Efficient Flooding in Ad hoc Networks: a Comparative Performance Study

Yunjung Yi and Mario Gerla
Computer Science Department
University of California
Los Angeles, CA 90095

***Abstract*— Many ad hoc network protocols (e.g., routing, service discovery, etc.) use *flooding* as the basic mechanism to propagate control messages. In flooding, a node transmits a message to all of its neighbors. The neighbors in turn transmit to their neighbors and so on until the message has been propagated to the entire network. Typically, only a subset of the neighbors is required to forward the message in order to guarantee complete flooding to the entire network. If the node geographic density (i.e., the number of neighbors within a node's radio reach) is too high, one can easily see that flooding can become very inefficient because of redundant, "superfluous" forwarding. In fact, superfluous flooding increases link overhead and wireless medium congestion. In a large network, with heavy load, this extra overhead can have severe impact on performance and should be eliminated.**

Many efficient flooding schemes have been recently proposed in ad hoc networks. In this paper, we compare the performance of a set of representative schemes via simulation using as criteria the flooding efficiency and the delivery ratio.

I. INTRODUCTION

Many ad hoc network protocols (e.g., routing, service discovery, etc.) use *flooding* as the basic mechanism to propagate control messages. In flooding, a node transmits a message to all of its neighbors. The neighbors in turn transmit to their neighbors and so on until the message has been propagated to the entire network. In this paper, we call such flooding as *blind flooding*. As one can easily see, the performance of blind flooding is closely related to the average number of neighbors (neighbor degree) in the CSMA/CA network. As the neighbor degree gets higher, the blind flooding suffers from the increases of (1) redundant and superfluous packets, (2) the probability of collision, and (3) congestion of wireless medium [1]. Consequently, performance of blind flooding is severely impaired especially in large and dense networks [2].

When topology or neighborhood information is available, only a subset of neighbors is required to participate in flooding to guarantee the complete flooding. We call such flooding as *efficient flooding*. The characteristics of MANETs (e.g., node mobility, the limited bandwidth and resource), however, make colleting topological information very difficult. It generally needs huge extra overhead due to the periodic message exchanges or event driven updates with optional deployment of GPS (Global Positioning System)-like system. For that reason, many on-demand ad hoc routing schemes and service discovery protocols simply use blind flooding [9] [20] [21]. With periodic route table exchanges, proactive ad hoc routing schemes, unlike on-demand routing methods, can gather topological information without big extra overhead (through piggybacking topology information or learning neighbors). Thus, a few proactive ad hoc routing mechanisms proposed route aggregation methods so that the route information is propagated by only a subset of nodes in the network [16] [17].

Various efficient flooding schemes have been recently proposed in the MANETs [1] [5] [6] [7] [11] [12] [13] [14]. We classify and evaluate those protocols in a realistic common environment. Through simulation study, we provide quantitative performance analysis of efficient flooding schemes with different node mobility and the node geographic density. In [23], the authors compare efficient flooding protocols. However, it does not include efficient flooding schemes based on underlying cluster architecture and the source tree. Our work further investigates those schemes as well as selective protocols introduced in [23].

In total five flooding mechanisms are simulated in diverse network scenarios: F-MPR (Flooding with Multipoint Relay), F-AC (Flooding with Lowest Id clustering algorithm), RPF (Flooding with Reverse Path Forwarding), PC (Passive Clustering) and BF (Blind Flooding). First, we change the average neighbor degree of each node by adjusting the physical network size. Our main interest is to examine a relationship between the performance of each protocol and the network density in terms of the delivery ratio of a flood packet and the reduction rate of re-broadcast packets (flooding efficiency). And, we investigate the impact of node mobility on performance of each protocol by varying node mobility. Finally, we apply efficient flooding mechanism to a reactive routing protocol, AODV [9]. The purpose of this application is to show that efficient flooding improves the scalability of a protocol that uses massive flooding as a basic mechanism to propagate its protocol-specific information. Also, we want to verify that the ranking and tradeoff among flooding schemes established in the first set of experiments is valid also in the AODV application.

The organization of the rest part of the paper is as follows. We will briefly describe the existing efficient flooding schemes in Chapter II. And, we demonstrate the contributions of our work through extensive simulation studies in Chapter III and Chapter IV. Finally, we conclude the paper in Chapter V.

