Contents

List of Figures

List of Tables

Receiver-Oriented Multiple Access in Ad Hoc Networks with Directional Antennas [*]

Lichun Bao and J.J. Garcia-Luna-Aceves
Computer Engineering Department
School of Engineering, BE 399, University of California
Santa Cruz, CA 95064
Email: {baolc, jj}@soe.ucsc.edu

Abstract

Directional antennas can adaptively select radio signals of interest in specific directions, while filtering out unwanted interference from other directions. A couple of medium access protocols based on random access schemes have been proposed for networks with directional antennas, using the omnidirectional mode for the transmission or reception of control packets in order to establish directional links. We propose a distributed receiver-oriented multiple access (ROMA) scheduling protocol, capable of utilizing multi-beam forming directional antennas in ad hoc networks. Unlike random access schemes that use on-demand handshakes or signal scanning to resolve communication targets, ROMA computes a link activation schedule in each time slot using two-hop topology information. It is shown that significant improvements on network throughput and delay can be achieved by exploiting the multi-beam forming capability of directional antennas in both transmission and reception. The performance of ROMA is studied by simulation, and compared with a well-know static scheduling scheme that is based on global topology information.

Keywords

Ad hoc networks, directional antenna, channel access scheduling, multi-beam adaptive array (MBAA).

1 Introduction

An omnidirectional transmission spreads the electromagnetic energy of the signal over a large regions of space, while only a very small portion is actually received by the intended station. This limits system performance and capacity due to multipath fading, delay spread, and co-channel interference (CCI) [16]. Currently, the availability of low-cost computing capacity and the development of new algorithms for processing signals from arrays of simple antennas have made such “smart” directional antennas possible for wireless communication systems [2]. By actively controlling the temporal pacing between the radiating elements of an antenna array with the digital signal processing (DSP) component, directional antennas can enhance or cancel out the radiating electromagnetic waves in certain directions. In this way, radio propagation energy is concentrated in specific directions from the standpoint of the transmitter. Similarly, the receiver can enhance the sensitivity of the antenna in certain directions, thus eliminating many of the multipath effects and co-channel interference (CCI). With M antenna elements, an antenna array generally provides an increased antenna gain of M plus a diversity gain against multipath fading [12] [16]. When a constant signal gain is maintained along the direction of interest and the nulls are adjusted toward the sources of interference so as to reject CCI, it can dramatically increase the performance characteristics of a wireless system in its capacity, coverage, and quality. Based on more complex DSP technologies than the directional antennas that are capable of forming a single beam, an antenna array, called multi-beam adaptive array (MBAA) capable of forming multiple beams for several simultaneous receptions or transmissions, can even enlarge the capacity of the networks by many folds [15].

In ad hoc networks, communicating stations move in unexpected directions. When mobile nodes are equipped with directional antennas for both transmission and reception, the media access control (MAC) protocols face two problems:

---

[*] This work was supported in part by Advanced Technology Office of the Defense Advanced Research Projects Agency (DARPA) under grant No. DAAD19-01-C-0026, and by the U.S. Air Force/OSR under grant No. F49620-00-1-0330.

1. How to track the directional positions of mobile neighbor stations in order to point antenna beams.
2. How to couple neighboring stations for concurrent transmissions and receptions, given that every node has multiple neighbors and each node may intend to either transmit or receive.

Medium access protocols adapt one of the following schemes: on-demand and scheduled. An on-demand channel access scheme determines the communicating pair by exchanging short control signals before each transmission session. The omnidirectional mode of the antenna is usually utilized during the signal exchange period to allow the detection of neighbor intentions and their angular positions [10].

Zander [17] and Ward [15] presented channel access protocols based on slotted ALOHA and directional antennas with single-beam and multiple-beam forming capabilities, respectively. Data packets are transmitted in omnidirectional fashion and are received in directional mode. A special preamble is added to each packet for signal detection and beam orientation at the receivers. In contrast, Ko *et al.* [5] and Nasipuri *et al.* [7] presented carrier-sense multiple access with collision avoidance (CSMA/CA) schemes in which the transmitters use the directional mode of the antennas for transmitting request-to-send (RTS) signals and receiving the corresponding clear-to-send (CTS) replies. The receiver antennas stay in omnidirectional mode for both RTS and CTS. Nasipuri *et al.* proposed to utilize the switched multi-beam forming capability of the directional antennas for establishing communicating pairs [7].

Because omnidirectional transmission and reception are susceptible to interference and collisions, these on-demand access protocols eventually degrade from severe control packet collisions when the channel access demand increases. Using a technique that caches the angle-of-arrival (AoA) information, Takai *et al.* [13] partly eliminated the dependency on the omnidirectional mode of directional antennas, and only fell back to the omnidirectional mode if the AoA profile is not available.

On the other hand, scheduled access schemes prearrange or negotiate a set of timetables for individual nodes or links, such that the communicating nodes couple with each other accordingly, and the transmissions are collision-free in the time and frequency axes. The computation of an optimal channel access schedule given the complete topology information of the ad hoc network has long been known to be an NP-hard problem in graph theory [3] [4] [11]. Ramanathan [9] provided a unified framework, called UxDMA, for time-, frequency- or code-division multiple access channel assignment using polynomial steps. Obviously, collecting the complete topology of the network and distributing the corresponding schedule pose a major challenge for applying UxDMA in ad hoc networks.

We propose a new channel access protocol based on a link activation scheme, which we call Receiver-Oriented Multiple Access (ROMA), to fully utilize the multiple-beam forming capability of MBAA antennas. Section 2 introduces assumptions and relevant terminology for ad hoc networks with MBAA antennas. Section 3 specifies ROMA. Unlike most random access protocols for directional antennas that form only a single beam, both transmissions and receptions are carried out in the directional mode of the antennas in ROMA. ROMA adopts the neighbor-aware contention resolution algorithm (NCR) proposed by Bao and Garcia-Luna-Aceves [1] to derive channel access schedules for a node. According to NCR, each entity among a group of contending entities knows its direct and indirect contenders to a shared resource. Contention to the shared resource is resolved in each context (e.g., a time slot) according to the priorities assigned to the entities based on the context number and their respective identifiers. The entities with the highest priorities among their contenders are elected to access the common resource without conflicts. In ROMA, the channel is time-slotted, and the contention context is identified by the time slot number.

Section 4 presents the neighbor protocol and time division scheme for topology maintenance. To allow nodes to find their neighbors, periodic time slots are allocated to a neighbor protocol, which is in charge of maintaining the two-hop topology information for each node and detecting the location of each neighbor by sending out short signals using the omnidirectional mode of the antenna.

Section 5 addresses the performance of ROMA and compares it against UxDMA by simulation experiments. ROMA offers four key advantages over prior approaches to the channel access problem. First, ROMA allows both transmitters and receivers to use the directional mode of the antenna, instead of requiring one end of the communication to stay in omnidirectional mode, as adopted by random access schemes. Second, ROMA relies on the local topology information within two hops for computing the channel access schedules, in contrast to the reliance on global topology information in UxDMA. Third, ROMA evenly splits nodes in the network into transmitters and receivers in each time slot, which are then paired together for the maximum throughput. Whereas UxDMA allocates each link only one time slot for activation per time frame, ROMA may activate the link multiple times during the same period. Fourth, ROMA is capable of scheduling link activations for MBAA antennas, which have never been handled before in ad hoc networks.

2 Network Assumptions

2.1 Directional Antenna System

Dipole or isotropic antennas propagate radio frequency (RF) energy equally in horizontal or spherical directions. In contrast, directional antennas install multiple antenna elements so that individual omnidirectional RF radiations from these elements interfere constructively or destructively with each other in space, and the signal strength is increased in one or multiple directions. Antenna gain measures the increase of signal strength in those directions in decibels over either a dipole (dBd) or a theoretical isotropic (dBi) antenna. Relative to the center of the antenna pattern, the angle of the directions where the radiated power drops to one-half the maximum value of the lobe is defined as the antenna beamwidth, denoted by β in this paper. The beamwidth can be as narrow as 5° to 10° [14]. Due to the fast fading outside of the lobe, the beamwidth is approximately the effective range of the transmission.

