping square zones, and a leader is elected in each zone to serve as a representative of its local zone on the upper tier. The leader collects the local zone's group membership information and represents its associated zone to join or leave the multicast sessions as required. At the upper tier, the leaders of the member zones report the zone memberships to the sources directly along a virtual reverse-tree-based structure or through the home zone.

The position information is used to implement a hierarchical group membership management. EGMP maintains the tree by introducing a concept called zone depth, which is the depth of the member zone and root of the tree. The sender node sends the multicast packet directly in the tree, and then it will flow along the multicast tree at the upper tier. When a zone leader receives the packet, it will send the packet to the group members in its local zone. Nodes inside the same zone are within each other's transmission range and forwarding nodes are used for the communication between nodes in different zones. In this protocol the location service is combined with the hierarchical zone structure. So, the packet is forwarded to the center of the destination zone and then it is forwarded to the specific zone or broadcasted depending on the message type. The multicast session is initiated by flooding a message into the whole network and the node that is interested in the multicast group can join this session. The flooding of the multicast session initiation is easy to be implemented but it introduces a large overhead. Also, the size of the multicast message is large as it contains the list of next hop for all destinations in addition to the destination list.

Cheng, et al. [8] proposes a hybrid source-based tree on-demand multicast routing protocol (GMZRP) that combines the advantages of both topology and geographic routing schemes. GMZRP is inspired from the unicast protocol ZRP [9]. In this protocol, the network is partitioned into small equal circle shape zones starting from the network center and spreads outwards. Each circle contains a hexagon with same side length as the circle radius.

GMZRP maintains a multicast forwarding tree at two levels of granularities: zone granularity and node granularity. Zone granularity looks like source routing where the source keeps a zone ID chain connecting the source to each receiver. Intermediate nodes also keep zone ID chain connecting its own zone to each downstream receiver zone. On the other hand, at node granularity, the source and the intermediate nodes only keeps information about its child nodes. GMZRP works independently of any geographic unicast protocol and it shows competing packet delivery ratio and lower overhead compared with ODMRP protocol. However, GMZRP incurs large overhead to handle multicast group management. This is due to the large amount of broadcast of MRREQ packets.

SPBM [10] is designed to provide scalability for PBM protocol [11]. The network is sub-divided into quad-tree with a predefined level of aggregation. The top level is the whole network and the bottom level is constructed by basic squares. The higher level is constructed by larger squares with each square covering four smaller squares at the next lower level. All the nodes in a basic square are within each other's transmission range. In a basic square, a node periodically broadcasts its position and membership information. At each level, the membership of every square is periodically flooded to its upper level square. This periodic flooding is repeated for every two neighboring levels and the top level is the whole network.

Each node has an aggregate view of the position of the group members. SPBM uses the geographic positions of the nodes from global and local member tables provided by group management scheme to make the forwarding decision of the data packets. The forwarding algorithm checks if the current node is a member of the multicast group in order to deliver the data packet to that node.

The main focus of SPBM is scalability for the group management through using hierarchical aggregation of membership information. However, SPBM relies on unicast geographic routing for each destination, which makes it fails to provide efficient multicast forwarding. In this protocol, to determine the most suitable next hop for a packet to a given destination, the source compares the geographic progress for each of the neighbors in respect to the destination and picks the neighbor with the highest progress.

In SPBM, the use of periodic messages to update the membership information increases the control overhead specially in large area networks. Also, the interchange of routing tables between neighbors causes the protocol not

to scale well to the number of multicast groups as PBM [12].
In [13] a zone based multicast routing protocol (HSZMP) is proposed to perform scalable multicast routing. this protocol constructs a two-teir zone based structure and perform election algorithm to elect zone representative. The zone representative is responsible to process the control packets, forward the data packets and maintain the multicast membership. HSZMP uses greedy forwarding mechanism to forward the data packets in the network based on the current nodes locations. In this mechanism the next zone is determined by comparing the distance between the sending zone and the target zone. This mechanism is scalable, however, the periodic beacons lead to network congestion and nodes energy consumption. Also, the probability of finding the shortest path is reduced, especially in sparse networks.
In HSZMP, the multicast session is initiated by broadcasting a message to the entire network. The multicast members inside each zone reply to the zone representative which forward the replies back to the source node, which use greedy strategy to find the routing paths. HSZMP use the neighbor nodes of the multicast members as backup nodes and perform periodic communication with these nodes to be used when the routing path fail. This incurs extra overhead and inefficient in scalable networks.

3. Proposed Work

This section discusses the proposed Scalable Position-based Multicast Routing Protocol (SPMRP) in detail. We first give an overview of the proposed protocol. Then, the network clustering strategy and network maintenance are described. Next, we explain the location service and multicast tree formation. Finally, our position-based route discovery mechanism, route reply, data delivery and the route maintenance mechanism are introduced.

