Title
Distributed Channel Access Scheduling for Ad Hoc Networks

Author
Garcia-Luna-Aceves, J.J.

Publication Date
2006

Distributed Channel Access Scheduling for Ad Hoc Networks

Lichun Bao and J.J. Garcia-Luna-Aceves
School of ENgineering
University of California, Santa Cruz
{baolc, jj}@soe.ucsc.edu

ABSTRACT

Using two-hop neighborhood information, we develop four approaches for time-division channel access scheduling in ad hoc networks with omni-directional antennas, which are derived from a novel approach to contention resolution that allows a group of contenders to elect deterministically one or multiple winners in a given contention context (e.g., a time slot). Except for the node activation multiple access (NAMA) which schedules communications through a single channel, the link activation multiple access (LAMA), pair-wise link activation multiple access (PAMA) and hybrid activation multiple access (HAMA) are all dependent on the physical layer that is capable of creating multiple channels using the code division multiplexing scheme. The throughput and delay characteristics of these protocols in randomly generated multihop wireless networks are studied by analyses and simulation, and their performances are compared against a well-known static scheduling algorithm based on complete topology information, and the ideal CSMA and CSMA/CA protocols.

Keywords—**Channel access scheduling, medium access control protocol, MAC, ad hoc network.**

I. Introduction

Channel access protocols for ad hoc networks can be non-deterministic or deterministic. The non-deterministic approach started with ALOHA and CSMA [10] and continued with several collision avoidance schemes, of which the IEEE 802.11(b) standard for wireless LANs [5] being the most popular example to date. However, as the network load increases, network throughput drastically degrades because the probability of collisions rises, preventing stations from acquiring the channel.

On the other hand, deterministic access schemes set up timetables for individual nodes or links, such that the transmissions from the nodes or over the links are conflict-free in the code, time, frequency or space divisions of the channel. The schedules for conflict-free channel access can be established based on the topology of the network, or it can be topology independent.

Topology-dependent channel access control algorithms can establish transmission schedules by either dynamically exchanging and resolving time slot requests [4] [17], or pre-arrange a time-table for each node based on the network topologies. Setting up a conflict-free channel access time-table is typically treated as a node- or link- coloring problem on graphs representing the network topologies. The problem of optimally scheduling access to a common channel is one of the classic NP-hard problems in graph theory (k-colorability on nodes or edges) [6] [7] [14]. Polynomial algorithms are known to achieve suboptimal solutions using randomized approaches or heuristics based on such graph attributes as the degree of the nodes.

A unified framework for TDMA/FDMA/CDMA channel assignments, called UxDMA algorithm, was described by Ramanathan [13]. UxDMA summarizes the patterns of many other channel access scheduling algorithms in a single framework. These algorithms are represented by UxDMA with different parameters. The parameters in UxDMA are the constraints put on the graph entities (nodes or links) such that entities related by the constraints are colored differently. Based on the global topology, UxDMA computes the node or edge coloring, which correspond to channel assignments to these nodes or links in the time, frequency or code domain.

A number of topology-transparent scheduling methods have been proposed [3] [9] [11] to provide conflict-free channel access that is independent of the radio connectivity around any given node. The basic idea of the topology-transparent scheduling approach is for a node to transmit in a number of time slots in each time frame. The times when node i transmits in a frame corresponds to a unique code such that, for any given neighbor k of i, node i has at least one transmission slot during which node k and none of k's own neighbors are transmitting. Therefore, within any given time frame, any neighbor of node i can receive at least one packet from node i conflict-free. An enhanced topology-transparent scheduling protocol, TSMA (Time Spread Multiple Access), was proposed by Krishnan and Sterbenz [11] to reliably transmit control messages with acknowledgments. However, TSMA performs worse than CSMA in terms of delay and throughput [11].

We propose a neighbor-aware contention resolution (NCR) algorithm. Using only the identifiers of the contenders and the current contention context number, NCR derives a randomized priority for each contender in a given contention context. Then, each contender locally determines its eligibility to access the resource in the contention context by comparing its priority with other contenders'. Because the scheduling is dynamic, depending on the contention contexts, a different schedule is established in each contention context. Equivalently, only two colors are needed in the graph coloring for two possible states at any moment — transmission or reception. The color for transmission is used to the maximal extent in each contention situation.

In ad hoc networks, the contention to the channel happens among neighbors within two hops from each node, and the contention context corresponds to the time slot in time-division multiple access scheme. Based on the NCR algorithm, four multiple access protocols are derived, which respectively schedule node activation (NAMA) suitable for broadcast communication, link activation (LAMA) and pair-wise link activation (PAMA) for unicast communication, and hybrid activation (HAMA) for both unicast and broadcast communications.

Section II presents the NCR algorithm and analyzes the packet delay encountered in a general queuing model under certain contention level. Section III describes the four scheduling protocols. Section IV derives the channel access probabilities of the four protocols in randomly generated ad hoc networks, and compares the throughput attributes of two of the protocols with those of ideal carrier sensing multiple access (CSMA) and carrier sensing multiple access with collision avoidance (CSMA/CA) schemes. Section V presents the results of simulations that provide further insights on the performance differences among the four scheduling protocols and the corresponding static scheduling approaches based on UxDMA.

II. Neighbor-Aware Contention Resolution

A. Specification

No limited to ad hoc network scenarios, the neighbor-aware contention resolution (NCR) envisions a special election problem for an entity to locally decide the leadership status of itself among a known set of contenders in any given contention context. We assume that the knowledge of the contenders for each entity is acquired by an appropriate means, depending on the specific applications. For example, in the ad hoc networks of our interest, the contenders of each node are the neighbors within two hops, which can be obtained by each node periodically broadcasting the identifiers of its one-hop neighbors [1]. Furthermore, NCR requires that each contention context be identifiable, such as the time slot number in networks based on a time-division multiple access scheme.

Thus, the election problem for neighbor-aware contention resolution is be formulated as :"Given a set of contenders, M_i, against entity i in contention context t, how should the precedence of entity i in the set $M_i \cup \{i\}$ be established, such that every other contender yields to entity i whenever entity i establishes itself as the leader for the shared resource?"

To decide the precedence of an entity without incurring communication overhead among the contenders, we assign the entity a priority that depends on the identifier of the entity and varies according to the known contention context so that the criterion for the leadership is deterministic and fair among the contenders. Eq. (1) provides a formula to derive the priority, denoted by $i.\texttt{prio}$, for entity i in contention context t.

$$i.\texttt{prio} = \texttt{Hash}(i \oplus t) \oplus i, \tag{1}$$

where the function $\texttt{Hash}(x)$ is a fast message digest generator that returns a random integer in range $[0, M]$ by hashing the input value x, and the sign '$\oplus$' is designated to carry out the concatenation operation on its two operands. Note that, while the `Hash` function can generate the same number on different inputs, each priority number is unique because the priority is appended with identifier of the entity.

```
NCR($i$, $t$)
{
    /* Initialize. */
    for ($k \in M_i \cup \{i\}$)
        $k$.prio = Hash($k \oplus t$) $\oplus k$;

    /* Resolve leadership. */
    if ($\forall k \in M_i,\ i$.prio $> k$.prio)
        $i$ is the leader;
} /* End of NCR. */
```

Figure 1. NCR Specification.

Figure 1 describes the NCR algorithm. Basically, NCR generates a *permutation* of the contending members, the order of which is decided by the priorities of all participants. Since the priority is a pseudo-random number generated from the contention context that changes from time to time, the permutation also becomes random such that each entity has certain probability, commensurate to its contention level,

$$q_i = \frac{1}{|M_i \cup \{i\}|} \tag{2}$$

being elected in each contention context.

Because it is assumed that contenders have mutual knowledge and t is synchronized, the order of contenders based on the priority numbers is consistent at every participant, thus avoiding any conflict among contenders.

B. Dynamic Resource Allocation

The description of NCR provided thus far evenly divides the shared resource among the contenders. In practice, the demands from different entities may vary, which requires appropriate allocation of the shared resource. There are several approaches for allocating variable portion of the resource according to individual demands. In any approach, an entity, say i, needs to specify its demand by an integer value chosen from a given integer set, denoted by p_i. Because the demands need to be propagated to the contenders before the contention resolution process, the integer set should be small and allow enough granularity to accommodate the demand variations while avoiding the excess control overhead caused by the demand fluctuations.

Suppose the integer set is from 0 to P, inclusive, the following three approaches provide resource allocation schemes, differing in the portion of the resource allocated on a given integer value. If the resource demand is 0, the entity has no access to the shared resource.

B.1 Pseudo identities

An entity assumes p pseudo identities, each defined by the concatenation of the entity identifier and a number from 1 to p. For instance, entity i with resource demand p_i is assigned with the following pseudo identities: $i \oplus 1$, $i \oplus 2$, $\cdots$, $i \oplus p_i$. Each identity works for the entity as a contender to the shared resource. Figure 2 specifies NCR with pseudo identities (NCR-PI) for resolving contentions among contenders with different resource demands.

