

Multichannel Time-Spread Scheduling: A New Approach to Handling Heavy Traffic Loads in Ad Hoc Networks

Xuejun Tian

Department of Information Systems
Aichi Prefectural University
Nagakute Aichi, Japan
Email: tan@ist.aichi-pu.ac.jp

Yuguang Fang

Department of Electrical
and Computer Engineering
Gainesville, Florida 32611, USA
Email: fang@ece.ufl.edu

Tetsuo Ideguchi

Department of Applied Information
Science and Technology
Aichi Prefectural University
Nagakute Aichi, Japan
Email: ideguchi@ist.aichi-pu.ac.jp

Abstract—Wireless mobile networks that do not have infrastructure or centralized administration, known as the ad hoc networks, have received considerable attention in the last few years. The salient characteristics of such networks—time-varying topology and lack of centralized control design—have made medium access control design more complicated and challenging, which is particularly when multiple channels are employed. Recently, many multichannel transmission protocols, such as Collision-Avoidance Transmission Scheduling (CATS), have been investigated for their higher efficiency although their problems are abundant. In this paper, we propose a new protocol, namely, Multichannel Time-spread Scheduling (MATS), to improve the throughput performance under heavy traffic loads. In MATS, nodes with transmission requests are divided into three groups, and carry out channel reservations in parallel with a short overhead. We carry out simulation study and the results show that the performance of this protocol under high traffic loads is significantly improved.

I. INTRODUCTION

Advances in wireless technologies and portable computing, along with high demands for greater user mobility have driven the development of an emerging class of self-organizing, rapidly deployable networks, namely, ad hoc networks [1][2][3]. An ad hoc network is a collection of wireless mobile nodes that dynamically form a network without the use of any existing network infrastructure or centralized administration. It has found many applications in military tactical mission, disaster rescue and various conference events.

One important issue in the ad hoc networks is the design of the medium access control (MAC) protocol [4] [5][6][7]. MAC protocols can be divided into two dominant types: random access and scheduled access. Although numerous collision-avoidance protocols have been proposed, some of which are capable of solving the hidden-terminal problem in ad hoc networks [8][9], these protocols do not efficiently support broadcasting or multicasting.

Scheduled access guarantees successful information exchanges using reserved links without collision. Previous

studies on scheduling have concentrated on single channel systems. Recently, multichannel systems have received intensive attention because of the advantages offered by such systems[10][11]. Multichannel systems outperform single channel systems in several aspects: (1) multichannel Time Division Multiple Access (TDMA) systems are usually more reliable; (2) individual channel operates at a lower rate, and synchronization is easier in multichannel TDMA systems; and (3) multichannel TDMA systems have greater flexibility in response to system growth because they allow the addition of new channels[11]. Different from some studies which have focused on distributed sub-optimal solutions [12][13][14], which are designed either for broadcasting or unicasting, recently, a multichannel scheduled-access protocol named collision-avoidance transmission scheduling (CATS) has been proposed[15] that can simultaneously support unicasting, multicasting and broadcasting. In CATS, there are five mini-slots used for channel reservation. In the first two mini-slots, the nodes attempting to reserve a channel detect if intended channels are available while the nodes occupying existing links send signals (Beacons) to keep the reserved channel from being interfered or reserved. All nodes send their reservation requests, if any, in one mini-slot, i.e., in the third mini-slot, which may result in high contention probability in this mini-slot. In the following two mini-slots, intended receiving nodes exchange information to confirm if the reservations are successful. However, because all nodes send reservation requests in one mini-slot simultaneously, CATS poses several unresolved problems, including the sudden throughput drop-off problem, i.e., the throughput quickly drops near zero as the traffic load increases. Another problem is that in CATS the broadcasting and the multicasting cannot be set to high priority over unicasting because broadcasting and multicasting requests are treated the same as unicasting requests, which may result in the scenario that broadcast transmission or multicasting transmission cannot be established except when the unicast traffic load is very low.

In this paper, we propose a new scheduling protocol for multichannel TDMA ad hoc networks, namely, MATS

This work was supported in part by the Office of Naval Research under Young Investigator Award N000140210464 and under grant N000140210554.