With the advance of silicon and DSP technologies, practical DSP modules in a directional antenna system with multiple antenna elements, called multi-beam adaptive array (MBAA), are implemented, which combine more than one set of weights to form several antenna beam-patterns simultaneously [15]. Smaller parasitic side-lobes exist around an MBAA antenna, which are harmful interference to the desired signals. The side lobes can be steered toward areas without nodes owing to the adaptability of the directional antenna beams. The receivers may steer the nulls as well to filter out harmful interference. For simplicity, side lobes are omitted from discussions for the rest of the paper. Note that because radio reception and transmission are reciprocal, any directivity pattern achievable for reception is also achievable for transmission.

We consider the use of MBAA antennas in ad hoc networks. When used in an ad hoc network, an MBAA antenna can successfully receive and transmit one or more overlapping packets at the same time by pointing its beams toward individual packet directions, while annulling all other unwanted directions. The number of beams that an MBAA antenna is capable of forming is denoted by K.

We assume that an MBAA antenna is also capable of broadcast, using the omnidirectional mode of the antenna. Broadcasting capability is useful in mobile ad hoc networks for control information propagation and neighbor-direction findings. Using an electrically steerable switched-parasitic antenna array, we assume that an MBAA antenna is able to detect the precise angular location of a single radiating source in about one hundred microseconds [8]. On carefully chosen frequencies or power levels, respective transmissions in the omnidirectional and directional modes of an MBAA antenna are controlled to reach approximately the same distance.

A directional antenna may either transmit or receive data packets during a given time, but not both.

Figure 1 illustrates two data communication sessions using MBAA antennas. The solid lines indicate the RF radiation beams, while the dotted lines indicate reception beams of the receivers. The arrows point to the directions of the data flows. Node d is transmitting two separate data packets to nodes b and c, respectively. In scheduled channel access protocols, node b may still orient its reception beam to node a, even though node a has no packet to transmit.

2.2 Network Topology

We assume that each node in the network is assigned a unique ID number and mounted with an MBAA antenna. The topology of a packet radio network is represented by a directed graph $G = (V, E)$, where V is the set of nodes, and E is the set of directional links between nodes ($E \subseteq V \times V$). If a link (u, v) belongs to E, they can be activated when node u directs its transmission beam toward node v, and node v points its reception beam toward node u. Node u and v are called *one-hop neighbors* to each other. Regarding link (u, v), node u is called the *head* of the link, while node v is the *tail*. A link (u, v) always has a companion link (v, u) in the opposite direction. The set of one-hop neighbors of a node u is denoted by N_u^1.

Every link of the network has a weight that reflects the data flow demand over the link, and is determined dynamically by the head of the link, which monitors traffic demands or receives bandwidth requests from the upper-layer applications. The weight of a link (u, v) is denoted by (u, v).`wt`. To prevent instability in the channel access schedules due to frequent link weight changes, the weight values are limited to the values in the set $\{0,\ 1,\ 2,\ 3\}$. A link with weight 0 can never be activated, whereas a link with weight 3 gets the most share of the channel as we will discuss in the specification of ROMA.

We use the notation in Table 1 for the specification of ROMA. At any time slot t, the antenna at node i is either in transmission mode (Tx) or reception mode (Rx), while the state of a link (u, v) is active (ACT) or inactive (INACT). If (u, v).`state` is ACT, node u may transmit a data packet to node v using the main lobe of the directional antenna.

Each node $i \in V$ maintains angular profiles of its one-hop neighbors for antenna-beam orientation purposes. For simplicity, the nodes in the network are assumed to be placed on a flat plane. The horizon seen by a node is evenly divided into $360°/\frac{\beta}{2}$ = $720°/\beta$ segments, and every two continuous segments define one group. A group corresponds to the coverage of a directional beam from the node, and a segment determines the minimum angular separation of two neighbors for receiving non-interfering individual antenna beams. Consequently, $720°/\beta$ groups are identified. Each one-hop neighbor j of a node i belongs to two groups that overlap at node j. The set of angular groups to which a one-hop neighbor j of node i belongs to is denoted by A_i^j.

For example, in Figure 2, the set of the angular groups for link (a, b) is $A_a^b = \{1,\ 2\}$, for link (a, c) is $A_a^c = \{2,\ 3\}$, and for link (a, d) is $A_a^d = \{3,\ 4\}$.

Based on the above definitions, the attributes of a one-hop neighbor j of a node i can now be represented by the tuple: (j, $(i, j).\mathtt{wt}$, $(j, i).\mathtt{wt}$, A_i^j). The attributes of a neighbor is used for contention resolution. Every node is required to promptly propagate its one-hop neighbor information to all of its one-hop neighbors whenever the attributes of a neighbor change, which is handled by the neighbor protocol described in Section 4.

Last but not least, we assume that time is synchronized on all mobile nodes to such a precision that the time difference between any pair of one-hop neighbors does not exceed the maximum signal propagation delay between the one-hop neighbors. Time synchronization can be achieved by a physical-layer protocol attaching the real-time clock information to data packets before transmissions, and aligning time slots to the latest starting point of a complete packet received [6].

3 ROMA

3.1 Contention Modeling

As Figure 3 illustrates, a channel access protocol has to consider four types of contention in multihop wireless networks: transmissions should not cause interference to other communication sessions (Figure 3 (a)), each transmission can convey only one packet (Figure 3 (b)), each reception accepts only one packet (Figure 3 (c)), and a node cannot transmit and receive at the same time (Figure 3 (d)).

The hidden-terminal problem in networks with directional antennas is illustrated in Figure 4, in which link (i, j) and (u, v) are simultaneously activated. Interference happens at node v because both radiation lobes from node i and node u cover node v. When node v orients its reception lobe to node u, it accidently becomes sensitive to the signals from node i as well.

The other type of hidden-terminal interference comes from the side lobes of irrelevant communication sessions at the receivers. However, because the receivers can adaptively adjust their reception beams to nullify the sources of the side lobes, we do not consider the harmful effect from side lobes. Therefore, Figure 4 is the only situation in which the hidden-terminal problem happens, and node i is responsible for avoiding the problem, because both nodes u and v are in the one-hop neighborhood of node i, and node i has complete knowledge of the situation.

3.2 Specification

Nodes and links are assigned priorities based on their identifiers and the current time slot. When the current time slot is t, the priority of a node i is computed by

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

where the sign $\oplus$ is designated to carry out the bit-wise concatenation operation on its operands, and has lower order than other operations. Function $\mathtt{Hash}(x)$ is a fast pseudo-random number generator that produces an unsigned integer message digest of the input bit stream x. The identifier of node i is appended to the result to distinguish the priority from those of other nodes.

The priority of a link $(u, v) \in E$ is computed by

$$(u, v).\mathtt{prio} = (i.\mathtt{prio} \ \mathbf{mod}\ 2) \oplus (\mathtt{Hash}(u \oplus v \oplus t) \cdot (u, v).\mathtt{wt}) \oplus u \oplus v \ , \tag{2}$$

which uses the same hashing function and distinguishing feature as that of the node-priority computation. The variable $(u, v).\mathtt{wt}$ denotes the weight of link (u, v), and is discussed subsequently.

ROMA is a link-activation receiver-oriented multiple access protocol that exploits the multi-beam forming capability of MBAA antennas. Figure 5 specifies ROMA using C-style pseudo code. In essence, ROMA has to decide the active incoming links of each node in reception mode before the actual link activations at the transmitters.

The priority of each node is derived per time slot, and then used to decide the mode of the node according to the oddity of the node priority (lines 1-7). If $k.\texttt{prio}$ is odd, node k is a transmitter for the current time slot; otherwise, node k is in reception mode. As a result, nodes are randomly separated into two classes. It is possible that a node and all its one-hop neighbors are put into the same class, such that the node can neither transmit or receive. Lines 8-15 break the stalemate by converting the mode of the node into the opposite state if the node has the highest priority in its one-hop neighborhood. Then, up to K active incoming links with the highest priorities at each node in reception mode are computed in the one-hop neighborhood of node i (lines 16-28). Besides, some of its incoming links of nodes in reception mode are deactivated if the links cause direct interference (lines 23-25), as shown in Figure 3 (c).