3.1 Protocol overview

SPMRP is a source-tree multicast routing protocol proposed to provide scalable multicast routing over MANETs. SPMRP aims to be implemented in large networks with large number of multicast members. To achieve this, virtual clustering strategy has been introduced. This strategy based on partitioning the network into square clusters. Each cluster has a ClusterHead(CH) node elected to maintain information about all the nodes in its cell till they join a new cell.
When a source node wants to send data packets to a particular multicast group, the cluster structure is efficiently used to gather information about the subscribing nodes and provide the source node with this information. SPMRP reduces the number of nodes participating in forwarding route discovery packets through using Restricted Directional Flooding (RDF) based on nodes' position information. Using this mechanism eliminates broadcast storm and efficiently utilizes the network resources.

3.2 Network Setup

In this section, the needed steps to deploy the network are explained. The main objective of building this virtual structure is to maintain a stable network hierarchy to be utilized to perform efficient routing. The virtual structure also helps in coping with increasing number of nodes; i.e., having a more scalable routing protocol.
Network setup phase consists of two major steps: partitioning the network into virtual clusters and the election of a ClusterHeads(CHs) inside each cluster. The network partitioning is discussed in subsection 2.1, while subsection 2.2 explains the selection of the clusterheads.

3.2.1 Area Partitioning

The entire network area is divided into an arbitrary number of equal-size virtual clusters. Each cluster has a unique and positive coordinates, such as (C[x,y]). Our design doesn't put any restriction on the shape of the clusters. In theory, the topology can be divided into several (non-overlapping) clusters of any shape. However, by using regular cluster structures (which repeat over the entire area), we can make use of the geometric properties of that shape. Each shape will have its own advantages. A hexagon can closely resemble a coverage area for a given cluster, while a square shape is easier to divide into smaller areas if the density of nodes in a cluster is high. In this work, we consider the square shape for simplicity.
We assume the routing area is a fixed two-dimensional plane for simplicity (move in a fixed territory). This is applicable for many applications in MANETs. Such applications include soldiers in military battlefields, disaster relief scenarios, conferences and public events. Thus, for a given cluster area A, the coordinates of each cluster is also assumed to be fixed. As a result, given the location of a node, it is possible map the node location to the virtual cluster it belongs to. It is also assumed that each node to be equipped with GPS device and, hence, the node position is always available.

3.2.2 Size of the Cluster

The size of the cluster must be considered in cluster-based routing. If size of the clusters is chosen to be small (for example assuming the side length of the cluster is equal to transmission range of individual nodes), the overhead inside each cluster is considerably reduced. Because all nodes inside a specific cluster are within the transmission range of each other, all communications between CHs and regular nodes in a cluster will take place using one-hop only. Additionally, small clusters make the number of CHs increase, having more CHs increases the control overhead resulted from the election process.

On the other hand, if the cluster size is chosen to be large, then position packets sent among CHs of different clusters are reduced since most communications will be local. Also, reducing the number of cells means less number of election processes and, hence, the number of election control packets will be reduced. However, internal overhead increases since communication inside a cluster is carried out using restricted directional flooding instead of one-hop communication. In case of high mobility, the location update packets will increase and probability of the boundary crossing of packets as well. So, using large cluster size effectively reduces the overhead of triggering this kind of packets.

As a result, it has been decided to determine the most suitable cluster size (or most suitable number of clusters) through simulations (Section 5.2.4).

3.2.3 Selection of Cluster Representative

After partitining the network area into several clusters according to location information, an election algorithm is executed inside each cluster independently through cooperation between the nodes in the cluster. Some of the most-valuable nodes among all nodes in the network are selected to take the role of ClusterHeads(*CHs*) and form the virtual backbone. Upon electing the *CHs*, it is important to select the nodes that expected to survive the longest possible time to keep the network construction as stable as possible.

Each cluster has a CH node. The responsibility of the CH node is to receive packets from other CH nodes in the neighbor clusters, and send it to the nodes in its local cluster to maintain network connectivity by CH nodes. Also, the CH is the only node in the cluster that forward and process the control packets. The nodes that are not CLH nodes are called ordinary nodes [1]; each node is identified by (Node_ID, CLR_ID). These nodes are attached with only one CH node and they assigned less functions than the CH nodes. Ordinary nodes has a neighbor table which contains information about the neighbor nodes.

Since the elected *CH* should be the most-valuable node among all the nodes in a specific cluster [2], the following metrics are taken into consideration upon the *CH* selection: the distance of the node from the cluster center (D_{CC}), battery remaining life time (B_n), node's speed (S_n), nodes computational ability(C_n) and available memory(M_n). Each of these metrics is assigned a weighting factor. Considering these metrics essintially, results in fair election. Selecting the node closer to the cell center maintains the *CH* node for longer time. This is because the node has longer distance away from the side of cluster, hence it takes more time to roam out of this region. On the other hand, selecting a node with less battry life would result in immediate drain of its remaining power. This results in another election before the scheduled timeout and increasing the probability of node failure. Moreover, mobility speed of each node is an important metric. When *CHs* with low mobility gets elected, it will take more time to leave the cell, so that it will not initiate new election within short period of time.

