MATS: Multichannel Time-Spread Scheduling in Mobile Ad Hoc Networks

Xuejun Tian, Yuguang Fang, *Senior Member, IEEE*, and Tetsuo Ideguchi

***Abstract*— Wireless Mobile ad hoc networks (MANETs) have received considerable attention in the last few years. Most research works focus on single-channel MANETs with a single power-level in order to simplify the network design and analysis. How to take advantage of multiple channels and multiple power levels in MANETs poses a serious challenging problem. Recently, a few multichannel transmission protocols such as Collision-Avoidance Transmission Scheduling (CATS) have been proposed to harvest the advantage of high transmission efficiency when multiple channels are deployed. Although such protocols do provide ways to coordinate the use of multiple channels, there exist some serious problems such as the throughput fast drop-off under heavy traffic loads. In this paper, we propose a new protocol, namely, MultichAnnel Time-Spread Scheduling (MATS), which attempts to tackle these problems. In MATS, nodes with transmission requests are divided into three groups, which carry out channel reservations in parallel and can simultaneously support unicasting, multicasting and broadcasting at the link level. MATS ensures successful and collision-free data transmissions using the reserved channels and allows multicasting and broadcasting high priorities over unicasting. Both theoretical analysis and extensive simulation studies are carried out which show that the performance of MATS under high traffic loads significantly outperforms the existing schemes.**

***Index Terms*— Mobile ad hoc networks, MAC, multichannel network, channel reservation.**

I. INTRODUCTION

ADVANCES in wireless technology and portable computing, along with high demands for user mobility have driven the development of an emerging class of self-organizing and rapidly deployable networks referred to as *ad hoc networks* [1] [2] [3]. A mobile ad hoc network (MANET) is a collection of wireless mobile communication nodes that can dynamically form a communication network on the fly without the use of any existing network infrastructure or centralized administration. These kinds of networks have found applications in tactical battle field, disaster rescue and conventions.

One important issue in mobile ad hoc networks is the the medium access control (MAC) protocol [4] [5] [6] [7], which attempts to efficiently coordinate the use of the shared communication medium. MAC protocols can be classified into two major categories: random access and scheduled access. Although numerous collision-avoidance protocols have been proposed, some of which are capable of mitigating the effect of the hidden-terminal problem in mobile ad hoc networks [8] [9], these protocols do not efficiently support broadcasting/multicasting and quality of service (QoS).

Scheduled access guarantees successful information exchanges using reserved links without collisions. Previous studies on scheduling have concentrated on single-channel systems. However, how to effectively utilize the multiple channels in MANETs is still a challenging problem. This is particularly true when medium access control (MAC) such as IEEE 802.11 MAC has to rely on the carrier sensing and collision avoidance schemes. Recently, multichannel systems have received a great deal of 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 channels can operate 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]. In view of the superior performance of multichannel systems, we will focus on the scheduling problem in TDMA networks with multiple radio channels (multichannel). Scheduling for such a system consists of allocating stations to different frequencies at different times (or their combinations) so that no collisions occur and efficient spatial reuse of the available bandwidth can be achieved. Effective scheduling can lead to much higher channel efficiency. To date, most works on transmission scheduling algorithms have concentrated on the fair and conflict-free algorithms that maximize the system throughput. However, changes of network topology due to the movements of mobile nodes may render any optimal design obsolete. Many researchers have been searching for distributed sub-optimal solutions [13] [14] [15] [16] [17], which are designed either for broadcasting or unicasting, but not for both. In [18], SYN-MAC protocol is proposed for mobile ad hoc networks, which used binary countdown algorithm for contention resolution. SYN-MAC does gain improved throughput and packet delay. However, the performance of SYN-MAC depends on the number of contention slots and turnaround time, which is a disadvantage for wireless communications. Recently, a multichannel scheduled-access protocol named *collision-avoidance transmission scheduling (CATS)*

Manuscript received January 5, 2004; revised December 3, 2004; accepted January 17, 2005. The associate editor coordinating the review of this paper and approving it for publication was W. Liao. This work was supported in part by National Science Foundation Faculty Early Career Development Award under grant ANI-0093241 and the Office of Naval Research Young Investigator Award under grant N000140210464.

Xuejun Tian is with the Department of Information Systems, Aichi Prefectural University, Aichi, JAPAN (email: tan@ist.aichi-pu.ac.jp).

Yuguang Fang is with the Department of Electrical and Computer Engineering, University of Florida, 435 Engineering Building, P.O. Box 116130, Gainesville, FL 32611 USA (email: fang@ece.ufl.edu).

Tetsuo Ideguchi is with the Department of Applied Information Science and Technology, Aichi Prefectural University, Aichi, JAPAN.

Digital Object Identifier 10.1109/TWC.2006.03018

has been proposed [19] to 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 whether the intended channels are available and the nodes with existing links send signals (Beacon) to keep already reserved channels from being interfered. In the third mini-slot, all nodes send their reservation requests, if any, which may result in high contention probability. In the next two mini-slots, intended nodes confirm and inform if requested links are acceptable. However, because all nodes send reservation requests in one mini-slot (i.e., the third mini-slot) simultaneously, CATS has several unresolved problems such as the sudden drop-off problem: the throughput drops to almost zero as the traffic load increases. Although a backoff algorithm can be introduced to mitigate this problem, the cost is high because all nodes contend on one single mini-slot. Another problem is that broadcasting and multicasting cannot be set different priorities over unicasting because broadcasting and multicasting requests are treated equally as the unicasting requests, which results in the situation that broadcast transmission cannot be established unless unicast traffic load is very low.

In this paper we propose a new protocol for multichannel TDMA ad hoc networks, referred to as MATS (Multichannel Time-Spread Scheduling) that overcomes the existing problems in CATS. In MATS, nodes with transmission requests are divided into three groups, one for broadcast and multicast, and two for unicast. The nodes in these groups carry out link reservations in parallel with a short overhead. MATS is distributed and simultaneously supports unicasting, multicasting and broadcasting. MATS also allows the multicasting and broadcasting to bear higher (or lower) priority over unicasting. Analysis of MATS shows that, in comparison to the existing protocols, it gives higher throughput and is more robust under high traffic load.

The remainder of this paper is organized as follows. Section 2 presents the background materials. In Section 3, we describe our protocol MATS and demonstrate its correctness. We give both theoretical results and simulation results on the performance of this new protocol. Section 5 concludes the paper.