The portion of the resource available to an entity i in NCR-PI

```
NCR-PI($i$, $t$)
{
    /* Initialize each entity $k$ with demand $p_k$. */
    for ($k \in M_i \cup \{i\}$ and $1 \le l \le p_k$)
        ($k \oplus l$).prio = Hash($k \oplus l \oplus t$) $\oplus k \oplus l$;

    /* Resolve leadership. */
    if ($\exists k, l : k \in M_i, 1 \le l \le p_k$ and
        $\forall m : 1 \le m \le p_i, (k \oplus l)$.prio $>$ ($i \oplus m$).prio)
        $i$ is not the leader;
    else
        $i$ is the leader;
} /* End of NCR-PI. */
```

Figure 2. NCR-PI Specification.

is proportional to its resource demand as follows:

$$q_i = \frac{p_i}{\sum_{k \in M_i \cup \{i\}} p_k}. \tag{3}$$

B.2 Root operation

Assuming enough computing power for floating point operations at each node, we can use the root operator to achieve the same proportional allocation of the resource among the contenders as in NCR-PI.

Given that the upper bound of function `Hash` in Eq. (1) is M, substituting line 2 in Figure 1 with the following formula generates a new algorithm which provides the same resource allocation characteristic as shown in Eq. (3).

$$k.\texttt{prio} = \left(\frac{\texttt{Hash}(k \oplus t)}{M}\right)^{\frac{1}{p_k}} . \tag{4}$$

B.3 Multiplication

Simpler operations, such as multiplication in the priority computation, can provide non-linear resource allocation according to the resource demands. Substituting line 2 in Figure 1, Eq. (5) offers another way of computing the priorities for entities.

$$k.\texttt{prio} = (\texttt{Hash}(k \oplus t) \cdot p_k) \oplus k . \tag{5}$$

According to Eq. (5), the priorities corresponding to different demands are mapped onto different ranges, and entities with smaller demand values are less competitive against those with larger demand values in the contentions, thus creating greater difference in resource allocations than the linear allocation schemes provided by Eq. (3) and Eq. (4). For example, among a group of entities, a, b and c, suppose $p_a = 1$, $p_b = 2$, $p_c = 3$ and $P = 3$. Then the resource allocations to a, b and c are statistically $\frac{1}{3} \cdot \frac{1}{3} = 0.11$, $\frac{1}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{2} = 0.28$, $\frac{1}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{1} = 0.61$, respectively.

For simplicity, the rest of this paper addresses NCR without dynamic resource allocation.

C. Performance

C.1 System delay

We assume NCR as an access mechanism to a shared resource at a server (an entity), and analyze the average delay experienced by each client in the system according to the M/G/1 queuing model, where clients arrive at the server according to a Poisson process with rate λ and are served according to the first-come-first-serve (FIFO) discipline. Specifically, we consider the time-division scheme in which the server computes the access schedules by the time-slot boundaries, and the contention context is the time slot. Therefore, the queuing system with NCR as the access mechanism is an M/G/1 queuing system with server vacations, where the server takes a fixed vacation of one time slot when there is no client in the queue at the beginning of each time slot.

The system delay of a client using NCR scheduling algorithm can be easily derive from the extended Pollaczek-Kinchin formula, which computes the service waiting time in an M/G/1 queuing system with server vacations [2]

$$W = \frac{\lambda \overline{X^2}}{2(1 - \lambda \overline{X})} + \frac{\overline{V^2}}{2\overline{V}},$$

where X is the service time, and V is the vacation period of the server.

According to the NCR algorithm, the service time X of a head-of-line client is a discrete random variable, governed by a geometric distribution with parameter q, where q is the probability of the server accessing the shared resource in a time slot, as given by Eq. (2). Therefore, the probability distribution function of service time X is

$$P\{X = k\} = (1 - q)^{k-1} q ,$$

where $k \geq 1$. Therefore, the mean and second moments of random variable X are:

$$\overline{X} = \frac{1}{q} , \quad \overline{X^2} = \frac{2 - q}{q^2} .$$

Because V is a fixed parameter, it is obvious that $\overline{V} = \overline{V^2} = 1$. Therefore, the average waiting period in the queue is:

$$W = \frac{\lambda(2 - q)}{2q(q - \lambda)} + \frac{1}{2} .$$

Adding the average service time to the queuing delay, we get the overall delay in the system:

$$T = W + \overline{X} = \frac{2 + q - 2\lambda}{2(q - \lambda)} . \tag{6}$$

The probabilities of the server winning a contention context are different, and so are the delays of clients going through the server. Figure 3 shows the relation between the arrival rate and the system delay of clients in the queuing system, given different resource access probabilities. To keep the queuing system in a steady state, it is necessary that $\lambda < q$ as implied by Eq. (6).

C.2 System throughput

Because of the collision freedom, NCR guarantees successful service to the clients. Therefore, the throughput of the server (the entity) over the shared resource is the minimum of the client arrival rate and the resource access probability. Considering all contenders for the shared resource, the overall system throughput is the summary of the throughput at individual entities. We

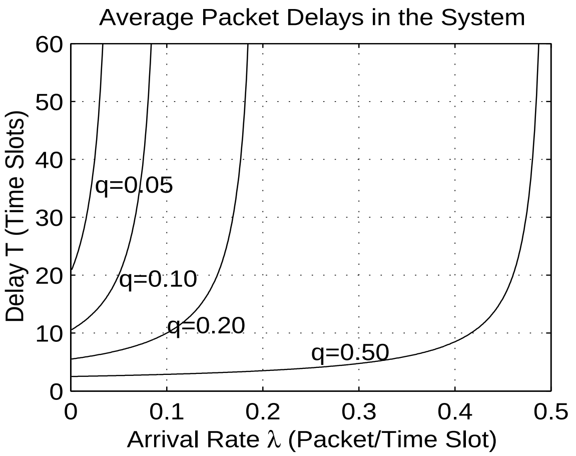

Figure 3. Average system delay of packets.

TABLE I
NOTATION

i.`prio`	The priority of node i.
(u,v).`prio`	The priority of link (u,v).
i.`code`	The code assigned to node i for either reception or transmission.
i.`state`	The activation state of node i for either reception or transmission.
Tx	Transmission state.
Rx	Reception state.
i.`in`	The transmitter to node i.
i.`out`	The receiver set of node i.
i.Q(i.`out`)	The packet queues for the eligible receivers in i.`out`.
N_i^c	The set of one-hop neighbors assigned with code c at node i.
[statement]	A more complex and yet easy-to-implement operation than an atomic statement, such as a function call.

have the following system throughput S combined from each and every entity k that competes for the shared resource:

$$S = \sum_k \min(\lambda_k, q_k) \tag{7}$$

where q_k is the probability that k may access the resource, and λ_k is the client arrival rate at k.

III. Channel Access Protocols

In this section, we apply the NCR algorithm to derive four channel access protocols in ad hoc networks with omni-directional antennas.

A. Modeling of Network and Contention

We assume that each node is assigned a unique identifier, and is mounted with an omni-directional radio transceiver that is capable of communicating using DSSS (direct sequence spread spectrum) on a pool of well-chosen spreading codes. The radio of each node only works in half-duplex mode, *i.e.*, either transmit or receive data packet at a time, but not both.

In multihop wireless networks, signal collisions may be avoided if the received radio signals are spread over different codes or scattered onto different frequency bands. Because the same codes on certain different frequency bands can be equivalently considered to be on different codes, we only consider channel access based on a code division multiple access scheme.

Time is synchronized at each node, and nodes access the channel based on slotted time boundaries. Each time slot is long enough to transmit a complete data packet, and is numbered relative to a consensus starting point. Although global time synchronization is desirable, only limited-scope synchronization is necessary for scheduling conflict-free channel access in multihop ad hoc networks, as long as the consecutive transmissions in any part of the network do not overlap across time slot boundaries. Time synchronization has to depend on physical layer timing and labeling for accuracy, and is outside the scope of this paper.

The topology of a packet radio network is represented by a graph $G = (V, E)$, where V is the set of network nodes, and E is the set of links between nodes. The existence of a link $(u, v) \in E$ implies that $(v, u) \in E$, and that node u and v are within the transmission range of each other, so that they can exchange packets via the wireless channel. In this case, node u and v are called *one-hop neighbors* to each other. The set of one-hop neighbors of a node i is denoted as N_i^1. Two nodes are called *two-hop neighbors* to each other if they are not adjacent, but have at least one common one-hop neighbor. The neighbor information of node i refers to the union of the one-hop neighbors of node i itself and the one-hop neighbors of i's one-hop neighbors, which equals

$$N_i^1 \cup (\bigcup_{j \in N_i^1} N_j^1) \; .$$

In multihop wireless networks, a single radio channel is spatially reused at different parts of the network. Hidden-terminal problem is the main cause of interference and collision in ad hoc networks, and involves nodes within at most two hops. To ensure conflict-free transmissions, it is sufficient for nodes within *two hops* to not transmit on the same time, code and frequency coordinates. Therefore, the topology information within two hops provides the contender information required by the NCR algorithm. When describing the operation of the channel access protocols, we assume that each node already knows its neighbor information within two hops. Bao and Garcia-Luna-Aceves described a neighbor protocol for acquiring this information in mobile ad hoc networks [1].

B. Code Assignment

We assume that the physical layer is capable of direct sequence spread spectrum (DSSS) transmission technique. In DSSS, the code assignments are categorized into transmitter-oriented, receiver-oriented or a per-link-oriented schemes, which are also referred to as TOCA, ROCA and POCA, respectively (*e.g.*, [8] [12]). The four channel access protocols described in this paper adopt different code assignment schemes, thus providing different features.