(Multichannel Time-spread Scheduling), that overcomes the problems in CATS mentioned above. In MATS, nodes with transmission requests are divided into three groups, one for broadcast and multicast, two for unicast. The nodes in these groups carry out link reservations in parallel with a short overhead. MATS is distributed and supports unicasting, multicasting and broadcasting simultaneously. MATS also allows the multicasting and broadcasting to be carried out prior to unicasting. Performance evaluation results show that, in comparison to the existing protocol CATS, MATS gives higher throughput and is more robust under high traffic loads.

The remainder of this paper is organized as follows. Section 2 gives background material and definitions. In Section 3, we describe MATS. And then, we carry out the performance evaluation using simulations in Section 4. Section 5 concludes the paper.

II. MODEL AND DEFINITIONS

An ad hoc network is a collection of communication devices (nodes). Every node can reach a given subset of other nodes, depending on the power of its transmitter and the topographic characteristics of the surrounding region. An ad hoc network can be modeled as a set of network nodes and a set of edges between nodes with direct links. Nodes linked by an edge are considered to be neighbors. Here, we assume that every node has the same transmitting power, which makes the reachability graph of the network symmetric. Each node sends messages in synchronized time slots. In every time slot, each node acts either as a transmitter or a receiver. The node acting as a receiver in a given slot receives a message if exactly one of its neighbors transmits in this slot. If more than two neighbors transmit in the same slot, the node receives none of the messages. In such cases, a conflict or collision occurs. Here, we assume that every node is capable of detecting collisions (i.e., determine whether the transmitted message is correctly received or not). Multiple radio channels, termed as multichannel, are available, which are divided into different radio channels, a control channel (CCH), a broadcast data channel (BCH), and data channels (DCH). The CCH is used for transmission of control packets, the BCH for broadcasting, and the DCH for multicasting and unicasting. A channel reservation means that a node reserves a time slot and a radio channel for transmission and we will use radio channel to denote radio frequency channel and slot channel to denote the time slot channel.

III. SCHEDULING PROTOCOL: MATS

As the traffic load in an ad hoc network increases, the probability of collisions among node reservation requests increases, which results in low throughput. To solve this problem, the proposed protocol, MATS, carries out channel reservation in such a way that nodes with reservation requests are divided into three groups, transmit their requests asynchronously in different mini-slots and carry out reservations in parallel.

As shown in Fig. 1, a time frame consists of L slots in MATS, every node reserves a slot for transmission. Each slot

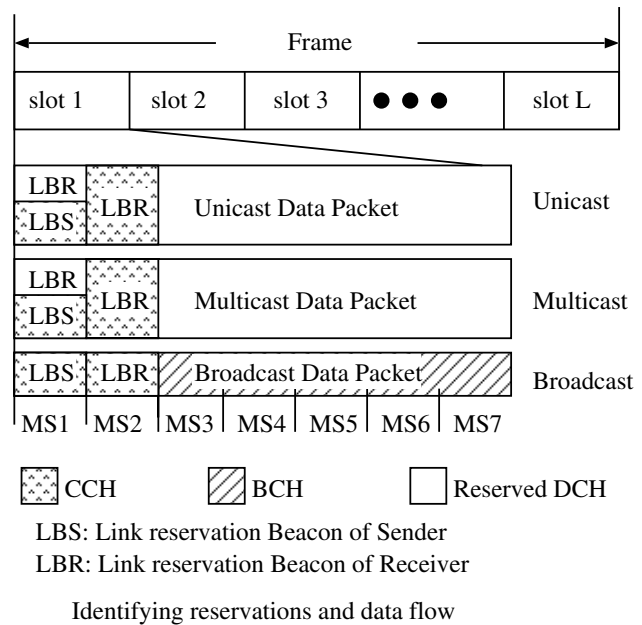
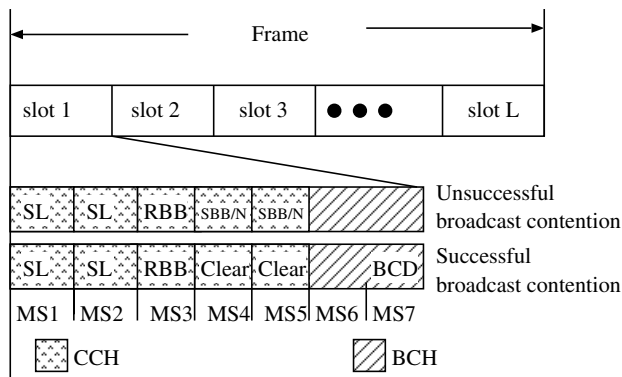


