Fair and Reliable Multicast Routing for Wireless Mesh Networks

Nasrin Razavi[1], Mohammad Reaz Meybodi[2]

[1] Department of Computer Engineering, Islamic Azad University, qazvin Branch
qazvin, Iran
n.razavee@yahoo.com

[2] Department of Computer Engineering and IT, Amirkabir University of Technology
Tehran, Iran
m.meybodi@aut.ac.ir

Abstract

Multicast is a fundamental routing service in wireless mesh networks (WMNs) due to its many potential applications such as video conferencing, online games, and webcast. Recently, researchers proposed using link-quality-based routing metrics for finding high-throughput paths for multicast routing. However, the performance of such link-quality-based multicast routing is still limited by severe unfairness. Two major artifacts that exist in WMNs are fading which leads to low quality links, and interference which leads to unfair channel allocation in the 802.11 MAC protocol. These artifacts cause the multicast application to behave unfairly with respect to the performance achieved by the multicast receivers.

In some applications such as electronic test, competitions and etc. in addition to need for establishing the fairness among the members of the multicast group, it is need to reliable routing with 100% packet delivery ratio (PDR). one of proposed method for increasing PDR is Automatic Repeat Request(ARQ). Previously, numerous multicast protocols based on ARQ have been proposed to improve the packet delivery ratio. Therefore, these ARQ-based protocols can lead to excessive control overhead and drastically reduced throughput. Another technique for increasing PDR is Forward Error Correction (FEC) this method unlike the ARQ prevents of forward control packets and decreases overhead network partly.

In this paper, we present methods that in addition of the fairness improvement between the members of multicast group, increase PDR . two methods ARQ-based and FEC-based present in this paper, will be compared the gained fairness and PDR by 802.11.

Keywords: *wireless mesh network, multicast, routing*

1. Introduction

Wireless mesh networks (WMNs) have been proposed as an efficient solution for ubiquitous last-mile broadband access .The deployment and use of WMNs has increased significantly and several cities have planned and/or deployed WMNs [1-4]. WMNs are characterized by static mesh routers connected by wireless links to each other and to a few gateway nodes. These networks typically have low maintenance overhead, high data rates, and are not energy constrained. Compared to mobile Ad-hoc networks (MANETs), the static nature of mesh routers has shifted the major design challenges from maintaining connectivity among routers and conserving energy consumption to improving applications' performance, in particular, providing high-throughput and guaranteeing reliability in network access[5].

In a typical WMN environment, two major factors can severely affect both throughput and fairness of the whole network: fading and interference. Fading is the random attenuation of the signal due to reflections, scatterings and multipath propagation. Fading leads to inherently low quality links (even if the two end nodes are within transmission range) which can incur random packet losses.

Interference; when nodes defer their transmissions after sensing other transmissions, and packet drops, when the noise due to signals and multipaths from other transmitters decreases the SINR below the packet reception threshold. Hence, any practical solution that provides high throughput or reliability in a WMN has to deal with these two factors. Because of its random nature, more of papers don, t consider this about losing of data packets [6,7,8,9].

Multicast is another fundamental routing service in multihop mesh networks. It provides an efficient means of supporting collaborative applications such as video conferencing, online games, webcast and distance learning, among a group of users [10]. In spite of its significance, there has been little work on multicast routing in wireless networks. Moreover, there is no experimental work on providing fairness among multicast members in a multihop wireless network. Most recently, the authors of [11] studied link-quality-based routing metrics for finding high-throughput paths for multicast routing. Although these metrics have been shown to improve throughput compared to the widely used hop-count metric, one significant problem unfairness among different members of a multicast session remains unsolved.

In some application such as online game, video conferencing and etc. there is need to establishment of fairness among the multicast members until the group members obtain the equal throughput. In some of the applications such as electronic experiments, competitions and etc. in addition need to fairness establishment among the group members, there is the need to 100% PDR until questions, instruction packages and etc. attained to all members completely and simultaneously. In this paper, we present the methods that by MAC protocol namely IAFS [12] , automatic Repeat Request (ARQ) [13] and

Forward Error Correction (FEC) [13] methods, retained the fairness among the group and increased the packet delivery ratio(PDR).

