# Multicast Routing in Mobile Ad hoc Networks Using Source Grouped Flooding

**Karthikeyan Chandrashekar and John S. Baras**[1]

**Institute for Systems Research,**
**University of Maryland,**
**College Park, Maryland, USA.**
**e-mail: {karthikc,baras}@isr.umd.edu**

***Abstract*: In this paper, we address the multicast routing problem for mobile ad hoc networks (MANETs). We present the Source Grouped Flooding approach to achieve multicast in MANETs. In this protocol, each source creates a flooding group consisting of nodes connecting the source to the multicast members. The nodes in the flooding group are recruited based on hop count distance constraints obtained during a request-reply phase. The flooding group though robust may result in redundant data transmissions. We also propose a probabilistic *data* forwarding mechanism to achieve efficient data dissemination. The protocol aims to achieve the robustness of flooding and data distribution efficiency of tree based protocols. Simulation results verify performance.**

## 1. Introduction

Mobile ad hoc networks are usually flat networks comprised of mobile wireless devices. The ease and speed of deployment of these networks makes them ideal for situations where fixed infrastructure is not readily available (e.g. battlefield communications, disaster recovery). Limited bandwidth, energy constraints and unpredictable dynamic topologies pose difficult problems for the design of applications for these networks. Multicast applications like video conferencing and subscription services have become very popular with the advancements in current technology. Multicast is an important communication paradigm in ad hoc networks due to the inherent broadcast nature of the medium. Multicast routing protocols for ad hoc networks are either tree based or mesh based. Tree based protocols like [10, 8, 13] achieve efficient data distribution by creating a tree structure. However, these protocols suffer when the network is highly dynamic as the tree structure is fragile and does not provide any redundant paths. Mesh protocols like [7, 9, 11] create a mesh structure and therefore are robust against network dynamics due to redundant transmission of data. Clearly the data distribution will not be efficient. Flooding a network is equivalent to creating a mesh structure incorporating all the nodes in the network. Hence flooding will be highly robust against topology changes. Flooding is typically used to achieve network wide broadcast and therefore it can also be considered as a multicast protocol.

The Source Grouped Flooding protocol is designed to provide robustness similar to that of flooding i.e. to create a stable multicast structure at high node speeds. Hop count distance metrics between the source and the group members are used to recruit nodes in the flooding group for the source. The flooding group incorporates redundant paths between the source and the group members, and the size of the group can be controlled by varying the hop count constraints. The hop counts are updated each time the source initiates a request-reply phase to update the nodes in the flooding group. At the same time, the protocol improves the efficiency of data delivery by using a probabilistic data forwarding mechanism based on the hop counts of the nodes in the flooding group. For an extensive description of the approach refer to [3].

[1]This work was partially supported by contracts DARPA F3060200020510, DARPA MDA 9720010025, Telcordia Technologies 10073139, Lockheed Martin Corporation and the Maryland Industrial Partnership Program.

## 2. Related Work

Some of the tree based protocols are: The Adhoc Multicast Routing using Increased Sequence ids (AMRIS) [13] protocol creates a shared multicast tree structure rooted at a special node (Sid). Nodes adapt to connectivity changes based on id numbers obtained from the Sid. A multicast extension to Adhoc On-demand Distance Vector (MAODV) [10] creates a shared multicast tree rooted at the group leader which periodically updates routes through destination sequence numbers. The Adhoc Multicast Routing Protocol (AMRoute)[8] creates a user level shared multicast tree consisting of unicast tunnels between the group members. Some of the mesh based protocols are: The On-demand Multicast Routing Protocol (ODMRP)[7] creates a mesh of nodes connecting the sources and the group members. Multiple paths provide stability against topology changes. The Core Assisted Mesh Protocol (CAMP) [9] relies on affiliations to core nodes to create multicast structure. The core nodes forward the data. Flooding as a multicast protocol is discussed in [4], in this paper the authors point out that under high mobility conditions, flooding is the most reliable protocol for achieving one to many communications. The broadcast storm problem which addresses the overhead of flooding is discussed in [12].Here, the authors have proposed several schemes like counter based, location based retransmissions to reduce the redundant data retransmissions. Gossip based protocols like [6] are used as auxillary protocols to reduce the overhead of the flooding. This paper describe a method to reduce the redundant transmissions resulting from flooding of control information in most MANET protocols (eg.; AODV route requests ...). The protocol uses a probabilistic mechanism for controlling the number of retransmissions, where the retransmission probability is experimentally tuned. In contrast to these ap-

proaches our probabilistic mechanism reduces the number of *data* retransmissions and the probability distribution is adaptive to the current state of the flooding group i.e.; the hop counts of the nodes and the number of duplicate packets will determine the retransmission probability. Thus the retransmission probability is different for different peer levels and adapts as the membership of the source created flooding group changes.