We assume that a pool of well-chosen orthogonal pseudo-noise codes, $C_{pn} = \{c_k \mid k = 0,\ 1, \cdots\}$, is available in the signal spreading function. The spreading code assigned to node i is denoted by $i.\texttt{code}$. During each time slot t, a new spreading code is assigned to node i derived from the priority of node i, using Eq. (8).

$$i.\texttt{code} = c_k,\ k = i.\texttt{prio}\ \textbf{mod}\ |C_{pn}|\ . \qquad (8)$$

Table I summarizes the notation used in the the paper to describe the channel access protocols.

C. NAMA

The node-activation multiple access (NAMA) protocol requires that the transmission from a node is received by the one-hop neighbors of the node without collisions. That is, when a node is activated for channel access, the neighbors within two hops of the node should not transmit. Therefore, the contender set M_i of node i is the one-hop and two-hop neighbors of node i, which is $N_i^1 \cup (\bigcup_{j \in N_i^1} N_j^1) - \{i\}$.

```
NAMA($i$, $t$)
{
    /* Initialize. */
    $M_i = N_i^1 \cup \left(\bigcup_{j \in N_i^1} N_j^1\right) - \{i\}$;
    for ($k \in M_i \cup \{i\}$))
        $k$.prio = Hash($k \oplus t$) $\oplus k$;

    /* Resolve nodal state. */
    if ($\forall k \in M_i,\ i$.prio $> k$.prio) {
        $i$.state = Tx;
        $i$.out = $N_i^1$;
        [ Transmit the earliest packet in $i$.Q ($i$.out) ];
    }
    else {
        $i$.state = Rx;
        [ Listen to the channel ];
    }
} /* End of NAMA. */
```

Figure 4. NAMA Specification.

Figure 4 specifies NAMA. Because only node i is able to transmit within its two-hop neighborhood when node i is activated, data transmissions from node i can be successfully received by all of its one-hop neighbor. Therefore, NAMA is capable of collision-free broadcast, and does not necessarily require code-division channelization for data transmissions.

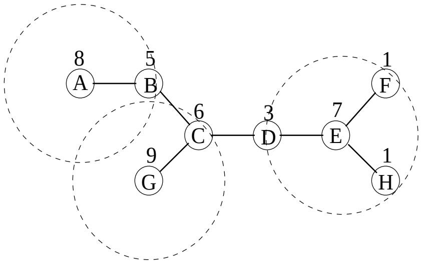

Figure 5. An example of NAMA operation.

Figure 5 provides an example of how NAMA operates in a multihop network. In the figure, the lines between nodes indicate the one-hop relationship, the dotted circles indicate the effective transmission ranges from nodes, and the node priorities in the current time slot are given beside each node. According to NAMA, there are three nodes A, G and E able to transmit because their priorities are the highest in their respective two-hop neighborhood.

D. LAMA

In LAMA (Link Activation Multiple Access), the code assignment for data transmission is receiver-oriented, which is suitable for unicasting using a link-activation scheme. The purpose of LAMA is to determine which node is eligible to transmit, and find out which outgoing link from the node can be activated in the current time slot.

```
LAMA($i$, $t$)
{
        /* Initialize. */
 1      for ($k \in N_i^1 \cup \left(\bigcup_{j \in N_i^1} N_j^1\right)$)
 2          $k$.prio = Hash($k \oplus t$) $\oplus k$;
 3          $n = k$.prio mod $|C_{pn}|$;
 4          $k$.code = $c_n$;
 5      }

        /* Resolve nodal state. */
 6      if ($\forall k \in N_i^1,\ i$.prio $> k$.prio) {
 7          $i$.state = Tx;
 8          $i$.out = $\emptyset$;
 9          for ($c$ : $\exists k \in N_i^1,\ c \equiv k$.code) {
10              $M_i = N_i^1 \cup \left(\bigcup_{j \in N_i^c} N_j^1\right) - \{i\}$;
11              if ($\forall j \in M_i,\ i$.prio $> j$.prio)
12                  $i$.out = $i$.out $\cup N_i^c$;
13          }

14          if ($\exists k$ :  $k \in i$.out and
                [ $k$ has the earliest packet in $i$.Q ($i$.out) ])
15              [ Transmit the packet in $i$.Q ($\{k\}$) on $k$.code ];
16      }
17      else {
18          $i$.state = Rx;
19          [ Listen to transmissions on $i$.code ];
20      }
} /* End of LAMA. */
```

Figure 6. LAMA Specification.

Figure 6 specifies LAMA for activating a link from node i in time slot t. Node i first initializes the priorities and code assignments of nodes within two hops (lines 1-5), and determines its eligibility to transmit (line 6). If eligible, node i examines each reception code c assigned to its one-hop neighbors, and decides whether node i can activate links to the one-hop neighbor subset N_i^c, in which all nodes are assigned code c (lines 9-12). Here, the set of contenders to node i is N_i^c and one-hop neighbors of nodes in N_i^c, excluding node i (line 10). Then node i selects and transmits the earliest packet to one of the receivers in $i.\texttt{out}$ (lines 14-15 according to FIFO). If node i is not able to transmit, it listens on the code assigned to itself (lines 17-20).

Figure 7 illustrates a contention situation at node i in a time slot. The topology is represented by an undirected graph. The number beside each node represents the current priority of the node. Node j and k happen to have the same code x. To determine if node i can activate links on code x, we compare priorities of nodes according to LAMA. Node i has the highest priority within one-hop neighbors, and higher priority than node j and k as well as their one-hop neighbors. Therefore, node i can activate either (i, j) or (i, k) in the current time slot t depending

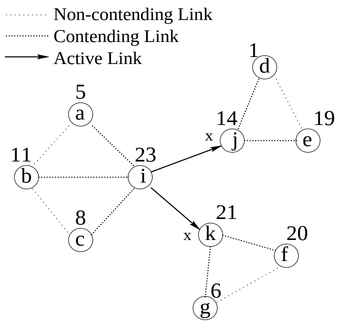

Figure 7. An example of LAMA operation.

on the back-logged data flows at node i. In addition, node e may activate link (e, d) if node d is assigned a code other than code x.

E. PAMA

PAMA (Pairwise-link Activation Multiple Access) is different from NAMA and LAMA in that the link priorities are used in the contention resolution for channel access, instead of the node priorities. The priority of link (u, v) is computed according to Eq. (9), which is an adaptation of Eq. (1).

$$(u, v).\texttt{prio} = \texttt{Hash}(u \oplus v \oplus t) \oplus u \oplus v, \tag{9}$$

Based on the priorities of the incident links to a node, PAMA chooses the link with the highest priority for reception or transmission at the node. Hence, the set of contenders of a link includes all other links incident to the endpoints of the link.

PAMA uses POCA code assignment scheme, in which a code is assigned per transmitter-receiver pair. However, because a node can activate only one incident link for either transmission or reception in each time slot, the POCA is equivalent to the transmitter-oriented (TOCA) scheme.

Figure 8 specifies PAMA. Lines 1-5 assign codes to the nodes in the two-hop neighborhood of node i. Then the priorities of the incident links at node i and its one-hop neighbors are computed (lines 7-10). The link with the highest priority at each node is marked for active incoming link (lines 13-16) or active outgoing link (lines 17-20). If node i has an active outgoing link, which is also an active incoming link at the receiver (line 21), node i further examines the hidden terminal problem at other nodes (lines 23-26). If node i can still transmit, it selects the packet for the active outgoing link and transmits on i.`code` (lines 28-29). Otherwise, node i listens on the code assigned to the active incoming link (lines 31-34).

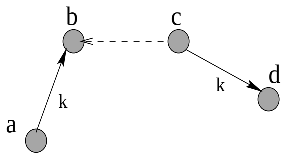

Figure 9. An example of hidden terminal problem in PAMA.

Figure 9 illustrates a simple example network, in which a collision happens at node b when link (a, b) and (c, d) are activated using the same code k. However, node c is able to know the possible collision and deactivate link (c, d) for the current time slot using PAMA lines 23-26 in Figure 8.

F. HAMA

Unlike previous channel access scheduling protocols that activate either nodes or links only, HAMA (hybrid activation multiple access) is a node-activation channel access protocol that is capable of broadcast transmissions, while also maximizing the chance of link activations for unicast transmissions. The code assignment in HAMA is the TOCA scheme.

In each time slot, a node derives its state by comparing its own priority with the priorities of its neighbors. We require that only nodes with higher priorities transmit to those with lower priorities. Accordingly, HAMA defines the following node states:

R Receiver: The node has an intermediate priority among its one-hop neighbors.

D Drain: The node has the lowest priority among its one-hop neighbors, and can only receive a packet in the time slot.

BT Broadcast Transmitter: the node has the highest priority within its two-hop neighborhood, and can broadcast to its one-hop neighbors.

UT Unicast Transmitter: the node has the highest priority among its one-hop neighbors, instead of two-hop. Therefore, the node can only transmit to a selected subset of its one-hop neighbors.

DT Drain Transmitter: the node has the highest priority among the one-hop neighbors of a *Drain* neighbor.

Y Yield: The node could have been in either UT- or DT-state, but chooses to abandon channel access because its transmission may incur unwanted collisions due to potential hidden sources from its two-hop neighbors.