If node i is in transmission mode, it needs to determine the active outgoing links to its one-hop neighbors (lines 31-33) according to the results of lines 16-28. Furthermore, node i needs to avoid activating multiple links in the same angular group (lines 35-38), and avoid causing any hidden-terminal problem to its one-hop neighbors (lines 39-42).

If node i is a receiver, node i may orient its antenna beams toward the one-hop neighbors in the incoming link set (lines 44-45). Otherwise, node i may select up to K outgoing links for transmissions using MBAA antennas according to traffic scheduling criteria (lines 46-47).

The computation of link priorities is carried out as follows:

- The oddity of a node is prepended to the link priority (term ($k.\texttt{prio} \bmod 2$) in Eq. (2), and **ROMA** line 19). This operation differentiates the transmitters converted from reception-mode nodes (**ROMA** lines 12-14) against transmitters computed by regular means (**ROMA** lines 3-4), such that incoming links in the latter case always have higher priorities than those in the former case. The converted transmitters may join the active incoming links of the receiver (**ROMA** lines 26-27), only when the transmitters derived from regular means cannot fulfill the reception capacity of the receiving MBAA antennas.
- The priority of a link (u, v) is proportional to its weight, $w^t_{(u,v)}$ (Eq. (2), and **ROMA** line 20). Even though the weight of a link ranges over only four integer values, the bandwidth allocations change dramatically according to the different weight values. For instance, given that three links (x, i), (y, i) and (z, i) have weight $(x, i).\texttt{wt} = 1$, $(y, i).\texttt{wt} = 2$ and $(z, i).\texttt{wt} = 3$, and only one incoming link of node i can be activated at a given time, the bandwidth allocations to the three links are $\frac{1}{3} \cdot \frac{1}{3} = 11\%$, $\frac{1}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{2} = 28\%$ and $\frac{1}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{2} + \frac{1}{3} = 61\%$ of the total incoming bandwidth at node i, respectively, because of the differences in the link priority ranges. When carefully chosen, the limited number of weight values in ROMA can satisfy wide ranges of the bandwidth demands. However, the choice of link weights depends on the traffic requirements in the application layer, and is outside the scope of this paper.

Figure 6 illustrates the operation of ROMA in a sample network with MBAA antennas capable of forming up to three antenna beams. Nodes denoted by solid circles indicate the nodes are in Tx mode (transmitter), and nodes denoted by empty circles indicate that they are in Rx mode (receiver). Arrows leading into each receiver are the incoming links chosen by the receiver for activation. Lobes depicted by solid lines indicates the traffic needs from the transmitters. However, because node b detects hidden-terminal contention at node c incurred by node a and node b itself, link (b, d) is not activated (dashed lobe). On the other hand, node g is ready to receive from node f, but node f has no traffic for node g, and link (f, g) is not activated, either (dashed lobe).

3.3 Correctness

Lemma 1 *ROMA is live in each time slot.*

Proof: In each time slot, the receiving or transmitting mode of a node is determined by two criteria, (a) the priority of the node and (b) the modes of its one-hop neighbors. If the mode depends on criterion (a) alone (ROMA lines 1-7), there is positive probability that all nodes in the network be assigned the same mode, which causes deadlock in channel access. Criterion (b) allows some nodes to flip their modes to break the deadlock (ROMA lines 8-15). Because ROMA requires receiving nodes to listen to the top K incoming links, regardless of the modes of the heads for those links (ROMA lines 16-28), there are always active links in the network at any time slot. $\square$

Lemma 2 *ROMA is fair.*

Proof: The fairness of ROMA is achieved by two means: the even division of transmitting and receiving nodes, and weight-proportional link activation.

By Lemma 1, ROMA separates nodes into transmitters and receivers with equal probability so that the activation probabilities of nodes are the same.

In ROMA lines 17-21, the priorities of links are multiplied by their respective weights so that links with greater weights have higher probabilities to be selected by the receivers for activation. And all links have positive probabilities to be activated because of the randomness of the link priorities. □

4 Neighbor Protocol

4.1 Random Access with Signals

In ad hoc networks, the two-hop neighbor information needed by topology-dependent scheduling protocols is acquired by each node propagating its one-hop neighbor states. However, exchanging neighborhood information among known and unknown neighbors cannot take advantage of the dynamic collision-free scheduling mechanisms described so far, because those mechanisms assume a-priori knowledge of the neighborhood. Hence, neighborhood information needs to be transmitted over a common channel on a best-effort basis using the omnidirectional mode of the directional antennas. The neighbor protocol relies on an additional time section for coordinating neighbor information.

Figure 7 shows that the additional time section is inserted after every T_{sched} scheduled-access time slots, and lasts for T_{nbr} time slots. In addition, every time slot for random access is subdivided into a number of smaller time segments, called *signal slots*, for transmitting short signals, each containing up to a couple of hundreds of bytes.

Signals are used by the neighbor protocol for two purposes. One is for a node to say "hello" to its one-hop neighbors periodically in order to maintain connectivity. The other is to send neighbor updates when a neighbor is added, deleted or needs to be refreshed. If a new link is established, both ends of the link need to notify their one-hop neighbors of the new link, and exchange their complete one-hop neighbor information with each other. The weight of a new link is initialized to one.

If a link breaks, a neighbor-delete update needs to be sent out, which is indicated by zero weights assigned to both the incoming and outgoing links with the neighbor. For robustness, an existing neighbor connection also has to be refreshed periodically to the one-hop neighbors. If a neighbor-delete update is not delivered to some one-hop neighbors, those neighbors age out the obsolete link after a period of time.

4.2 Signal Transmission Scheduling

In order to keep inter-nodal connectivity current, each node broadcasts a signal packet on a common-code channel periodically. To avoid such periodic transmissions from synchronizing with one another, which would result in undue collisions of signal packets, the neighbor protocol adds random jitters to the interval value between signal packet transmissions. However, because of the randomness of signal packet transmissions, it is still possible for a signal sent by a node to collide with signals sent by some of its two-hop neighbors. Due to the lack of acknowledgments in signal transmissions, multiple retransmissions are needed for a node to reassure the delivery of the same message to its one-hop neighbors.

Retransmissions of a signal packet can only achieve a certain probability of delivery without acknowledgments. Even though the message delivery probability approaches one as the neighbor protocol sends out the same message in signals repetitively for extended period of time, the neighbor protocol has to regulate the rhythm of sending signals, so that the desired probability of the message delivery is achieved with a small number of retransmissions in a short time, while incurring a little amount of interference to other neighbors' signal transmissions.

We analyze the time interval and the number of retransmissions needed to achieve a certain probability of message delivery by broadcasting signals.

For simplicity, denote the number of neighbors within two hops by N, the retransmission interval in terms of the number of signal slots by T, the number of retransmissions by n, and the desired probability of message delivery by p. After a period of time during which the neighbor protocol has been in operations, we assume that the signal slots chosen by two-hop neighbors to transmit signals are uniformly distributed over the time interval T. Therefore, the success probability of a transmission is $(1 - 1/T)^N$. When a single message is retransmitted for n times, the probability p of at least one successful delivery to all one-hop neighbors satisfies the following formula:

$$1 - \left(1 - \left(1 - \frac{1}{T}\right)^N\right)^n = p$$

which gives

$$n = \frac{\ln(1-p)}{\ln\left(1 - \left(1 - \frac{1}{T}\right)^N\right)} \; . \tag{3}$$

Hence, the duration of the required retransmissions is represented by the function:

$$f(T) = T \cdot n = \frac{T \ln(1-p)}{\ln\left(1 - \left(1 - \frac{1}{T}\right)^N\right)} \; . \tag{4}$$

Because a signal needs to be statistically delivered to one-hop neighbors as soon as possible, the parameter T should be chosen such that $f(T)$ is minimal for a given N and p. Let $f'(T) = 0$, we obtain

$$\frac{1}{\ln(1 - \left(1 - \frac{1}{T}\right)^N)} \cdot \frac{N(1-\frac{1}{T})^N}{1 - \left(1 - \frac{1}{T}\right)^N} \cdot \frac{1}{T-1} = -1 \; , \tag{5}$$

which becomes independent of p.