Furthermore, selecting a node with high processing power and large memory significantly affects the network performance and results in less processing delay. Network nodes like laptops have more computational potentiality and sufficient memory than simple PDA or mobile phones.

In each cluster, a node floods an election packet to all nodes in the cluster (*Hello*(*Node_ID*)). All nodes in the cluster reply by sending a *Prob*(*Node_ID*, *Pr*) packet after it calculates its probability to be a leader (*Pr*) according to equation (1). To reduce the collision, a random timer is used to delay the propaget of the *Prob* packet. Each node compares the probabilities it receives from other nodes inside the cluster. The node with the highest probability will be chosen as a *CH* node for its cluster.

The probability (*Pr*) of node *i* in cluster *X* to be elected as a *CH* is given as:

$$Pr_{[i]} = Wd \times (1 - \frac{D_{cc}}{Dmax}) + Ws \times (1 - \frac{S_n}{Smax}) + Wb \times (\frac{B_n}{Bmax}) + Wc \times (\frac{C_n}{Cmax}) + Wm \times (\frac{M_n}{Mmax})$$

Where:

Wd: weight of distance between a node and middle point of a cluster boundary,
Ws: weight of node movement speed,
Wb: weight of node battery remaining life,
Wc: weight of node *CPU* power,
Wm: weight of node memory capacity,
Dmax: maximum possible distance between a node and middle point of a cluster boundary,
Smax: maximum possible node movement speed,
Bmax: maximum possible battery life time,
Cmax: maximum *CPU* power found in the market,
Mmax: maximum memory capacity exists in the market.

Values of the weights *Wd*, *Ws*, *Wb*, *Wc* and *Wm* are chosen as follows since we believe that speed, distance and battery lifetime are the most important when selecting the *CH*.

$Wd = Ws = 0.25$
$Wb = 0.2$
$Wc = Wm = 0.15$

Distance D_{CC} between node's position $P_n = (X_n, Y_n)$ and middle point (Xm_{cs}, Ym_{cs}) of cluster c is given as:

$$D_{cc} = \sqrt{(X_n - Xm_{cs})^2 + (Y_n - Ym_{cs})^2}$$

Dmax is calculated once as distance between the cluster center and any point on cluster side. Referring to Fig. 1, *Dmax* can be calculated, for example, as the distance between the middle point (Xm_{13}, Ym_{13}) of boundary 3 in zone 1 and one of the zone corners in front of it ((Xc_{11}, Yc_{11}) or (Xc_{14}, Yc_{14})).

$$Dmax = \sqrt{(Xc_{11} - Xm_{13})^2 + (Yc_{11} - Ym_{13})^2}$$

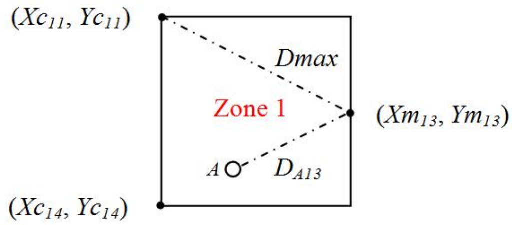


Fig. 1: The maximum possible distance between the middle point of a zone boundary and a node inside the zone

Smax is a pre-defined value that depends on the environment where the protocol is deployed. *Bmax*, *Cmax* and *Mmax* are pre-defined values that depend on the current technology found in the market.

Each elected *CH* declares its leadership by sending a *NEW_CH* packet only to the nodes inside its cluster. It also sends a *NEW_CH_NBR* packet to the *CH* nodes of the neighbor clusters; rather than flooding it to all the *CHs* in the network. This helps in reducing the number of control packets and the overhead associated from maintaining information about the global network.

CH_ID[x,y] $\longrightarrow$ *Cluter_Nodes*[x,y]: [*NEW_CH*, *CH_ID*, *C*[x,y]]

CH_ID[x,y] $\longrightarrow$ *C*[z,w]:[*NEW_CH_NBR*,*CH_ID*[x,y],*C*[x,y],*CH_ID*[z,w], [z,w],*CH_Pos*[x,y],*CH_Pos*[z,w], *Dist, Uni*]

When a node receives the *NEW_CH_NBR* packet, it checks if it is a *CH* for one of the neighbor clusters of the sending cluster. If this is true, it stores information about the neighbor *CH* node. Otherwise, it computes the distance to the intended destination and compares it with the distance included in the packet field *"Dist"*. If the computed distance is less than the old value of "*Dist*", the new value replaces the old value and the packet is forwarded until it reaches the intended *CH* node. Otherwise, the packet is discarded.

When the field "*Uni*" is enabled, the nodes that receive *NEW_CH_NBR* packet will drop it, because this means that this packet is unicasted to the neighbor *CH* node using 1-hop communication.

3.3.3 Periodic election and CH failure

1.1 When the node capability decreased or *CH* moves out the cluster, the *CH* will send *CH_ RETIRE* packet to the candidate node of its cluster. The candidate nodes receiving the *CH_ RETIRE* packet compute its current capability and elect new *CH* as disucssed before.