Figure 10 specifies HAMA. Lines 1-8 compute the priorities and code assignments of the nodes within the two-hop neighborhood of node i using Eq. (1) and Eq. (8), respectively. Depending on the one-hop neighbor information of node i and node $j \in N_i^1$, node i classifies the status of node j and itself into receiver (R or D) or transmitter (UT) state (lines 9-14).

If node i happens to be a unicast transmitter (UT), then i further checks whether it can broadcast by comparing its priority with those of its two-hop neighbors (lines 15-17). If node i is a *Receiver* (R), it checks whether it has a neighbor j in *Drain* state (D) to which it can transmit, instead (lines 18-21). If yes, before node i becomes the *drain* transmitter (DT), it needs to make sure that it is not receiving from any one-hop neighbor (lines 22-25).

After that, node i decides its receiver set if it is in transmitter state (BT, UT or DT), or its sources if in receiver state (R or D). A receiver i always listens to its one-hop neighbor with the highest priority by tuning its reception code into that neighbor's transmission code (lines 26-42).

If a transmitter i unicasts (UT or DT), the hidden terminal problem should be avoided, in which case node i's one-hop receiver may be receiving from two transmitters on the same code (lines 43-45).

Finally, node i in transmission state may send the earliest arrived packet (FIFO) to its receiver set i.`out`, or listens if it is a receiver (lines 46-58). In case of the broadcast state (BT), i may choose to send a unicast packet if broadcast buffer is empty.

Figure 11 provides an example of how HAMA operates in a

```
PAMA($i$, $t$)
{
    /* Initialize. */
    for ($k \in N_i^1 \cup (\bigcup_{j \in N_i^1} N_j^1)$) {
        $k$.prio = Hash($k \oplus t$) $\oplus k$;
        $n$ = $k$.prio mod $|C_{pn}|$;
        $k$.code = $c_n$;
    }

    for ($k \in N_i^1 \cup \{i\}$) {
        /* Link priorities. */
        for ($j \in N_k^1$) {
            $(k, j)$.prio = Hash($k \oplus j \oplus t$) $\oplus k \oplus j$;
            $(j, k)$.prio = Hash($j \oplus k \oplus t$) $\oplus j \oplus k$;
        }

        $k$.in = -1;
        $k$.out = $\emptyset$;
        /* Active incoming or outgoing link. */
        if ($\exists j \in N_k^1, \forall u \in N_k^1$,
            $((j, k)$.prio $> (u, k)$.prio $\mid u \neq j)$ and
            $(j, k)$.prio $> (k, u)$.prio)
            $k$.in = $j$;
        else if ($\exists j \in N_k^1, \forall u \in N_k^1$,
            $((k, j)$.prio $> (k, u)$.prio $\mid u \neq j)$ and
            $(k, j)$.prio $> (u, k)$.prio)
            $k$.out = $\{j\}$;

    /* Nodal states. */
    if ($i$.out $\equiv \{k\}$ and $k$.in $\equiv i$) {
        $i$.state = Tx;
        /* Hidden terminal avoidance. */
        if ($\exists u \in N_i^1 - \{k\}$, $u$.in $\equiv v$, $v \neq i$ and
            $i$.code $\equiv v$.code and
            (($v \in N_i^1$ and $u \in v$.out) or ($v \notin N_i^1$)))
            $i$.out = $\emptyset$;

        if ([ There is a packet in $i$.Q ($i$.out) ])
            [ Transmit the packet on $i$.code ];
    }
    else if ($i$.in $\equiv k$) {
        $i$.state = Rx;
        [ Listen to transmissions on $k$.code ];
    }
} /* End of PAMA. */
```

Figure 8. PAMA Specification.

```
HAMA($i$, $t$)
{
    /* Every node is initialized in Receiver state. */
    $i$.state = R;
    $i$.in = -1;
    $i$.out = $\emptyset$;

    /* Priority and code assignments. */
    for ($k \in N_i^1 \cup (\bigcup_{j \in N_i^1} N_j^1)$) {
        $k$.prio = Hash($t \oplus k$);
        $n$ = $k$.prio mod $|C_{pn}|$;
        $k$.code = $c_n$;
    }

    /* Find UT and Drain. */
    for ($\forall j \in N_i^1 \cup \{i\}$) {
        if ($\forall k \in N_j^1$, $j$.prio $> k$.prio)
            $j$.state = UT; /* May unicast. */
        elseif ($\forall k \in N_j^1$, $j$.prio $< k$.prio)
            $j$.state = D; /* A Drain. */
    }

    /* If $i$ is UT, see further if $i$ can become BT */
    if ($i$.state $\equiv$ UT and
        $\forall k \in \bigcup_{j \in N_i^1} N_j^1$, $k \neq i$, $i$.prio $> k$.prio)
        $i$.state = BT;

    /* If $i$ is Receiver, $i$ may become DT. */
    if ($i$.state $\equiv$ R and
        $\exists j \in N_i^1$, $j$.state $\equiv$ D and
        $\forall k \in N_j^1$, $k \neq i$, $i$.prio $> k$.prio) {
        $i$.state = DT;

        /* Check if $i$ should listen instead. */
        if ($\exists j \in N_i^1$, $j$.state $\equiv$ UT and
            $\forall k \in N_i^1$, $k \neq j$, $j$.prio $> k$.prio)
            $i$.state = R; /* $i$ has a UT neighbor $j$. */
    }

    /* Find dests for Txs, and srcs for Rxs. */
    switch ($i$.state) {
        case BT:
                $i$.out = {-1}; /* Broadcast. */
        case UT:
            for ($j \in N_i^1$)
                if ($\forall k \in N_j^1$, $k \neq i$, $i$.prio $> k$.prio)
                    $i$.out = $i$.out $\cup \{j\}$;
        case DT:
            for ($j \in N_i^1$)
                if ($j$.state $\equiv$ D and $\forall k \in N_j^1$, $k \neq i$, $i$.prio $> k$.prio)
                    $i$.out = $i$.out $\cup \{j\}$;
        case D, R:
            if ($\exists j \in N_i^1$ and $\forall k \in N_i^1$, $k \neq j$, $j$.prio $> k$.prio) {
                $i$.in = $j$;
                $i$.code = $j$.code;
            }
    }

    /* Hidden Terminal Avoidance. */
    if ($i$.state $\in$ { UT, DT } and $\exists j \in N_i^1$, $j$.state $\neq$ UT and
        $\exists k \in N_j^1$, $k$.prio $> i$.prio and $k$.code $\equiv i$.code)
        $i$.state = Y;

    /* Ready to communicate. */
    switch ($i$.state) { /* FIFO */
        case BT:
            if ($i$.Q($i$.out) $\neq \emptyset$)
                pkt = The earliest packet in $i$.Q($i$.out);
            else
                pkt = The earliest packet in $i$.Q($N_i^1$);
            Transmit pkt on $i$.code;
        case UT, DT:
            pkt = The earliest packet in $i$.Q($i$.out);
            Transmit pkt on $i$.code;
        case D, R:
            Receive pkt on $i$.code;
    }
} /* End of HAMA. */
```

Figure 10. HAMA Specification.

multihop network during a time slot. In the figure, the priorities are noted beside each node. Node A has the highest priority among its two-hop neighbors, and becomes a broadcast transmitter (BT). Nodes F, G and H are receivers in the *drain* state, because they have the lowest priorities among their one-hop neighbors. Nodes C and E become transmitters to *drain*s, because they have the highest priorities around their respective *drain*s. Nodes B and D stay in *receiver* state because of their

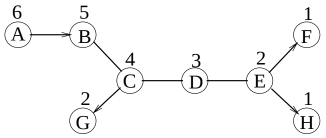

Figure 11. An example of HAMA operation.

low priorities. Notice that in this example, only node A would be activated in NAMA, because node C would defer to node A, and node E would defer to node C. This illustrates that HAMA can provide better channel access opportunities over NAMA, although NAMA does not requires code-division channelization.

In contrast to NAMA, HAMA provides similar broadcasting capability, in addition to the extra opportunities for sending unicast traffic with only a little more processing required on the neighbor information.

IV. Throughput Analyses

In a fully connected network, it comes natural that the channel bandwidth is evenly shared among all nodes using any of the above channel access protocols, because the priorities of nodes or links are uniformly distributed. However, in an ad hoc network model where nodes are randomly placed over an infinite plane, bandwidth allocation to a node is more generic, and much more complex. We first analyze the accurate channel access probabilities of HAMA and NAMA, then the upper bound of the channel access probability of PAMA and LAMA in this model. Using the results in [16] and [15], the throughput of NAMA and HAMA is compared with that of ideal CSMA and CSMA/CA.

For simplicity, we assumed that infinitely many codes are available such that hidden terminal collision on the same code was not considered.

A. Geometric Modeling

Similar to the network modeling in [16] and [15], the network topology is generated by randomly placing many nodes on an infinitely large two-dimensional area independently and uniformly, where the node density is denoted by ρ. The probability of having k nodes in an area of size S follows a Poisson distribution:

$$p(k, S) = \frac{(\rho S)^k}{k!} e^{-\rho S} \; .$$

The mean of the number of nodes in the area of size S is ρS.

Based on this modeling, the channel access contention of each node, is related with node density ρ and node transmission range r. Let N_1 be the average number of one-hop neighbors covered by the circular area under the radio transmission range of a node, we have $N_1 = \rho \pi r^2$.