To find out the relation between T and N from Eq. (5), Eq. (3) and (4) are plotted in the left and right diagrams of Figure 8, respectively, when the required message delivery probability is $p = 0.99$. As shown in the figure, the minimum number and duration of retransmissions required to achieve the desired probability of message delivery are not constant, but vary depending on the interval T chosen to send signals. However, the lowest point on each curve happens at $T \approx 1.44N$, which suggests an approximately *linear relation* between parameter T and N for achieving the desired probability within the shortest time. If we let $t = 1 - \frac{1}{T}$ to simplify the expression, Eq. (5) becomes:

$$Nt^N = (1 - t^N)\left(1 - \frac{1}{t}\right)\ln(1 - t^N) \; .$$

The monotony of the two sides of the equation can be examined if we let

$$\begin{cases} g(t) = Nt^N \; , \\ h(t) = (1 - t^N)\left(1 - \dfrac{1}{t}\right)\ln(1 - t^N) \; , \end{cases}$$

and take the derivatives of the two functions. Because $g(t)$ monotonically increases ($g'(t) > 0$), and $h(t)$ monotonically decreases ($h'(t) < 0$), there is only one root $t = t_0$ to the equation $g(t) = h(t)$. Because function $t = 1 - \frac{1}{T}$ is 1-1 mapping, Eq. (5) also has only one root $T = T_0$ expressed by parameter N. Therefore, there is only one minimal point on each curve of the right diagram of Figure 8.

Assume that N is large, and $T \approx kN$, Eq. (5) becomes

$$\frac{1}{\ln(1 - e^{-1/k})} \cdot \frac{Ne^{-1/k}}{1 - e^{-1/k}} \cdot \frac{1}{kN} + 1 \approx 0 \; ,$$

which can be solved using numeric estimation, and gives $k \approx 1.44$. This reinforces the conjecture that $T \approx 1.44N$, meaning that when the signal transmission interval is 1.44 times the number of neighbors within two hops, the time required to statistically deliver a message to all one-hop neighbors becomes the shortest.

Substituting $T \approx 1.44N$ ($N \gg 1$) in Eq. (3), we obtain:

$$n = \frac{\ln(1-p)}{\ln(1 - \left(1 - \frac{1}{1.44N}\right)^N)} \approx \frac{\ln(1-p)}{\ln(1 - e^{-\frac{1}{1.44}})} \; ,$$

$$n = 1.45 \ln \frac{1}{1-p} \ , \tag{6}$$

which is a function of p only. When $p = 0.99$, $n = 6.7$.

When N is small, a more detailed linear relation between T and N has to be considered, which is $T = 1.44N + 1.55$, derived from the minimum points in the right diagram of Figure 8. Substituting $T = 1.44N + 1.55$ in Eq. (3) and plotting n against N, it appears that n monotonically increases with N. In practice, n takes the derived value from Eq. (6) regardless of N, while T takes value $T = 1.44N + 1.55$ if N is small ($N < 20$) or $T = 1.44N$ otherwise, thus preserving the desired probability of message delivery.

For instance, using the above results, if a node has $N = 20$ neighbors within two hops, then the signal packet interval is set to $T = 1.44N = 29$ signal slots, and the same message has to be retransmitted for $n = 7$ times to achieve 0.99 delivery rate. Accordingly, the duration of the retransmissions is $f(T) = nT \approx 194$ signal slots, matching the result in the right diagram of Figure 8.

The interval values have been based on signal slots. As we stated in Section 4.1, every T_{sched} time slots for scheduled access are followed by T_{nbr} time slots for random access to send signals. Therefore, the latency of delivering a message with the desired probability depends on three factors: (a) the duration of regular time slots and signal slots, (b) the portion of time for random access, and (c) the channel bandwidth. Because the duration of regular time slots and signal slots are determined by the bandwidth and the sizes of packets carried in these slots, independent of the neighbor protocol, we assume the signal-slot duration to be a constant and denote it by t_s. Then, the portion of random-access sections for achieving a desired latency L for message delivery satisfies:

$$\frac{T_{nbr}}{T_{nbr} + T_{sched}} = \frac{Tnt_s}{L} \ .$$

The more neighbors a node has, the longer the interval value T is set for signal retransmissions and the more the portion of time needed for random access. For instance, if the neighbor protocol is to handle the following requirements in an ad hoc network with moderate node-density, such as the number of two-hop neighbors is $N = 20$, the signal slot lasts $t_s = 1ms$, the desired message delivery probability is $p = 0.99$ and the latency requires $L = 2s$, then the portion of time for random access overhead should be set in practice equal to

$$\frac{T_{nbr}}{T_{nbr} + T_{sched}} = \frac{1.44N \cdot 1.45 \ln \frac{1}{1-p} \cdot t_s}{L} = 9.6\% \ . \tag{7}$$

4.3 Mobility Handling

Because of the direction sensitivity in ROMA, the neighbor protocol needs to promptly update the one-hop neighbor locations soon after nodes move, so that the next round of channel access scheduling is free of errors. Therefore, the random access section should be allocated as frequently as possible for better responsiveness of the neighbor protocol. For example, if the random access section is allocated every second, the neighbor protocol needs $100ms$ for neighbor information update purpose, using the result in Eq. (7). Because L in Eq. (7) is an upper bound of the latency in delivering a message to all one-hop neighbors at once, the real latency in delivering the neighbor updates can be much lower if we consider that the message can also arrive asynchronously at one-hop neighbors during the process of retransmissions.

5 Performance

5.1 Static Multiple Access Scheduling

Channel assignment problems in the time, frequency and code domains have traditionally been treated as graph coloring problems. The basic characteristic of these channel access schemes is that the schedule is static as long as network topology remains unchanged. Inherently, topology information needs to be collected and frequent schedule broadcasts have to be carried out in mobile networks.

We compare ROMA with the best-known static schedule approximation algorithms that are summarized in a unified framework by Ramanathan [9]. Assuming the global topology of the network, Ramanathan [9] provided a unified algorithm for coloring the nodes or links of the graph in polynomial time.

The constraints on nodes or edges of the graph are represented by eleven atomic relations between nodes or edges. A constraint set characterizes a channel assignment problem on the graph using various technologies, such as TDMA, FDMA or CDMA. However, it did not specify the modeling of constraints in spatial division multiple access (SDMA) scheme. It happens that the only change necessary in UxDMA for SDMA is the procedure for choosing the first available least color. For comparison purposes, we modify the algorithm for searching the first available color in the SDMA scheme such that the color selection process considers angular profiles of one-hop neighbors as well as the maximum number of incident links in the same color.

The number of colors used by UxDMA determines the time frame during which every link is able to access the channel once. When computing the colorings on the graphs in UxDMA, an optimal ordering, PMNF (Progressive Minimum Neighbors First) heuristic, has been applied in each computation so that the colorings "perform quite close to optimum" [9].

An MBAA antenna may only activate K incoming or K outgoing links simultaneously. Therefore, the constraint set in UxDMA in networks with directional antennas is:

$$\{E^0_{tr}, E^0_{rr}, E^0_{tt}, E^1_{tr}\},$$

where E^0_{tr} denotes the self-interference case in Figure 3, and E^1_{tr} represents the hidden-terminal case. E^0_{rr} and E^0_{tt} constrain multiple simultaneous transmission or reception sessions from a node. However, because of the multi-beam capability of the antenna systems, the constraints E^0_{rr} and E^0_{tt} are allowed as long as the number of instances at an antenna does not exceed K.

5.2 Simulation Assumptions

We study the performance of ROMA by running simulations in two scenarios: fully connected networks and multihop networks. Fully connected networks exhibit homogeneous contention situations for each link, while links in multihop networks encounter different levels of contentions because of the variations in node density. The fully connected networks are generated by setting the size of the square plane to 100×100 square meters, and tuning the transmission range of directional antennas to 100 meters, so that every node is reachable from all other nodes. The contention level in fully connected networks is affected only by the number of nodes. We study the performance differences when the network has 5 and 20 nodes. In multihop networks, contention levels for each link are determined not only by the number of nodes in the network, but also by the antenna coverage. We generate the multihop networks by randomly placing 100 nodes within a square plane of 1000×1000 square meters, and set the antenna transmission ranges to 200 and 400 meters, respectively. Because both ROMA and UxDMA can support channel access scheduling with multiple antenna beam activations, directional antennas with one, two and four beams are simulated, respectively, as well. The performance is measured in terms of the network throughput and packet delay of the protocols in each simulation case.