The rest of this paper is as follows: in section 2, represent a method for fairness establishment among multicast group members. Sections 3 and 4, respective, are illustrated FEC and ARQ methods for increasing PDR. In sections 5 and 6, are illustrated the proposal method for fair multicast based on ARQ and FEC. In section 7, has been represented the simulation and evaluation results of given methods. In section 8 is represented a summary of the paper.

2. Fair establishment method among members of multicast groups

In this section, we present IAFS for multicast, a TDMA-like MAC solution to improve fairness and throughput of multicast in WMNs. We describe the four components of IAFS: the tree construction, the interference model, the scheduling algorithm, and the propagation of the schedule to the WMN nodes.

2.1. Tree construction

Constructing a good multicast tree is critical to the performance of any scheduling algorithm. In a realistic environment, if a tree consists mainly of lossy links due to fading, its performance will be poor, even with a perfect scheduling algorithm. In [13], the authors showed that SPP, which selects the path with the highest probability of packet delivery, gives the highest throughput among various link-quality-based metrics. Hence, in IAFS, we use SPP as the tree construction metric. SPP for a link is defined as the probability for a packet to be successfully transmitted from the sender to the receiver of that link, which can be easily calculated through offline measurements. The SPP for a whole path is equal to the product of the SPP values of the links constituting the path. A modified Dijkstra algorithm is used to find the paths with highest SPP value from the source to each receiver.

2.2. Interference model

In [14], Padhye et al. showed that it is difficult to accurately model interference among links and that simple heuristics (e.g., assuming that interference range is twice the transmission range) fail to provide accurate results. They also proposed an empirical methodology to predict pairwise interference in a network of n nodes using O(n2) measurements. For two links LAB and LCD, they define "broadcast interference ratio" (BIR) as follows:

$$BIR = (R_{AB}^{AC} + R_{CD}^{AC})/(R_{AB} + R_{CD})$$

where RAB, RCD are the packet delivery rates from A and C at B and D when only A or C broadcasts packets, respectively, and R_{AB}^{AC}, R_{CD}^{AC} are the packet delivery rates at B and D when both A and C broadcast packets simultaneously. The two links do not interfere when BIR is 1, and they interfere if BIR < 1.

In IAFS, since multicast forwarders use MAC layer broadcast to send data packets, the interference of interest is between nodes. This is in contrast to between links in [14], which is to be used for scheduling unicast. To measure the node interference, we perform all pairwise measurements similarly as in [28]. 2 However, since we consider nodes and not links in our scheduling, as we explain in the next section, we cannot use the BIR definition as above. We define two nodes A and C to be interfering if C's transmission affects the reception of any of A's children and vice versa. To measure the effect of C's transmission on A's child B, we measure the PDRs from A at B when only A broadcasts packets and when both A and C broadcast packets, respectively, and denote them as R_{AB}^{A}, R_{AB}^{AC}. We then define interference ratio (IR) as IRAC = min B ∈ children(A) R_{AB}^{AC} where $IR_{AB}^{AC} = R_{AB}^{AC}/R_{AB}^{A}$. Thus, node C is an interferer for link A → B (and it cannot transmit simultaneously with A), if IRAC < 1.

Similarly to the original BIR definition, the IR can take any value between 0 and 1. This gives us two different interference models. A schedule that assigns two forwarding nodes the same time slot only if they do not interfere as defined above guarantees fairness among all multicast receivers, assuming all tree links are of equal quality. We call this interference model the binary model. The binary model is conservative and may lead to long cycles and hence reduced sender rates (inversely proportional to the cycle lengths) and throughput. In this paper, we also consider the threshold-based model which considers node C as an interferer for node A only if IRAC < IT where IT is a selected interference threshold. The threshold-based model is more aggressive in finding nodes that can transmit simultaneously. This can lead to a reduced cycle length, which in turn leads to an increased sending rate and potentially increased throughput. However, it can also lead to unfairness as weakly interfering nodes are now competing to transmit in the same slot. Thus the threshold-based model effectively trades reduced fairness for increased throughput .We experimentally study this tradeoff in Section 5.