Let N_2 be the average number of neighbors within two hops. As shown in Figure 12, two nodes become two-hop neighbors only if there is at least one common neighbor in the shaded area. The average number of nodes in the shaded area is:

$$B(t) = 2\rho r^2 a(t) \; ,$$

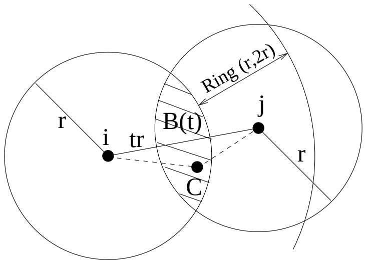

Figure 12. Becoming two-hop neighbors.

where

$$a(t) = \arccos \frac{t}{2} - \frac{t}{2}\sqrt{1 - \left(\frac{t}{2}\right)^2} \; . \qquad (10)$$

Thus, the probability of having at least one node in the shaded area is $1 - e^{-B(t)}$. Adding up all nodes covered by the ring $(r,\ 2r)$ around the node, multiplied by the corresponding probability of becoming two-hop neighbors, the average number of two-hop neighbors of a node is:

$$n_2 = \rho \pi r^2 \int_1^2 2t \left(1 - e^{-B(t)}\right) dt \; .$$

Because the number of one-hop neighbors is $N_1 = \rho \pi r^2$, adding the average number of one-hop and two-hop neighbors, we obtain the number of neighbors within two hops as:

$$N_2 = N_1 + n_2 = N_1 \left(1 + \int_1^2 2t \left(1 - e^{-B(t)}\right) dt\right) \; .$$

For convenience, symbol $T(N)$, $U(N)$ and $W(N)$ are introduced to denote three probabilities when the average number of contenders is N.

$T(N)$ denotes the probability of a node winning among its contenders. Because the number of contenders follows Poisson distribution with mean N, and that all nodes have equal chances of winning, the probability $T(N)$ is the average over all possible numbers of the contenders using Eq. (2):

$$T(N) = \sum_{k=1}^{\infty} \frac{1}{k+1} \frac{N^k}{k!} e^{-N} = \frac{e^N - 1 - N}{N e^N} \; .$$

Note that k starts from 1 in the expression for $T(N)$, because a node with no contenders does not win at all.

$U(N)$ is the probability that a node has at least one contender, which is simply

$$U(N) = 1 - e^{-N}.$$

$W(N)$ is introduced to denote

$$W(N) = U(N) - T(N) = 1 - \frac{1}{N}(1 - e^{-N}) \; .$$

B. NAMA

Because N_2 denotes the average number of two-hop neighbors, which is the number of contenders for each node in NAMA, it follows that the probability that the node broadcasts is $T(N_2)$. Therefore, the channel access probability of a node in NAMA is

$$q_{NAMA} = T(N_2) \; . \qquad (11)$$

C. HAMA

HAMA includes the node activation cases in NAMA in the broadcast state (BT). In addition, HAMA provides two more states for a node to transmit in the unicast mode (UT and DT). Overall, if node i transmits in the unicast state (UT and DT), node i must have at least one neighbor j, of which the probability is

$$p_u = U(N_1)\ .$$

In addition, the chances of unicast transmissions in either the UT or the DT states depend on three factors: (a) the number of one-hop neighbors of the source, (b) the number of one-hop neighbors of the destination, and (c) the distance between the source and destination.

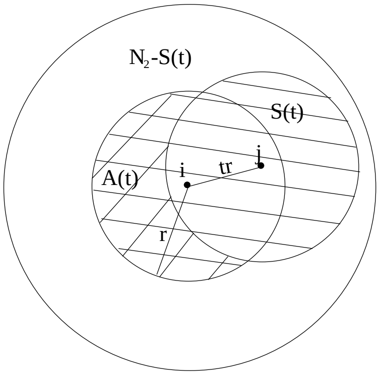

Figure 13. The unicast between two nodes.

First, we consider the probability of unicast transmissions from node i to node j in the UT state, in which case, node i contend with nodes residing in the combined one-hop coverage of nodes i and j, as illustrated in Figure 13. Given that the transmission range is r and the distance between nodes i and j is tr ($0 < t < 1$), we denote the number of nodes within the combined coverage by k_1 excluding nodes i and j, of which the average is

$$S(t) = 2\rho r^2 \left[\pi - a(t)\right]\ .$$

$a(t)$ is defined in Eq. (10). Therefore, the probability of node i winning in the combined one-hop coverage is:

$$p_1 = \sum_{k_1=0}^{\infty} \frac{1}{k_1+2} \frac{S(t)^{k_1}}{k_1!} e^{-S(t)} = \frac{W(S(t))}{S(t)}\ .$$

Furthermore, because node i cannot broadcast when it enters the UT state, there has to be at least one two-hop neighbor with higher priority than node i outside the combined one-hop coverage in Figure 13. Denote the number of nodes outside the coverage by k_2, of which the average is $N_2 - S(t)$. The probability of node i losing outside the combined coverage is thus:

$$p_2 = \sum_{k_2=1}^{\infty} \frac{[N_2 - S(t)]^{k_2}}{k_2!} e^{-(N_2 - S(t))} \frac{k_2}{k_2+1} = W(N_2 - S(t))\ .$$

In all, the probability of node i transmitting in the UT state is:

$$p_3 = p_1 \cdot p_2 = \frac{W(N_2 - S(t))\, W(S(t))}{S(t)}\ .$$

The probability density function (PDF) of node j at position t is $p(t) = 2t$. Therefore, integrating p_3 on t over the range $(0,1)$ with PDF $p(t) = 2t$ gives the average probability of node i becoming a transmitter in the UT state:

$$p_{UT} = \int_0^1 p_3 2t dt = \int_0^1 2t\, \frac{W(N_2 - S(t))\, W(S(t))}{S(t)}\, dt\ .$$

Second, we consider the probability of unicast transmissions from node i to node j in the DT state. We denote the number of one-hop neighbors of node j by k_3, excluding nodes i and j, of which the average is N_1. Then, node j requires the lowest priority among its k_3 neighbors to be a *drain*, and node i requires the highest priority to transmit to node j, of which the average probability over all possible values of k_3 is:

$$p_4 = \sum_{k_3=0}^{\infty} \frac{N_1^{k_3}}{k_3!} e^{-N_1} \frac{1}{k_3+2} \frac{1}{k_3+1} = \frac{T(N_1)}{N_1}\ .$$

In addition, node i has to lose to nodes residing in the side lobe, marked by $A(t)$ in Figure 13. Otherwise, node i would enter the UT state. Denote the number of nodes in the side lobe by k_4, of which the average is

$$A(t) = 2\rho r^2 \left[\frac{\pi}{2} - a(t)\right]\ .$$

The probability of node i losing in the side lobe is thus

$$p_5 = \sum_{k_4=1}^{\infty} \frac{A(t)^{k_4}}{k_4!} e^{-A(t)} \frac{k_4}{k_4+1} = W(A(t))\ .$$

In all, the probability of node i entering the DT state for transmission to node j is the product of p_4 and p_5:

$$p_6 = p_4 \cdot p_5 = \frac{T(N_1)}{N_1}\, W(A(t))\ .$$

Using the PDF $p(t) = 2t$ for node j at position t, the integration of the above result over range $(0,1)$ gives the average probability of node i entering the DT state, denoted by p_{DT}:

$$p_{DT} = \int_0^1 p_6 2t dt = \frac{T(N_1)}{N_1} \int_0^1 2t\, W(A(t))\, dt\ .$$

In summary, the average channel access probability of a node in the network is the chance of becoming a transmitter in the three mutually exclusive broadcast or unicast states (BT, UT or DT), which is given by

$$\begin{aligned} q_{HAMA} &= q_{NAMA} + p_u(p_{UT} + p_{DT}) \\ &= T(N_2) + U(N_1) \cdot \left(\frac{T(N_1)}{N_1} \int_0^1 2t\, W(A(t))\, dt \right. \\ &\quad \left. + \int_0^1 2t\, \frac{W(N_2 - S(t))\, W(S(t))}{S(t)}\, dt \right)\ . \end{aligned} \tag{12}$$

The above analyses for HAMA have made four simplifications. Firstly, we assumed that the number of two-hop neighbors also follows Poisson distribution, just like that of one-hop neighbors. Secondly, we let $N_2 - S(t) \geq 0$ even though N_2 may be smaller than $S(t)$ when the transmission range r is small. Thirdly, only one neighbor j is considered when making node i to become a unicast transmitter in the DT or the UT state, although node i may have multiple chances to do so owning to other one-hop neighbors. The results of the simulation experiments reported in Section V validate these approximations.

D. PAMA

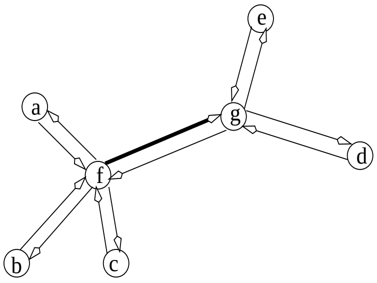

Figure 14. Link Activation in PAMA.

In PAMA, a link is activated only if the link has the highest priority among the incident links of the head and the tail of the link. For example, in Fig. 14, link (f, g) is activated only if it has the highest priority among the links with f and g as the heads or tails.

