Improving QoS in Multicasting Through Adaptive Redundancy

Tolga Numanoglu and Wendi Heinzelman
Department of Electrical and Computer Engineering
University of Rochester, Rochester, NY 14627 USA
Email:{numanogl, wheinzel}@ece.rochester.edu

Abstract—In mobile ad hoc networks (MANETs), Quality of Service (QoS) of a multicast protocol is one of the most important performance metrics. Channel conditions and network topology frequently change, and in order to achieve a certain QoS, complex algorithms and protocols are needed. Often channel conditions are neglected during the design of a multicast protocol. However, vulnerability against channel errors can severely cripple the performance of a multicast protocol. Mesh networking inspired multicasting approaches introduce increased redundancy in the routing process to overcome the performance loss due to channel errors. Although utilizing multiple paths from senders to receivers results in higher reliability, under better channel conditions the additional redundancy may not be needed in terms of reliability, and increased redundancy causes increased overhead. Therefore, we propose a mesh networking inspired approach that adapts the amount of redundancy according to the current link conditions. We show that this approach can achieve good QoS levels for real-time traffic scenarios while simultaneously reducing unnecessary energy dissipation.

I. INTRODUCTION

The objective of a multicast routing protocol for MANETs is to support the dissemination of information from a sender to all multicast group members while trying to use the available bandwidth efficiently in the presence of frequent topology and channel condition changes. In MANETs, node mobility and the fact that wireless links are more prone to transmission errors result in higher packet drop probability when compared to wired networks. Therefore, multicast routing protocols that provide route redundancy (i.e., routing packets along multiple paths from source to receivers), typically outperform multicast routing mechanisms that offer no redundancy. However, increased redundancy can cause significant overhead in a resource-constrained MANET, even though it provides higher packet delivery ratios.

When channel conditions are good (*i.e.,* link reliability is high), having larger redundancy in the network does not significantly increase packet delivery ratio (PDR). However, when channel conditions get worse, having greater redundancy does have a considerable impact on the packet delivery ratio. Our proposed multicasting mechanism with adaptive redundancy varies the redundancy in the network according to the local packet reception history. Redundancy is managed locally and adaptive behavior is controlled by the multicast members (*i.e,*

receivers). A multicast member node controls the amount of redundancy depending on the stability of its upstream node and incoming traffic. If the upstream node is volatile and data packets do not arrive in order, the number of upstream nodes is increased. If a consistent upstream node exists and data packets continue to arrive in order, the number of upstream nodes is decreased. This process is repeated by each relay to provide a greater number of non-overlapping redundant branches between the source and the multicast member.

We implement our adaptive redundancy algorithm on a tree based multicasting protocol called Multicasting through Time Reservation using Adaptive Control for Energy efficiency (MC-TRACE) [1], and compare the performance of adaptive redundancy multicasting against its non-adaptive version and a non-adaptive mesh-based routing protocol, On Demand Multicast Routing Protocol (ODMRP) [2]. Through ns-2 simulations, we show that adaptive redundancy can maintain high packet delivery ratios at higher protocol efficiency when compared to both non-adaptive mesh and tree-based multicast protocols.

We present related work in Section II. Section III summarizes the core features of MC-TRACE. Motivation to our work is given in Section IV. In Section V, we provide limits of adaptive redundancy and a detailed description of our approach. We present simulation results comparing the adaptive and original versions of MC-TRACE to ODMRP in Section VI. Conclusions are drawn in Section VII.

II. RELATED WORK

A. Multicast Routing Protocols

There are many multicast routing protocols designed for mobile ad hoc networks, and they can be categorized into two broad categories [3]: (i) tree-based approaches and (ii) mesh-based approaches. Tree-based approaches create trees originating at the source and terminating at multicast group members with an objective of minimizing a cost function. For example, the cost function to be minimized can be the distance between the source and every destination in the multicast group [4]. A multicast protocol for ad hoc wireless networks (AMRIS) [5] constructs a shared delivery tree rooted at one of the nodes with IDs increasing as they radiate from the source, which reduces the route discovery time and also confines route recovery overhead to the proximity of the link failure.