2.3. Scheduling algorithm

Our scheduling algorithm is based on spatial TDMA, first proposed in [15], and consists of three phases: compatibility matrix (CM) and compatibility graph construction, clique enumeration, and clique selection, similar to [16]. In the following, we explain the three phases of the scheduling algorithm using a single multicast tree in Fig. 1(a) as an example .We will explain how the algorithm can be easily extended for multiple concurrent trees at the end of this section. In this figure, the tree connects the gateway (node 1) to the receivers (nodes 5, 6, 7, 9, 10, 11). The solid lines denote the tree links, while the dashed lines denote links (with two end nodes within transmission range of each other) that are not part of the tree. In this example, S = {1}, R = {5, 6, 7, 9, 10, 11}, and FG = {1, 2, 3, 4, 8, 10}, and the scheduling algorithm has to schedule transmissions of nodes 1, 2, 3, 4, 8, 10.

CM construction: since forwarding nodes in IAFS use MAC layer broadcast to send data, the interference of interest is between nodes. This is in contrast to between links, as in the original spatial TDMA algorithm [15] and its later variations (e.g., [17,16]) which assign transmission rights to links.

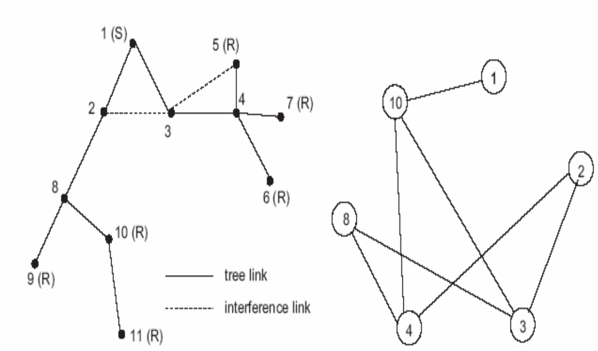

Fig1. Example physical tree formation and compatibility graph.

Consequently, in our scheduling algorithm, we define the CM to describe if pairs of nodes can transmit simultaneously. Specifically, we define CM as:

CM = [cmij], 1≤ i≤ j ||FG||

Where

$$cm_{ij} = \begin{cases} 0 & \text{if nodes } i, j \text{ cannot transmit,} \\ & \text{simultaneously,} \\ 1 & \text{otherwise.} \end{cases}$$

Two nodes cannot transmit simultaneously if any of the following two conditions hold: (i) any child node of node i is within transmission range of node j (collision), or (ii) one of the two nodes is an interferer for any of the children of the other node (interference). This second condition can be mathematically expressed as: $CM[ij] = 0 \text{ if } IR^{ij}_{i\text{child}^k(i)} < IT$ for any k, where childk (i) is the kth child of node i. The reason condition (1) (collision) is not treated as a special case of condition (2) is that our interference measurement uses 802.11 CSMA and hence the child node of i may still get an IR above zero, which can cause nodes i and j to be scheduled in the same slot (e.g., if IR > IT).

The CM for the tree in Fig. 1(a), assuming IT = 1, is

$$CM = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \end{pmatrix},$$

where the rows correspond to nodes 1, 2, 3, 4, 8, and 10.
Using the CM, we can construct the compatibility graph shown in Fig. 1(b). In this graph, vertices correspond to nodes in FG, and an edge between two vertices denotes that these two nodes can transmit simultaneously.
Clique enumeration: After the CM construction, we enumerate all possible cliques in the compatibility graph. Although the problem of clique enumeration is NP-hard [16], the relatively small size of a WMN makes it easy to solve. For the graph in Fig. 1(b), the set of all cliques is

{{1}, {2}, {3}, {4}, {8}, {10}, {1, 10}, {2, 3}, {2, 4}, {3, 8}, {3, 10}, {4, 8}, {4, 10}}

Clique selection: We define a schedule as a set of cliques s that fulfills the following two conditions: (i) all nodes in FG are included in the schedule and (ii) each node in FG is included only once. Together, these two conditions ensure fairness among the receivers.