II. MODEL AND DEFINITIONS

An ad hoc network is a collection of communication devices referred to as nodes, which can exchange information. Every node can reach a given subset of other nodes, depending on the transmitting power and the topographic characteristics of the surrounding area. An ad hoc network can be thought of as a set of network nodes and a set of edges between nodes capable of reaching each other. 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 synchronous 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 one neighbor transmit to a node in the same slot, a collision occurs and the node receives none of the messages. Here, we assume that every receiver is capable of determining whether the transmission is successful or not (in our protocol description, we will use "collision" to mean that the receiver does not receive the intended message when there is no confusion). We also assume multiple radio channels (multichannel) are available which are classified 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 DCHs for multicasting and unicasting. A channel reservation means that a node reserves a time slot and a radio frequency for transmission, two factors used to determine a channel. We will use radio channel to denote radio frequency and slot channel to denote time slot to avoid possible confusion.

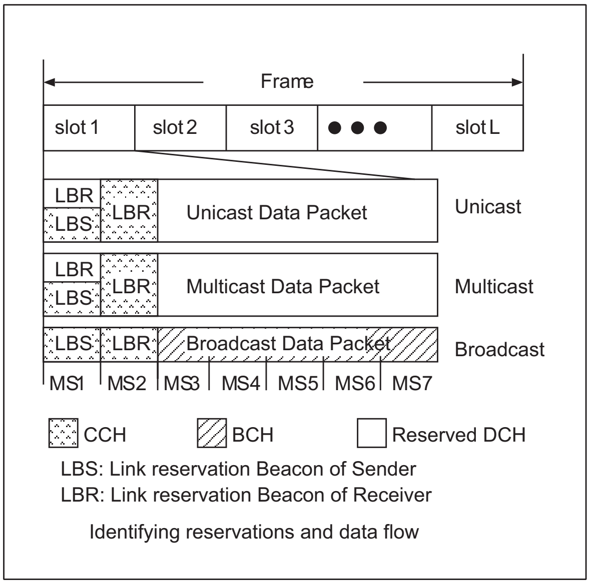


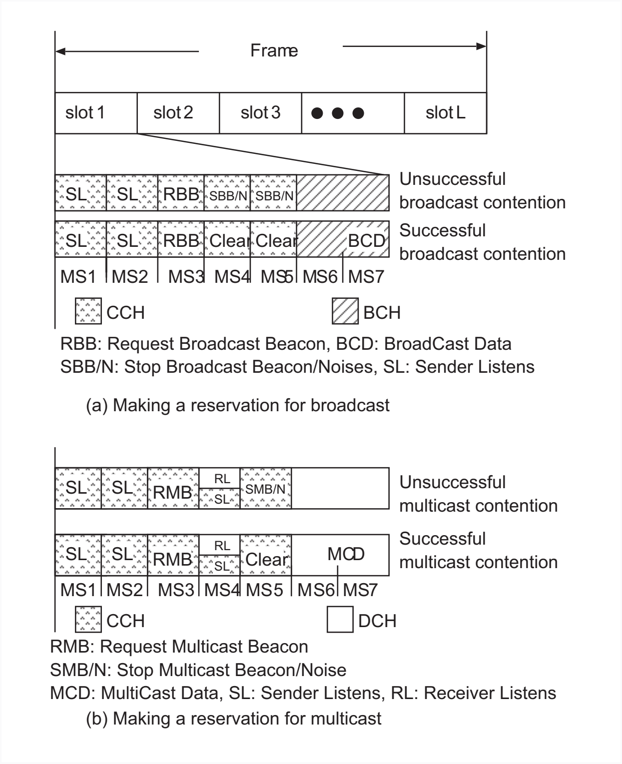
Fig. 1. Basic operations in MATS.

III. RESERVATION PROTOCOL MATS

A. MATS

As the number of active nodes in an ad hoc network increases, the number of collisions among nodes requesting for channels increases, which results in low throughput. Although a backoff algorithm can be introduced to resolve the collisions, the cost is high because many time slots will be wasted in case of unsuccessful reservation. To solve this problem, the proposed protocol, MATS, carries out channel reservation in a way that nodes with reservation requests are divided into three groups, send their requests asynchronously in different mini-slots and carry out reservations in parallel for the remaining reservation process with a short overhead.

As shown in Fig. 1, one time frame consists of L slots in MATS and every node reserves a slot beforehand for transmission. Each slot has two parts, one part consisting of six mini-slots (MS1–6) used for reservation and the other part consisting of a single mini-slot (MS7) used for data transmission. Small control packets called beacons carrying 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



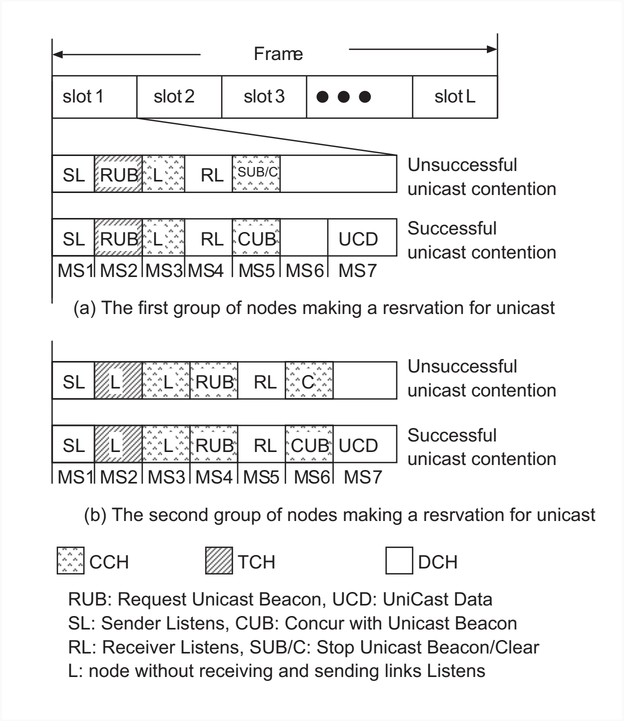
RBB: Request Broadcast Beacon, BCD: BroadCast Data
SBB/N: Stop Broadcast Beacon/Noises, SL: Sender Listens

(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.



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

(b) The second group of nodes making a resrvation 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.

channel. After a node succeeds in making a reservation in MS1-MS6, it will transmit data in the following MS7 and continues transmitting from MS3 to MS7 in the same slot of the following frames until the flow (stream of packets) is over. MS7 can be much longer than the other mini-slots because it is used for data transmission.

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

Figures 2 (a) and (b) show the processes for reserving broadcast and multicast links, respectively. To reserve a broadcast link, an active node, upon detecting clear over CCH in both MS1 and MS2, sends a RBB (Request Broadcast Beacon) over the CCH in MS3. At the same time, nodes that are neither transmitting nor receiving listen on the CCH in MS3, and if they detect a “‘collision” (i.e., they could not decode the broadcast beacon in MS3), they send an SBB (Stop Broadcast Beacon) in MS4. If the active broadcast node detects an SBB or noise in MS4, it realizes that its reservation request has failed (i.e., at least one node could not receive for the broadcast), it will stop making the reservation in the remaining mini-slots and then turn to an inactive mode (backoff will be used, which is not discussed in this paper). Otherwise, it continues listening on the CCH in MS5 and in case of clear status on the CCH, it successfully reserves the channel.