This work is supported in part by ONR "MINUTEMAN" project under contract N00014 - 01 - C - 0016

II. BRIEF OVERVIEW OF EFFICIENT FLOODING

In this section, we present a brief review of key efficient flooding protocols proposed for MANETs. We categorize those schemes into “heuristic-based protocol”, “topology-based protocol” and “cluster-based protocol”. The “topology-based protocol” is sub-categorized into “neighbor topology based protocol” and “source-tree based protocol”, and the “cluster-based protocol” is sub-classified into “active clustering” and “passive clustering”.

Note that the problem of finding, in a distributed way, a subset of dominating forwarding nodes in MANETs is NP-complete [1]. Thus, all works about efficient flooding have been focusing on developing a sub-optimum solution that chooses a sub-optimal dominant set with low overhead.

A. Heuristic-based Protocol

[1] [5] proposed several heuristics to reduce the number of rebroadcasts. In their idea, upon receiving a flooding packet, a node decides whether this node relays the packet to its neighbor or not using one of following heuristics: (1) probabilistic scheme where this node rebroadcasts the packet with the randomly chosen probability; (2) counter-based scheme where this node rebroadcasts if the number of received duplicate packets is less than a threshold (e.g., threshold =2); (3) distance-based scheme that uses the relative distance between hosts to make the decision; (4) location-based scheme based on pre-acquired location information of neighbors.

B. Topology-based Protocol

Another approach of efficient flooding is to exploit topological information [5] [6] [7] [11] [12] [13] [14]. With node mobility and the absence of pre-existing infrastructure in the ad hoc network, most works use the periodic *hello message* exchange method to collect topological information.

1) Neighbor Topology Based Protocols: [7] and [5] propose *self-pruning* and *neighbor-coverage* scheme based on one-hop neighbor information. With self-pruning scheme, each forwarding node piggybacks the list of neighbors of itself on outgoing packet. A node rebroadcasts (becomes a forwarding node) only when this node has neighbors not covered by afore forwarding nodes. The basic idea of neighbor-coverage scheme and self-pruning is same. However, neighbor-coverage scheme does not piggyback neighbor list on outgoing broadcast packets. Instead, a node propagates its neighbor list upon exchanging hello messages by piggybacking. Thus, self-pruning scheme increases control overhead as the frequency of flooding increases compared with neighbor-coverage scheme. On the other hand, however, self-pruning is more resilient to mobility because of fresh neighbor list.

The *Multipoint Relay* [6] scheme extends the range of neighbor information to two-hop away neighbors. In *Multipoint Relay* scheme (MPR), a node periodically exchanges the list of adjacent nodes with its neighbors so that each node can collect the information of two-hop away neighbors. Each node, based on the gathered information, selects the minimal subset of forwarding neighbors, which covers all neighbors within two-hop away. Each node piggybacks its chosen forwarding nodes (MPRNs) on the hello messages. A node re-broadcasts a flood packet if the sender chooses this node as a MPRN. Otherwise, this node does not relay the flood packet. To choose a sub-optimal subset of forwarding nodes, in MPR, a node p first collects the set of neighbor nodes within two hops, N_p^2. Now, with a set of uncovered nodes T_p = N_p^2 - N_p -$\{p\}$, where N_p is the set of neighbors of p, p selects MPRNs in two steps. First, p selects a set of neighbor nodes $\{q$, where q is the only neighbor of some node $r \in T_p\}$, and removes the neighbors of the set $\{q\}$ from T_p (i.e., T_p = T_p - $\cup$ N_q). And then, p chooses a neighbor node m which has the largest number of neighbors in T_p and eliminates neighbors of m from T_p. It repeats this step until T_p becomes empty.

Besides mentioned works, many schemes have proposed the heuristic to choose a dominant set [11] [12] [13] based on topological information. [12] proposed an algorithm named *Span* to choose a set of coordinate nodes that covers all nodes within two-hop neighbors. Span, however, is different to MPR in that a node aggressively declares as a *coordinate* node (dominating node) if the node detects insufficient coordinate neighbor nodes. [13] proposed a protocol based on the node location knowledge named *GAF*.