Given the list of cliques obtained in the previous step, we can enumerate the set S of all possible schedules. Each schedule s □ S corresponds to a cycle length Ts=||s|| , where ||s|| is the number of cliques in schedule s. To maximize spatial reuse, we need to select the schedule s□ that minimizes the cycle length:

$$T_{s^*} = \min_{s \in S} T_s .$$

Finding the optimal schedule requires an exhaustive search of all possible schedules. To reduce the computation cost, we propose a simple heuristic. The basic idea is to incrementally select and add to our schedule cliques that include many FG nodes, so that the total number of cliques is minimized. However, a straight forward implementation of the basic idea may not yield good schedules. For example, in Fig. 1(b), two possible schedules that may result from simply selecting cliques based on their sizes are
s1 = {{1, 10}, {4, 8}, {2, 3}}
s2 = {{4, 10}, {3, 8}, {2, }, {1}}
which have different cycles. This example shows that arbitrarily breaking ties may not lead to a good solution. To address this issue, we propose a heuristic called least overlapped first (LOF). In LOF, each clique is assigned a rank, equal to the number of common nodes this clique has with all other cliques of the same size. Then at each step of the scheduling algorithm, we select the clique that has the smallest rank, among the cliques of the same size. The intuition behind this heuristic is that if we can schedule a large clique that does not have many common nodes with other cliques of the same size, it will be easier to find other cliques with the same size, and form a schedule with a small cycle.
The steps of the LOF algorithm are as follows:
1. set s = { };
2. while not all nodes in FG are included in s
search for the clique CLi that includes the maximum number of nodes among the cliques that do not intersect with the members of s. If there are more than one cliques with the same number of nodes, select the one with the lowest rank.
add CLi to s.
The above LOF algorithm generates schedule s1 in the example of Fig. 1. In this example, clique {1, 10} has rank 2 as it overlaps with cliques {3, 10} and {4, 10}, cliques {2, 3}, {2, 4}, {3,8}, {4, 8} have rank 3, and cliques {3, 10}, {4, 10} have rank 4. Hence, LOF will first select the clique with the smallest rank, {1, 10}, and then two cliques with rank 3, resulting in schedule s1 with a cycle of three slots.
The main idea of the scheduling algorithm is common in many different scheduling algorithms, e.g., [18,19,20], although details may differ. In most of these works, the term flow contention graph or conflict graph is used, which is the complement term of compatibility graph in our paper, and the term independent set is then used instead of the term clique, which contains nodes that are not connected in the conflict graph. Then finding the minimum number of cliques to cover the compatibility graph is equivalent to finding the minimum number of independent sets to cover the conflict graph. We point out that the two approaches are essentially equivalent. We could have easily used the conflict graph-related terminology as well.

Scheduling multiple trees: The basic algorithm can be easily extended to schedule multiple multicast trees. In this case we simply enumerate each forwarder in each multicast tree separately in the CM construction. If the same node appears as forwarder in k multiple trees, it will appear as k separate nodes in the CM, and be assigned k time slots in a cycle, one for each of its tree appearance.

2.4. Schedule propagation

After computing the schedule, the network operator has to propagate it to the FG nodes. The schedule is propagated along the same tree the operator formed for the multicast session. However, since we need reliable delivery of the schedule to the forwarding nodes, each forwarding node unicasts it to its child nodes using hop-by-hop TCP sessions. The file of the schedule size is very small, containing only information of which nodes transmit at each slot of the cycle (and for which multicast session as there can be multiple concurrent ones).

3. Automatic Repeat Request

We selected ReMHoc [21] as a representative ARQ protocol for our comparison. ReMHoc follows the design principles of SRM [22], perhaps the most popular ARQ protocol for reliable multicast in the wired Internet. ReMHoc is receiver-initiated; each receiver is responsible for detecting losses, by detecting gaps in the packet sequence numbers. When a packet loss is detected, the receiver schedules a Request packet, asking for retransmission of the lost packet. To prevent the implosion of control packets, receivers wait for a random period of time before sending a request for a lost packet. If they receive a request for the same packet from another receiver before their timer expires, they postpone their own request by resetting the timer. This backoff is exponential in SRM, but linear in ReMHoc (proportional to the number of times this request has already been scheduled). This is because the loss rate is much higher in wireless networks than in the Internet, and hence faster response is required.