Fig. 1. Basic operations in MATS

has two parts, one part consisting of six mini-slots (MS1–6) used for reserving channels and the other part consisting of a single mini-slot (MS7) used for data transmission. Small control packets called *beacons* containing necessary reservation information are sent during MS1–6. In general, a beacon specifies (a) the source address, (b) the destination address, (c) the reserved or intended broadcast and multicast slots, and (d) the reserved or intended data channel.

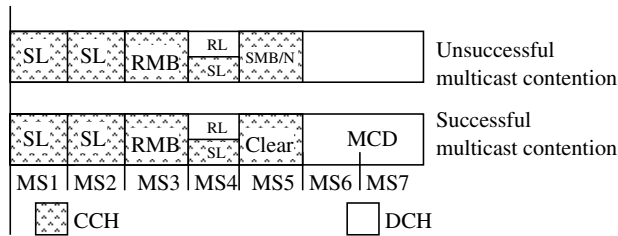
Figure 1 illustrates how data are transmitted over reserved links without interruptions. Every node attempting to transmit in the current slot sends an LBS (Link reservation Beacon of Sender) over the reserved CCH in MS1 to prevent other nodes from establishing multicast or broadcast links, while the intended receiving node during the current slot sends an LBR (Link reservation Beacon of Receiver) over the reserved DCH to prevent other nodes from establishing unicast links with the same DCH. In MS2, the intended receiving node in the current slot sends an LBS over the CCH to prevent other nodes from attempting to reserve multicast or broadcast links again. For a node with broadcast or multicast request, only when detecting the CCH clear in both of MS1 and MS2, which means that none of its neighbors is sending or receiving, will the node continue the reservation process. For a node with unicast request, it just needs to know none of its neighbors is receiving over the intended DCH to continue the reservation. Otherwise, nodes with requests will abort their reservation processes. For convenience, we refer to a node in reservation process as active node (or sender) in the mini-slot of interest. By sending LBS and LBR, the reserved links are prevented from being reserved and used again by other nodes.

Figures 2(a) and (b) show the processes for reserving broadcast and multicast links, respectively. To reserve a broadcast channel, an active node sends an RBB (Request Broadcast



RBB: Request Broadcast Beacon, BCD: BroadCast Data
SBB/N: Stop Broadcast Beacon/Noises, SL: Sender Listens

(a) Making a reservation for broadcast



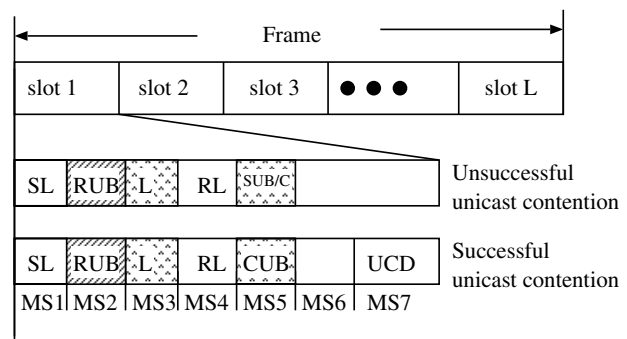
RMB: Request Multicast Beacon
SMB/N: Stop Multicast Beacon/Noise
MCD: MultiCast Data, SL: Sender Listens, RL: Receiver Listens

(b) Making a reservation for multicast

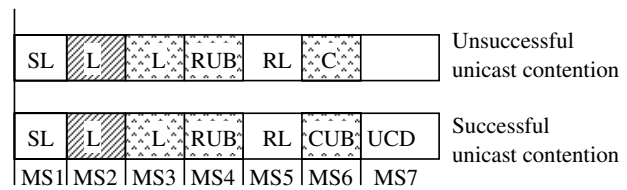
Fig. 2. Making reservations for broadcast and multicast