Among those protocols, we choose MPR and implement to compare with other protocols. In [7], the authors showed that a scheme based on two-hop topology works better than a mechanism using one hop neighbor information. With this in mind, we limit ourselves to protocols with two-hop neighbor topology.

2) Source-Tree Based Protocols: The source-tree based flooding [17] is another scheme using topological information. With the source-tree, a node re-broadcasts the flood packet only if this node is not a leaf node in the source-tree whose root is the source node. In this paper, we use the reverse path forwarding scheme [18] to construct the source tree. In reverse path forwarding protocol, each node re-broadcasts if this node is not a leaf node in the spanning tree formed by the minimum-hop paths from all nodes to the source node. To construct and manage a source tree, each node updates its tree status upon receiving a packet from neighbor nodes and periodically performs blind flooding initiated by the source node. Each node piggybacks (addr_parent, addr_mine, numHop) information, where *addr_parent* is the address of previous hop of the minimum-hop path from the source node to this node, *addr_mine* is the address of this node and *numHop* is the number of hops of the path from this node to the source node. For sake of simplicity, in this paper, we assume only one source node. With multiple sources, the tree maintenance mechanism becomes more complex (e.g., ALP (Adaptive Link-state routing Protocol)[19], TBRPF (Topology Broadcast Based on Reverse-Path Forwarding)[17]).

C. Cluster-based Protocol

Clustering is another method to select forwarding nodes as addressed in [1]. Clustering in this paper can be described as *grouping nodes into clusters*. A representative of each group

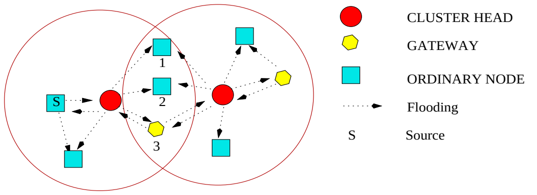

Fig. 1. An example of Efficient Flooding with Clustering. Only cluster heads and gateways rebroadcast and ordinary nodes stop forwarding.

(cluster) is named as a *cluster head* and a node belonging to more than two clusters at the same time is called a *gateway*. Other members are called *ordinary nodes*. A cluster is defined by the transmission area of the cluster head. We use a 2-hop clustering where any node in a cluster can reach any other node in the same cluster with at most 2 hops as defined in [8]. With an underlying cluster platform, non-ordinary nodes can be the dominant forwarding nodes as in Fig. 1.

1) Active Clustering: A node with the lower ID among neighbor nodes (Lowest ID algorithm [3]) or the highest neighbor degree (Highest Degree algorithm [3]) can become a cluster head. In this paper, we use LID (Lowest ID algorithm) to support efficient flooding. To decide the lowest ID node, each node exchanges hello message with node address. After collecting neighbor information, each node broadcasts the cluster status with (ch_id, node_id), where *ch_id* is the address of the cluster head who has the lowest ID among neighbors (ch_id $\leq$ node_id). To stabilize clustering, each node broadcasts its cluster status if this node has received the cluster declaration packets from neighbor nodes that have the lower ID than *node_id*. Thus, a packet loss from the lower ID can block this node from proceeding *cluster declaration*.

LID algorithm, however, tends to generate too many gateways and works inefficiently in the dense network, since all nodes are reachable from more than two cluster heads at the same time are gateways by LID. For example, node “1”, “2”, and “3” in Fig. 1 are gateways and forwarding nodes. Thus, we refine LID algorithm to choose a minimum number of gateways to be applied for efficient flooding. We use the heuristic of MPR scheme. A *cluster head* chooses the list of gateways and sends that list when it broadcasts the cluster information. A *cluster head* chooses a subset of nodes among neighbors which covers up all of nodes within two hop away. A *cluster head* broadcasts the list of *gateways* by piggybacking the chosen set of nodes on the clustering broadcast packet. We can easily prove that those selected gateways are enough to guarantee the complete coverage with assumption of the reliable packet delivery. Like MPR scheme, each node piggybacks the neighbor list on hello messages to exchange two hop neighbors' information.

2) Passive Clustering: Passive clustering [15] eliminates setup latency and major control overhead of active clustering required to collect neighbor information. Instead of exchanging neighbor information through extra control packets, passive clustering exploits on-going traffic to propagate cluster status of each node. Each node piggybacks “cluster-related information” (e.g., the state of a node in a cluster, the IP address of the node) and collects neighbor information through promiscuous packet receptions.