If the timer expires, a request is sent. But there is no guarantee that the request itself will not be lost, or that the repair packet will reach this receiver. Hence, the request timer is reset. In ReMHoc there is no upper bound on the number of times a request can be sent. However, we found that by allowing an infinite number of requests, the control overhead grows too fast and the PDR is reduced. Therefore, we decided to allow up to five retransmissions of t e same request. After requesting a packet for five times, a receiver considers this packet permanently lost and no further action is taken for that packet in the future. Request packets are multicast toward the whole group. Any multicast member that receives a request packet and has the requested packet, sends a Repair packet and does not propagate the request further.

Similarly as for the re- quest packets, a node postpones its transmission of a repair packet for a random period of time, and cancels it if in this time it hears another node retransmitting the same repair packet. Each repair packet is multicast to the whole group, so that all nodes that are missing the same data packet can recover by using the same repair packet.

4.Forward Error Correction

The key idea behind forward error correction (FEC) [23,24] is that k data packets are encoded at the sender to produce n encoded packets, where n > k, in such a way that any subset of k encoded packets suffices to reconstruct the original k data packets. Such a code is called an ǹ; kق FEC code and allows the receiver to recover from up to n _ k packet losses in a group of n encoded packets. A code is called systematic when the first k encoded packets are the original data packets. Systematic codes are much cheaper to decode and they allow partial reconstruction of data even when fewer than k encoded packets are received (with non-systematic codes receiving fewer than k packets is equivalent to receiving zero packets, since decoding is not possible). In a systematic code, the n _ k encoded packets that are different from the original k data packets are called parities.

In this paper we use a particular class of FEC codes, called Reed–Solomon (RS) codes, which use Vandermonde matrices to encode the data packets. In RS codes, k data packets are interpreted as the coefficients of a polynomial P of degree k _ 1. The encoding process involves multiplying the original data by an n _ k Vandermonde matrix G, and the decoding process requires the inversion of a k _ k submatrix G0 taken from G', and multiplication of the received data by G0_1. Rizzo [23] gives a description and a software implementation of RS codes which is used by many protocols, such as [25,26]. It works for packet sizes up to 1024 bytes, and the proposed values for k and n are 32 and 255, respectively.

5. Fair ARQ-based Multicast (FAM)

In this method, the data were forwarded to multicast members by given method in section 2, and if the receiver doesn't receive any packet because fading and etc. it forward the request packet with the aim of given ARQ method in section 3, and so enhanced PDR. The results of this method will present in section 7. the objection for ARQ method is the much overhead of it' s control packets, but by fair IAFS protocol, the rate of packet lost is less and the ARQ method doesn’t have much overhead.

6. Fair FEC-based Multicast (FFM)

In this method, the data was encoded before forward by source in presented FEC method in section 4. Then by means of given MAC protocol in section 2, were scheduled for data forwarding.

In this method the packet loss can be retrieval by forwarded parities in encoded packets. FFM unlike FAM has less overhead and also preservation fairness among the multicast members and increases the PDR. In section 7 , the results of this method is presented.

7. Simulation evaluation

7.1 simulation environment

We used the GloMoSim [29] simulator in our simulation study. We simulated a network of 50 static nodes placed randomly in a 1000m×1000m area, and we simulated 10 different topologies. The radio propagation range was 250m and the nominal bit rate was 2 Mbps (the data rate used for broadcast in 802.11 MAC protocol). environment. In our experiments, we used the two-ray propagation model, along with thermal noise and

Rayleigh fading. The noise factor was set to 7 dB. The Rayleigh fading model is appropriate for modeling environments with many reflectors, e.g., trees and buildings, where the sender and the receiver are not in line-of-sight of each other. Such environments are common in WMNs.Table.1. lists the general parameters that characterize the simulation environment. We compared results of our proposed methods with IAFS and 802.11 protocols.