To reserve a multicast channel, an active node with multicast request sends a RMB (Request Multicast Beacon) over the CCH in MS3 after detecting there is no transmission over CCH in MS1 and MS2, and then listens to determine if there is an SBB in MS4 sent by a node that detects a collision in MS3. The node receiving a RMB correctly listens to the intended DCH in MS4 and, if it detects that the DCH is clear, sends back no signal; otherwise it sends an SMB (Stop Multicast Beacon) over CCH in MS5 as a negative acknowledgment. Only after detecting no signal over CCH in MS4 and MS5, does it recognize the multicast request is successful.

As a remark, we have more stringent requirements here for both multicasting and broadcasting. The above proposed reservation process is to guarantee that all nodes (for broadcasting) or the intended nodes (for multicasting) should have clear links from the broadcasting/multicasting source before the transmissions. For certain applications, setting higher priority for broadcasting/multicasting may be necessary.

The algorithm for reserving a unicast link is shown in Fig. 3.

To distribute reservations and resolve high collisions, 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 attempting 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.

i) The algorithm for Node A of 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 busy, stops the unicast reservation process.

MS2: Node A sends a RUB (Request Unicast Beacon) over a TCH (Temporary CHannel)–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 any other receiver candidates.

MS4: If Node A receives a RBB or a RMB as the intended receiver in MS3, then it interrupts its reservation and behaves as a receiver of a broadcast or a multicast in the remaining mini-slots. If Node A detects a collision in MS3, it sends an SBB over the CCH in this slot. If Node B confirms that the CCH is clear or has a collision in MS3 and receives a 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 is 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 a collision detected in MS3 or does nothing if there is no collision detected in MS3. Only when Node A receives a CUB, is the reservation declared successful.

The TCH used in MS2 is selected from the set of DCHs. The TCH can be any arbitrary DCH known by all nodes because the use of TCH will not affect the reserved links due to the fact that the existing link transmissions do not begin before MS2. In MS3, the node that sent or received a RUB in MS2 also has the possibility of receiving a RBB or RMB as an intended receiver. If Node B received RUB in MS2 and no correct RBB or RMB in MS3, it listens over DCH in MS4, then sends SUB or CUB in MS5. In the meantime, the sender of RBB realizes failure of broadcast reservation on finding any signal or noise. In this way, Node B can avoid missing possible unicast reservation. The form of the scheduling algorithm depends on the priority policies. In the above algorithm, broadcast and multicast are treated with high priority over unicast in MS4 in the way that after a node correctly receives RBB or RMB, it abandons its own reservation and the candidate senders of NTRU2 will not send RUB in MS4 as stated afterward. Even if broadcast or multicast reservation does not succeed, the corresponding node still has the opportunity to reserve a unicast link.

ii) The algorithm for Node A of 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 a RBB or RMB in MS3, it abandons its own reservation and behaves as a broadcast or multicast receiver in the remaining mini-slots. Similarly, if Node A receives a RUB sent from a node of NTRU1 in MS2, it stops its own reservation and behaves as a receiving candidate. Otherwise, Node A sends a RUB over the CCH.

MS5: Upon receipt of a 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 then established successfully.

Here, unicast senders are divided into two groups NTRU1 and NTRU2 as follows in order to spread reservation requests in time to quickly resolve possible collisions. There is a probability that the candidate senders of NTRU2 receive RUB from candidate senders of NTRU1 in MS2 and abandon their own requests, which means senders from NTUR1 have a higher priority over senders in NTUR2. We divide available radio channels into two sets C_1 and C_2 with c_1 and c_2 channels, respectively, where $c_1 + c_2 = c$. Before a node sends RUB, it randomly selects a radio channel. If the selected radio channel belongs to C_1, then the node recognizes that it belongs to NTRU1, otherwise it belongs to NTRU2. So, the radio channel used for a node in NTRU1 is always different from that used for a node in NTRU2, which guarantees no interference between links established by nodes in NTRU1 and NTRU2, respectively. The explanation will be given later.

As shown above, in MATS, nodes with broadcast, multicast and unicast send requests RBB, RMB and RUB in different mini-slot, that RBB and RMB in MS3, RUB of NTRU1 in MS2 and RUB of NTRU2 in MS4, leading 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 reservation processes of three groups are in parallel, so MATS only consumes a short overhead. It is be expected that MATS will have a better performance.

B. Correctness of MATS

Next, we show how MATS simultaneously establishes transmission links for broadcast, multicast and unicast without collision in the subsequent transmission. For convenience, we refer to the newly established links in a particular time slot as "new" links in order to distinguish these links from previously established links. Assuming that Node A is acting as a sender, we denote all the neighbors of Node A by N(A).

Theorem 1: *New broadcast or multicast links will not interfere with each other, with the existing links, or with other new unicast links during in data transmission.*

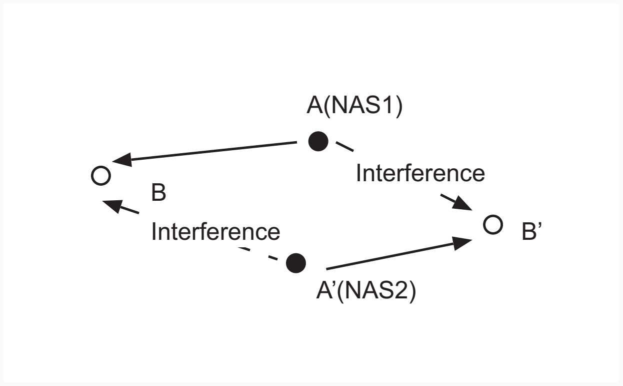


Fig. 4. The interference between reserving links by NTRU1 and NTRU2 nodes.

Proof: 1). We first show that the new broadcast or multicast links will not interfere with existing links. In fact, during the process of reserving a broadcast or multicast link, the sender A (Node A) must listen on the CCH in MS1 and MS2 and, only in the case that the CCH is clear, does it send a RBB or a RMB for the reservation, so the sender A of a new link has no neighbor transmitting or receiving data with other nodes. For broadcast, there is only one channel (BCD) for broadcast. If a neighbor of a candidate broadcast source can hear from the other source sending over the unique broadcast channel BCH, it should be a receiver of the broadcast source and would inform the candidate broadcast source in MS1 and MS2. Then the candidate broadcast source will not send RBB in MS3. So, a successful reservation of broadcast guarantees no interference with other broadcast links.

2). Next, we show that the new broadcast or multicast links will not interfere with each other. We observe that every neighboring node N(A) could only receive a RBB or a RMB from Node A, because if a node other than A transmits another RBB or RMB to the same neighboring node, a collision occurs and thus Node A will not establish a new link. Hence the nodes in N(A) will not simultaneously receive RBB or RMB over other new links.