With the *First Declaration Wins* rule of passive clustering, a node that first claims to be a *cluster head* “rules” the rest of nodes in its clustered area (radio coverage). Concurrent declarations can be resolved with the *lowest ID wins* rule where the lower ID node wins and becomes a *cluster head*. Thus, there is no waiting period (to make sure all the neighbors have been checked) unlike that in all the weight-driven clustering mechanisms [3] [4].

Furthermore, without extra message exchanges, passive clustering provides a procedure, *Gateway Selection Heuristic*, to elect the minimal number of gateways (including distributed gateways) required to maintain the connectivity in a distributed manner.

Passive clustering maintains clusters using implicit timeout. A node assumes that some nodes are out of locality if they have not sent any data longer than timeout duration. With reasonable offered load, a node can catch dynamic topology changes.

III. SIMULATION STUDIES

We simulate flooding protocols using Global Mobile Simulation (GloMoSim) library [10], which is a scalable simulation environment for wireless networks. Our aim of simulation study is to investigate the impact of the neighbor degree and node mobility on the performance of each efficient flooding scheme. Thus, we first simulate static networks (i.e., no node mobility) by increasing the node geographic density and then vary node mobility with the fixed neighbor degree.

Among existing flooding protocols, we choose following schemes, which are representatives of different approaches, to study the performance: (1) flooding with MPR (F-MPR), a mechanism based on two hop neighbor topology; (2) flooding with active clustering (F-AC), flooding with clustering; (3) flooding with passive clustering (F-PC), flooding with low overhead clustering; (4) flooding with reverse path forwarding (F-RPF), efficient flooding with source-tree; (5) blind flooding (BF). Note that we only compare the flooding efficiency in terms of the flooding application, instead of focusing on various applications that use flooding as a basic mechanism to propagate control messages (e.g., link-state routing). Moreover, we omit several protocols that (1) are threshold-based protocols (such as probability, counter-based [1]; (2) require the location of nodes (such as GAF [13], distance/location-based scheme [1]); (3) choose a set of dominating nodes in a similar way of MPR, where a set of dominating nodes of a node covers the two-hop neighbors of this node (e.g., Span [12]).

For simulation, we use UDP(User Data Protocol), IEEE 802.11 DCF and two-ray propagation model. The radio propagation of each node reaches up to 250 meters and channel capacity is 2 Mbits/second. The random-way point model is used for node mobility. To illustrate flooding efficiency, we employ a new flooding application where one random source initiates 4 flood packets/sec with default size 100 bytes. Each simulation runs for 200 seconds. The results are averaged over 20 randomly generated node topologies. F-MPR and F-AC

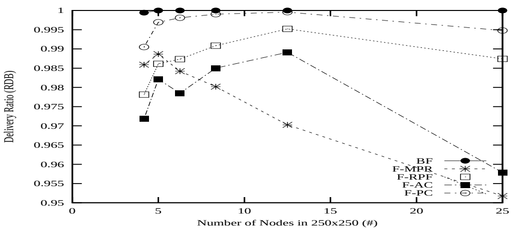

Fig. 2. The RDB of each protocol with single source and data rate 4pkts/sec .

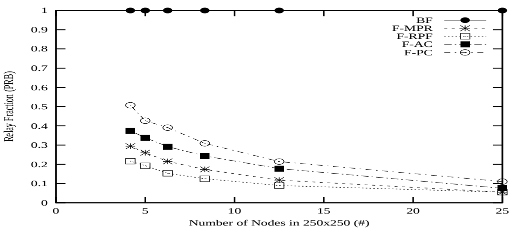

Fig. 3. The PRB of each protocol with single source and data rate 4pkts/sec .

send hello messages in every 2 seconds, and F-RPF performs blind flooding to update the source-tree in every 2 seconds. PC_LID uses 2 seconds cluster timeout. In other words, all entries must be removed from the neighbor list if they are not updated for 2 seconds. The neighbor timeout of F-RPF, F-MPR or F-AC to remove inactive neighbors (or parents) is 6 seconds.

We analyze flooding efficiency in terms of the re-broadcast reduction rate and the delivery ratio. Each metric is computed as follows.