Table.1.simulation environment

simulator	GlomoSim
Total Node	50
Simulation Time	400 seconds
Simulation Area	1000m ×1000m
Node Placement	Random
Radio Range	250m
Channel capacity	2 Mbps
Data Packet Size	1024 byte

7.2. Evaluation metrics

We use the following metrics in comparing our proposed methods with IAFS and 802.11.

PDR: the number of packets received by each receiver divided by the number of packets sent by the multicast source. PDR equals the throughput divided by the sending rate of the source.

1. Throughput (Kbps): the number of packets delivered to each receiver divided by the duration of the multicast session.

2. Fairness: fairness is characterized using the notion of Fairness Index γ defined in [10]. Although this metric is used in [10] to characterize spatial bias across comparable nodes in terms of hop-count and contention, the definition itself is not related to spatial bias and it simply gives an idea of the difference between throughputs received by different receivers.

Let the throughput of receiver i be denoted as Ti. To characterize the average fairness over all receivers, we define

$$\gamma_{ave} = average \frac{\max(Tj, Tk)}{\min(Tj, Tk)} \Big| \ \forall j, k \in [1..N]$$

and to characterize the worst case fairness over all receivers, we define

$$\gamma_{max} = \max \frac{\max(Tj, Tk)}{\min(Tj, Tk)} \Big| \ \forall j, k \in [1..N]$$

where N is the number of receivers. Ideally, $\gamma ave = \gamma max = 1$, but it is generally larger than 1 due to unfairness. The larger the γ value is, the more unfair the protocol is.

7.3. simulation results

In this section, we present the simulation results of proposed methods in PDR, fairness and throughput metrics, then compare the results with MAC 802.11 and IAFS protocols.

Fig.2. shows the packet delivery ratio by four approaches. So that, is clear in the figure the proposed approach in all scenarios is better than 802.11 and IAFS, but in comparing between FFM and FAM, can be seen that PDR in FFM almost in all cases is equal with 100%, and can obtained better results than FAM. In average case the PDR in FFM is equal with 100%. Therefore, FFM has been better results than FAM, IAFS and 802.11 respective, 2% , 7% and 23%.

Fig.3. a and b respective, shows the fairness average and worst case fairness by four approach in 10 different scenario. just as is clear of results, FAM gets better results than 802.11 protocols, but doesn't show good results than the two other approaches. IAFS and FFM get same results in most scenarios.

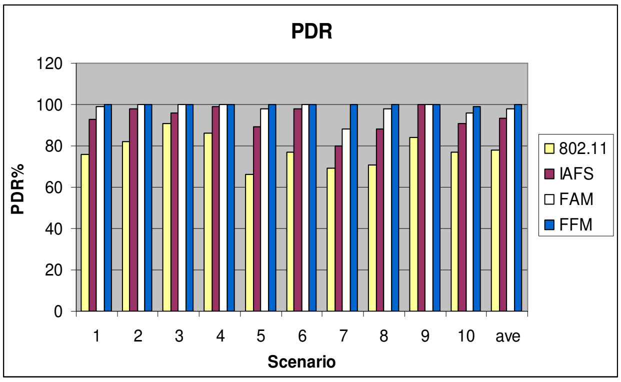

Fig.2. PDR comparison for 802.11, IAFS, FAM and FFM for 10 different scenarios

Just as is seen in fig.4.a , the gained fairness average by these two approaches in all scenario is equal with 1 and only in scenario 5 , fairness average isn't as good as another scenarios. Fig.3.b. shows the worst case fairness results in 10 scenarios. The results for IAFS and FFM are similar and better than two another approaches.