UxDMA is simulated in each scenario with corresponding the constraint parameters as well. Because UxDMA is a static scheduling algorithm, the coloring of links in each scenario is carried out at the beginning of each simulation.

We model the packets arrivals at each node as a Poisson process (packet inter-arrival intervals are exponentially distributed with parameter λ), and packets are served in first-in-first-out (FIFO) order. All nodes have the same packet arrival rate λ. Because every node has equal probability of being activated in UxDMA, the data packets are evenly dispatched onto each outgoing link. In ROMA, each link has different probability of activation depending on the number of contenders of each link, thus the traffic is proportionally distributed to outgoing links according to the activation probability of that link. The simulations are guided by the following parameters and assumptions:

- The beamwidth of directional antennas is 30°.
- Because UxDMA is not capable of dynamic bandwidth allocations, ROMA has the weight of each link fixed to one.
- Antenna beams always have the same transmission range in each simulation scenario. We do not consider power management for communicating with one-hop neighbors at different distances.
- Signal propagation in the channel follows the free-space model and the effective transmission range is determined by the power level of the antenna alone.

- The bandwidth of the radio channel is 2 Mbps. In all simulations, the bandwidths of all links are assigned 1 for simplicity.
- A time unit in the simulation equals one time slot. A time slot last for 8 milliseconds, including guard time, which is long enough to transmit a 2KB packet.
- Only static networks are considered in the simulations, so that the two-hop neighbor information or the entire topology is known beforehand in the corresponding protocols. The networks are generated by randomly placing a number of nodes onto a square plane. To simulate an infinite plane that has constant node placement density, the opposite sides of the square are seamed together, which visually turns the square area into a torus.
- At each node, the number of the memory buffers holding packets for each neighbor is 20. Generally, dropping packets has very minor influence on the system throughput, because there are most likely other fresher data packets waiting when the older packets are dropped, and channel access opportunities are not likely to be wasted. However, we assume an infinite buffer size for simulations using single beam-forming antennas.
- The duration of the simulation is 800 seconds (equal to 100000 time slots), which is long enough to compute the metrics of interests.

5.3 Analysis of Results

Figures 9 and 10 show the throughput and delay attributes of ROMA and UxDMA in fully-connected networks when the number of active antenna beams is one, two, and four. The appended numbers in the legends represent the number of beams that each antenna can form.

Figure 9 shows that the greater number of beams an MBAA antenna can form, the higher throughput the network achieves for both ROMA and UxDMA. Except for the scenarios with one-beam MBAA antenna configurations, ROMA performs better than UxDMA in terms of throughput and delay attributes. It is worth noting that there is no deadlock in ROMA for channel access in all the scenarios, including 2-node network setting. Without the capability of flipping the mode of some nodes to the opposite modes, The 2-node network setting could have only achieve 50% of the channel capacity. The other advantage of ROMA over UxDMA is the separation of nodes into transmitters and receivers with even probability. This feature maximizes the multi-beam activation possibilities of a node for both transmissions and receptions. While UxDMA tends to maximize the transmission capability of a single node, and creates more receivers than transmitters, thus limiting the network throughput overall.

Figure 11 and 12 provides the throughput and delay characteristics of ROMA and UxDMA in multi-hop network scenarios. The performance of ROMA and UxDMA demonstrates similar patterns as shown in fully-connected scenarios. The throughput and delay attributes of ROMA is always better than UxDMA in networks that have 2-beam and 4-beam antennas due to the same reason as stated in fully-connected network scenarios. For 1-beam settings, UxDMA performs better because of its knowledge about global topology information. However, UxDMA underperforms ROMA when the transmission range is 100 meters. This is because the network connection density varies more dramatically in low transmission range. While UxDMA has to adopt a unique time frame size overall where every link is activated only once, ROMA can arrange to have larger link activation interval at dense network areas, and smaller link activation interval in other areas. That is, ROMA can dynamically adjust the link activation frequency in the low transmission range scenario, whereas UxDMA cannot.

In both Figure 10 and 12, when the packet arrival rates become higher than what the network can serve, the packet delay begins to drop at the verge of the network capacity because only fresher packets are kept in the buffers as packets arrive faster.

ROMA demonstrated superior adaptiveness over the link scheduling algorithm (UxDMA) in all simulation scenarios of multihop networks, because two-hop neighbor information is necessary and sufficient to insure collision-freedom in multihop packet radio networks. In addition, ROMA tries to evenly separate network nodes into transmitters and receivers, so that link activations are maximized in each time slot. UxDMA lacks a mechanism to balance transmissions and receptions.

ROMA does have some disadvantages, in that the intervals between successive activations of a single link is non-deterministic, and is governed by a geometric distribution, which renders a large channel-access delay variance. This is an inherent property of channel access schemes when random functions are involved, as seen in any other on-demand channel access protocol. Only global and relatively static scheduling can guarantee bounds on packet delays.

6 Conclusion

We have introduced ROMA, a very efficient distributed channel access scheduling protocol for ad hoc networks with directional antennas that are capable of forming multiple beams to carry out several simultaneous data-communication sessions. ROMA shows superior performance over the best-known polynomial time approximation algorithm (UxDMA) for scheduling in ad hoc networks in terms of the network throughput and packet delay. A novel neighbor protocol was proposed that uses an allocated random access section for sending signals to track neighbor positions for ROMA. The neighbor protocol reliably exchanges neighbor information to synchronize topology information within two hops of each node. The ability of ROMA to achieve collision freedom for channel access using only two-hop topology information is more efficient than in UxDMA with respect to the control overhead incurred by the two approaches.

References

[1] L. Bao and J.J. Garcia-Luna-Aceves. A New Approach to Channel Access Scheduling for Ad Hoc Networks. In *Proc. ACM Seventh Annual International Conference on Mobile Computing and networking*, Rome, Italy, Jul. 16-21 2001.

[2] M. Cooper and M. Goldburg. Intelligent Antennas: Spatial Division Multiple Access. *Annual Review of Communications*, pages 999–1002, 1996.

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

[4] 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.

[5] Y.B. Ko, V. Shankarkumar, and N.H. Vaidya. Medium access control protocols using directional antennas in ad hoc networks. In *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*, volume 1(3), pages 13–21, Tel Aviv, Israel, Mar. 26-30 2000.

[6] L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21(7):558–65, Jul. 1978.

[7] A. Nasipuri, S. Ye, J. You, and R.E. Hiromoto. A MAC Protocol for Mobile Ad Hoc Networks Using Directional Antennas. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, Chicago, IL, Sep. 23-28 2000.

[8] S.L. Preston, D.V. Thiel, T.A. Smith, S.G. O'Keefe, and Jun Wei Lu. Base-station tracking in mobile communications using a switched parasitic antenna array. *IEEE Transactions on Antennas and Propagation*, 46(6):841–4, Jun. 1998.

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

[10] R. Ramanathan. On the Performance of Ad Hoc Networks with Beamforming Antennas. In *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, Long Beach, California, USA, Oct. 4-5 2001.

[11] 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.

[12] J. Razavilar, F. Rashid-Farrokhi, and K.J.R. Liu. Software radio architecture with smart antennas: a tutorial on algorithms and complexity. *IEEE Journal on Selected Areas in Communications*, pages 662–76, Apr. 1999.

[13] M. Takai, J. Martin, and R. Bagrodia. Directional Virtual Carrier Sensing for Directional Antennas in Mobile Ad Hoc Networks. In *Proc. ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, Lausanne, Switzerland, Jun. 9-11 2002.

[14] V. Voipio and P. Vainikainen. Narrowbeam cylindrical antenna array with sparse antenna spacing. In *48th IEEE Vehicular Technology Conference*, volume 1, pages 465–9, Ottawa, Ont., Canada, May 18-21 1998.

[15] J. Ward and Jr. Compton, R.T. High throughput slotted ALOHA packet radio networks with adaptive arrays. *IEEE Transactions on Communications*, 41(3):460–70, Mar. 1993.

[16] J.H. Winters. Smart antennas for wireless systems. *IEEE Personal Communications*, 5(1):23–7, Feb. 1998.