- *PRB (the Probability of Re-broadcast of Broadcast)*: The total number of flood packets forwarded from a node is divided by the total number of issued broadcast packets from the source. For example, RPB = 0.1 means that 10% of total nodes (e.g., 10 among 100) rebroadcast the packet.
- *RDB (the Ratio of Delivered Broadcast packets)* : The total number of delivered broadcast packets to a node is divided by the total number of packets supposed to be delivered to the node.
- *TCPB (the Total Control Packets in Bytes)*: The total bytes of sent control packets from a node. This includes hello messages, clustering packets and piggybacked information (e.g., cluster-related information of passive clustering).

A. No Mobility with Various Network Size

We use the static network and increase the geographic density by reducing physical network size. In this experiment, 100 nodes are placed randomly over "x" x 1000 m^2 terrain where "x" states the horizontal range. We fix the vertical range of the network to 1000 meter and change the horizontal range from 250 to 1500 meter in step of 250. In the results, the "x" axis represents the average number of nodes placed within 250 x 250 m^2.

Fig. 2 shows the RDB i.e., delivery ratio of each scheme as the neighbor degree increases. And, Fig. 3 illustrates the flooding efficiency of each protocol. Note that this result does not include the protocol-specific control overhead (e.g., hello messages, clustering broadcast packets). Active clustering requires 2* $\frac{total\_time}{hello\_interval}$ (one packet for hello message and the other packet for cluster declaration), and MPR and RPF requires $\frac{total\_time}{hello\_interval}$ extra control packets to exchange hello messages.

We observe following results in those figures. First of all, there is a clear ranking and tradeoff of each protocol. The order of the reception probability is F-BF $>>$ F-PC $>>$ F-RPF $>>$ F-AC $>>$ F-MPR and the rank of flooding efficiency is F-RPF $>>$ F-MPR $>>$ F-AC $>>$ F-PC $>>$ F-BF. With keeping a reasonable but lower than F-PC and F-BF delivery ratio, F-RPF outperforms other schemes in terms of flooding efficiency. While, F-PC trades flooding efficiency for high reception rate. We should note that we use single source i.e., very low offered load here. Thus, blind flooding outperforms efficient flooding protocols. However, blind flooding significantly wastes the network bandwidth and also may suffer from heavy contention and collision in the presence of heavy traffic and thus result in low delivery ratio.

Moreover, performance of flooding protocols using MPR (F-MPR and F-AC) is considerably low compared with other schemes. Note that we use "MPR" scheme to reduce the number of gateways in active clustering. This result is mainly caused by "incomplete" neighbor information. In MPR, each node calculates its set of MPRNs and forwards the chosen MPRNs to neighbor nodes with hello messages. The nodes re-broadcast a flood packet only if the neighbor node, the sender (not the source) of the flood packet to this node, chooses this node as a MPR. Thus, the lost hello packets from node "A" to node "B" may mislead node "B" so that the chosen set of MPRNs by "B" does not cover the neighbors of node "A". Also, the lost hello packets from node "B" to "A" may prevent "A" from forwarding a broadcast packet received from node "B" even though node "B" chose node "A" as a MPRN. Whereas, in other protocols, each node stops re-broadcast only if this node certainly belongs to the set of non-dominating nodes. For example, in F-RPF, a node does not relay a broadcast packet only if this node is a leaf node in the source tree. And the source tree is built with blind flooding initiated by the source node. Thus, a node "A", which is the leaf node in the virtual source tree observed by Oracle, will be a forwarding node until it collects all neighbor information so that it can decide that there is no child of this node in the source tree. Thus, the lost hello messages delay the convergence of each node and thus increases the number of floating (unsettled) nodes. In Fig. 3, one can easily see that the number of forwarding nodes of F-RPF increases as the neighbor degree increases. The node geographic density clearly increases the hello packet loss due to heavy contention and collision. For that reason, F-MPR works better in the sparse network than in the dense network as Fig. 2 illustrates.