1.2 The CH sends a beacon packet (BEACON) every interval (T_{CH}) to announce its leadership. If the time (T_{CH}) is elapsed without receiving this packet from the *CH*, then candidate nodes will discover the *CH* failure and initiate a new election process to elect new *CH*. Fig. 2 shows a general overview about the proposed model.

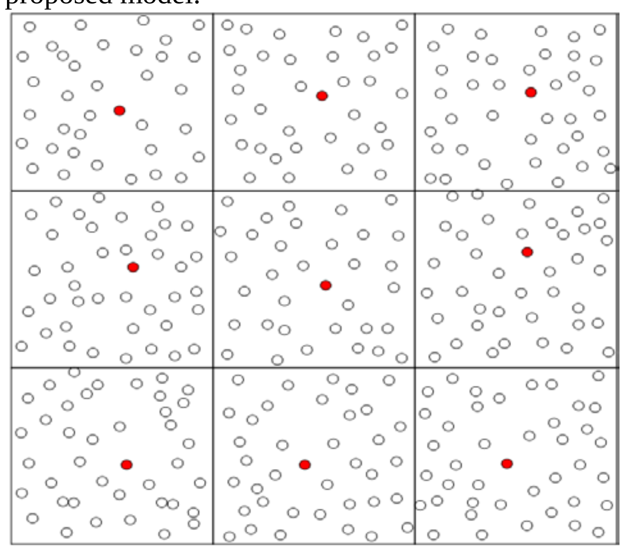


Fig 2. General overview of the network architecture.

4. Route Discovery Process

When a source node has data to be sent to a multicast group, the following steps take place:

A) When a source node decides to initiate a multicast session, a *Source Invitation Request (S_INV_REQ)* packet is first directed to its local *CH* node to ask for possible participating nodes in the held multicast session. This packet needs only 1-hop communication operation if the side length of each cluster is chosen to be R/2 or less, otherwise it is sent via *Restricted Directional Flooding (RDF)*. Using *RDF* gives high probability of finding a path compared to greedy. This also reduces the resulting overhead compared with blind broadcasting to the entire network.

If the source node is the *CH* of the cluster, there is no need to initiate such a packet; *CH* nodes start immediately with step B).

If a source node *S* at *C[X,Y]* is asking for the nodes that are interested to join a session related to a multicast group number *G_ID*, then the following *S_INV_REQ* packet is sent:

$IP_S \longrightarrow CH[X,Y]$: [*S_INV_REQ*, *G_ID*, IP_S, Seq_s, *C[X,Y]*,*Uni*, *Dist*]

The first two fields are the *G_ID* field that represents the *ID* of the multicast group, and the IP_S field that represents the *IP* of the source node. Each node in the network has Seq_s which is increased monotonically with each invitation packet. The fields (IP_S and Seq_s) are used to uniquely identify each invitation packet. The field *C[X ,Y]* represents the source cluster number.

The field *Dist* is used to store the distance between the previous node and the destination. The field *Uni* is used to indicate the type of forwarding (unicast or restricted). If the intended *CH* is reached directly from the source, only 1-hop unicast operation is needed to forward this packet to the *CH* node. In this case the *Uni* field is set to 1 in order to notify the intermediate nodes to drop this packet. Also there is no need to include the *Dist* field.

If the intended *CH* is not within transmission range of the source, the packet is forwarded using *RDF* mechanism towards the *CH*. In this case the *Uni* field is set to 0. Also, the source node calculates the distance between itself and the local *CH* and stores it in the *Dist* field.

When an intermediate node receives *S_INV_REQ* packet with *Uni* field is set to 0, it computes the distance between itself and the destined *CH* node and compares it with the "*Dist*" field which is stored in the packet. If the intermediate node is further than the previous node, the packet is dropped. Otherwise, it stores its previous hop node to be used in the reverse path. Also, it modifies the *Dist* field to represent the distance between itself and the destined *CH* node. The intermediate node then adds its *IP* address (IP_I) to the packet to be used by its next hop node. For example, node *I* will forward the following packet:

$IP_S \longrightarrow CH[X,Y]$: [*S_INV_REQ*, *G_ID*, IP_S, Seq_s, IP_I, *C[X,Y]*,*Uni*, *Dist*]

The *S_INV_REQ* packet continues to be propagated restrictedly until it reaches the intended *CH*.

When an intermediate node receives a *S_INV_REQ* packet with a (IP_S, Seq_s) pair that has been processed previously, the packet will not be processed again.