3) Finally, we show that the new broadcast or multicast links will not interfere with new unicast links. The sender of a new unicast link cannot simultaneously receive RBB or RMB of new broadcast or multicast links, because if a node receives a RBB or a RMB as an intended receiver, it will stop making a unicast reservation. If the sender of a new unicast link detects a collision in MS3, it will send an SBB or an SMB, resulting in interruption of the broadcast or multicast reservation. Similarly, a receiver of a new unicast link cannot simultaneously receive RBBs or RMBs of new broadcast or multicast links.

Theorem 2: *New unicast links will not interfere with existing links, or with other new links.*

Proof: 1) We first show that new unicast links will not interfere with existing links. Node A, as the sender of a new unicast link belonging to either NTRU1 or NTRU2, sends a RUB on the condition that no node in N(A) is receiving data on the intended DCH, which is then confirmed by listening on the DCH in MS1. Hence the new unicast link will not interfere with existing links. On the other hand, a new unicast link will not be interfered by existing links because the unicast reservation is successful only if Node B hears nothing on the intended DCH in MS4 (NTRU1 node) or MS5 (NTRU2 node). This means that only the signal from Node A can reach the intended receiver in the resulting transmission.

2) We then show that new unicast links will not interfere with other new links. Given that it is established above that there is no interference between the new unicast links and the new broadcast or multicast links, here we only need to prove that the new unicast links will not interfere with each other. The sender and the receiver of a new unicast link cannot simultaneously be a sender or receiver of another unicast link because if a node sends or receives a RUB in MS2, it will ignore other RUB in MS4. Then, the only possible interference between new unicast links is the one in which two unicast links with senders belonging to NTRU1 and NTRU2, respectively, use the same DCH. As shown in Fig. 4, if the same DCH is used to establish two new unicast links whose senders belong to NTRU1 and NTRU2, respectively, the two new links will interfere with each other if at least one receiver is the common neighbor of the two senders. Because the radio channels used by NTRU1 nodes and the radio channels used by NTRU2 nodes are different, the interference mentioned above between new unicast links will not occur. This completes the proof.

IV. PERFORMANCE EVALUATION

A. Theoretical Analysis

In this subsection, we evaluate the performance of MATS when only unicast traffic is present. To compare MATS with CATS [19], we use the same model of a symmetric hypercube network topology, in which each node has at most d neighbors and the neighbors of the same node are hidden from each other. Although this topology is not typically found in real ad hoc networks, it constitutes the worst-case scenario for hidden-terminal interference and provides useful insights into the performance of reservation-based protocols. In this model, we assume that each node can only buffer one message; it simply discards any message passed from the upper layer if there is an unsent message in the buffer. This means that a node can attempt to reserve a link in every slot, which makes our results the worst possible. We refer to all the data, which is divided into packets each of which is sent by one slot per frame, as a message flow (or flow), and Average Flow Length is referred to as AFL (number of slots). We assume that every node can only reserve no more than one channel (slot) in a frame, which means that the node with a reserved channel will not attempt to reserve another channel. Under this condition, we can find, for example, that node A attempts to make reservation to transmit data to node B, the worst case is that node B uses $d-1$ slots for receiving from d-1 neighbors except node A and uses 1 slot for transmitting. In the meantime, node A uses $d-1$ slots for receiving from d-1 neighbors except node B. Then the total number of slots occupied become $(d-1)+1+(d-1) = 2d-1$. If node A has $2d$ channel available, it can always find 1 slot to make reservation to transmit data to node B.

The frame length L in MATS for a node to be able to unicast once in every frame in the worst case scenario is set to $2d$ slots, assuming that there are at least $c = d$ data channels available.

We assume that the reserved transmission slot for a node is uniformly distributed among the slots in a frame and the data channel used for the transmission is uniformly distributed among the available data channels. Assuming the system is in steady state, let P_T be the steady state probability that a node has reserved a link for a transmission and let P_R be the probability that a node is receiving in a frame. Due to the symmetry of the network topology and the traffic model for the whole system, we obtain $P_T = P_R$. The idle periods consisting of idle frames (i.e., the frames in which a node has no link or attempted to make reservation but failed) alternate with the transmission periods because any successful data transmission must follow a successful reservation. If the probability P_W that a node successfully reserves a slot within a frame is known, we can obtain the average idle period and derive the P_T using AFL. Then, using a set of nonlinear equations, we can calculate the value of P_T. Assuming the reservation requests generated at each node form a Poisson source with a mean rate λ in a slot, the throughput increases up to a maximum as λ increases.

Consider the situation in which a node, say Node A, attempts to reserve a slot to send data to one of its neighbors, say Node B, on a particular radio channel.

In case of unicast, we have the following:

1. Node A belongs to NTRU1.
2. Node A has a unicast request.
3. None of Node A's neighbors other than Node B is sending data to Node A or receiving data on the intended DCH.
4. None of Node B's neighbors other than Node A does any of the following: (1) sending data to Node B, (2) sending data on the intended DCH, or (3) sending a RUB in MS2.
5. Node B neither sends data nor sends a RUB in MS2.

Let the probabilities of the above five conditions be $P_{11}, P_{12}, P_{13}, P_{14}$ and P_{15}, respectively. Then, we have

$$\begin{aligned} P_{11} &= P_{c1} = c_1/c \\ P_{12} &= 1 - \mathrm{e}^{-\lambda} \end{aligned} \tag{1}$$

Let P_{Anif} represent the probability that a node of Node A's neighbors is not sending data to Node A and, not receiving on the intended DCH, we have

$$\begin{aligned} &P_{Anif} = \\ &1 - (1 - P_T)P_R\frac{1}{Lc} - P_T\left[\frac{1}{Ld} + \left(1 - \frac{1}{L}\right)P_R\frac{1}{L-1}\frac{1}{c}\right] \end{aligned} \tag{2}$$

where, the term $(1 - P_T)P_R\frac{1}{Lc}$ is the probability of the case that a neighbor of Node A has no link for sending and has a link for receiving just in the intended slot and intended DCH. The term $P_T[\frac{1}{Ld} + (1 - \frac{1}{L})P_R\frac{1}{L-1}\frac{1}{c}]$ is the probability of the case that a neighbor of Node A is sending to Node A, or receiving over the intended DCH in intended slot.