Secondly, clustering protocols generally deliver more packets than F-RPF or F-MPR does. Since F-AC uses MPR to decide gateways, the performance of AC is closely related to

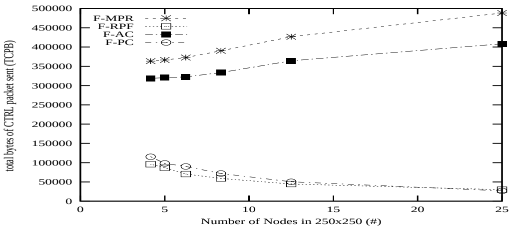

Fig. 4. The TCPB of each protocol with single source and the data rate 4pkts/sec .

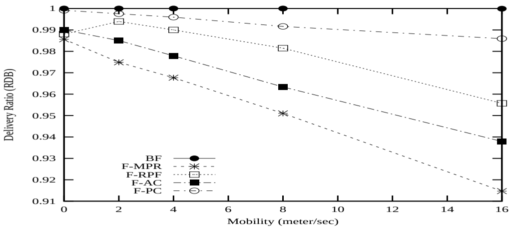

Fig. 5. The RDB of each protocol with single source and the packet rate of 4 pkts/sec. The 100 nodes are place randomly over 1000 x 1000 m^2.

that of MPR and the delivery ratio of MPR is very low. With those in mind, we can infer that the performance degradation of F-AC comes from underlying MPR scheme instead of from clustering mechanism. And F-AC still works better than F-MPR. As seen in Fig. 3, clustering protocols choose more dominating nodes compared with MPR and RPF and thus provide higher reception rate.

Lastly, each efficient flooding scheme has the optimal point where it exhibits the best delivery ratio (e.g., F-PC at # neighbors = 12, F-MPR at # neighbors = 5). With the sparse network (e.g., # neighbors =4), the set of dominating nodes chosen by efficient flooding is not sufficient to cover the whole network. As the network density increases, each protocol chooses comparably more relaying nodes. With Fig. 3, one can notice that each efficient flooding schemes slightly increases average number of relaying nodes as the density increases (e.g., F-PC exhibits 2.1, 2.25, 2.26, 2.6, 3.0 and 2.5 at each point). In very dense network, heavier contention and collision due to periodic hello messages increase the probability of packet loss (both data and hello packets). Since efficient flooding reduces redundant re-broadcasts, unlike blind flooding, the data packet loss cannot be easily recovered. However, F-PC is comparably immune to the increase of neighbor degree because F-PC does not exchange hello packets.

As mentioned earlier, the total number of control packet of each protocol is obvious. Thus, we measure the total control overhead in bytes to compare protocol overhead between flooding schemes. Fig. 4 shows the outcome. This result clearly demonstrates the low overhead of F-PC and F-RPF. Due to piggybacked neighbor information, F-MPR and F-AC produce heavy control overhead and increase the overhead with the neighbor degree. However, we should note that the control overhead of F-RPF is closely related and actually proportional to the number of source. Thus, the control overhead of F-RPF will become bigger with the increase of number of sources. While the control overhead of passive clustering is independent of the number of sources and keeps low.

B. Fixed Network Size with Node Mobility

For the second experiment, we simulate 100 mobile nodes placed randomly within 1000 x 1000 m^2. With the fixed network size, we increase node mobility from 0 m/s to 16 m/s with 10 seconds pause time. The delivery ratio in Fig. 5 shows a few remarkable results. First, the performance of each efficient flooding scheme slightly degrades as the mobility increases. However, passive clustering exhibits very high delivery ratio even in the presence of high node mobility (e.g., 16 m/s). Secondly, MPR shows low delivery ratio because the coverage of periodically updated forwarding nodes is temporarily impaired by dynamic topology changes with high mobility. Lastly, even with node mobility, the rank of each protocol does not change i.e., is same to that of static results (F-PC $>>$ F-RPF $>>$ F-AC $>>$ F-MPR). With this observation, we can conclude that (1) passive clustering works well regardless of node mobility and network density in spite of its low communicational overhead (thanks to piggybacking cluster-related information); (2) MPR suffers from incomplete set of chosen MPRNs due to inaccurate neighbor topological information. And thus, efficient flooding in MANET scenarios should be designed to cope with incomplete neighbor information; (3) RPF works effectively with a few sources. However, it may suffer from heavy overhead as the number of sources increases.