## 3. Source Grouped Flooding Protocol

This is an on-demand protocol that creates and maintains a mesh of nodes called the *flooding group* based on hop count distance metrics. Nodes in the network learn these metrics during a request-reply phase.

### 3.1. Creation of the flooding group

#### 3.1.1. Request Phase

When a source 's' has packets to send to a multicast group it initiates the request phase by broadcasting a JOIN REQUEST message. The request message contains the *multicast group address* and a *hop count* field. When a node 'n' in the network receives a non-duplicate request packet, it stores the *hop count* for that source ($D_{sn}$) i.e., the hop count of the node from the source. The node then increments the hop count and re-broadcasts the packet. This is illustrated in Figure 1(a). 'S' is the source and 'M1' and 'M2' are the multicast members. The number in each node indicates hop count distance to the source 'S'. A combination of the source address and a counter is used as a unique packet identifier to identify duplicate packets. An active source will periodically update the flooding group every *refresh_interval* seconds. Thus during the request phase all nodes in the network learn the hop counts to the source and update this information.

#### 3.1.2. Reply Phase

A multicast group member 'm' upon receiving the JOIN REQUEST, stores the hop count distance to the source $D_{sm}$, waits for a short fixed interval and then broadcasts a JOIN REPLY message. This small delay ($10ms$) prevents collision of the request and the reply messages in the region of the group member. The JOIN REPLY contains the multicast group information and the hop count distance from the group member to the source. The TTL (Time To Live field in the IP header) for this message is set to the hop count from the source ($D_{sm}$). This ensures that the reply message does not propagate beyond the source. Thus even though the reply message is broadcast it will propagate within a fixed radius. When a node receives a JOIN REPLY the node will compare its stored hop count to the source (stored during the request phase $D_{sn}$), and the value in the *hop count* field of the reply message ($D_{sm}$). If the hop count distance constraint (1) is satisfied the node becomes a *flooding node* else the packet is dropped. The nodes marked 'FN' in Figure 1(b) are the flooding nodes for the source 'S'. The propagation of the reply message is limited by the distance constraint (2), where ($D_{mn}$) is the distance of the current node 'n' from the group member 'm'. Only nodes that are activated as flooding nodes, propagate the reply message. Moreover, once a node becomes a flooding node during a particular route refresh sequence, it no longer re-broadcasts a reply message for that route refresh phase. Therefore, a node will re-broadcast only the first reply message for each source during a particular refresh sequence. The protocol thus creates the flooding group for each source consisting of nodes that satisfy hop count distance constraint (1); the set of nodes being determined by constraint (2). Constraint (2) directly follows from the fact that the group member sets the TTL in the reply message to $D_{sm}$, which was obtained during the request phase. Each source thus creates its own *flooding group*, connecting the source to all the group members. The source maintains a different *flooding group* for each multicast group, as the group membership is different for different groups.

$$D_{sn} \leq D_{sm} \tag{1}$$

$$D_{mn} \leq D_{sm} \tag{2}$$

where $D_{sm}$, $D_{sn}$, $D_{mn}$ are as described above.

Controlling the flooding group membership with the above relaxed distance constraint could lead to large flooding groups per source, as can be seen in Figure 1(b). An ideal flooding group would be one that consists of nodes that form the shortest paths between the source and the group members. We derive the following distance constraints recognizing that a node lies in the shortest path between a source and a member if the sum of the node's distance to the source and the node's distance to the member is less than or equal to the distance between the source and the member.

$$D_{sn} + (D_{sm} - TTL_{rep}) \leq D_{sm} \Rightarrow D_{sn} \leq TTL_{rep} \tag{3}$$

$D_{sm}$ is the initial value of the TTL in the reply message sent by the member, and $TTL_{rep}$ is the decremented value of TTL in the reply message that the node receives. Thus ($D_{sm}$ - $TTL_{rep}$) is the hop count distance between the node and the group member. The nodes use the reduced form of this constraint to decide to join the flooding group and thus only the nodes that form the shortest path can become members of the flooding group. This is illustrated in Figure 1(c); clearly only the nodes in the shortest path between the source and the members become flooding nodes. As before, the propagation of the reply messages is controlled by the distance constraint (2). If multiple shortest paths exist then all nodes in these paths are included in the flooding group. Thus, the reduced constraint limits the size of the flooding group while ensuring that the shortest path(s) between the source and the members are always included.