Beacon) over the CCH in MS3. At the same time, nodes that are neither sending nor receiving listen on the CCH in MS3, and if they detect a collision, they send an SBB (Stop Broadcast Beacon) in MS4. If the active node detects an SBB or noise, it realizes that its reservation request has failed and stops making the reservation in the remaining mini-slots and will not be an active node. Otherwise, it continues listening on the CCH in MS5 and in case of clear channel, it successfully reserves a channel. To reserve a multicast, an active node sends an RMB (Request Multicast Beacon) over the CCH in MS3 and listens to determine if there is an SBB in MS4 sent by a node that detected a collision in MS3. The node receiving an RMB listens to the intended DCH in MS4 and, if it detects that the DCH is clear, it sends no signal; otherwise it sends an SBB over CCH in MS5 as a negative acknowledgment. Only after detecting no signal in MS4 and MS5, does it recognize that the multicast request is successful.

The algorithm for reserving a unicast link is shown in Fig. 3. To distribute reservations for unicast, active nodes are divided into two groups, referred to as NTRU1 (Node To Reserve a Unicast, group 1) and NTRU2, according to the procedure outlined below. In the following, assuming Node A is trying to reserve a unicast link to send data to Node B, we present the algorithms for two cases: (1) Node A belongs to NTRU1, and (2) Node A belongs to NTRU2.



(a) The first group of nodes making a reservation for unicast



(b) The second group of nodes making a reservation for unicast

RUB: Request Unicast Beacon, UCD: UniCast Data
SL: Sender Listens, CUB: Concur with Unicast Beacon
RL: Receiver Listens, SUB/C: Stop Unicast Beacon/Clear
L: node without receiving and sending links Listens

Fig. 3. Making a reservation for unicast

i) The algorithm for Node A in NTRU1 to reserve a unicast link (Fig. 3 (a)).

- MS1: Every receiver of an existing link sends an LBR over a reserved DCH. Node A listens on the intended DCH and, if the DCH is not clear, stops making the unicast reservation.
- MS2: Node A sends an RUB (Request Unicast Beacon) over a TCH (Temporary Channel)—a DCH determined beforehand and known by all nodes. Nodes that do not have links and do not need to reserve a link listen on this TCH.
- MS3: Nodes A and B listen on the CCH like other receiver candidates.
- MS4: If Node A receives an RBB or RMB as the intended receiver in MS3, then it interrupts its reservation and behaves as a receiver of a broadcast or multicast in the remaining mini-slots. If Node A detected a collision in MS3, it sends an SBB over the CCH in this slot. If Node B confirms that the CCH was clear or has a collision in MS3 and receives an RUB as the intended receiver in MS2, it listens on the DCH indicated in the RUB.
- MS5: If Node B confirms that the DCH indicated in the RUB was clear in MS4, it sends a CUB (Concur with Unicast Beacon) over the CCH. If the intended DCH is not clear in MS4, Node B sends an SUB (Stop Unicast Beacon) in case of receipt of a collision in MS3 or nothing if there is no collision in MS3. Only when Node

A receives a CUB, is the reservation successful.

The TCH used in MS2 is selected from the set of DCHs. The TCH can be an arbitrary DCH, which is known by all nodes, because the use of TCH will not affect the reserved links and the existing link transmissions do not begin before MS2. In MS3, the node that sends or receives an RUB in MS2 also has the possibility of receiving an RBB or RMB as an intended receiver. The form of the scheduling algorithm depends on the priority policies employed. In the above algorithm, the broadcast and multicast are treated prior to the unicast in MS4 and, even if broadcast and multicast could not succeed, a interested node still has the opportunity to reserve a unicast link.

ii) The algorithm for Node A in NTRU2 to reserve a unicast link (Fig. 3 (b)).

MS1: The behaviors of nodes are the same as those described for NTRU1 above.

MS2: Nodes A and B listen on the TCH like other receiver candidates.