[17] J. Zander. Slotted ALOHA multihop packet radio networks with directional antennas. *Electronics Letters*, 26(25):2098–100, Dec. 6 1990.

Authors' Biography

Lichun Bao Lichun Bao received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 1994, the M.E. degree in computer engineering from Tsinghua University, Beijing, China, in 1997, and the Ph.D. degree in computer science from the University of California, Santa Cruz, CA, in 2002. He joins the Computer Science Department of University of California, Irvine on July, 2003. He was a senior software engineer at Meru Networks, Inc. from 2002 to 2003, involved in the innovation and development of advanced core technologies for corporate wireless networks, based on IEEE 802.11b. Before that, he worked as a research engineer at Cenus Technologies, Inc. from 2000 to 2002. He worked for Sun Labs at Menlo Park, California in 2000. His research interests are wireless network communication theories and technologies, with an emphasis on low-overhead channel access protocols, efficient routing mechanisms, topology managements, and network management in ad hoc networks. He is a student member of the IEEE.

J. J. Garcia-Luna-Aceves J. J. Garcia-Luna-Aceves received the B.S. degree in electrical engineering from the Universidad Iberoamericana, Mexico City, Mexico, in 1977, and the M.S. and Ph.D. degrees in electrical engineering from the University of Hawaii, Honolulu, HI, in 1980 and 1983, respectively. He is Professor of Computer Engineering at the University of California, Santa Cruz (UCSC). He directs the Computer Communication Research Group (CCRG), which is part of the Information Technologies Institute of the Baskin School of Engineering at UCSC.

He has been a Visiting Professor at Sun Laboratories and a consultant on protocol design for Nokia. Prior to joining UCSC in 1993, he was a Center Director at SRI International (SRI) in Menlo Park, California.

Dr. Garcia-Luna-Aceves has published a book and more than 225 refereed papers. He has been Program Co-Chair of ACM MobiHoc 2002 and ACM Mobicom 2000; Chair of the ACM SIG Multimedia; General Chair of ACM Multimedia '93 and ACM SIGCOMM '88; and Program Chair of IEEE MULTIMEDIA '92, ACM SIGCOMM '87, and ACM SIGCOMM '86. He has served in the IEEE Internet Technology Award Committee, the IEEE Richard W. Hamming Medal Committee, and the National Research Council Panel on Digitization and Communications Science of the Army Research Laboratory Technical Assessment Board. He has been on the editorial boards of the IEEE/ACM Transactions on Networking, the Multimedia Systems Journal, and the Journal of High Speed Networks. He received the SRI International Exceptional-Achievement Award in 1985 and 1989, and is a senior member of the IEEE

Table 1: Notation

K	The maximum number of beams formed by an MBAA antenna.
i.`prio`	The priority of node i.
(u, v).`prio`	The priority of link (u, v).
(u, v).`wt`	The weight of link (u, v).
i.`mode`	The antenna mode of node i for either reception or transmission.
Tx	Transmission mode.
Rx	Reception mode.
(u, v).`state`	The activation state of link (u, v).
ACT	Active state.
INACT	Inactive state.
i.`income`	The set of active incoming links to node i in reception mode.
i.`outgo`	The set of active outgoing links from node i in transmission mode.
[statement]	A more complex and yet easy-to-implement operation than an atomic statement, such as a function call.

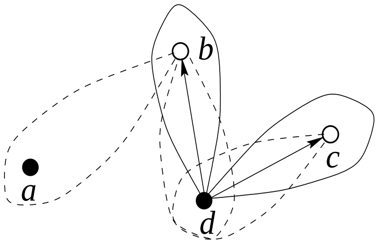


Figure 1: Communications using MBAA antennas.

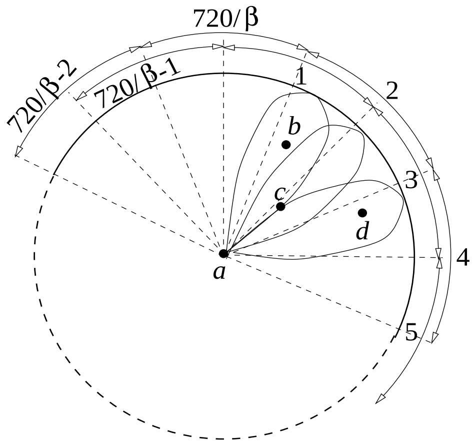
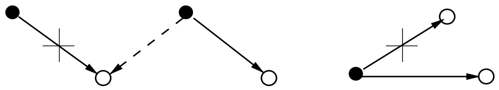


Figure 2: Neighbor grouping based on angular division and antenna patterns.



(a) Hidden Terminal Problem (b) Single Transmission

(c) Direct Interference (d) Half Duplex Only

Figure 3: Contention types.

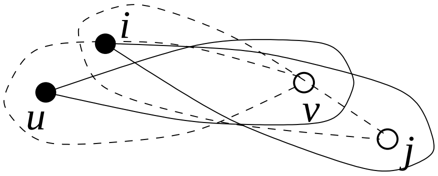


Figure 4: Hidden-terminal problem in directional antenna systems.

```
ROMA($i$, $t$)
{
    /* Priority and Tx/Rx mode assignments. */
    for ($k \in N_i^1 \cup (\bigcup_{j \in N_i^1} N_j^1)$) {
        $k$.prio = Hash($t \oplus k$);
        if ($k$.prio mod 2 $\equiv$ 1)
            $k$.mode = Tx; /* Transmit mode. */
        else
            $k$.mode = Rx; /* Receive mode. */
    }

    /* Break unanimous Tx/Rx tie
       in one-hop neighborhood. */
    for ($j \in N_i^1 \cup \{i\}$) {
        if (($\forall k \in N_j^1 \cup \{j\}$, $k$.mode $\equiv$ Tx and
             ($\forall k \in N_j^1$, $j$.prio $>$ $k$.prio))
               $j$.mode = Rx; /* All transmitters? */
        else if (($\forall k \in N_j^1 \cup \{j\}$, $k$.mode $\equiv$ Rx and
             ($\forall k \in N_j^1$, $j$.prio $>$ $k$.prio))
               $j$.mode = Tx; /* All receivers? */
    }

    /* Compute active incoming links for
       one-hop neighbors in Rx mode. */
    for ($j \in N_i^1 \cup \{i\}$, and $j$.mode $\equiv$ Rx) {
        /* Initialization. */
        for ($k \in N_j^1$) {
            $(k, j)$.state = ACT;
            $(k, j)$.prio = ($k$.prio mod 2)$\oplus$
                    (Hash($k \oplus j \oplus t$)$\cdot$ $(k, j)$.wt)$\oplus k \oplus j$;
        }

        /* Hidden-terminal avoidance. */
        for ($k \in N_j^1$ and $(k, j)$.state $\equiv$ ACT) {
            if ($\exists m \in N_j^1$, $A_j^k \cap A_j^m \neq \emptyset$ and
                $(m, j)$.prio $>$ $(k, j)$.prio)
                $(k, j)$.state = INACT;

        /* Select up to $K$ active incoming links. */
        [ Sort $(k, j)$ according to $(k, j)$.prio
            in descending order, where $k \in N_j^1$ and
            $(k, j)$.state $\equiv$ ACT ];

        $j$.income =
            $\{k \mid k$ belongs to top $K$ of the sorted list$\}$;
    }

    /* Collect active outgoing links. */
    if ($i$.mode $\equiv$ Tx) {
        $i$.outgo = $\emptyset$;

        /* Active outgoing links are the active
           incoming links at one-hop neighbors. */
        for ($j \in N_i^1$)
            if ($j$.mode $\equiv$ Rx and $i \in j$.income)
                $i$.outgo = $i$.outgo $\cup \{j\}$;

        for ($j \in i$.outgo)
            if ( /* Figure 3 (b). */
                $\exists k \in i$.outgo and $A_i^k \cap A_i^j \neq \emptyset$ and
                (Packet to $k$ is earlier than to $j$))
                $i$.outgo = $i$.outgo - $\{j\}$;
            else if ( /* Figure 4. */
                $\exists v \in N_i^1,\ v$.mode $\equiv$ Rx and $A_i^j \cap A_i^v \neq \emptyset$ and
                ($\exists u \in N_i^1 \cap N_v^1,\ u$.mode $\equiv$ Tx and $A_v^i \cap A_v^u \neq \emptyset$))
                $i$.outgo = $i$.outgo - $\{j\}$;
    }

    if ($i$.mode $\equiv$ Rx)
        [ Tune antenna beams to members
            in $i$.income for reception ];
    else if ($i$.outgo $\neq \emptyset$) {
        [ Select up to $K$ members from $i$.outgo
            with the earliest packets, and tune
            antenna beams for transmission ];
} /* End of ROMA. */
```

Figure 5: ROMA Specification.