Mesh-based multicasting is better suited to highly dynamic topologies, simply due to the redundancy associated with this approach [2], [6]. In mesh-based approaches there is more than one path between the source and multicast group members (i.e., a redundant multicast-tree); thus, even if one of the paths is broken due to mobility the other paths may be available.

One such mesh-based protocol is ODMRP [2], which is based on periodic flooding of the network by the source node through control packets to create a multicast mesh. Instead of using a tree, ODMRP utilizes a mesh structure, which is robust, to compensate for the frequent route failures and trades-off bandwidth for stability, which comes with multiple redundant routes between the source and destinations).

B. Improving QoS Under Varying Link Conditions

We can summarize the previous work on improving QoS under varying link conditions in ad-hoc networks in three categories; (i) adapting the channel coding to overcome the varying link conditions [7], (ii) choosing routes according to link conditions in order to avoid unnecessary retransmissions and improve the delay performance [8], [9], [10], and (iii) varying the number of redundant links between the nodes of the network to increase the efficiency [11], [12], [13]. The common goal of these different types of approaches is to increase the reliability of relaying information over multiple hops in a wireless medium.

Our approach differs from the literature in the way that we take channel errors and node mobility into consideration at the same time and propose an adaptive approach that aims to keep the energy consumption of the network under control while providing better QoS support. We employ a bottom-up approach by pointing out the effects of increasing redundancy and showing the trade-off between reliability and energy consumption, as opposed to the previous adaptive approaches, which start with a highly redundant routing protocol and strip off additional redundancy in less dynamic scenarios.

III. MC-TRACE SUMMARY

Our multi-rate multicasting protocol is built on the MC-TRACE multicasting protocol. In this section, in the first part we summarize the Multi-Hop Time Reservation using Adaptive Control for Energy efficiency (MH-TRACE) protocol [14]. MH-TRACE is the MAC protocol on which Multicasting through Time Reservation using Adaptive Control for Energy efficiency (MC-TRACE) is tailored [1]. Both of these protocols, MH-TRACE and MC-TRACE, are used for energy-efficient real-time data communications.

The TRACE family of protocols utilize a periodic frame structure, and each frame consists of a control sub-frame for transmission of control packets and a contention-free data sub-frame for data transmission (see Figure 1). Beacon packets are used for the announcement of the start of a new frame; Clusterhead Announcement (CA) packets are used for reducing co-frame cluster interference; contention slots are used for initial channel access requests; the header packet is used for announcing the data transmission schedule for the
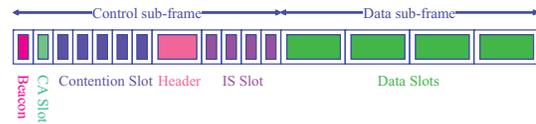
Fig. 1.   MH-TRACE frame structure.

current frame; and Information Summarization (IS) packets are used for announcing the upcoming data packets. IS packets are crucial in energy saving. Each scheduled node transmits its data at the reserved data slot.

The basic design philosophy behind the networking part of the architecture is to establish and maintain a multicast tree within a mobile ad-hoc network. This is achieved by using broadcasting to establish the desired tree branches and pruning the redundant branches of the multicast tree based on feedback obtained from the multicast leaf nodes. Although these techniques have been used in many multicasting architectures in the past, the novelty in MC-TRACE is the re-engineering of these techniques in a highly energy-efficient manner for ad-hoc network multicast routing. Energy efficiency of the architecture is partially due to the medium access part, where the nodes can switch to sleep mode frequently; and partially due to the network layer part, where the number of redundant data retransmissions and receptions are mostly eliminated. Further details of MC-TRACE can be found in [1].

IV. MOTIVATION

Although redundancy in a multicast mesh promises greater stability in the face of changing conditions or failure at single nodes [15], [16], [17], careful steps need to be taken in order to achieve the desired design goals and a balanced trade-off between the increased use of resources and performance improvement. In addition to this, when link breakages are low, even protocols that do not utilize redundancy have excellent packet delivery ratios [18]. Therefore, the additional cost incurred by redundant mesh based protocols is unnecessary and often wasteful. The next section presents our proposed method to adaptively vary the level of redundancy in a multicast mesh according to the channel conditions.