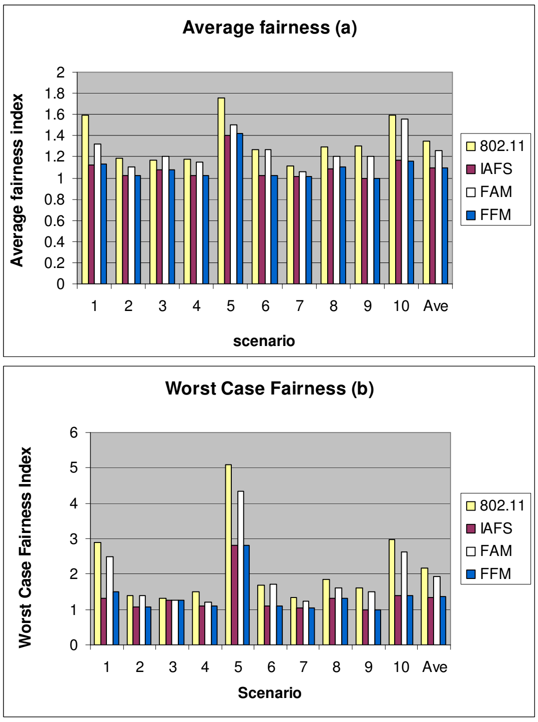

Fig.4. PDR comparison for 802.11, IAFS, FAM and FFM for 10 different scenarios.

Fig.4. shows throughput comparison for 802.11, IAFS, FAM and FFM for 10 different scenarios. Just as the results are clear, FAM doesn't get good results than the IAFS and FFM approaches. But FFM in all scenario gets better results than another approaches and the difference of the gained throughput by it with IAFS was 2 to 8 Kbps.

Just as is clear of results, FFM can with fairness established among multicast members, got to 100% PDR and higher throughput than another protocols.

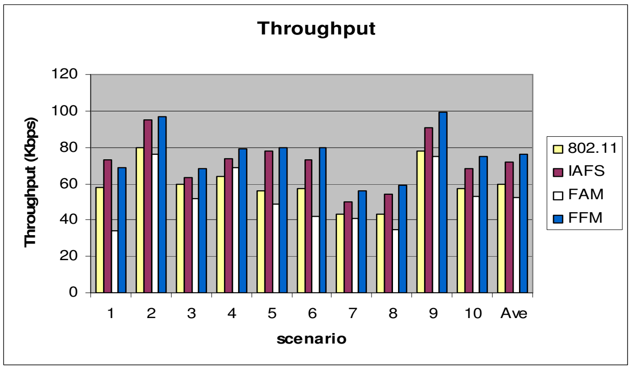

Fig. 3. throughput comparison for 802.11, IAFS, FAM and FFM for 10 different scenarios.

8. conclusion

In this paper, is presented two fair approaches for increasing packet delivery ratio in multicast for wireless mesh networks. In FAM, after data fair forward, each receiver requests the packet that it doesn't receive, further, so PDR has been increased. Just as the results show, this method have overhead partly for the forwarding control packets. But in FFM, first data encoded by RS method , then forwarded fairly. If the receiver doesn't take packet correctly, will retrieval the data by forwarding parities in encoded packets.

The results show that both proposed method increase PDR , but FFM has better throughput anf retain fairness among members correctly.

Reference

[1]ay area wireless users group, <http://www.bawug.org>.

[2] Champaign-Urbana community wireless network, <http://www.cuwireless.net>.

[3]MIT Roofnet, <http://www.pdos.lcs.mit.edu/roofnet>.

[4] Seattle wireless, <http://www.seattlewireless.net>

[5] D. Koutsonikolas and Y. Charlie Hu, Exploring the Design Space of Reliable Multicast Protocols for Wireless Mesh Networks, Journal of Ad Hoc Networks, 2009, Vol. 7, pp. 932–954

[6] D.S.J.D. Couto, D. Aguayo, J. C. Bicket, R. Morris, A high throughput path metric for multi-hop wireless routing, in: Proceedings of ACM MobiCom, 2003.

[7] R. Draves, J. Padhye, B. Zill, Routing in multi-radio, multi-hop wireless mesh networks, in: Proceedings of ACM MobiCom, 2004

[8] P. Gupta, P.R. Kumar, The capacity of wireless networks, IEEE Trans. Inform. Theory 46 (2) (2000) 388–404.