### 3.2. Data Forwarding

#### 3.2.1. Hop Count Data Forwarding

When a source sends data packets to a multicast group, the nodes in the flooding group for this source will be the only nodes that re-transmit or forward the data packet. All duplicate packets identified based on source address and a counter value are dropped. In order to ensure that

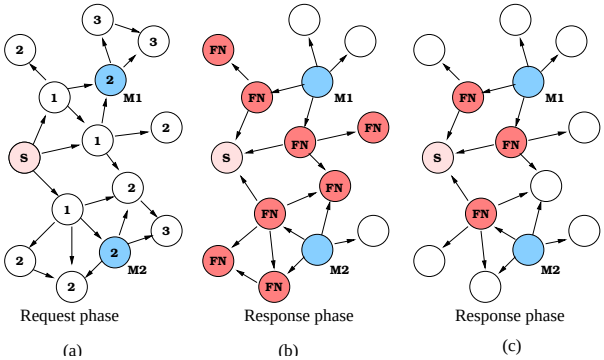


Figure 1: Flooding Group Formation

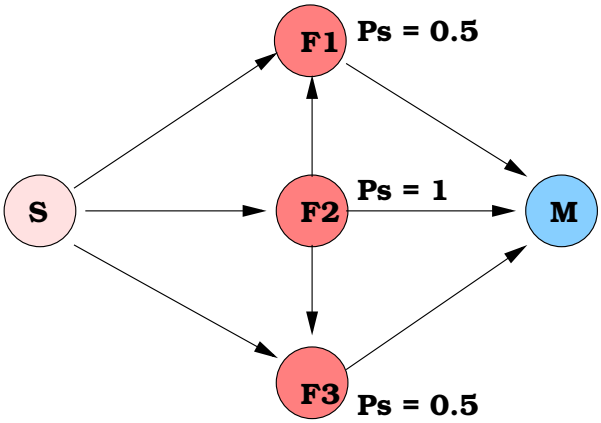


Figure 2: Probabilistic Forwarding of data

the data wave progresses towards the group members from the source, a *hop count* field is introduced in the data packet. The source initializes the hop count to 0 and each node updates this value with its stored hop count to the source before retransmitting the packet. When an active flooding node receives a data packet, it compares its latest hop count value for this source ($D_{sn}$) with the hop count field in the data packet. The node re-broadcasts the packet only if the stored hop count is greater than or equal to the hop count value in the packet. The node stores its hop count distance to the source in the data packet before retransmitting it. This mechanism ensures that MAC layer contention and collision is reduced by avoiding the propagation of the data packet in the same region more than once.

### 3.2.2. Probabilistic Data Forwarding

The *flooding group* provides multiple paths from the source to the group members. Redundant transmission of data along these paths will improve data delivery, however it will result in excessive overhead. We propose a probabilistic data forwarding mechanism to reduce data overhead and describe a method to determine a meaningful value for the retransmission probability ($P_{send}$) of a packet. The above described hop count forwarding is used to determine the distance of the data packet from the source. In this scheme, when a node receives a non-duplicate data packet, it stores the packet, and waits for a short random interval of time $(3-10ms)$ for arrival of duplicate packets. The node increments a counter for every data packet received from a node in its peer distance level from the source, i.e., data packets having hop count value same as this node's stored hop count value. All other duplicate data packets are dropped. When the wait interval is over, the node calculates the retransmission probability of the packet using (4). The node decides to retransmit the packet with probability $P_{send}$ and drop the packet with probability $(1 - P_{send})$. Once the wait interval is over, all duplicates irrespective of hop count value will be dropped. Thus the probability of a data packet being retransmitted adapts to the density of the flooding group and the hop count distance of the nodes in the flooding group.

$$P_{send} = \frac{1}{1+n} \tag{4}$$

where, *n* is the number of duplicate packets received from the same hop count peer level.