To analyze the channel access probability of a node in PAMA, we simplify the problem by assuming that the one-hop neighbor sets of the one-hop neighbors of a given node are disjoint (*i.e.*, any two-hop neighbor of a node is reachable through a single one-hop neighbor only). Using the simplification, the sizes of the two-hop neighbor sets become identical independent random variables following Poisson distribution with mean N_1, so as to avoid handling the correlation between the sizes of the two-hop neighbor sets.

Suppose that a node i has $k_1 \geq 1$ one-hop neighbors. The probability that the node is eligible for transmission is $k_1/2k_1 = 1/2$ because the node has $2k_1$ incident links, and k_1 of them are outgoing. Further suppose that link (i, j) out of the k_1 outgoing links has the highest priority, then node i is able to activate link (i, j) if link (i, j) also has the highest priority among the links incident to node j. Denote the number of one-hop neighbors of node j by k_2. Then the probability of link (i, j) having the highest priority among the incident links of node j is a conditional probability, based on the fact that link (i, j) already has the highest priority among the incident links of node i.

We denote the conditional probability of link (i, j) having the highest priority among the incident links of node j as $P\{A \mid B\}$, where A is the event that link (i, j) wins among the $2k_2$ incident links of node j, and B is the event that link (i, j) wins among the $2k_1$ incident links of node i. We have:

$$P\{B\} = \frac{1}{2k_1},\ P\{A \cap B\} = \frac{1}{2k_1 + 2k_2}\ ,$$

$$P\{A \mid B\} = \frac{P\{A \cap B\}}{P\{B\}} = \frac{k_1}{k_1 + k_2}\ .$$

Therefore, the condition of node i being able to transmit is that node i has an outgoing link (i, j) with the highest priority, of which the probability is $\frac{1}{2}$, and that link (i, j) has the highest priority among the incident links of node j, of which the probability is $\frac{k_1}{k_1+k_2}$. Considering all possible values of random variables k_1 and k_2, which follow the Poisson distribution, we have:

$$\begin{aligned} q_{PAMA} &= \sum_{k_1=1}^{\infty} \frac{N_1^{k_1}}{k_1!} e^{-N_1} \frac{1}{2} \sum_{k_2=0}^{\infty} \frac{N_1^{k_2}}{k_2!} e^{-N_1} \frac{k_1}{k_1 + k_2} \\ &= \frac{N_1}{2} (e^{-2N_1} + T(2N_1)). \end{aligned} \tag{13}$$

q_{PAMA} is the upper bound of the channel access probability of a node in PAMA, because if we have not assumed that the one-hop neighbor sets of the head and tail of a link are disjoint, the number of one-hop neighbors of the tail of the activated link, k_2, could have started from a larger number than 0 in the expressions above, and the actual channel access probability in PAMA would be less than q_{PAMA}.

E. LAMA

In LAMA, a node can activate an outgoing link only if the node has the highest priority among its one-hop neighbors, as well as among its two-hop neighbors reachable through the tail of the outgoing link. For convenience, we make the same assumption as in the analysis of PAMA that the one-hop neighbor sets of the one-hop neighbors of a given node are disjoint.

Similarly, suppose a node i has k_1 one-hop neighbors, and the number of the two-hop neighbors reachable through a one-hop neighbor j is k_2. The probability of node i winning in its one-hop neighbor set N_i^1 is $1/(k_1 + 1)$. The probability of node i winning in the one-hop neighbor set of node j is $(k_1 + 1)/(k_1 + k_2 + 1)$, which is conditional upon the fact that node i already wins in N_i^1, and is derived in the same way as in the PAMA analysis. Because k_2 is a random variable following the Poisson distribution,

$$p_7 = \sum_{k_2=0}^{\infty} \frac{N_1^{k_2}}{k_2!} e^{-N_1} \frac{k_1 + 1}{k_1 + k_2 + 1}$$

is the average conditional probability of node i activating link (i, j). Besides node j, node i has other one-hop neighbors. If node i has the highest priority in any one-hop neighbor set of its one-hop neighbors, node i is able to transmit. Therefore, the probability of node i being able to transmit is

$$p_8 = 1 - (1 - p_7)^{k_1}.$$

Because k_1 is also a random variable following the Poisson distribution, the channel access probability of node i in LAMA is:

$$p_9 = \sum_{k_1=1}^{\infty} \frac{N_1^{k_1}}{k_1!} e^{-N_1} \frac{1}{k_1 + 1} p_8\ .$$

When k_1 increases, p_8 edges quickly towards the probability limit 1. Since we are only interested in the upper bound of channel access probability in LAMA, assuming $p_8 = 1$ simplifies the calculation of p_9 and provides a less tight upper bound. Let $p_8 = 1$, the upper bound of channel access probability in

LAMA is thus:

$$q_{LAMA} = \sum_{k_1=1}^{\infty} \frac{N_1^{k_1}}{k_1!} e^{-N_1} \frac{1}{k_1+1} = T(N_1) \tag{14}$$

F. *Comparison among NAMA, HAMA, PAMA and LAMA*

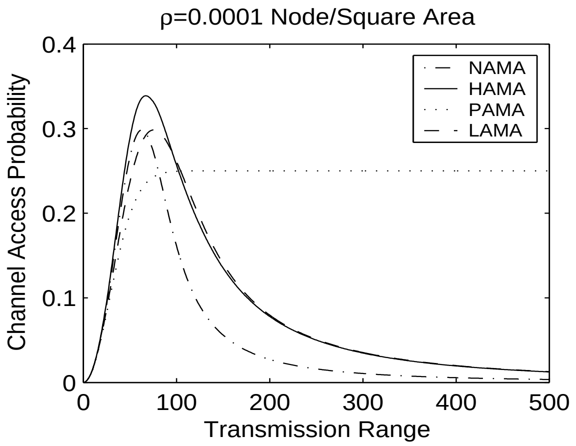

Figure 15. Channel access probability of NAMA, HAMA, PAMA and LAMA.

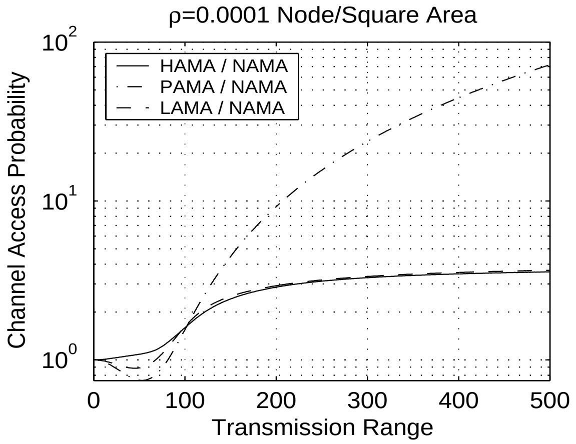

Figure 16. Channel access probability ratio of HAMA, PAMA and LAMA to NAMA.

Assuming a network density of $\rho = 0.0001$, equivalent to placing 100 nodes on a 1000×1000 square plane, the relation between transmission range and the channel access probability of a node in NAMA, HAMA, PAMA and LAMA is shown in Figure 15, based on Eq. (11), Eq. (12), Eq. (13) and Eq. (14), respectively.

Because a node barely has any neighbor in a multihop network when the node transmission range is too short, Figure 15 shows that the system throughput is close to none at around zero transmission range, but it increases quickly to the peak when the transmission range covers around one neighbor on the average, except for that of PAMA, which is an upper bound. Then network throughput drops when more and more neighbors are contacted and the contention level increases.

Figure 16 shows the performance ratio of the channel access probabilities of HAMA, PAMA and LAMA to that NAMA. At shorter transmission ranges, HAMA, PAMA and LAMA performs very similar to NAMA, because nodes are sparsely connected, and node or link activations are similar to broadcasting. When transmission range increases, HAMA, LAMA and PAMA obtains more and more opportunities to leverage its unicast capability and the relative throughput also increases more than three times that of NAMA. HAMA and LAMA perform very similarly.

G. *Comparison with CSMA and CSMA/CA*

Because the analyses about NAMA and PAMA are more accurate than the analyses of PAMA and LAMA, which simply derive the upper bounds, we only compare the throughput of HAMA and NAMA that of the idealized CSMA and CSMA/CA protocols, which are analyzed in [16] and [15]. We consider only unicast transmissions, because CSMA/CA does not support collision-free broadcast.

Scheduled access protocols are modeled differently from CSMA and CSMA/CA. In time-division scheduled channel access, a time slot can carry a complete data packet, while the time slot for CSMA and CSMA/CA only lasts for the duration of a channel round-trip propagation delay, and multiple time slots are used to transmit a data packet once the channel is successfully acquired. In addition, Wang *et al.* [15] and Wu *et al.* [16] assumed a heavily loaded scenario in which a node always has a data packet during the channel access, which is not true for the throughput analyses of HAMA and NAMA, because using the heavy load approximation would always result in the maximum network capacity according to Eq. (7).