B) When the source *CH* node (*CH* in the same cluster where the source node resides) receives the *S_INV_REQ* packet, it sends an *External Invitation Request (E_INV_REQ)* packet to the four neighbor *CH*s. For example, the following *E_INV_REQ* packet is sent to neighbor cluster *C[x,y]*:

$CH[X,Y] \longrightarrow CH[x,y]$: [*E_INV_REQ*, *G_ID*, IP_S, Seq_s, *C[X,Y]*, *CH[X,Y]*, *CH[x,y]*, *Pos_CH[x,y]*, *Uni*, *Dist*]

The *G_ID*, IP_S, Seq_s and *C[X,Y]* fields are same as in *S_INV_REQ*. The field *CH[X,Y]* represents the *IP* address of the cluster head that is currently forwarding the packet, and the fields *CH[x,y]*, and *Pos_CH[x,y]* represent the *IP* address and position of the neighbor cluster head. The field *Dist* is used to store the distance between the previous node and the destination. The field *Uni* is used to indicate the type of forwarding (unicast or restricted). These fields are used as with *S_INV_REQ*.

C) The source *CH* also sends an *Internal Invitation Request (I_INV_REQ*) packet searching for possible participating nodes within its cluster. Each node upon receiving the packet will process it only if it is in the intended cluster *C[X,Y]*, otherwise the packet is dropped.

$CH[X,Y] \longrightarrow$ *nodes in C[X,Y]*: [*I_INV_REQ*, *G_ID*, IP_S, Seq_s, IP_I, *C[X,Y]*, *Brd*]

The *G_ID*, IP_S, Seq_s and *C[X,Y]* fields are same as in *S_INV_REQ*

The field IP_I represents the *IP* address of the node that is currently forwarding the packet (which is set initially as *CH[X,Y]*).

This packet is sent using 1-hop communication if the cluster side length is chosen to be R/2 or less, otherwise it is sent using cluster broadcast (*Brd* is set to 1). In 1-hop communication the broadcast field (*Brd*) is set to 0 and each node processing the packet will not forward it. In cluster broadcast, upon processing the packet the node stores the *IP* address of its previous hop (IP_I) to be used in the reverse path. Also, it modifies the IP_I field to be its own *IP* address and continues forwarding the packet.

D) Upon receiving an *E_INV_REQ* for the first time, the intended neighbor *CH* continues the route discovery process by finding a route between itself and the neighbor *CHs* (by sending *E_INV_REQ)*; and later between itself and other destinations in its cluster (by sending *I_INV_REQ)*. The *E_INV_REQ* packet is propagated until it reaches all the network clusters using the forwarding strategy as will be discussed in section 2.

4.1 External Invitation Request Packets Propagation

The proposed protocol utilizes the network division to forward the invitation packets to discover the anticipated group members with very low overhead as well as to prevent sending duplicate packets. In this subsection, the forwarding of *E_INV_REQ* packet between the network clusters is explained. The decision to forward the *E_INV_REQ* packet to neighbor clusters is the responsibility of the *CH* node.

Referring to Figure 3, assume that the source node resides in cluster *C[4,4]*. Firstly, the *E_INV_REQ* packet is forwarded towards the border of the four neighbor clusters as a first forwarding step (in our example, clusters *C[3,4]*, *C[4,3]*, *C[4,5]* and *C[5,4]*).
If each cluster receiving the invitation packet resends it to all its 4 neighbors, a lot of duplicate packets are definitely produced. To overcome this situation, an efficient forwarding strategy is proposed. This algorithm enables the *CH* of each cluster to take part in delivering the packet to at most two neighbor clusters. In this forwarding, the *CH* based on the number of the source cluster *C[X,Y]*, and the coordinates of the intermediate cluster that is currently forwarding the packet *C[x,y]*. This forwarding strategy insures that the *E_INV_REQ* packet is propagated through the network with no duplicates and all the network clusters are visited only once (refer to Fig. 3).
For example, assume that the packet is sent out from cluster *C[4,4]*. Here, the *CH* node of clusters *C[1,4]*, *C[2,4]* and *C[3,4]* forward the packet to the clusters that are above and to the right of the current cluster. In a following step, clusters *C[1,5]*, *C[2,5]* and *C[3,5]* send the packet only towards clusters to their right. A similar strategy is used for packets forwarding to other network parts to eliminate duplicate packets.

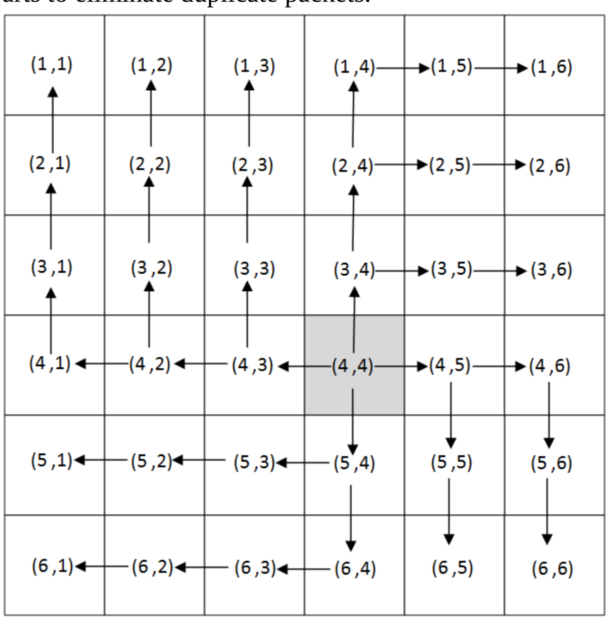


Figure 3: Forwarding E_INV_REQ packet.