MS3: Nodes A and B listen on the CCH like other receiver candidates.

MS4: If Node A correctly receives an RBB or RMB in MS3 as an intended receiver, it stops making a reservation and behaves as a broadcast or multicast receiver in the remaining mini-slots. Similarly, if Node A receives an RUB sent from a node of NTRU1 in MS2, it stops its own reservation and behaves as a receiver candidate. Otherwise, Node A sends an RUB over the CCH.

MS5: Upon receipt of an RUB from Node A in MS4, Node B listens on the indicated DCH.

MS6: If Node B detects that the intended DCH is clear in MS5, it sends a CUB over CCH and, if Node A correctly receives the CUB, the unicast link is established successfully.

Here, unicast senders are divided into two groups NTRU1 and NTRU2 as follows in order to spread reservation requests. We divide radio channels into two sets with c_1 and c_2 radio channels, respectively, where $c_1 + c_2 = c$. Before a node sends RUB, it randomly selects a radio channel, with probability c_1/c , it selects a channel from first set of radio channels and otherwise from the other set. Thus, radio channel used by a node in NTRU1 is different from that in NTRU2, which guarantees that no interference between links established by NTRU1 and NTRU2 in case that one or more receiver is the common neighbor of two senders using the same time slot and radio channel.

As shown above, in MATS, nodes with broadcast, multicast and unicast send requests RBB, RMB and RUB in different mini-slots, that RBB and RMB in MS3, RUB of NTRU1 in MS2 and RUB of NTRU2 in MS4, which leads to a lower contention probability. In contrast, all requests are sent in one mini-slot in CATS. In MATS, when nodes of one group send requests, the other nodes can listen and receive the requests from other nodes if there is no collision. In the meantime,

the reserving processes of three groups are parallel, so MATS does not have much overhead. It can be expected that MATS will have a better performance.

We can also confirm that MATS can simultaneously establish transmission links for broadcast, multicast and unicast without collisions in the subsequent transmissions, here we omit the explanation.

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of MATS.

A. Simulation Scenario

The simulation model we are using is shown in Fig. 4, in which each node is assumed as a Poisson source with the same arrival rate for packet generation and has eight neighbors. When a node generates a unicast request, it selects a neighbor randomly as its receiver. To avoid edge effect, we assume that nodes in the low edge shown in Fig. 4 are the neighbors of those in the up edge, and nodes in the left edge are the neighbors of those in the right edge, so the topological model becomes a virtual ring (wrap-around model). We assume that each node can only buffer one message and simply discards any message passed from the upper layer if there is any message in the buffer. The message is discarded if the reservation for it is unsuccessful. This means that a node can attempt to reserve a link in every slot, a worst case scenario to test the effectiveness of our protocol. We refer to all the data, which is divided into a series of slots to be sent, as a message flow or flow, and the average number of slots of a flow is referred to as AFL (Average Flow Length). The frame length L of MATS for a node to transmit once in every frame in the worst case is set to $2d$ slots, assuming that there are at least $c = d$ data channels available. Otherwise, we use the same settings used in the theoretical evaluation for CATS for comparison.

B. Results and Discussions

Figure 5 shows throughput results for CATS and MATS when only unicast is present with different AFLs. The throughput is expressed as the average number of links in one frame for one node. As shown in Fig. 5, the throughputs for CATS rise to the peaks and then quickly drop down to zero as the traffic load increases. For MATS, however, we observe the throughput drop-off problem has been mitigated significantly and the throughputs are much higher than that for CATS, and we find only the throughput drop-off for MATS with AFL=1 at a relatively higher arrival rate. In the cases that AFL is more than 1, the throughputs for MATS do not have drop-off, but a little decrease in the range of high traffic load.