Thus

$$\begin{aligned} P_{13} &= (P_{Anif})^{d-1} \\ P_{14} &= \{1 - (1 - P_T)(1 - P_R/L)P_{11}P_{12}P_{13} \\ &\qquad - P_T\frac{1}{L}[\frac{1}{d} + (1 - \frac{1}{d})\frac{1}{c}]\}^{d-1} \\ P_{15} &= P_T(1 - \frac{1}{L}) + (1 - P_T)(1 - P_{11}P_{12}) \end{aligned} \tag{3}$$

where, P_{13} is the probability that Node A sends RUB in MS2, P_{14} is the probability that Node B can receive RUB sent from Node A in MS2 without collision from other RUB and can accept the request. The term $P_T(1 - \frac{1}{L})$ in P_{15} is the probability that Node B is sending but not in the intended slot and the term $(1 - P_T)(1 - P_{11}P_{12})$ is the probability that Node B does not have link for sending and does not meet the condition to send its RUB.

Therefore, when Node A belongs to NTRU1, the probability that a reservation succeeds is $P_{W1}^S = P_{11}P_{12}P_{13}P_{14}P_{15}$.

When Node A belongs to NTRU2, we have:

1. Node A belongs to NTRU2.
2. Node A has a request in the previous slot.
3. None of Node A's neighbors other than Node B is sending data to Node A or receiving data on the intended DCH. Additionally, Node A has not received a RUB as an intended receiver in MS2 sent from a node of NTRU1.
4. None of Node B's neighbors other than Node A does any of the following: (1) sending data to Node B, (2) sending data on the intended DCH, or (3) sending a RUB in MS4. Additionally, Node B has not received a RUB as an intended receiver in MS2 sent from a node of NTRU1.
5. Node B neither sends data nor sends an RUB in MS2 or MS4.

Let the probabilities of the above five conditions be $P_{21}, P_{22}, P_{23}, P_{24}$ and P_{25}, respectively. Then, we have

$$\begin{aligned} P_{21} &= P_{c2} = c_2/c \\ P_{22} &= P_{12} = 1 - \mathrm{e}^{-\lambda} \end{aligned} \tag{4}$$

Let P_{Arub} represent the probability that one of Node A's neighbors sends a RUB in MS2. This probability can be expressed as $P_{Arub} = (1 - P_T)(1 - P_R\frac{1}{L})P_{11}P_{12}P_{13}$.

Thus,

$$P_{23} = P_{13} - (d-1)P_{Arub}\frac{1}{d} \times (P_{Anif} - P_{Arub})^{d-2} \tag{5}$$

where, the second term on the right side of the equation is the probability that Node A correctly receives a RUB in MS2 as the intended receiver. Because the condition that Node B does not send RUB is guaranteed by P_{25}, the number of RUB candidate senders becomes $d - 1$.

Let P_{Bnf} represent the probability that one of Node B's neighbors does none of the following: (1) sending a RUB in MS4, (2) sending data to Node B, and (3) sending on the intended DCH. This probability can be expressed as $P_{Bnf} = 1 - (1 - P_T)(1 - P_R\frac{1}{L})P_{21}P_{22}P_{23} - P_T\frac{1}{L}[\frac{1}{d} + (1 - \frac{1}{d})\frac{1}{c}]$.

Then

$$\begin{aligned} P_{24} &= (P_{Bnf})^{d-1} - (d-1)P_{Arub}\frac{1}{d} \\ &\qquad \times (P_{Bnf} - P_{Arub})^{d-2} \\ P_{25} &= P_T(1 - \frac{1}{L}) + (1 - P_T)(1 - P_{22}) \end{aligned} \tag{6}$$

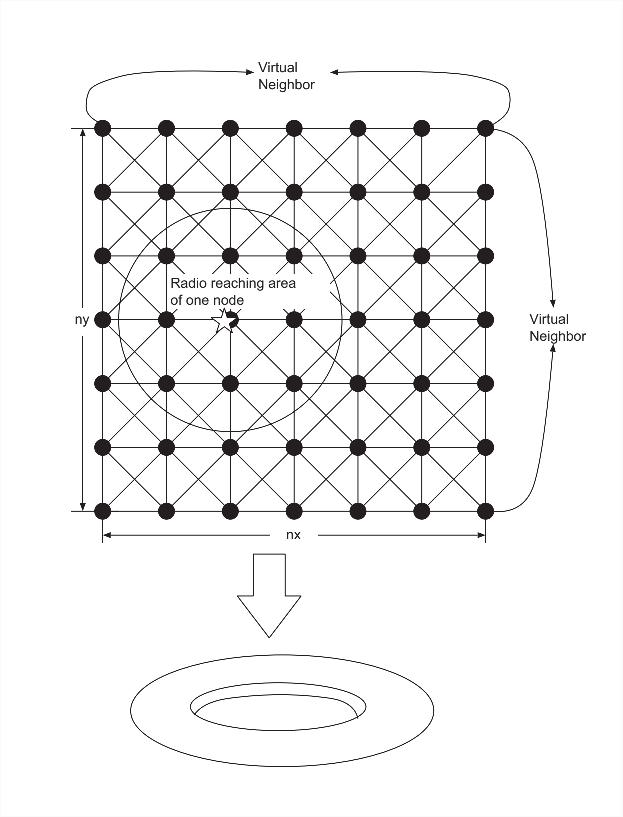


Fig. 5. Simulation Model.

where, the same as in Equation (5), the second term on the right side of the equation P_{24} is the probability that Node B correctly receives RUB sent from other candidate sender in MS2.

Therefore, when Node A belongs to NTRU2, the probability that a reservation succeeds is $P^S_{W2} = P_{21}P_{22}P_{23}P_{24}P_{25}$. The probability P^S_W that a node successfully reserves a slot within a time slot can be expressed as $P^S_{W1} + P^S_{W2}$; hence, we have

$$P_W \;\; = \;\; 1 - (1 - P^S_W)^L \tag{7}$$

The average number of idle frames I can be expressed as $1/P_W - 1$, and P_T can be related to the AFL by

$$P_T \;\; = \;\; \frac{AFL}{AFL + I} \tag{8}$$

Careful inspection of the above probability, we observe that P_T means the ratio of the number of frames in which a node sends data in a reserved slot. Here, we use P_T as throughput, which can be obtained according to the above formula by repeatedly substituting the new P_T for the old.

B. Simulation Setup

The above theoretical results are obtained for unicasting based on a model with specific topology and under certain assumptions. However, theoretical results under more realistic assumptions may not be possible. Moreover, if we take broadcasting and multicasting into account in addition to unicasting, it is very difficult, if not impossible, to obtain analytical results. Thus, we will carry out simulation study to demonstrate our protocol. Figure 5 shows the simulation model in which every node has eight neighbors and all nodes generate Poisson traffic with the same arrival rate. When a node generates a unicast request, it selects a neighbor randomly as its receiver. The nodes on the border have more neighbors than nodes far from the border in the area. If we do not consider this edge effect, the throughput for a node will depend on which node to select. To overcome this edge effect, we assume nodes on one side can hear from the nodes on the other side as shown in Fig. 5, so all nodes have the same number of neighbors and may transmit in the same topological situation. Nodes on the lower border shown in Fig. 5 are the neighbors of those on the above border, and nodes on the left border are the neighbors of those on the right border, so the topological model becomes a ring (the wrap-around model). To compare with the theoretical results, we carry out simulation in the same conditions used in the theoretical analysis except topology.