The probability of channel access at each time slot in CSMA and CSMA/CA is parameterized by the symbol p'. For comparison purposes, we assume that *every attempt* to access the channel in CSMA or CSMA/CA is an *indication* of a packet arrival at the node. Though the attempt may not succeed in CSMA and CSMA/CA due to packet or RTS/CTS signal collisions in the common channel, and end up dropping the packet, conflict-free scheduling protocols can always deliver the packet if it is offered to the channel. In addition, we assume that no packet arrives during the packet transmission. Accordingly, the traffic load for a node is equivalent to the portion of time for transmissions at the node. Denote the average packet size as l_{data}, the traffic load for a node is given by

$$\lambda = \frac{l_{data}}{1/p' + l_{data}} = \frac{p' l_{data}}{1 + p' l_{data}}$$

because the average interval between successive transmissions follows Geometric distribution with parameter p'.

The network throughput is measured by the successful data packet transmission rate within the one-hop neighborhood of a node in [15] [16], instead of the whole network. Therefore, the comparable network throughput in HAMA and NAMA is the sum of the packet transmissions by each node and all of its one-hop neighbors. We reuse the symbol N in this section to represent the number of one-hop neighbors of a node, which is the same as N_1 defined in Section IV-A. Because every node is assigned the same load λ, and has the same channel access probability (q_{HAMA}, q_{NAMA}), the throughput of HAMA and

NAMA becomes

$$S_{HAMA} = N \cdot \min(\lambda,\ q_{HAMA})\ .$$

$$S_{NAMA} = N \cdot \min(\lambda,\ q_{NAMA})\ .$$

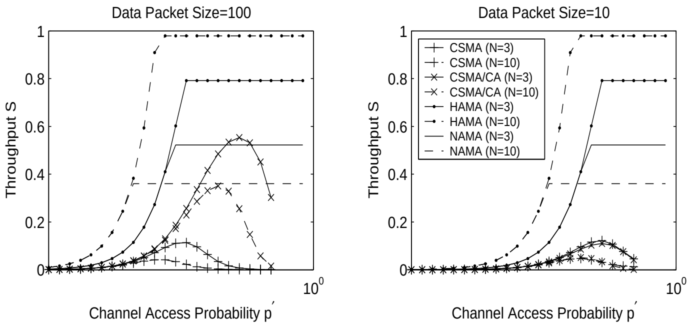

Figure 17. Comparison between HAMA, NAMA and CSMA, CSMA/CA.

Figure 17 compares the throughput attributes of HAMA, NAMA, the idealized CSMA [16], and CSMA/CA [15] with different numbers of one-hop neighbors in two scenarios. The first scenario assumes that data packets last for $l_{data} = 100$ time slots in CSMA and CSMA/CA, and the second assumes a 10-time-slot packet size average.

The network throughput decreases when a node has more contenders in NAMA, CSMA and CSMA/CA, which is not true for HAMA. In addition, HAMA and NAMA provide higher throughput than CSMA and CSMA/CA, because all transmissions are collision-free even when the network is heavily loaded. In contrast to the critical role of packet size in the throughput of CSMA and CSMA/CA, it is almost irrelevant in that of scheduled approaches, except for shifting the points of reaching the network capacity.

V. Simulations

The delay and throughput attributes of NAMA, LAMA, PAMA and HAMA are studied by comparing their performance with UxDMA [13] in two simulation scenarios: fully connected networks with different numbers of nodes, and multihop networks with different radio transmission ranges.

In the simulations, we use the normalized *packets per time slot* for both arrival rates and throughput. This metric can be translated into concrete throughput metrics, such as *Mbps* (megabits per second), if the time slot sizes and the channel bandwidth are instantiated.

Because the channel access protocols based on NCR have different capabilities regarding broadcast and unicast, we only simulate unicast traffic at each node in all protocols. All nodes have the same load, and the destinations of the unicast packets at each node are evenly distributed over all one-hop neighbors.

In addition, the simulations are guided by the following parameters and behavior:

- The network topologies remain static during the simulations to examine the performance of the scheduling algorithms only.
- Signal propagation in the channel follows the free-space model and the effective range of the radio is determined by the power level of the radio. Radiation energy outside the effective transmission range of the radio is considered negligible interference to other communications. All radios have the same transmission range.
- Each node has an unlimited buffer for data packets.
- 30 pseudo-noise codes are available for code assignments, *i.e.*, $|C_{pn}| = 30$.
- Packet arrivals are modeled as Poisson arrivals. Only one packet can be transmitted in a time slot.
- The duration of the simulation is 100,000 time slots, long enough to collect the metrics of interests.

We note that assuming static topologies does not favor NCR-based channel access protocols or UxDMA, because the same network topologies are used. Nonetheless, exchanging the full topology information required by UxDMA in a dynamic network would be far more challenging that exchanging the identifiers of nodes within two hops of each node.

Except for HAMA, which schedules both node- and link-activations, UxDMA has respective constraint sets for NAMA, LAMA and PAMA. Table II gives the corresponding constraint sets for NAMA, LAMA and PAMA.

TABLE II
CONSTRAINT SETS FOR NCR-BASED PROTOCOLS.

Protocol	Entity	Constraint Set
UxDMA-NAMA	Node	$\{V_{tr}^0, V_{tt}^1\}$
UxDMA-LAMA	Link	$\{E_{rr}^0, E_{tr}^0\}$
UxDMA-PAMA	Link	$\{E_{rr}^0, E_{tt}^0, E_{tr}^0, E_{tr}^1\}$

The meaning of each symbol is illustrated by Figure 18. Using the solid dots as transmitters, and the circles as receivers, node constraint V_{tr}^0 forbids a node from transmitting and receiving at the same time, and V_{tt}^1 eliminates hidden terminal problem and direct interference. Using wide lines as activated lines, and thin lines as interferences, link constraints E_{rr}^0, E_{tt}^0 and E_{tr}^0 restrict concurrent receptions, concurrent transmissions and simultaneous transmission and reception at a single node, respectively. Constraint E_{tr}^1 prevents hidden terminal problem in link activation scheme.

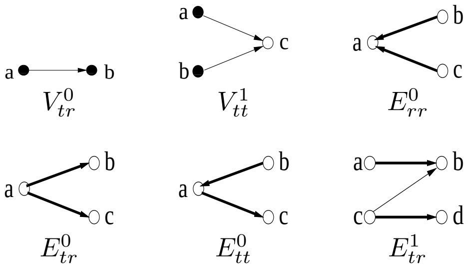

Figure 18. Constraints used by UxDMA for channel access scheduling.

Simulations were carried out in four configurations in the fully connected scenario: 2-, 5-, 10-, 20-node networks, to manifest the effects of different contention levels. Figure 19 shows the maximum throughput of each protocol in fully-connected

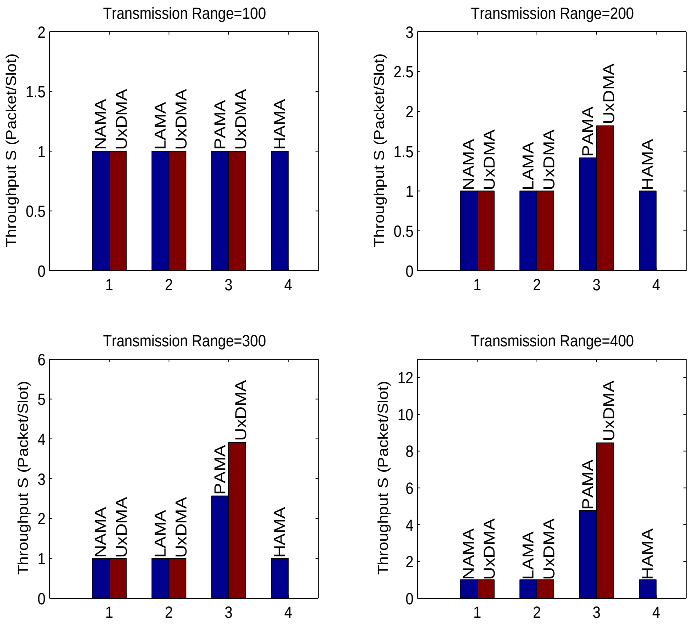

Figure 19. Packet throughput in fully-connected networks

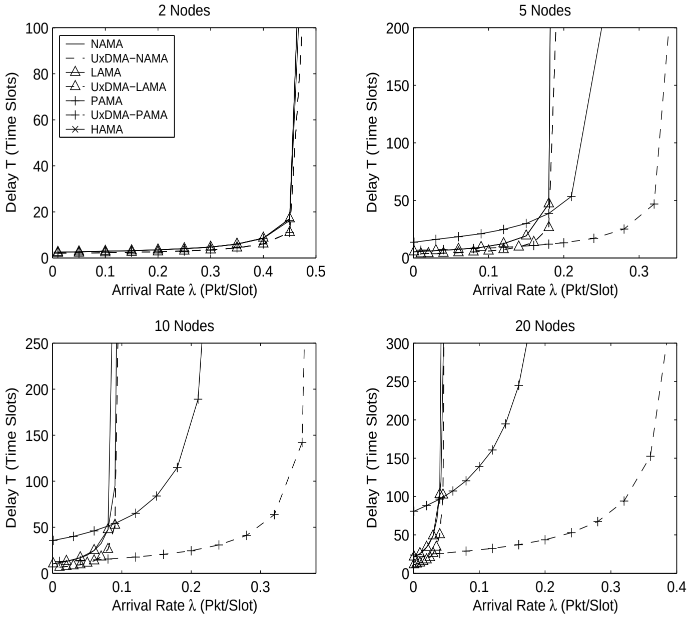

Figure 20. Average packet delays in fully-connected networks

networks. Except for PAMA and UxDMA-PAMA, the maximum throughput of every other protocol is one because their contention resolutions are based on the node priorities, and only one node is activated in each time slot. Because PAMA schedules link activations based on link priorities, multiple links can be activated on different codes in the fully-connected networks, and the channel capacity is greater in PAMA than in the other protocols.