Figure 2 demonstrates the benefit of the probabilistic forwarding scheme. Source S is connected to member M through flooding nodes F1, F2 and F3 that form the shortest paths between S and M. When the source S transmits a packet, F1, F2, and F3 receive the packet. Let us assume, node F2 times out first and transmits with probability 1. Nodes F1 and F3 which are in the same peer hop count level will increment their duplicate counters upon receiving the packet from F2. Thus F3 and F1 will retransmit the packet with probability 0.5. Thus the number of retransmissions is potentially reduced and at the same time, at least one packet is forwarded in each peer hop count level ensuring that the member receives the packet.

## 4. Simulation Setup and Results

### 4.1. Simulation setup

OPNET 7.0 [2] discrete event engine was used to simulate our algorithms. The simulation modeled a network of 50 nodes randomly placed within a $1000m$ x $1000m$ area. Nodes in the network move according to the "Billiard Mobility" model [1]. In this model the wait time is 0. Nodes choose a random direction and move in that direction with a fixed speed until they reach the boundary. Upon hitting the boundary nodes choose another random direction and move in that direction with the same fixed speed. At the physical layer, radio propagation distance for each node was set to $250m$ and the shared channel capacity was $1Mbps$. Our model does not support radio capture [5] so, in the case of packet collisions all packets are dropped. The IEEE 802.11 (DCF) was used as the Medium Access Control (MAC) protocol. The communication medium is broadcast and nodes have bi-directional connectivity. Group members and sources are randomly chosen from the nodes in the network. A source generates CBR traffic at $2packets/secs$ with each packet having a payload of 128 bytes. Each simulation was run for 100 seconds. Multiple runs were conducted with different seed values for each scenario and the collected data were averaged over these runs. The multicast algorithms were developed as separate OPNET routing layer protocols.

The performance of the following schemes are evaluated:

- flooding: flooding as a multicast routing protocol is used as a baseline.
- basic-sgfp: this scheme uses the relaxed or basic distance constraints (1) and (2) to create the group and hop count data forwarding.
- sp-sgfp: this scheme uses the shortest path distance constraints (3) and hop count data forwarding.
- p-sgfp: this scheme uses relaxed distance constraints and probabilistic data forwarding.
- psp-sgfp: this scheme uses shortest path distance constraints and probabilistic data forwarding.

The following simulation metrics are considered for comparing the schemes:

- Packet Delivery Ratio: the ratio of the number of data packets received by the group members to the number of data packets expected to be received by the group members (number of packets sent by the source times the number of members).
- Total Overhead: is defined as the ratio of the total packets transmitted in the network (control + data) to the number of data packets received by the group members.

### 4.2. Simulation Results

Figures 3 and 4 show the Packet Delivery Ratio (PDR) and the Total Overhead as a function of node speed $(0 - 30m/s)$. The network has $5$ sources and $20$ group members. The refresh interval is $4$ seconds. The *flooding* scheme has the best PDR performance (around $95\%$) for all mobilty speeds as every node rebroadcasts every packet. Redundant data transmission contributes to total overhead and this remains constant against mobility as every node retransmits the packet. All the source initiated schemes show a linear decrease in packet delivery with increased mobility speed; this is to be expected as the movement of the nodes will disrupt the flooding group resulting in loss of packets. However, it should be noted that even at node speeds of $30m/s$ the PDR is around $84\%$ indicating that the flooding group is a very robust multicast structure. The total overhead of the probabilistic schemes is less than that of flooding. Particularly, the total overhead of psp-sgpf is $20\%$ less than that of flooding. Thus the source initiated multicast protocol using shortest path flooding groups and probabilistic data forwarding achieves comparable robustness to flooding while significantly reducing the total overhead. The total overhead for the basic-sgfp scheme is more that of plain flooding because the basic-sgfp scheme creates a large flooding group (almost all nodes in the network). Therefore, the flooding group setup by basic-sgfp incurs nearly the same data overhead as generated by plain flooding. The added control overhead of setting up the flooding group results in basic-sgfp having a higher total overhead than plain flooding.