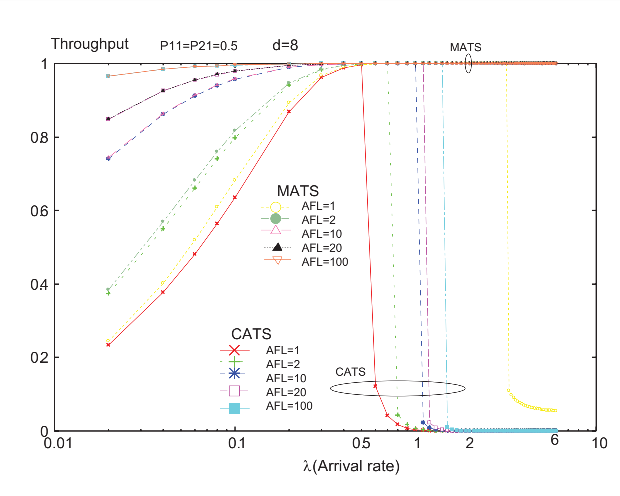


Fig. 6. Theoretical throughputs of CATS and MATS with $d = 8$.

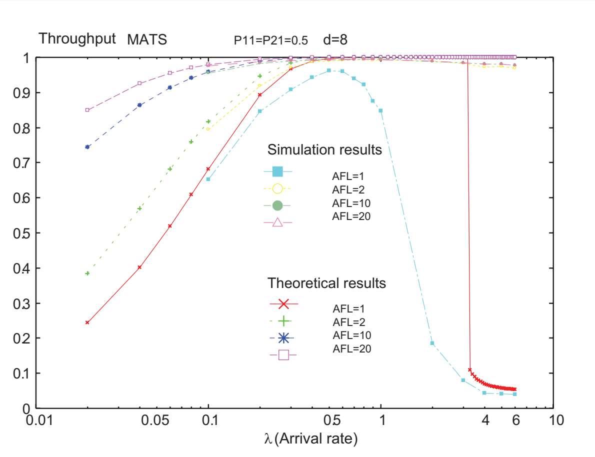


Fig. 7. Simulated and theoretical throughput results for MATS in case of unicasting.

C. Discussions

Figure 6 shows the theoretical results when only unicast traffic is present with $P_{11} = P_{21} = 0.5$. As shown in Fig. 6,

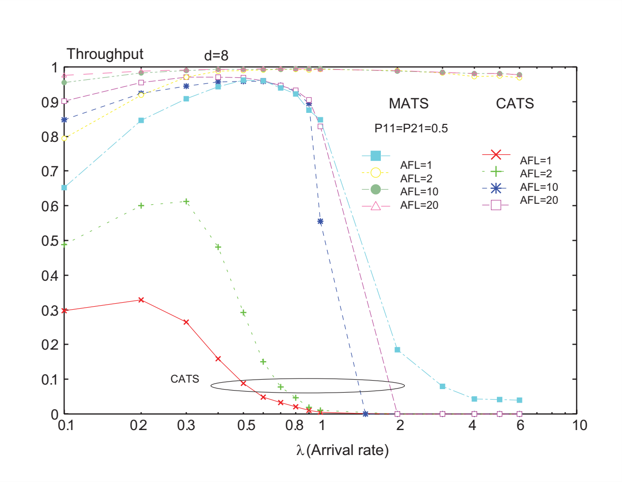


Fig. 8. Simulation results for CATS and MATS in case of unicasting.

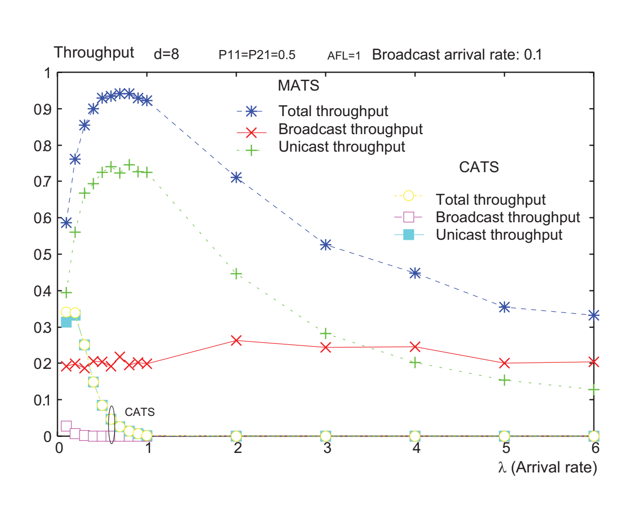


Fig. 9. Simulation results of CATS and MATS with broadcast arrival rate 0.1 and $AFL = 1$.

the throughputs of CATS for $d = 8$ increase with arrival rate (λ) up to a certain critical value of λ, at which they rapidly decrease; this critical arrival rate decreases with decreasing AFL. In comparison to CATS, MATS gives significantly better performance.

Figure 7 shows the simulation results and the theoretical results of MATS for comparison in case of unicasting, from which we find that the simulation and theoretical throughputs match well at lower arrival rate.

In case that the arrival rate is over 1, there is a little difference between the simulated and theoretical throughputs with AFL greater than 2, and the throughputs of AFL=1 have significant differences in the range of high arrival rate. We think that the differences results from the hyper-cube model used in the theoretical analysis, in which the neighbors of one node are hidden from each other. In contrast, in the simulation model, neighbors of one node interfere with each other in transmissions, and interference becomes obvious when the arrival rate increases, specially for the case with AFL=1. In general, the theoretical analysis is effective in case of low interference and low contention probability. For a more complicated scenario, we will use simulations for our performance evaluation.

Figure 8 shows the simulation results for CATS and MATS, respectively, when only unicast traffic is present with different AFLs. The throughputs of CATS increase to the peaks and then drop sharply down to zero. For MATS, we can find the problem of sudden drop-off is mitigated greatly and the throughputs are higher than those of CATS. Throughput of MATS with only AFL=1 has a drop-off at a comparatively higher arrival rate. In the cases that AFL is greater than 2, the throughputs of MATS have no drop-off with relatively small decreases in the range of high arrival rate.