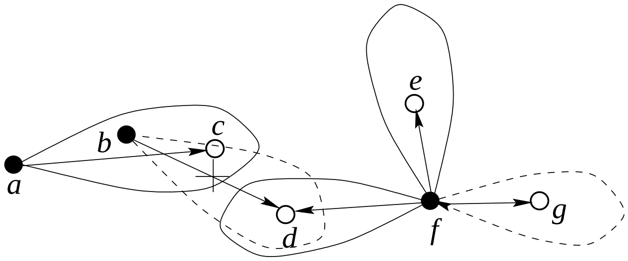


Figure 6: Example of ROMA operation.

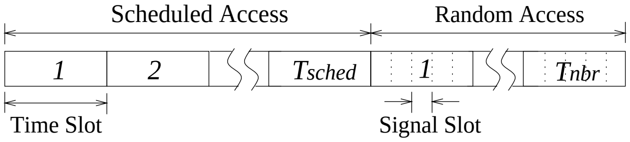


Figure 7: Time division scheme: T_{sched} time slots for scheduled channel access are followed by T_{nbr} time slots for random access to send smaller signal frames.

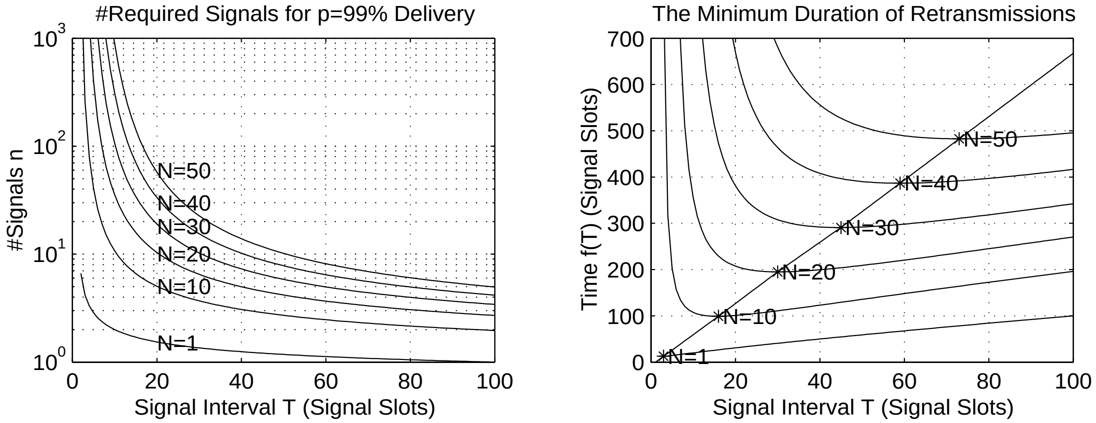


Figure 8: The minimum number of retransmissions and the minimum retransmission duration required to successfully deliver signals with probability $p = 0.99$.

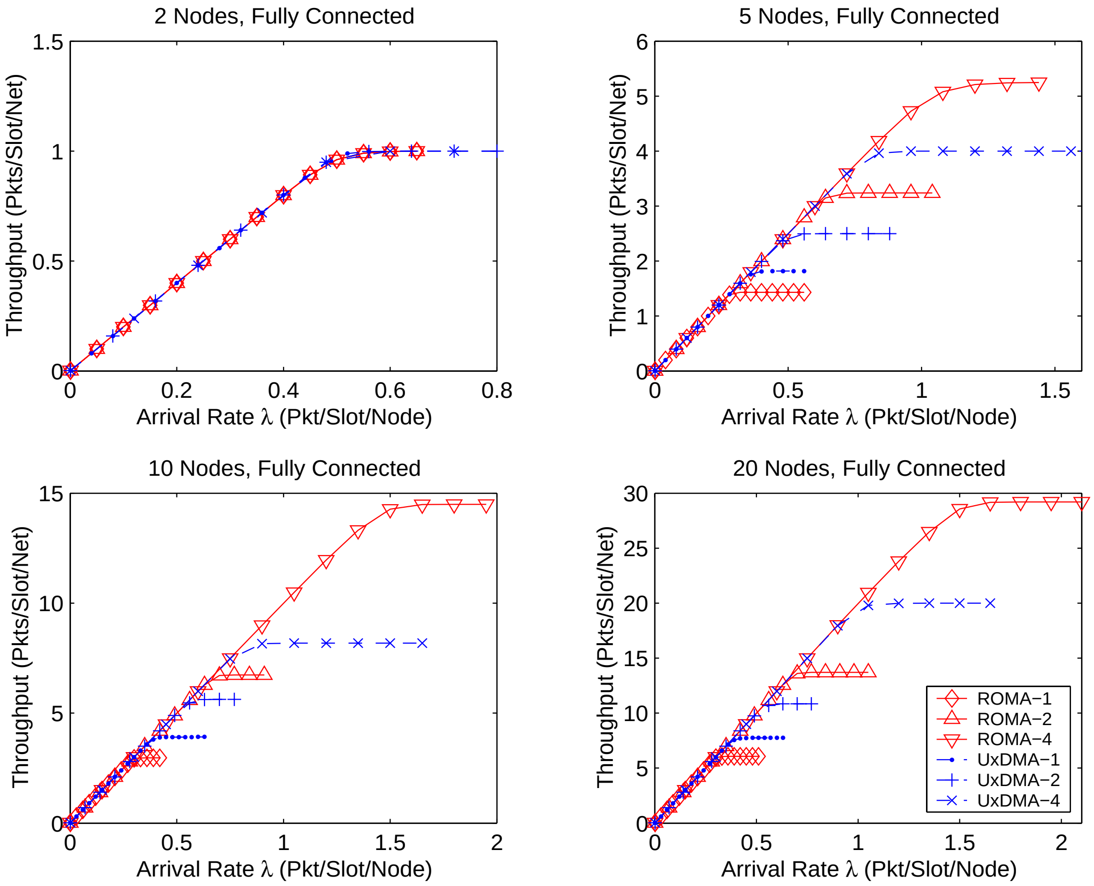


Figure 9: Average packet throughput in fully-connected networks with MBAA antennas having different numbers of beams.