4.2 Route Setup

The next step, after propagating the request packets, is to setup the routes via sending the reply packets. The following steps are carried out during this phase:
A) After forwarding the *I_INV_REQ* packet and if it is interested in participating in the session, node *J* starts setting up a route from the local *CH* (*CH* of the cluster where node *J* currently resides) to itself by sending an *Internal Invitation Reply* (*I_INV_REP*) packet.
Each intermediate node forwards this packet to the node from which it received the corresponding *I_INV_REQ* packet; i.e., the *I_INV_REQ* with the same (IP_S, Seq_s) tuple. This process continues till the packet reaches the intended *CH*. For example node *J* will send the following packet:

$$IP_J \longrightarrow CH[x,y] : [I\_INV\_REP,\ IP_S,\ Seq_s,\ IP_J,\ IP_I,\ C[x,y]]$$

This reply packet contains IP_S and Seq_s to specify that this packet is a reply to the original unique *I_INV_REQ* packet. It also contains the *IP* address of the node interested in joining the session (IP_J), the *IP* address of the node from which it received the corresponding *I_INV_REQ* packet (IP_I), the number of the cluster where the node currently resides *C[x,y]*.
B) To reduce the overhead in the network, each cluster head *CH[x,y]* sends only one *External Invitation Reply* (*E_INV_REP)* to the neighbor *CH* that forwarded the original *E_INV_REQ* to it (*CH[z,w]* for example). If the *CH* itself is interested in the conducted session, it sends the *E_INV_REP* packet immediately after forwarding the request packets (*E_INV_REQ* and *I_INV_REQ*). Otherwise, it sends the *E_INV_REP* packet upon receiving the first *E_INV_REP* or *I_INV_REP* packet.
This packet is sent using the reverse path until it reaches the *CH* node that issued the original *E_INV_REQ* packet. The format of the *E_INV_REP* packet from *CH[x,y]* to *CH[z,w]* is:

$$CH[x,y] \longrightarrow CH[z,w]: [E\_INV\_REP,\ IP_S,\ Seq_s,\ C[x,y],\ C[z,w]]$$

C) To further reduce the overhead in the network, the source cluster head *CH[X,Y]* sends only one *Source Invitation Reply (S_INV_RE)P* to the source node *S*. If the *CH* itself is interested in the conducted session, it sends the *S_INV_REP* packet immediately after forwarding the request packets. Otherwise, it sends the *S_INV_REP* packet upon receiving the first *I_INV_REP* or *E_INV_REP* packet. Each node sends this packet to the previous hop from which it received the original *S_INV_REQ* packet, till the packet reaches node *S*.

$$CH[X,Y] \longrightarrow IP_S: [S\_INV\_REP,\ IP_S,\ Seq_s,\ C[X,Y]]$$

5. Performance Evaluation

In MANETs, evaluating and testing a routing protocol is a mandatory phase to ensure its success in the real world applications. To perform this evaluation, researchers have four options: using test-beds, emulators, analytical modeling or using simulation tools. The performance of the proposed protocol is evaluated through developing both an analytical investigation and extensive simulation using the GloMoSim simulator environment [13]. In this section, we study the performance of SPMRP by simulations. We are interested to study the protocol's scalability and robustness in a dynamic environment.

Simulation Overview

We implemented the EGMP protocol using Global Mobile Simulation (GloMoSim) [14] library, and compare it with ODMRP [15]. ODMRP is probably one of the most well-studied protocols. The major advantage of ODMRP is that it produces high packet delivery ratio and considered to be robust over dynamic networks. The disadvantage of ODMRP is that the high control overhead introduced. Also, we choose to compare the performance of our protocol with the position-based multicast protocol SPBM [20] which is designed to improve the scalability of multicast routing.

The simulation models a MANET network of nodes placed randomly within a square terrain area for 600s of simulation time. The initial positions of the nodes are chosen randomly, after that all nodes are granted the full mobility with node density of 60nodes/km2. Each node is uniquely identified and its identity is maintained during the network lifetime.

In the conducted simulations, it was assumed that all nodes have identical and fixed radio transmission range of 250m. The IEEE 802.11 MAC layer and Constant Bit Rate (CBR) traffic over User Datagram Protocol (UDP) have been used with a maximum channel capacity of 2Mb/s. Each source sends CBR data packets at 8 Kbps with packet length 512 bytes. The parameters of ODMRP followed the specification in the Internet Draft *draft-ietf-manet-odmrp-02.tx* (IETF Internet Draft, 2011). For each simulated scenario, each data point represents the average of five runs with identical configuration but different initial seeds.