Above simulation results are obtained when only unicasting is present. In the following, we present the throughput results when both of unicasting and broadcasting are present. Considering the fact that broadcast link does not just consist of a pair of a sender and a receiver but one sender and many receivers, and that the throughputs of broadcast and unicast are affected each other, we here use a different criterion to evaluate the throughput. The throughput here is expressed as the average number of links in one frame for a node. One broadcast transmission has 8 links if one node has 8 neighbors. In the simulation, same as in the analysis before, one node can reserve no more than one slot per frame either in case of unicast or broadcast transmission. To evaluate the broadcast transmissions, we present the results that show the throughput changes when the unicast load increases with a certain broadcast load, say, 0.1. In Figs. 9, 10, 11 and 12, we show the throughputs with AFL=1, AFL=2, AFL=10 and AFL=20, respectively. There are three pairs of curves express the total throughput, broadcast throughput and unicast throughput, each pair includes two curves for MATS and CATS, respectively. The total throughput is the sum of the throughputs for broadcast and unicast.

Figure 9 shows the throughputs with AFL=1, in which we find that the total throughput of MATS is considerably higher than that of CATS. For CATS, broadcast and unicast throughputs are low and drop down to zero when the unicast arrival rate increases. For MATS, broadcast and unicast throughputs are 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 one is sent, it will be replaced by a new one. So, the probability of broadcast reservation request will decrease when the unicast arrival rate increases. Since RBB for broadcast reservation is sent in MS3, which is not interfered with RUB that is sent in MS2 or MS4, moderate decrease in the number of nodes sending RBB will increase the successful rate of broadcast reservation. Certainly, excessive decrease also results in broadcast throughput drop.

Figure 10 shows the throughputs with AFL=2, in which the total throughput of MATS has no drop-off and broadcast throughput is considerably higher than that of CATS. The unicast throughput also has no drop-off that occurs in CATS, but in the range of low arrival rate, it is somehow the same as that of CATS. This is because that a node cannot receive from more than one node simultaneously, so if broadcast throughput increases, then the unicast throughput will decrease. With uni-

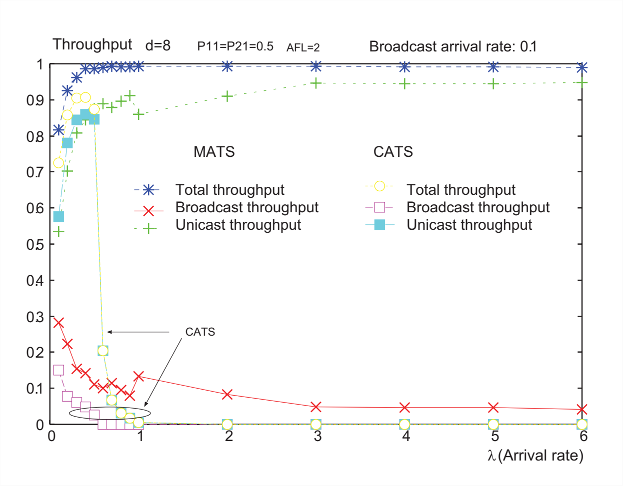


Fig. 10. Simulation results of CATS and MATS with broadcast arrival rate 0.1 and $AFL = 2$.

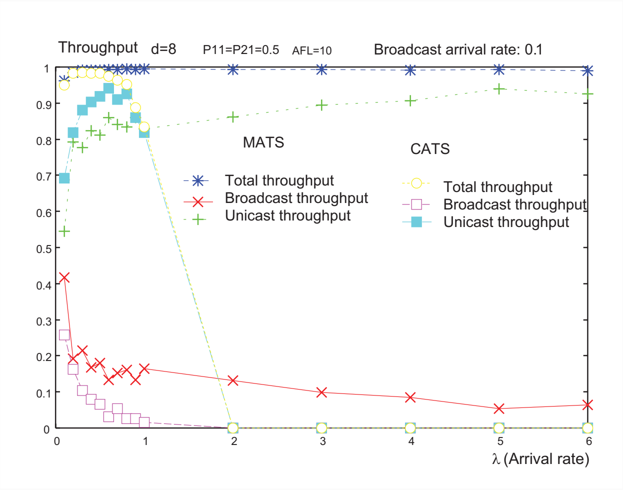


Fig. 11. Simulation results of CATS and MATS with broadcast arrival rate 0.1 and $AFL = 10$.

cast arrival rate continues to increase, the unicast throughput increases and broadcast throughput will decrease.

Figures 11 and 12 show throughputs with AFL=10 and AFL=20, respectively, in which we can observe the similar trends shown in Fig. 10. Both of broadcast and unicast throughputs of MATS have no drop-off and the broadcast throughput becomes slightly higher than that of AFL=2 because of a longer AFL. As shown in Fig. 11 and 12, as λ increases, the probability of a new successful reservation will rise, however, increase in the number of existing links will resist new reservations. Finally, the throughputs of MATS reach maximum and then tend to stay approximately constant. On the contrary, the throughputs of CATS reach zero because nodes failed with reservation in previous slot will send requests in the next slot with one group, which differs from MATS.

The same is demonstrated in Fig. 10, where broadcast and unicast throughputs of MATS are related to some degree. When contentions occur between nodes with broadcast and unicast, their behaviors depend on the priority policy. For example, in MATS, when a node receives a RBB, it will abort

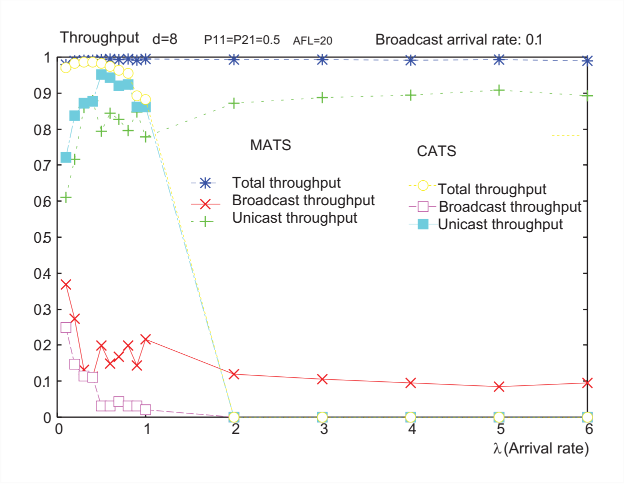


Fig. 12. Simulation results of CATS and MATS with broadcast arrival rate 0.1 and $AFL = 20$.

other unicast reservation process and if a node receives RUB in MS2 from NTRU1, it will not listen or send RUB in MS4. Certainly different priority policy can be adapted in MATS.

From the above results, we conclude that the throughput drop-off problem is significantly mitigated in our protocol MATS.