In this paper, we consider constant bit rate (CBR) real-time data traffic where data packets need to be delivered within a pre-determined delay bound in the order they are transmitted (*e.g.,* real-time voice traffic). Nodes do not utilize any retransmission policy. Moreover, cooperative broadcasting, where receiving the same packet from different sources increases the chance of successful decoding, is not considered.

Before going into the details of our approach, we would like to point out that there are mechanisms within the MC-TRACE protocol that significantly improve system performance in the face of node mobility. However, these mechanisms cannot completely solve the problem of broken trees when, in addition to node mobility, channel errors cause packets to be dropped. For example, if a data packet is randomly dropped due to channel errors, this may not trigger any of the existing repair mechanisms unless this behavior continues for some time.
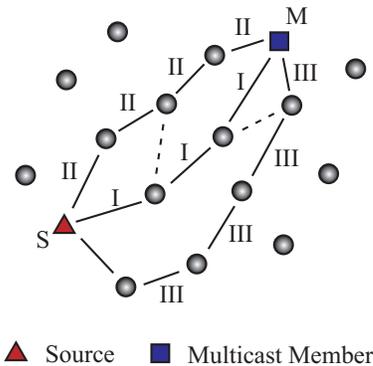
Fig. 2. Illustration of multiple branches between a source-multicast member pair. Solid lines represent possible non-overlapping routes between the source and member. Dashed lines represent possible interconnections between the branches.

Fig. 3. Solid lines are plots of Equation 3 for $k = [1, 5]$, $L_D = 128 \times 8 = 1024bits$, $N_{hop} = 4$, and varying BER. The dashed line is P_{drop} versus BER.

Moreover, the repair mechanisms are not designed to deal with random link breaks that last shorter than the pre-defined timers. The problem of random link failures cannot be handled by the original MC-TRACE protocol. Thus, we introduce adaptive redundancy, which, in conjunction with the branch formation and repair mechanisms of MC-TRACE, helps in dealing with link breaks due to channel errors as well as mobility.

Adaptive systems need to react to changing conditions at a rate fast enough to sustain a desired level of performance. In our case we aim to achieve a better QoS under increasing channel errors while keeping the unnecessary redundancy as low as possible. In order to keep our approach as a distributed and lightweight addition to a multicasting protocol, we locally vary the redundancy in the multicast tree according to the data packet reception history. In order to do this, each node monitors data packet receptions and the corresponding upstream nodes to make sure that data packets are arriving regularly from the same upstream node. If there is a disruption of data packet flow, our mechanism will increase the number of upstream nodes until a maximum number of allowed upstream nodes is reached.

V. PROVIDING ADAPTIVE REDUNDANCY

A. Limits of Redundancy

We can demonstrate the need for an upper limit to the number of upstream nodes with the help of an example given in Figure 2. For a given bit error rate BER, if we assume that a single bit error is enough to corrupt a data packet and cause it to be dropped, the probability of having at least one bit in error in a data packet of size L_D bits becomes,

$$P_{drop} = 1 - (1 - BER)^{L_D}. \quad (1)$$

In a real-time traffic scenario, where retransmissions are not utilized, the packet delivery ratio (PDR) of the branch I in Figure 2 can be written in terms of P_{drop} and the number of hops in the branch (N_{hop}).
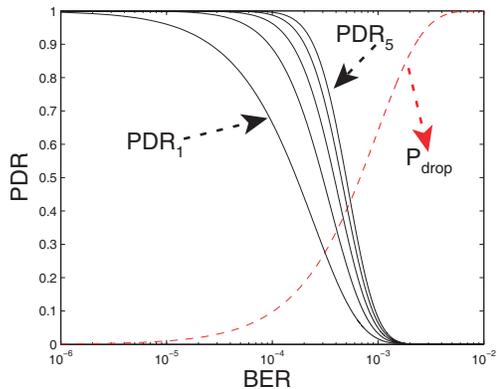
$$PDR_1 = [(1 - BER)^{L_D}]^{N_{hop}}. \quad (2)$$