IV. APPLICATION OF EFFICIENT FLOODING

In this section, we apply the flooding protocols to the reactive ad hoc routing protocol AODV (Ad-hoc On-demand Distance Vector) [9]. AODV relies on the massive flooding of the route request packets. Thus, we expect that the efficient flooding scheme will improve the AODV performance. We slightly modify the route information flooding mechanism of AODV so that a node selectively re-broadcasts (relays) a control packet based on underlying flooding mechanism (i.e., F-MPF, F-AC and F-PC). Note that we omit F-RPF since it requires underlying proactive routing scheme to maintain the shared source-trees. This is inconsistent with the use of an on demand strategy such as AODV. We simulate 100 nodes placed randomly within a 1000 x 500 m^2 terrain. Nodes are moving randomly with minimum speed 2 m/s, maximum speed 20 m/s and 100 seconds pause time. We increase the offered load using a number of CBR (Constant Bit Rate) sessions ranging from 10 to 40. Each CBR source starts a session randomly after the initialization time (10 seconds) with the data ratio 4 packets/second and 512 bytes payload size.

Fig. 6 and 7 demonstrate the performance gain with efficient flooding. From Fig. 7, we see that F-MPR and F-AC increase the control overhead due to periodic hello messages. This is most evident when offered load is low since in that case the AODV routing overhead is also low.

With heavy offered load (N $>$ 30), F-MPR outperforms the other flooding protocols. Since F-MPR reduces more re-broadcasts packets of route queries than clustering protocols

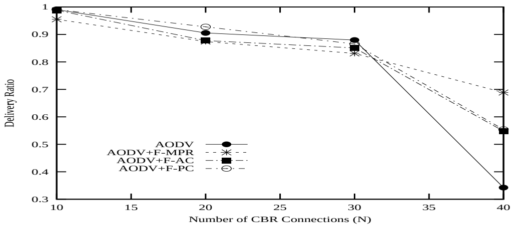

Fig. 6. The Delivery Ratio of each protocol with AODV (100 nodes placed randomly within 1000 x 500 m^2)

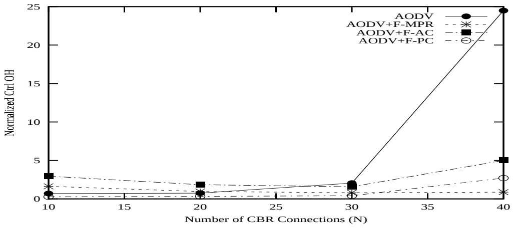

Fig. 7. The Normalized Total Control Overhead including protocol overhead of each protocol with AODV (100 nodes place randomly within 1000 x 500 m^2)

do as shown in previous experiments (see Fig. 3), F-MPR generates less routing packets compared with clustering protocols. With less congestion due to routing overhead, F-MPR exhibits more chances of data transmission and thus higher delivery ratio of data traffic. However, F-MPR overall performs worse than clustering schemes i.e., results in lower delivery ratio with less given traffic (e.g., $N \leq 30$). As we observed through flooding experiments (see Fig. 2), F-MPR fails to achieve high delivery ratio of flooding. For that reason, AODV with F-MPR often fails to find a shortest path due to route query packets loss at intermediate nodes. Instead, it finds a longer path that is more fragile with node mobility than a shorter one. In fact, the average hop of AODV+F-MPR is longer than that of AODV+F-PC (1.77 vs 1.56 at N =10).

In summary, efficient flooding helps the scalability of AODV with the increase of offered load by decreasing the normalized control overhead [22]. Different flood reduction schemes work best in different situations. F-MPR works best at high offered loads, but it shows poor performance at light load and high mobility. Passive clustering shows the best overall performance for broad range of node mobility and offered load. Note that, because of space limitation, we have not shown performance as a function of mobility. However, from Fig. 5 one can easily infer that performance of MPR and AC in the AODV context will be much poorer than that of PC and BF beyond speeds of 8 m/s.

V. CONCLUSION

Through a comparative study of efficient flooding, we have shown the following results.

First, passive clustering performs well for a broad range of node mobility and network density values. It appears to be the most robust scheme overall.

Secondly, accurate neighbor information collection is very hard in ad hoc networks due to unreliable packet delivery, low bandwidth, and node mobility. As a result, a scheme that works effectively only with complete neighbor topology information (e.g., MPR, active clustering) is severely impaired by the increase of neighbor degree or node mobility (as both these factors make it more difficult to maintain neighbor information).