[9] M. Kodialam, T. Nandagopal, Characterizing achievable rates in multihop wireless networks: the joint routing and scheduling problem, in: Proceedings of ACM MobiCom, 2003.

[10] S.J. Lee, W. Su, M. Gerla, On-demand multicast routing protocol in multihop wireless mobile networks, ACM/Kluwer Mobile Networks and Applications 7 (6) (2002)

[11] S. Banerjee, A. Misra, Minimum energy paths for reliable communication in multi-hop wireless networks, in: ACM MobiHoc, 2002.

[12] D. Koutsonikolas, S. Das, Y. Charlie Hu, An interference-aware fair scheduling for multicast in wireless mesh networks, , journal of parallel and destributied computing , 2008, Vol. 68, pp. 372 – 386

[13] D. Koutsonikolas and Y. Charlie Hu, Exploring the Design Space of Reliable Multicast Protocols for Wireless Mesh Networks, Journal of Ad Hoc Networks, 2009, Vol. 7, pp. 932–954

[14] J. Padhye, S. Agarwal, V. Padmanabhan, L. Qiu, A. Rao, B. Zill, Estimation of link interference in static multi-hop wirele s networks, in: Proceedings of IMC, 2005.

[15] R. Nelson, L. Kleinrock, Spatial tdma: a colllision-free multihop channel access protocol, IEEE Trans. Comm. 33 (9) (1985).

[16] N.B. Salem, J.-P. Hubaux, A fair scheduling for wireless mesh networks, in: Proceedings of WiMesh, 2005.

[17] H. Luo, S. Lu, V. Bharghavan, A new model for packet scheduling in multihop wireless networks, in: Proceedings of ACM MobiCom, 2000.

[18] A. Ephremides, T.V. Truong, Scheduling broadcasts in multihop radio networks, IEEE Trans. Comm. 38 (4) (1990).

[19] H. Luo, P. Medvedev, J. Cheng, S. Lu, A self coordinating approach to distributed fair queueing in ad hoc wireless networks, in: Proceedings of IEEE Infocom, 2001.

[20] H. Luo, S. Lu, A topology-independent fair queueing model in ad hoc wireless networks, in: Proceedings of IEEE ICNP, 2000.

[21] A. Sobeih, H. Baraka, A. Fahmy, Remhoc: a reliable multicast protocol for wireless mobile multihop ad hoc networks, in:Proceedings of the IEEE Consumer Communications and Networkin Conference (CCNC), 2004.

[22] S. Floyd, V. Jacobson, C.-G. Liu, S. McCanne, L. Zhang, A reliable framework for light-weight sessions and application level framing, IEEE/ACM Transactions on Networking, 1997.

[23] L. Rizzo, Effective erasure codes for reliable computer communication protocols, ACM Computer Communications Review 27 (2) (1997).

[24] A.J. McAuley, Reliable broadband communication using a burst erasure correcting code, in: Proceedings of ACM SIGCOMM, 1990.

[25] J. Nonnenmacher, E. Biersack, D. Towsley, Parity based loss recovery for reliable multicast transmission, in:Proceedings of the ACM SIGCOMM, 1997.

[26] L. Rizzo, L. Visicano, Rmdp: an fec-based reliable multicast protocol for wireless environments, Mobile Computing and Communications Review 2 (2) (1998).

Author Biographies

Nasrin Razavi received the B.S. in Computer Engineering in Iran, in 2007. currently she is master in computer engineering in iran, in qazvin university. Her research interests include wireless networks, , learning systems, and distributed database.

Mohammad Reza Meybodi received the B.S. and M.S. degrees in Economics from Shahid Beheshti University in Iran, in 1973 and 1977, respectively. He also received the M.S. and Ph.D. degree from Oklahoma University, USA, in 1980 and 1983, respectively in Computer Science. Currently, he is a full professor in Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran. Priorto current position, he worked from 1983 to 1985 as an assistant professor at Western Michigan University, and from 1985 to 1991 as an associate professor at Ohio University, USA. His research interests include wireless networks, fault tolerant systems, learning systems, parallel algorithms, soft computing and software development.