Following the same idea, we can rewrite the effective PDR of having k non-overlapping branches between the source-multicast member pair.

$$PDR_k = 1 - (1 - PDR_1)^k \quad (3)$$

For $k = 3$ in Equation 3 and assuming the number of hops in all three branches is the same, we can calculate the effective PDR of the scenario illustrated in Figure 2. The existence of the interconnections between the nodes (*i.e.,* the dashed lines in Figure 2) do not contribute to the effective PDR since each node rebroadcasts the same data packet only once. Moreover, non-overlapping branches always will outperform partially overlapping routes. As Equation 3 suggests, increasing the number of non-overlapping branches will result in an increased effective PDR. However, as the number of branches increases, the number of nodes participating in the relay process also increases. This results in increased traffic, bandwidth usage, and energy consumption.

In Figure 3, we plotted the effective PDR equation (Equation 3) against increasing BER. As k increases, one can see that the increase in the effective PDR saturates. On the other hand, the amount of traffic generated and the energy consumption are linearly related with k. Therefore, it is necessary and wise to limit the number of non-overlapping branches in order to strike a well balanced trade-off between the amount of redundancy and energy consumption.

B. The Algorithm

We can describe the operation of adaptive redundancy with the help of the following example. Under perfect channel conditions and in the presence of a well maintained broadcast tree, multicast member node node A periodically receives data packets broadcasted by the source node through the branch formed between itself and the source node. However, when an expected data packet is not received (*i.e.,* dropped or never routed to node A), node A starts the process to
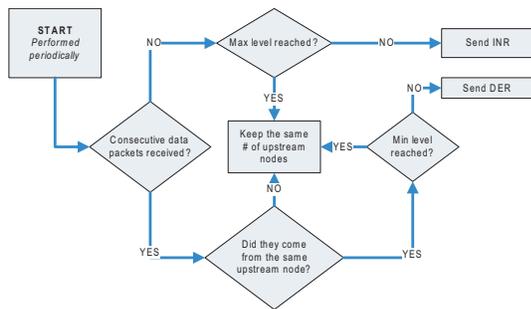
Fig. 4. Simple flow chart of the adaptive redundancy algorithm.

TABLE I
SIMULATION PARAMETERS

Acronym	Description	Value
T_{SF}	Superframe duration	32 ms
N/A	Data packet payload	128 B
T_{drop}	Packet drop threshold	160 ms
D_{Tr}	Transmission range	250 m
D_{CS}	Carrier Sense range	507 m
N/A	Network Area	$1km \times 1km$
P_T	transmit power	600 mW
P_R	receive power	300 mW
P_I	idle power	100 mW
P_S	sleep power	10 mW
C	Channel Rate	2 Mbps
N/A	Max. number of Upstream Nodes	4

increase its number of upstream nodes. Node A switches to Increase Redundancy status and announces this information via an *INcrease Redundancy* (INR) packet. An INR packet is transmitted by using one of the empty IS slots (see Figure 1), which is chosen randomly. Upon receiving an INR packet, all the nodes in the receive range of the transmitting node switch to INR status if their own hop distance to source (HDTS) is less than or equal to the HDTS of the sender (e.g., if node A's HDTS is equal to 4, nodes with an HDTS less than or equal to 4 switch to INR status; however, nodes with an HDTS larger than 4 do not). When a node switches to INR mode, it starts to relay the data packets if it has data packets for the desired multicast group. Moreover, it propagates the INR request by broadcasting an INR packet to its one-hop neighbors and starts ACKing an additional upstream node (a node with a lower HDTS) in order to sustain the required redundancy level. This procedure is repeated by all the nodes until the source node is reached. After this point, newly established links are maintained by ACK and pruning mechanisms.

We utilize the IS slots because all the nodes listen to the IS slots regardless of their energy saving mode (*e.g.,* clusterheads and ordinary nodes have different energy saving modes and an ordinary node sleeps more than a clusterhead). Upon receiving the first sequential set of data packets from the same upstream node, node A will reduce the number of its upstream nodes by sending out a *DEcrease Redundancy* (DER) packet, again using the IS slot. Therefore, the data packet reception history plays the main role in determining the level of redundancy in the network.