Lastly, every scheme has a different set of suitable applications. For example, F-PC can be well adapted to reactive ad hoc routing protocols because of its low overhead. While F-MPR, F-AC and F-RPF work very effectively with link-state proactive routing protocols where periodic, background overhead is not a critical issue.

REFERENCES

[1] Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan C hen, Jang-Ping Sheu, *The broadcast storm problem in a mobile ad hoc network,* MobiCom, 1999.
[2] Jinyang Li, Charles Blake, Douglas S.J., De Couto, Hu Imm Lee, Robert Morris, *Capacity of Ad Hoc Wireless Networks,* Mobicom, 2001.
[3] C.R. Lin and M. Gerla, *Adaptive Clustering for Mobile Wireless Networks,* IEEE Journal on Selected Areas in Communications, Vol. 15, No. 7, Sep. 1997, pp. 1265-1275.
[4] Gerla, M and Tasi, J., *A Multicluster, mobile, multimedia radio network,* ACM-Baltzer Journal of Wireless Networks, Vol. 1 No. 3, 1995
[5] Yu-Chee Tseng, Sze-Yao Ni, En-Yu Shih, *Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network,* Infocom, 2001.
[6] Amir Qayyum, Laurent Viennot, Anis Laouiti, *Multipoint relaying: An efficient technique for flooding in mobile wireless networks*, INRIA report, March 2000.
[7] H.Lim and C. Kim, *Flooding in wireless ad hoc networks*, IEEE computer communications 2000.
[8] Krishna, P., Vaidya, N.H., Chatterjee, M., Pradhan, D.K., *A cluster-based approach for routing in dynamic networks*, Computer Communication Review, vol.27, (no.2), ACM, April 1997.
[9] C. Perkins, E. Royer, S. Das, *Ad hoc on demand distance vector (AODV) routing*, http://www.ietf.org/internet-drafts/draft-ieft-manet-aodv-03.txt, 1999
[10] UCLA, *Glomosim: A scalable simulation environment for Wireless and Wired Network systems,* http://pcl.cs.ucla.edu-/projects/glomosim.
[11] J. Wu and H. Li, *On calculating connected dominating set for efficient routing in ad hoc networks,* DIAL M. 1999
[12] B. Chen, K. H. Jamieson, and R. Morris *An energy-efficient coordination algorithm for topology maintenance in Ad Hoc wireless networks,* Mobicom, 2001
[13] Y. Xu, J. Heidemann and D. Estrin *Geography-informed Energy Conservation for Ad Hoc Routing* Mobicom, 2001
[14] M. Seddigh, J. Solano and I. Stojmenovic, *Internal nodes based broadcasting algorithms in wireless networks,* Proceedings of the 34th Annual HICSS 2001
[15] Yunjung Yi, Taek Jin Kwon and Mario Gerla *Passive Clustering (PC) in Ad Hoc Networks,* Internet Draft, draft-ietf-yi-manet-pac-00.txt, Nov. 2001.
[16] T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, L. Viennot *Optimized Link State Routing Protocol*, Internet Draft, draft-ietf-manet-olsr-05.txt, Nov. 2001.
[17] R. G. Ogier, F. L. Templin, B. Bellur, M. G. Lewis *Topology Broadcast Based on Reverse-Path Forwarding (TBRPF),* Internet Draft, draft-ietf-manet-tbrpf-03.txt, Nov. 2001.
[18] Yogen K. Dalal , Robert M. Metcalfe *Reverse path forwarding of broadcast packets,* Communications of the ACM December 1978
[19] JJ. Garcia-Luna-Aceves and M. Spohn *Scalable Link-State Internet Routing,* ICNP '98
[20] D.B.Johnson and D.A. Maltz, *Dynamic source routing in ad hoc wireless networks,* Mobile Computing, Academic Publishers, 1996. pp. 153-181.
[21] S.-J. Lee, W. Su, and M. Gerla *On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks,* Internet Draft, draft-ietf-manet-odmrp-02.txt, Jan. 2000.
[22] Yunjung Yi, Taek Jin Kwon, Mario Gerla, *A load aware routing (LWR) based on local information,* PIMRC 2001.
[23] B. Williams and T. Camp *Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks,* MOBIHOC 2002