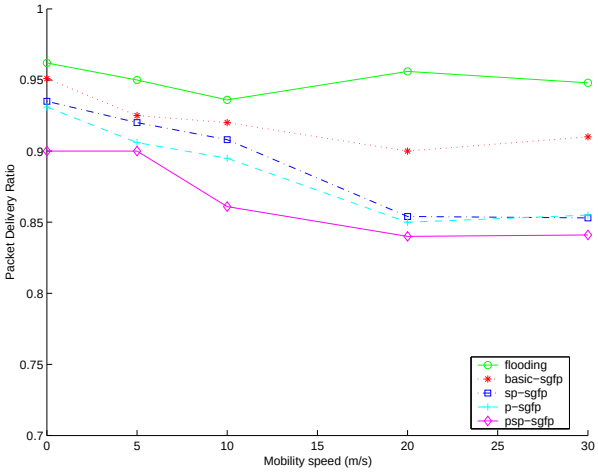


Figure 3: Packet Delivery Ratio vs Mobilty Speed

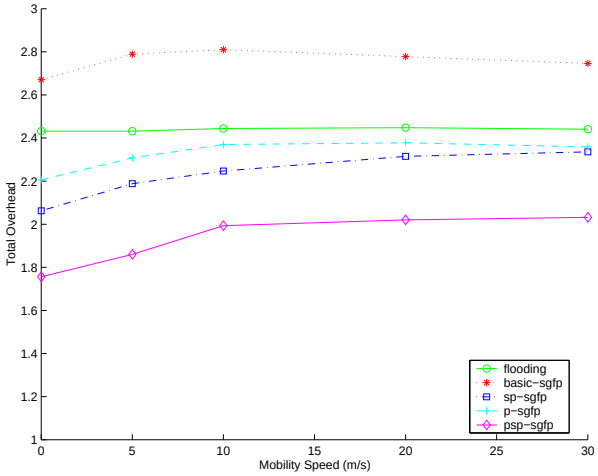


Figure 4: Total Overhead vs Mobilty Speed

Figures 5 and 6 show the Packet Delivery Ratio (PDR)and the Total Overhead as a function of the number of sources $(1 - 20)$. Node mobility was set to $5m/s$. The network had $20$ group members. The refresh interval is $4$ seconds. The PDR decreases linearly with increase in the number of sources. As the number of sources increases, more data and control packets are generated. This causes increased MAC layer collisions resulting in loss of data packets and outdated flooding groups. In spite of this, the source grouped schemes have a packet delivery ratio that decreases linearly with a small gradient. Thus, the behaviour of the source grouped schemes is similar to that of flooding as the number of sources increases. The total overhead for all the schemes remains the same. This is because the total overhead is a function of the number of data packets delivered. As the number of sources increases, more control and redundant packets are generated, and at the same time the number of data packets delivered increases proportionally. Therefore the total overhead remains the same. The source initiated schemes imitate the performance of flooding. The psp-sgpf scheme achieves efficient data distribution while maintaining a comparable Packet Delivery Ratio to flooding.