The main functionality of INR and DER packets is to add (remove) another (redundant) gateway. Once the redundant gateway is established, then the flow of data packets, possibly from multiple flows and/or from multiple sources, can reach the nodes with greater reliability. Note that increased redundancy causes more energy consumption, and the improvement in the performance of MC-TRACE slows down after a certain level of redundancy. Therefore, we limit the maximum level of redundancy that can be introduced by INR packets in order to avoid unnecessary energy consumption and traffic.

Figure 4 shows a simple flow chart for our adaptive redundancy mechanism. After each superframe nodes check whether or not consecutive data packets have been received

from the same upstream node. The level of redundancy is updated periodically after each superframe according to two conditions; (i) a node must receive two packets in a row from the same upstream node to be able to transmit a DNR packet, and (ii) failure to receive any consecutive data packets is enough to increase the level of redundancy.

VI. SIMULATIONS

Our next step is to perform simulations with the modified version of MC-TRACE and compare it with the original version and ODMRP. We implement the adaptive redundancy approach into the source code of MC-TRACE without modifying the frame structure and initial branch formation mechanisms. Our patch activates itself when a packet drop happens and before one of the MC-TRACE repair mechanisms can take action. We deliberately choose a shorter reaction time because our goal is to improve the performance of MC-TRACE under lossy channels.

We investigate the performance increase achieved by the INR and DER mechanisms through a set of ns-2 simulations that compare MC-TRACE and ODMRP against this new adaptive redundancy version of MC-TRACE by first choosing 8 multicast members among 128 nodes. We increase the number of multicast members to 32 while keeping the number of nodes at 128 in order to observe the effect of the traffic load on these three different multicasting approaches. In all our simulations we use mobile nodes moving within a 1 km by 1 km simulation area for 100 seconds according to the Random Way-Point (RWP) mobility model with node speeds (S) chosen from a uniform distribution where $0.0m/s < S \leq 5.0m/s$ (average pace of a marathon runner). The average instantaneous node speeds as a function of time vary from $2.5m/s$ to $2.3m/s$. A short summary of our simulation setup can be found in Table I.

A. Packet Delivery Ratio

Figure 5 shows the average and minimum PDR values of simulations with a group of 8 and 32 multicast members, respectively. Results obtained with ODMRP, original MC-TRACE, and MC-TRACE with adaptive redundancy are displayed together to offer a relative comparison between these protocols. Adaptive redundancy improves both the average

(a) (8 multicast members) Average packet delivery ratio versus BER.

(b) (8 multicast members) Min packet delivery ratio versus BER.

(c) (32 multicast members) Average packet delivery ratio versus BER.

(d) (32 multicast members) Min packet delivery ratio versus BER.
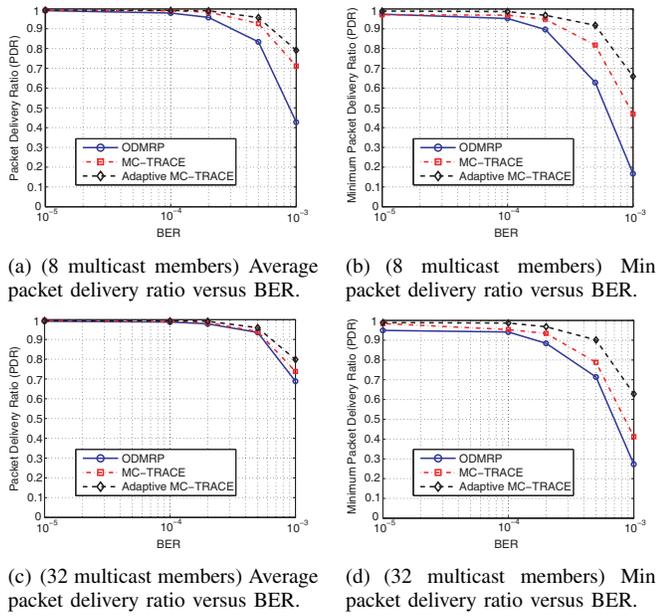
Fig. 5. Average and minimum PDRs versus BER (128 nodes).