The effect of three important parameters of Ad-Hoc network has been tested. These parameters are node mobility speed, area size and group size. For each parameter three performance metrics are evaluated. These metrics are:

1. **Packet Delivery Ratio (PDR):** The ratio between the number of multicast data packets delivered to all multicast receivers and the number of multicast data packets supposed to be delivered to multicast receivers. This ratio represents the effectiveness of the multicast routing protocol.
2. **Packet Routing Load (PRL):** The ratio of control packets transmitted to data packets delivered. This ratio investigates the efficiency of utilizing the control packets in delivering data packets. Sending a control packet over one link is counted as one packet. For example, if a control packet traverses a route of N hops, N packets are counted.

Effect of Node Mobility Speed

It is still critical and challenging for a multicast routing protocols to maintain good performance in the presence of node mobility in an Ad hoc network. The pause time is fixed at 30s. We evaluate the protocol performance by varying maximum moving speed from 5 to 40 m/s.

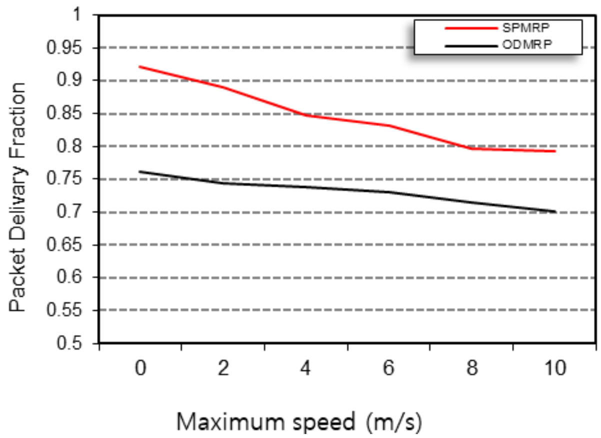
Fig 4. (a) show the effect of mobility on performance of three routing protocols. The maximum node speed varies from0 to 40 m/s with a fixed pause time at 30s. As expected, PDR for the all protocols is very sensitive to mobility and as the node mobility increases the delivery ratio decreases. This is expected, since fast movement of the nodes increases the probability of link failure and topology change which leads to higher packets' dropout.

As depicted in Fig.4 (a), our protocol is able to achieve higher packet delivery ratios and maintain acceptable levels even when the node mobility increase compared with ODMRP and SPBM. This is as expected, since using restricted geographic forwarding enhance transmission of packets and it is more robust to the network topology and can adjust more quickly to the topology change.

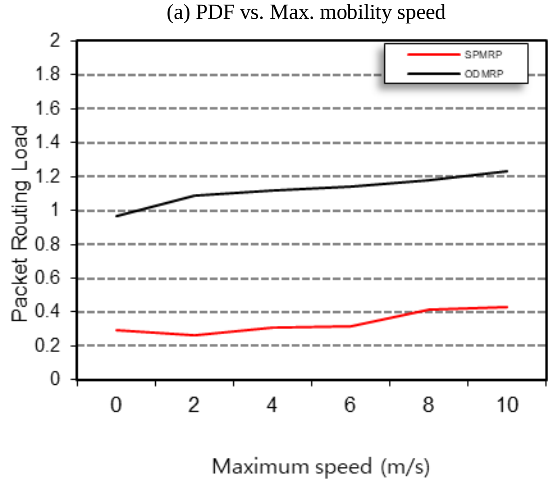
On the other hand, ODMRP shows lower performance. This is due to the mesh structure of ODMRP which provides multiple paths. this mesh structure built through the back learning scheme which make is easier to become invalid as the nodes move. In SPBM, periodic multilevel membership maintenance mechanism leads to more collision in the network as the nodes mobility increase which lead to the reduction of its delivery ratio.

Fig. 4(b) represents the packet routing load as a result of mobility conditions. The control overhead of both protocols increases as mobility increases. As depicted (fig.4). SPMRP has lower control routing overhead than those of SPBM and ODMRP at different moving speeds. SPMRP, does not broadcast the routing packets to whole area. In SPMRP the packets is sent using restricted directional flooding towards the destination; this is the reason behind reducing the overall routing load.

For ODMRP, NPO is slightly increases when the mobility speed increases. This is because the periodic Join_Query packets are flooded out at the same rate for different mobility speeds. When the nodes move faster, the next hop is more likely to move away, which makes the reverse routes learned through Join_Query are not reliable. So, if the Join_Reply is sent to the next hop that is no longer available, it gets dropped. That causes the Join_Reply to be sent several times without finding the next hop. Then, the node broadcasts another Join_Query packet to search for a route to the source node. All neighbor nodes receiving such packet need to generate their own Join_Reply packets, which causes the increase in normalized packet overhead when the mobility is increased.



(a) PDF vs. Max. mobility speed



(b)PRL vs. Max. mobility speed

Fig. 4 Effect of mobility.