Figure 20 shows the average delay of data packets in NAMA, LAMA and PAMA with their corresponding UxDMA counterparts, and HAMA with regard to different loads on each node in fully-connected networks. NAMA, UxDMA-NAMA, LAMA, UxDMA-LAMA and HAMA have the same delay characteristic, because of the same throughput is achieved in these protocols. PAMA and UxDMA-PAMA can sustain higher loads and have longer "tails" in the delay curves. However, because the number of contenders for each link is more than the number of nodes, the contention level is higher for each link than for each node. Therefore, packets have higher starting delay in PAMA than other NCR-based protocols.

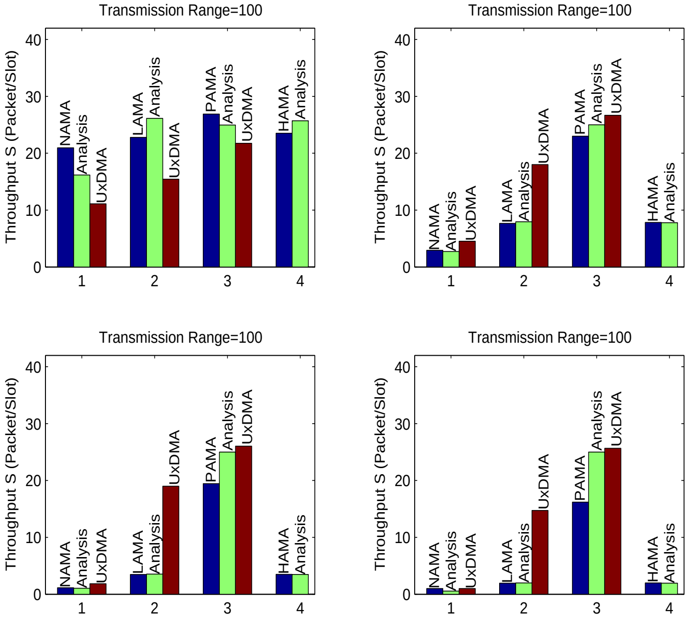

Figure 21. Packet throughput in multihop networks

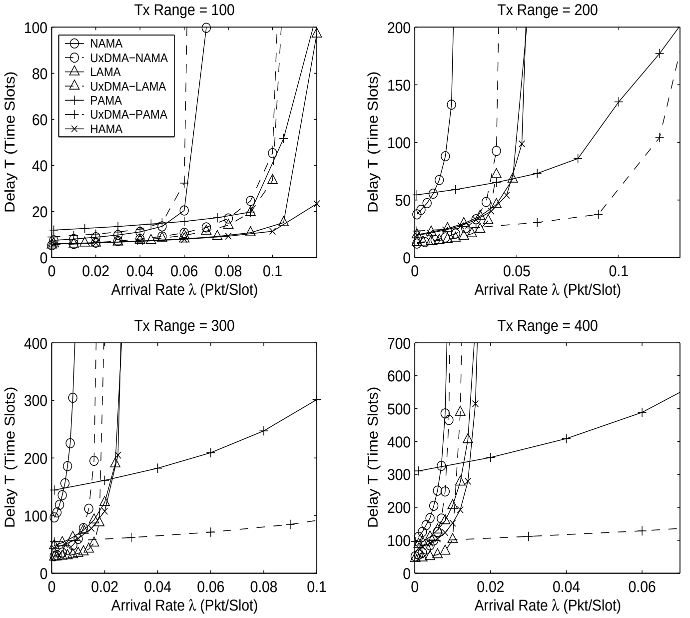

Figure 22. Average packet delays in multihop networks

Figure 21 and 22 show the throughput and the average packet delay of NAMA, LAMA, PAMA, HAMA and the UxDMA variations.

Except for the ad hoc network generated using transmission range one hundred meters in Figure 21, UxDMA always outperforms its NCR-based counterparts — NAMA, LAMA and PAMA at various levels. For example, UxDMA-NAMA is only slightly better than NAMA in all cases, and UxDMA-PAMA is 10-30% better than PAMA. LAMA is comparatively the worst, with much lower throughput than its counterpart UxDMA-LAMA. One interesting point is the similarity between the throughput of LAMA and HAMA, which has been shown by Figure 17 as well, even though they have different code assignment schemes and transmission schedules. Especially, the network throughput of NAMA, LAMA, PAMA and HAMA based on Eq. (7) and the analyses in Section IV is compared with the corresponding protocols in the simulations. The analytical results fits well with the simulations results. Note that the analysis bars with regard to PAMA and LAMA are the upper bounds, although the analysis of LAMA is very close to the simulation results.

In Figure 22, PAMA still gives higher starting point to delays than the other two even when network load is low due to similar reasons as in fully connected scenario. However, PAMA appears to have slower increases when the network load goes larger, which explains the higher spectrum and spatial reuse of the common channel by pure link-oriented scheduling.

VI. Conclusion

We have introduced a new approach to contention resolution that eliminates much of the complexity of prior collision-free scheduling approaches by using two-hop neighborhood information to dynamically determine which node should be allowed to transmit in each collision-resolution context. Based on this approach and time-division channel access scheme, four protocols were introduced for both node-activation and link-activation channel access scheduling in packet radio networks. The advantages of the protocols are that (a) they do not need the contention phases or schedule broadcasts, as adopted by many other channel access scheduling algorithms; (b) they only need the local topology information within two hops, as opposed to other schedule broadcasting algorithms that require complete network topology. We have provided analyses to some of the protocols and compared them with the random-access protocols. The performance of these protocols was also compared with the static scheduling algorithms, which require the global topology information.

REFERENCES

[1] L. Bao and J.J. Garcia-Luna-Aceves. Transmission Scheduling in Ad Hoc Networks with Directional Antennas. In *Proc. ACM Eighth Annual International Conference on Mobile Computing and networking*, Atlanta, Georgia, USA, Sep. 23-28 2002.

[2] D. Bertsekas and R. Gallager. *Data Networks, 2nd edition.* Prentice Hall, Englewood Cliffs, NJ, 1992.

[3] I. Chlamtac, A. Farago, and H. Zhang. Time-spread multiple-access (TSMA) protocols for multihop mobile radio networks. *IEEE/ACM Transactions on Networking*, 6(5):804–12, Dec. 1997.

[4] I. Cidon and M. Sidi. Distributed assignment algorithms for multihop packet radio networks. *IEEE Transactions on Computers*, 38(10):1353–61, Oct 1989.

[5] B.P. Crow, I. Widjaja, L.G. Kim, and P.T. Sakai. IEEE 802.11 Wireless Local Area Networks. *IEEE Communications Magazine*, 35(9):116–26, Sept 1997.

[6] A. Ephremides and T.V. Truong. Scheduling broadcasts in multihop radio networks. *IEEE Transactions on Communications*, 38(4):456–60, Apr. 1990.

[7] S. Even, O. Goldreich, S. Moran, and P. Tong. On the NP-completeness of certain network testing problems. *Networks*, 14(1):1–24, Mar. 1984.

[8] M. Joa-Ng and I.T. Lu. Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, pages 776–83, New York, NY, USA, Mar. 21-25 1999.

[9] J.H. Ju and V.O.K. Li. An optimal topology-transparent scheduling method in multihop packet radio networks. *IEEE/ACM Transactions on Networking*, 6(3):298–306, Jun. 1998.

[10] L. Kleinrock and F.A. Tobagi. Packet switching in radio channels. I. Carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, COM-23(12):1400–16, Dec 1975.

[11] R. Krishnan and J.P.G. Sterbenz. An Evaluation of the TSMA Protocol as a Control Channel Mechanism in MMWN. Technical report, BBN Technical Memorandum No. 1279, 2000.

[12] T. Makansi. Trasmitter-Oriented Code Assignment for Multihop Radio Net-works. *IEEE Transactions on Communications*, 35(12):1379–82, Dec. 1987.

[13] R. Ramanathan. A unified framework and algorithm for channel assignment in wireless networks. *Wireless Networks*, 5(2):81–94, 1999.

[14] R. Ramaswami and K.K. Parhi. Distributed scheduling of broadcasts in a radio network. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, volume 2, pages 497–504, Ottawa, Ont., Canada, Apr. 23-27 1989. IEEE Comput. Soc. Press.

[15] Y. Wang and J.J. Garcia-Luna-Aceves. Performance of Collision Avoidance Protocols in Single-Channel Ad Hoc Networks. In *Proc. of IEEE International Conference on Network Protocols (ICNP)*, Paris, France, Nov. 12-15 2002.

[16] L. Wu and P. K. Varshney. Performance Analysis of CSMA and BTMA Protocols in Multihop Networks (I), Single Shannel Case. *Information Sciences*, 120(1-4):159–177, 1999.

[17] C. Zhu and M.S. Corson. A five-phase reservation protocol (FPRP) for mobile ad hoc networks. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, volume 1, pages 322–31, San Francisco, CA, USA, Mar. 29-Apr. 2 1998.