(a) Multicast members only (8).

(b) All nodes

(c) Multicast members only (32).

(d) All nodes

Fig. 6. Average energy consumption per node versus BER (8 and 32 multicast members, 128 nodes).

PDR and minimum PDR of MC-TRACE at most by $\sim 11\%$ and $\sim 41\%$, respectively. In addition to this, as the number of multicast group members is increased from 8 to 32, ODMRP exhibits a closer performance to that of MC-TRACE due to the fact that a bigger more reliable mesh cloud, which covers almost all of the network area, is formed for 32 multicast member nodes. In this scenario, the improvement in the average PDR and minimum PDR values for adaptive MC-TRACE reaches up to $\sim 8\%$ and $\sim 52\%$, respectively.

These results can be explained by referring to the basics of these two protocols. ODMRP is an on-demand routing protocol that creates a mesh cloud between the members and the source node. In this cloud, each node participates in routing in order to increase the redundancy in the packet delivery process. On the other hand, MC-TRACE is a tree based multicasting protocol that incorporates branches between the members and the source of a multicast. Redundancy in the multicasting process is kept at minimum by pruning the unnecessary branches. Under increasing channel errors, these different characteristics of ODMRP and MC-TRACE might have favored the highly redundant mesh cloud of ODMRP to have a better performance when compared with the optimized tree structure of MC-TRACE. However, MC-TRACE's proactive nature in maintaining its tree structure helps MC-TRACE in outperforming the redundant mesh structure of ODMRP. We also believe that the mesh maintenance parameters of ODMRP can be adjusted to improve the performance under heavy channel errors.

In conclusion, the adaptive redundancy enables more packets to be delivered in a more consistent manner to all nodes in the network. In particular, improvement becomes more visible at higher bit error rates. High minimum PDR values are vital to any protocol that aims to offer QoS.
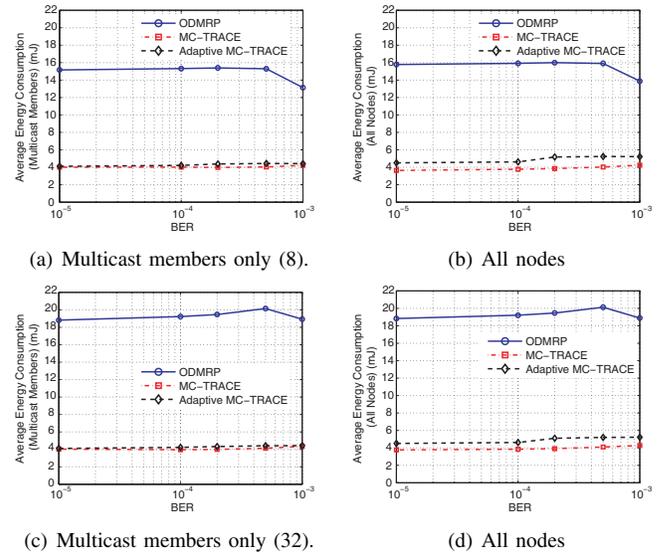
B. Energy

Figure 6 shows the energy consumption values for the three different protocols with 8 and 32 multicast group members. Average energy consumption values for both the multicast members and also all nodes in the network are provided to show that the overall energy consumption increases as the channel errors force more redundancy to be introduced in the network. However, average energy consumption among the multicast members of Adaptive MC-TRACE stays close to that of original MC-TRACE as BER increases.

With these results, our choice to improve tree based routing through adaptive redundancy becomes clear, mainly because we consume $\sim 65 - 70\%$ less than what ODMRP does. We would like to point out that the increased energy consumption due to the adaptive redundancy approach compared with original MC-TRACE can be seen as a trade-off between PDR and energy consumption. ODMRP's energy consumption reduces for the last data point in both figures since fewer data packets are relayed due to the high BER.