V. CONCLUSIONS

In this paper, we have proposed a new reservation protocol for distributed mobile ad hoc networks, referred to as MATS, that avoids sudden throughput drop-off problem in previously known protocols. The idea is to divide the nodes in a network into different groups in order to decrease the probability of collisions during channel reservation process. We have derived some approximate theoretical throughput of MATS under certain assumptions and carried out extensive simulations to evaluate this protocol. Comparing theoretical and simulation results, we observe that the analytical result is valid in case of comparatively low contention probability. Through extensive simulation results, we have shown that MATS has several unique characteristics: (1) MATS can achieve high throughput without sudden drop-off under heavy traffic loads; (2) the reservations of nodes are distributed and carried out in parallel with a short overhead; and (3) broadcast and multicast can be carried out separately from unicast reservations and can be assigned different priorities. This protocol provides an efficient approach to coordinating the use of multiple channels in mobile ad hoc networks.

REFERENCES

[1] J. J. Garcia-Luna-Aceves and E. L. Madruga, “The core-addidted mesh protocol,” *IEEE J. Select. Areas Commun.*, vol. 17, no. 8, Aug. 1999.
[2] M. Grossglauser and D. Tse, “Mobility increases the capacity of ad hoc wireless networks,” *IEEE/ACM Trans. Networking*, vol. 10, no. 4, pp. 477-486, Aug. 2002.
[3] L. Butty’an and J.-P. Hubaux, “Enforcing service availability in mobile ad-hoc WANs,” in *Proc. 1st IEEE/ACM Workshop on Mobile Ad Hoc Networking and Computing*, Aug. 2000.
[4] C. Zhu and M. S. Corson, “A five phase reservation protocol (FPRP) for a mobile ad hoc networks,” in *Proc. IEEE INFOCOM*, vol. 1, pp. 322-331, Mar. 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*, vol. 1, pp. 70-74, Oct. 1989.

[6] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," IEEE Computer Society LAN MAN Standards Committee, IEEE Standard 802.11-1997, 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. Bharghava, A. Demers, S. Shenkar, and L. Zhang, "MACAW: a media access protocol for wireless LANs," in *Proc. SIGCOMM Conference on Communications Architectures, Protocols and Applications*, pp. 212-225, Aug. 1994.

[9] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," in *Proc. ACM SIGCOMM*, vol. 27, no. 4, pp. 39-49, 1997.

[10] J.-H. Ju and V. O. K. Li, "TDMA scheduling design of multihop packet radio network based on Latin squares," *IEEE J. Select. Areas Commun.*, vol. 17, no. 8, pp. 1345-1352, Aug. 1999.

[11] J. F. Chang and T. H. Yang, "Multichannel ARQ protocols," *IEEE Trans. Commun.*, vol. 41, pp. 592-598, Apr. 1993.

[12] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Inform. Theory*, vol. 34, pp. 910-917, Sep. 1998.

[13] A. Ephremides and T. Truong, "Scheduling broadcasts in multihop radio networks," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 456-460, Apr. 1990.

[14] I. Cidon and M. Sidi, "Distributed assignment algorithms for multihop packet radio networks," *IEEE Trans. Comput.*, vol. 38, no. 10, pp. 1353-1361, Oct. 1989.

[15] I. Chlamtac and S. Kutten, "A spatial-reuse TDMA/FDMA for mobile multi-hop radio networks," in *Proc. IEEE INFOCOM*, pp. 389-394, Mar. 1985.

[16] I. Chlamtac and A. Lerner, "Link allocation in mobile radio networks with noisy channel," in *Proc.. IEEE INFOCOM*, Apr. 1986.

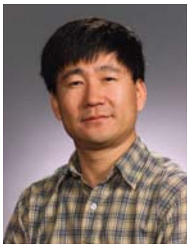
[17] J.-H. Ju and V. O. K. Li, "An optimal topology-transparent scheduling method in multihop pocket radio networks," *IEEE/ACM Trans. Networking*, vol. 6, pp. 298-306, June 1998.

[18] H. Wu, A. Utgikar, and N.-F. Tzeng, "SYN-MAC: A distributed medium access control protocol for synchronized wireless networks," *ACM Mobile Networks and Applications Journal*, vol. 10, no. 5, pp. 627-637, Oct. 2005.

[19] Z. Tang and J.J. Garcia-Luna-Aceves, "Collision-avoidance transmission scheduling for Ad-Hoc networks," in *Proc. IEEE ICC*, vol. 3, pp. 1788-1794, June 2000.

Xuejun Tian graduated from Hebei University, China in 1985. He received his MS degree from Department of Electrical and Mechanical Engineering, Tianjin Institute of Technology, China in 1991, and Ph.D. degree from Department of Intelligence and Computer Science, Nagoya Institute of Technology, Japan, in 1998, respectively. Since 1998, he has been an assistant professor in Department of Information Systems, Faculty of Information Science and Technology, Aichi Prefectural University, Japan. From Jul. 2003 to Jun. 2004, he was a Visiting Assistant Professor in Department of Electrical and Computer Engineering at University of Florida, Gainesville, FL.

Dr. Tian is a member of the IEICE (the Institute of Electronics, Information and Communication Engineers, Japan) and a member of the IEEJ (the Institute of Electrical Engineers of Japan). His research interests includes QoS, wireless networks, mobile communications and ubiquitous computing. His email address is *tan@ist.aichi-pu.ac.jp*.



Yuguang Fang (S'92-M'94-S'96-M'97-SM'99) received a Ph.D degree in Systems and Control Engineering from Case Western Reserve University in January 1994, and a Ph.D degree in Electrical Engineering from Boston University in May 1997.

From June 1997 to July 1998, he was a Visiting Assistant Professor in Department of Electrical Engineering at the University of Texas at Dallas. From July 1998 to May 2000, he was an Assistant Professor in the Department of Electrical and Computer Engineering at New Jersey Institute of Technology. In May 2000, he joined the Department of Electrical and Computer Engineering at University of Florida where he got the early promotion to Associate Professor with tenure in August 2003 and to Full Professor in August 2005. He has published over 180 papers in refereed professional journals and conferences. He received the National Science Foundation Faculty Early Career Award in 2001 and the Office of Naval Research Young Investigator Award in 2002.

He is currently serving as an Editor for many journals including IEEE Transactions on Communications, IEEE Transactions on Wireless Communications, IEEETransactions on Mobile Computing, and ACM Wireless Networks. He is also actively participating in conference organization such as the Program Vice-Chair for IEEE INFOCOM'2005, Program Co-Chair for the Global Internet and Next Generation Networks Symposium in IEEE Globecom'2004 and the Program Vice Chair for 2000 IEEE Wireless Communications and Networking Conference (WCNC'2000). His email address is *fang@ece.ufl.edu*.

Tetsuo Ideguchi received the B.S. degree in telecommunication engineering from the University of Electro- Communications in 1972, and the Ph.D. degree in telecommunication engineering from Tohoku University in 1993. He is a professor in the Faculty of Information Science and Technology , Aichi Prefectural University, Aichi, Japan and working on the research of network architecture, LAN, network management and mobile communications. He is a member of IEEE, and IEICE and IPSJ in Japan.