Next, we study the throughput when both of unicast and broadcast are present. Here, broadcast link does not consider a pair of a sender and a receiver, but a sender and multiple receivers, and the throughputs for broadcast and unicast are different. As we mentioned before, the throughput is expressed as the average number of links in one frame for a node. For our simulated network, one broadcast transmission has 8 links

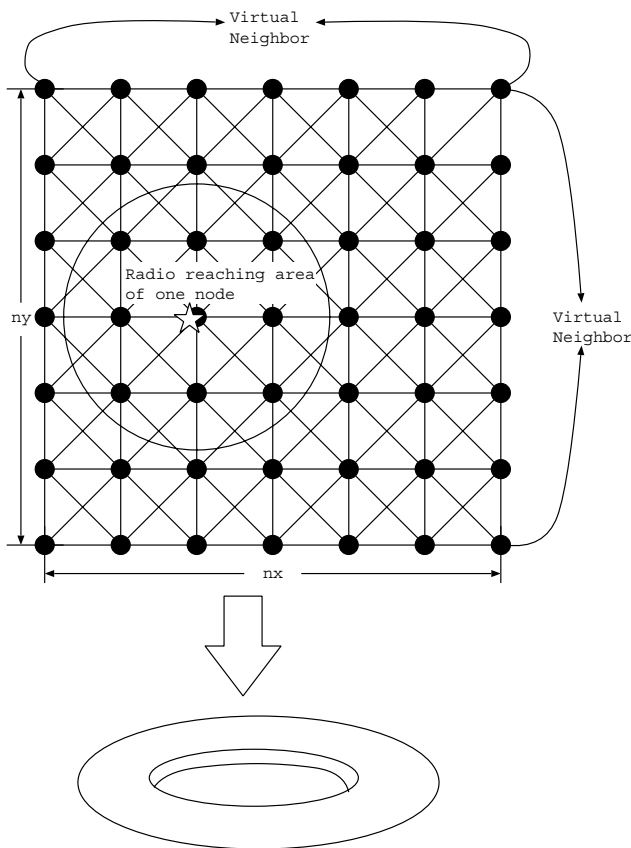


Fig. 4. Simulation Model

if one node has 8 neighbors. In the simulation, the same as in the analysis before, one node can reserve no more than one slot per frame in case of either unicast or broadcast transmission. To evaluate the broadcast transmissions, we present the results that show the throughput changes as the unicast load varies while the broadcast load is fixed at certain level, say, 0.1. In the Fig. 6, 7 and 8, we present the throughputs with AFL=1, AFL=2 and AFL=10, respectively. There are three pairs of curves for the total throughput, broadcast throughput and unicast throughput, respectively. Each pair includes two curves which are throughputs for MATS and CATS, respectively. The total throughput is the sum of broadcast and unicast throughputs.

Fig. 6 shows throughputs with AFL=1, in which we observe that the total throughput for MATS is considerably higher than that for CATS. For CATS, broadcast and unicast throughputs are much low and drop down nearly to zero as the unicast arrival rate increases. For MATS, broadcast and unicast throughputs show relatively higher and the unicast throughput drops down but not to zero. The broadcast throughput increases slightly when unicast throughput decreases. Because each node can only buffer one message, if an old message is not sent out successfully, it will be replaced by a newly arriving one. Thus, the probability of broadcast link reservation will decrease as the unicast arrival rate increases. Since RBB for broadcast link reservation is sent in MS3, which is not interfered with