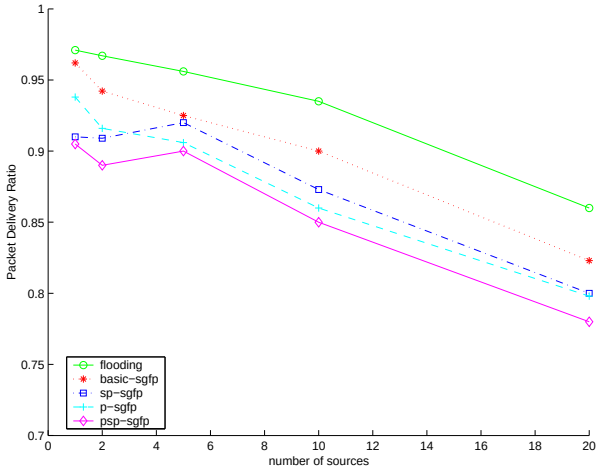


Figure 5: Packet Delivery Ratio vs Number of Sources

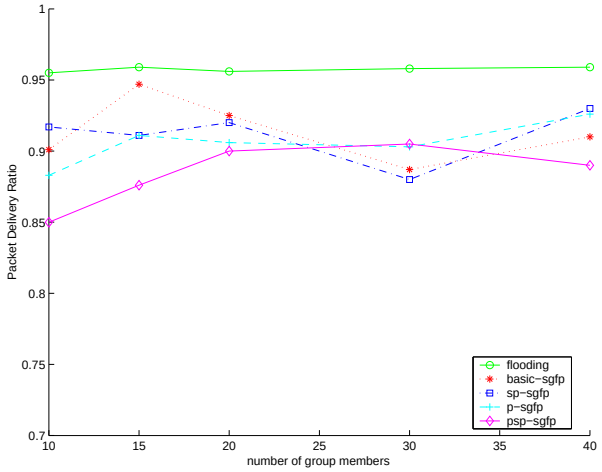


Figure 7: Packet Delivery Ratio vs Multicast Group Size

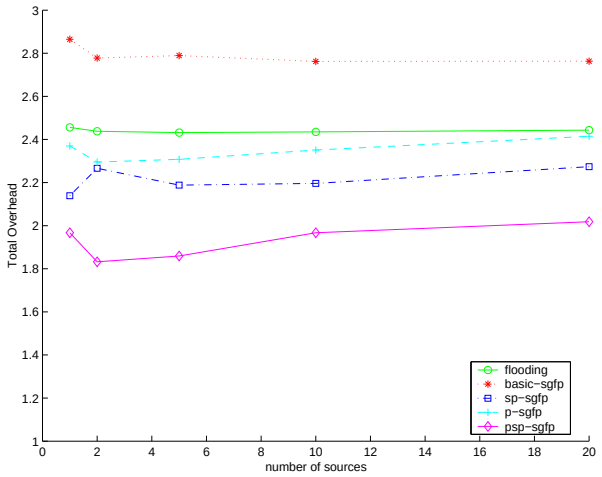


Figure 6: Total Overhead vs Number of Sources

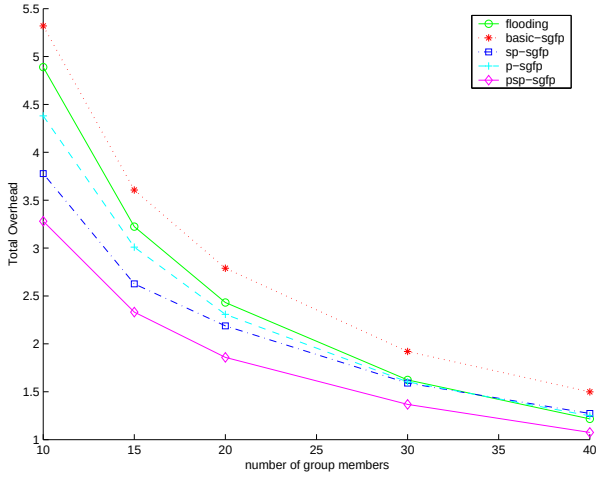


Figure 8: Total Overhead vs Multicast Group Size

Figures 7 and 8 show the Packet Delivery Ratio (PDR) and the Total Overhead as a function of the multicast group size ($10 - 40$). Node mobility was set to $5m/s$. The network had $5$ sources.The refresh interval is $4$ seconds. PDR for the flooding scheme remains constant as the group size increases. Since every node rebroadcasts the packet, every node receives the packet irrespective of whether it is a group member or not. The source initiated schemes have packet delivery performance within $10\%$ of that of flooding. Particularly, the PDR for *psp-sgfp* is around $90\%$ as the group size increases. This is because of the efficient data distribution achieved due to the shortest path flooding group and probabilistic data forwarding. The total overhead decreases for all the schemes as the group size increases. This is because the total overhead is a function of the number of packets delivered and clearly as the member size increases, the number of packets delivered increases. We see that the overhead for all the schemes converges, this is because as the group size increases multicast resembles broadcast.

Figure 9 shows the tradeoff between the Packet Delivery Ratio and the total overhead as a function of the refresh interval i.e. the frequency of flooding group update. The network had $5$ sources, $20$ group members and the nodes moved at $5m/s$. This interesting curve shows the impact of the refresh interval on the packet delivery ratio and the total overhead in the same graph. Clearly as the refresh interval increases, the total overhead will reduce as the flooding groups are reinforced less frequently. At the same time, we see that the Packet Delivery Ratio remains almost the same as the refresh interval increases, particularly for the psp-sgfp scheme. This indicates that the flooding group is a stable multicast structure and need not be reinforced very often. Therefore when the refresh interval is $8secs$, the psp-spfp scheme can achieve comparable packet delivery to that of flooding while having a $40\%$ lesser overhead than that of flooding.