6. Conclusion

This paper has presented SPMRP, a novel clustering scheme that supports multicast routing in MANETs. SPMRP exploits the geographic information to construct a virtual cluster backbone to handle the dynamic topology network. This cluster is utilized to perform a location service algorithm without duplicate packets. Then an on-demand multicast tree is constructed between the source and all destinations using the location information of the mobile nodes. We conducted a performance study of the proposed protocol using simulation. The results demonstrate the efficiency of the proposed protocol in supporting large-scale networks and maintaining a stable routing topology.

Acknowledgment

The author wish to thank Palestine Technical University Kadoorie, Palestine for their cooperation and support to publish this research.

References

[1] SK. Meera, D. Kumar & D. Rao, "Global Journal of Computer Science and Technology: E Network, Web & Security", vol 16, no 4,2016.

[2] Y. S. Yen, H. C. Chao, R. S. Chang, and A. Vasilakos, "Flooding-limited and multi-constrained QoS multicast routing based on the genetic algorithm for MANETs," Mathematical and Computer Modelling, vol. 53, no. 11, pp. 2238-2250, 2011.

[3] J. Huang and Y. Liu, "MOEAQ: A QoS-aware multicast routing algorithm for MANET," Expert Systems with Applications, vol. 37, no. 2, pp. 1391-1399, 2010.

[4] X. Xiang, X. Wang, and Y. Yang, "Stateless multicasting in mobile ad hoc networks," IEEE Transactions on Computers, vol. 59, no. 8, pp. 1076-1090, 2010.

[5] KasunSamarasinghe, Ricardo Wehbe and Pierre Leone, “Greedy Zone Routing: Robust and Scalable Routing in Wireless Ad-hoc Networks”, 2016 30th International Conference on Advanced Information Networking and Applications Workshops, pp 557-564.

[6] S. H. Yang and L. Bao, "Scalable mobility management in large-scale wireless mesh networks," in IEEE Wireless Communications and Networking Conference (WCNC 2011), Cancun, Mexico, 2011, pp. 1230-1235.

[7] A. S. Nargunam and M. Sebastian, "Hierarchical multicast routing scheme for mobile ad hoc network," 2007, pp. 464-475.

[8] L. K. Qabajeh, M. Kiah, and M. M. Qabajeh, "A scalable and secure position-based routing protocol for ad-hoc networks," Malaysian J. Comput. Sci, vol. 22, no. 2, pp. 99-120, 2009.

[9] J. N. Al-Karaki and A. E. Kamal, "Efficient virtual-backbone routing in mobile ad hoc networks," Computer Networks, vol. 52, no. 2, pp. 327-350, 8 February 2008.

[10] M. Vijayalakshmi, D. Sreenivasa Rao, "Cluster based Multicast Adhoc on Demand Routing Protocol for Increasing Link Stability in Manets", vol 17, no 2,2017.

[11] S. Jin, C. Park, D. Choi, K. Chung, and H. Yoon, "Cluster-based trust evaluation scheme in an ad hoc network," ETRI journal, vol. 27, no. 4, pp. 465-468, 2005.

[12] SK. Nagula Meera, D. Srinivasa Kumar, D. Srinivasa Rao, "ECMST: Minimal Energy Usage Competent Multicast Steiner Tree based Route Discovery for Mobile Ad hoc Networks", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 10 (2016) pp 6970-6975.

[13] UCLA Parallel Computing Laboratory. Network Simulator (GloMoSim). Available: http://pcl.cs.ucla.edu/projects/glomosim/

[14] Mohammad Qabajeh, Aisha Adballa, Othman Khalifa, and Liana Qabajeh, "A Cluster-based QoS Multicast Routing Protocol for Scalable MANETs " KSII Transaction on Internet and Information Systems, vol. 5, no. 4, pp. 741-762, April 2011.

[15] IETF Internet Draft, "On-Demand Multicast Routing Protocol (ODMRP) for AdHoc Networks," ed. draft-ietf-manet-odmrp-02.txt, 2011.
[16] Tracy Camp, "Location Information Services in Mobile Ad Hoc Networks," The Colorado School of Mines, Technical Report MCS-03-15, October 21 2003.
[17] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," Ad Hoc Networks, vol. vol. 1, no. no. 4, pp. 351-369, 2003.
[18] Dingde Jiang et al., “An Energy-Efficient Multicast Algorithm with Maximum Network Throughput in Multi-hop Wireless Networks” Journal of Communications and Networks, vol. 18, no. 5, pp 713-724, october 2016.
[19] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in 6th Annual International Conference on Mobile Computing and Networking (MOBICOM), 2000, pp. 243-254.
[20] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (DREAM)," in 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), Dallas, TX, USA, 1998, pp. 76-84.

Mohammad M. Qabajeh (1976) received his B.Sc. in Computer Engineering from Palestine Polytechnic University, Palestine in 2000 and M.Sc. in Computer Engineering from Jordan University of Science and Technology, Jordan in 2006. In 2012 he received his PhD in Computer Engineering from International Islamic University Malaysia, Malaysia. Since 2013, he is an Assistant Professor at Department of Computer Science, Palestine Technical University Khoori, Palestine. His current research interests are Distributed Systems and Ad hoc Networks.