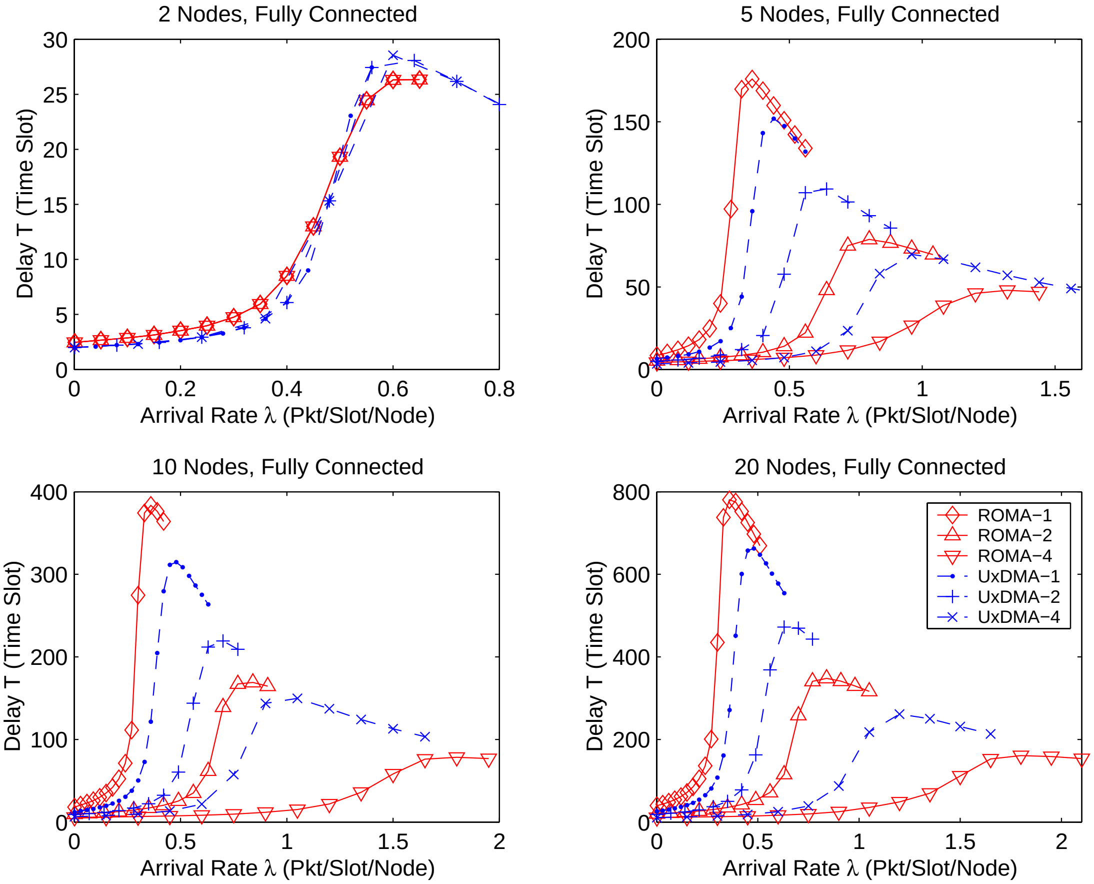


Figure 10: Average packet delay in fully-connected networks with MBAA antennas having different numbers of beams.

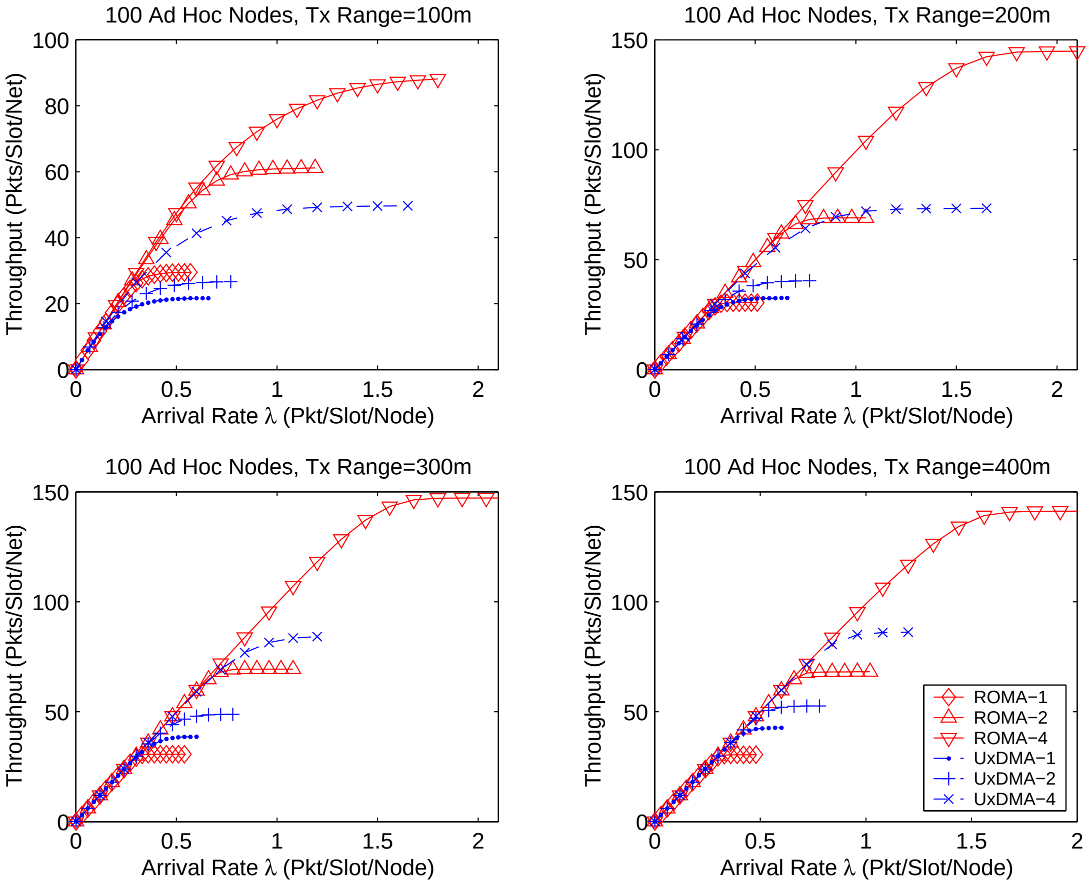


Figure 11: Average packet throughput in multi-hop networks with MBAA antennas having different numbers of beams.

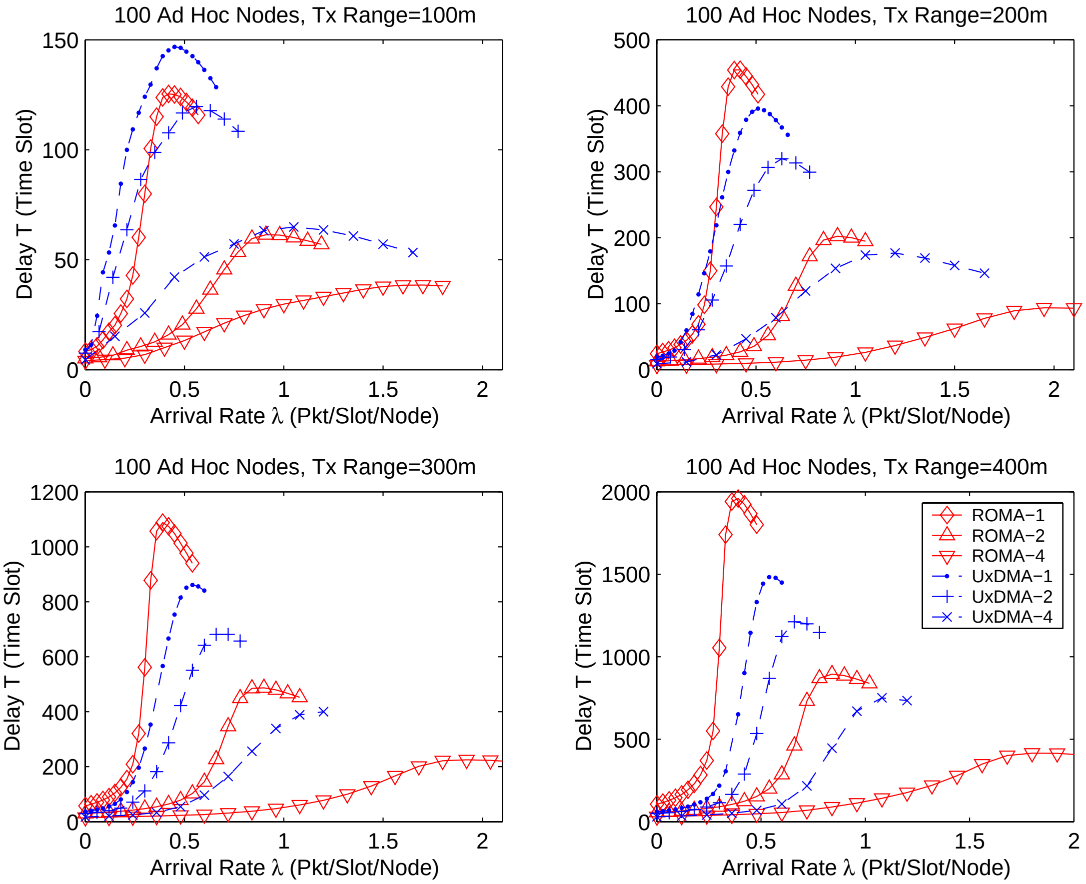


Figure 12: Average packet delay in multi-hop networks with MBAA antennas having different numbers of beams.