## 5. Conclusions

The inherent constraints of MANETs viz mobility, bandwidth and energy limitations pose difficult challenges in designing multicast routing protocols. Thus, it is necessary for a multicast protocol to not only be efficient but also be robust against mobility and other network dynamics. In this paper we have described a novel way of creating the multicast structure based on hop-count distance metrics and also have described a method

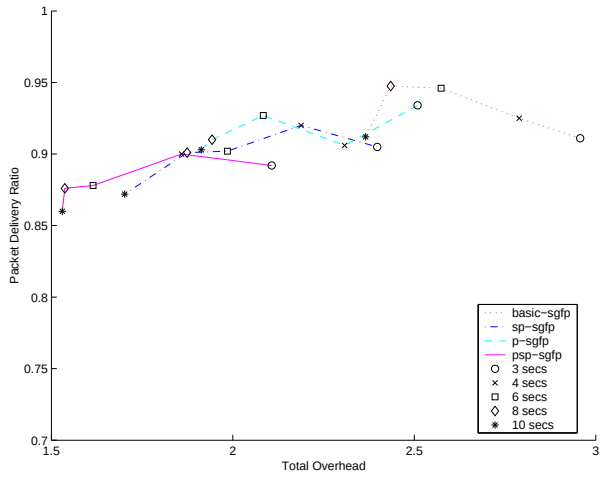


Figure 9: Trade-off curve for refresh intervals

to obtain a meaningful retransmission probability for the probabilistic data forwarding mechnism. The Probabilistic Shortest Path Source Grouped protocol (PSP-SGFP) described in this paper achieves robustness similar to that of flooding while at the same time considerably improving the data delivery efficiency. The steady packet delivery performance of the protocol even at high node speeds ($30m/s$) proves the robustness of the flooding group multicast structure. At the same time the total overhead is $20\%$ less than that of plain flooding. Moreover, the tradeoff curve as a function of the refresh interval indicates that the protocol can be $40\%$ more efficient than plain flooding without compromising robustness. The protocol provides a highly robust multicast structure for a wide range of node speeds while achieving significant reduction in overhead.

## REFERENCES


[1] Billiard mobility . http://w3.antd.nist.gov/wctg/manet/prd_aodvfiles.html.

[2] Opnet modeler version 7.0. www.opnet.com.

[3] K. Chandrashekar. Multicast routing in mobile wireless ad hoc networks using source grouped flooding. Master's thesis, University of Maryland, College Park, MD, 2002 www.glue.umd.edu/ karthikc.

[4] C.Ho, K.Obraczka, and G.Tsudik K.Vishwanath. Flooding for reliable multicast in multi-hop ad hoc networks. In *MobiCom Workshop on Discrete Algorithms and Methods for Mobility*, 1999.

[5] C.Ware, T.Wysocki, and J.F.Chicharo. Simulation of capture behaviour in ieee 802.11 radio modems. *Journal of Telecommunications and Information Theory*, 2001.

[6] Zygmunt J. Haas, Joseph Y. Halpern, and Li Li. Gossip-based ad hoc routing. In *Proceedings of IEEE INFOCOM*, August 2002.

[7] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-demand multicast routing protocol. In *Proceedings of IEEE WCNC*, pages 1298–1304, New Orleans, LA, September 1999.

[8] M. Liu, R. Talpade, A. McAuley, and E. Bommaiah. AMRoute: Ad hoc multicast routing protocol. Technical Report 8, University of Maryland, 1999.

[9] E. L. Madruga and J. J. Garcia-Luna-Aceves. Scalable multicasting: The core assisted mesh protocol. *ACM/Baltzer Mobile Network and Applications Journal, Special Issue on Management of Mobility*, 1999.

[10] E. Royer and C. E. Perkins. Multicast operation of ad hoc on-demand distance vector routing protocol. In *Proceedings of MobiCom*, Seattle, WA, August 1999.

[11] P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: Multicast core extraction distributed ad hoc routing. In *Proceedings of the Wireless Communications and Networking Conference*, 1999.

[12] S.Y.Ni, Y.-C.Tseng, Y.-S.Chen, and J.-P. Sheu. The broadcast problem in a mobile ad hoc network. In *Proceedings of MobiCom*, August 1999.

[13] C. W. Wu and Y. C. Tay. AMRIS: A multicast protocol for ad hoc wireless networks. In *Proceedings of IEEE MILCOM*, Atlantic City, NJ, November 1999.