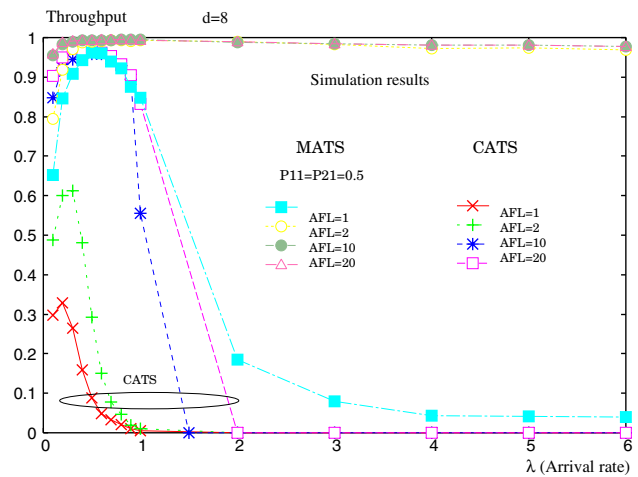


Fig. 5. Throughput results for CATS and MATS in case of unicast

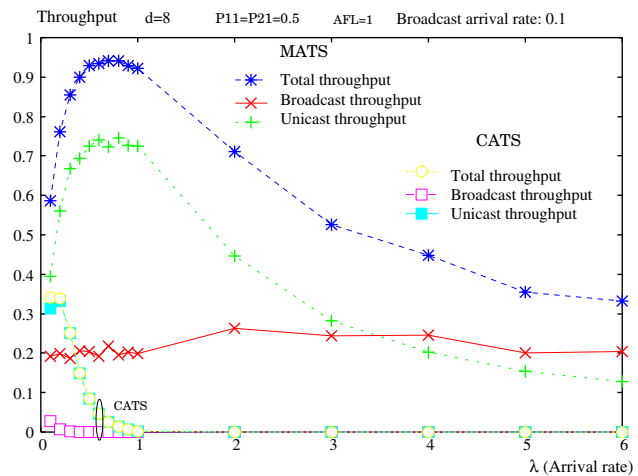


Fig. 6. Throughput results for CATS and MATS with broadcast arrival rate 0.1 and AFL=1

RUB that is sent in MS2 or MS4, moderate decrease in the number of nodes sending RBB will increase the successful rate of broadcast link reservation. Certainly, excessive decrease in the number of nodes sending RBB results in lower broadcast throughput.

Figure 7 shows the throughputs with AFL=2, in which the total throughput for MATS has no sudden drop-off and broadcast throughput is considerably higher than that for CATS. The unicast throughput also has no sudden drop-off that occurs in case of CATS, but in the range of low arrival rate, is somehow the same as that for CATS. This is because a node cannot receive from more than one node simultaneously, if broadcast throughput increases, then unicast throughput will decrease. As unicast arrival rate continues to rise, the unicast throughput increases and broadcast throughput decreases.

Figure 8 shows the throughputs with AFL=10. We observe the similar changes as shown in Fig. 7. Both of broadcast and unicast throughputs for MATS have no sudden drop-off and the broadcast throughput becomes slightly higher than that for

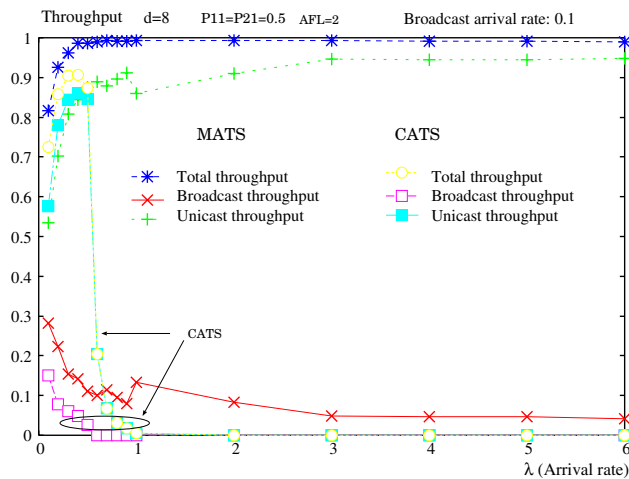


Fig. 7. Throughput results for CATS and MATS with broadcast arrival rate 0.1 and AFL=2

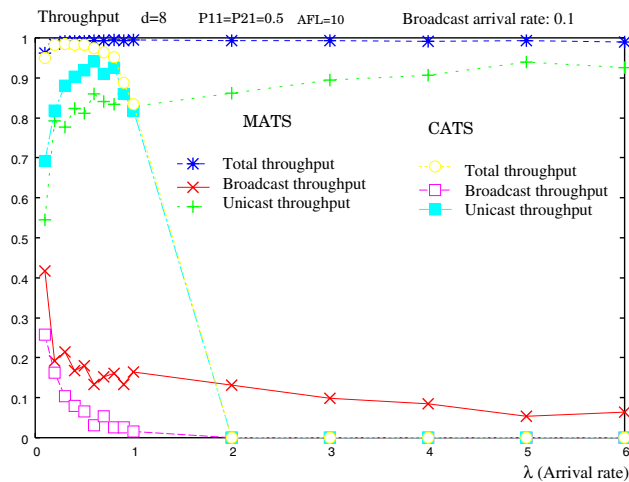


Fig. 8. Throughput results for CATS and MATS with broadcast arrival rate 0.1 and AFL=10

the case when $AFL=2$ because of a longer AFL.