C. Average Number of Retransmissions (ARN)

We measure the bandwidth efficiency as the number of required data forwards to cover all the multicast nodes. We call this metric Average Number of Retransmissions (ARN). ARN is the ratio of the total number of data transmissions to the total number of data packets sent down from the application layer. Thus, the higher the ARN is the lower the bandwidth efficiency. Figure 7 shows the ARN values for the three different protocols with 8 and 32 multicast group members.

Differences between Figures 7(a) and 7(b) reveal the behavior of our adaptive redundancy approach. As the number of multicast members is increased from 8 to 32, more nodes can be involved in the redundant mesh formed by both ODMRP and Adaptive MC-TRACE. However, Adaptive MC-TRACE

(a) (8 multicast members) ARN versus BER.

(b) (32 multicast members) ARN versus BER.
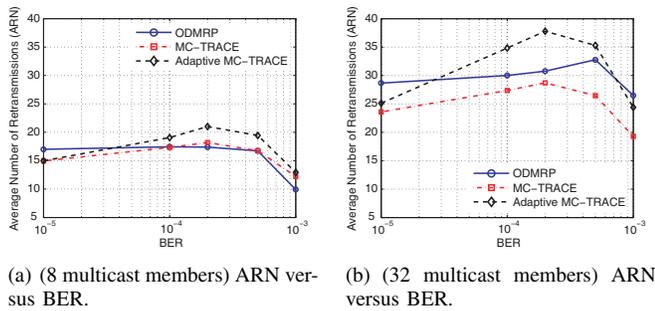
Fig. 7. ARN versus BER (8 and 32 multicast members, 128 nodes).

(a) (8 multicast members) Average data packet delay versus BER.

(b) (8 multicast members) Average data packet jitter versus BER.

(c) (32 multicast members) Average data packet delay versus BER.

(d) (32 multicast members) Average data packet jitter versus BER.
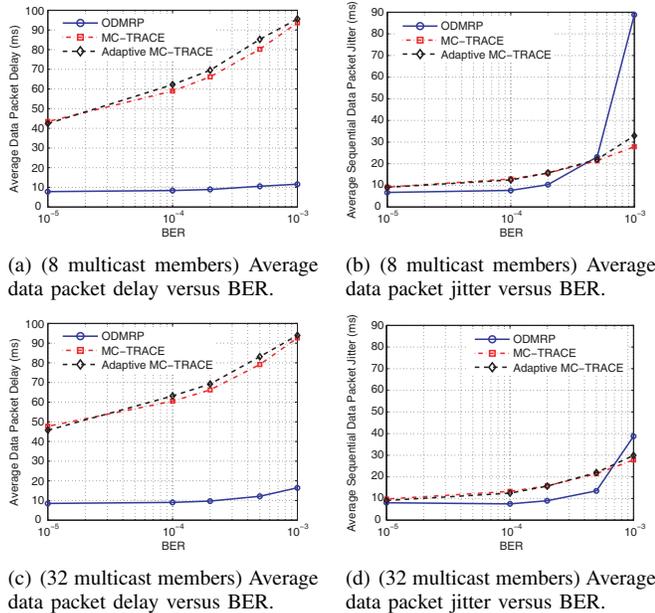
Fig. 8. Average delay and jitter of data packets versus BER (8 and 32 multicast members, 128 nodes).

manages to limit its ARN for low BER values where ODMRP results in higher ARN. With increasing BER Adaptive MC-TRACE forces more nodes to participate in the relaying process resulting in increased ARN. As we approach to high BER levels, increased packet drop rate cripples the data traffic causing ARN to drop.

D. Delay and Jitter

The data packet delay and jitter values presented in Figure 8 are too close to each other to draw a simple conclusion. The fact that MC-TRACE's cyclic superframe structure makes it virtually impossible to achieve data packet delay values lower than the superframe time $(32ms)$ causes higher delay values for MC-TRACE compared with ODMRP. Jitter values, however, remain comparable for all the protocols for low to mid BER levels. ODMRP performs worse as we reach high BER levels.

VII. CONCLUSIONS

In this paper we proposed an adaptive redundancy algorithm, implemented on a tree-based multicasting approach, and compared it to a mesh-based multicast protocol (ODMRP). We explored the limits of redundancy while trying to strike a balanced trade-off between the amount of redundancy and energy consumption. Through ns-2 simulations, we showed that adaptive redundancy maintains high packet delivery ratios at higher protocol efficiency when compared to both non-adaptive mesh and tree-based multicast protocols.