Similarly, in Fig. 7, the broadcast and unicast throughputs for MATS are relatively compared. When contentions between nodes with broadcast and unicast occur, their behaviors depend on the priority policy. For example, in MATS, when a node receives an RBB, it will abort the other unicast reservation process, and if a node receives an RUB in MS2 from a node in NTRU1, it will not listen or send an RUB in MS4. Certainly, different priority policy can be adapted in MATS.

V. CONCLUSIONS

In ad hoc networks with shared resource among various users, efficient scheduling of transmissions is crucial to achieve acceptable network performance. In this paper, we have proposed a new reservation protocol, namely, MATS that can avoid sudden throughput drop-off by dividing the nodes into different groups to decrease the probability of collisions during channel reservation process. We present simulation results by taking unicast and broadcast into consideration and show that

MATS has several desirable features: (1) MATS can increase throughput without sudden throughput drop-off under heavy traffic loads; (2) the reservation scheme in MATS is distributed and carried out in parallel with minimal overhead; and (3) the broadcast, multicast and unicast can be scheduled separately and can be assigned with different priorities without much interference. We demonstrate that MATS can overcome several problems in a well-known reservation scheme CATS.

REFERENCES

- [1] Special issue on wireless Ad Hoc networks, IEEE J-SAC, 17(8), Aug. 1999.
- [2] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad hoc wireless networks," IEEE/ACM Transactions on Networking, 10(4):477-486, August 2002.
- [3] L. Butty'an and J.-P. Hubaux, "Enforcing Service Availability in Mobile Ad-Hoc WANs," In Proceedings of the 1st IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing (MobiHOC), Boston, August 2000.
- [4] C. Zhu and M. S. Corson, "A five phase reservation protocol (FPRP) for a mobile ad hoc networks," in Proc. IEEE INFOCOM, 1998.
- [5] L. C. Pond and V. O. K. Li, "A distributed time-slot assignment protocol for mobile multi-hop broadcast packet radio networks," In Proc. IEEE MILCOM, 1989.
- [6] IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," IEEE Standard 802.11-1997. IEEE, New York, NY, 1997.
- [7] I. Chlamtac and A. Farago, "Making transmission schedules immune to topology changes in multi-hop pocket radio networks," IEEE/ACM Trans. Networking, vol. 2, pp. 23-29, Feb. 1994.
- [8] V. Bharghavan, A. Demers, S. Shenkar, and L. Zhang, "MACAW: a media access protocol for wireless LANs," in Proceedings of the SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications, PP. 212-225, August 1994.
- [9] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," in Proc. ACM SIGCOMM, 1997.
- [10] J.-H. Ju and V. O. K. Li, "TDMA scheduling Design of multihop packet radio network based on Latin squares," IEEE J-SAC, 17(8), pp. 1345-1352, August 1999.
- [11] J. F. Chang and T. H. Yang, "Multichannel ARQ protocols," IEEE Trans. Commun., vol. 41, pp. 592-598, Apr. 1993.
- [12] A. Ephremides and T. Truong, "Scheduling broadcasts in multihop radio networks," IEEE Transactions on Communications, COM-38 (4), April 1990.
- [13] I. Cidon and M. Sidi, "Distributed assignment algorithms for multihop packet radio networks," IEEE Trans. on Computers, 38:1353-1361, 1989.
- [14] J.-H. Ju and V. O. K. Li, "An optimal topology-transparent scheduling method in multihop pocket radio networks," IEEE/ACM Trans. Networking, vol. 6, pp. 298-306, June 1998.
- [15] Z. Tang and J.J. Garcia-Luna-Aceves, "Collision-avoidance transmission scheduling for Ad-Hoc networks," in Proc. IEEE ICC'2000.