As our future work, we plan to implement our approach for different types of multicast protocols. For example, we believe that the poor performance of highly redundant ODMRP can be improved by adapting the periodic update parameters according to the channel conditions. Adapting redundancy to current channel conditions will allow multicast protocols to be more responsive and provide an ideal trade-off between QoS and efficiency.

REFERENCES

[1] B. Tavli and W. B. Heinzelman, "MC-TRACE: Multicasting through time reservation using adaptive control for energy efficiency," in *Proceedings of the IEEE Military Communications Conference*, 2005.

[2] S. J. Lee, W. Su, and M. Gerla, "On-demand multicast routing protocol in multihop wireless mobile networks," *ACM/Kluwer MONET*, vol. 7, pp. 441–453, 2002.

[3] H. Moustafa and H. Labiod, "A performance comparison of multicast routing protocols in ad hoc networks," in *IEEE PIMRC*, 2003, pp. 497–501.

[4] D. Bertsekas and R. Gallager, *Data Networks*. Prentice Hall, 1992.

[5] C. W. Wu and Y. C. Tay, "Amris: a multicast protocol for ad hoc wireless networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, 1999, pp. 25–29.

[6] J. G. Jetcheva and D. B. Johnson, "Adaptive demand-driven multicast routing in multi-hop wireless ad hoc networks," in *ACM MOBIHOC*, 2001, pp. 33–34.

[7] X.-H. Lin, Y.-K. Kwok, and V. Lau, "A quantitative comparison of ad hoc routing protocols with and without channel adaptation," *IEEE Trans on Mobile Computing*, vol. 4, no. 2, pp. 111–128, Apr 2005.

[8] R. Agrawal, "Performance of routing strategy (bit error based) in fading environments for mobile adhoc networks," in *IEEE Personal Wireless Communications Conference*, Jan 2005, pp. 554–556.

[9] N. Wisitpongphan, G. Ferrari, S. Panichpapiboon, J. Parikh, and O. Tonguz, "Qos provisioning using ber-based routing in ad hoc wireless networks," in *IEEE Vehicular Technology Conference*, May 2005, pp. 2483–2487.

[10] L. Tan, P. Yang, and S. Chan, "An error-aware and energy efficient routing protocol in manets," in *Computer Communications and Networks ICCCN*, Aug 2007, pp. 1095–2055.

[11] R. Vaishampayan, J. Garcia-Luna-Aceves, and K. Obraczka, "An adaptive redundancy protocol for mesh based multicasting," *Elsevier Computer Communications*, vol. 30, no. 5, pp. 1015–1028, Mar 2007.

[12] M. Burmester, T. V. Le, and A. Yasinsac, "Adaptive gossip protocols: Managing security and redundancy in dense ad hoc networks," *Elsevier Ad-hoc Networks*, vol. 5, no. 3, pp. 313–323, Apr 2007.

[13] M. Yoshida, M. Terada, and T. Miki, "Adaptive sector-based flooding for mobile ad hoc networks," *IEICE Transactions on Communications*, vol. 90, no. 4, pp. 788–798, 2007.

[14] B. Tavli and W. B. Heinzelman, "MH-TRACE: Multi hop time reservation using adaptive control for energy efficiency," *IEEE Journal on Selected Areas of Communications*, vol. 22, no. 5, pp. 942–953, June 2004.

[15] S. Lee and C. Kim, "Neighbor supporting ad hoc multicast routing protocol," in *MobiHoc*, Aug 2000.

[16] S. Lee and M. Gerla, "On-demand multicast routing protocol," in *IEEE WCNC*, Sep 1999.

[17] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Communications Magazine*, vol. 43, pp. 23–30, 2005.

[18] B. Tavli and W. B. Heinzelman, "Energy and spatial reuse efficient network wide real-time data broadcasting in mobile ad hoc networking," *IEEE Trans on Mobile Computing*, vol. 10, pp. 1297–1312, 2006.